# AURIX™ TC35x

## About this document

### Scope and purpose

The Appendix supplies information specific for the TC35x supplementing the family documentation.

# Table of Contents

# 1 Introduction

For Introduction, block diagrams and feature set consult the family document.

For Pinning consult the Data Sheet.

# 2 Memory Maps (MEMMAP)

This is the automatically generated memory map of the TC35x.

## 2.1 Overview

The memory map describes the address locations and access possibilities for the units, memories, and reserved areas as “seen” from the different on-chip buses’ point of view.

## 2.2 Functional Description

The bus-specific address maps describe how the different bus master devices react on accesses to on-chip memories and modules, and which address ranges are valid or invalid for the corresponding buses.

The detailed address mapping of e.g. control registers, SRAM blocks or flash banks/sectors within a module is described in the related module chapter.

The SFI is an uni-directional bridge for access from SPB to SRI and therefore not mentioned here as an SRI master in the Address Map. The SFI is fully transparent and does not include an address translation mechanism.

*Note: In addition to the here described system address map, each TriCore has a TriCore IP internal access to its PSPR via $C000\_0000_H$ and an internal access to its DSPR via $D000\_0000_H$. This additional/private view to the local scratch pad SRAMs is described in the CPU chapter.*

**Table 1** defines the acronyms and other terms that are used in the address maps.

**Table 1 Definition of Acronyms and Terms**

| Term | Description |
|---|---|
| BE | A bus access is terminated with a bus error. |
| ok | A bus access is allowed and is executed. |
| 16 | A bus access with width 16 and 32 bits is allowed and executed. |
| 32 | A bus access with width 32 bits is allowed and executed. |
| Access | A bus access is allowed and is executed. |

### 2.2.1 Segments

This section summarizes the contents of the segments.

#### Segments 0 and 2

These memory segments are reserved.

#### Segments 1 and 3-7

These memory segments allow access to the CPUs Program and Data Scratch Pad SRAM (PSPR, DSPR), Program and Data Cache SRAMs (PCACHE, DCACHE) as well as TAG SRAMs related to Program and Data Cache (PTAG SRAM[1)] and DTAG SRAM[1)]).

Where DCACHE is supported, DCACHE and DTAG SRAM[1)] can be only accessed if the Data Cache is disabled.

PCACHE and PTAG SRAMs[1)] can be only accessed if the related Program Cache is disabled.

1) TAG SRAMs are not meant to be used as general SRAMs and can be accessed only with single data access and only with 64 bit aligned address.

The attribute of these segments (cached / non-cached) can be partially configured[1] for each CPUs data and program side individually (see CPU chapter: Physical Memory Attribute Registers, PMAx).

**Segment 8**

This memory segment allows cached access to PFlash and BROM.

**Segment 9**

This memory segment allows cached access to LMU and to EMEM.

**Segment 10**

This memory segment allows non-cached access to PFlash, DFlash and BROM.

**Segment 11**

This memory segment allows non-cached access to LMU and to EMEM.

**Segment 12**

This memory segment is reserved.

**Segment 13**

This memory segment is reserved.

**Segment 14**

This memory segment is reserved.

**Segment 15**

The lower 128 Mbyte is SPB address space and the upper 128 Mbyte is SRI address space.

## 2.3 Bus Fabric SRI

This is the merged view of all SRI Bus Segments as used in the TC35x.

**Table 2 Address Map as seen by Bus Masters on Bus SRI**

| Address Range | | Size | Unit | Access Type | |
|---|---|---|---|---|---|
| from | to | | | Read | Write |
| $00000000_H$ | $4FFFFFFF_H$ | - | Reserved | BE | BE |
| $50000000_H$ | $50017FFF_H$ | 96 Kbyte | Data ScratchPad RAM (CPU2) | ok | ok |
| $50018000_H$ | $5001BFFF_H$ | 16 Kbyte | Data Cache RAM (CPU2) | ok | ok |
| $5001C000_H$ | $500BFFFF_H$ | - | Reserved | BE | BE |
| $500C0000_H$ | $500C17FF_H$ | 6 Kbyte | Data Cache Tag RAM (CPU2) | ok | ok |
| $500C1800_H$ | $500FFFFF_H$ | - | Reserved | BE | BE |
| $50100000_H$ | $5010FFFF_H$ | 64 Kbyte | Program ScratchPad RAM (CPU2) | ok | ok |
| $50110000_H$ | $50117FFF_H$ | 32 Kbyte | Program Cache RAM (CPU2) | ok | ok |
| $50118000_H$ | $501BFFFF_H$ | - | Reserved | BE | BE |

1) Mapping of Cache and TAG SRAMs is controlled via the MTU register MTU_MEMMAP.

**Table 2 Address Map as seen by Bus Masters on Bus SRI** (cont'd)

| Address Range | | Size | Unit | Access Type | |
|---|---|---|---|---|---|
| from | to | | | Read | Write |
| $501C0000_H$ | $501C2FFF_H$ | 12 Kbyte | Program Cache TAG RAM (CPU2) | ok | ok |
| $501C3000_H$ | $5FFFFFFF_H$ | - | Reserved | BE | BE |
| $60000000_H$ | $6003BFFF_H$ | 240 Kbyte | Data ScratchPad RAM (CPU1) | ok | ok |
| $6003C000_H$ | $6003FFFF_H$ | 16 Kbyte | Data Cache RAM (CPU1) | ok | ok |
| $60040000_H$ | $600BFFFF_H$ | - | Reserved | BE | BE |
| $600C0000_H$ | $600C17FF_H$ | 6 Kbyte | Data Cache Tag RAM (CPU1) | ok | ok |
| $600C1800_H$ | $600FFFFF_H$ | - | Reserved | BE | BE |
| $60100000_H$ | $6010FFFF_H$ | 64 Kbyte | Program ScratchPad RAM (CPU1) | ok | ok |
| $60110000_H$ | $60117FFF_H$ | 32 Kbyte | Program Cache RAM (CPU1) | ok | ok |
| $60118000_H$ | $601BFFFF_H$ | - | Reserved | BE | BE |
| $601C0000_H$ | $601C2FFF_H$ | 12 Kbyte | Program Cache TAG RAM (CPU1) | ok | ok |
| $601C3000_H$ | $6FFFFFFF_H$ | - | Reserved | BE | BE |
| $70000000_H$ | $7003BFFF_H$ | 240 Kbyte | Data ScratchPad RAM (CPU0) | ok | ok |
| $7003C000_H$ | $7003FFFF_H$ | 16 Kbyte | Data Cache RAM (CPU0) | ok | ok |
| $70040000_H$ | $700BFFFF_H$ | - | Reserved | BE | BE |
| $700C0000_H$ | $700C17FF_H$ | 6 Kbyte | Data Cache Tag RAM (CPU0) | ok | ok |
| $700C1800_H$ | $700FFFFF_H$ | - | Reserved | BE | BE |
| $70100000_H$ | $7010FFFF_H$ | 64 Kbyte | Program ScratchPad RAM (CPU0) | ok | ok |
| $70110000_H$ | $70117FFF_H$ | 32 Kbyte | Program Cache RAM (CPU0) | ok | ok |
| $70118000_H$ | $701BFFFF_H$ | - | Reserved | BE | BE |
| $701C0000_H$ | $701C2FFF_H$ | 12 Kbyte | Program Cache TAG RAM (CPU0) | ok | ok |
| $701C3000_H$ | $7FFFFFFF_H$ | - | Reserved | BE | BE |
| $80000000_H$ | $801FFFFF_H$ | 2 Mbyte | Program Flash (PFI0) | ok | ok |
| $80200000_H$ | $802FFFFF_H$ | - | Reserved | BE | BE |
| $80300000_H$ | $804FFFFF_H$ | 2 Mbyte | Program Flash (PFI1) | ok | ok |
| $80500000_H$ | $8FDFFFFF_H$ | - | Reserved | BE | BE |
| $8FE00000_H$ | $8FE7FFFF_H$ | 512 Kbyte | Online Data Acquisition (OLDA) (DOM0) | BE | ok |
| $8FE80000_H$ | $8FFEFFFF_H$ | - | Reserved | BE | BE |
| $8FFF0000_H$ | $8FFFFFFF_H$ | 64 Kbyte | Boot ROM (BROM) (DMU) | ok | ok |
| $90000000_H$ | $9000FFFF_H$ | 64 Kbyte | DLMU RAM (CPU0) | ok | ok |
| $90010000_H$ | $9001FFFF_H$ | 64 Kbyte | DLMU RAM (CPU1) | ok | ok |
| $90020000_H$ | $9002FFFF_H$ | 64 Kbyte | DLMU RAM (CPU2) | ok | ok |
| $90030000_H$ | $9003FFFF_H$ | - | Reserved | BE | BE |
| $90040000_H$ | $9007FFFF_H$ | 256 Kbyte | LMU RAM (LMU0) | ok | ok |
| $90080000_H$ | $900BFFFF_H$ | 256 Kbyte | LMU RAM (LMU1) | ok | ok |
| $900C0000_H$ | $98FFFFFF_H$ | - | Reserved | BE | BE |

**Table 2 Address Map as seen by Bus Masters on Bus SRI** (cont'd)

| Address Range | | Size | Unit | Access Type | |
|---|---|---|---|---|---|
| from | to | | | Read | Write |
| $99000000_H$ | $990FFFFF_H$ | 1 Mbyte | SRI slave interface 0 (access to EMEM module RAM, cached segment) (EMEMRAM0) | ok | ok |
| $99100000_H$ | $991FFFFF_H$ | 1 Mbyte | SRI slave interface 1 (access to EMEM module RAM, cached segment) (EMEMRAM1) | ok | ok |
| $99200000_H$ | $9FFFFFFF_H$ | - | Reserved | BE | BE |
| $A0000000_H$ | $A01FFFFF_H$ | 2 Mbyte | Program Flash (PFI0_NC) | ok | ok |
| $A0200000_H$ | $A02FFFFF_H$ | - | Reserved | BE | BE |
| $A0300000_H$ | $A04FFFFF_H$ | 2 Mbyte | Program Flash (PFI1_NC) | ok | ok |
| $A0500000_H$ | $A7FFFFFF_H$ | - | Reserved | BE | BE |
| $A8000000_H$ | $A8003FFF_H$ | 16 Kbyte | Erase Counter (PFI0) | ok | ok |
| $A8004000_H$ | $A807FFFF_H$ | - | Reserved | BE | BE |
| $A8080000_H$ | $A80FFFFF_H$ | 512 Kbyte | Register address space (PFI0) | ok | ok |
| $A8100000_H$ | $A82FFFFF_H$ | - | Reserved | BE | BE |
| $A8300000_H$ | $A8303FFF_H$ | 16 Kbyte | Erase Counter (PFI1) | ok | ok |
| $A8304000_H$ | $A837FFFF_H$ | - | Reserved | BE | BE |
| $A8380000_H$ | $A83FFFFF_H$ | 512 Kbyte | Register address space (PFI1) | ok | ok |
| $A8400000_H$ | $AEFFFFFF_H$ | - | Reserved | BE | BE |
| $AF000000_H$ | $AF01FFFF_H$ | 128 Kbyte | Data Flash 0 EEPROM (DF0) and Host Command Sequence Interpreter (DMU) | ok | ok |
| $AF020000_H$ | $AF3FFFFF_H$ | - | Reserved | BE | BE |

**Table 2 Address Map as seen by Bus Masters on Bus SRI** (cont'd)

| Address Range | | Size | Unit | Access Type | |
|---|---|---|---|---|---|
| from | to | | | Read | Write |
| $AF400000_H$ | $AF405FFF_H$ | 24 Kbyte | UCB_BMHD0_ORIG (UCB) | ok | ok |
| | | | UCB_BMHD1_ORIG (UCB) | ok | ok |
| | | | UCB_BMHD2_ORIG (UCB) | ok | ok |
| | | | UCB_BMHD3_ORIG (UCB) | ok | ok |
| | | | UCB_SSW (UCB) | ok | ok |
| | | | UCB_USER (UCB) | ok | ok |
| | | | UCB_TEST (UCB) | ok | ok |
| | | | UCB_HSMCFG (UCB) | ok | ok |
| | | | UCB_BMHD0_COPY (UCB) | ok | ok |
| | | | UCB_BMHD1_COPY (UCB) | ok | ok |
| | | | UCB_BMHD2_COPY (UCB) | ok | ok |
| | | | UCB_BMHD3_COPY (UCB) | ok | ok |
| | | | UCB_REDSEC (UCB) | ok | ok |
| | | | Reserved (UCB) | ok | ok |
| | | | Reserved (UCB) | ok | ok |
| | | | UCB_RETEST (UCB) | ok | ok |
| | | | UCB_PFLASH_ORIG (UCB) | ok | ok |
| | | | UCB_DFLASH_ORIG (UCB) | ok | ok |
| | | | UCB_DBG_ORIG (UCB) | ok | ok |
| | | | UCB_HSM_ORIG (UCB) | ok | ok |
| | | | UCB_HSMCOTP0_ORIG (UCB) | ok | ok |
| | | | UCB_HSMCOTP1_ORIG (UCB) | ok | ok |
| | | | UCB_ECPRIO_ORIG (UCB) | ok | ok |
| | | | UCB_SWAP_ORIG (UCB) | ok | ok |
| | | | UCB_PFLASH_COPY (UCB) | ok | ok |
| | | | UCB_DFLASH_COPY (UCB) | ok | ok |
| | | | UCB_DBG_COPY (UCB) | ok | ok |
| | | | UCB_HSM_COPY (UCB) | ok | ok |
| | | | UCB_HSMCOTP0_COPY (UCB) | ok | ok |
| | | | UCB_HSMCOTP1_COPY (UCB) | ok | ok |
| | | | UCB_ECPRIO_COPY (UCB) | ok | ok |
| | | | UCB_SWAP_COPY (UCB) | ok | ok |

**Table 2 Address Map as seen by Bus Masters on Bus SRI** (cont'd)

| Address Range | | Size | Unit | Access Type | |
|---|---|---|---|---|---|
| from | to | | | Read | Write |
| cont'd | | | UCB_OTP0_ORIG (UCB)<br>UCB_OTP1_ORIG (UCB)<br>UCB_OTP2_ORIG (UCB)<br>UCB_OTP3_ORIG (UCB)<br>UCB_OTP4_ORIG (UCB)<br>UCB_OTP5_ORIG (UCB)<br>UCB_OTP6_ORIG (UCB)<br>UCB_OTP7_ORIG (UCB)<br>UCB_OTP0_COPY (UCB)<br>UCB_OPT1_COPY (UCB)<br>UCB_OPT2_COPY (UCB)<br>UCB_OPT3_COPY (UCB)<br>UCB_OPT4_COPY (UCB)<br>UCB_OPT5_COPY (UCB)<br>UCB_OPT6_COPY (UCB)<br>UCB_OPT7_COPY (UCB) | ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok | ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok<br>ok |
| $AF406000_H$ | $AF7FFFFF_H$ | - | Reserved | BE | BE |
| $AF800000_H$ | $AF80FFFF_H$ | 64 Kbyte | Configuration Sector Layout (CFS) | ok | ok |
| $AF810000_H$ | $AFBFFFFF_H$ | - | Reserved | BE | BE |
| $AFC00000_H$ | $AFC1FFFF_H$ | 128 Kbyte | Data Flash 1 EEPROM (DF1) and HSM Command Sequence Interpreter (DMU) | ok | ok |
| $AFC20000_H$ | $AFDFFFFF_H$ | - | Reserved | BE | BE |
| $AFE00000_H$ | $AFE7FFFF_H$ | 512 Kbyte | Online Data Acquisition (OLDA) (DOM0_NC) | BE | ok |
| $AFE80000_H$ | $AFFEFFFF_H$ | - | Reserved | BE | BE |
| $AFFF0000_H$ | $AFFFFFFF_H$ | 64 Kbyte | Boot ROM (BROM) (DMU) | ok | ok |
| $B0000000_H$ | $B000FFFF_H$ | 64 Kbyte | DLMU RAM (CPU0_NC) | ok | ok |
| $B0010000_H$ | $B001FFFF_H$ | 64 Kbyte | DLMU RAM (CPU1_NC) | ok | ok |
| $B0020000_H$ | $B002FFFF_H$ | 64 Kbyte | DLMU RAM (CPU2_NC) | ok | ok |
| $B0030000_H$ | $B003FFFF_H$ | - | Reserved | BE | BE |
| $B0040000_H$ | $B007FFFF_H$ | 256 Kbyte | LMU RAM (LMU0) | ok | ok |
| $B0080000_H$ | $B00BFFFF_H$ | 256 Kbyte | LMU RAM (LMU1) | ok | ok |
| $B00C0000_H$ | $B8FFFFFF_H$ | - | Reserved | BE | BE |
| $B9000000_H$ | $B90FFFFF_H$ | 1 Mbyte | SRI slave interface 0 (access to EMEM module RAM, non-cached segment) (EMEMRAM0) | ok | ok |
| $B9100000_H$ | $B91FFFFF_H$ | 1 Mbyte | SRI slave interface 1 (access to EMEM module RAM, non-cached segment) (EMEMRAM1) | ok | ok |
| $B9200000_H$ | $B93FFFFF_H$ | - | Reserved | BE | BE |
| $B9400000_H$ | $B947FFFF_H$ | 512 Kbyte | Non-Cached XTM Ram address range (SFIBRIDGE2) Bridge to Bus BBB (SFIBRIDGE2) | ok | ok |
| $B9480000_H$ | $F801FFFF_H$ | - | Reserved | BE | BE |
| $F8020000_H$ | $F8027FFF_H$ | 32 Kbyte | sri slave interface (FSIRAM) | ok | ok |

**Table 2 Address Map as seen by Bus Masters on Bus SRI** (cont'd)

| Address Range | | Size | Unit | Access Type | |
|---|---|---|---|---|---|
| from | to | | | Read | Write |
| $F8028000_H$ | $F802FFFF_H$ | - | Reserved | BE | BE |
| $F8030000_H$ | $F80300FF_H$ | 256 byte | sri slave interface (FSI) | ok | ok |
| $F8030100_H$ | $F8037FFF_H$ | - | Reserved | BE | BE |
| $F8038000_H$ | $F803FFFF_H$ | 32 Kbyte | sri slave interface (PMU) | ok | ok |
| $F8040000_H$ | $F807FFFF_H$ | 256 Kbyte | sri slave interface (DMU) | ok | ok |
| $F8080000_H$ | $F80FFFFF_H$ | - | Reserved | BE | BE |
| $F8100000_H$ | $F810FFFF_H$ | 64 Kbyte | Special Function Registers (LMU0) | ok | ok |
| $F8110000_H$ | $F811FFFF_H$ | 64 Kbyte | Special Function Registers (LMU1) | ok | ok |
| $F8120000_H$ | $F86FFFFF_H$ | - | Reserved | BE | BE |
| $F8700000_H$ | $F870FFFF_H$ | 64 Kbyte | sri slave interface (DOM0) | ok | ok |
| $F8710000_H$ | $F87FFFFF_H$ | - | Reserved | BE | BE |
| $F8800000_H$ | $F881FFFF_H$ | 128 Kbyte | Safety Memory Protection Register (CPU0) | ok | ok |
| | | | DLMU Safety Memory Protection registers (CPU0) | ok | ok |
| | | | Safety register protection registers (CPU0) | ok | ok |
| | | | Kernel Reset registers (CPU0) | ok | ok |
| | | | Flash Configuration registers (CPU0) | ok | ok |
| | | | Overlay Block Control registers (CPU0) | ok | ok |
| | | | Memory Integrity Registers (CPU0) | ok | ok |
| | | | Core Special Function Registers (CPU0) | ok | ok |
| | | | General Purpose Registers (CPU0) | ok | ok |
| | | | Memory Protection Registers (CPU0) | ok | ok |
| | | | Temporal Protection System registers (CPU0) | ok | ok |
| | | | Floating point register (CPU0) | ok | ok |
| | | | Core Debug Performance Counter registers (CPU0) | ok | ok |
| | | | Data Memory Interface registers (CPU0) | ok | ok |
| | | | Program Memory Interface registers (CPU0) | ok | ok |
| $F8820000_H$ | $F883FFFF_H$ | 128 Kbyte | Safety Memory Protection Register (CPU1) | ok | ok |
| | | | DLMU Safety Memory Protection registers (CPU1) | ok | ok |
| | | | Safety register protection registers (CPU1) | ok | ok |
| | | | Kernel Reset registers (CPU1) | ok | ok |
| | | | Flash Configuration registers (CPU1) | ok | ok |
| | | | Overlay Block Control registers (CPU1) | ok | ok |
| | | | Memory Integrity Registers (CPU1) | ok | ok |
| | | | Core Special Function Registers (CPU1) | ok | ok |
| | | | General Purpose Registers (CPU1) | ok | ok |
| | | | Memory Protection Registers (CPU1) | ok | ok |
| | | | Temporal Protection System registers (CPU1) | ok | ok |
| | | | Floating point register (CPU1) | ok | ok |
| | | | Core Debug Performance Counter registers (CPU1) | ok | ok |
| | | | Data Memory Interface registers (CPU1) | ok | ok |
| | | | Program Memory Interface registers (CPU1) | ok | ok |

**Table 2 Address Map as seen by Bus Masters on Bus SRI** (cont'd)

| Address Range | | Size | Unit | Access Type | |
|---|---|---|---|---|---|
| from | to | | | Read | Write |
| $F8840000_H$ | $F885FFFF_H$ | 128 Kbyte | Safety Memory Protection Register (CPU2) | ok | ok |
| | | | DLMU Safety Memory Protection registers (CPU2) | ok | ok |
| | | | Safety register protection registers (CPU2) | ok | ok |
| | | | Kernel Reset registers (CPU2) | ok | ok |
| | | | Flash Configuration registers (CPU2) | ok | ok |
| | | | Overlay Block Control registers (CPU2) | ok | ok |
| | | | Memory Integrity Registers (CPU2) | ok | ok |
| | | | Core Special Function Registers (CPU2) | ok | ok |
| | | | General Purpose Registers (CPU2) | ok | ok |
| | | | Memory Protection Registers (CPU2) | ok | ok |
| | | | Temporal Protection System registers (CPU2) | ok | ok |
| | | | Floating point register (CPU2) | ok | ok |
| | | | Core Debug Performance Counter registers (CPU2) | ok | ok |
| | | | Data Memory Interface registers (CPU2) | ok | ok |
| | | | Program Memory Interface registers (CPU2) | ok | ok |
| $F8860000_H$ | $F9FFFFFF_H$ | - | Reserved | BE | BE |
| $FA000000_H$ | $FAFFFFFF_H$ | 16 Mbyte | Non-Cached XTM Ram address range (SFIBRIDGE2) | ok | ok |
| $FB000000_H$ | $FB00FFFF_H$ | 64 Kbyte | EMEM RAM SRI Interface Control Registers (EMEMMPU0) | ok | ok |
| $FB010000_H$ | $FB01FFFF_H$ | 64 Kbyte | EMEM RAM SRI Interface Control Registers (EMEMMPU1) | ok | ok |
| $FB020000_H$ | $FFBFFFFF_H$ | - | Reserved | BE | BE |
| $FFC00000_H$ | $FFC1FFFF_H$ | 128 Kbyte | Data Flash 1 EEPROM (DF1) and HSM Command Sequence Interpreter (DMU) | ok | ok |
| $FFC20000_H$ | $FFFFFFFF_H$ | - | Reserved | BE | BE |

## 2.4 Bus Instance SPB

**Table 3 Address Map as seen by Bus Masters on Bus SPB**

| Address Range | | Size | Unit | Access Type | |
|---|---|---|---|---|---|
| from | to | | | Read | Write |
| $00000000_H$ | $0FFFFFFF_H$ | - | Reserved | BE | BE |
| $10000000_H$ | $EFFFFFFF_H$ | 3584 Mbyte | Redirection of SRI ranges (SFIBRIDGE1)<br>Bridge to Bus Segment 00 of SRI (SFIBRIDGE1) | ok | ok |
| $F0000000_H$ | $F00001FF_H$ | 512 byte | FPI slave interface (FCE) | ok | ok |
| $F0000200_H$ | $F00003FF_H$ | - | Reserved | BE | BE |
| $F0000400_H$ | $F00005FF_H$ | 512 byte | FPI slave interface (CBS) | ok | ok |
| $F0000600_H$ | $F00006FF_H$ | 256 byte | FPI slave interface (ASCLIN0) | ok | ok |
| $F0000700_H$ | $F00007FF_H$ | 256 byte | FPI slave interface (ASCLIN1) | ok | ok |
| $F0000800_H$ | $F00008FF_H$ | 256 byte | FPI slave interface (ASCLIN2) | ok | ok |
| $F0000900_H$ | $F00009FF_H$ | 256 byte | FPI slave interface (ASCLIN3) | ok | ok |

**Table 3 Address Map as seen by Bus Masters on Bus SPB** (cont'd)

| Address Range | | Size | Unit | Access Type | |
|---|---|---|---|---|---|
| from | to | | | Read | Write |
| $F0000A00_H$ | $F0000FFF_H$ | - | Reserved | BE | BE |
| $F0001000_H$ | $F00010FF_H$ | 256 byte | FPI slave interface (STM0) | ok | ok |
| $F0001100_H$ | $F00011FF_H$ | 256 byte | FPI slave interface (STM1) | ok | ok |
| $F0001200_H$ | $F00012FF_H$ | 256 byte | FPI slave interface (STM2) | ok | ok |
| $F0001300_H$ | $F00017FF_H$ | - | Reserved | BE | BE |
| $F0001800_H$ | $F00018FF_H$ | 256 byte | FPI slave interface (GPT120) | ok | ok |
| $F0001900_H$ | $F0001BFF_H$ | - | Reserved | BE | BE |
| $F0001C00_H$ | $F0001CFF_H$ | 256 byte | Register block QSPI0 (QSPI0) | ok | ok |
| $F0001D00_H$ | $F0001DFF_H$ | 256 byte | Register block QSPI1 (QSPI1) | ok | ok |
| $F0001E00_H$ | $F0001EFF_H$ | 256 byte | Register block QSPI2 (QSPI2) | ok | ok |
| $F0001F00_H$ | $F0001FFF_H$ | 256 byte | Register block QSPI3 (QSPI3) | ok | ok |
| $F0002000_H$ | $F00029FF_H$ | - | Reserved | BE | BE |
| $F0002A00_H$ | $F0002AFF_H$ | 256 byte | FPI slave interface (CCU60) | ok | ok |
| $F0002B00_H$ | $F0002BFF_H$ | 256 byte | FPI slave interface (CCU61) | ok | ok |
| $F0002C00_H$ | $F000FFFF_H$ | - | Reserved | BE | BE |
| $F0010000_H$ | $F0013FFF_H$ | 16 Kbyte | FPI slave interface (DMA) | ok | ok |
| $F0014000_H$ | $F001BFFF_H$ | - | Reserved | BE | BE |
| $F001C000_H$ | $F001CFFF_H$ | 4 Kbyte | FPI slave interface (ERAY0)<br>ERAY RAM (ERAY0) | ok<br>ok | ok<br>ok |
| $F001D000_H$ | $F001F0FF_H$ | 8.2 Kbyte | FPI bus interface (GETH)<br>FPI bus interface (GETH) | ok<br>ok | ok<br>ok |
| $F001F100_H$ | $F001FFFF_H$ | - | Reserved | BE | BE |
| $F0020000_H$ | $F0023FFF_H$ | 16 Kbyte | FPI slave interface (EVADC) | ok | ok |
| $F0024000_H$ | $F0024FFF_H$ | - | Reserved | BE | BE |
| $F0025000_H$ | $F00250FF_H$ | 256 byte | FPI slave interface (CONVCTRL) | ok | ok |
| $F0025100_H$ | $F002FFFF_H$ | - | Reserved | BE | BE |
| $F0030000_H$ | $F00300FF_H$ | 256 byte | BCU Registers (SBCU) | ok | ok |
| $F0030100_H$ | $F0034FFF_H$ | - | Reserved | BE | BE |
| $F0035000_H$ | $F00351FF_H$ | 512 byte | FPI slave interface (IOM) | ok | ok |
| $F0035200_H$ | $F0035FFF_H$ | - | Reserved | BE | BE |
| $F0036000_H$ | $F00363FF_H$ | 1 Kbyte | SCU: Connections to FPI/BPI bus (SCU)<br>Clocking System Registers (SCU)<br>Power Management Registers (SCU) | ok<br>ok<br>ok | ok<br>ok<br>ok |
| $F0036400_H$ | $F00367FF_H$ | - | Reserved | BE | BE |
| $F0036800_H$ | $F0036FFF_H$ | 2 Kbyte | FPI slave interface (SMU) | ok | ok |
| $F0037000_H$ | $F0037FFF_H$ | 4 Kbyte | IR Status and Control Registers (INT) | ok | ok |
| $F0038000_H$ | $F0039FFF_H$ | 8 Kbyte | IR Service Request Control Registers (SRC) (SRC) | ok | ok |
| $F003A000_H$ | $F003A0FF_H$ | 256 byte | SPB bus slave interface (P00) | ok | ok |

**Table 3 Address Map as seen by Bus Masters on Bus SPB** (cont'd)

| Address Range | | Size | Unit | Access Type | |
|---|---|---|---|---|---|
| from | to | | | Read | Write |
| $F003A100_H$ | $F003A1FF_H$ | - | Reserved | BE | BE |
| $F003A200_H$ | $F003A2FF_H$ | 256 byte | SPB bus slave interface (P02) | ok | ok |
| $F003A300_H$ | $F003A9FF_H$ | - | Reserved | BE | BE |
| $F003AA00_H$ | $F003AAFF_H$ | 256 byte | SPB bus slave interface (P10) | ok | ok |
| $F003AB00_H$ | $F003ABFF_H$ | 256 byte | SPB bus slave interface (P11) | ok | ok |
| $F003AC00_H$ | $F003ACFF_H$ | 256 byte | SPB bus slave interface (P12) | ok | ok |
| $F003AD00_H$ | $F003ADFF_H$ | - | Reserved | BE | BE |
| $F003AE00_H$ | $F003AEFF_H$ | 256 byte | SPB bus slave interface (P14) | ok | ok |
| $F003AF00_H$ | $F003AFFF_H$ | 256 byte | SPB bus slave interface (P15) | ok | ok |
| $F003B000_H$ | $F003B3FF_H$ | - | Reserved | BE | BE |
| $F003B400_H$ | $F003B4FF_H$ | 256 byte | SPB bus slave interface (P20) | ok | ok |
| $F003B500_H$ | $F003B5FF_H$ | 256 byte | SPB bus slave interface (P21) | ok | ok |
| $F003B600_H$ | $F003B6FF_H$ | 256 byte | SPB bus slave interface (P22) | ok | ok |
| $F003B700_H$ | $F003B7FF_H$ | 256 byte | SPB bus slave interface (P23) | ok | ok |
| $F003B800_H$ | $F003BFFF_H$ | - | Reserved | BE | BE |
| $F003C000_H$ | $F003C0FF_H$ | 256 byte | SPB bus slave interface (P32) | ok | ok |
| $F003C100_H$ | $F003C1FF_H$ | 256 byte | SPB bus slave interface (P33) | ok | ok |
| $F003C200_H$ | $F003C2FF_H$ | 256 byte | SPB bus slave interface (P34) | ok | ok |
| $F003C300_H$ | $F003FFFF_H$ | - | Reserved | BE | BE |
| $F0040000_H$ | $F005FFFF_H$ | 128 Kbyte | System Registers (HSM)<br>Debug Registers (HSM)<br>Communication Registers (HSM)<br>HSM Reset (HSM) | 32<br>32<br>32<br>32 | 32<br>32<br>32<br>32 |
| $F0060000_H$ | $F006FFFF_H$ | 64 Kbyte | FPI slave interface (MTU)<br>FPI slave interface (MTU) | ok<br>ok | ok<br>ok |
| $F0070000_H$ | $F00BFFFF_H$ | - | Reserved | BE | BE |
| $F00C0000_H$ | $F00D00FF_H$ | 64.2 Kbyte | FPI slave interface (I2C0)<br>FPI slave interface (I2C0) | ok<br>ok | ok<br>ok |
| $F00D0100_H$ | $F01FFFFF_H$ | - | Reserved | BE | BE |
| $F0200000_H$ | $F0208FFF_H$ | 36 Kbyte | RAM Area (CAN0)<br>Register Area (CAN0) | ok<br>ok | ok<br>ok |
| $F0209000_H$ | $F020FFFF_H$ | - | Reserved | BE | BE |
| $F0210000_H$ | $F0218FFF_H$ | 36 Kbyte | RAM Area (CAN1)<br>Register Area (CAN1) | ok<br>ok | ok<br>ok |
| $F0219000_H$ | $F023FFFF_H$ | - | Reserved | BE | BE |
| $F0240000_H$ | $F0241FFF_H$ | 8 Kbyte | Standby Controller XRAM (PMS) | ok | ok |
| $F0242000_H$ | $F0247FFF_H$ | - | Reserved | BE | BE |

**Table 3 Address Map as seen by Bus Masters on Bus SPB** (cont'd)

| Address Range | | Size | Unit | Access Type | |
|---|---|---|---|---|---|
| from | to | | | Read | Write |
| $F0248000_H$ | $F02481FF_H$ | 512 byte | FPI slave interface (PMS)<br>SMU registers in Standby power domain (PMS) | ok<br>ok | ok<br>ok |
| $F0248200_H$ | $F027FFFF_H$ | - | Reserved | BE | BE |
| $F0280000_H$ | $F0281FFF_H$ | 8 Kbyte | RAM Area (HSPDM) | ok | ok |
| $F0282000_H$ | $F02820FF_H$ | 256 byte | FPI slave interface for BPI registers access (HSPDM) | ok | ok |
| $F0282100_H$ | $F7FFFFFF_H$ | - | Reserved | BE | BE |
| $F8000000_H$ | $FFFFFFFF_H$ | 128 Mbyte | Redirection of SRI ranges (SFIBRIDGE1) | ok | ok |

## 2.5 Bus Instance BBB

**Table 4 Address Map as seen by Bus Masters on Bus BBB**

| Address Range | | Size | Unit | Access Type | |
|---|---|---|---|---|---|
| from | to | | | Read | Write |
| $00000000_H$ | $98FFFFFF_H$ | - | Reserved | BE | BE |
| $99000000_H$ | $990FFFFF_H$ | 1 Mbyte | BBB slave interface 0 (access to EMEM module RAM, cached segment) (EMEMRAM0) | ok | ok |
| $99100000_H$ | $991FFFFF_H$ | 1 Mbyte | BBB slave interface 1 (access to EMEM module RAM, cached segment) (EMEMRAM1) | ok | ok |
| $99200000_H$ | $B8FFFFFF_H$ | - | Reserved | BE | BE |
| $B9000000_H$ | $B90FFFFF_H$ | 1 Mbyte | BBB slave interface 0 (access to EMEM module RAM, non-cached segment) (EMEMRAM0) | ok | ok |
| $B9100000_H$ | $B91FFFFF_H$ | 1 Mbyte | BBB slave interface 1 (access to EMEM module RAM, non-cached segment) (EMEMRAM1) | ok | ok |
| $B9200000_H$ | $B93FFFFF_H$ | - | Reserved | BE | BE |
| $B9400000_H$ | $B947FFFF_H$ | 512 Kbyte | XTM FPI slave interface (XTM) | ok | ok |
| $B9480000_H$ | $EFFFFFFF_H$ | - | Reserved | BE | BE |
| $F0000000_H$ | $FA0000FF_H$ | - | Reserved | BE | BE |
| $FA000100_H$ | $FA0001FF_H$ | 256 byte | BCU Registers (EBCU) | ok | ok |
| $FA000200_H$ | $FA000FFF_H$ | - | Reserved | BE | BE |
| $FA001000_H$ | $FA0010FF_H$ | 256 byte | FPI slave interface (AGBT) | ok | ok |
| $FA001100_H$ | $FA005FFF_H$ | - | Reserved | BE | BE |
| $FA006000_H$ | $FA0060FF_H$ | 256 byte | BPI SFF (access to EMEM core registers) (EMEM) | ok | ok |
| $FA006100_H$ | $FA00FFFF_H$ | - | Reserved | BE | BE |
| $FA010000_H$ | $FA01FFFF_H$ | 64 Kbyte | FPI slave interface (MCDSLIGHT) | ok | ok |
| $FA020000_H$ | $FA03FFFF_H$ | - | Reserved | BE | BE |
| $FA040000_H$ | $FA0401FF_H$ | 512 byte | FPI slave interface (RIF0) | ok | ok |
| $FA040200_H$ | $FA0403FF_H$ | 512 byte | FPI slave interface (RIF1) | ok | ok |
| $FA040400_H$ | $FA6FFFFF_H$ | - | Reserved | BE | BE |

**Table 4 Address Map as seen by Bus Masters on Bus BBB** (cont'd)

| Address Range | | Size | Unit | Access Type | |
|---|---|---|---|---|---|
| from | to | | | Read | Write |
| $FA700000_H$ | $FA7000FF_H$ | 256 byte | SPU Lockstep Registers (SPULCKSTP) | ok | ok |
| $FA700100_H$ | $FA7FFFFF_H$ | - | Reserved | BE | BE |
| $FA800000_H$ | $FA8007FF_H$ | 2 Kbyte | SPU Registers (SPU0) | ok | ok |
| $FA800800_H$ | $FA9FFFFF_H$ | - | Reserved | BE | BE |
| $FAA00000_H$ | $FAA0FFFF_H$ | 64 Kbyte | SPU Configuration Memory (SPU0) | 32 | 32 |
| $FAA10000_H$ | $FABFFFFF_H$ | - | Reserved | BE | BE |
| $FAC00000_H$ | $FAC007FF_H$ | 2 Kbyte | SPU Registers (SPU1) | ok | ok |
| $FAC00800_H$ | $FADFFFFF_H$ | - | Reserved | BE | BE |
| $FAE00000_H$ | $FAE0FFFF_H$ | 64 Kbyte | SPU Configuration Memory (SPU1) | 32 | 32 |
| $FAE10000_H$ | $FFFFFFFF_H$ | - | Reserved | BE | BE |

## 2.6 Revision History

**Table 5 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V0.1.12** | | |
| – | Formal change: for some memory ranges (e.g. “PFI0”) the name was changed by appending “_NC” to “PFI0_NC” to ensure that derived tool files contain different symbols for cached and non-cached memory ranges. | – |
| **Page 10** | Changed description of range starting at F0240000 from “SCR XRAM (PMS)” to “Standby Controller XRAM (PMS)”. | – |
| **V0.1.13** | | |
| – | No changes. Only version number changed to keep alignment with family address map. | – |
| **V0.1.14** | | |
| – | No changes. Only version number changed to keep alignment with family address map. | – |
| **V0.1.15** | | |
| – | No changes. Only version number changed to keep alignment with family address map. | – |
| **V0.1.16** | | |
| – | No changes. Only version number changed to keep alignment with family address map. | – |
| **V0.1.17** | | |
| – | No changes. Only version number changed to keep alignment with family address map. | – |
| **V0.1.18** | | |
| – | No changes. Only version number changed to keep alignment with family address map. | – |
| **V0.1.19** | | |
| – | No changes. Only version number changed to keep alignment with family address map. | |
| **V0.1.20** | | |
| – | No changes. Only version number changed to keep alignment with family address map. | |
| **V0.1.21** | | |
| **Page 8**, **Page 11** | In bus instances SPB and BBB several address ranges corrected to “BE”. | |

# 3 TC35x Firmware

This chapter supplements the family documentation with device specific information for TC35x devices.

## 3.1 Checker Software exit information for ALL CHECKS PASSED

Below the SCU_STMEM3...SCU_STMEM6 registers' content corresponding to "ALL CHECKS PASSED" result from Checker Software (CHSW) upon different device reset types is shown.

**Table 6 "ALL CHECKS PASSED" indication by CHSW for TC35x**

| Reset type | Additional conditions | SCU_STMEM3 | SCU_STMEM4 | SCU_STMEM5 | SCU_STMEM6 |
|---|---|---|---|---|---|
| Cold power-on [1] | -- | $A0F3FB1F_H$ | $00000001_H$ | $A0F3FB1F_H$ | $A0F3FB1F_H$ |
| Warm power-on | -- | $A0E3F82F_H$ | $00000001_H$ | $A0E3F82F_H$ | $A0E3F82F_H$ |
| System reset | -- | $20E0B84F_H$ | $00000001_H$ | $20E0B84F_H$ | $20E0B84F_H$ |
| Application reset | CCUCON5.GETHDIV<>0<br>CCUCON5.ADASDIV<>0 | $20E0088F_H$ | $00000001_H$ | $20E0088F_H$ | $20E0088F_H$ |
| | CCUCON5.GETHDIV<>0<br>CCUCON5.ADASDIV=0 | $20E0088F_H$ | $00C00001_H$ | $2020088F_H$ | $20E0088F_H$ |
| | CCUCON5.GETHDIV=0<br>CCUCON5.ADASDIV<>0 | $20E0088F_H$ | $00200001_H$ | $20C0088F_H$ | $20E0088F_H$ |
| | CCUCON5.GETHDIV=0<br>CCUCON5.ADASDIV=0 | $20E0088F_H$ | $00E00001_H$ | $2000088F_H$ | $20E0088F_H$ |

1) Device start-up after LBIST execution is handled by AURIX™ TC3xx Firmware as cold power-on, therefore the SCU_STMEMx values in this row apply also in such a case (after LBIST).

*Note: The result from some check(s) depends on additional conditions as follows:*

1. *The check for Gigabit Ethernet MAC module(s) calibration will fail after application reset, if the application software has not enabled GETH clock (in SCU_CCUCON5.GETHDIV register) after the previous power-on/system reset(s) - meaning if the module is not used by application - therefore in such use-case anyway the check for this' module calibration is not relevant.*
2. *The check for RIF module(s) calibration will fail after application reset, if the application software has not enabled ADAS clock (in SCU_CCUCON5.ADASDIV register) after the previous power-on/system reset(s) - meaning if the module is not used by application - therefore in such use-case anyway the check for this' module calibration is not relevant.*

## 3.2 Revision History

**Table 7 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V1.1.0.1.14, V1.1.0.1.15, V1.1.0.1.16** | | |
| | No change | |
| **V1.1.0.1.17** | | |
| Table 6 | Footnote added, explaining FW handling after LBIST execution (documentation improvement only, no change in implementation) | |

**Table 7 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V1.1.0.1.18** | | |
| – | No functional changes | |

# 4 On-Chip System Connectivity {and Bridges}

Text with reference to family spec.

## 4.1 TC35x Specific IP Configuration

**Table 8 TC35x specific configuration of DOM**

| Parameter | DOM0 |
|---|---|
| **Application Reset** | Application Reset |
| **Access only when any Endinit (SCU_WDTCPUxCON0.EI = 0 for any CPUx)** | ENDINIT |
| **Access only when Safety Endinit (SCU_SEICON.EI = 0)** | Safety ENDINIT |
| **Access only from HSM Master or HSM debug (or FPRO.PROINHSMCFG=0)** | HSM Access |
| **Access only when PSW = Supervisor Mode** | Supervisor Mode |
| **Access only when PSW = User Mode 0 or 1** | User Mode |
| **Access only when OCDS enabled** | Debug Mode |
| **Access only from Master x (when MOD_ACCEN0.ENx = 1 or MOD_ACCEN1.ENx = 1)** | Valid Master |
| **Access only from Master x (when MOD_ACCEN0.ENx = 1)** | Valid Master (0) |
| **Access only from Master x (when MOD_ACCEN1.ENx = 1)** | Valid Master (1) |
| **Number of SCI interfaces** | 16 |
| **sri base address** | $F8700000_H$ |
| **sri address range** | $10000_H$ |
| **OLDA base address** | $8FE00000_H$ |
| **OLDA range** | $80000_H$ |
| **OLDA base address (non-cached)** | $AFE00000_H$ |
| **OLDA range (non-cached)** | $80000_H$ |

## 4.2 TC35x Specific Register Set

### Register Address Space Table

**Table 9 Register Address Space - DOM**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| (DOM0) | $8FE00000_H$ | $8FE7FFFF_H$ | Online Data Acquisition (OLDA) |
| | $AFE00000_H$ | $AFE7FFFF_H$ | Online Data Acquisition (OLDA) |
| DOM0 | $F8700000_H$ | $F870FFFF_H$ | sri slave interface |

### Register Overview Table

**Table 10 Register Overview - DOM0 (ascending Offset Address)**

| Short Name | Description | Offset Address | Access Mode | | Page Number |
|---|---|---|---|---|---|
| | | | Read | Write | |
| DOM0_PECONx (x=0-15) | Protocol Error Control Register x | $00000_H$+ x*$20_H$ | 32,U,SV | 32,P,SV | See Family Spec |
| DOM0_PRIORITYx (x=0-15) | SCIx Arbiter Priority Register | $00008_H$+ x*$20_H$ | 32,U,SV | 32,P,SV | See Family Spec |
| DOM0_ERRADDRx (x=0-15) | SCI x Error Address Capture Register | $00010_H$+ x*$20_H$ | 32,U,SV | 32,P,SV | See Family Spec |
| DOM0_ERRx (x=0-15) | SCI x Error Capture Register | $00018_H$+ x*$20_H$ | 32,U,SV | 32,P,SV | See Family Spec |
| DOM0_ID | Identification Register | $00408_H$ | 32,U,SV | BE | See Family Spec |
| DOM0_PESTAT | Protocol Error Status Register | $00410_H$ | 32,U,SV | 32,P,SV | See Family Spec |
| DOM0_TIDSTAT | Transaction ID Status Register | $00418_H$ | 32,U,SV | 32,P,SV | See Family Spec |
| DOM0_TIDEN | Transaction ID Enable Register | $00420_H$ | 32,U,SV | 32,P,SV | See Family Spec |
| DOM0_BRCON | Domain 0 Bridge Control Register | $00430_H$ | 32,U,SV | 32,P,SV | 4 |

**Table 10 Register Overview - DOM0 (ascending Offset Address)** (cont'd)

| Short Name | Description | Offset Address | Access Mode | | Page Number |
|---|---|---|---|---|---|
| | | | Read | Write | |
| DOM0_ACCEN0 | Access Enable Register 0 | $004F0_H$ | 32,U,SV | 32,SV,SE | See Family Spec |
| DOM0_ACCEN1 | Access Enable Register 1 | $004F8_H$ | 32,U,SV | 32,SV,SE | See Family Spec |

## 4.3 TC35x Specific Registers

### 4.3.1 sri slave interface

**Domain 0 Bridge Control Register**

**DOM0_BRCON**
**Domain 0 Bridge Control Register** **($00430_H$)** **Application Reset Value: $0000\ 0200_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | 0 | | | |
| r | | | | | | | | | | | | rw | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | 0 | | 0 | 1 | 0 | | 0 | 0 | | | | | OLDAEN |
| rw | | | r | | rw | rw | r | | rw | r | | | | | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **OLDAEN** | 0 | rw | **Online Data Acquisition Enable**<br>This bit is used to control trap generated for write accesses to the OLDA address range associated with this domain.<br>$0_B$ Trap generated on a write access to the OLDA memory range.<br>$1_B$ No trap generated on a write access to the OLDA memory range. |
| **0** | 5:1,<br>8:7,<br>12:11,<br>31:20 | r | **Reserved**<br>Read as 0; shall be written with 0. |
| **0** | 6,<br>10,<br>19:13 | rw | **Reserved**<br>Read as 0; shall be written with 0. |
| **1** | 9 | rw | **Reserved**<br>Read as 1; shall be written with 1. |

## 4.4 Connectivity

No connections in TC35x

## 4.5 Interconnection Matrices

### 4.5.1 Domain 0 Interconnection Matrix

By default the crossbar connects MCIs to SCIs. The following connectivity matrix highlights (in red and yellow) the MCI to SCI interconnects that are non-standard in the TC35x. The unimplemented connections are redundant as they would not be useful (illegal or trapping) therefore do not restrict the functionality.


**Figure 1 TC35x Domain0 Connectivity Matrix**

*Note: DMA MIF0 (MCI0) connections to EMEM0/EMEM1 (SCI9/SCI10) are only used to access the EMEM SFR ranges.*

## 4.6 Revision History

**Table 11 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V1.1.13** | | |
| **Page 4** | Updated BRCON bitfield to show that bit 9 while reserved will read as 1 and should be updated to 1 (The bitfield was incorrectly showing value 0 previously). Restored correct access permission (r/w) for bit fields which are not intended for customer function. | |
| **V1.1.14** | | |
| | No change. | |
| **V1.1.15** | | |
| | No change. | |
| **V1.1.16** | | |
| | No change. | |
| **V1.1.17** | | |
| | No change. | |

## 4.7 FPI Bus Control Units (SBCU, EBCU)

This chapter supplements the family documentation with device specific information for TC35x.

### 4.7.1 TC35x Specific IP Configuration

The TC35x includes two FPI Bus instances. Each FPI Bus instances has its dedicated Bus Control Unit:

**Table 12 Register Address Space - BCU**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| (EBCU) | $F0000000_H$ | $FFFFFFFF_H$ | FPI default slave |
| EBCU | $FA000100_H$ | $FA0001FF_H$ | BCU Registers |
| (SBCU) | $F0000000_H$ | $F7FFFFFF_H$ | FPI default slave |
| SBCU | $F0030000_H$ | $F00300FF_H$ | BCU Registers |

- System Peripheral Bus (SPB) -> SBCU. The SBCU registers are described in **Chapter 4.7.2**
- Back Bone bus (BBB) -> EBCU. The EBCU registers are described in **Chapter 4.7.3**

### 4.7.2 SBCU Control Unit Registers

**Figure 2** and **Table 13** are showing the address maps with all registers of the System Bus Control Unit (SBCU) module.

List of used Reset Class abbreviations:

- Reset Class 1 -> Debug Reset, Power-on Reset (see chapter SCU / Reset Types)
- Reset Class 3 -> Application Reset, System Reset, Power-on Reset (see chapter SCU / Reset Types)

## SBCU Control Registers Overview

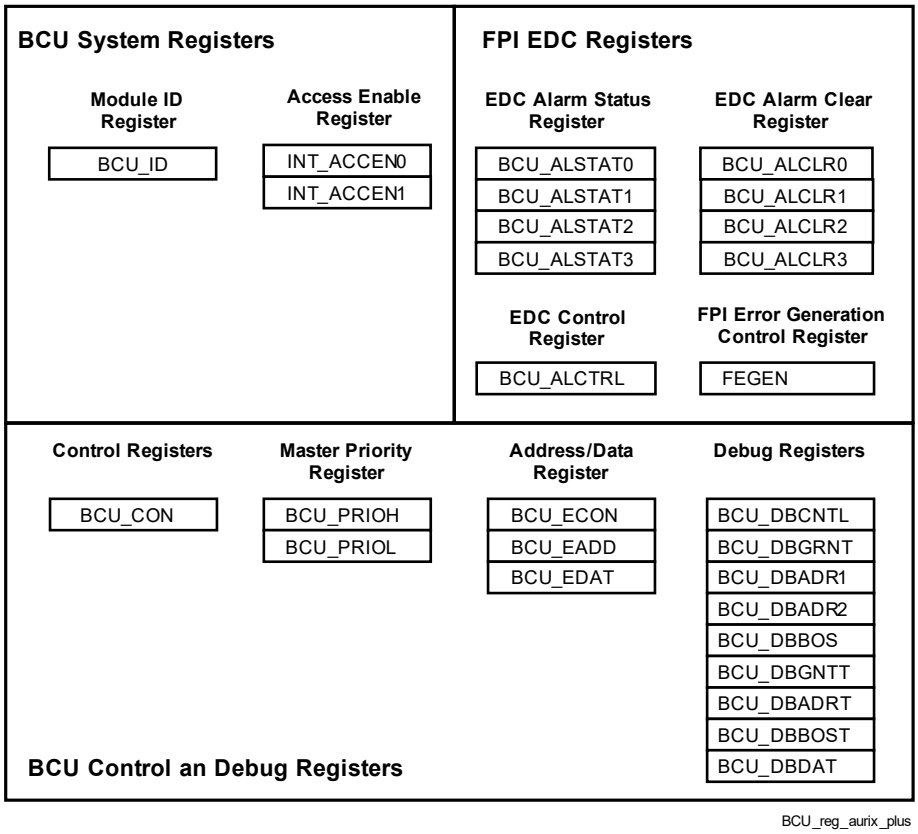



**Figure 2 SBCU Registers**

**Table 13 Register Overview - SBCU (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SBCU_ID | Module Identification Register | $0008_H$ | U,SV | BE | Application Reset | See Family Spec |
| SBCU_CON | BCU Control Register | $0010_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| SBCU_PRIOH | Arbiter Priority Register High | $0014_H$ | U,SV | SV,E,P | Application Reset | 10 |
| SBCU_PRIOL | Arbiter Priority Register Low | $0018_H$ | U,SV | SV,E,P | Application Reset | 11 |
| SBCU_ECON | BCU Error Control Capture Register | $0020_H$ | U,SV | SV,P | Application Reset | See Family Spec |

**Table 13 Register Overview - SBCU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SBCU_EADD | BCU Error Address Capture Register | $0024_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| SBCU_EDAT | BCU Error Data Capture Register | $0028_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| SBCU_DBCNTL | BCU Debug Control Register | $0030_H$ | U,SV | SV,P | Debug Reset | See Family Spec |
| SBCU_DBGRNT | SBCU Debug Grant Mask Register | $0034_H$ | U,SV | SV,P | Debug Reset | 11 |
| SBCU_DBADR1 | BCU Debug Address 1 Register | $0038_H$ | U,SV | SV,P | Debug Reset | See Family Spec |
| SBCU_DBADR2 | BCU Debug Address 2 Register | $003C_H$ | U,SV | SV,P | Debug Reset | See Family Spec |
| SBCU_DBBOS | BCU Debug Bus Operation Signals Register | $0040_H$ | U,SV | SV,P | Debug Reset | See Family Spec |
| SBCU_DBGNTT | SBCU Debug Trapped Master Register | $0044_H$ | U,SV | BE | Debug Reset | 13 |
| SBCU_DBADRT | BCU Debug Trapped Address Register | $0048_H$ | U,SV | BE | Debug Reset | See Family Spec |
| SBCU_DBBOST | BCU Debug Trapped Bus Operation Signals Register | $004C_H$ | U,SV | BE | Debug Reset | See Family Spec |
| SBCU_DBDAT | BCU Debug Data Status Register | $0050_H$ | U,SV | BE | Debug Reset | See Family Spec |
| SBCU_ALSTATx (x=0-3) | BCU EDC Alarm Status Register x | $0060_H$+x *4 | U,SV | SV,P | Application Reset | 14 |
| SBCU_ALCLRx (x=0-3) | BCU EDC Alarm Clear Register x | $0070_H$+x *4 | U,SV | SV,P | Application Reset | See Family Spec |
| SBCU_ALCTRL | BCU EDC Alarm Control Register | $0080_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| SBCU_FEGEN | FPI Error Generation Control Register | $0084_H$ | U,SV | SV,SE | Application Reset | See Family Spec |

**Table 13 Register Overview - SBCU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SBCU_ACCEN1 | Access Enable Register 1 | $00F8_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| SBCU_ACCEN0 | Access Enable Register 0 | $00FC_H$ | U,SV | SV,SE | Application Reset | See Family Spec |

### 4.7.2.1 SBCU Control Registers Descriptions

*Note: For all PRIOH / PRIOL bit fields, a lower number has a higher priority in the arbitration round than a higher one.*

**Arbiter Priority Register High**

**SBCU_PRIOH**
**Arbiter Priority Register High** **($0014_H$)** **Application Reset Value: $FEDC\ 8888_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RESERVED | | | | RESERVED | | | | HSMCMI | | | | HSMRMI | | | |
| rw | | | | rw | | | | rw | | | | rw | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RESERVED | | | | RESERVED | | | | RESERVED | | | | CPU2 | | | |
| rw | | | | rw | | | | rw | | | | rw | | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CPU2** | 3:0 | rw | **CPU2 Priority (Index 8)**<br>This bit field defines the priority on the SPB for CPU2 access to the SPB. |
| **RESERVED** | 7:4,<br>11:8,<br>15:12,<br>27:24,<br>31:28 | rw | **Reserved**<br>Read as reset value or last written value; should be written with 0. |
| **HSMRMI** | 19:16 | rw | **HSMRMI Priority (Index 12)**<br>This bit field defines the priority on the SPB for HSMRMI access to the SPB. |
| **HSMCMI** | 23:20 | rw | **HSMCMI Priority (Index 13)**<br>This bit field defines the priority on the SPB for HSMCMI access to the SPB. |

**Arbiter Priority Register Low**

**SBCU_PRIOL**
**Arbiter Priority Register Low** **($0018_H$)** **Application Reset Value: $8854\ 3210_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CPU1 | | | | CPU0 | | | | RESERVED | | | | RESERVED | | | |
| rw | | | | rw | | | | rw | | | | rw | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RESERVED | | | | RESERVED | | | | RESERVED | | | | DMA | | | |
| rw | | | | rw | | | | rw | | | | rw | | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **DMA** | 3:0 | rw | **DMA / Cerberus Priority (Index 0)**<br>This bit field defines the priority on the SPB for DMA and Cerberus access to the SPB. |
| **RESERVED** | 7:4,<br>11:8,<br>15:12,<br>19:16,<br>23:20 | rw | **Reserved**<br>Read as reset value or last written value; should be written with 0. |
| **CPU0** | 27:24 | rw | **CPU0 Priority (Index 6)**<br>This bit field defines the priority on the SPB for CPU0 access to the SPB. |
| **CPU1** | 31:28 | rw | **CPU1 Priority (Index 7)**<br>This bit field defines the priority on the SPB for CPU1 access to the SPB. |

## 4.7.2.2 SBCU OCDS Registers Descriptions

**SBCU Debug Grant Mask Register**

**SBCU_DBGRNT**
**SBCU Debug Grant Mask Register** **($0034_H$)** **Debug Reset Value: $0000\ FFFF_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | HSMCMI | HSMRMI | 1 | 1 | 1 | CPU2 | CPU1 | CPU0 | 1 | 1 | 1 | 1 | 1 | DMA |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **DMA** | 0 | rw | **DMA / Cerberus Trigger Enable**<br>$0_B$ FPI Bus transactions with DMA / Cerberus as bus master are enabled for grant trigger event generation.<br>$1_B$ FPI Bus transactions with DMA / Cerberus as bus master are disabled for grant trigger event generation. |
| **CPU0** | 6 | rw | **CPU0 Grant Trigger Enable**<br>$0_B$ FPI Bus transactions with CPU0 as bus master are enabled for grant trigger event generation.<br>$1_B$ FPI Bus transactions with CPU as bus master are disabled for grant trigger event generation. |
| **CPU1** | 7 | rw | **CPU1 Grant Trigger Enable**<br>$0_B$ FPI Bus transactions with CPU1 as bus master are enabled for grant trigger event generation.<br>$1_B$ FPI Bus transactions with CPU1 as bus master are disabled for grant trigger event generation. |
| **CPU2** | 8 | rw | **CPU2 Grant Trigger Enable**<br>$0_B$ FPI Bus transactions with CPU2 as bus master are enabled for grant trigger event generation.<br>$1_B$ FPI Bus transactions with CPU2 as bus master are disabled for grant trigger event generation. |
| **HSMRMI** | 12 | rw | **HSM Register Master Interface Grant Trigger Enable**<br>$0_B$ FPI Bus transactions requested by the HSM bus master are enabled for grant trigger event generation.<br>$1_B$ FPI Bus transactions requested by the HSM bus master are disabled for grant trigger event generation. |
| **HSMCMI** | 13 | rw | **HSM Cache Master Interface Grant Trigger Enable**<br>$0_B$ FPI Bus transactions requested by the HSM bus master are enabled for grant trigger event generation.<br>$1_B$ FPI Bus transactions requested by the HSM bus master are disabled for grant trigger event generation. |
| **1** | 1,<br>2,<br>3,<br>4,<br>5,<br>9,<br>10,<br>11,<br>14,<br>15 | rw | **Reserved**<br>Read as 1 after reset; reading these bits will return the value last written. |
| **0** | 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

### SBCU Debug Trapped Master Register

**SBCU_DBGNTT**
**SBCU Debug Trapped Master Register (0044$_H$) Debug Reset Value: 0000 FFFF$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| rh | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | HSMCMI | HSMRMI | 1 | 1 | 1 | CPU2 | CPU1 | CPU0 | 1 | 1 | 1 | 1 | 1 | DMA |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **DMA** | 0 | rh | **DMA / Cerberus FPI Bus Master Status**<br>$0_B$ The DMA or Cerberus was the FPI bus master.<br>$1_B$ Neither DMA nor Cerberus was the FPI Bus master. |
| **CPU0** | 6 | rh | **CPU0 FPI Bus Master Status**<br>This bit indicates whether the CPU0 was FPI Bus master when the break trigger event occurred.<br>$0_B$ The CPU0 was the FPI Bus master.<br>$1_B$ The CPU0 was not the FPI Bus master. |
| **CPU1** | 7 | rh | **CPU1 FPI Bus Master Status**<br>This bit indicates whether the CPU1 was FPI Bus master when the break trigger event occurred.<br>$0_B$ The CPU1 was the FPI Bus master.<br>$1_B$ The CPU1 was not the FPI Bus master. |
| **CPU2** | 8 | rh | **CPU2 Grant Trigger Enable**<br>$0_B$ FPI Bus transactions with CPU2 as bus master are enabled for grant trigger event generation<br>$1_B$ FPI Bus transactions with CPU2 as bus master are disabled for grant trigger event generation |
| **HSMRMI** | 12 | rh | **HSM Register FPI Bus Master Interface Status**<br>This bit indicates whether the HSM was FPI Bus master when the break trigger event occurred.<br>$0_B$ HSMRMI was the FPI bus master.<br>$1_B$ HSMRMI was not the FPI Bus master. |
| **HSMCMI** | 13 | rh | **HSM Cache FPI Bus Master Interface Status**<br>This bit indicates whether the HSM was FPI Bus master when the break trigger event occurred.<br>$0_B$ HSMCMI was the FPI bus master.<br>$1_B$ HSMCMI was not the FPI Bus master. |

| Field | Bits | Type | Description |
|---|---|---|---|
| **1** | 1,<br>2,<br>3,<br>4,<br>5,<br>9,<br>10,<br>11,<br>14,<br>15 | rh | **Reserved**<br>Read as 1 after reset; reading these bits will return the value last written. |
| **0** | 31:16 | rh | **Reserved**<br>Read as 1 after reset; reading these bits will return the value last written. |

### BCU EDC Alarm Status Register x

The BCU provides one Alarm Status Register bit for each implemented FPI master and FPI slave.

Register bits without constant definition are reserved in this product.

**SBCU_ALSTATx (x=0)**

**BCU EDC Alarm Status Register x** **($0060_H$+x*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **AL31** | **AL30** | **AL29** | **AL28** | **AL27** | **AL26** | **AL25** | **AL24** | **AL23** | **AL22** | **AL21** | **AL20** | **AL19** | **AL18** | **AL17** | **AL16** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **AL15** | **AL14** | **AL13** | **AL12** | **AL11** | **AL10** | **AL09** | **AL08** | **AL07** | **AL06** | **AL05** | **AL04** | **AL03** | **AL02** | **AL01** | **AL00** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ALy (y=00)** | y | rh | **Alarm y**<br>$1_B$ **SBCU_S**, an EDC error was detected in an active phase of the SBCU Slave Interface. |
| **ALy (y=01)** | y | rh | **Alarm y**<br>$1_B$ **DMA_S**, |
| **ALy (y=02)** | y | rh | **Alarm y**<br>$1_B$ **IR_S**, |
| **ALy (y=03)** | y | rh | **Alarm y**<br>$1_B$ **SFI_F2S_S**, |
| **ALy (y=04)** | y | rh | **Alarm y**<br>$1_B$ **SCU_S**, |
| **ALy (y=05)** | y | rh | **Alarm y**<br>$1_B$ **SMU_S**, |
| **ALy (y=06)** | y | rh | **Alarm y**<br>$1_B$ **PMC_SCR_S**, |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ALy (y=07)** | y | rh | **Alarm y**<br>$1_B$ **MTU_S**, |
| **ALy (y=08)** | y | rh | **Alarm y**<br>$1_B$ **IOM_S**, |
| **ALy (y=09,12-13,18-19,21,25-29,31)** | y | rh | **Alarm y** |
| **ALy (y=10)** | y | rh | **Alarm y**<br>$1_B$ **ASCLIN01_S**, |
| **ALy (y=11)** | y | rh | **Alarm y**<br>$1_B$ **ASCLIN23_S**, |
| **ALy (y=14)** | y | rh | **Alarm y**<br>$1_B$ **QSPI0_S**, |
| **ALy (y=15)** | y | rh | **Alarm y**<br>$1_B$ **QSPI1_S**, |
| **ALy (y=16)** | y | rh | **Alarm y**<br>$1_B$ **QSPI2_S**, |
| **ALy (y=17)** | y | rh | **Alarm y**<br>$1_B$ **QSPI3_S**, |
| **ALy (y=20)** | y | rh | **Alarm y**<br>$1_B$ **FCE0_S**, |
| **ALy (y=22)** | y | rh | **Alarm y**<br>$1_B$ **STM0_S**, |
| **ALy (y=23)** | y | rh | **Alarm y**<br>$1_B$ **STM1_S**, |
| **ALy (y=24)** | y | rh | **Alarm y**<br>$1_B$ **STM2_S**, |
| **ALy (y=30)** | y | rh | **Alarm y**<br>$1_B$ **ERAY0_S**, |

**SBCU_ALSTATx (x=1)**

**BCU EDC Alarm Status Register x** **($0060_H$+x*4)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **AL31** | **AL30** | **AL29** | **AL28** | **AL27** | **AL26** | **AL25** | **AL24** | **AL23** | **AL22** | **AL21** | **AL20** | **AL19** | **AL18** | **AL17** | **AL16** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **AL15** | **AL14** | **AL13** | **AL12** | **AL11** | **AL10** | **AL09** | **AL08** | **AL07** | **AL06** | **AL05** | **AL04** | **AL03** | **AL02** | **AL01** | **AL00** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ALy (y=00)** | y | rh | **Alarm y**<br>$1_B$ **GPT12_S**, an EDC error was detected in an active phase of the GPT12 Slave Interface. |
| **ALy (y=01)** | y | rh | **Alarm y**<br>$1_B$ **CCU6_S**, |
| **ALy (y=02-07,10,12,15,17-18,20-27,30)** | y | rh | **Alarm y** |
| **ALy (y=08)** | y | rh | **Alarm y**<br>$1_B$ **ETH_S**, |
| **ALy (y=09)** | y | rh | **Alarm y**<br>$1_B$ **EVADC_S**, |
| **ALy (y=11)** | y | rh | **Alarm y**<br>$1_B$ **HSM_S**, |
| **ALy (y=13)** | y | rh | **Alarm y**<br>$1_B$ **CAN0_S**, |
| **ALy (y=14)** | y | rh | **Alarm y**<br>$1_B$ **CAN1_S**, |
| **ALy (y=16)** | y | rh | **Alarm y**<br>$1_B$ **I2C0_S**, I2C0_S |
| **ALy (y=19)** | y | rh | **Alarm y**<br>$1_B$ **CONVCTRL_S**, |
| **ALy (y=28)** | y | rh | **Alarm y**<br>$1_B$ **HSPDM_SRAM_S**, |
| **ALy (y=29)** | y | rh | **Alarm y**<br>$1_B$ **HSPDM_SFR_S**, |
| **ALy (y=31)** | y | rh | **Alarm y**<br>$1_B$ **CERBERUS_S**, |

**SBCU_ALSTATx (x=2)**

**BCU EDC Alarm Status Register x** **($0060_H$+x*4)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **AL31** | **AL30** | **AL29** | **AL28** | **AL27** | **AL26** | **AL25** | **AL24** | **AL23** | **AL22** | **AL21** | **AL20** | **AL19** | **AL18** | **AL17** | **AL16** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **AL15** | **AL14** | **AL13** | **AL12** | **AL11** | **AL10** | **AL09** | **AL08** | **AL07** | **AL06** | **AL05** | **AL04** | **AL03** | **AL02** | **AL01** | **AL00** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ALy (y=00)** | y | rh | **Alarm y**<br>$1_B$ **P00_S**, an EDC error was detected in an active phase of the Port 00 Slave Interface. |
| **ALy (y=01,03,07,10,15-20,24-26,29-30)** | y | rh | **Alarm y** |
| **ALy (y=02)** | y | rh | **Alarm y**<br>$1_B$ **P02_S**, |
| **ALy (y=04)** | y | rh | **Alarm y**<br>$1_B$ **P10_S**, |
| **ALy (y=05)** | y | rh | **Alarm y**<br>$1_B$ **P11_S**, |
| **ALy (y=06)** | y | rh | **Alarm y**<br>$1_B$ **P12_S**, |
| **ALy (y=08)** | y | rh | **Alarm y**<br>$1_B$ **P14_S**, |
| **ALy (y=09)** | y | rh | **Alarm y**<br>$1_B$ **P15_S**, |
| **ALy (y=11)** | y | rh | **Alarm y**<br>$1_B$ **P20_S**, |
| **ALy (y=12)** | y | rh | **Alarm y**<br>$1_B$ **P21_S**, |
| **ALy (y=13)** | y | rh | **Alarm y**<br>$1_B$ **P22_S**, |
| **ALy (y=14)** | y | rh | **Alarm y**<br>$1_B$ **P23_S**, |
| **ALy (y=21)** | y | rh | **Alarm y**<br>$1_B$ **P32_S**, |
| **ALy (y=22)** | y | rh | **Alarm y**<br>$1_B$ **P33_S**, |
| **ALy (y=23)** | y | rh | **Alarm y**<br>$1_B$ **P34_S**, |
| **ALy (y=27)** | y | rh | **Alarm y**<br>$1_B$ **P50_S**, |
| **ALy (y=28)** | y | rh | **Alarm y**<br>$1_B$ **P51_S**, |
| **ALy (y=31)** | y | rh | **Alarm y**<br>$1_B$ **SBCU_M**, an EDC error was detected in an active phase of the SBCU Master Interface. |

**SBCU_ALSTATx (x=3)**
**BCU EDC Alarm Status Register x** **($0060_H$+x*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **AL31** | **AL30** | **AL29** | **AL28** | **AL27** | **AL26** | **AL25** | **AL24** | **AL23** | **AL22** | **AL21** | **AL20** | **AL19** | **AL18** | **AL17** | **AL16** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **AL15** | **AL14** | **AL13** | **AL12** | **AL11** | **AL10** | **AL09** | **AL08** | **AL07** | **AL06** | **AL05** | **AL04** | **AL03** | **AL02** | **AL01** | **AL00** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ALy (y=00)** | y | rh | **Alarm y**<br>$1_B$ **A_EN**, multiple output enables active: A_EN_N (Master) |
| **ALy (y=01)** | y | rh | **Alarm y**<br>$1_B$ **ABORT_EN**, multiple output enables active: ABORT_EN_N (Master) |
| **ALy (y=02)** | y | rh | **Alarm y**<br>$1_B$ **ACK_EN**, multiple output enables active: ACK_EN_N (Default Master and Slave) |
| **ALy (y=03)** | y | rh | **Alarm y**<br>$1_B$ **D_EN**, multiple output enables active: D_EN_N (Master and Slave) |
| **ALy (y=04-15,17-21,25-27,30-31)** | y | rh | **Alarm y** |
| **ALy (y=16)** | y | rh | **Alarm y**<br>$1_B$ **DMA_M**, an EDC error was detected in an active phase of the DMA / Cerberus Master Interface. |
| **ALy (y=22)** | y | rh | **Alarm y**<br>$1_B$ **CPU0_M**, |
| **ALy (y=23)** | y | rh | **Alarm y**<br>$1_B$ **CPU1_M**, |
| **ALy (y=24)** | y | rh | **Alarm y**<br>$1_B$ **CPU2_M**, |
| **ALy (y=28)** | y | rh | **Alarm y**<br>$1_B$ **HSMRMI_M**, |
| **ALy (y=29)** | y | rh | **Alarm y**<br>$1_B$ **HSMCMI_M**, |

### 4.7.3 EBCU Control Unit Registers

**Figure 3** and **Table 14** are showing the address maps with all registers of the Back Bone Bus (BBB) Bus Control Unit (EBCU) module.

List of used Reset Class abbreviations:

- Reset Class 1 -> Debug Reset, Power-on Reset (see chapter SCU / Reset Types)
- Reset Class 3 -> Application Reset, System Reset, Power-on Reset (see chapter SCU / Reset Types)

## EBCU Control Registers Overview

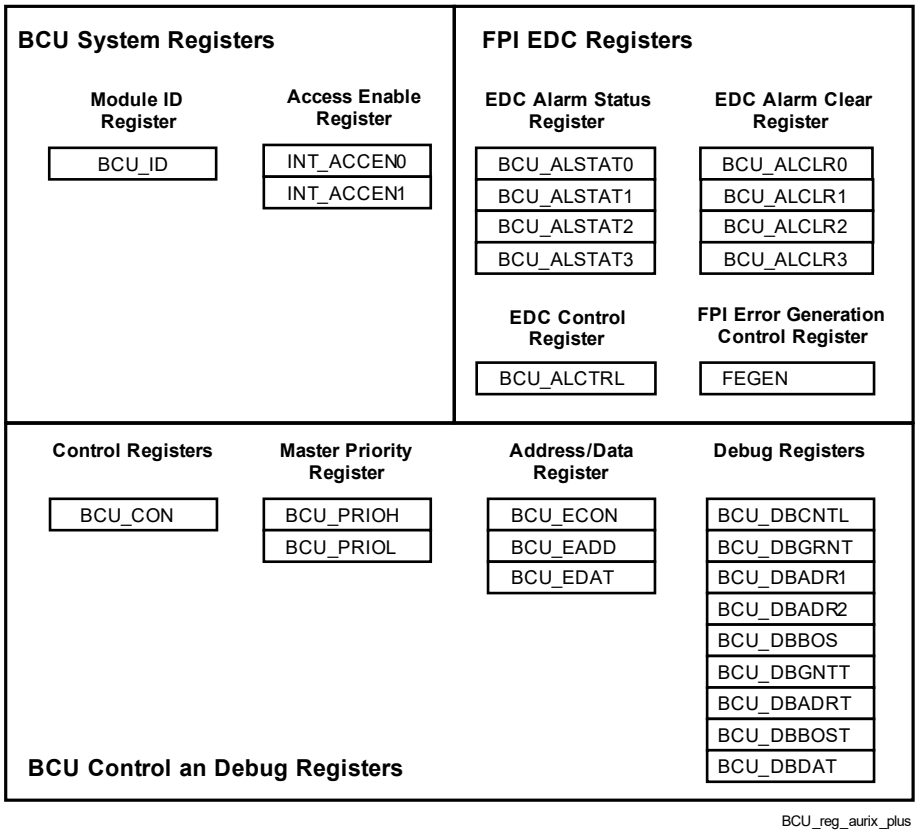



**Figure 3 EBCU Registers**

**Table 14 Register Overview - EBCU (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| EBCU_ID | Module Identification Register | $0008_H$ | U,SV | BE | Application Reset | See Family Spec |
| EBCU_CON | BCU Control Register | $0010_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| EBCU_PRIOH | Arbiter Priority Register High | $0014_H$ | U,SV | SV,E,P | Application Reset | 21 |
| EBCU_PRIOL | Arbiter Priority Register Low | $0018_H$ | U,SV | SV,E,P | Application Reset | 21 |
| EBCU_ECON | BCU Error Control Capture Register | $0020_H$ | U,SV | SV,P | Application Reset | See Family Spec |

**Table 14 Register Overview - EBCU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| EBCU_EADD | BCU Error Address Capture Register | $0024_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| EBCU_EDAT | BCU Error Data Capture Register | $0028_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| EBCU_DBCNTL | BCU Debug Control Register | $0030_H$ | U,SV | SV,P | Debug Reset | 22 |
| EBCU_DBGRNT | EBCU Debug Grant Mask Register | $0034_H$ | U,SV | SV,P | Debug Reset | 24 |
| EBCU_DBADR1 | BCU Debug Address 1 Register | $0038_H$ | U,SV | SV,P | Debug Reset | See Family Spec |
| EBCU_DBADR2 | BCU Debug Address 2 Register | $003C_H$ | U,SV | SV,P | Debug Reset | See Family Spec |
| EBCU_DBBOS | BCU Debug Bus Operation Signals Register | $0040_H$ | U,SV | SV,P | Debug Reset | See Family Spec |
| EBCU_DBGNTT | EBCU Debug Trapped Master Register | $0044_H$ | U,SV | BE | Debug Reset | 25 |
| EBCU_DBADRT | BCU Debug Trapped Address Register | $0048_H$ | U,SV | BE | Debug Reset | See Family Spec |
| EBCU_DBBOST | BCU Debug Trapped Bus Operation Signals Register | $004C_H$ | U,SV | BE | Debug Reset | See Family Spec |
| EBCU_DBDAT | BCU Debug Data Status Register | $0050_H$ | U,SV | BE | Debug Reset | See Family Spec |
| EBCU_ALSTATx (x=0-3) | BCU EDC Alarm Status Register x | $0060_H$+x *4 | U,SV | SV,P | Application Reset | 26 |
| EBCU_ALCLRx (x=0-3) | BCU EDC Alarm Clear Register x | $0070_H$+x *4 | U,SV | SV,P | Application Reset | See Family Spec |
| EBCU_ALCTRL | BCU EDC Alarm Control Register | $0080_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| EBCU_FEGEN | FPI Error Generation Control Register | $0084_H$ | U,SV | SV,SE | Application Reset | See Family Spec |

**Table 14 Register Overview - EBCU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| EBCU_ACCEN1 | Access Enable Register 1 | $00F8_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| EBCU_ACCEN0 | Access Enable Register 0 | $00FC_H$ | U,SV | SV,SE | Application Reset | See Family Spec |

### 4.7.3.1 EBCU Control Registers Descriptions

*Note: For all PRIOH / PRIOL bit fields, a lower number has a higher priority in the arbitration round than a higher one.*

**Arbiter Priority Register High**

**EBCU_PRIOH**
**Arbiter Priority Register High ($0014_H$) Application Reset Value: FEDC BA98$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RESERVED | | | | RESERVED | | | | RESERVED | | | | RESERVED | | | |
| rw | | | | rw | | | | rw | | | | rw | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RESERVED | | | | RESERVED | | | | RESERVED | | | | RESERVED | | | |
| rw | | | | rw | | | | rw | | | | rw | | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **RESERVED (i=8-15)** | 4*i-29:4*i-32 | rw | **Reserved** Read as reset value or last written value; should be written with 0. |

**Arbiter Priority Register Low**

**EBCU_PRIOL**
**Arbiter Priority Register Low ($0018_H$) Application Reset Value: 7658 8210$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RESERVED | | | | SFI_S2F | | | | RESERVED | | | | RESERVED | | | |
| rw | | | | rw | | | | rw | | | | rw | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IOC32E | | | | RESERVED | | | | RESERVED | | | | IOC32P | | | |
| rw | | | | rw | | | | rw | | | | rw | | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **IOC32P** | 3:0 | rw | **IOC32P Priority (Index 0)**<br>This bit field defines the priority on the BBB for IOC32P access to the BBB. |
| **RESERVED** | 7:4,<br>11:8,<br>19:16,<br>23:20,<br>31:28 | rw | **Reserved**<br>Read as reset value or last written value; should be written with 0. |
| **IOC32E** | 15:12 | rw | **IOC32E Priority (Index 3)**<br>This bit field defines the priority on the BBB for IOC32E access to the BBB. |
| **SFI_S2F** | 27:24 | rw | **SFI Bridge SRI2FPI Priority (Index 6)**<br>This bit field defines the priority on the BPB for SFI_S2F access to the BBB. |

### 4.7.3.2 EBCU OCDS Registers Descriptions

**BCU Debug Control Register**

**EBCU_DBCNTL**
**BCU Debug Control Register** **($0030_H$)** **Debug Reset Value: $0000\ 7003_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **ONBOS3** | **ONBOS2** | **ONBOS1** | **ONBOS0** | **0** | | **ONA2** | | **0** | | **ONA1** | | **0** | | | **ONG** |
| rw | rw | rw | rw | r | | rw | | r | | rw | | r | | | rw |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **CONCOM2** | **CONCOM1** | **CONCOM0** | **0** | | | | **0** | **0** | | **RA** | **0** | | **OA** | **EO** |
| r | rw | rw | rw | r | | | | r | r | | w | r | | rh | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **EO** | 0 | r | **Status of BCU Debug Support Enable**<br>This bit is controlled by the Cerberus and enables the BCU debug support.<br>$0_B$ BCU debug support is disabled<br>$1_B$ BCU debug support is enabled (default after reset) |
| **OA** | 1 | rh | **Status of BCU Breakpoint Logic**<br>The OA bit is set by writing a 1 to bit RA. When OA is set, registers DBGNTT, DBADRT and DBDAT are reset. Also DBBOST is reset with the exception of the bit field FPIRST.<br>$0_B$ The BCU breakpoint logic is disarmed. Any further breakpoint activation is discarded<br>$1_B$ The BCU breakpoint logic is armed |

| Field | Bits | Type | Description |
|---|---|---|---|
| **RA** | 4 | w | **Rearm BCU Breakpoint Logic**<br>Writing a 1 to this bit rearms BCU breakpoint logic and sets bit OA = 1. RA is always reads as 0. |
| **CONCOM0** | 12 | rw | **Grant and Address Trigger Relation**<br>$0_B$ The grant phase trigger condition and the address trigger condition (see CONCOM1) are combined with a logical OR for further control<br>$1_B$ The grant phase trigger condition and the address trigger condition (see CONCOM1) are combined with a logical AND for further control. |
| **CONCOM1** | 13 | rw | **Address 1 and Address 2 Trigger Relation**<br>$0_B$ Address 1 trigger condition and address 2 trigger condition are combined with a logical OR to the address trigger condition for further control<br>$1_B$ Address 1 trigger condition and address 2 trigger condition are combined with a logical AND to the address trigger condition for further control |
| **CONCOM2** | 14 | rw | **Address and Signal Trigger Relation**<br>$0_B$ Address trigger condition (see CONCOM1) and signal status trigger conditions are combined with a logical OR for further control<br>$1_B$ Address phase trigger condition (see CONCOM1) and the signal status trigger conditions are combined with a logical AND for further control |
| **ONG** | 16 | rw | **Grant Trigger Enable**<br>$0_B$ No grant debug event trigger is generated<br>$1_B$ The grant debug event trigger is enabled and generated according the settings of register DBGRNT |
| **ONA1** | 21:20 | rw | **Address 1 Trigger Control**<br>$00_B$ No address 1 trigger is generated<br>$01_B$ An address 1 trigger event is generated if the FPI Bus address is equal to DBADR1<br>$10_B$ An address 1 trigger event is generated if FPI Bus address is greater or equal to DBADR1<br>$11_B$ same as $00_B$ |
| **ONA2** | 25:24 | rw | **Address 2 Trigger Control**<br>$00_B$ No address 2 trigger is generated.<br>$01_B$ An address 2 trigger event is generated if the FPI Bus address is equal to DBADR2<br>$10_B$ An address 2 trigger event is generated if FPI Bus address is less or equal to DBADR2<br>$11_B$ same as $00_B$ |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ONBOS0** | 28 | rw | **Op code Signal Status Trigger Condition**<br>$0_B$ A signal status trigger is generated for all FPI Bus op-codes except a “no operation” op-code<br>$1_B$ A signal status trigger is generated if the FPI Bus op-code matches the op-code as defined in DBBOS.OPC |
| **ONBOS1** | 29 | rw | **Supervisor Mode Signal Trigger Condition**<br>$0_B$ The signal status trigger generation for the FPI Bus Supervisor Mode signal is disabled.<br>$1_B$ A signal status trigger is generated if the FPI Bus Supervisor Mode signal state is equal to the value of DBBOS.SVM |
| **ONBOS2** | 30 | rw | **Write Signal Trigger Condition**<br>$0_B$ The signal status trigger generation for the FPI Bus write signal is disabled.<br>$1_B$ A signal status trigger is generated if the FPI Bus write signal state is equal to the value of DBBOS.WR |
| **ONBOS3** | 31 | rw | **Read Signal Trigger Condition**<br>$0_B$ The signal status trigger generation for the FPI Bus read signal is disabled.<br>$1_B$ A signal status trigger is generated if the FPI Bus read signal state is equal to the value of DBBOS.RD |
| **0** | 3:2, 6:5, 7, 11:8, 15, 19:17, 23:22, 27:26 | r | **Reserved**<br>Read as 0; should be written with 0. |

### EBCU Debug Grant Mask Register

**EBCU_DBGRNT**

**EBCU Debug Grant Mask Register** **($0034_H$)** **Debug Reset Value: 0000 FFFF$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1** | **1** | **1** | **1** | **1** | **1** | **1** | **1** | **1** | **SFI_S2F** | **1** | **1** | **IOC32E** | **1** | **1** | **IOC32P** |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **IOC32P** | 0 | rw | **IOC32P Trigger Enable**<br>$0_B$ FPI Bus transactions with IOC32P as bus master are enabled for grant trigger event generation<br>$1_B$ FPI Bus transactions with IOC32P as bus master are disabled for grant trigger event generation |
| **IOC32E** | 3 | rw | **IOC32E Grant Trigger Enable**<br>$0_B$ FPI Bus transactions with IOC32E as bus master are enabled for grant trigger event generation<br>$1_B$ FPI Bus transactions with IOC32E as bus master are disabled for grant trigger event generation |
| **SFI_S2F** | 6 | rw | **SFI_S2F Grant Trigger Enable**<br>$0_B$ FPI Bus transactions with SFI_S2F as bus master are enabled for grant trigger event generation<br>$1_B$ FPI Bus transactions with SFI_S2F as bus master are disabled for grant trigger event generation |
| **1** | 1,<br>2,<br>4,<br>5,<br>7,<br>8,<br>9,<br>10,<br>11,<br>12,<br>13,<br>14,<br>15 | rw | **Reserved**<br>Read as 1 after reset; reading these bits will return the value last written. |
| **0** | 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

### EBCU Debug Trapped Master Register

**EBCU_DBGNTT**
**EBCU Debug Trapped Master Register** **($0044_H$)** **Debug Reset Value: 0000 FFFF$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | | | | | | | | |
| rh | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1** | **1** | **1** | **1** | **1** | **1** | **1** | **1** | **1** | **SFI_S2F** | **1** | **1** | **IOC32E** | **1** | **1** | **IOC32P** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **IOC32P** | 0 | rh | **IOC32P FPI Bus Master Status**<br>This bit indicates whether the IOC32P was FPI Bus master when the break trigger event occurred.<br>$0_B$ The IOC32P was the FPI bus master.<br>$1_B$ The IOC32P was not the FPI Bus master. |
| **IOC32E** | 3 | rh | **IOC32E FPI Bus Master Status**<br>This bit indicates whether the IOC32E was FPI Bus master when the break trigger event occurred.<br>$0_B$ The IOC32E was the FPI bus master.<br>$1_B$ The IOC32E was not the FPI Bus master. |
| **SFI_S2F** | 6 | rh | **SFI_S2F FPI Bus Master Status**<br>This bit indicates whether the SFI_S2F with a medium priority request was FPI Bus master when the break trigger event occurred.<br>$0_B$ The medium-priority SFI_S2F was the FPI bus master.<br>$1_B$ The medium-priority SFI_S2F was not the FPI Bus master. |
| **1** | 1,<br>2,<br>4,<br>5,<br>7,<br>8,<br>9,<br>10,<br>11,<br>12,<br>13,<br>14,<br>15 | rh | **Reserved**<br>Read as 1 after reset; reading these bits will return the value last written. |
| **0** | 31:16 | rh | **Reserved**<br>Read as 1 after reset; reading these bits will return the value last written. |

**EBCU_ALSTATx (x=0)**

**BCU EDC Alarm Status Register x** **($0060_H$+x*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **AL31** | **AL30** | **AL29** | **AL28** | **AL27** | **AL26** | **AL25** | **AL24** | **AL23** | **AL22** | **AL21** | **AL20** | **AL19** | **AL18** | **AL17** | **AL16** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **AL15** | **AL14** | **AL13** | **AL12** | **AL11** | **AL10** | **AL09** | **AL08** | **AL07** | **AL06** | **AL05** | **AL04** | **AL03** | **AL02** | **AL01** | **AL00** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ALy (y=00)** | y | rh | **Alarm y**<br>$1_B$ **EBCU_S**, |
| **ALy (y=01)** | y | rh | **Alarm y**<br>$1_B$ **MCDS_S**, |
| **ALy (y=02-05,10-15,23-31)** | y | rh | **Alarm y** |
| **ALy (y=06)** | y | rh | **Alarm y**<br>$1_B$ **EMEM_XTMRAM_S**, |
| **ALy (y=07)** | y | rh | **Alarm y**<br>$1_B$ **EMEM_CTRL_S**, |
| **ALy (y=08)** | y | rh | **Alarm y**<br>$1_B$ **EMEM0_S**, |
| **ALy (y=09)** | y | rh | **Alarm y**<br>$1_B$ **EMEM1_S**, |
| **ALy (y=16)** | y | rh | **Alarm y**<br>$1_B$ **RIF0_S**, |
| **ALy (y=17)** | y | rh | **Alarm y**<br>$1_B$ **RIF1_S**, |
| **ALy (y=18)** | y | rh | **Alarm y**<br>$1_B$ **SPU0_S**, |
| **ALy (y=19)** | y | rh | **Alarm y**<br>$1_B$ **SPU_CFG0_S**, |
| **ALy (y=20)** | y | rh | **Alarm y**<br>$1_B$ **SPU1_S**, |
| **ALy (y=21)** | y | rh | **Alarm y**<br>$1_B$ **SPU_CFG1_S**, |
| **ALy (y=22)** | y | rh | **Alarm y**<br>$1_B$ **SPU_LS_SFR_S**, |

**EBCU_ALSTATx (x=1)**
**BCU EDC Alarm Status Register x** **($0060_H$+x*4)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **AL31** | **AL30** | **AL29** | **AL28** | **AL27** | **AL26** | **AL25** | **AL24** | **AL23** | **AL22** | **AL21** | **AL20** | **AL19** | **AL18** | **AL17** | **AL16** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **AL15** | **AL14** | **AL13** | **AL12** | **AL11** | **AL10** | **AL09** | **AL08** | **AL07** | **AL06** | **AL05** | **AL04** | **AL03** | **AL02** | **AL01** | **AL00** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ALy (y=00-31)** | y | rh | **Alarm y** |

**EBCU_ALSTATx (x=2)**

**BCU EDC Alarm Status Register x** **($0060_H$+x*4)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **AL31** | **AL30** | **AL29** | **AL28** | **AL27** | **AL26** | **AL25** | **AL24** | **AL23** | **AL22** | **AL21** | **AL20** | **AL19** | **AL18** | **AL17** | **AL16** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **AL15** | **AL14** | **AL13** | **AL12** | **AL11** | **AL10** | **AL09** | **AL08** | **AL07** | **AL06** | **AL05** | **AL04** | **AL03** | **AL02** | **AL01** | **AL00** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ALy (y=00-30)** | y | rh | **Alarm y** |
| **ALy (y=31)** | y | rh | **Alarm y**<br>$1_B$ **EBCU_M**, an EDC error was detected in an active phase of the EBCU Master Interface. |

**EBCU_ALSTATx (x=3)**

**BCU EDC Alarm Status Register x** **($0060_H$+x*4)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **AL31** | **AL30** | **AL29** | **AL28** | **AL27** | **AL26** | **AL25** | **AL24** | **AL23** | **AL22** | **AL21** | **AL20** | **AL19** | **AL18** | **AL17** | **AL16** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **AL15** | **AL14** | **AL13** | **AL12** | **AL11** | **AL10** | **AL09** | **AL08** | **AL07** | **AL06** | **AL05** | **AL04** | **AL03** | **AL02** | **AL01** | **AL00** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ALy (y=00)** | y | rh | **Alarm y**<br>$1_B$ **A_EN**, multiple output enables active: A_EN_N (Master) |
| **ALy (y=01)** | y | rh | **Alarm y**<br>$1_B$ **ABORT_EN**, multiple output enables active: ABORT_EN_N (Master) |
| **ALy (y=02)** | y | rh | **Alarm y**<br>$1_B$ **ACK_EN**, multiple output enables active: ACK_EN_N (Default Master and Slave) |
| **ALy (y=03)** | y | rh | **Alarm y**<br>$1_B$ **D_EN**, multiple output enables active: D_EN_N (Master and Slave) |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ALy (y=04-15,17-18,20-21,23-31)** | y | rh | **Alarm y** |
| **ALy (y=16)** | y | rh | **Alarm y**<br>$1_B$ **IOC32P_M**, |
| **ALy (y=19)** | y | rh | **Alarm y**<br>$1_B$ **IOC32E_M**, |
| **ALy (y=22)** | y | rh | **Alarm y**<br>$1_B$ **SFI_S2F_M**, |

## 4.7.4 Connectivity

### 4.7.4.1 SBCU Connectivity

**Table 15 Connections of SBCU**

| Interface Signals | connects | | Description |
|---|---|---|---|
| SBCU:INT | to | INT:sbcu_INT | Bus Control Unit SPB Service Request |

### 4.7.4.2 EBCU Connectivity

**Table 16 Connections of EBCU**

| Interface Signals | connects | | Description |
|---|---|---|---|
| EBCU:INT | to | INT:bbbcu_INT | Bus Control Unit BBB Service Request |

## 4.7.5 Revision History

**Table 17 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V1.2.7** | | |
| | No functional change. | |
| **V1.2.8** | | |
| – | No functional changes. | – |
| **V1.2.9** | | |
| **Page 25** | Missing reserved bit field added | |

# 5 CPU Subsystem (CPU)

This chapter describes the CPU subsystem module of the TC35x.

## 5.1 TC35x Specific Configuration

No product specific configuration for CPU

## 5.2 TC35x Specific Register Set

### Register Address Space Table

**Table 18 Register Address Space - CPU**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| (CPU0) | $70000000_H$ | $7003BFFF_H$ | Data ScratchPad RAM interface |
| | $7003C000_H$ | $7003FFFF_H$ | Data Cache RAM interface |
| | $700C0000_H$ | $700C17FF_H$ | Data Cache Tag RAM interface |
| | $70100000_H$ | $7010FFFF_H$ | Program ScratchPad RAM interface |
| | $70110000_H$ | $70117FFF_H$ | Program Cache RAM interface |
| | $701C0000_H$ | $701C2FFF_H$ | Program Cache TAG RAM interface |
| | $90000000_H$ | $9000FFFF_H$ | DLMU RAM interface (cached) |
| | $B0000000_H$ | $B000FFFF_H$ | DLMU RAM interface (non-cached) |
| CPU0 | $F8800000_H$ | $F881FFFF_H$ | SRI slave interface for SFR+CSFR |
| (CPU1) | $60000000_H$ | $6003BFFF_H$ | Data ScratchPad RAM interface |
| | $6003C000_H$ | $6003FFFF_H$ | Data Cache RAM interface |
| | $600C0000_H$ | $600C17FF_H$ | Data Cache Tag RAM interface |
| | $60100000_H$ | $6010FFFF_H$ | Program ScratchPad RAM interface |
| | $60110000_H$ | $60117FFF_H$ | Program Cache RAM interface |
| | $601C0000_H$ | $601C2FFF_H$ | Program Cache TAG RAM interface |
| | $90010000_H$ | $9001FFFF_H$ | DLMU RAM interface (cached) |
| | $B0010000_H$ | $B001FFFF_H$ | DLMU RAM interface (non-cached) |
| CPU1 | $F8820000_H$ | $F883FFFF_H$ | SRI slave interface for SFR+CSFR |
| (CPU2) | $50000000_H$ | $50017FFF_H$ | Data ScratchPad RAM interface |
| | $50018000_H$ | $5001BFFF_H$ | Data Cache RAM interface |
| | $500C0000_H$ | $500C17FF_H$ | Data Cache Tag RAM interface |
| | $50100000_H$ | $5010FFFF_H$ | Program ScratchPad RAM interface |
| | $50110000_H$ | $50117FFF_H$ | Program Cache RAM interface |
| | $501C0000_H$ | $501C2FFF_H$ | Program Cache TAG RAM interface |
| | $90020000_H$ | $9002FFFF_H$ | DLMU RAM interface (cached) |
| | $B0020000_H$ | $B002FFFF_H$ | DLMU RAM interface (non-cached) |
| CPU2 | $F8840000_H$ | $F885FFFF_H$ | SRI slave interface for SFR+CSFR |

## Register Overview Table

### Register Overview Tables of CPU

**Table 19 Register Overview - CPU0 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU0_FLASHCON0 | CPUx Flash Configuration Register 0 | $01100_H$ | 25 |
| CPU0_FLASHCON1 | CPUx Flash Configuration Register 1 | $01104_H$ | See Family Spec |
| CPU0_FLASHCON2 | CPUx Flash Configuration Register 2 | $01108_H$ | See Family Spec |
| CPU0_FLASHCON3 | CPUx Flash Configuration Register 3 | $0110C_H$ | See Family Spec |
| CPU0_FLASHCON4 | CPUx Flash Configuration Register 4 | $01110_H$ | See Family Spec |
| CPU0_KRST0 | CPUx Reset Register 0 | $0D000_H$ | See Family Spec |
| CPU0_KRST1 | CPUx Reset Register 1 | $0D004_H$ | See Family Spec |
| CPU0_KRSTCLR | CPUx Reset Clear Register | $0D008_H$ | See Family Spec |
| CPU0_SPR_SPROT_RGNLAi (i=0-7) | CPUx Safety Protection SPR Region Lower Address Register i | $0E000_H$+i*$10_H$ | See Family Spec |
| CPU0_SPR_SPROT_RGNUAi (i=0-7) | CPUx Safety Protection SPR Region Upper Address Register i | $0E004_H$+i*$10_H$ | See Family Spec |
| CPU0_SPR_SPROT_RGNACCENAi_W (i=0-7) | CPUx Safety Protection SPR Region Write Access Enable Register Ai | $0E008_H$+i*$10_H$ | See Family Spec |
| CPU0_SPR_SPROT_RGNACCENBi_W (i=0-7) | CPUx Safety Protection SPR Region Write Access Enable Register Bi | $0E00C_H$+i*$10_H$ | See Family Spec |
| CPU0_SPR_SPROT_RGNACCENAi_R (i=0-7) | CPUx Safety Protection SPR Region Read Access Enable Register Ai | $0E088_H$+i*$10_H$ | See Family Spec |

**Table 19 Register Overview - CPU0 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU0_SPR_SPROT_RGNACCENBi_R (i=0-7) | CPUx Safety Protection SPR Region Read Access Enable Register Bi | $0E08C_H$+i*$10_H$ | See Family Spec |
| CPU0_SFR_SPROT_ACCENA_W | CPUx Safety Protection Register Access Enable Register A | $0E100_H$ | See Family Spec |
| CPU0_SFR_SPROT_ACCENB_W | CPUx Safety Protection Region Access Enable Register B | $0E104_H$ | See Family Spec |
| CPU0_LPB_SPROT_ACCENA_R | CPUx Safety Protection Region LPB Read Access Enable Register A | $0E110_H$ | See Family Spec |
| CPU0_LPB_SPROT_ACCENB_R | CPUx Safety Protection Region LPB Read Access Enable Register B | $0E114_H$ | See Family Spec |
| CPU0_DLMU_SPROT_RGNLAi (i=0-7) | CPUx Safety Protection DLMU Region Lower Address Register i | $0E200_H$+i*$10_H$ | See Family Spec |
| CPU0_DLMU_SPROT_RGNUAi (i=0-7) | CPUx Safety protection DLMU Region Upper Address Register i | $0E204_H$+i*$10_H$ | See Family Spec |
| CPU0_DLMU_SPROT_RGNACCENAi_W (i=0-7) | CPUx Safety Protection Region DLMU Write Access Enable Register Ai | $0E208_H$+i*$10_H$ | See Family Spec |
| CPU0_DLMU_SPROT_RGNACCENBi_W (i=0-7) | CPUx Safety Protection Region DLMU Write Access Enable Register Bi | $0E20C_H$+i*$10_H$ | See Family Spec |
| CPU0_DLMU_SPROT_RGNACCENAi_R (i=0-7) | CPUx Safety Protection Region DLMU Read Access Enable Register Ai | $0E288_H$+i*$10_H$ | See Family Spec |
| CPU0_DLMU_SPROT_RGNACCENBi_R (i=0-7) | CPUx Safety Protection Region DLMU Read Access Enable Register Bi | $0E28C_H$+i*$10_H$ | See Family Spec |
| CPU0_OSEL | CPUx Overlay Range Select Register | $0FB00_H$ | See Family Spec |
| CPU0_RABRi (i=0-31) | CPUx Redirected Address Base Register i | $0FB10_H$+i*12 | See Family Spec |
| CPU0_OTARi (i=0-31) | CPUx Overlay Target Address Register i | $0FB14_H$+i*12 | See Family Spec |

**Table 19 Register Overview - CPU0 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU0_OMASKi (i=0-31) | CPUx Overlay Mask Register i | $0FB18_H$+i*12 | See Family Spec |
| CPU0_SEGEN | CPUx SRI Error Generation Register | $11030_H$ | See Family Spec |
| CPU0_TASK_ASI | CPUx Task Address Space Identifier Register | $18004_H$ | See Family Spec |
| CPU0_PMA0 | CPUx Data Access CacheabilityRegister | $18100_H$ | See Family Spec |
| CPU0_PMA1 | CPUx Code Access CacheabilityRegister | $18104_H$ | See Family Spec |
| CPU0_PMA2 | CPUx Peripheral Space Identifier register | $18108_H$ | See Family Spec |
| CPU0_DCON2 | CPUx Data Control Register 2 | $19000_H$ | See Family Spec |
| CPU0_SMACON | CPUx SIST Mode Access Control Register | $1900C_H$ | See Family Spec |
| CPU0_DSTR | CPUx Data Synchronous Trap Register | $19010_H$ | See Family Spec |
| CPU0_DATR | CPUx Data Asynchronous Trap Register | $19018_H$ | See Family Spec |
| CPU0_DEADD | CPUx Data Error Address Register | $1901C_H$ | See Family Spec |
| CPU0_DIEAR | CPUx Data Integrity Error Address Register | $19020_H$ | See Family Spec |
| CPU0_DIETR | CPUx Data Integrity Error Trap Register | $19024_H$ | See Family Spec |
| CPU0_DCON0 | CPUx Data Memory Control Register | $19040_H$ | See Family Spec |

**Table 19 Register Overview - CPU0 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU0_PSTR | CPUx Program Synchronous Trap Register | $19200_H$ | See Family Spec |
| CPU0_PCON1 | CPUx Program Control 1 | $19204_H$ | See Family Spec |
| CPU0_PCON2 | CPUx Program Control 2 | $19208_H$ | See Family Spec |
| CPU0_PCON0 | CPUx Program Control 0 | $1920C_H$ | See Family Spec |
| CPU0_PIEAR | CPUx Program Integrity Error Address Register | $19210_H$ | See Family Spec |
| CPU0_PIETR | CPUx Program Integrity Error Trap Register | $19214_H$ | See Family Spec |
| CPU0_COMPAT | CPUx Compatibility Control Register | $19400_H$ | See Family Spec |
| CPU0_FPU_TRAP_CON | CPUx Trap Control Register | $1A000_H$ | See Family Spec |
| CPU0_FPU_TRAP_PC | CPUx Trapping Instruction Program Counter Register | $1A004_H$ | See Family Spec |
| CPU0_FPU_TRAP_OPC | CPUx Trapping Instruction Opcode Register | $1A008_H$ | See Family Spec |
| CPU0_FPU_TRAP_SRC1 | CPUx Trapping Instruction Operand Register | $1A010_H$ | See Family Spec |
| CPU0_FPU_TRAP_SRC2 | CPUx Trapping Instruction Operand Register | $1A014_H$ | See Family Spec |
| CPU0_FPU_TRAP_SRC3 | CPUx Trapping Instruction Operand Register | $1A018_H$ | See Family Spec |
| CPU0_DPRy_L (y=0-17) | CPUx Data Protection Range y, Lower Bound Register | $1C000_H$+y*8 | See Family Spec |

**Table 19 Register Overview - CPU0 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU0_DPRy_U (y=0-17) | CPUx Data Protection Range y, Upper Bound Register | $1C004_H$+y*8 | See Family Spec |
| CPU0_CPRy_L (y=0-9) | CPUx Code Protection Range y Lower Bound Register | $1D000_H$+y*8 | See Family Spec |
| CPU0_CPRy_U (y=0-9) | CPUx Code Protection Range y Upper Bound Register | $1D004_H$+y*8 | See Family Spec |
| CPU0_CPXE_y (y=0-3) | CPUx Code Protection Execute Enable Register Set y | $1E000_H$+y*4 | See Family Spec |
| CPU0_DPRE_y (y=0-3) | CPUx Data Protection Read Enable Register Set y | $1E010_H$+y*4 | See Family Spec |
| CPU0_DPWE_y (y=0-3) | CPUx Data Protection Write Enable Register Set y | $1E020_H$+y*4 | See Family Spec |
| CPU0_CPXE_y (y=4-5) | CPUx Code Protection Execute Enable Register Set y | $1E040_H$+(y-4)*4 | See Family Spec |
| CPU0_DPRE_y (y=4-5) | CPUx Data Protection Read Enable Register Set y | $1E050_H$+(y-4)*4 | See Family Spec |
| CPU0_DPWE_y (y=4-5) | CPUx Data Protection Write Enable Register Set y | $1E060_H$+(y-4)*4 | See Family Spec |
| CPU0_TPS_CON | CPUx Temporal Protection System Control Register | $1E400_H$ | See Family Spec |
| CPU0_TPS_TIMERy (y=0-2) | CPUx Temporal Protection System Timer Register y | $1E404_H$+y*4 | See Family Spec |
| CPU0_TPS_EXTIM_ENTRY_LVAL | CPUx Exception Entry Timer Load Value | $1E440_H$ | See Family Spec |
| CPU0_TPS_EXTIM_ENTRY_CVAL | CPUx Exception Entry Timer Current Value | $1E444_H$ | See Family Spec |
| CPU0_TPS_EXTIM_EXIT_LVAL | CPUx Exception Exit  Timer Load Value | $1E448_H$ | See Family Spec |

**Table 19 Register Overview - CPU0 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU0_TPS_EXTIM_EXIT_CVAL | CPUx Exception Exit Timer Current Value | $1E44C_H$ | See Family Spec |
| CPU0_TPS_EXTIM_CLASS_EN | CPUx Exception Timer Class Enable Register | $1E450_H$ | See Family Spec |
| CPU0_TPS_EXTIM_STAT | CPUx Exception Timer Status Register | $1E454_H$ | See Family Spec |
| CPU0_TPS_EXTIM_FCX | CPUx Exception Timer FCX Register | $1E458_H$ | See Family Spec |
| CPU0_TRiEVT (i=0-7) | CPUx Trigger Event i | $1F000_H$+i*8 | See Family Spec |
| CPU0_TRiADR (i=0-7) | CPUx Trigger Address i | $1F004_H$+i*8 | See Family Spec |
| CPU0_CCTRL | CPUx Counter Control | $1FC00_H$ | See Family Spec |
| CPU0_CCNT | CPUx CPU Clock Cycle Count | $1FC04_H$ | See Family Spec |
| CPU0_ICNT | CPUx Instruction Count | $1FC08_H$ | See Family Spec |
| CPU0_M1CNT | CPUx Multi-Count Register 1 | $1FC0C_H$ | See Family Spec |
| CPU0_M2CNT | CPUx Multi-Count Register 2 | $1FC10_H$ | See Family Spec |
| CPU0_M3CNT | CPUx Multi-Count Register 3 | $1FC14_H$ | See Family Spec |
| CPU0_DBGSR | CPUx Debug Status Register | $1FD00_H$ | See Family Spec |
| CPU0_EXEVT | CPUx External Event Register | $1FD08_H$ | See Family Spec |

**Table 19 Register Overview - CPU0 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU0_CREVT | CPUx Core Register Access Event | $1FD0C_H$ | See Family Spec |
| CPU0_SWEVT | CPUx Software Debug Event | $1FD10_H$ | See Family Spec |
| CPU0_TRIG_ACC | CPUx TriggerAddressx | $1FD30_H$ | See Family Spec |
| CPU0_DMS | CPUx Debug Monitor Start Address | $1FD40_H$ | See Family Spec |
| CPU0_DCX | CPUx Debug Context Save Area Pointer | $1FD44_H$ | See Family Spec |
| CPU0_DBGTCR | CPUx Debug Trap Control Register | $1FD48_H$ | See Family Spec |
| CPU0_PCXI | CPUx Previous Context Information Register | $1FE00_H$ | See Family Spec |
| CPU0_PSW | CPUx Program Status Word | $1FE04_H$ | See Family Spec |
| CPU0_PC | CPUx Program Counter | $1FE08_H$ | See Family Spec |
| CPU0_SYSCON | CPUx System Configuration Register | $1FE14_H$ | See Family Spec |
| CPU0_CPU_ID | CPUx Identification Register TC1.6.2P | $1FE18_H$ | See Family Spec |
| CPU0_CORE_ID | CPUx Core Identification Register | $1FE1C_H$ | See Family Spec |
| CPU0_BIV | CPUx Base Interrupt Vector Table Pointer | $1FE20_H$ | See Family Spec |
| CPU0_BTV | CPUx Base Trap Vector Table Pointer | $1FE24_H$ | See Family Spec |

**Table 19 Register Overview - CPU0 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU0_ISP | CPUx Interrupt Stack Pointer | $1FE28_H$ | See Family Spec |
| CPU0_ICR | CPUx Interrupt Control Register | $1FE2C_H$ | See Family Spec |
| CPU0_FCX | CPUx Free CSA List Head Pointer | $1FE38_H$ | See Family Spec |
| CPU0_LCX | CPUx Free CSA List Limit Pointer | $1FE3C_H$ | See Family Spec |
| CPU0_CUS_ID | CPUx Customer ID register | $1FE50_H$ | See Family Spec |
| CPU0_Dy (y=0-15) | CPUx Data General Purpose Register y | $1FF00_H$+y*4 | See Family Spec |
| CPU0_Ay (y=0-15) | CPUx Address General Purpose Register y | $1FF80_H$+y*4 | See Family Spec |

**Table 20 Register Overview - CPU1 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU1_FLASHCON0 | CPUx Flash Configuration Register 0 | $01100_H$ | 25 |
| CPU1_FLASHCON1 | CPUx Flash Configuration Register 1 | $01104_H$ | See Family Spec |
| CPU1_FLASHCON2 | CPUx Flash Configuration Register 2 | $01108_H$ | See Family Spec |
| CPU1_FLASHCON3 | CPUx Flash Configuration Register 3 | $0110C_H$ | See Family Spec |
| CPU1_FLASHCON4 | CPUx Flash Configuration Register 4 | $01110_H$ | See Family Spec |
| CPU1_KRST0 | CPUx Reset Register 0 | $0D000_H$ | See Family Spec |

**Table 20 Register Overview - CPU1 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU1_KRST1 | CPUx Reset Register 1 | $0D004_H$ | See Family Spec |
| CPU1_KRSTCLR | CPUx Reset Clear Register | $0D008_H$ | See Family Spec |
| CPU1_SPR_SPROT_RGNLAi (i=0-7) | CPUx Safety Protection SPR Region Lower Address Register i | $0E000_H$+i*$10_H$ | See Family Spec |
| CPU1_SPR_SPROT_RGNUAi (i=0-7) | CPUx Safety Protection SPR Region Upper Address Register i | $0E004_H$+i*$10_H$ | See Family Spec |
| CPU1_SPR_SPROT_RGNACCENAi_W (i=0-7) | CPUx Safety Protection SPR Region Write Access Enable Register Ai | $0E008_H$+i*$10_H$ | See Family Spec |
| CPU1_SPR_SPROT_RGNACCENBi_W (i=0-7) | CPUx Safety Protection SPR Region Write Access Enable Register Bi | $0E00C_H$+i*$10_H$ | See Family Spec |
| CPU1_SPR_SPROT_RGNACCENAi_R (i=0-7) | CPUx Safety Protection SPR Region Read Access Enable Register Ai | $0E088_H$+i*$10_H$ | See Family Spec |
| CPU1_SPR_SPROT_RGNACCENBi_R (i=0-7) | CPUx Safety Protection SPR Region Read Access Enable Register Bi | $0E08C_H$+i*$10_H$ | See Family Spec |
| CPU1_SFR_SPROT_ACCENA_W | CPUx Safety Protection Register Access Enable Register A | $0E100_H$ | See Family Spec |
| CPU1_SFR_SPROT_ACCENB_W | CPUx Safety Protection Region Access Enable Register B | $0E104_H$ | See Family Spec |
| CPU1_LPB_SPROT_ACCENA_R | CPUx Safety Protection Region LPB Read Access Enable Register A | $0E110_H$ | See Family Spec |
| CPU1_LPB_SPROT_ACCENB_R | CPUx Safety Protection Region LPB Read Access Enable Register B | $0E114_H$ | See Family Spec |
| CPU1_DLMU_SPROT_RGNLAi (i=0-7) | CPUx Safety Protection DLMU Region Lower Address Register i | $0E200_H$+i*$10_H$ | See Family Spec |
| CPU1_DLMU_SPROT_RGNUAi (i=0-7) | CPUx Safety protection DLMU Region Upper Address Register i | $0E204_H$+i*$10_H$ | See Family Spec |

**Table 20 Register Overview - CPU1 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU1_DLMU_SPROT_RGNACCENAi_W (i=0-7) | CPUx Safety Protection Region DLMU Write Access Enable Register Ai | $0E208_H$+i*$10_H$ | See Family Spec |
| CPU1_DLMU_SPROT_RGNACCENBi_W (i=0-7) | CPUx Safety Protection Region DLMU Write Access Enable Register Bi | $0E20C_H$+i*$10_H$ | See Family Spec |
| CPU1_DLMU_SPROT_RGNACCENAi_R (i=0-7) | CPUx Safety Protection Region DLMU Read Access Enable Register Ai | $0E288_H$+i*$10_H$ | See Family Spec |
| CPU1_DLMU_SPROT_RGNACCENBi_R (i=0-7) | CPUx Safety Protection Region DLMU Read Access Enable Register Bi | $0E28C_H$+i*$10_H$ | See Family Spec |
| CPU1_OSEL | CPUx Overlay Range Select Register | $0FB00_H$ | See Family Spec |
| CPU1_RABRi (i=0-31) | CPUx Redirected Address Base Register i | $0FB10_H$+i*12 | See Family Spec |
| CPU1_OTARi (i=0-31) | CPUx Overlay Target Address Register i | $0FB14_H$+i*12 | See Family Spec |
| CPU1_OMASKi (i=0-31) | CPUx Overlay Mask Register i | $0FB18_H$+i*12 | See Family Spec |
| CPU1_SEGEN | CPUx SRI Error Generation Register | $11030_H$ | See Family Spec |
| CPU1_TASK_ASI | CPUx Task Address Space Identifier Register | $18004_H$ | See Family Spec |
| CPU1_PMA0 | CPUx Data Access CacheabilityRegister | $18100_H$ | See Family Spec |
| CPU1_PMA1 | CPUx Code Access CacheabilityRegister | $18104_H$ | See Family Spec |
| CPU1_PMA2 | CPUx Peripheral Space Identifier register | $18108_H$ | See Family Spec |
| CPU1_DCON2 | CPUx Data Control Register 2 | $19000_H$ | See Family Spec |

**Table 20 Register Overview - CPU1 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU1_SMACON | CPUx SIST Mode Access Control Register | $1900C_H$ | See Family Spec |
| CPU1_DSTR | CPUx Data Synchronous Trap Register | $19010_H$ | See Family Spec |
| CPU1_DATR | CPUx Data Asynchronous Trap Register | $19018_H$ | See Family Spec |
| CPU1_DEADD | CPUx Data Error Address Register | $1901C_H$ | See Family Spec |
| CPU1_DIEAR | CPUx Data Integrity Error Address Register | $19020_H$ | See Family Spec |
| CPU1_DIETR | CPUx Data Integrity Error Trap Register | $19024_H$ | See Family Spec |
| CPU1_DCON0 | CPUx Data Memory Control Register | $19040_H$ | See Family Spec |
| CPU1_PSTR | CPUx Program Synchronous Trap Register | $19200_H$ | See Family Spec |
| CPU1_PCON1 | CPUx Program Control 1 | $19204_H$ | See Family Spec |
| CPU1_PCON2 | CPUx Program Control 2 | $19208_H$ | See Family Spec |
| CPU1_PCON0 | CPUx Program Control 0 | $1920C_H$ | See Family Spec |
| CPU1_PIEAR | CPUx Program Integrity Error Address Register | $19210_H$ | See Family Spec |
| CPU1_PIETR | CPUx Program Integrity Error Trap Register | $19214_H$ | See Family Spec |
| CPU1_COMPAT | CPUx Compatibility Control Register | $19400_H$ | See Family Spec |

**Table 20 Register Overview - CPU1 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU1_FPU_TRAP_CON | CPUx Trap Control Register | $1A000_H$ | See Family Spec |
| CPU1_FPU_TRAP_PC | CPUx Trapping Instruction Program Counter Register | $1A004_H$ | See Family Spec |
| CPU1_FPU_TRAP_OPC | CPUx Trapping Instruction Opcode Register | $1A008_H$ | See Family Spec |
| CPU1_FPU_TRAP_SRC1 | CPUx Trapping Instruction Operand Register | $1A010_H$ | See Family Spec |
| CPU1_FPU_TRAP_SRC2 | CPUx Trapping Instruction Operand Register | $1A014_H$ | See Family Spec |
| CPU1_FPU_TRAP_SRC3 | CPUx Trapping Instruction Operand Register | $1A018_H$ | See Family Spec |
| CPU1_DPRy_L (y=0-17) | CPUx Data Protection Range y, Lower Bound Register | $1C000_H$+y*8 | See Family Spec |
| CPU1_DPRy_U (y=0-17) | CPUx Data Protection Range y, Upper Bound Register | $1C004_H$+y*8 | See Family Spec |
| CPU1_CPRy_L (y=0-9) | CPUx Code Protection Range y Lower Bound Register | $1D000_H$+y*8 | See Family Spec |
| CPU1_CPRy_U (y=0-9) | CPUx Code Protection Range y Upper Bound Register | $1D004_H$+y*8 | See Family Spec |
| CPU1_CPXE_y (y=0-3) | CPUx Code Protection Execute Enable Register Set y | $1E000_H$+y*4 | See Family Spec |
| CPU1_DPRE_y (y=0-3) | CPUx Data Protection Read Enable Register Set y | $1E010_H$+y*4 | See Family Spec |
| CPU1_DPWE_y (y=0-3) | CPUx Data Protection Write Enable Register Set y | $1E020_H$+y*4 | See Family Spec |
| CPU1_CPXE_y (y=4-5) | CPUx Code Protection Execute Enable Register Set y | $1E040_H$+(y-4)*4 | See Family Spec |

**Table 20 Register Overview - CPU1 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU1_DPRE_y (y=4-5) | CPUx Data Protection Read Enable Register Set y | $1E050_H$+(y-4)*4 | See Family Spec |
| CPU1_DPWE_y (y=4-5) | CPUx Data Protection Write Enable Register Set y | $1E060_H$+(y-4)*4 | See Family Spec |
| CPU1_TPS_CON | CPUx Temporal Protection System Control Register | $1E400_H$ | See Family Spec |
| CPU1_TPS_TIMERy (y=0-2) | CPUx Temporal Protection System Timer Register y | $1E404_H$+y*4 | See Family Spec |
| CPU1_TPS_EXTIM_ENTRY_LVAL | CPUx Exception Entry Timer Load Value | $1E440_H$ | See Family Spec |
| CPU1_TPS_EXTIM_ENTRY_CVAL | CPUx Exception Entry Timer Current Value | $1E444_H$ | See Family Spec |
| CPU1_TPS_EXTIM_EXIT_LVAL | CPUx Exception Exit Timer Load Value | $1E448_H$ | See Family Spec |
| CPU1_TPS_EXTIM_EXIT_CVAL | CPUx Exception Exit Timer Current Value | $1E44C_H$ | See Family Spec |
| CPU1_TPS_EXTIM_CLASS_EN | CPUx Exception Timer Class Enable Register | $1E450_H$ | See Family Spec |
| CPU1_TPS_EXTIM_STAT | CPUx Exception Timer Status Register | $1E454_H$ | See Family Spec |
| CPU1_TPS_EXTIM_FCX | CPUx Exception Timer FCX Register | $1E458_H$ | See Family Spec |
| CPU1_TRiEVT (i=0-7) | CPUx Trigger Event i | $1F000_H$+i*8 | See Family Spec |
| CPU1_TRiADR (i=0-7) | CPUx Trigger Address i | $1F004_H$+i*8 | See Family Spec |
| CPU1_CCTRL | CPUx Counter Control | $1FC00_H$ | See Family Spec |

**Table 20 Register Overview - CPU1 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU1_CCNT | CPUx CPU Clock Cycle Count | $1FC04_H$ | See Family Spec |
| CPU1_ICNT | CPUx Instruction Count | $1FC08_H$ | See Family Spec |
| CPU1_M1CNT | CPUx Multi-Count Register 1 | $1FC0C_H$ | See Family Spec |
| CPU1_M2CNT | CPUx Multi-Count Register 2 | $1FC10_H$ | See Family Spec |
| CPU1_M3CNT | CPUx Multi-Count Register 3 | $1FC14_H$ | See Family Spec |
| CPU1_DBGSR | CPUx Debug Status Register | $1FD00_H$ | See Family Spec |
| CPU1_EXEVT | CPUx External Event Register | $1FD08_H$ | See Family Spec |
| CPU1_CREVT | CPUx Core Register Access Event | $1FD0C_H$ | See Family Spec |
| CPU1_SWEVT | CPUx Software Debug Event | $1FD10_H$ | See Family Spec |
| CPU1_TRIG_ACC | CPUx TriggerAddressx | $1FD30_H$ | See Family Spec |
| CPU1_DMS | CPUx Debug Monitor Start Address | $1FD40_H$ | See Family Spec |
| CPU1_DCX | CPUx Debug Context Save Area Pointer | $1FD44_H$ | See Family Spec |
| CPU1_DBGTCR | CPUx Debug Trap Control Register | $1FD48_H$ | See Family Spec |
| CPU1_PCXI | CPUx Previous Context Information Register | $1FE00_H$ | See Family Spec |

**Table 20 Register Overview - CPU1 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU1_PSW | CPUx Program Status Word | $1FE04_H$ | See Family Spec |
| CPU1_PC | CPUx Program Counter | $1FE08_H$ | See Family Spec |
| CPU1_SYSCON | CPUx System Configuration Register | $1FE14_H$ | See Family Spec |
| CPU1_CPU_ID | CPUx Identification Register TC1.6.2P | $1FE18_H$ | See Family Spec |
| CPU1_CORE_ID | CPUx Core Identification Register | $1FE1C_H$ | See Family Spec |
| CPU1_BIV | CPUx Base Interrupt Vector Table Pointer | $1FE20_H$ | See Family Spec |
| CPU1_BTV | CPUx Base Trap Vector Table Pointer | $1FE24_H$ | See Family Spec |
| CPU1_ISP | CPUx Interrupt Stack Pointer | $1FE28_H$ | See Family Spec |
| CPU1_ICR | CPUx Interrupt Control Register | $1FE2C_H$ | See Family Spec |
| CPU1_FCX | CPUx Free CSA List Head Pointer | $1FE38_H$ | See Family Spec |
| CPU1_LCX | CPUx Free CSA List Limit Pointer | $1FE3C_H$ | See Family Spec |
| CPU1_CUS_ID | CPUx Customer ID register | $1FE50_H$ | See Family Spec |
| CPU1_Dy (y=0-15) | CPUx Data General Purpose Register y | $1FF00_H$+y*4 | See Family Spec |
| CPU1_Ay (y=0-15) | CPUx Address General Purpose Register y | $1FF80_H$+y*4 | See Family Spec |

**Table 21 Register Overview - CPU2 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU2_FLASHCON0 | CPUx Flash Configuration Register 0 | $01100_H$ | See Family Spec |
| CPU2_FLASHCON1 | CPUx Flash Configuration Register 1 | $01104_H$ | See Family Spec |
| CPU2_FLASHCON2 | CPUx Flash Configuration Register 2 | $01108_H$ | See Family Spec |
| CPU2_FLASHCON3 | CPUx Flash Configuration Register 3 | $0110C_H$ | See Family Spec |
| CPU2_FLASHCON4 | CPUx Flash Configuration Register 4 | $01110_H$ | See Family Spec |
| CPU2_KRST0 | CPUx Reset Register 0 | $0D000_H$ | See Family Spec |
| CPU2_KRST1 | CPUx Reset Register 1 | $0D004_H$ | See Family Spec |
| CPU2_KRSTCLR | CPUx Reset Clear Register | $0D008_H$ | See Family Spec |
| CPU2_SPR_SPROT_RGNLAi (i=0-7) | CPUx Safety Protection SPR Region Lower Address Register i | $0E000_H+i*10_H$ | See Family Spec |
| CPU2_SPR_SPROT_RGNUAi (i=0-7) | CPUx Safety Protection SPR Region Upper Address Register i | $0E004_H+i*10_H$ | See Family Spec |
| CPU2_SPR_SPROT_RGNACCENAi_W (i=0-7) | CPUx Safety Protection SPR Region Write Access Enable Register Ai | $0E008_H+i*10_H$ | See Family Spec |
| CPU2_SPR_SPROT_RGNACCENBi_W (i=0-7) | CPUx Safety Protection SPR Region Write Access Enable Register Bi | $0E00C_H+i*10_H$ | See Family Spec |
| CPU2_SPR_SPROT_RGNACCENAi_R (i=0-7) | CPUx Safety Protection SPR Region Read Access Enable Register Ai | $0E088_H+i*10_H$ | See Family Spec |
| CPU2_SPR_SPROT_RGNACCENBi_R (i=0-7) | CPUx Safety Protection SPR Region Read Access Enable Register Bi | $0E08C_H+i*10_H$ | See Family Spec |

**Table 21 Register Overview - CPU2 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU2_SFR_SPROT_ACCENA_W | CPUx Safety Protection Register Access Enable Register A | $0E100_H$ | See Family Spec |
| CPU2_SFR_SPROT_ACCENB_W | CPUx Safety Protection Region Access Enable Register B | $0E104_H$ | See Family Spec |
| CPU2_LPB_SPROT_ACCENA_R | CPUx Safety Protection Region LPB Read Access Enable Register A | $0E110_H$ | See Family Spec |
| CPU2_LPB_SPROT_ACCENB_R | CPUx Safety Protection Region LPB Read Access Enable Register B | $0E114_H$ | See Family Spec |
| CPU2_DLMU_SPROT_RGNLAi (i=0-7) | CPUx Safety Protection DLMU Region Lower Address Register i | $0E200_H+i*10_H$ | See Family Spec |
| CPU2_DLMU_SPROT_RGNUAi (i=0-7) | CPUx Safety protection DLMU Region Upper Address Register i | $0E204_H+i*10_H$ | See Family Spec |
| CPU2_DLMU_SPROT_RGNACCENAi_W (i=0-7) | CPUx Safety Protection Region DLMU Write Access Enable Register Ai | $0E208_H+i*10_H$ | See Family Spec |
| CPU2_DLMU_SPROT_RGNACCENBi_W (i=0-7) | CPUx Safety Protection Region DLMU Write Access Enable Register Bi | $0E20C_H+i*10_H$ | See Family Spec |
| CPU2_DLMU_SPROT_RGNACCENAi_R (i=0-7) | CPUx Safety Protection Region DLMU Read Access Enable Register Ai | $0E288_H+i*10_H$ | See Family Spec |
| CPU2_DLMU_SPROT_RGNACCENBi_R (i=0-7) | CPUx Safety Protection Region DLMU Read Access Enable Register Bi | $0E28C_H+i*10_H$ | See Family Spec |
| CPU2_OSEL | CPUx Overlay Range Select Register | $0FB00_H$ | See Family Spec |
| CPU2_RABRi (i=0-31) | CPUx Redirected Address Base Register i | $0FB10_H+i*12$ | See Family Spec |
| CPU2_OTARi (i=0-31) | CPUx Overlay Target Address Register i | $0FB14_H+i*12$ | See Family Spec |
| CPU2_OMASKi (i=0-31) | CPUx Overlay Mask Register i | $0FB18_H+i*12$ | See Family Spec |

**Table 21 Register Overview - CPU2 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU2_SEGEN | CPUx SRI Error Generation Register | $11030_H$ | See Family Spec |
| CPU2_TASK_ASI | CPUx Task Address Space Identifier Register | $18004_H$ | See Family Spec |
| CPU2_PMA0 | CPUx Data Access CacheabilityRegister | $18100_H$ | See Family Spec |
| CPU2_PMA1 | CPUx Code Access CacheabilityRegister | $18104_H$ | See Family Spec |
| CPU2_PMA2 | CPUx Peripheral Space Identifier register | $18108_H$ | See Family Spec |
| CPU2_DCON2 | CPUx Data Control Register 2 | $19000_H$ | See Family Spec |
| CPU2_SMACON | CPUx SIST Mode Access Control Register | $1900C_H$ | See Family Spec |
| CPU2_DSTR | CPUx Data Synchronous Trap Register | $19010_H$ | See Family Spec |
| CPU2_DATR | CPUx Data Asynchronous Trap Register | $19018_H$ | See Family Spec |
| CPU2_DEADD | CPUx Data Error Address Register | $1901C_H$ | See Family Spec |
| CPU2_DIEAR | CPUx Data Integrity Error Address Register | $19020_H$ | See Family Spec |
| CPU2_DIETR | CPUx Data Integrity Error Trap Register | $19024_H$ | See Family Spec |
| CPU2_DCON0 | CPUx Data Memory Control Register | $19040_H$ | See Family Spec |
| CPU2_PSTR | CPUx Program Synchronous Trap Register | $19200_H$ | See Family Spec |

**Table 21 Register Overview - CPU2 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU2_PCON1 | CPUx Program Control 1 | $19204_H$ | See Family Spec |
| CPU2_PCON2 | CPUx Program Control 2 | $19208_H$ | See Family Spec |
| CPU2_PCON0 | CPUx Program Control 0 | $1920C_H$ | See Family Spec |
| CPU2_PIEAR | CPUx Program Integrity Error Address Register | $19210_H$ | See Family Spec |
| CPU2_PIETR | CPUx Program Integrity Error Trap Register | $19214_H$ | See Family Spec |
| CPU2_COMPAT | CPUx Compatibility Control Register | $19400_H$ | See Family Spec |
| CPU2_FPU_TRAP_CON | CPUx Trap Control Register | $1A000_H$ | See Family Spec |
| CPU2_FPU_TRAP_PC | CPUx Trapping Instruction Program Counter Register | $1A004_H$ | See Family Spec |
| CPU2_FPU_TRAP_OPC | CPUx Trapping Instruction Opcode Register | $1A008_H$ | See Family Spec |
| CPU2_FPU_TRAP_SRC1 | CPUx Trapping Instruction Operand Register | $1A010_H$ | See Family Spec |
| CPU2_FPU_TRAP_SRC2 | CPUx Trapping Instruction Operand Register | $1A014_H$ | See Family Spec |
| CPU2_FPU_TRAP_SRC3 | CPUx Trapping Instruction Operand Register | $1A018_H$ | See Family Spec |
| CPU2_DPRy_L (y=0-17) | CPUx Data Protection Range y, Lower Bound Register | $1C000_H$+y*8 | See Family Spec |
| CPU2_DPRy_U (y=0-17) | CPUx Data Protection Range y, Upper Bound Register | $1C004_H$+y*8 | See Family Spec |

**Table 21 Register Overview - CPU2 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU2_CPRy_L (y=0-9) | CPUx Code Protection Range y Lower Bound Register | $1D000_H$+y*8 | See Family Spec |
| CPU2_CPRy_U (y=0-9) | CPUx Code Protection Range y Upper Bound Register | $1D004_H$+y*8 | See Family Spec |
| CPU2_CPXE_y (y=0-3) | CPUx Code Protection Execute Enable Register Set y | $1E000_H$+y*4 | See Family Spec |
| CPU2_DPRE_y (y=0-3) | CPUx Data Protection Read Enable Register Set y | $1E010_H$+y*4 | See Family Spec |
| CPU2_DPWE_y (y=0-3) | CPUx Data Protection Write Enable Register Set y | $1E020_H$+y*4 | See Family Spec |
| CPU2_CPXE_y (y=4-5) | CPUx Code Protection Execute Enable Register Set y | $1E040_H$+(y-4)*4 | See Family Spec |
| CPU2_DPRE_y (y=4-5) | CPUx Data Protection Read Enable Register Set y | $1E050_H$+(y-4)*4 | See Family Spec |
| CPU2_DPWE_y (y=4-5) | CPUx Data Protection Write Enable Register Set y | $1E060_H$+(y-4)*4 | See Family Spec |
| CPU2_TPS_CON | CPUx Temporal Protection System Control Register | $1E400_H$ | See Family Spec |
| CPU2_TPS_TIMERy (y=0-2) | CPUx Temporal Protection System Timer Register y | $1E404_H$+y*4 | See Family Spec |
| CPU2_TPS_EXTIM_ENTRY_LVAL | CPUx Exception Entry Timer Load Value | $1E440_H$ | See Family Spec |
| CPU2_TPS_EXTIM_ENTRY_CVAL | CPUx Exception Entry Timer Current Value | $1E444_H$ | See Family Spec |
| CPU2_TPS_EXTIM_EXIT_LVAL | CPUx Exception Exit Timer Load Value | $1E448_H$ | See Family Spec |
| CPU2_TPS_EXTIM_EXIT_CVAL | CPUx Exception Exit Timer Current Value | $1E44C_H$ | See Family Spec |

**Table 21 Register Overview - CPU2 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU2_TPS_EXTIM_CLASS_EN | CPUx Exception Timer Class Enable Register | $1E450_H$ | See Family Spec |
| CPU2_TPS_EXTIM_STAT | CPUx Exception Timer Status Register | $1E454_H$ | See Family Spec |
| CPU2_TPS_EXTIM_FCX | CPUx Exception Timer FCX Register | $1E458_H$ | See Family Spec |
| CPU2_TRiEVT (i=0-7) | CPUx Trigger Event i | $1F000_H$+i*8 | See Family Spec |
| CPU2_TRiADR (i=0-7) | CPUx Trigger Address i | $1F004_H$+i*8 | See Family Spec |
| CPU2_CCTRL | CPUx Counter Control | $1FC00_H$ | See Family Spec |
| CPU2_CCNT | CPUx CPU Clock Cycle Count | $1FC04_H$ | See Family Spec |
| CPU2_ICNT | CPUx Instruction Count | $1FC08_H$ | See Family Spec |
| CPU2_M1CNT | CPUx Multi-Count Register 1 | $1FC0C_H$ | See Family Spec |
| CPU2_M2CNT | CPUx Multi-Count Register 2 | $1FC10_H$ | See Family Spec |
| CPU2_M3CNT | CPUx Multi-Count Register 3 | $1FC14_H$ | See Family Spec |
| CPU2_DBGSR | CPUx Debug Status Register | $1FD00_H$ | See Family Spec |
| CPU2_EXEVT | CPUx External Event Register | $1FD08_H$ | See Family Spec |
| CPU2_CREVT | CPUx Core Register Access Event | $1FD0C_H$ | See Family Spec |

**Table 21 Register Overview - CPU2 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU2_SWEVT | CPUx Software Debug Event | $1FD10_H$ | See Family Spec |
| CPU2_TRIG_ACC | CPUx TriggerAddressx | $1FD30_H$ | See Family Spec |
| CPU2_DMS | CPUx Debug Monitor Start Address | $1FD40_H$ | See Family Spec |
| CPU2_DCX | CPUx Debug Context Save Area Pointer | $1FD44_H$ | See Family Spec |
| CPU2_DBGTCR | CPUx Debug Trap Control Register | $1FD48_H$ | See Family Spec |
| CPU2_PCXI | CPUx Previous Context Information Register | $1FE00_H$ | See Family Spec |
| CPU2_PSW | CPUx Program Status Word | $1FE04_H$ | See Family Spec |
| CPU2_PC | CPUx Program Counter | $1FE08_H$ | See Family Spec |
| CPU2_SYSCON | CPUx System Configuration Register | $1FE14_H$ | See Family Spec |
| CPU2_CPU_ID | CPUx Identification Register TC1.6.2P | $1FE18_H$ | See Family Spec |
| CPU2_CORE_ID | CPUx Core Identification Register | $1FE1C_H$ | See Family Spec |
| CPU2_BIV | CPUx Base Interrupt Vector Table Pointer | $1FE20_H$ | See Family Spec |
| CPU2_BTV | CPUx Base Trap Vector Table Pointer | $1FE24_H$ | See Family Spec |
| CPU2_ISP | CPUx Interrupt Stack Pointer | $1FE28_H$ | See Family Spec |

**Table 21 Register Overview - CPU2 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CPU2_ICR | CPUx Interrupt Control Register | $1FE2C_H$ | See Family Spec |
| CPU2_FCX | CPUx Free CSA List Head Pointer | $1FE38_H$ | See Family Spec |
| CPU2_LCX | CPUx Free CSA List Limit Pointer | $1FE3C_H$ | See Family Spec |
| CPU2_CUS_ID | CPUx Customer ID register | $1FE50_H$ | See Family Spec |
| CPU2_Dy (y=0-15) | CPUx Data General Purpose Register y | $1FF00_H$+y*4 | See Family Spec |
| CPU2_Ay (y=0-15) | CPUx Address General Purpose Register y | $1FF80_H$+y*4 | See Family Spec |

## 5.3 TC35x Specific Registers

### 5.3.1 SRI slave interface for SFR+CSFR

**CPUx Flash Configuration Register 0**

Software may program a Flash Prefetch Buffer with a master tag identifier stored in Flash Configuration Register 0.

If a CPU instance does not have a local PFlash bank then the FLASHCON0 register associated with that instance will have no functionality.

**CPU0_FLASHCON0**
**CPUx Flash Configuration Register 0** **($01100_H$)** **Reset Value: Table 22**
**CPU1_FLASHCON0**
**CPUx Flash Configuration Register 0** **($01100_H$)** **Reset Value: Table 23**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RES | | TAG4 | | | | | | RES | | TAG3 | | | | | |
| r | | rw | | | | | | r | | rw | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RES | | TAG2 | | | | | | RES | | TAG1 | | | | | |
| r | | rw | | | | | | r | | rw | | | | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **TAG1** | 5:0 | rw | **Flash Prefetch Buffer 1 Configuration**<br>FPB is assigned to on chip bus master with master tag id equal to TAG1. |
| **RES** | 7:6,<br>15:14,<br>23:22,<br>31:30 | r | **Reserved**<br>Always read as 0; should be written with 0. |
| **TAG2** | 13:8 | rw | **Flash Prefetch Buffer 2 Configuration**<br>FPB is assigned to on chip bus master with master tag id equal to TAG2. |
| **TAG3** | 21:16 | rw | **Flash Prefetch Buffer 3 Configuration**<br>FPB is assigned to on chip bus master with master tag id equal to TAG3. |
| **TAG4** | 29:24 | rw | **Flash Prefetch Buffer 4 Configuration**<br>FPB is assigned to on chip bus master with master tag id equal to TAG4. |

**Table 22 Reset Values of CPU0_FLASHCON0**

| Reset Type | Reset Value | Note |
|---|---|---|
| Application Reset | $3F3F\ 3F3F_H$ | |
| CFS Value | $2221\ 2120_H$ | |

**Table 23 Reset Values of CPU1_FLASHCON0**

| Reset Type | Reset Value | Note |
|---|---|---|
| Application Reset | 3F3F 3F3F$_H$ | |
| CFS Value | 2220 2021$_H$ | |

## 5.4 Connectivity

No connections in TC35x

## 5.5 Revision History

**Table 24 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V1.1.16** | | |
| | No change | |
| **V1.1.17** | | |
| | No change | |
| **V1.1.18** | | |
| | No change | |
| **V1.1.19** | | |
| | No change | |
| **V1.1.20** | | |
| Page 2, 9 , 17 | Change index variable from 'x' to intended 'i' for registers SPR_SPROT_RGNACCENAi_R and SPR_SPROT_RGNACCENBi_R to remove confusion with CPU instance variable. | |
| Page 2, 9 , 17 | Change index variable from 'x' to intended 'i' for registers all DLMU_SPROT registers to remove confusion with CPU instance variable. | |
| **V1.1.21** | | |
| | No change | |

# 6 Non Volatile Memory (NVM) Subsystem

## 6.1 Overview

The Non Volatile Memory (NVM) Subsystem comprises of the Data Memory Unit (DMU), Program Flash Interface (PFI), and Non Volatile Memory module (comprising of the Flash Standard Interface (FSI), Program and Data Flash memories and Program Flash Read Write buffer (PFRWB)).

- Data Memory Unit (DMU): Controls command sequences executed on all program and data flash memories.
- Flash Standard Interface (FSI): Executes erase, program and verify operations on all flash memories.
- Program Flash (PFLASH): Divided into one or more banks each connected to a CPU. It is used by the application to store program code and data constants. Compute performance is optimized by using a point-to-point interface to minimize latency and maximize bandwidth. Each PFLASH is connected to a PFlash Read Write Buffer (PFRWB) that performs the ECC correction and detection and provides the read data to the system.
- Program Flash Interface (PFI): Each PFLASH bank has a unique point-to-point fast connection to a CPU provided by a PFI. The PFI interfaces between the CPU and the PFRWB and contains the Prefetch Buffers for storing speculative data.
- Data Flash (DFLASH): The Data Flash Module is used to emulate EEPROM and store data and divided into two banks. DFLASH read accesses are relatively slow compared to PFLASH accesses. The DFlash Read Write Buffer (DFRWB) in the FSI interfaces to the DFLASH to provide the read data. Data Flash Module also contains regions to store configuration data - User Configuration Blocks (UCBs), and Configuration Sector (CFS) which is not accessible by user.
- Boot ROM (BROM): Connected to the system via the DMU SRI port.
  - Tuning protection (commonly called the “Secure Watchdog”) to protect user software and data from maltuning data.

***Attention:*** ***The ‘Non Volatile Memory Subsystem’ chapter is the AURIX PMU chapter re-structured for closer alignment to AURIXTC3XX product architecture. It comprises of the DMU, PFI, NVM and UCB Chapters. Please note that the application accessible registers located in the FSI, and the PFLASH read status and control registers are described in the NVM chapter.***

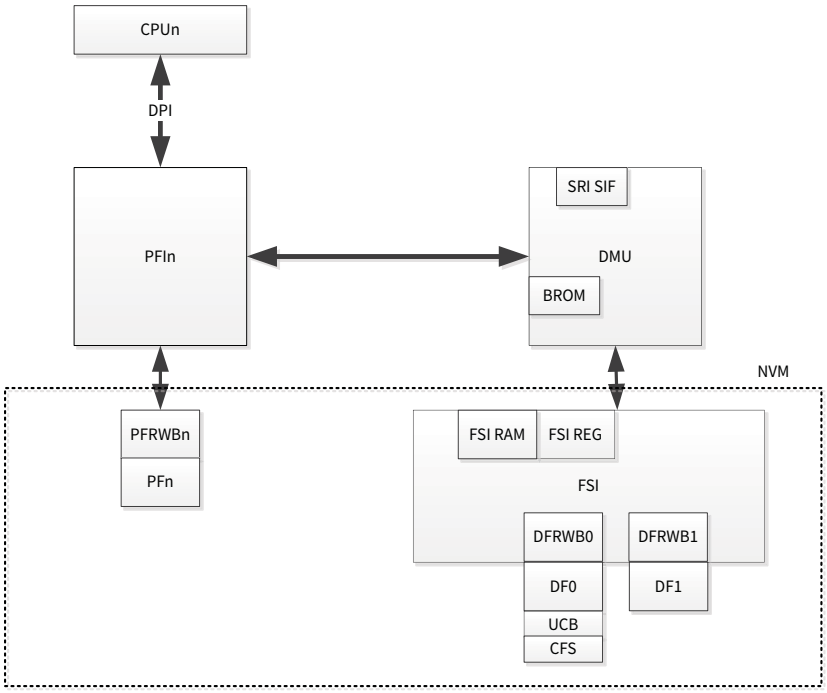



**Figure 4 Non Volatile Memory (NVM) Subsystem**

The purpose of the PFLASH NVM is:

- One or more PFLASH banks stores program code and data constants.
- Implementation of Erase Counters.

The purpose of the DFLASH NVM is:

- Emulation of Electrically Erasable Programmable Read Only Memory (EEPROM):
  - CPU-EEPROM used by the user application.
  - HSM-EEPROM used by the security application.
- Multiple User Configuration Blocks (UCB) used for:
  - Password based read protection combined with write protection.
  - Read-only UCB configured by IFX with unique chip identifier and trimming data.
- Configuration Sector (CFS) stores system set-up data not accessible by the user.

Data stored in the NVM is protected by ECC checksum.

- An ECC decoder at the output of the NVM corrects and detects faults in the NVM array.
- The NVM is fault tolerant and supports system operation in the presence of a number of NVM bit errors.
- For Program Flash the calculation of the ECC checksum is extended across the address to provide read protection against addressing faults.

If the Flash is not operating in the application then the NVM may be programmed and erased by command sequences executed by the FSI micro controller. All read accesses to Flash are memory mapped reads. Margin read levels may be used to check how completely a cell is programmed or erased.

The Non Volatile Memory interface micro architecture includes a security layer and a safety layer.

### Security Layer (provided by DMU and PFI)

- Read protection is enabled/disabled with a Flash Module (Bank) granularity.

- Write protection is enabled/disabled with a Flash Module sector based granularity.

**Safety Layer**

- Master specific read access protection to each Flash Module (Bank).
- Master specific read and write access control to individual Special Function Registers (SFRs).
- Integrity of data stored in the NVM is ensured by an ECC checksum
- Integrity of PFlash read path is ensured by monitoring of read parameters in the FSI (MISR, redundant Flip Flops etc.), PFI partial lockstep mechanism, protection of PFlash wait cycles with ECC checksum, protection of data from PFI to CPU by ECC checksum and an additional safety mechanism to ensure that the local PFlash is not being programmed/erased when not expected by PFI.

## 6.2 Revision History

**Table 25 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V2.0.3** | | |
| | Created to form a concise introduction chapter for the appendices | |
| **V2.0.4** | | |
| | No Changes. | |
| **V2.0.5** | | |
| | No Changes. | |
| **V2.0.6** | | |
| | No Changes. | |
| **V2.0.7** | | |
| | No Changes. | |

## 6.3 Data Memory Unit (DMU)

This chapter supplements the family documentation with the device specific information for TC35x.

### 6.3.1 TC35x Specific Register Set

#### Register Address Space Table

**Table 26 Register Address Space - PMU**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| PMU | $F8038000_H$ | $F803FFFF_H$ | sri slave interface |

**Table 27 Register Address Space - DMU**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| (DMU) | $8FFF0000_H$ | $8FFFFFFF_H$ | Boot ROM (BROM) |
| | $AF000000_H$ | $AF01FFFF_H$ | Data Flash 0 EEPROM (DF0) and Host Command Sequence Interpreter |
| | $AFC00000_H$ | $AFC1FFFF_H$ | Data Flash 1 EEPROM (DF1) and HSM Command Sequence Interpreter |
| | $AFFF0000_H$ | $AFFFFFFF_H$ | Boot ROM (BROM) |
| DMU | $F8040000_H$ | $F807FFFF_H$ | SRI slave interface - Register Address Space |
| (DMU) | $FFC00000_H$ | $FFC1FFFF_H$ | Data Flash 1 EEPROM (DF1) and HSM Command Sequence Interpreter |

#### Register Overview Table

**Table 28 Register Overview - PMU (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| PMU_ID | Module Identification Register | $0508_H$ | U,SV | BE | Application Reset | See Family Spec |

**Table 29 Register Overview - DMU (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| DMU_HF_ID | Module Identification Register | 0000008 H | U,SV | BE | Application Reset | See Family Spec |
| DMU_HF_STATUS | Flash Status Register | 0000010 H | U,SV | BE | Application Reset | 11 |

**Table 29 Register Overview - DMU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| DMU_HF_CONTROL | Flash Control Register | 0000014H | U,SV | P,SV,E | Application Reset | See Family Spec |
| DMU_HF_OPERATION | Flash Operation Register | 0000018H | U,SV | BE | System Reset | See Family Spec |
| DMU_HF_PROTECT | Flash Protection Status Register | 000001CH | U,SV | BE | Application Reset | 13 |
| DMU_HF_CONFIRM0 | Flash Confirm Status Register 0 | 0000020H | U,SV | BE | Application Reset | See Family Spec |
| DMU_HF_CONFIRM1 | Flash Confirm Status Register 1 | 0000024H | U,SV | BE | Application Reset | See Family Spec |
| DMU_HF_CONFIRM2 | Flash Confirm Status Register 2 | 0000028H | U,SV | BE | Application Reset | See Family Spec |
| DMU_HF_EER | Enable Error Interrupt Control Register | 0000030H | U,SV | P,SV | Application Reset | See Family Spec |
| DMU_HF_ERRSR | Error Status Register | 0000034H | U,SV | BE | Application Reset | See Family Spec |
| DMU_HF_CLRE | Clear Error Register | 0000038H | U,SV | P,SV | Application Reset | See Family Spec |
| DMU_HF_ECCR | DF0 ECC Read Register | 0000040H | U,SV | BE | Application Reset | See Family Spec |
| DMU_HF_ECCS | DF0 ECC Status Register | 0000044H | U,SV | BE | Application Reset | See Family Spec |
| DMU_HF_ECCC | DF0 ECC Control Register | 0000048H | U,SV | P,SV,E | Application Reset | See Family Spec |
| DMU_HF_ECCW | DF0 ECC Write Register | 000004CH | U,SV | P,SV,E | Application Reset | See Family Spec |
| DMU_HF_CCONTROL | Cranking Control Register | 0000050H | U,SV | P,SV | System Reset | See Family Spec |

**Table 29 Register Overview - DMU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| DMU_HF_PSTATUS | Power Status Register | 0000060H | U,SV | BE | Application Reset | See Family Spec |
| DMU_HF_PCONTROL | Power Control Register | 0000064H | U,SV | P,SV | Application Reset | See Family Spec |
| DMU_HF_PWAIT | PFLASH Wait Cycle Register | 0000068H | U,SV | P,SV,E | System Reset | See Family Spec |
| DMU_HF_DWAIT | DFLASH Wait Cycle Register | 000006CH | U,SV | P,SV,E | System Reset | See Family Spec |
| DMU_HF_PROCONUSR | DF0 User Mode Control | 0000074H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HF_PROCONPF | PFLASH Protection Configuration | 0000080H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HF_PROCONTP | Tuning Protection Configuration | 0000084H | U,SV | BE | See page 15 | 15 |
| DMU_HF_PROCONDF | DFLASH Protection Configuration | 0000088H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HF_PROCONRAM | RAM Configuration | 000008CH | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HF_PROCONDBG | Debug Interface Protection Configuration | 0000090H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HF_SUSPEND | Suspend Control Register | 00000F0H | U,SV | P,U,SV | Application Reset | See Family Spec |
| DMU_HF_MARGIN | Margin Control Register | 00000F4H | U,SV | P,U,SV | Application Reset | See Family Spec |
| DMU_HF_ACCEN1 | Access Enable Register 1 | 00000F8H | U,SV | SV,SE | Application Reset | See Family Spec |
| DMU_HF_ACCEN0 | Access Enable Register 0 | 00000FCH | U,SV | SV,SE | Application Reset | See Family Spec |

**Table 29 Register Overview - DMU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| DMU_HP_PROCONPi0 (i=0-1) | PFLASH Bank i Protection Configuration 0 | 0010000 H+i*100$_H$ | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_PROCONPi1 (i=0-1) | PFLASH Bank i Protection Configuration 1 | 0010004 H+i*100$_H$ | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_PROCONPi2 (i=0-1) | PFLASH Bank i Protection Configuration 2 | 0010008 H+i*100$_H$ | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_PROCONPi3 (i=0-1) | PFLASH Bank i Protection Configuration 3 | 001000C H+i*100$_H$ | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_PROCONPi4 (i=0-1) | PFLASH Bank i Protection Configuration 4 | 0010010 H+i*100$_H$ | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_PROCONPi5 (i=0-1) | PFLASH Bank i Protection Configuration 5 | 0010014 H+i*100$_H$ | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_PROCONOTPi0 (i=0-1) | PFLASH Bank i OTP Protection Configuration 0 | 0010040 H+i*100$_H$ | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_PROCONOTPi1 (i=0-1) | PFLASH Bank i OTP Protection Configuration 1 | 0010044 H+i*100$_H$ | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_PROCONOTPi2 (i=0-1) | PFLASH Bank i OTP Protection Configuration 2 | 0010048 H+i*100$_H$ | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_PROCONOTPi3 (i=0-1) | PFLASH Bank i OTP Protection Configuration 3 | 001004C H+i*100$_H$ | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_PROCONOTPi4 (i=0-1) | PFLASH Bank i OTP Protection Configuration 4 | 0010050 H+i*100$_H$ | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_PROCONOTPi5 (i=0-1) | PFLASH Bank i OTP Protection Configuration 5 | 0010054 H+i*100$_H$ | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_PROCONWOPi0 (i=0-1) | PFLASH Bank i WOP Configuration 0 | 0010080 H+i*100$_H$ | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_PROCONWOPi1 (i=0-1) | PFLASH Bank i WOP Configuration 1 | 0010084 H+i*100$_H$ | U,SV | BE | See Family Spec | See Family Spec |

**Table 29 Register Overview - DMU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| DMU_HP_PROCONWOPi2 (i=0-1) | PFLASH Bank i WOP Configuration 2 | 0010088H+i*100H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_PROCONWOPi3 (i=0-1) | PFLASH Bank i WOP Configuration 3 | 001008CH+i*100H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_PROCONWOPi4 (i=0-1) | PFLASH Bank i WOP Configuration 4 | 0010090H+i*100H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_PROCONWOPi5 (i=0-1) | PFLASH Bank i WOP Configuration 5 | 0010094H+i*100H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_ECPRIOi0 (i=0-1) | PFLASH Bank i Erase Counter Priority configuration 0 | 00100A0H+i*100H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_ECPRIOi1 (i=0-1) | PFLASH Bank i Erase Counter Priority Configuration 1 | 00100A4H+i*100H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_ECPRIOi2 (i=0-1) | PFLASH Bank i Erase Counter Priority Configuration 2 | 00100A8H+i*100H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_ECPRIOi3 (i=0-1) | PFLASH Bank i Erase Counter Priority Configuration 3 | 00100ACH+i*100H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_ECPRIOi4 (i=0-1) | PFLASH Bank i Erase Counter Priority Configuration 4 | 00100B0H+i*100H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_HP_ECPRIOi5 (i=0-1) | PFLASH Bank i Erase Counter Priority Configuration 5 | 00100B4H+i*100H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_SF_STATUS | HSM Flash Status Register | 0020010H | H | BE | Application Reset | See Family Spec |
| DMU_SF_CONTROL | HSM Flash Configuration Register | 0020014H | H | H | Application Reset | See Family Spec |
| DMU_SF_OPERATION | HSM Flash Operation Register | 0020018H | H | BE | System Reset | See Family Spec |
| DMU_SF_EER | HSM Enable Error Interrupt Control Register | 0020030H | H | H | Application Reset | See Family Spec |

**Table 29 Register Overview - DMU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| DMU_SF_ERRSR | HSM Error Status Register | 0020034 H | H | BE | Application Reset | See Family Spec |
| DMU_SF_CLRE | HSM Clear Error Register | 0020038 H | H | H | Application Reset | See Family Spec |
| DMU_SF_ECCR | HSM DF1 ECC Read Register | 0020040 H | H | BE | Application Reset | See Family Spec |
| DMU_SF_ECCS | HSM DF1 ECC Status Register | 0020044 H | H | BE | Application Reset | See Family Spec |
| DMU_SF_ECCC | HSM DF1 ECC Control Register | 0020048 H | H | H | Application Reset | See Family Spec |
| DMU_SF_ECCW | HSM DF1 ECC Write Register | 002004C H | H | H | Application Reset | See Family Spec |
| DMU_SF_PROCONUSR | HSM DF1 User Mode Control | 0020074 H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_SF_SUSPEND | HSM Suspend Control Register | 00200E8 H | H | H | Application Reset | See Family Spec |
| DMU_SF_MARGIN | HSM DF1 Margin Control Register | 00200EC H | H | H | Application Reset | See Family Spec |
| DMU_SP_PROCONHSMCFG | HSM Protection Configuration | 0030000 H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_SP_PROCONHSMCBS | HSM Code Boot Sector | 0030004 H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_SP_PROCONHSMCX0 | HSM Code Exclusive Protection Configuration | 0030008 H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_SP_PROCONHSMCX1 | HSM Code Exclusive Protection Configuration | 003000C H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_SP_PROCONHSMCOTP0 | HSM Code OTP Protection Configuration | 0030010 H | U,SV | BE | See Family Spec | See Family Spec |

**Table 29 Register Overview - DMU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| DMU_SP_PROCONHSMCOTP1 | HSM Code OTP Protection Configuration | 0030014H | U,SV | BE | See Family Spec | See Family Spec |
| DMU_SP_PROCONHSM | HSM Interface Protection Configuration | 0030040H | U,SV | BE | See Family Spec | See Family Spec |

## 6.3.2 TC35x Specific Registers

### 6.3.2.1 SRI slave interface - Register Address Space

**Flash Status Register**

The Flash Status Register reflects the status of the Flash Banks after reset.

*Note: The DxBUSY and PxBUSY flags cannot be cleared with the "Clear Status" command or with the "Reset to Read" command. These flags are controlled by HW.*

*Note: After every reset, the busy bits are set while the Flash module is busy with startup (until the operation mode is entered). Also the protection installation bits are always set until end of startup.*

**DMU_HF_STATUS**
**Flash Status Register** **(0000010$_H$)** **Application Reset Value: 0000 00FF$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RES | | | | | | RES | | RES | RES | PFPAGE | DFPAGE | RES | RES | RES | RES |
| r | | | | | | rX | | r | rX | rh | rh | rX | rX | rX | rX |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RES | | | | | | | | RES | RES | RES | RES | P1BUSY | P0BUSY | D1BUSY | D0BUSY |
| r | | | | | | | | r | r | r | r | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **D0BUSY** | 0 | rh | **Data Flash Bank 0 Busy**<br>HW-controlled status flag.<br>Indication of busy state of DFLASH bank 0 because of active execution of an operation; DF0 busy state is also indicated during Flash startup after reset or in sleep mode; while in busy state the DF0 does not allow read access.<br>$0_B$ DF0 ready, not busy; DF0 in operation mode.<br>$1_B$ DF0 busy; DF0 not in operation mode. |

| Field | Bits | Type | Description |
|---|---|---|---|
| **D1BUSY** | 1 | rh | **Data Flash Bank 1 Busy**<br>HW-controlled status flag.<br>Indication of busy state of DFLASH bank 1 because of active execution of an operation; DF1 busy state is also indicated during Flash startup after reset or in sleep mode; while in busy state the DF1 does not allow read access.<br>Bit is not set for program/erase operations initiated by the HSM interface.<br>$0_B$ DF1 ready, not busy; DF1 in operation mode.<br>$1_B$ DF1 busy; DF1 not in operation mode. |
| **PxBUSY (x=0-1)** | x+2 | rh | **Program Flash PFxBUSY**<br>HW-controlled status flag.<br>Indication of busy state of PFx because of active execution of an operation; PFx busy state is also indicated during Flash startup after reset or in sleep mode; while in busy state the PFx does not allow read access.<br>$0_B$ PFx ready, not busy; PFx in operation mode.<br>$1_B$ PFx busy; PFx not in operation mode. |
| **RES (x=2-5)** | x+2 | r | **Reserved**<br>Always read as 0; should be written with 0. |
| **RES** | 15:8,<br>23,<br>31:26 | r | **Reserved**<br>Always read as 0; should be written with 0. |
| **RES** | 16,<br>17,<br>18,<br>19,<br>22,<br>25:24 | rX | **Reserved**<br>Undefined. |
| **DFPAGE** | 20 | rh | **Data Flash in Page Mode**<br>HW-controlled status flag.<br>Set with Enter Page Mode for DFLASH, cleared with Write Page command.<br>This bit is not set by “Enter Page Mode” initiated by the HSM interface.<br><br>*Note: Read accesses are allowed while in page mode.*<br><br>$0_B$ Data Flash not in page mode<br>$1_B$ Data Flash in page mode |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PFPAGE** | 21 | rh | **Program Flash in Page Mode**<br>HW-controlled status flag.<br>Set with Enter Page Mode for Flash, cleared with Write Page command<br>This bit is not set by “Enter Page Mode” initiated by the HSM interface.<br><br>*Note: Read accesses are allowed while in page mode.*<br><br>$0_B$ Flash not in page mode.<br>$1_B$ Flash in page mode. |

### Flash Protection Status Register

This register reports the state of the Flash protection and contains protection relevant control fields.

**DMU_HF_PROTECT**
**Flash Protection Status Register ($000001C_H$) Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RES | | | | | | | SRT | RES | | | | | | | |
| r | | | | | | | rh | r | | | | | | | |

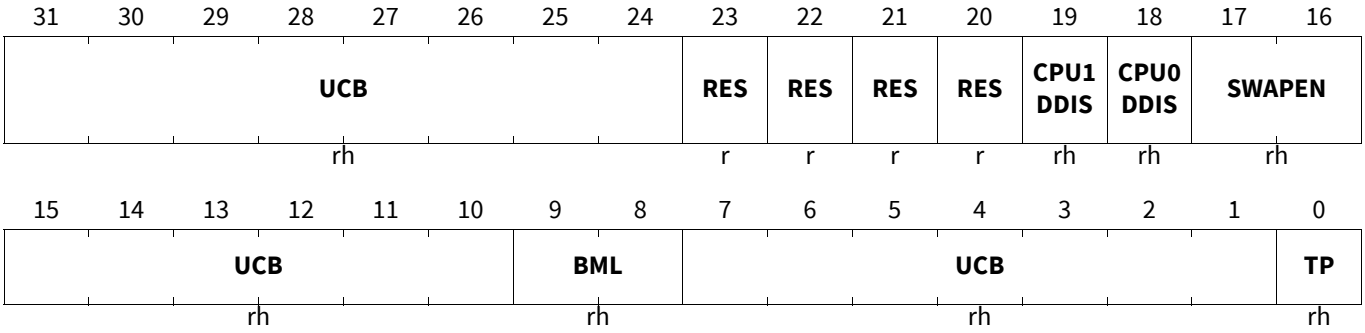
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RES | | RES | RES | RES | RES | RES | PRODISP1 | PRODISP0 | RES | | PRODISSWAP | PRODISBMHD | PRODISEC | PRODISDBG | PRODISD | PRODISP |
| r | | r | r | r | r | r | rh | rh | r | | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PRODISP** | 0 | rh | **PFLASH Protection Disabled**<br>The protection configured by UCB_PFLASH_ORIG and UCB_PFLASH_COPY was successfully disabled by supplying the correct password to “Disable Protection”.<br><br>*Note: Cleared with command “Resume Protection”.* |
| **PRODISD** | 1 | rh | **DFLASH Protection Disabled**<br>The protection configured by UCB_DFLASH_ORIG and UCB_DFLASH_COPY was successfully disabled by supplying the correct password to “Disable Protection”.<br><br>*Note: Cleared with command “Resume Protection”.* |
| **PRODISDBG** | 2 | rh | **Debug Interface Password Protection Disabled**<br>The password configured by UCB_DBG_ORIG and UCB_DBG_COPY was correctly received with “Disable Protection”.<br>When DMU_SP_PROCONHSMCFG.DESTDBG is “destructive” then only the SSW can disable this protection.<br><br>*Note: Cleared with command “Resume Protection”.* |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PRODISEC** | 3 | rh | **Erase Counter Priority Protection Disabled**<br>The protection configured by UCB_ECPRIO_ORIG and UCB_ECPRIO_COPY was successfully disabled by supplying the correct password to “Disable Protection”.<br><br>*Note: Cleared with command "Resume Protection".* |
| **PRODISBMHD** | 4 | rh | **BMHD Protection Disabled**<br>The protection configured by UCB_BMHD0_ORIG and UCB_BMHD0_COPY was successfully disabled by supplying the correct password to “Disable Protection”.<br><br>*Note: Cleared with command "Resume Protection".* |
| **PRODISSWAP** | 5 | rh | **UCB_SWAP protection Disabled**<br>The protection configured by UCB_SWAP_ORIG and UCB_SWAP_COPY was successfully disabled by supplying the correct password to “Disable Protection”.<br><br>*Note: Cleared with command "Resume Protection".* |
| **RES** | 7:6,<br>23:14,<br>31:25 | r | **Reserved**<br>Always read as 0; should be written with 0. |
| **PRODISPx (x=0-1)** | x+8 | rh | **Program Flash Protection Disable PRODISPx**<br>The protection configured for PFx by UCB_PFLASH_ORIG and UCB_PFLASH_COPY was successfully disabled by supplying the correct password to “Disable Protection”.<br><br>*Note: Cleared with command "Resume Protection".* |
| **RES (x=2-5)** | x+8 | r | **Reserved**<br>Always read as 0; should be written with 0. |
| **SRT** | 24 | rh | **Secure Retest Password Protection Disabled**<br><br>*Note: Cleared with command “Resume Protection”.*<br><br>$0_B$ Secure Retest protection is not disabled.<br>$1_B$ Secure Retest protection is disabled. |

### Tuning Protection Configuration

**DMU_HF_PROCONTP**
**Tuning Protection Configuration** **(0000084$_H$)** **Reset Value: Table 30**




| Field | Bits | Type | Description |
|---|---|---|---|
| **TP** | 0 | rh | **Tuning Protection**<br>This bit indicates whether tuning protection is installed or not.<br>$0_B$ Tuning protection is not configured.<br>$1_B$ Tuning protection is configured and installed, if correctly confirmed. |
| **UCB** | 7:1,<br>15:10,<br>31:24 | rh | **Reserved for UCB**<br>Deliver the corresponding content of UCB. |
| **BML** | 9:8 | rh | **Boot Mode Lock**<br>Used by the SSW to restrict the boot mode selection.<br>$00_B$ Boot flow with standard evaluation of boot headers.<br>$01_B$ Restricted boot flow, never evaluating HWCFG pins and without fallback to boot loader.<br>…<br>$11_B$ Restricted boot flow, never evaluating HWCFG pins and without fallback to boot loader. |
| **SWAPEN** | 17:16 | rh | **Enable SOTA mode**<br>This field enables the entry into "Software update Over the Air(SOTA) mode". In this mode, an alternate PFLASH address map can be selected. Please refer to the SOTA section of the Introduction chapter for more details.<br>$00_B$ **Disabled**, SOTA mode disabled.<br>…<br>$10_B$ **Disabled**, SOTA mode disabled.<br>$11_B$ **Enabled**, SOTA mode enabled. |
| **CPUxDDIS (x=0-1)** | x+18 | rh | **Disable direct LPB access**<br>Disable direct LPB access by the CPU to the Local PFlash Bank (LPB).<br>$0_B$ Direct LPB access is enabled.<br>$1_B$ Direct LPB access is disabled. |
| **RES (x=2-5)** | x+18 | r | **Reserved**<br>Always read as 0; should be written with 0. |

**Table 30 Reset Values of DMU_HF_PROCONTP**

| Reset Type | Reset Value | Note |
|---|---|---|
| Application Reset | $0000\ 0000_H$ | |
| CFS Value | $0000\ 0000_H$ | |

### 6.3.3 Connectivity

**Table 31 Connections of DMU**

| Interface Signals | connects | | Description |
|---|---|---|---|
| DMU:HOST_INT | to | INT:dmu.HOST_INT | PMU Host Service Request |
| DMU:FSI_INT | to | INT:dmu.FSI_INT | PMU FSI Service Request |

### 6.3.4 Revision History

**Table 32 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V2.0.9** | | |
| | No document changes - version update to remain aligned with family document. | |
| **V2.0.10** | | |
| | No document changes - version update to remain aligned with family document. | |
| **V2.0.11** | | |
| **Page 11** | Updated register **DMU_HF_STATUS**. | |
| **Page 16** | **Connectivity** - Table updated. | |
| | No functional changes. | |
| **V2.0.12** | | |
| – | No functional changes. | |

## 6.4 Non Volatile Memory (NVM)

This chapter supplements the family documentation with the device specific information for TC35x.

### 6.4.1 TC35x Specific Register Set

#### Register Address Space Table

**Table 33 Register Address Space - FSI**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| FSI | $F8030000_H$ | $F80300FF_H$ | sri slave interface |

**Table 34 Register Address Space - PFI**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| (PFI0) | $80000000_H$ | $801FFFFF_H$ | Program Flash cached address space |
| | $A0000000_H$ | $A01FFFFF_H$ | Program Flash non-cached address space |
| | $A8000000_H$ | $A8003FFF_H$ | Erase Counter address space |
| PFI0 | $A8080000_H$ | $A80FFFFF_H$ | Register address space |
| (PFI1) | $80300000_H$ | $804FFFFF_H$ | Program Flash cached address space |
| | $A0300000_H$ | $A04FFFFF_H$ | Program Flash non-cached address space |
| | $A8300000_H$ | $A8303FFF_H$ | Erase Counter address space |
| PFI1 | $A8380000_H$ | $A83FFFFF_H$ | Register address space |

#### Register Overview Table

**Table 35 Register Overview - FSI (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| FSI_COMM_1 | Communication Register 1 | $0004_H$ | U,SV | U,SV | System Reset | See Family Spec |
| FSI_COMM_2 | Communication Register 2 | $0005_H$ | U,SV | U,SV | System Reset | See Family Spec |
| FSI_HSMCOMM_1 | HSM Communication Register 1 | $0006_H$ | H | H | System Reset | See Family Spec |
| FSI_HSMCOMM_2 | HSM Communication Register 2 | $0007_H$ | H | H | System Reset | See Family Spec |

**Table 36 Register Overview - PFI (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| PFI0_ECCR | ECC Read Register | $000000_H$ | See Family Spec |
| PFI1_ECCR | ECC Read Register | $000000_H$ | See Family Spec |
| PFI0_ECCS | ECC Status Register | $000020_H$ | See Family Spec |
| PFI1_ECCS | ECC Status Register | $000020_H$ | See Family Spec |
| PFI0_SBABRECORDx (x=0-16) | SBAB Record x | $002000_H$+x*20 H | See Family Spec |
| PFI1_SBABRECORDx (x=0-16) | SBAB Record x | $002000_H$+x*20 H | See Family Spec |
| PFI0_DBABRECORDx (x=0-1) | DBAB Record x | $004000_H$+x*20 H | See Family Spec |
| PFI1_DBABRECORDx (x=0-1) | DBAB Record x | $004000_H$+x*20 H | See Family Spec |
| PFI0_MBABRECORDx (x=0) | MBAB Record 0 | $008000_H$ | See Family Spec |
| PFI1_MBABRECORDx (x=0) | MBAB Record 0 | $008000_H$ | See Family Spec |
| PFI0_ZBABRECORDx (x=0-3) | ZBAB Record x | $00C000_H$+x*20 H | See Family Spec |
| PFI1_ZBABRECORDx (x=0-3) | ZBAB Record x | $00C000_H$+x*20 H | See Family Spec |

### 6.4.2 Connectivity

No connections in device.

### 6.4.3 Revision History

**Table 37 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V2.0.4** | | |
| | No document changes - version update to remain aligned with family document. | |
| **V2.0.5** | | |
| **Page 17** | **Register Address Space Table** - PFI instances not used in this device removed. | |
| **V2.0.6** | | |
| | No functional changes. | |

# 7 Local Memory Unit (LMU)

This is a description of the TC35x specific features of the LMU of the AURIXTC3XX product family.

## 7.1 TC35x Specific IP Configuration

The LMU instance in the TC35x provides 128 KiB of SRAM.

## 7.2 TC35x Specific Register Set

**Table 38 Register Address Space - LMU**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| (LMU0) | $90040000_H$ | $9007FFFF_H$ | sri slave interface (RAM Address Range cached) |
| | $B0040000_H$ | $B007FFFF_H$ | sri slave interface (RAM Address Range non-cached) |
| LMU0 | $F8100000_H$ | $F810FFFF_H$ | sri slave interface |
| (LMU1) | $90080000_H$ | $900BFFFF_H$ | sri slave interface (RAM Address Range cached) |
| | $B0080000_H$ | $B00BFFFF_H$ | sri slave interface (RAM Address Range non-cached) |
| LMU1 | $F8110000_H$ | $F811FFFF_H$ | sri slave interface |

**Table 39 Register Overview - LMU (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| LMU0_CLC | LMU Clock Control Register | $00000_H$ | See Family Spec |
| LMU1_CLC | LMU Clock Control Register | $00000_H$ | See Family Spec |
| LMU0_MODID | LMU Module ID Register | $00008_H$ | See Family Spec |
| LMU1_MODID | LMU Module ID Register | $00008_H$ | See Family Spec |
| LMU0_ACCEN0 | LMU Access Enable Register 0 | $00010_H$ | See Family Spec |
| LMU1_ACCEN0 | LMU Access Enable Register 0 | $00010_H$ | See Family Spec |
| LMU0_ACCEN1 | LMU Access Enable Register 1 | $00014_H$ | See Family Spec |

**Table 39 Register Overview - LMU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| LMU1_ACCEN1 | LMU Access Enable Register 1 | $00014_H$ | See Family Spec |
| LMU0_MEMCON | LMU Memory Control Register | $00020_H$ | See Family Spec |
| LMU1_MEMCON | LMU Memory Control Register | $00020_H$ | See Family Spec |
| LMU0_SCTRL | LMU Safety Control Register | $00024_H$ | See Family Spec |
| LMU1_SCTRL | LMU Safety Control Register | $00024_H$ | See Family Spec |
| LMU0_RGNLAx (x=0-15) | LMU Region Lower Address Register | $00050_H+x*10_H$ | See Family Spec |
| LMU1_RGNLAx (x=0-15) | LMU Region Lower Address Register | $00050_H+x*10_H$ | See Family Spec |
| LMU0_RGNUAx (x=0-15) | LMU Region Upper Address Register | $00054_H+x*10_H$ | See Family Spec |
| LMU1_RGNUAx (x=0-15) | LMU Region Upper Address Register | $00054_H+x*10_H$ | See Family Spec |
| LMU0_RGNACCENWAx (x=0-15) | LMU Region Write Access Enable Register A | $00058_H+x*10_H$ | See Family Spec |
| LMU1_RGNACCENWAx (x=0-15) | LMU Region Write Access Enable Register A | $00058_H+x*10_H$ | See Family Spec |
| LMU0_RGNACCENWBx (x=0-15) | LMU Region Write Access Enable Register B | $0005C_H+x*10_H$ | See Family Spec |
| LMU1_RGNACCENWBx (x=0-15) | LMU Region Write Access Enable Register B | $0005C_H+x*10_H$ | See Family Spec |
| LMU0_RGNACCENRAx (x=0-15) | LMU Region Read Access Enable Register A | $00158_H+x*10_H$ | See Family Spec |

**Table 39 Register Overview - LMU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| LMU1_RGNACCENRAx (x=0-15) | LMU Region Read Access Enable Register A | $00158_H$+x*$10_H$ | See Family Spec |
| LMU0_RGNACCENRBx (x=0-15) | LMU Region Read Access Enable Register B | $0015C_H$+x*$10_H$ | See Family Spec |
| LMU1_RGNACCENRBx (x=0-15) | LMU Region Read Access Enable Register B | $0015C_H$+x*$10_H$ | See Family Spec |

**Table 40 Register Overview - LMU0 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| LMU0_CLC | LMU Clock Control Register | $00000_H$ | See Family Spec |
| LMU0_MODID | LMU Module ID Register | $00008_H$ | See Family Spec |
| LMU0_ACCEN0 | LMU Access Enable Register 0 | $00010_H$ | See Family Spec |
| LMU0_ACCEN1 | LMU Access Enable Register 1 | $00014_H$ | See Family Spec |
| LMU0_MEMCON | LMU Memory Control Register | $00020_H$ | See Family Spec |
| LMU0_SCTRL | LMU Safety Control Register | $00024_H$ | See Family Spec |
| LMU0_RGNLAx (x=0-15) | LMU Region Lower Address Register | $00050_H$+x*$10_H$ | See Family Spec |
| LMU0_RGNUAx (x=0-15) | LMU Region Upper Address Register | $00054_H$+x*$10_H$ | See Family Spec |
| LMU0_RGNACCENWAx (x=0-15) | LMU Region Write Access Enable Register A | $00058_H$+x*$10_H$ | See Family Spec |

**Table 40 Register Overview - LMU0 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| LMU0_RGNACCENWBx (x=0-15) | LMU Region Write Access Enable Register B | $0005C_H$+x*$10_H$ | See Family Spec |
| LMU0_RGNACCENRAx (x=0-15) | LMU Region Read Access Enable Register A | $00158_H$+x*$10_H$ | See Family Spec |
| LMU0_RGNACCENRBx (x=0-15) | LMU Region Read Access Enable Register B | $0015C_H$+x*$10_H$ | See Family Spec |

**Table 41 Register Overview - LMU1 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| LMU1_CLC | LMU Clock Control Register | $00000_H$ | See Family Spec |
| LMU1_MODID | LMU Module ID Register | $00008_H$ | See Family Spec |
| LMU1_ACCEN0 | LMU Access Enable Register 0 | $00010_H$ | See Family Spec |
| LMU1_ACCEN1 | LMU Access Enable Register 1 | $00014_H$ | See Family Spec |
| LMU1_MEMCON | LMU Memory Control Register | $00020_H$ | See Family Spec |
| LMU1_SCTRL | LMU Safety Control Register | $00024_H$ | See Family Spec |
| LMU1_RGNLAx (x=0-15) | LMU Region Lower Address Register | $00050_H$+x*$10_H$ | See Family Spec |
| LMU1_RGNUAx (x=0-15) | LMU Region Upper Address Register | $00054_H$+x*$10_H$ | See Family Spec |
| LMU1_RGNACCENWAx (x=0-15) | LMU Region Write Access Enable Register A | $00058_H$+x*$10_H$ | See Family Spec |

**Table 41 Register Overview - LMU1 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| LMU1_RGNACCENWBx (x=0-15) | LMU Region Write Access Enable Register B | $0005C_H+x*10_H$ | See Family Spec |
| LMU1_RGNACCENRAx (x=0-15) | LMU Region Read Access Enable Register A | $00158_H+x*10_H$ | See Family Spec |
| LMU1_RGNACCENRBx (x=0-15) | LMU Region Read Access Enable Register B | $0015C_H+x*10_H$ | See Family Spec |

## 7.3 TC35x Specific Registers

There are no TC35x specific registers in the LMU

## 7.4 Connectivity

**Table 42 List of LMU0 Interface Signals**

| Interface Signals | I/O | Description |
|---|---|---|
| sx_sri | | **sri slave interface** |
| | | **sri slave interface (RAM Address Range non-cached)** |
| | | **sri slave interface (RAM Address Range cached)** |
| SX_ALARM_LMU | | **LMU Alarm Outputs to SMU** |

**Table 43 List of LMU1 Interface Signals**

| Interface Signals | I/O | Description |
|---|---|---|
| sx_sri | | **sri slave interface** |
| | | **sri slave interface (RAM Address Range non-cached)** |
| | | **sri slave interface (RAM Address Range cached)** |
| SX_ALARM_LMU | | **LMU Alarm Outputs to SMU** |

No connections in TC35x

## 7.5 Revision History

**Table 44 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V3.1.15** | | |
| | Revision history update, no functional changes. | |

**Table 44 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V3.1.16** | | |
| – | No functional change. | |

# 8 Default Application Memory (LMU_DAM)

This device doesn’t contain LMU_DAM.

# 9 System Control Unit (SCU)

This chapter describes the System Control Unit (short SCU) Module of the TC35x.

## 9.1 TC35x Specific IP Configuration

**Table 45 TC35x specific configuration of SCU**

| Parameter | SCU |
|---|---|
| **Number of WDT linked to the number of CPU** | 3 |
| **Name of the ssw value** | After SSW execution |
| **CFS value for DTSCBGOCTRL register** | $40_H$ |
| **CFS value for DTSCCON register** | $200_H$ |

The following sections describe several differences that are device specific at the SCU level.

### 9.1.1 LBIST considerations for TC35x

The LBIST function can be controlled via four registers available at SCU level: LBISTCTRL0, LBISTCTRL1, LBISTCTRL2 and LBISTCTRL3 (for a complete description of these register, please address the family specification).

The LBISTCTRL3 register contains the MISR signature value that can be read back via software, after the LBIST execution (and the execution is valid).

#### 9.1.1.1 TC35x AA

**LBIST Configuration A**

LBISTCTRL0.PATTERNS = 0x100;

LBISTCTRL2.LENGTH = 0x40;

With LBISTCTRL1.BODY = 0:

- LBISTCTRL1 = 0x54000007
- LBISTCTRL3 = 0x97A61165

With LBISTCTRL1.BODY = 1:

- LBISTCTRL1 = 0x5C000007
- LBISTCTRL3 = 0x31327160

#### 9.1.1.2 TC35x AB

**LBIST Configuration A**

LBISTCTRL0.PATTERNS = 0x100;

LBISTCTRL2.LENGTH = 0x40;

With LBISTCTRL1.BODY = 0:

- LBISTCTRL1 = 0x54000007
- LBISTCTRL3 = 0x3B7272AC

With LBISTCTRL1.BODY = 1:

- LBISTCTRL1 = 0x5C000007

- LBISTCTRL3 = 0x9DE612A9

## 9.2 TC35x Specific Register Set

The address space for the module registers is defined in **Register Address Space - SCU**.

**Table 46 Register Address Space - SCU**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| SCU | $F0036000_H$ | $F00363FF_H$ | SCU: Connections to FPI/BPI bus |

**Table 47 Register Overview - SCU (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| | Reserved ($0010_H$ Byte) | $0000_H$ | BE | BE | | |
| SCU_ID | Identification Register | $0008_H$ | U,SV | BE | System Reset | See Family Spec |
| | Reserved ($0010_H$ Byte) | $000C_H$ | BE | BE | | |
| SCU_OSCCON | OSC Control Register | $0010_H$ | U,SV | SV,SE,P0 | See Family Spec | See Family Spec |
| SCU_SYSPLLSTAT | System PLL Status Register | $0014_H$ | U,SV | BE | See Family Spec | See Family Spec |
| SCU_SYSPLLCON0 | System PLL Configuration 0 Register | $0018_H$ | U,SV | SV,SE,P0 | System Reset | See Family Spec |
| SCU_SYSPLLCON1 | System PLL Configuration 1 Register | $001C_H$ | U,SV | SV,SE,P0 | System Reset | See Family Spec |
| SCU_SYSPLLCON2 | System PLL Configuration 2 Register | $0020_H$ | U,SV | SV,SE,P0 | System Reset | See Family Spec |
| SCU_PERPLLSTAT | Peripheral PLL Status Register | $0024_H$ | U,SV | BE | System Reset | See Family Spec |
| SCU_PERPLLCON0 | Peripheral PLL Configuration 0 Register | $0028_H$ | U,SV | SV,SE,P0 | System Reset | See Family Spec |
| SCU_PERPLLCON1 | Peripheral PLL Configuration 1 Register | $002C_H$ | U,SV | SV,SE,P0 | System Reset | See Family Spec |
| SCU_CCUCON0 | CCU Clock Control Register 0 | $0030_H$ | U,SV | SV,SE,P0 | See Family Spec | See Family Spec |

**Table 47 Register Overview - SCU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SCU_CCUCON1 | CCU Clock Control Register 1 | $0034_H$ | U,SV | SV,SE,P0 | System Reset | See Family Spec |
| SCU_FDR | Fractional Divider Register | $0038_H$ | U,SV | SV,SE,P0 | System Reset | See Family Spec |
| SCU_EXTCON | External Clock Control Register | $003C_H$ | U,SV | SV,SE,P0 | System Reset | See Family Spec |
| SCU_CCUCON2 | CCU Clock Control Register 2 | $0040_H$ | U,SV | SV,SE,P0 | System Reset | 24 |
| SCU_CCUCON3 | CCU Clock Control Register 3 | $0044_H$ | U,SV | SV,SE,P0 | System Reset | See Family Spec |
| SCU_CCUCON4 | CCU Clock Control Register 4 | $0048_H$ | U,SV | SV,SE,P0 | System Reset | See Family Spec |
| SCU_CCUCON5 | CCU Clock Control Register 5 | $004C_H$ | U,SV | SV,SE,P0 | System Reset | See Family Spec |
| SCU_RSTSTAT | Reset Status Register | $0050_H$ | U,SV | BE | See page 17 | 17 |
| | Reserved ($0004_H$ Byte) | $0054_H$ | BE | BE | | |
| SCU_RSTCON | Reset Configuration Register | $0058_H$ | U,SV | SV,SE,P0 | See page 19 | 19 |
| SCU_ARSTDIS | Application Reset Disable Register | $005C_H$ | U,SV | SV,E,P0 | PowerOn Reset | 21 |
| SCU_SWRSTCON | Software Reset Configuration Register | $0060_H$ | U,SV | SV,E,P0 | See Family Spec | See Family Spec |
| SCU_RSTCON2 | Additional Reset Control Register | $0064_H$ | U,SV | SV,E,P0 | See Family Spec | See Family Spec |
| SCU_RSTCON3 | Reset Configuration Register 3 | $0068_H$ | U,SV | SV,E,P0 | See Family Spec | See Family Spec |
| | Reserved ($0004_H$ Byte) | $006C_H$ | BE | BE | | |
| SCU_ESRCFGx (x=0-1) | ESRx Input Configuration Register | $0070_H$+x *4 | U,SV | SV,E,P0 | System Reset | See Family Spec |
| SCU_ESROCFG | ESR Output Configuration Register | $0078_H$ | U,SV | SV,E,P0 | System Reset | See Family Spec |

**Table 47 Register Overview - SCU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SCU_SYSCON | System Control Register | $007C_H$ | U,SV | U,SV,P0 | System Reset | See Family Spec |
| SCU_CCUCON6 | CCU Clock Control Register 6 | $0080_H$ | U,SV | SV,SE,P0 | System Reset | See Family Spec |
| SCU_CCUCON7 | CCU Clock Control Register 7 | $0084_H$ | U,SV | SV,SE,P0 | System Reset | See Family Spec |
| SCU_CCUCON8 | CCU Clock Control Register 8 | $0088_H$ | U,SV | SV,SE,P0 | System Reset | See Family Spec |
| | Reserved ($0004_H$ Byte) | $0098_H$ | BE | BE | | |
| SCU_PDR | ESR Pad Driver Mode Register | $009C_H$ | U,SV | SV,E,P0 | System Reset | See Family Spec |
| SCU_IOCR | Input/Output Control Register | $00A0_H$ | U,SV | U,SV,P0 | System Reset | See Family Spec |
| SCU_OUT | ESR Output Register | $00A4_H$ | U,SV | U,SV,P0 | System Reset | See Family Spec |
| SCU_OMR | ESR Output Modification Register | $00A8_H$ | U,SV | U,SV,P0 | System Reset | See Family Spec |
| SCU_IN | ESR Input Register | $00AC_H$ | U,SV | BE | System Reset | See Family Spec |
| | Reserved ($0004_H$ Byte) | $00BC_H$ | BE | BE | | |
| SCU_STSTAT | Start-up Status Register | $00C0_H$ | U,SV | BE | PowerOn Reset | See Family Spec |
| SCU_STCON | Start-up Configuration Register | $00C4_H$ | U,SV | ST,P0 | Application Reset | See Family Spec |
| SCU_PMCSR0 | Power Management Control and Status Register | $00C8_H$ | U,SV | SE,CE0,SV, P0 | Application Reset | See Family Spec |
| SCU_PMCSR1 | Power Management Control and Status Register | $00CC_H$ | U,SV | SE,CE1,SV, P0 | Application Reset | See Family Spec |

**Table 47 Register Overview - SCU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SCU_PMCSR2 | Power Management Control and Status Register | $00D0_H$ | U,SV | SE,CE2,SV, P0 | Application Reset | See Family Spec |
| SCU_PMCSR3 | Power Management Control and Status Register | $00D4_H$ | U,SV | SE,CE3,SV, P0 | Application Reset | See Family Spec |
| SCU_PMCSR4 | Power Management Control and Status Register | $00D8_H$ | U,SV | SE,CE4,SV, P0 | Application Reset | See Family Spec |
| SCU_PMCSR5 | Power Management Control and Status Register | $00DC_H$ | U,SV | SE,CE5,SV, P0 | Application Reset | See Family Spec |
| SCU_PMSTAT0 | Power Management Status Register 0 | $00E4_H$ | U,SV | BE | Application Reset | See Family Spec |
| SCU_PMSWCR1 | Standby and Wake-up Control Register 1 | $00E8_H$ | U,SV | SV,SE,P0 | Cold PowerOn Reset | See Family Spec |
| | Reserved ($0020_H$ Byte) | $00F0_H$ | BE | BE | | |
| SCU_EMSR | Emergency Stop Register | $00FC_H$ | U,SV | SV,SE,P0 | Application Reset | See Family Spec |
| SCU_EMSSW | Emergency Stop Software set and clear register | $0100_H$ | U,SV | U,SV,P0 | Application Reset | See Family Spec |
| SCU_DTSCSTAT | Core Die Temperature Sensor Status Register | $0104_H$ | U,SV | BE | Application Reset | See Family Spec |
| SCU_DTSCLIM | Core Die Temperature Sensor Limit Register | $0108_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| | Reserved ($0060_H$ Byte) | $0114_H$ | BE | BE | | |
| SCU_TRAPDIS1 | Trap Disable Register 1 | $0120_H$ | U,SV | SV,E,P0 | Application Reset | 22 |
| SCU_TRAPSTAT | Trap Status Register | $0124_H$ | U,SV | BE | System Reset | See Family Spec |
| SCU_TRAPSET | Trap Set Register | $0128_H$ | U,SV | SV,E,P0 | System Reset | See Family Spec |

**Table 47 Register Overview - SCU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SCU_TRAPCLR | Trap Clear Register | $012C_H$ | U,SV | U,SV,P0 | System Reset | See Family Spec |
| SCU_TRAPDIS0 | Trap Disable Register 0 | $0130_H$ | U,SV | SV,E,P0 | Application Reset | 23 |
| SCU_LCLCON0 | LCL CPU0 and CPU2 Control Register | $0134_H$ | U,SV | SV,SE,ST,P0 | See page 11 | 11 |
| SCU_LCLCON1 | LCL CPU1 and CPU3 Control Register | $0138_H$ | U,SV | SV,SE,ST,P0 | See page 11 | 11 |
| SCU_LCLTEST | LCL Test Register | $013C_H$ | U,SV | U,SV,P0 | System Reset | 12 |
| SCU_CHIPID | Chip Identification Register | $0140_H$ | U,SV | ST,P0 | See Family Spec | See Family Spec |
| SCU_MANID | Manufacturer Identification Register | $0144_H$ | U,SV | BE | System Reset | See Family Spec |
| SCU_SWAPCTRL | Address Map Control Register | $014C_H$ | U,SV | ST,P0 | System Reset | See Family Spec |
| | Reserved ($0060_H$ Byte) | $0158_H$ | BE | BE | | |
| | Reserved ($0060_H$ Byte) | $015C_H$ | BE | BE | | |
| | Reserved ($0060_H$ Byte) | $0160_H$ | BE | BE | | |
| SCU_LBISTCTRL0 | Logic BIST Control 0 Register | $0164_H$ | U,SV | SV,SE,P0 | See Family Spec | See Family Spec |
| SCU_LBISTCTRL1 | Logic BIST Control 1 Register | $0168_H$ | U,SV | SV,SE,P0 | See Family Spec | See Family Spec |
| SCU_LBISTCTRL2 | Logic BIST Control 2 Register | $016C_H$ | U,SV | SV,SE,P0 | See page 14 | 14 |
| SCU_LBISTCTRL3 | Logic BIST Control 3 Register | $0170_H$ | U,SV | BE | See Family Spec | See Family Spec |
| | Reserved ($0020_H$ Byte) | $0178_H$ | BE | BE | | |
| SCU_STMEM1 | Start-up Memory Register 1 | $0184_H$ | U,SV | ST,P0 | PowerOn Reset | See Family Spec |
| SCU_STMEM2 | Start-up Memory Register 2 | $0188_H$ | U,SV | ST,P0 | System Reset | See Family Spec |

**Table 47 Register Overview - SCU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SCU_PDISC | Pad Disable Control Register | $018C_H$ | U,SV | SV,E,P0 | System Reset | See Family Spec |
| | Reserved ($0020_H$ Byte) | $0194_H$ | BE | BE | | |
| SCU_PMTRCSR0 | Power Management Transition Control and Status Register 0 | $0198_H$ | U,SV | SV,SE,P0 | Cold PowerOn Reset | See Family Spec |
| SCU_PMTRCSR1 | Power Management Transition Control and Status Register 1 | $019C_H$ | U,SV | SV,SE,P0 | Cold PowerOn Reset | See Family Spec |
| SCU_PMTRCSR2 | Power Management Transition Control and Status Register 2 | $01A0_H$ | U,SV | SV,SE,P0 | Cold PowerOn Reset | See Family Spec |
| SCU_PMTRCSR3 | Power Management Transition Control and Status Register 3 | $01A4_H$ | U,SV | SV,SE,P0 | Cold PowerOn Reset | See Family Spec |
| SCU_STMEM3 | Start-up Memory Register 3 | $01C0_H$ | U,SV | ST,P0 | Application Reset | See Family Spec |
| SCU_STMEM4 | Start-up Memory Register 4 | $01C4_H$ | U,SV | ST,P0 | Cold PowerOn Reset | See Family Spec |
| SCU_STMEM5 | Start-up Memory Register 5 | $01C8_H$ | U,SV | ST,P0 | PowerOn Reset | See Family Spec |
| SCU_STMEM6 | Start-up Memory Register 6 | $01CC_H$ | U,SV | ST,P0 | System Reset | See Family Spec |
| SCU_OVCENABLE | Overlay Enable Register | $01E0_H$ | U,SV | SV,SE,P0 | Application Reset | 14 |
| SCU_OVCCON | Overlay Control Register | $01E4_H$ | U,SV | SV,P0 | Application Reset | 15 |
| SCU_EIFILT | External Input Filter Register | $020C_H$ | U,SV | SE,SV,P0 | Application Reset | See Family Spec |
| SCU_EICRi (i=0-3) | External Input Channel Register i | $0210_H$+i*4 | U,SV | SE,SV,P0 | Application Reset | See Family Spec |
| SCU_EIFR | External Input Flag Register | $0220_H$ | U,SV | BE | Application Reset | See Family Spec |

**Table 47 Register Overview - SCU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SCU_FMR | Flag Modification Register | $0224_H$ | U,SV | U,SV,P0 | Application Reset | See Family Spec |
| SCU_PDRR | Pattern Detection Result Register | $0228_H$ | U,SV | BE | Application Reset | See Family Spec |
| SCU_IGCRj (j=0-3) | Flag Gating Register j | $022C_H$+j*4 | U,SV | SE,SV,P0 | Application Reset | See Family Spec |
| | Reserved ($0030_H$ Byte) | $023C_H$ | BE | BE | | |
| SCU_WDTCPUyCON0 (y=0) (y=1) (y=2) | CPUy WDT Control Register 0 | $024C_H$+y*12 | U,SV | U,SV,32,CPUy (y=CPU number) | Application Reset | See Family Spec |
| SCU_WDTCPUyCON1 (y=0) (y=1) (y=2) | CPUy WDT Control Register 1 | $0250_H$+y*12 | U,SV | SV,CEy,P0 | Application Reset | See Family Spec |
| SCU_WDTCPUySR (y=0) (y=1) (y=2) | CPUy WDT Status Register | $0254_H$+y*12 | U,SV | BE | Application Reset | See Family Spec |
| SCU_EICON0 | ENDINIT Global Control Register 0 | $029C_H$ | U,SV | U,SV,32,P0 | Application Reset | See Family Spec |
| SCU_EICON1 | ENDINIT Global Control Register 1 | $02A0_H$ | U,SV | SV,E,P0 | Application Reset | See Family Spec |
| SCU_EISR | ENDINIT Timeout Counter Status Register | $02A4_H$ | U,SV | BE | Application Reset | See Family Spec |
| SCU_WDTSCON0 | Safety WDT Control Register 0 | $02A8_H$ | U,SV | U,SV,32,P1 | Application Reset | See Family Spec |
| SCU_WDTSCON1 | Safety WDT Control Register 1 | $02AC_H$ | U,SV | SV,SE,P1 | Application Reset | See Family Spec |
| SCU_WDTSSR | Safety WDT Status Register | $02B0_H$ | U,SV | BE | Application Reset | See Family Spec |

**Table 47 Register Overview - SCU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SCU_SEICON0 | Safety ENDINIT Control Register 0 | $02B4_H$ | U,SV | U,SV,32,P1 | Application Reset | See Family Spec |
| SCU_SEICON1 | Safety ENDINIT Control Register 1 | $02B8_H$ | U,SV | SV,SE,P1 | Application Reset | See Family Spec |
| SCU_SEISR | Safety ENDINIT Timeout Status Register | $02BC_H$ | U,SV | BE | Application Reset | See Family Spec |
| | Reserved ($0440_H$ Byte) | $02DC_H$ | BE | BE | | |
| SCU_ACCEN11 | Access Enable Register 11 | $03F0_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| SCU_ACCEN10 | Access Enable Register 10 | $03F4_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| SCU_ACCEN01 | Access Enable Register 01 | $03F8_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| SCU_ACCEN00 | Access Enable Register 00 | $03FC_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| | Reserved ($0440_H$ Byte) | $0400_H$ | BE | BE | | |

## 9.3 TC35x Specific Registers

### 9.3.1 SCU: Connections to FPI/BPI bus

#### LCL CPU0 and CPU2 Control Register

Provides control for CPU0and CPU2 Lockstep Comparator Logic blocks.

**SCU_LCLCON0**

**LCL CPU0 and CPU2 Control Register** **($0134_H$)** **Reset Value: Table 48**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **LSEN0** | | | | | | | | **0** | | | | | | | **LS0** |
| rw | | | | | | | | r | | | | | | | rh |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1** | | | | | | | | **0** | | | | | | | **0** |
| rw | | | | | | | | r | | | | | | | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **LS0** | 16 | rh | **Lockstep Mode Status**<br>This bit indicates whether CPU0 is currently running in lockstep monitor mode<br>$0_B$ Not in lockstep mode<br>$1_B$ Running in lockstep mode |
| **LSEN0** | 31 | rw | **Lockstep Enable**<br>This bit may only be written by SSW during boot.<br>Enable lockstep CPU monitoring for the associated processor core, CPU0.<br>After cold reset, lockstep is enabled by default. The LSEN bit may be cleared during the boot to disable lockstep mode. SMU lockstep fault reporting should be disabled when lockstep is disabled.<br>$0_B$ Lockstep is disabled<br>$1_B$ Lockstep enabled (Default after Cold Power-On Reset) |
| **0** | 0,<br>14:1,<br>30:17 | r | **Reserved in this product**<br>Reserved |
| **1** | 15 | rw | **Reserved in this product**<br>Reserved |

**Table 48 Reset Values of SCU_LCLCON0**

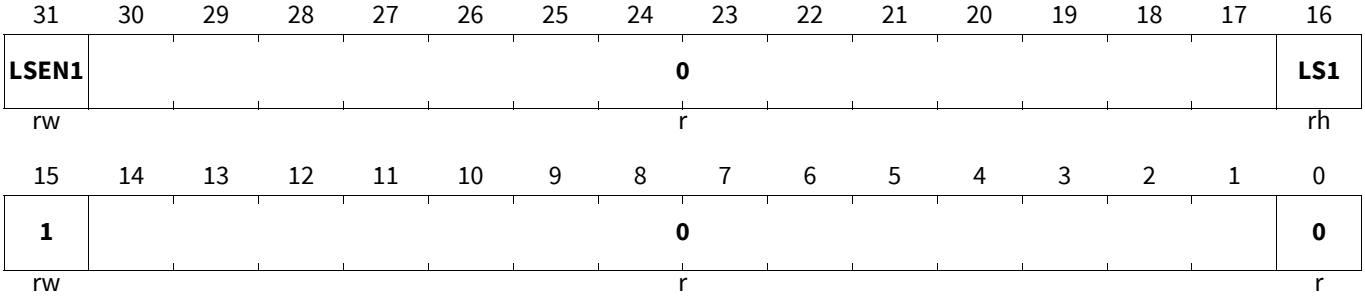
| Reset Type | Reset Value | Note |
|---|---|---|
| Cold PowerOn Reset | $8001\ 0000_H$ | |

#### LCL CPU1 and CPU3 Control Register

Provides control for CPU1 and CPU3 Lockstep Comparator Logic blocks.

### SCU_LCLCON1

**LCL CPU1 and CPU3 Control Register** **($0138_H$)** **Reset Value: Table 49**




| Field | Bits | Type | Description |
|---|---|---|---|
| **LS1** | 16 | rh | **Lockstep Mode Status**<br>This bit indicates whether CPU1 is currently running in lockstep monitor mode<br>$0_B$ Not in lockstep mode<br>$1_B$ Running in lockstep mode |
| **LSEN1** | 31 | rw | **Lockstep Enable**<br>This bit may only be written by SSW during boot.<br>Enable lockstep CPU monitoring for the associated processor core, CPU1.If the product has no lockstep capability for CPU1, then this enables only the PFLASH access monitoring for CPU1.<br>After cold reset, lockstep is enabled by default. The LSEN bit may be cleared during the boot to disable lockstep mode. SMU lockstep fault reporting should be disabled when lockstep is disabled.<br>$0_B$ Lockstep is disabled<br>$1_B$ Lockstep enabled (Default after Cold Power-On Reset) |
| **0** | 0,<br>14:1,<br>30:17 | r | **Reserved in this product**<br>Reserved |
| **1** | 15 | rw | **Reserved in this product**<br>Reserved |

**Table 49 Reset Values of SCU_LCLCON1**

| Reset Type | Reset Value | Note |
|---|---|---|
| Cold PowerOn Reset | $8001\ 0000_H$ | |

### LCL Test Register

Provides the capability for software to inject a fault condition into the comparators of each Lockstep Comparator Logic block. The implementation should generate a single cycle fault each time the bit is written with ‘1’.

## SCU_LCLTEST

**LCL Test Register** **(013C$_H$)** **System Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | 0 | 0 | 0 | PLCLT2 | PLCLT1 | PLCLT0 |
| r | | | | | | | | | | r | r | r | w | w | w |

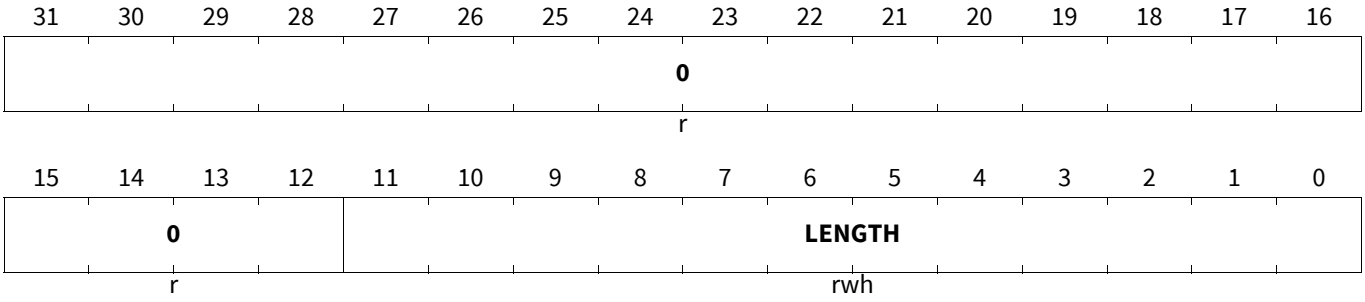
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | 0 | 0 | 0 | LCLT2 | LCLT1 | LCLT0 |
| r | | | | | | | | | | r | r | r | w | w | w |

| Field | Bits | Type | Description |
|---|---|---|---|
| **LCLT0** | 0 | w | **LCL0 Lockstep Test**<br>Fault injection for LCL0. Reads as zero.<br>$0_B$ No action<br>$1_B$ Inject single fault in LCL0 |
| **LCLT1** | 1 | w | **LCL1 Lockstep Test**<br>Fault injection for LCL1. Reads as zero.<br>$0_B$ No action<br>$1_B$ Inject single fault in LCL1 |
| **LCLT2** | 2 | w | **LCL2 Lockstep Test**<br>Fault injection for LCL2. Reads as zero.<br>$0_B$ No action<br>$1_B$ Inject single fault in LCL2 |
| **PLCLT0** | 16 | w | **PFI0 Lockstep Test**<br>Fault injection for PFI0 lockstep. Reads as zero.<br>$0_B$ No action<br>$1_B$ Inject single fault in PFI0 lockstep |
| **PLCLT1** | 17 | w | **PFI1 Lockstep Test**<br>Fault injection for PFI1 lockstep. Reads as zero.<br>$0_B$ No action<br>$1_B$ Inject single fault in PFI1 lockstep |
| **PLCLT2** | 18 | w | **PFI2 Lockstep Test**<br>Fault injection for PFI2 lockstep. Reads as zero.<br>$0_B$ No action<br>$1_B$ Inject single fault in PFI2 lockstep |
| **0** | 3,<br>4,<br>5,<br>15:6,<br>19,<br>20,<br>21,<br>31:22 | r | **Reserved in this product**<br>will be read as 0 , should be written as 0 |

### Logic BIST Control 2 Register

**SCU_LBISTCTRL2**

**Logic BIST Control 2 Register** **($016C_H$)** **Reset Value: Table 50**




| Field | Bits | Type | Description |
|---|---|---|---|
| **LENGTH** | 11:0 | rwh | **LBIST Maximum Scan-Chain Length**<br>This field defines the number of shift-cycles for each LBIST scan-load. It will be automatically loaded with the product-specific value, stored in Flash config-sector during startup-software execution. |
| **0** | 31:12 | r | **Reserved**<br>Read as 0; should be written with 0. |

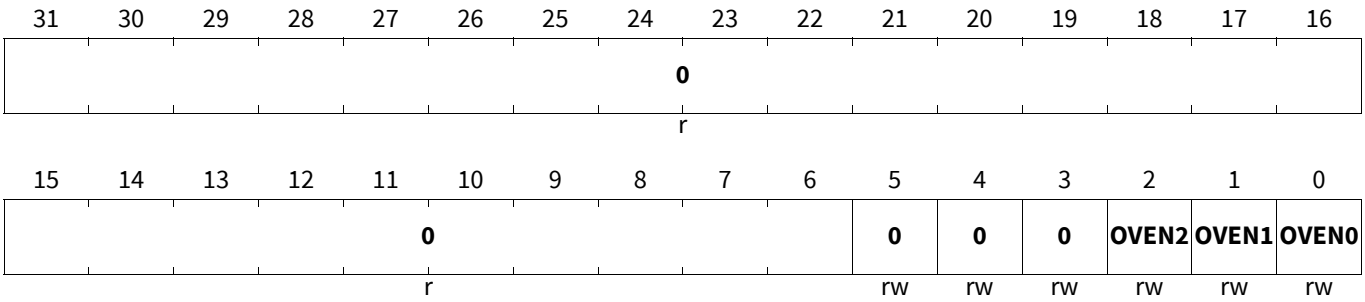
**Table 50 Reset Values of SCU_LBISTCTRL2**

| Reset Type | Reset Value | Note |
|---|---|---|
| System Reset | $0000\ 0000_H$ | |
| CFS Value | $0000\ 0040_H$ | |

### Overlay Enable Register

**SCU_OVCENABLE**

**Overlay Enable Register** **($01E0_H$)** **Application Reset Value: $0000\ 0000_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **OVEN0** | 0 | rw | **Overlay Enable 0**<br>$0_B$ OVC is disabled on CPU0. All Overlay redirections are disabled regardless of the state of OVC0_RABRy.OVEN.<br>$1_B$ OVC is enabled on CPU0. |

| Field | Bits | Type | Description |
|---|---|---|---|
| **OVEN1** | 1 | rw | **Overlay Enable 1 (If product has CPU1)**<br>$0_B$ OVC is disabled on CPU1. All Overlay redirections are disabled regardless of the state of OVC1_RABRy.OVEN.<br>$1_B$ OVC is enabled on CPU1. |
| **OVEN2** | 2 | rw | **Overlay Enable 2 (If product has CPU2)**<br>$0_B$ OVC is disabled on CPU2. All Overlay redirections are disabled regardless of the state of OVC2_RABRy.OVEN.<br>$1_B$ OVC is enabled on CPU2. |
| **0** | 3,<br>4,<br>5 | rw | **Reserved in this Product**<br>will be read as 0 , should be written as 0 |
| **0** | 31:6 | r | **Reserved**<br>Read/write 0. |

### Overlay Control Register

**SCU_OVCCON**

**Overlay Control Register (01E4$_H$) Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | POVCONF | OVCONF | 0 | | | | | DCINVAL | OVSTP | OVSTRT |
| r | | | | | | w | rw | r | | | | | w | w | w |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | 0 | 0 | 0 | CSEL2 | CSEL1 | CSEL0 |
| r | | | | | | | | | | r | r | r | w | w | w |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CSEL0** | 0 | w | **CPU Select 0**<br>Return 0 if read.<br>$0_B$ CPU0 not affected,<br>$1_B$ Action selected by OVSTRT, OVSTP, DCINVAL bits, set by the same register write access, is applied to CPU0. |
| **CSEL1** | 1 | w | **CPU Select 1 (If product has CPU1)**<br>Return 0 if read.<br>$0_B$ CPU1 not affected,<br>$1_B$ Action selected by OVSTRT, OVSTP, DCINVAL bits, set by the same register write access, is applied to CPU1. |
| **CSEL2** | 2 | w | **CPU Select 2 (If product has CPU2)**<br>Return 0 if read.<br>$0_B$ CPU2 not affected,<br>$1_B$ Action selected by OVSTRT, OVSTP, DCINVAL bits, set by the same register write access, is applied to CPU2. |

| Field | Bits | Type | Description |
|---|---|---|---|
| **OVSTRT** | 16 | w | **Overlay Start**<br>CPUs which are not selected are not affected.<br>No action is taken if OVSTP is also set.<br>Return 0 if read.<br>$0_B$ No action<br>$1_B$ For each CPU selected with CSEL, all the blocks selected with OVCx_OSEL will be activated. In the selected CPUs all the blocks deselected with OVCx_OSEL will be deactivated. |
| **OVSTP** | 17 | w | **Overlay Stop**<br>CPUs which are not selected are not affected<br>No action is taken if OVSTRT is also set.<br>Return 0 if read.<br>$0_B$ No action<br>$1_B$ For CPUs selected with CSEL, all the overlay blocks are deactivated. OVCx_RABRy.OVEN bits are cleared. |
| **DCINVAL** | 18 | w | **Data Cache Invalidate**<br>No function in devices without data cache in CPU.<br>Data Cache is affected only in the CPUs selected with CSEL.<br>Return 0 if read.<br>$0_B$ No action<br>$1_B$ Data Cache Lines in DMI are invalidated[1)] |
| **OVCONF** | 24 | rw | **Overlay Configured**<br>Overlay configured status bit<br>This bit may be used as handshake bit between a debug device (via JTAG interface and Cerberus) and CPU(s).<br>$0_B$ Overlay is not configured or it has been already started<br>$1_B$ Overlay block control registers are configured and ready for overlay start |
| **POVCONF** | 25 | w | **Write Protection for OVCONF**<br>This bit enables OVCONF write during OVCCON write. Return 0 if read.<br>$0_B$ OVCONF remains unchanged.<br>$1_B$ OVCONF can be changed with write access to register OVCCON |
| **0** | 3,<br>4,<br>5,<br>15:6,<br>23:19,<br>31:26 | r | **Reserved in this Product**<br>Return 0 if read. |

1) Dirty (modified) cache lines are not effected by this operation. If data cache contains modified data, it is not invalidated, and has to be written-back and invalidated by the user. Therefore, it is highly recommended to either: access overlaid data in read-only mode, or use only non-cached access.

## Reset Status Register

### SCU_RSTSTAT

**Reset Status Register** **($0050_H$)** **Reset Value: Table 51**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **LBTERM** | **LBPORST** | **STBYR** | **HSMA** | **HSMS** | **SWD** | **EVR33** | **EVRC** | **R22** | **R21** | **CB3** | **CB1** | **CB0** | **0** | **PORST** |
| r | rh | rh | rh | rh | rh | rh | rh | rh | rX | rX | rh | rh | rh | r | rh |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | **0** | | | **0** | **0** | **0** | **STM2** | **STM1** | **STM0** | **SW** | **SMU** | **0** | **ESR1** | **ESR0** |
| | | r | | | r | r | r | rh | rh | rh | rh | rh | r | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ESR0** | 0 | rh | **Reset Request Trigger Reset Status for ESR0**<br>$0_B$ The last reset was not requested by this reset trigger<br>$1_B$ The last reset was requested by this reset trigger |
| **ESR1** | 1 | rh | **Reset Request Trigger Reset Status for ESR1**<br>$0_B$ The last reset was not requested by this reset trigger<br>$1_B$ The last reset was requested by this reset trigger |
| **SMU** | 3 | rh | **Reset Request Trigger Reset Status for SMU**<br>(See SMU section for SMU trigger sources, including Watchdog Timers)<br>$0_B$ The last reset was not requested by this reset trigger<br>$1_B$ The last reset was requested by this reset trigger |
| **SW** | 4 | rh | **Reset Request Trigger Reset Status for SW**<br>$0_B$ The last reset was not requested by this reset trigger<br>$1_B$ The last reset was requested by this reset trigger |
| **STM0** | 5 | rh | **Reset Request Trigger Reset Status for STM0 Compare Match**<br>$0_B$ The last reset was not requested by this reset trigger<br>$1_B$ The last reset was requested by this reset trigger |
| **STM1** | 6 | rh | **Reset Request Trigger Reset Status for STM1 Compare Match (If Product has STM1)**<br>$0_B$ The last reset was not requested by this reset trigger<br>$1_B$ The last reset was requested by this reset trigger |
| **STM2** | 7 | rh | **Reset Request Trigger Reset Status for STM2 Compare Match (If Product has STM2)**<br>$0_B$ The last reset was not requested by this reset trigger<br>$1_B$ The last reset was requested by this reset trigger |
| **PORST** | 16 | rh | **Reset Request Trigger Reset Status for PORST**<br>This bit is also set if the bits CB0, CB1, and CB3 are set in parallel.<br>$0_B$ This reset trigger has not occurred since the last clear (by RSTCON2.CLRC)<br>$1_B$ This reset trigger has occurred since the last clear (by RSTCON2.CLRC) |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CB0** | 18 | rh | **Reset Request Trigger Reset Status for Cerberus System Reset**<br>$0_B$ The last reset was not requested by this reset trigger<br>$1_B$ The last reset was requested by this reset trigger |
| **CB1** | 19 | rh | **Reset Request Trigger Reset Status for Cerberus Debug Reset**<br>$0_B$ The last reset was not requested by this reset trigger<br>$1_B$ The last reset was requested by this reset trigger |
| **CB3** | 20 | rh | **Reset Request Trigger Reset Status for Cerberus Application Reset**<br>$0_B$ The last reset was not requested by this reset trigger<br>$1_B$ The last reset was requested by this reset trigger |
| **R21** | 21 | rX | **Reserved - 0**<br>Read as 0; should be written with 0. |
| **R22** | 22 | rX | **Reserved - 0**<br>Read as 0; should be written with 0. |
| **EVRC** | 23 | rh | **Reset Request Trigger Reset Status for EVRC**<br>$0_B$ This reset trigger has not occurred since the last clear (by RSTCON2.CLRC)<br>$1_B$ This reset trigger has occurred since the last clear (by RSTCON2.CLRC) |
| **EVR33** | 24 | rh | **Reset Request Trigger Reset Status for EVR33**<br>$0_B$ This reset trigger has not occurred since the last clear (by RSTCON2.CLRC)<br>$1_B$ This reset trigger has occurred since the last clear (by RSTCON2.CLRC) |
| **SWD** | 25 | rh | **Reset Request Trigger Reset Status for Supply Watchdog (SWD)**<br>The Supply Watchdog trigger is described in Power Management Controller “Supply Monitoring” chapter<br>$0_B$ This reset trigger has not occurred since the last clear (by RSTCON2.CLRC)<br>$1_B$ This reset trigger has occurred since the last clear (by RSTCON2.CLRC) |
| **HSMS** | 26 | rh | **Reset Request Trigger Reset Status for HSM System Reset (HSM S)**<br>$0_B$ The last reset was not requested by this reset trigger<br>$1_B$ The last reset was requested by this reset trigger |
| **HSMA** | 27 | rh | **Reset Request Trigger Reset Status for HSM Application Reset (HSM A)**<br>$0_B$ The last reset was not requested by this reset trigger<br>$1_B$ The last reset was requested by this reset trigger |
| **STBYR** | 28 | rh | **Reset Request Trigger Reset Status for Standby Regulator Watchdog (STBYR)**<br>$0_B$ This reset trigger has not occurred since the last clear (by RSTCON2.CLRC)<br>$1_B$ This reset trigger has occurred since the last clear (by RSTCON2.CLRC) |

| Field | Bits | Type | Description |
|---|---|---|---|
| **LBPORST** | 29 | rh | **LBIST termination due to PORST**<br>This bitfield indicates if the LBIST was early terminated due to the occurrence of a Power On Reset. If the status of this bitfield is 0, the application must still check the LBTERM to check if the LBIST was terminated properly. This bitfield is cleared when the RSTCON2.CLRC is set.<br>$0_B$ LBIST was not terminated early due to a Power On Reset<br>$1_B$ LBIST early termination due to the occurrence of Power On Reset |
| **LBTERM** | 30 | rh | **LBIST was properly terminated**<br>This bitfield indicates if the LBIST was terminated properly. This bitfield is cleared when the RSTCON2.CLRC is set.<br>$0_B$ LBIST was not terminated properly<br>$1_B$ LBIST was terminated properly |
| **0** | 2,<br>8,<br>9,<br>10,<br>15:11,<br>17,<br>31 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 51 Reset Values of SCU_RSTSTAT**

| Reset Type | Reset Value | Note |
|---|---|---|
| Cold PowerOn Reset | $0XX1\ 0000_H$ | RSTSTAT |
| Cold PowerOn Reset | $1001\ 0000_H$ | RSTSTAT (Triggered by LVD Reset) |

### Reset Configuration Register

**SCU_RSTCON**
**Reset Configuration Register ($0058_H$) Reset Value: Table 52**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | | | **0** | | **0** | | **0** | |
| rw | | | | | | | | | | rw | | rw | | rw | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **STM2** | | **STM1** | | **STM0** | | **SW** | | **SMU** | | **0** | | **ESR1** | | **ESR0** | |
| rw | | rw | | rw | | rw | | rw | | rw | | rw | | rw | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ESR0** | 1:0 | rw | **ESR0 Reset Request Trigger Reset Configuration**<br>This bit field defines which reset is generated by a reset request trigger from ESR0 reset.<br>$00_B$ No reset is generated for a trigger of ESR0<br>$01_B$ A System Reset is generated for a trigger of ESR0 reset<br>$10_B$ An Application Reset is generated for a trigger of ESR0 reset<br>$11_B$ Reserved, do not use this combination |
| **ESR1** | 3:2 | rw | **ESR1 Reset Request Trigger Reset Configuration**<br>This bit field defines which reset is generated by a reset request trigger from ESR1 reset.<br>$00_B$ No reset is generated for a trigger of ESR1<br>$01_B$ A System Reset is generated for a trigger of ESR1 reset<br>$10_B$ An Application Reset is generated for a trigger of ESR1 reset<br>$11_B$ Reserved, do not use this combination |
| **SMU** | 7:6 | rw | **SMU Reset Request Trigger Reset Configuration**<br>This bit field defines which reset is generated by a reset request trigger from SMU reset.<br>$00_B$ No reset is generated for a trigger of SMU<br>$01_B$ A System Reset is generated for a trigger of SMU reset<br>$10_B$ An Application Reset is generated for a trigger of SMU reset<br>$11_B$ Reserved, do not use this combination |
| **SW** | 9:8 | rw | **SW Reset Request Trigger Reset Configuration**<br>This bit field defines which reset is generated by a reset request trigger from software reset.<br>$00_B$ No reset is generated for a trigger of software reset<br>$01_B$ A System Reset is generated for a trigger of Software reset<br>$10_B$ An Application Reset is generated for a trigger of Software reset<br>$11_B$ Reserved, do not use this combination |
| **STM0** | 11:10 | rw | **STM0 Reset Request Trigger Reset Configuration**<br>This bit field defines which reset is generated by a reset request trigger from STM0 compare match reset.<br>$00_B$ No reset is generated for an STM0 trigger<br>$01_B$ A System Reset is generated for a trigger of STM0 reset<br>$10_B$ An Application Reset is generated for a trigger of STM0 reset<br>$11_B$ Reserved, do not use this combination |
| **STM1** | 13:12 | rw | **STM1 Reset Request Trigger Reset Configuration (If Product has STM1)**<br>This bit field defines which reset is generated by a reset request trigger from STM1 compare match reset.<br>$00_B$ No reset is generated for a trigger of STM1<br>$01_B$ A System Reset is generated for a trigger of STM1 reset<br>$10_B$ An Application Reset is generated for a trigger of STM1 reset<br>$11_B$ Reserved, do not use this combination |

| Field | Bits | Type | Description |
|---|---|---|---|
| **STM2** | 15:14 | rw | **STM2 Reset Request Trigger Reset Configuration (If Product has STM2)**<br>This bit field defines which reset is generated by a reset request trigger from STM2 compare match reset.<br>$00_B$ No reset is generated for a trigger of STM2<br>$01_B$ A System Reset is generated for a trigger of STM2 reset<br>$10_B$ An Application Reset is generated for a trigger of STM2 reset<br>$11_B$ Reserved, do not use this combination |
| **0** | 5:4,<br>17:16,<br>19:18,<br>21:20,<br>31:22 | rw | **Reserved**<br>Should be written with 0. |

**Table 52 Reset Values of SCU_RSTCON**

| Reset Type | Reset Value | Note |
|---|---|---|
| PowerOn Reset | $0000\ 0282_H$ | RSTCON |

### Application Reset Disable Register

**SCU_ARSTDIS**
**Application Reset Disable Register ($005C_H$) PowerOn Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

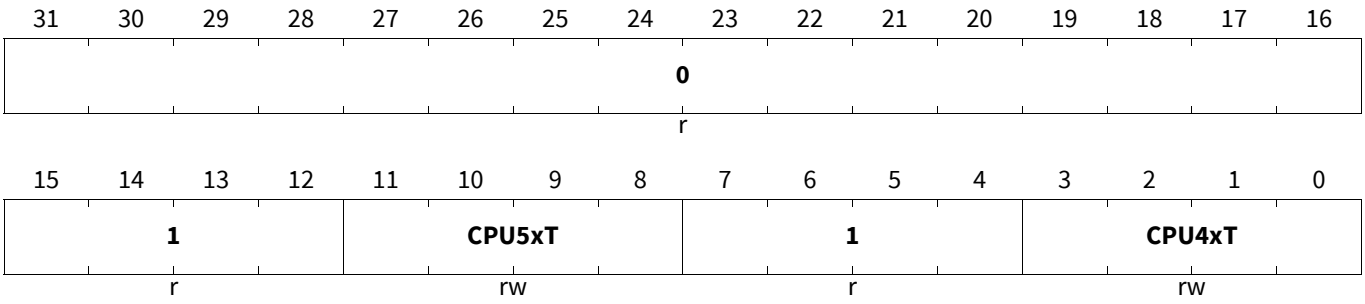
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | 0 | | 0 | 0 | 0 | STM2 DIS | STM1 DIS | STM0 DIS |
| r | | | | | | | | rw | | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **STM0DIS** | 0 | rw | **STM0 Disable Reset**<br>This bit field defines if an Application Reset leads to an reset for the STM0.<br>$0_B$ An Application Reset resets the STM0<br>$1_B$ An Application Reset has no effect for the STM0 |
| **STM1DIS** | 1 | rw | **STM1 Disable Reset**<br>This bit field defines if an Application Reset leads to an reset for the STM1.<br>$0_B$ An Application Reset resets the STM1<br>$1_B$ An Application Reset has no effect for the STM1 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **STM2DIS** | 2 | rw | **STM2 Disable Reset**<br>This bit field defines if an Application Reset leads to an reset for the STM2.<br>$0_B$ An Application Reset resets the STM2<br>$1_B$ An Application Reset has no effect for the STM2 |
| **0** | 3,<br>4,<br>5,<br>7:6 | rw | **Reserved**<br>Should be written with 0. |
| **0** | 31:8 | r | **Reserved**<br>Read as 0; should be written with 0. |

### Trap Disable Register 1

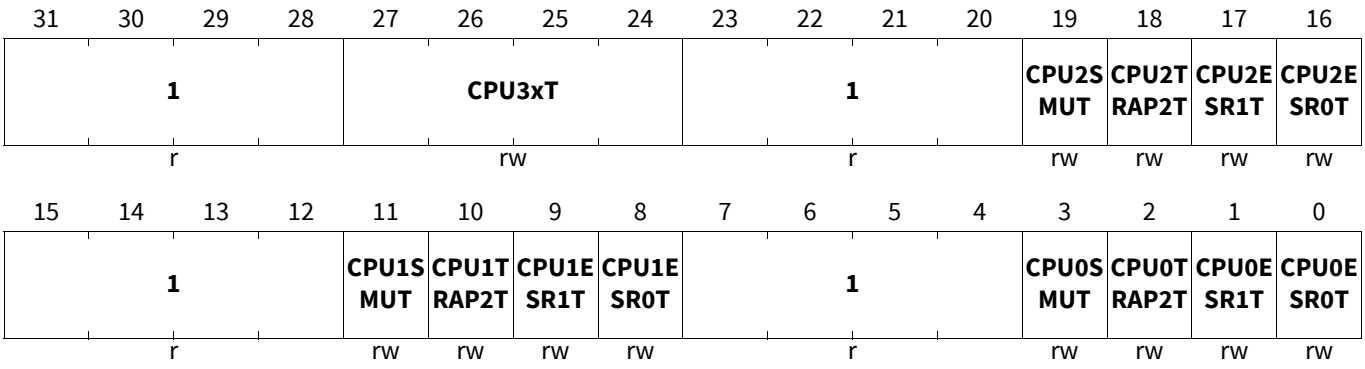
**SCU_TRAPDIS1**
**Trap Disable Register 1** **($0120_H$)** **Application Reset Value: 0000 FFFF$_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **CPU4xT** | 3:0 | rw | **Reserved in this product** |
| **CPU5xT** | 11:8 | rw | **Reserved in this product** |
| **1** | 7:4,<br>15:12 | r | **Reserved**<br>Must only be written with one. Read as one. |
| **0** | 31:16 | r | **Reserved**<br>Read as zero |

## Trap Disable Register 0

**SCU_TRAPDIS0**
**Trap Disable Register 0** **($0130_H$)** **Application Reset Value: FFFF FFFF$_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **CPU0ESR0T** | 0 | rw | **Disable Trap Request ESR0T on CPU0**<br>$0_B$ A CPU0 trap request can be generated for this source<br>$1_B$ No trap request can be generated for this source |
| **CPU0ESR1T** | 1 | rw | **Disable Trap Request ESR1T on CPU0**<br>$0_B$ A CPU0 trap request can be generated for this source<br>$1_B$ No trap request can be generated for this source |
| **CPU0TRAP2T** | 2 | rw | **Disable Trap Request TRAP2T on CPU0**<br>$0_B$ A CPU0 trap request can be generated for this source<br>$1_B$ No trap request can be generated for this source |
| **CPU0SMUT** | 3 | rw | **Disable Trap Request SMUT on CPU0**<br>$0_B$ A CPU0 trap request can be generated for this source<br>$1_B$ No trap request can be generated for this source |
| **CPU1ESR0T** | 8 | rw | **Disable Trap Request ESR0T on CPU1 (If product has CPU1)**<br>$0_B$ A CPU1 trap request can be generated for this source<br>$1_B$ No trap request can be generated for this source |
| **CPU1ESR1T** | 9 | rw | **Disable Trap Request ESR1T on CPU1 (If product has CPU1)**<br>$0_B$ A CPU1 trap request can be generated for this source<br>$1_B$ No trap request can be generated for this source |
| **CPU1TRAP2T** | 10 | rw | **Disable Trap Request TRAP2T on CPU1 (If product has CPU1)**<br>$0_B$ A CPU1 trap request can be generated for this source<br>$1_B$ No trap request can be generated for this source |
| **CPU1SMUT** | 11 | rw | **Disable Trap Request SMUT on CPU1 (If product has CPU1)**<br>$0_B$ A CPU1 trap request can be generated for this source<br>$1_B$ No trap request can be generated for this source |
| **CPU2ESR0T** | 16 | rw | **Disable Trap Request ESR0T on CPU2 (If product has CPU2)**<br>$0_B$ A CPU2 trap request can be generated for this source<br>$1_B$ No trap request can be generated for this source |
| **CPU2ESR1T** | 17 | rw | **Disable Trap Request ESR1T on CPU2 (If product has CPU2)**<br>$0_B$ A CPU2 trap request can be generated for this source<br>$1_B$ No trap request can be generated for this source |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CPU2TRAP2T** | 18 | rw | **Disable Trap Request TRAP2T on CPU2 (If product has CPU2)**<br>$0_B$ A CPU2 trap request can be generated for this source<br>$1_B$ No trap request can be generated for this source |
| **CPU2SMUT** | 19 | rw | **Disable Trap Request SMUT on CPU2 (If product has CPU2)**<br>$0_B$ A CPU2 trap request can be generated for this source<br>$1_B$ No trap request can be generated for this source |
| **CPU3xT** | 27:24 | rw | **Reserved in this product** |
| **1** | 7:4, 15:12, 23:20, 31:28 | r | **Reserved**<br>Must only be written with one. Read as one. |

### CCU Clock Control Register 2

**SCU_CCUCON2**

**CCU Clock Control Register 2** **($0040_H$)** **System Reset Value: $0700\ 0101_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **LCK** | **0** | | | | **HSPDMPERON** | **ERAYPERON** | **1** | **0** | | | | | | | |
| rh | rw | | | | rw | rw | rw | rw | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | **CLKSELASCLINS** | | **ASCLINSDIV** | | | | **0** | | | | **ASCLINFDIV** | | | |
| rw | | rw | | rw | | | | rw | | | | rw | | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ASCLINFDIV** | 3:0 | rw | **ASCLIN Fast Divider Reload Value**<br>The resulting ASCLIN frequency is configured to $f_{ASCLINF} = f_{source2}$ / ASCLINFDIV for the allowed configurations. For ASCLINFDIV = $0000_B$ the clock is shut off. $f_{source2}$ could be configured either to $f_{PLL2}$ (CLKSEL = $01_B$) or $f_{BACK}$ (CLKSEL = $00_B$)<br>$0_H$ $f_{ASCLINF}$ is stopped<br>$1_H$ $f_{ASCLINF} = f_{source2}$<br>$2_H$ $f_{ASCLINF} = f_{source2}/2$<br>$3_H$ $f_{ASCLINF} = f_{source2}/3$<br>$4_H$ $f_{ASCLINF} = f_{source2}/4$<br>$5_H$ $f_{ASCLINF} = f_{source2}/5$<br>$6_H$ $f_{ASCLINF} = f_{source2}/6$<br>$7_H$ Reserved, do not use this combination<br>$8_H$ $f_{ASCLINF} = f_{source2}/8$<br>$9_H$ Reserved, do not use this combination<br>$A_H$ $f_{ASCLINF} = f_{source2}/10$<br>$B_H$ Reserved, do not use this combination<br>$C_H$ $f_{ASCLINF} = f_{source2}/12$<br>$D_H$ Reserved, do not use this combination<br>$E_H$ Reserved, do not use this combination<br>$F_H$ $f_{ASCLINF} = f_{source2}/15$ |
| **ASCLINSDIV** | 11:8 | rw | **ASCLIN Slow Divider Reload Value**<br>The resulting ASCLIN frequency is configured to $f_{ASCLINSI} = f_{source1}$ / ASCLINSDIV for the allowed configurations. For ASCLINSDIV = $0000_B$ the clock is shut off. $f_{source1}$ could be configured either to $f_{PLL1}$ (CLKSEL = $01_B$) or $f_{BACK}$ (CLKSEL = $00_B$)<br>$0_H$ $f_{ASCLINSI}$ is stopped<br>$1_H$ $f_{ASCLINSI} = f_{source1}$<br>$2_H$ $f_{ASCLINSI} = f_{source1}/2$<br>$3_H$ $f_{ASCLINSI} = f_{source1}/3$<br>$4_H$ $f_{ASCLINSI} = f_{source1}/4$<br>$5_H$ $f_{ASCLINSI} = f_{source1}/5$<br>$6_H$ $f_{ASCLINSI} = f_{source1}/6$<br>$7_H$ Reserved, do not use this combination<br>$8_H$ $f_{ASCLINSI} = f_{source1}/8$<br>$9_H$ Reserved, do not use this combination<br>$A_H$ $f_{ASCLINSI} = f_{source1}/10$<br>$B_H$ Reserved, do not use this combination<br>$C_H$ $f_{ASCLINSI} = f_{source1}/12$<br>$D_H$ Reserved, do not use this combination<br>$E_H$ Reserved, do not use this combination<br>$F_H$ $f_{ASCLINSI} = f_{source1}/15$ |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CLKSELASCLINS** | 13:12 | rw | **Clock Selection for ASCLINS**<br>This bit field defines the clock source that is used for the clock generation of $f_{ASCLINS}$.<br><br>*Note: For switching between two non-zero configurations the following sequence has to be applied: First step is to switch to $00_B$. Second step is to switch to the new target configuration.*<br><br>$00_B$ $f_{ASCLINS}$ clock is stopped<br>$01_B$ $f_{ASCLINSI}$ is used as clock $f_{ASCLINS}$<br>$10_B$ $f_{OSC0}$ is used as clock $f_{ASCLINS}$<br>$11_B$ Reserved, do not use this combination |
| **ERAYPERON** | 25 | rw | **Power Safe SwitchOff for ERAY Clock**<br>This bit is used to control the ERAY peripheral clock $f_{ERAY}$ for power saving purposes if the logic is not used by the application.<br>$0_B$ $f_{ERAY}$ is stopped<br>$1_B$ $f_{ERAY} = f_{source1} / 2$ |
| **HSPDMPERON** | 26 | rw | **Power Safe SwitchOff for HSPDM Clocks**<br>This bit is used to control the HSPDM peripheral clocks $f_{HSPDM\_320}$ and $f_{HSPDM\_160}$ for power saving purposes if the logic is not used by the application.<br>$0_B$ $f_{HSPDM\_320}$ is stopped; $f_{HSPDM\_160}$ is stopped<br>$1_B$ $f_{HSPDM\_320} = f_{src1}$; $f_{HSPDM\_160} = f_{source1}$ |
| **LCK** | 31 | rh | **Lock Status**<br>This bit indicates if the register can be updated with a new value or if the register is locked and a write action from the bus side has no effect.<br><br>*Note: The lock bit is set when at least one bit field is changed, and released when this change is executed.*<br><br>$0_B$ The register is unlocked and can be updated<br>$1_B$ The register is locked and can not be updated |
| **0** | 7:4, 23:14, 30:27 | rw | **Reserved**<br>Should be written with 0. |
| **1** | 24 | rw | **Reserved**<br>Can be written either 0/1, not connected to HW. |

## 9.4 Connectivity

**Table 53 Connections of SCU**

| Interface Signals | connects | | Description |
|---|---|---|---|
| SCU:CBS_ENDINIT_DIS | from | CBS:ocds_oc(3) | Watchdog ENDINIT disable from Cerberus |
| SCU:CBS_WDT_SUSP | from | CBS:ocds_wdtsus | Watchdog suspend from Cerberus |
| SCU:EMGSTOP_PORT_A | from | SMU:FSPSCU | Emergency stop Port Pin A input request |

**Table 53 Connections of SCU** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| SCU:EMGSTOP_PORT_B | from | P21.2:IN | Emergency stop Port Pin B input request |
| SCU:ESR0_PORT_IN | from | TC35x:ESR0 | ESR0 Port Pin input - can be used to trigger a reset or an NMI |
| SCU:ESR1_PORT_IN | from | TC35x:ESR1 | ESR1 Port Pin input - can be used to trigger a reset or an NMI |
| SCU:E_IOUT(0) | to | RIF0:RAMP1C | ERU IOUTn output (MSB is IOUT7 and LSB is IOUT0) |
| | | EVADC:G0REQTRH | |
| SCU:E_IOUT(1) | to | RIF1:RAMP1C | ERU IOUTn output (MSB is IOUT7 and LSB is IOUT0) |
| | | EVADC:G1REQTRH | |
| SCU:E_IOUT(3:2) | to | CAN0:ttc_ectt(4:3) | ERU IOUTn output (MSB is IOUT7 and LSB is IOUT0) |
| SCU:E_IOUT(4) | to | CAN0:ttc_ltrc_trig(4) | ERU IOUTn output (MSB is IOUT7 and LSB is IOUT0) |
| SCU:E_PDOUT(0) | to | CCU60:CTRAPD | ERU PDOUTn output (MSB is PDOUT7 and LSB is PDOUT0) |
| | | CCU60:T12HRH | |
| | | EVADC:G0REQGTM | |
| SCU:E_PDOUT(1) | to | CCU61:CTRAPD | ERU PDOUTn output (MSB is PDOUT7 and LSB is PDOUT0) |
| | | CCU61:T12HRH | |
| | | EVADC:G1REQGTM | |
| SCU:E_PDOUT(3:0) | to | ERAY0:STPWT(3:0) | ERU PDOUTn output (MSB is PDOUT7 and LSB is PDOUT0) |
| SCU:E_PDOUT(4) | to | CCU60:CC62IND | ERU PDOUTn output (MSB is PDOUT7 and LSB is PDOUT0) |
| | | GPT120:T3INC | |
| SCU:E_PDOUT(5) | to | CCU61:CC62IND | ERU PDOUTn output (MSB is PDOUT7 and LSB is PDOUT0) |
| SCU:E_PDOUT(6) | to | GPT120:CAPINB | ERU PDOUTn output (MSB is PDOUT7 and LSB is PDOUT0) |
| SCU:E_REQ0(0) | from | P15.4:IN | ERU Channel 0 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ0(1) | from | CCU60:COUT60 | ERU Channel 0 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ0(2) | from | P10.7:IN | ERU Channel 0 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ1(0) | from | P14.3:IN | ERU Channel 1 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ1(1) | from | CCU61:COUT60 | ERU Channel 1 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ1(2) | from | P10.8:IN | ERU Channel 1 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ1(3) | from | STM0:STMIR(0) | ERU Channel 1 input X; x=0-5, where 0 is input A and 5 is input F. |

**Table 53 Connections of SCU** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| SCU:E_REQ2(0) | from | P10.2:IN | ERU Channel 2 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ2(1) | from | P02.1:IN | ERU Channel 2 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ2(2) | from | P00.4:IN | ERU Channel 2 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ2(3) | from | ERAY0:MT | ERU Channel 2 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ3(0) | from | P10.3:IN | ERU Channel 3 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ3(1) | from | P14.1:IN | ERU Channel 3 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ3(2) | from | P02.0:IN | ERU Channel 3 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ3(3) | from | STM1:STMIR(0) | ERU Channel 3 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ4(0) | from | P33.7:IN | ERU Channel 4 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ4(2) | from | GPT120:T3OUT | ERU Channel 4 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ4(3) | from | P15.5:IN | ERU Channel 4 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ5(0) | from | P15.8:IN | ERU Channel 5 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ5(2) | from | GPT120:T6OUT | ERU Channel 5 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ5(3) | from | STM2:STMIR(0) | ERU Channel 5 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ6(0) | from | P20.0:IN | ERU Channel 6 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ6(1) | from | TC35x:ESR0 | ERU Channel 6 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ6(3) | from | P11.10:IN | ERU Channel 6 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ7(0) | from | P20.9:IN | ERU Channel 7 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ7(1) | from | TC35x:ESR1 | ERU Channel 7 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:E_REQ7(2) | from | P15.1:IN | ERU Channel 7 input X; x=0-5, where 0 is input A and 5 is input F. |
| SCU:RST_REQ_STM(10) | from | HSM:SYSRST | Reset request from STMn (MSB is STM5 and LSB is STM0) |

**Table 53 Connections of SCU** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| SCU:RST_REQ_STM(11) | from | HSM:APPRST | Reset request from STMn (MSB is STM5 and LSB is STM0) |
| SCU:SMU_EMGSTP_REQ | from | SMU:EMERGENCYSTOPREQ | Emergency stop request from SMU |
| SCU:SMU_TRAP_REQ | from | SMU:NMIREQ | TRAP request from the SMU |
| SCU:TRAP_CPU(0) | to | cpu_pfi_pfrwb_0:tc162p_nmi_trap | TRAP output to CPUn (MSB is CPU5 and LSB is CPU0) |
| SCU:TRAP_CPU(1) | to | cpu_pfi_pfrwb_1:tc162p_nmi_trap | TRAP output to CPUn (MSB is CPU5 and LSB is CPU0) |
| SCU:TRAP_CPU(2) | to | cpu_2:tc162p_nmi_trap | TRAP output to CPUn (MSB is CPU5 and LSB is CPU0) |
| SCU:ERU_INT(3:0) | to | INT:scu.ERU_INT(3:0) | SCU ERU Service Request x |

## 9.5 Revision History

The following table contains the revision history of the SCU that is relevant for the TC35x device. For a complete revision history please address the family user manual.

This section only includes entries, respective to the Clock System, that are related to register updates. For a complete Clock System revision history please address the family user manual.

**Table 54 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V2.1.21** | | |
| | Revision History entries up to V2.1.20 removed. | |
| **Page 1** | Updated MISR signature values for LBISTCTRL3, for Config A and Config B. | |
| **Page 26** | Connectivity information updated. | |
| **V2.1.22** | | |
| | Revision History entries up to V2.1.22 removed. | |
| **Page 17** | Cold PORST reset value changed from 1xx10000 to 0xx10000 in RSTSTAT register. | |
| **Page 17** | Additional cold_power_on_reset value "LVD Reset" added to RSTSTAT register. | |
| **Page 11** | LBISTCTRL0: System Reset value set to "Internal". Added note to CFS Value in reset table: "Value installed after System and Power-On Reset." | |
| **Page 11** | LCLCON0 and LCLCON1: Cold PORST reset table values updated/corrected. | |
| **V2.1.23** | | |
| | Revision History entries up to V2.1.21 removed. | |
| | Typo "Value" corrected in Revision History V2.1.22. | |
| **Page 1** | LBISTCTRL register configuration corrected. | |
| **Page 3**, **Page 11** | LBISTCTRL0 removed from specific registers, see Family Spec instead. | |

**Table 54 Revision History** (cont'd)

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V2.1.24** | | |
| **Page 11** | Updated Cold PowerOn Reset Value of LCLCONx. | |
| **V2.1.25** | | |
| - | No functional changes. | |
| **V2.1.26** | | |
| - | No functional changes. | |
| **V2.1.27** | | |
| | No functional changes. | |

# 10 Clocking System

Device specific information about the clocking system is contained in the SCU chapter as both modules share a common bus interface.

# 11 Power Management System (PMS)

This chapter describes the Power Management System (PMS) Module of the TC35x.

## 11.1 TC35x Specific IP Configuration

**Table 55 TC35x specific configuration of PMS**

| Parameter | PMS |
|---|---|
| **CFS value for the PMSWCR4 register** | $02000020_H$ |

## 11.2 TC35x Specific Register Set

The PMS related SCU registers are specified in the SCU section of this appendix.

**Table 56 Register Address Space - PMS**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| (PMS) | $F0240000_H$ | $F0241FFF_H$ | |
| PMS | $F0248000_H$ | $F02481FF_H$ | FPI slave interface |

**Table 57 Register Overview - PMS (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| PMS_ID | Identification Register | $0008_H$ | U,SV | BE | Application Reset | See Family Spec |
| PMS_EVRSTAT | EVR Status Register | $002C_H$ | U,SV | BE | See Family Spec | See Family Spec |
| PMS_EVRADCSTAT | EVR Primary ADC Status Register | $0034_H$ | U,SV | BE | LVD Reset | See Family Spec |
| PMS_EVRRSTCON | EVR Reset Control Register | $003C_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRRSTSTAT | EVR Reset Status Register | $0044_H$ | U,SV | BE | See Family Spec | See Family Spec |
| PMS_EVRTRIM | EVR Trim Control Register | $004C_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRTRIMSTAT | EVR Trim Status Register | $0050_H$ | U,SV | BE | See Family Spec | See Family Spec |
| PMS_EVRMONSTAT1 | EVR Secondary ADC Status Register 1 | $0060_H$ | U,SV | BE | See Family Spec | See Family Spec |
| PMS_EVRMONSTAT2 | EVR Secondary ADC Status Register 2 | $0064_H$ | U,SV | BE | See Family Spec | See Family Spec |
| PMS_EVRMONCTRL | EVR Secondary Monitor Control Register | $0068_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRMONFILT | EVR Secondary Monitor Filter Register | $0070_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |

**Table 57 Register Overview - PMS (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| PMS_PMSIEN | PMS Interrupt Enable Register | $0074_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRUVMON | EVR Secondary Under-voltage Monitor Register | $0078_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVROVMON | EVR Secondary Over-voltage Monitor Register | $007C_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRUVMON2 | EVR Secondary Under-voltage Monitor Register 2 | $0080_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVROVMON2 | EVR Secondary Over-voltage Monitor Register 2 | $0084_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_HSMUVMON | EVR Primary HSM Under-voltage Monitor Register | $0088_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_HSMOVMON | EVR Primary HSM Over-voltage Monitor Register | $008C_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVR33CON | EVR33 Control Register | $0090_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVROSCCTRL | EVR Oscillator Control Register | $00A0_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_PMSWCR0 | Standby and Wake-up Control Register 0 | $00B4_H$ | U,SV | SV,SE,P | LVD Reset | See Family Spec |
| PMS_PMSWCR2 | Standby and Wake-up Control Register 2 | $00B8_H$ | U,SV | SV,SE,P | LVD Reset | See Family Spec |
| PMS_PMSWCR3 | Standby and Wake-up Control Register 3 | $00C0_H$ | U,SV | SV,SE,P | LVD Reset | See Family Spec |
| PMS_PMSWCR4 | Standby and Wake-up Control Register 4 | $00C4_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_PMSWCR5 | Standby and Wake-up Control Register 5 | $00C8_H$ | U,SV | SV,SE,P | LVD Reset | See Family Spec |

**Table 57 Register Overview - PMS (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| PMS_PMSWSTAT | Standby and Wake-up Status Register | $00D4_H$ | U,SV | BE | LVD Reset | See Family Spec |
| PMS_PMSWSTAT2 | Standby and Wake-up Status Register 2 | $00D8_H$ | U,SV | BE | LVD Reset | See Family Spec |
| PMS_PMSWUTCNT | Standby WUT Counter Register | $00DC_H$ | U,SV | BE | LVD Reset | See Family Spec |
| PMS_PMSWSTATCLR | Standby and Wake-up Status Clear Register | $00E8_H$ | U,SV | SV,SE,P | LVD Reset | See Family Spec |
| PMS_EVRSDSTAT0 | EVR SD Status Register 0 | $00FC_H$ | U,SV | BE | See Family Spec | See Family Spec |
| PMS_EVRSDCTRL0 | EVRC SD Control Register 0 | $0108_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCTRL1 | EVRC SD Control Register 1 | $010C_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCTRL2 | EVRC SD Control Register 2 | $0110_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCTRL3 | EVRC SD Control Register 3 | $0114_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCTRL4 | EVRC SD Control Register 4 | $0118_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCTRL5 | EVRC SD Control Register 5 | $011C_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCTRL6 | EVRC SD Control Register 6 | $0120_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCTRL7 | EVRC SD Control Register 7 | $0124_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCTRL8 | EVRC SD Control Register 8 | $0128_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |

**Table 57 Register Overview - PMS (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| PMS_EVRSDCTRL9 | EVRC SD Control Register 9 | $012C_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCTRL10 | EVRC SD Control Register 10 | $0130_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCTRL11 | EVRC SD Control Register 11 | $0134_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCOEFF0 | EVRC SD Coefficient Register 0 | $0148_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCOEFF1 | EVRC SD Coefficient Register 1 | $014C_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCOEFF2 | EVRC SD Coefficient Register 2 | $0150_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCOEFF3 | EVRC SD Coefficient Register 3 | $0154_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCOEFF4 | EVRC SD Coefficient Register 4 | $0158_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCOEFF5 | EVRC SD Coefficient Register 5 | $015C_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCOEFF6 | EVRC SD Coefficient Register 6 | $0160_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCOEFF7 | EVRC SD Coefficient Register 7 | $0164_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCOEFF8 | EVRC SD Coefficient Register 8 | $0168_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_EVRSDCOEFF9 | EVRC SD Coefficient Register 9 | $016C_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_AG2i_STDBY (i=0-1) | Alarm Status Register | $0188_H$+i*4 | U,SV | SV,SE,P | LVD Reset | See Family Spec |

**Table 57 Register Overview - PMS (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| PMS_MONBISTSTAT | SMU_stdby BIST Status Register | $0190_H$ | U,SV | BE | See Family Spec | See Family Spec |
| PMS_MONBISTCTRL | SMU_stdby BIST Control Register | $0198_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_CMD_STDBY | SMU_stdby Command Register | $019C_H$ | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_AG2iFSP_STDBY (i=0-1) | SMU_stdby FSP Configuration Register | $01A4_H$+i*4 | U,SV | SV,SE,P | See Family Spec | See Family Spec |
| PMS_DTSSTAT | Die Temperature Sensor Status Register | $01C0_H$ | U,SV | BE | See Family Spec | See Family Spec |
| PMS_DTSLIM | Die Temperature Sensor Limit Register | $01C8_H$ | U,SV | U,SV,P | See Family Spec | See Family Spec |
| PMS_OTSS | OCDS Trigger Set Select Register | $01E0_H$ | U,SV | U,SV,P | See Family Spec | See Family Spec |
| PMS_OTSC0 | OCDS Trigger Set Control 0 Register | $01E4_H$ | U,SV | U,SV,P | See Family Spec | See Family Spec |
| PMS_OTSC1 | OCDS Trigger Set Control 1 Register | $01E8_H$ | U,SV | U,SV,P | See Family Spec | See Family Spec |
| PMS_ACCEN1 | Access Enable Register 1 | $01F8_H$ | U,SV | SV,SE,32 | Application Reset | See Family Spec |
| PMS_ACCEN0 | Access Enable Register 0 | $01FC_H$ | U,SV | SV,SE,32 | Application Reset | See Family Spec |

## 11.3 TC35x Specific Registers

No deviations from the Family Spec

## 11.4 Connectivity

**Table 58 Connections of PMS**

| Interface Signals | connects | | Description |
|---|---|---|---|
| PMS:DCDCSYNCO | to | P32.4:HWOUT(0) | DC-DC synchronization output |
| | | P32.2:ALT(6) | |
| | | P32.4:ALT(2) | |
| PMS:ESR0PORST | to | TC35x:ESR0 | ESR0 control output during PORST activation |
| PMS:ESR0WKP | from | TC35x:ESR0 | ESR0 pin input |
| PMS:ESR1WKP | from | TC35x:ESR1 | ESR1 pin input |
| PMS:HWCFG1IN | from | TC35x:P14.5 | HWCFG1 pin input |
| PMS:HWCFG2IN | from | TC35x:P14.2 | HWCFG2 pin input |
| PMS:HWCFG4IN | from | TC35x:P10.5 | HWCFG4 pin input |
| PMS:HWCFG5IN | from | TC35x:P10.6 | HWCFG5 pin input |
| PMS:HWCFG6IN | from | TC35x:P14.4 | HWCFG6 pin input |
| PMS:PINAWKP | from | TC35x:P14.1 | PINA ( P14.1) pin input |
| PMS:PINBWKP | from | TC35x:P33.12 | PINB (P33.12) pin input |
| PMS:PORSTIN | from | TC35x:PORST | PORST pin input |
| PMS:PORSTOUT | to | TC35x:PORST | PORST pin output |
| PMS:TESTMODEIN | from | TC35x:P20.2 | TESTMODE pin input |
| PMS:VDDMLVL | to | converter_0:converter_low_supp | VDDM monitor signal to Converter |
| PMS:VGATE1N | to | TC35x:CTRL1V3N | DCDC N ch. MOSFET gate driver output |
| | | TC35x:CTRL1V3N | |
| PMS:VGATE1P | to | TC35x:CTRL1V3P | DCDC P ch. MOSFET gate driver output |
| | | TC35x:CTRL1V3P | |

## 11.5 Revision History

**Table 59 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V2.2.28** | | |
| – | No changes. | |
| **V2.2.29** | | |
| – | No functional changes. | |
| **V2.2.30** | | |
| **Page 2** | Register “PMS_EVR33CON” now visible to the customer. | |
| **V2.2.31** | | |
| – | No functional changes. | |
| **V2.2.32** | | |
| – | No functional changes. | |

**Table 59 Revision History** (cont'd)

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V2.2.33** | | |
| – | No functional changes. | |
| **V2.2.34** | | |
| – | No functional changes. | |

# 12 Power Management System for Low-End (PMSLE)

This device doesn’t contain a PMSLE module.

# 13 Memory Test Unit (MTU)

For the generic description of the Memory Test Unit (MTU) and the SRAM Support Hardware (SSH), please refer to the platform chapter.

## 13.1 TC35x Specific IP Configuration

There is no device specific IP configuration. MTU+SSH is generic across all derivates in the platforms. Only the SSH instances vary.

## 13.2 Handling of Large DSPR SRAMs

On this device, the CPU0 and CPU1 have large DSPR SRAMs. Therefore, there are two SSHs to support these SRAMs. These are named as CPUxDMEM and CPUxDMEM1 (x=0,1).

Logically, these 2 SSHs behave as separate SSHs, with their own MEMTEST_EN bits, Alarms to the SMU, error status flags etc. However special handling is required to enter test mode for these SSHs.

When running a Non-Destructive-test, only one of the DMEM SSHs shall be enabled at a time. This is because the ECC encoder and decoder are shared between the two SSHs.

However, as long as destructive tests or SSH register accesses are to be performed, then both SSHs can be enabled together.

Please note that when one of the DMEM SSH is enabled, the complete DSPR+DCACHE is unavailable for functional access. This is due to internal interleaving of the logical address space.

Although the DCACHE is also split equally between the two SRAMs, the entire cache has to be mapped at once to the system address map. Hence only a single bit is provided in the MTU_MEMMAP register for the entire DMEM.

The partial-erase (i.e. erasing the Cache area; refer the family spec) is triggered separately when each MEMTEST_EN bit is set.

The MEMMAP bit triggers the partial erase in both SSHs at the same time.

If both the SSHs are enabled / disabled at the same time, the software should check either the bits in the MTU_MEMSTAT register, or ensure that the EN bits for both the SSHs are set / cleared in the MEMTEST registers to ensure that the partial-erase has completed.

## 13.3 TC35x Specific Register Set

### Register Address Space Table

**Table 60 Register Address Space - MTU**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| MTU | $F0060000_H$ | $F006FFFF_H$ | FPI slave interface |

### Register Overview Table

**Table 61 Register Overview - MTU (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| MTU_CLC | Clock Control Register | $0000_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| MTU_ID | Identification Register | $0008_H$ | U,SV | BE | Application Reset | See Family Spec |
| MTU_MEMTESTi (i=0-2) | Memory MBIST Enable Register i | $0010_H$+i* 4 | U,SV | SV,SE,P | Application Reset | 3 |
| MTU_MEMMAP | Memory Mapping Enable Register | $001C_H$ | U,SV | SV,SE,P | Application Reset | 9 |
| MTU_MEMSTATi (i=0-2) | Memory Status Register i | $0038_H$+i* 4 | U,SV | BE | Application Reset | 12 |
| MTU_MEMDONEi (i=0-2) | Memory Test Done Status Register i | $0050_H$+i* 4 | U,SV | BE | Application Reset | 16 |
| MTU_MEMFDAi (i=0-2) | Memory Test FDA Status Register i | $0060_H$+i* 4 | U,SV | BE | Application Reset | 21 |
| MTU_ACCEN1 | Access Enable Register 1 | $00F8_H$ | U,SV | BE | Application Reset | See Family Spec |
| MTU_ACCEN0 | Access Enable Register 0 | $00FC_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| MTU_MCi_CONFIG0 (i=0-95) | Configuration Registers | $1000_H$+i* $100_H$ | U,SV,16 | U,SV,P,16 | Application Reset | See Family Spec |
| MTU_MCi_CONFIG1 (i=0-95) | Configuration Register 1 | $1002_H$+i* $100_H$ | U,SV,16 | U,SV,P,16 | Application Reset | See Family Spec |

**Table 61 Register Overview - MTU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| MTU_MCi_MCONTROL (i=0-95) | MBIST Control Register | $1004_H$+i* $100_H$ | U,SV,16 | SV,SE,P,16 | Application Reset | See Family Spec |
| MTU_MCi_MSTATUS (i=0-95) | Status Register | $1006_H$+i* $100_H$ | U,SV,16 | BE | Application Reset | See Family Spec |
| MTU_MCi_RANGE (i=0-95) | Range Register, single address mode | $1008_H$+i* $100_H$ | U,SV,16 | U,SV,P,16 | Application Reset | See Family Spec |
| MTU_MCi_REVID (i=0-95) | Revision ID Register | $100C_H$+i* $100_H$ | U,SV,16 | BE | Application Reset | See Family Spec |
| MTU_MCi_ECCS (i=0-95) | ECC Safety Register | $100E_H$+i* $100_H$ | U,SV,16 | SV,SE,P,16 | Application Reset | See Family Spec |
| MTU_MCi_ECCD (i=0-95) | Memory ECC Detection Register | $1010_H$+i* $100_H$ | U,SV,16 | SV,P,16 | See Family Spec | See Family Spec |
| MTU_MCi_ETRRx (i=0-95;x=0-4) | Error Tracking Register x | $1012_H$+i* $100_H$+x* 2 | U,SV,16 | BE | PowerOn Reset | See Family Spec |
| MTU_MCi_RDBFLy (i=0-95;y=0-66) | Read Data and Bit Flip Register y | $1060_H$+i* $100_H$+y* 2 | U,SV,16 | U,SV,P,16 | Application Reset | See Family Spec |
| MTU_MCi_ALMSRCS (i=0-95) | Alarm Sources Configuration Register | $10EE_H$+i* $100_H$ | U,SV,16 | SV,SE,P,16 | Application Reset | See Family Spec |
| MTU_MCi_FAULTSTS (i=0-95) | SSH Safety Faults Status Register | $10F0_H$+i* $100_H$ | U,SV,16 | SV,SE,P,16 | PowerOn Reset | See Family Spec |
| MTU_MCi_ERRINFOx (i=0-95;x=0-4) | Error Information Register x | $10F2_H$+i* $100_H$+x* 2 | U,SV,16 | BE | PowerOn Reset | See Family Spec |

## 13.4 TC35x Specific Registers

### 13.4.1 MEMTEST Implementation

**Memory MBIST Enable Register i**

The memory test register MEMTEST holds CPU configurable select bits for the various SSH instances. See the product specific appendix for mapping of memory controller numbers.

### MTU_MEMTESTi (i=0)

**Memory MBIST Enable Register i** **($0010_H$+i*4)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| LMU10_EN | LMU00_EN | RES29 | RES28 | RES27 | RES26 | RES25 | RES24 | RES23 | RES22 | RES21 | RES20 | RES19 | RES18 | RES17 | RES16 |
| rwh | rwh | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RES15 | CPU2_DLMU_EN | CPU2_PTAG_EN | CPU2_PMEM_EN | CPU2_DTAG_EN | CPU2_DMEM_EN | CPU1_DLMU_STBY_EN | CPU1_PTAG_EN | CPU1_PMEM_EN | CPU1_DTAG_EN | CPU1_DMEM_EN | CPU0_DLMU_STBY_EN | CPU0_PTAG_EN | CPU0_PMEM_EN | CPU0_DTAG_EN | CPU0_DMEM_EN |
| r | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CPU0_DMEM_EN** | 0 | rwh | **CPU0 DMEM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **RESx (x=15-29)** | x | r | **Reserved**<br>Reserved. Shall be written with zero. |
| **CPU0_DTAG_EN** | 1 | rwh | **CPU0 DTAG SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **CPU0_PMEM_EN** | 2 | rwh | **CPU0 PMEM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **CPU0_PTAG_EN** | 3 | rwh | **CPU0 PTAG SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **CPU0_DLMU_STBY_EN** | 4 | rwh | **CPU0 STANDBY DLMU SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **CPU1_DMEM_EN** | 5 | rwh | **CPU1 DMEM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **CPU1_DTAG_EN** | 6 | rwh | **CPU1 DTAG SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **CPU1_PMEM_EN** | 7 | rwh | **CPU1 PMEM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **CPU1_PTAG_EN** | 8 | rwh | **CPU1 PTAG SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CPU1_DLMU_STBY_EN** | 9 | rwh | **CPU1 STANDBY DLMU SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **CPU2_DMEM_EN** | 10 | rwh | **CPU2 DMEM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **CPU2_DTAG_EN** | 11 | rwh | **CPU2 DTAG SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **CPU2_PMEM_EN** | 12 | rwh | **CPU2 PMEM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **CPU2_PTAG_EN** | 13 | rwh | **CPU2 PTAG SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **CPU2_DLMU_EN** | 14 | rwh | **CPU2 DLMU memory SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **LMU00_EN** | 30 | rwh | **LMU00 SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **LMU10_EN** | 31 | rwh | **LMU10 SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |

**MTU_MEMTESTi (i=1)**

**Memory MBIST Enable Register i** **($0010_H$+i*4)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **MCAN20_EN** | **MCAN10_EN** | **RES29** | **RES28** | **RES27** | **RES26** | **RES25** | **RES24** | **RES23** | **RES22** | **RES21** | **SPU_CONFIG1_EN** | **SPU_CONFIG0_EN** | **SPU_BUFFER1_EN** | **SPU_BUFFER0_EN** | **EMEM_XTM_EN** |
| rwh | rwh | r | r | r | r | r | r | r | r | r | rwh | rwh | rwh | rwh | rwh |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **RES15** | **RES14** | **EMEM1_EN** | **EMEM0_EN** | **MCDS_EN** | **RES10** | **SADMA_EN** | **R8** | **RES7** | **RES6** | **RES5** | **RES4** | **CPU1_DMEM1_EN** | **CPU0_DMEM1_EN** | **RES1** | **RES0** |
| r | r | rwh | rwh | rwh | r | rwh | rwh | r | r | r | r | rwh | rwh | r | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **RESx (x=0-1,4-7,10,14-15,21-29)** | x | r | **Reserved**<br>Reserved. Shall be written with zero. |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CPU0_DMEM1_EN** | 2 | rwh | **CPU0 DMEM1 SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled [1]. |
| **CPU1_DMEM1_EN** | 3 | rwh | **CPU1 DMEM1 SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled [2]. |
| **R8** | 8 | rwh | **Reserved - Res**<br>Reserved. Not used in this product.<br>Shall be written with zero. |
| **SADMA_EN** | 9 | rwh | **Safety DMA SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **MCDS_EN** | 11 | rwh | **MCDS memory SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **EMEM0_EN** | 12 | rwh | **EMEM0 SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **EMEM1_EN** | 13 | rwh | **EMEM1 SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **EMEM_XTM_EN** | 16 | rwh | **EMEM XTM memory SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **SPU_BUFFER0_EN** | 17 | rwh | **SPU BUFFER0 SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **SPU_BUFFER1_EN** | 18 | rwh | **SPU BUFFER1 SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **SPU_CONFIG0_EN** | 19 | rwh | **SPU CONFIG0 memory SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **SPU_CONFIG1_EN** | 20 | rwh | **SPU CONFIG1 memory SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **MCAN10_EN** | 30 | rwh | **MCAN10 memory SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **MCAN20_EN** | 31 | rwh | **MCAN20 memory SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |

1) Please refer to separate section related to handling of the large DMEM on this device.

2) Please refer to separate section related to handling of the large DMEM on this device.

### MTU_MEMTESTi (i=2)

**Memory MBIST Enable Register i** **($0010_H$+i*4)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SPU_FFT31_RAM_EN | SPU_FFT30_RAM_EN | SPU_FFT21_RAM_EN | SPU_FFT20_RAM_EN | SPU_FFT11_RAM_EN | SPU_FFT10_RAM_EN | SPU_FFT01_RAM_EN | SPU_FFT00_RAM_EN | RES23 | RES22 | HSPDM_RAM_EN | RES20 | GIGETH_TX_EN | GIGETH_RX_EN | RES17 | RES16 |
| rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | r | r | rwh | r | rwh | rwh | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RES15 | SCR_RAMINT_EN | SCR_XRAM_EN | R12 | R11 | R10 | R9 | R8 | RES7 | ERAY_MBF0_EN | RES5 | ERAY_TBF_IBF0_EN | RES3 | ERAY_OBF0_EN | RES1 | RES0 |
| r | rwh | rwh | rwh | rwh | rwh | rwh | rwh | r | rwh | r | rwh | r | rwh | r | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **RESx (x=0-1,3,5,7,15-17,20,22-23)** | x | r | **Reserved**<br>Reserved. Shall be written with zero. |
| **ERAY_OBF0_EN** | 2 | rwh | **ERAY OBF0 SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **ERAY_TBF_IBF0_EN** | 4 | rwh | **ERAY TBF IBF0 memory SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **ERAY_MBF0_EN** | 6 | rwh | **ERAY MBF0 memory SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **R8** | 8 | rwh | **Reserved - Res**<br>Reserved. Not used in this product.<br>Shall be written with zero. |
| **R9** | 9 | rwh | **Reserved - Res**<br>Reserved. Not used in this product.<br>Shall be written with zero. |
| **R10** | 10 | rwh | **Reserved - Res**<br>Reserved. Not used in this product.<br>Shall be written with zero. |
| **R11** | 11 | rwh | **Reserved - Res**<br>Reserved. Not used in this product.<br>Shall be written with zero. |
| **R12** | 12 | rwh | **Reserved - Res**<br>Reserved. Not used in this product.<br>Shall be written with zero. |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SCR_XRAM_EN** | 13 | rwh | **SCR XRAM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **SCR_RAMINT_EN** | 14 | rwh | **SCR Internal RAM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **GIGETH_RX_EN** | 18 | rwh | **Gigabit Ethernet RX SSH instance Enable**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **GIGETH_TX_EN** | 19 | rwh | **Gigabit Ethernet TX SSH instance Enable**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **HSPDM_RAM_EN** | 21 | rwh | **HDSPDM RAM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **SPU_FFT00_RAM_EN** | 24 | rwh | **SPU FFT00 RAM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **SPU_FFT01_RAM_EN** | 25 | rwh | **SPU FFT01 RAM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **SPU_FFT10_RAM_EN** | 26 | rwh | **SPU FFT10 RAM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **SPU_FFT11_RAM_EN** | 27 | rwh | **SPU FFT11 RAM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **SPU_FFT20_RAM_EN** | 28 | rwh | **SPU FFT20 RAM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **SPU_FFT21_RAM_EN** | 29 | rwh | **SPU FFT21 RAM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **SPU_FFT30_RAM_EN** | 30 | rwh | **SPU FFT30 RAM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |
| **SPU_FFT31_RAM_EN** | 31 | rwh | **SPU FFT31 RAM SSH instance Enable**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled |

## 13.4.2 MEMMAP Implementation

The Memory Mapping Enable register MEMMAP has configurable control bits to select memory-mapped test mode for each CPU memory.

Cache and Scratchpad memories are physically implemented as a single RAM, but this register function assumes two separate logical RAM partitions. In this register additional bits CPUxDCMAP and CPUxPCMAP are defined. These control the Cache partitions of the RAMs for Data Side and Program side respectively. Since cache content and tags of a cache must be simultaneously switched from memory mapped to cache functional mode, the control bits are mirrored and only one bit is writeable for each cache. The bits corresponding to the tag memories of the same cache will always take the same value as that written to the main Cache Memory control bit. This linkage is product specific.

Please note that the MEMMAP register is used only to map the Cache/Tag memories to system address space. These bits have no effect on testing these memories itself. For system address range into which the memories will be mapped, please refer to the memory map chapter.

### Memory Mapping Enable Register

The Memory Mapping Enable register MEMMAP has configurable control bits to select memory-mapped test mode. See the Integration Section for mapping of memory controller numbers.

**MTU_MEMMAP**
**Memory Mapping Enable Register** **($001C_H$)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R29 | | | MEM28MAP | MEM27MAP | MEM26MAP | MEM25MAP | R24 | MEM23MAP | MEM22MAP | MEM21MAP | MEM20MAP | R19 | MEM18MAP | MEM17MAP | MEM16MAP |
| r | | | r | r | r | r | r | r | r | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MEM15MAP | R14 | CPU2_PTMAP | CPU2_PCMAP | CPU2_DTMAP | CPU2_DCMAP | R9 | CPU1_PTMAP | CPU1_PCMAP | CPU1_DTMAP | CPU1_DCMAP | R4 | CPU0_PTMAP | CPU0_PCMAP | CPU0_DTMAP | CPU0_DCMAP |
| r | r | rh | rwh | rh | rwh | r | rh | rwh | rh | rwh | r | rh | rwh | rh | rwh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CPU0_DCMAP** | 0 | rwh | **CPU0 DCache Mapping**<br>$0_B$ Normal cache function<br>$1_B$ Memory-mapped |
| **MEMxMAP (x=15-18,20-23,25-28)** | x | r | **MEMx Mapping Enable**<br>Reserved; Not used in this product.<br>Shall be written with zero. |
| **CPU0_DTMAP** | 1 | rh | **CPU0 DTAG Mapping**<br>Read only. Mirrors the state of CPU0_DCMAP.<br>CPU D-cache memories may only be mapped simultaneously.<br>$0_B$ Normal cache function<br>$1_B$ Memory-mapped |
| **CPU0_PCMAP** | 2 | rwh | **CPU0 PCACHE Mapping**<br>$0_B$ Normal cache function<br>$1_B$ Memory-mapped |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CPU0_PTMAP** | 3 | rh | **CPU0 PTAG Mapping**<br>Read only. Mirrors the state of CPU0_PCMAP.<br>CPU P-cache memories may only be mapped simultaneously.<br>$0_B$ Normal cache function<br>$1_B$ Memory-mapped |
| **R4** | 4 | r | **Reserved - Res**<br>Reserved. Not used in this product. |
| **CPU1_DCMAP** | 5 | rwh | **CPU1 DCache Mapping**<br>$0_B$ Normal cache function<br>$1_B$ Memory-mapped |
| **CPU1_DTMAP** | 6 | rh | **CPU1 DTAG Mapping**<br>Read only. Mirrors the state of CPU1_DCMAP.<br>CPU D-cache memories may only be mapped simultaneously.<br>$0_B$ Normal cache function<br>$1_B$ Memory-mapped |
| **CPU1_PCMAP** | 7 | rwh | **CPU1 PCACHE Mapping**<br>$0_B$ Normal cache function<br>$1_B$ Memory-mapped |
| **CPU1_PTMAP** | 8 | rh | **CPU1 PTAG Mapping**<br>Read only. Mirrors the state of CPU1_PCMAP.<br>CPU P-cache memories may only be mapped simultaneously.<br>$0_B$ Normal cache function<br>$1_B$ Memory-mapped |
| **R9** | 9 | r | **Reserved - Res**<br>Reserved. Not used in this product. |
| **CPU2_DCMAP** | 10 | rwh | **CPU2 DCache Mapping**<br>$0_B$ Normal cache function<br>$1_B$ Memory-mapped |
| **CPU2_DTMAP** | 11 | rh | **CPU2 DTAG Mapping**<br>Read only. Mirrors the state of CPU2_DCMAP.<br>CPU D-cache memories may only be mapped simultaneously.<br>$0_B$ Normal cache function<br>$1_B$ Memory-mapped |
| **CPU2_PCMAP** | 12 | rwh | **CPU2 PCACHE Mapping**<br>$0_B$ Normal cache function<br>$1_B$ Memory-mapped |
| **CPU2_PTMAP** | 13 | rh | **CPU2 PTAG Mapping**<br>Read only. Mirrors the state of CPU2_PCMAP.<br>CPU P-cache memories may only be mapped simultaneously.<br>$0_B$ Normal cache function<br>$1_B$ Memory-mapped |
| **R14** | 14 | r | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R19** | 19 | r | **Reserved - Res**<br>Reserved. Not used in this product. |

| Field | Bits | Type | Description |
|---|---|---|---|
| **R24** | 24 | r | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R29** | 31:29 | r | **Reserved - Res**<br>Reserved. Not used in this product. |

### 13.4.3 MEMSTAT Implementation

The Memory Status Registers MEMSTATx have an implemented bit for each security relevant RAM.

The Data- and Program- Cache and Scratchpad memories are physically implemented as a single RAM with a single MBIST. Hence CPUx_DMEM_AIU and CPUx_PMEM_AIU give the status of the partial initialization of the cache partitions for the Data and Program memories respectively.

#### Memory Status Register i

The memory status register MEMSTAT shows whether each SSH instance is currently executing an automatic initialization sequence.

**MTU_MEMSTATi (i=0)**
**Memory Status Register i** **($0038_H$+i*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R29 | | | RES28 | RES27 | RES26 | RES25 | R24 | RES23 | RES22 | RES21 | RES20 | R19 | RES18 | RES17 | RES16 |
| | r | | r | r | r | r | r | r | r | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RES15 | R14 | CPU2_PTAG_AIU | CPU2_PMEM_AIU | CPU2_DTAG_AIU | CPU2_DMEM_AIU | R9 | CPU1_PTAG_AIU | CPU1_PMEM_AIU | CPU1_DTAG_AIU | CPU1_DMEM_AIU | R4 | CPU0_PTAG_AIU | CPU0_PMEM_AIU | CPU0_DTAG_AIU | CPU0_DMEM_AIU |
| r | r | rh | rh | rh | rh | r | rh | rh | rh | rh | r | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CPU0_DMEM_AIU** | 0 | rh | **CPU0 DMEM Partial AutoInitialize of Cache Partition Underway**<br>This bit indicates whether an automatic data initialization has been triggered by a change of state of MEMTEST.MEMxEN or MEMxMAP but that the intialization sequence has not yet completed.<br>$0_B$ MBIST not running autoinitialize<br>$1_B$ MBIST running autoinitialize |
| **RESx (x=15-18,20-23,25-28)** | x | r | **Reserved**<br>Reserved. Not used in this product. |
| **CPU0_DTAG_AIU** | 1 | rh | **CPU0 DTAG MBIST AutoInitialize Underway**<br>This bit indicates whether an automatic data initialization has been triggered by a change of state of MEMTEST.MEMxEN or MEMxMAP but that the intialization sequence has not yet completed.<br>$0_B$ MBIST not running autoinitialize<br>$1_B$ MBIST running autoinitialize |
| **CPU0_PMEM_AIU** | 2 | rh | **CPU0 PMEM Partial AutoInitialize of Cache Partition Underway**<br>This bit indicates whether an automatic data initialization has been triggered by a change of state of MEMTEST.MEMxEN or MEMxMAP but that the intialization sequence has not yet completed.<br>$0_B$ MBIST not running autoinitialize<br>$1_B$ MBIST running autoinitialize |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CPU0_PTAG_AIU** | 3 | rh | **CPU0 PTAG MBIST AutoInitialize Underway**<br>This bit indicates whether an automatic data initialization has been triggered by a change of state of MEMTEST.MEMxEN or MEMxMAP but that the intialization sequence has not yet completed.<br>$0_B$ MBIST not running autoinitialize<br>$1_B$ MBIST running autoinitialize |
| **R4** | 4 | r | **Reserved - Res**<br>Reserved. Not used in this product. |
| **CPU1_DMEM_AIU** | 5 | rh | **CPU1 DMEM Partial AutoInitialize of Cache Partition Underway**<br>This bit indicates whether an automatic data initialization has been triggered by a change of state of MEMTEST.MEMxEN or MEMxMAP but that the intialization sequence has not yet completed.<br>$0_B$ MBIST not running autoinitialize<br>$1_B$ MBIST running autoinitialize |
| **CPU1_DTAG_AIU** | 6 | rh | **CPU1 DTAG MBIST AutoInitialize Underway**<br>This bit indicates whether an automatic data initialization has been triggered by a change of state of MEMTEST.MEMxEN or MEMxMAP but that the intialization sequence has not yet completed.<br>$0_B$ MBIST not running autoinitialize<br>$1_B$ MBIST running autoinitialize |
| **CPU1_PMEM_AIU** | 7 | rh | **CPU1 PMEM Partial AutoInitialize of Cache Partition Underway**<br>This bit indicates whether an automatic data initialization has been triggered by a change of state of MEMTEST.MEMxEN or MEMxMAP but that the intialization sequence has not yet completed.<br>$0_B$ MBIST not running autoinitialize<br>$1_B$ MBIST running autoinitialize |
| **CPU1_PTAG_AIU** | 8 | rh | **CPU1 PTAG MBIST AutoInitialize Underway**<br>This bit indicates whether an automatic data initialization has been triggered by a change of state of MEMTEST.MEMxEN or MEMxMAP but that the intialization sequence has not yet completed.<br>$0_B$ MBIST not running autoinitialize<br>$1_B$ MBIST running autoinitialize |
| **R9** | 9 | r | **Reserved - Res**<br>Reserved. Not used in this product. |
| **CPU2_DMEM_AIU** | 10 | rh | **CPU2 DMEM Partial AutoInitialize of Cache Partition Underway**<br>This bit indicates whether an automatic data initialization has been triggered by a change of state of MEMTEST.MEMxEN or MEMxMAP but that the intialization sequence has not yet completed.<br>$0_B$ MBIST not running autoinitialize<br>$1_B$ MBIST running autoinitialize |
| **CPU2_DTAG_AIU** | 11 | rh | **CPU2 DTAG MBIST AutoInitialize Underway**<br>This bit indicates whether an automatic data initialization has been triggered by a change of state of MEMTEST.MEMxEN or MEMxMAP but that the intialization sequence has not yet completed.<br>$0_B$ MBIST not running autoinitialize<br>$1_B$ MBIST running autoinitialize |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CPU2_PMEM_AIU** | 12 | rh | **CPU2 PMEM Partial AutoInitialize of Cache Partition Underway**<br>This bit indicates whether an automatic data initialization has been triggered by a change of state of MEMTEST.MEMxEN or MEMxMAP but that the intialization sequence has not yet completed.<br>$0_B$ MBIST not running autoinitialize<br>$1_B$ MBIST running autoinitialize |
| **CPU2_PTAG_AIU** | 13 | rh | **CPU2 PTAG MBIST AutoInitialize Underway**<br>This bit indicates whether an automatic data initialization has been triggered by a change of state of MEMTEST.MEMxEN or MEMxMAP but that the intialization sequence has not yet completed.<br>$0_B$ MBIST not running autoinitialize<br>$1_B$ MBIST running autoinitialize |
| **R14** | 14 | r | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R19** | 19 | r | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R24** | 24 | r | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R29** | 31:29 | r | **Reserved - Res**<br>Reserved. Not used in this product. |

**MTU_MEMSTATi (i=1)**

**Memory Status Register i** **($0038_H$+i*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R9 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R9 | | | | | | | R8 | R5 | | | R4 | CPU1_DMEM1_AIU | CPU0_DMEM1_AIU | R0 | |
| r | | | | | | | rh | r | | | r | rh | rh | r | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **R0** | 1:0 | r | **Reserved - Res**<br>Reserved. Not used in this product. |
| **CPU0_DMEM1_AIU** | 2 | rh | **CPU0 DMEM1 Partial AutoInitialize of Cache Partition Underway**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled [1]. |
| **CPU1_DMEM1_AIU** | 3 | rh | **CPU1 DMEM1 Partial AutoInitialize of Cache Partition Underway**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled [2]. |
| **R4** | 4 | r | **Reserved - Res**<br>Reserved. Not used in this product. |

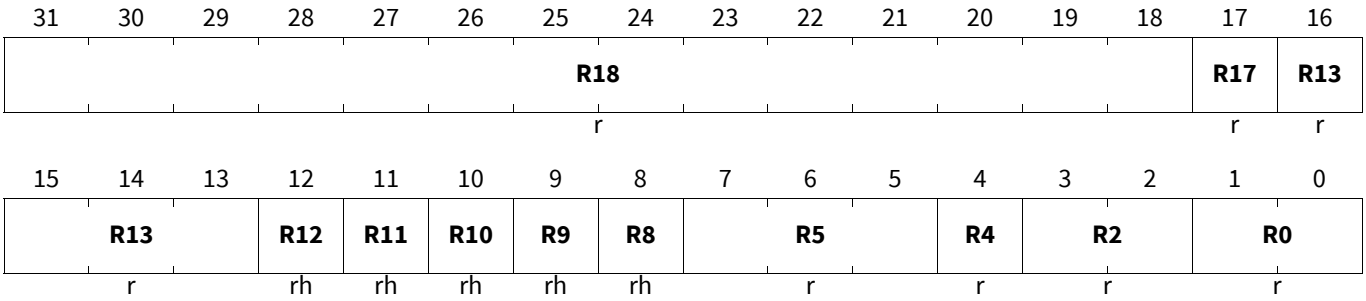
| Field | Bits | Type | Description |
|---|---|---|---|
| **R5** | 7:5 | r | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R8** | 8 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R9** | 31:9 | r | **Reserved - Res**<br>Reserved. Not used in this product. |

1) Please refer to separate section related to handling of the large DMEM on this device.

2) Please refer to separate section related to handling of the large DMEM on this device.

**MTU_MEMSTATi (i=2)**

**Memory Status Register i** **($0038_H$+i*4)** **Application Reset Value: 0000 $0000_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **R0** | 1:0 | r | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R2** | 3:2 | r | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R4** | 4 | r | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R5** | 7:5 | r | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R8** | 8 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R9** | 9 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R10** | 10 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R11** | 11 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R12** | 12 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R13** | 16:13 | r | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R17** | 17 | r | **Reserved - Res**<br>Reserved. Not used in this product. |

| Field | Bits | Type | Description |
|---|---|---|---|
| **R18** | 31:18 | r | **Reserved - Res**<br>Reserved. Not used in this product. |

## 13.4.4 MEMDONE Implementation

### Memory Test Done Status Register i

Each bit in one of the memory test done status registers MEMDONEx reflects the status of the MSTATUS.DONE bit in the corresponding SSH. See the implementation section for the implemented register bits.

**MTU_MEMDONEi (i=0)**
**Memory Test Done Status Register i** **($0050_H$+i*4)** **Application Reset Value: FFFF FFFF$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **LMU10_DONE** | **LMU00_DONE** | **RES29** | **RES28** | **RES27** | **RES26** | **RES25** | **RES24** | **RES23** | **RES22** | **RES21** | **RES20** | **RES19** | **RES18** | **RES17** | **RES16** |
| rh | rh | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **RES15** | **CPU2_DLMU_DONE** | **CPU2_PTAG_DONE** | **CPU2_PMEM_DONE** | **CPU2_DTAG_DONE** | **CPU2_DMEM_DONE** | **CPU1_DLMU_STBY_DON** | **CPU1_PTAG_DONE** | **CPU1_PMEM_DONE** | **CPU1_DTAG_DONE** | **CPU1_DMEM_DONE** | **CPU0_DLMU_STBY_DON** | **CPU0_PTAG_DONE** | **CPU0_PMEM_DONE** | **CPU0_DTAG_DONE** | **CPU0_DMEM_DONE** |
| r | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CPU0_DMEM_DONE** | 0 | rh | **CPU0 DMEM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **RESz (z=15-29)** | z | r | **Reserved**<br>Reserved. Not used in this product. |
| **CPU0_DTAG_DONE** | 1 | rh | **CPU0 DTAG Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **CPU0_PMEM_DONE** | 2 | rh | **CPU0 PMEM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **CPU0_PTAG_DONE** | 3 | rh | **CPU0 PTAG Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **CPU0_DLMU_STBY_DONE** | 4 | rh | **CPU0 STANDBY DLMU Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **CPU1_DMEM_DONE** | 5 | rh | **CPU1 DMEM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CPU1_DTAG_DONE** | 6 | rh | **CPU1 DTAG Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **CPU1_PMEM_DONE** | 7 | rh | **CPU1 PMEM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **CPU1_PTAG_DONE** | 8 | rh | **CPU1 PTAG Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **CPU1_DLMU_STBY_DONE** | 9 | rh | **CPU1 STANDBY DLMU Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **CPU2_DMEM_DONE** | 10 | rh | **CPU2 DMEM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **CPU2_DTAG_DONE** | 11 | rh | **CPU2 DTAG Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **CPU2_PMEM_DONE** | 12 | rh | **CPU2 PMEM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **CPU2_PTAG_DONE** | 13 | rh | **CPU2 PTAG Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **CPU2_DLMU_DONE** | 14 | rh | **CPU2 DLMU memory Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **LMU00_DONE** | 30 | rh | **LMU00 Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **LMU10_DONE** | 31 | rh | **LMU10 Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |

**MTU_MEMDONEi (i=1)**

**Memory Test Done Status Register i** **($0050_H$+i*4)** **Application Reset Value: FFFF FFFF$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MCAN20_DONE | MCAN10_DONE | RES29 | RES28 | RES27 | RES26 | RES25 | RES24 | RES23 | RES22 | RES21 | SPU_CONFIG1_DONE | SPU_CONFIG0_DONE | SPU_BUFFER1_DONE | SPU_BUFFER0_DONE | EMEM_XTM_DONE |
| rh | rh | r | r | r | r | r | r | r | r | r | rh | rh | rh | rh | rh |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RES15 | RES14 | EMEM1_DONE | EMEM0_DONE | MCDS_DONE | RES10 | SADMA_DONE | R8 | RES7 | RES6 | RES5 | RES4 | CPU1_DMEM1_DONE | CPU0_DMEM1_DONE | RES1 | RES0 |
| r | r | rh | rh | rh | r | rh | rh | r | r | r | r | rh | rh | r | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **RESz (z=0-1,4-7,10,14-15,21-29)** | z | r | **Reserved**<br>Reserved. Not used in this product. |
| **CPU0_DMEM1_DONE** | 2 | rh | **CPU0 DMEM1 Test Done Status**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled [1]. |
| **CPU1_DMEM1_DONE** | 3 | rh | **CPU1 DMEM1 Test Done Status**<br>$0_B$ SSH instance is disabled<br>$1_B$ SSH instance is enabled [2]. |
| **R8** | 8 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **SADMA_DONE** | 9 | rh | **Safety DMA Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **MCDS_DONE** | 11 | rh | **MCDS memory Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **EMEM0_DONE** | 12 | rh | **EMEM0 Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **EMEM1_DONE** | 13 | rh | **EMEM1 Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **EMEM_XTM_DONE** | 16 | rh | **EMEM XTM memory Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **SPU_BUFFER0_DONE** | 17 | rh | **SPU BUFFER0 Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SPU_BUFFER1_DONE** | 18 | rh | **SPU BUFFER1 Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **SPU_CONFIG0_DONE** | 19 | rh | **SPU CONFIG0 memory Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **SPU_CONFIG1_DONE** | 20 | rh | **SPU CONFIG1 memory Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **MCAN10_DONE** | 30 | rh | **MCAN10 memory Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **MCAN20_DONE** | 31 | rh | **MCAN20 memory Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |

1) Please refer to separate section related to handling of the large DMEM on this device.

2) Please refer to separate section related to handling of the large DMEM on this device.

**MTU_MEMDONEi (i=2)**

**Memory Test Done Status Register i** **($0050_H$+i*4)** **Application Reset Value: FFFF FFFF$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **SPU_FFT31_RAM_DONE** | **SPU_FFT30_RAM_DONE** | **SPU_FFT21_RAM_DONE** | **SPU_FFT20_RAM_DONE** | **SPU_FFT11_RAM_DONE** | **SPU_FFT10_RAM_DONE** | **SPU_FFT01_RAM_DONE** | **SPU_FFT00_RAM_DONE** | **RES23** | **RES22** | **HSPDM_RAM_DONE** | **RES20** | **GIGETH_TX_DONE** | **GIGETH_RX_DONE** | **RES17** | **RES16** |
| rh | rh | rh | rh | rh | rh | rh | rh | r | r | rh | r | rh | rh | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **RES15** | **SCR_RAMINT_DONE** | **SCR_XRAM_DONE** | **R12** | **R11** | **R10** | **R9** | **R8** | **RES7** | **ERAY_MBF0_DONE** | **RES5** | **ERAY_TBF_IBF0_DONE** | **RES3** | **ERAY_OBF0_DONE** | **RES1** | **RES0** |
| r | rh | rh | rh | rh | rh | rh | rh | r | rh | r | rh | r | rh | r | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **RESz (z=0-1,3,5,7,15-17,20,22-23)** | z | r | **Reserved**<br>Reserved. Not used in this product. |
| **ERAY_OBF0_DONE** | 2 | rh | **ERAY OBF0 Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **ERAY_TBF_IBF0_DONE** | 4 | rh | **ERAY TBF IBF0 memory Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ERAY_MBF0_DONE** | 6 | rh | **ERAY MBF0 memory Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **R8** | 8 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R9** | 9 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R10** | 10 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R11** | 11 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R12** | 12 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **SCR_XRAM_DONE** | 13 | rh | **SCR XRAM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **SCR_RAMINT_DONE** | 14 | rh | **SCR Internal RAM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **GIGETH_RX_DONE** | 18 | rh | **Gigabit Ethernet RX memoryTest Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **GIGETH_TX_DONE** | 19 | rh | **Gigabit Ethernet TX memoryTest Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **HSPDM_RAM_DONE** | 21 | rh | **HDSPDM RAM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **SPU_FFT00_RAM_DONE** | 24 | rh | **SPU FFT00 RAM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **SPU_FFT01_RAM_DONE** | 25 | rh | **SPU FFT01 RAM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **SPU_FFT10_RAM_DONE** | 26 | rh | **SPU FFT10 RAM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **SPU_FFT11_RAM_DONE** | 27 | rh | **SPU FFT11 RAM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **SPU_FFT20_RAM_DONE** | 28 | rh | **SPU FFT20 RAM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SPU_FFT21_RAM_DONE** | 29 | rh | **SPU FFT21 RAM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **SPU_FFT30_RAM_DONE** | 30 | rh | **SPU FFT30 RAM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **SPU_FFT31_RAM_DONE** | 31 | rh | **SPU FFT31 RAM Test Done Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |

## 13.4.5 MEMFDA Implementation

### Memory Test FDA Status Register i

Each bit in one of the memory test done status registers MEMFDAx reflects the status of the MSTATUS.FDA bit in the corresponding SSH. See the implementation section for the implemented register bits.

**MTU_MEMFDAi (i=0)**
**Memory Test FDA Status Register i** **($0060_H$+i*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **LMU10_FDA** | **LMU00_FDA** | **RES29** | **RES28** | **RES27** | **RES26** | **RES25** | **RES24** | **RES23** | **RES22** | **RES21** | **RES20** | **RES19** | **RES18** | **RES17** | **RES16** |
| rh | rh | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **RES15** | **CPU2_DLMU_FDA** | **CPU2_PTAG_FDA** | **CPU2_PMEM_FDA** | **CPU2_DTAG_FDA** | **CPU2_DMEM_FDA** | **CPU1_DLMU_STBY_FDA** | **CPU1_PTAG_FDA** | **CPU1_PMEM_FDA** | **CPU1_DTAG_FDA** | **CPU1_DMEM_FDA** | **CPU0_DLMU_STBY_FDA** | **CPU0_PTAG_FDA** | **CPU0_PMEM_FDA** | **CPU0_DTAG_FDA** | **CPU0_DMEM_FDA** |
| r | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CPU0_DMEM_FDA** | 0 | rh | **CPU0 DMEM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **RESz (z=15-29)** | z | r | **Reserved**<br>Reserved. Not used in this product. |
| **CPU0_DTAG_FDA** | 1 | rh | **CPU0 DTAG Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **CPU0_PMEM_FDA** | 2 | rh | **CPU0 PMEM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **CPU0_PTAG_FDA** | 3 | rh | **CPU0 PTAG Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CPU0_DLMU_STBY_FDA** | 4 | rh | **CPU0 STANDBY DLMU Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **CPU1_DMEM_FDA** | 5 | rh | **CPU1 DMEM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **CPU1_DTAG_FDA** | 6 | rh | **CPU1 DTAG Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **CPU1_PMEM_FDA** | 7 | rh | **CPU1 PMEM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **CPU1_PTAG_FDA** | 8 | rh | **CPU1 PTAG Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **CPU1_DLMU_STBY_FDA** | 9 | rh | **CPU1 STANDBY DLMU Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **CPU2_DMEM_FDA** | 10 | rh | **CPU2 DMEM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **CPU2_DTAG_FDA** | 11 | rh | **CPU2 DTAG Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **CPU2_PMEM_FDA** | 12 | rh | **CPU2 PMEM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **CPU2_PTAG_FDA** | 13 | rh | **CPU2 PTAG Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **CPU2_DLMU_FDA** | 14 | rh | **CPU2 DLMU memory Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **LMU00_FDA** | 30 | rh | **LMU00 Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **LMU10_FDA** | 31 | rh | **LMU10 Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |

**MTU_MEMFDAi (i=1)**

**Memory Test FDA Status Register i** **($0060_H$+i*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MCAN20_FDA | MCAN10_FDA | RES29 | RES28 | RES27 | RES26 | RES25 | RES24 | RES23 | RES22 | RES21 | SPU_CONFIG1_FDA | SPU_CONFIG0_FDA | SPU_BUFFER1_FDA | SPU_BUFFER0_FDA | EMEM_XTM_FDA |
| rh | rh | r | r | r | r | r | r | r | r | r | rh | rh | rh | rh | rh |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RES15 | RES14 | EMEM1_FDA | EMEM0_FDA | MCDS_FDA | RES10 | SADMA_FDA | R8 | RES7 | RES6 | RES5 | RES4 | CPU1_DMEM1_FDA | CPU0_DMEM1_FDA | RES1 | RES0 |
| r | r | rh | rh | rh | r | rh | rh | r | r | r | r | rh | rh | r | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **RESz (z=0-1,4-7,10,14-15,21-29)** | z | r | **Reserved**<br>Reserved. Not used in this product. |
| **CPU0_DMEM1_FDA** | 2 | rh | **CPU0 DMEM1 Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **CPU1_DMEM1_FDA** | 3 | rh | **CPU1 DMEM1 Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **R8** | 8 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **SADMA_FDA** | 9 | rh | **Safety DMA Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **MCDS_FDA** | 11 | rh | **MCDS memory Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **EMEM0_FDA** | 12 | rh | **EMEM0 Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **EMEM1_FDA** | 13 | rh | **EMEM1 Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **EMEM_XTM_FDA** | 16 | rh | **EMEM XTM memory Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **SPU_BUFFER0_FDA** | 17 | rh | **SPU BUFFER0 Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SPU_BUFFER1_FDA** | 18 | rh | **SPU BUFFER1 Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **SPU_CONFIG0_FDA** | 19 | rh | **SPU CONFIG0 memory Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **SPU_CONFIG1_FDA** | 20 | rh | **SPU CONFIG1 memory Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **MCAN10_FDA** | 30 | rh | **MCAN10 memory Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **MCAN20_FDA** | 31 | rh | **MCAN20 memory Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |

**MTU_MEMFDAi (i=2)**

**Memory Test FDA Status Register i** **($0060_H$+i*4)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **SPU_FFT31_RAM_FDA** | **SPU_FFT30_RAM_FDA** | **SPU_FFT21_RAM_FDA** | **SPU_FFT20_RAM_FDA** | **SPU_FFT11_RAM_FDA** | **SPU_FFT10_RAM_FDA** | **SPU_FFT01_RAM_FDA** | **SPU_FFT00_RAM_FDA** | **RES23** | **RES22** | **HSPDM_RAM_FDA** | **RES20** | **GIGETH_TX_FDA** | **GIGETH_RX_FDA** | **RES17** | **RES16** |
| rh | rh | rh | rh | rh | rh | rh | rh | r | r | rh | r | rh | rh | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **RES15** | **SCR_RAMINT_FDA** | **SCR_XRAM_FDA** | **R12** | **R11** | **R10** | **R9** | **R8** | **RES7** | **ERAY_MBF0_FDA** | **RES5** | **ERAY_TBF_IBF0_FDA** | **RES3** | **ERAY_OBF0_FDA** | **RES1** | **RES0** |
| r | rh | rh | rh | rh | rh | rh | rh | r | rh | r | rh | r | rh | r | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **RESz (z=0-1,3,5,7,15-17,20,22-23)** | z | r | **Reserved**<br>Reserved. Not used in this product. |
| **ERAY_OBF0_FDA** | 2 | rh | **ERAY OBF0 Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **ERAY_TBF_IBF0_FDA** | 4 | rh | **ERAY TBF IBF0 memory Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **ERAY_MBF0_FDA** | 6 | rh | **ERAY MBF0 memory Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **R8** | 8 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R9** | 9 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R10** | 10 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R11** | 11 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **R12** | 12 | rh | **Reserved - Res**<br>Reserved. Not used in this product. |
| **SCR_XRAM_FDA** | 13 | rh | **SCR XRAM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **SCR_RAMINT_FDA** | 14 | rh | **SCR Internal RAM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **GIGETH_RX_FDA** | 18 | rh | **Gigabit Ethernet RX memory Test FDA Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **GIGETH_TX_FDA** | 19 | rh | **Gigabit Ethernet TX SSH memory Test FDA Status**<br>$0_B$ SSH MSTATUS.DONE = 0<br>$1_B$ SSH MSTATUS.DONE = 1 |
| **HSPDM_RAM_FDA** | 21 | rh | **HDSPDM RAM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **SPU_FFT00_RAM_FDA** | 24 | rh | **SPU FFT00 RAM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **SPU_FFT01_RAM_FDA** | 25 | rh | **SPU FFT01 RAM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **SPU_FFT10_RAM_FDA** | 26 | rh | **SPU FFT10 RAM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **SPU_FFT11_RAM_FDA** | 27 | rh | **SPU FFT11 RAM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **SPU_FFT20_RAM_FDA** | 28 | rh | **SPU FFT20 RAM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **SPU_FFT21_RAM_FDA** | 29 | rh | **SPU FFT21 RAM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SPU_FFT30_RAM_FDA** | 30 | rh | **SPU FFT30 RAM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |
| **SPU_FFT31_RAM_FDA** | 31 | rh | **SPU FFT31 RAM Test FDA Status**<br>$0_B$ SSH MSTATUS.FDA = 0<br>$1_B$ SSH MSTATUS.FDA = 1 |

## 13.5 SSH Instances

The system SRAMs do not all have the same configuration. **Table 62 "SSH instances" on Page 27** shows the instance-specific configurations of the SRAM Support Hardware.

The ECC values for all SRAMs are computed only out of the data information*.

The base address of an SSH instance MCx can be calculated from the MC_BASE (defined in the platform chapter) as: Base Address of SSH instance x (MCx) = MC_BASE + x*0x100

**Table 62 SSH instances**

| (MCx) x = | Module | Error Addr Buffer (ETRR) Depth | ECC type | ECC granularity | Mux Factor |
|---|---|---|---|---|---|
| 0 | CPU0_DMEM | 5 | SECDED | 2 | 16 |
| 1 | CPU0_DTAG | 5 | SECDED | 2 | 4 |
| 2 | CPU0_PMEM | 5 | SECDED | 2 | 8 |
| 3 | CPU0_PTAG | 5 | DED | 2 | 4 |
| 4 | CPU0_DLMU_STBY | 5 | SECDED | 2 | 8 |
| 5 | CPU1_DMEM | 5 | SECDED | 2 | 16 |
| 6 | CPU1_DTAG | 5 | SECDED | 2 | 4 |
| 7 | CPU1_PMEM | 5 | SECDED | 2 | 8 |
| 8 | CPU1_PTAG | 5 | DED | 2 | 4 |
| 9 | CPU1_DLMU_STBY | 5 | SECDED | 2 | 8 |
| 10 | CPU2_DMEM | 5 | SECDED | 2 | 16 |
| 11 | CPU2_DTAG | 5 | SECDED | 2 | 4 |
| 12 | CPU2_PMEM | 5 | SECDED | 2 | 8 |
| 13 | CPU2_PTAG | 5 | DED | 2 | 4 |
| 14 | CPU2_DLMU | 5 | SECDED | 2 | 8 |
| 15 | Reserved | | | | |
| 16 | Reserved | | | | |
| 17 | Reserved | | | | |
| 18 | Reserved | | | | |
| 19 | Reserved | | | | |
| 20 | Reserved | | | | |
| 21 | Reserved | | | | |
| 22 | Reserved | | | | |
| 23 | Reserved | | | | |
| 24 | Reserved | | | | |
| 25 | Reserved | | | | |
| 26 | Reserved | | | | |
| 27 | Reserved | | | | |
| 28 | Reserved | | | | |
| 29 | Reserved | | | | |

**Table 62 SSH instances** (cont'd)

| (MCx) x = | Module | Error Addr Buffer (ETRR) Depth | ECC type | ECC granularity | Mux Factor |
|---|---|---|---|---|---|
| 30 | LMU00 | 5 | SECDED | 2 | 8 |
| 31 | LMU10 | 5 | SECDED | 2 | 8 |
| 32 | Reserved | | | | |
| 33 | Reserved | | | | |
| 34 | CPU0_DMEM1 | 5 | SECDED | 2 | 16 |
| 35 | CPU1_DMEM1 | 5 | SECDED | 2 | 16 |
| 36-37 | Reserved | | | | |
| 38 | Reserved | | | | |
| 39 | Reserved | | | | |
| 41 | SADMA | 5 | SECDED | 1 | 4 |
| 42 | Reserved | | | | |
| 43 | MCDS | 5 | DED | 1 | 4 |
| 44 | EMEM0 | 5 | SECDED | 1 | 8 |
| 45 | EMEM1 | 5 | SECDED | 1 | 8 |
| 46 | Reserved | | | | |
| 47 | Reserved | | | | |
| 48 | EMEM_XTM | 5 | SECDED | 1 | 4 |
| 49 | SPU_BUFFER0 | 5 | SECDED | 1 | 4 |
| 50 | SPU_BUFFER1 | 5 | SECDED | 1 | 4 |
| 51 | SPU_CONFIG0 | 5 | SECDED | 2 | 8 |
| 52 | SPU_CONFIG1 | 5 | SECDED | 2 | 8 |
| 53 | Reserved | | | | |
| 54 | Reserved | | | | |
| 55 | Reserved | | | | |
| 56 | Reserved | | | | |
| 57 | Reserved | | | | |
| 58 | Reserved | | | | |
| 59 | Reserved | | | | |
| 60 | Reserved | | | | |
| 61 | **Reserved** | | | | |
| 62 | M_CAN10 | 5 | SECDED | 1 | 16 |
| 63 | M_CAN20 | 5 | SECDED | 1 | 16 |
| 64 | Reserved | | | | |
| 65 | Reserved | | | | |
| 66 | ERAY_OBF0 | 5 | SECDED | 1 | 4 |
| 67 | Reserved | | | | |
| 68 | ERAY_TBF_IBF0 | 5 | SECDED | 1 | 4 |

**Table 62 SSH instances** (cont'd)

| (MCx) x = | Module | Error Addr Buffer (ETRR) Depth | ECC type | ECC granularity | Mux Factor |
|---|---|---|---|---|---|
| 69 | Reserved | | | | |
| 70 | ERAY_MBF0 | 5 | SECDED | 1 | 4 |
| 71 | Reserved | | | | |
| 77 | SCR_XRAM | 5 | SECDED | 2 | 8 |
| 78 | SCR_RAMINT | 5 | SECDED | 1 | 4 |
| 79 | Reserved | | | | |
| 80 | Reserved | | | | |
| 81 | Reserved | | | | |
| 82 | GIGETH_RX_RAM | 5 | SECDED | 1 | 4 |
| 83 | GIGETH_TX_RAM | 5 | SECDED | 1 | 4 |
| 84 | Reserved | | | | |
| 85 | HSPDM_RAM | 5 | SECDED | 1 | 8 |
| 86- 87 | Reserved | | | | |
| 88 | SPU_FFT00 | 5 | SECDED | 1 | 4 |
| 89 | SPU_FFT01 | 5 | SECDED | 1 | 4 |
| 90 | SPU_FFT10 | 5 | SECDED | 1 | 4 |
| 91 | SPU_FFT11 | 5 | SECDED | 1 | 4 |
| 92 | SPU_FFT20 | 5 | SECDED | 1 | 4 |
| 93 | SPU_FFT21 | 5 | SECDED | 1 | 4 |
| 94 | SPU_FFT30 | 5 | SECDED | 1 | 4 |
| 95 | SPU_FFT31 | 5 | SECDED | 1 | 4 |

### 13.5.1 Ganging for SRAM test and initialization

Whenever an MBIST test or SRAM initialization is started via the MTU/SSH, there is a certain jump in the current consumption, due to the parallel accesses to the SRAM cells during the test or initialization. This current jump is different for the different SRAMs in the product, and depends on the size of the SRAM, the clock frequency e.t.c.

If too many SRAMs are tested or initialized in parallel, it may result in a significant current jump, which may put the device outside of the specified operating conditions. On the other hand, in order to reduce the overall test and/or initialization time, it may be imperative for the application to perform the test or initialization of many SRAMs in parallel.

In order to achieve this trade-off between current jump and test/initialization time - it is advised to partition the available SRAMs into different “Gangs”. This is referred to as Ganging. The SRAMs in each Gang are all initialized/tested in parallel - while the Gangs themselves are executed in sequence, one after the other. This ensures that the current jump never exceeds the allowed limits, while at the same time the overall test time is minimized.

Gangs 0-16 are defined for the platform. Depending on the device and the number of implemented SRAMs, the number of gangs needed may be less. The numbering of the gangs is not important and does not signify any order.

In the below ganging tables, a ganging configuration for this product is shown. This assumes a 4-march element (4N) non-destructive test (r,w*,r*,w) on a zero-data background (i.e. the SRAM is cleared with ECC-correct zero data). The same gangs may be used for SRAM initialization.

The provided ganging example is ensured to satisfy the specified limitations in current jumps of this device, while at the same time optimizing the overall test time. All implemented SRAMs are considered here. If certain SRAMs are not of interest to a particular application, then the gangs may be modified appropriately.

**Table 63 GANG-0**

| MCx(x=) | Module / SRAM |
|---|---|
| 44 | EMEM0 |
| 62 | M_CAN10 |
| 63 | M_CAN20 |

**Table 64 GANG-1**

| MCx(x=) | Module / SRAM |
|---|---|
| 45 | EMEM1 |

**Table 65 GANG-2**

| MCx(x=) | Module / SRAM |
|---|---|
| 34 | CPU0_DMEM1 |
| 35 | CPU1_DMEM1 |
| 70 | ERAY_MBF0 |
| 77 | SCR_XRAM |

**Table 66 GANG-3**

| MCx(x=) | Module / SRAM |
|---|---|
| 04 | CPU0_DLMU_STBY |
| 30 | LMU00 |
| 31 | LMU10 |

**Table 67 GANG-4**

| MCx(x=) | Module / SRAM |
|---|---|
| 00 | CPU0_DMEM |
| 05 | CPU1_DMEM |
| 85 | HSPDM_RAM |

**Table 68 GANG-5**

| MCx(x=) | Module / SRAM |
|---|---|
| 09 | CPU1_DLMU_STBY |
| 10 | CPU2_DMEM |
| 43 | MCDS |
| 51 | SPU_CONFIG0 |
| 78 | SCR_RAMINT |
| 83 | GIGETH_TX_RAM |

**Table 69 GANG-6**

| MCx(x=) | Module / SRAM |
|---|---|
| 02 | CPU0_PMEM |
| 03 | CPU0_PTAG |
| 08 | CPU1_PTAG |
| 13 | CPU2_PTAG |
| 14 | CPU2_DLMU |
| 48 | EMEM_XTM |
| 52 | SPUCONFIG1 |

**Table 70 GANG-7**

| MCx(x=) | Module / SRAM |
|---|---|
| 01 | CPU0_DTAG |
| 07 | CPU1_PMEM |
| 12 | CPU2_PMEM |
| 68 | ERAY_TBF_IBF0 |
| 82 | GIGETH_RX_RAM |

**Table 70 GANG-7**

| MCx(x=) | Module / SRAM |
|---|---|
| 88 | SPU_FFT00 |
| 89 | SPU_FFT01 |

**Table 71 GANG-8**

| MCx(x=) | Module / SRAM |
|---|---|
| 06 | CPU1_DTAG |
| 11 | CPU2_DTAG |
| 49 | SPU_BUFFER0 |
| 50 | SPU_BUFFER1 |
| 90 | SPU_FFT10 |
| 91 | SPU_FFT11 |

**Table 72 GANG-9**

| MCx(x=) | Module / SRAM |
|---|---|
| 41 | SADMA |
| 66 | ERAY_OBF0 |
| 92 | SPU_FFT20 |
| 93 | SPU_FFT21 |
| 94 | SPU_FFT30 |
| 95 | SPU_FFT31 |

## 13.6 Connectivity

**Table 73 Connections of MTU**

| Interface Signals | connects | | Description |
|---|---|---|---|
| MTU:CPU0DCMAP | to | cpu_pfi_pfrwb_0:tc162p_dcache_map | CPU dcache mapped indicator per cpu |
| MTU:CPU1DCMAP | to | cpu_pfi_pfrwb_1:tc162p_dcache_map | CPU dcache mapped indicator per cpu |
| MTU:CPU2DCMAP | to | cpu_2:tc162p_dcache_map | CPU dcache mapped indicator per cpu |
| MTU:CPU0PCMAP | to | cpu_pfi_pfrwb_0:tc162p_pcache_map | CPU pcache mapped indicator per cpu |
| MTU:CPU1PCMAP | to | cpu_pfi_pfrwb_1:tc162p_pcache_map | CPU pcache mapped indicator per cpu |
| MTU:CPU2PCMAP | to | cpu_2:tc162p_pcache_map | CPU pcache mapped indicator per cpu |
| MTU:dmu_no_ram_init | from | DMU:MTU_NO_RAMIN | Disable RAM auto-initialization |
| MTU:scu_hsm_dbg | from | SCU:scu_hsm_dbg | HSM debug enable from SCU |

**Table 73 Connections of MTU** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| MTU:sleep_n | from | SCU:scu_syst_sleep_n | Sleep request |
| MTU:DONE_INT | to | INT:mtu.DONE_INT | MTU Done Service Request |
| MTU:tcu_hsm_dbg_analysis_en | from | TCU:hsm_debug_mode | HSM debug request from TCU |

## 13.7 Revision History

**Table 74 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V7.4.7** | | |
| **Page 33** | Revision History entries up to V7.4.6 removed. | |
| **Page 30** - **Page 32** | Ganging information updated. | |
| **Page 32** | Connectivity information updated. | |
| **V7.4.8** | | |
| **Page 33** | Revision History entries up to V7.4.7 removed. | |
| **Page 9** | MEMMAP Reserved (not implemented) bits changed to "read". | |
| **Page 3**, **Page 16**, **Page 21** | "SADMA" changed to "Safety DMA" in short description of MEMTEST1/9, MEMDONE1/9 and MEMFDA1/9 bit fields. | |
| **V7.4.9** | | |
| **Page 30** | Typo fixed (CPUx_DMEM2 -> CPUx_DMEM1). | |
| **V7.4.10** | | |
| – | No functional changes. | |
| **V7.4.11** | | |
| – | No functional changes. | |
| **V7.4.12** | | |
| – | No functional changes. | |
| **V7.4.13** | | |
| **Page 12** | Wrongly mentioned bit field in MTU_MEMSTATi (i=2) fixed. | |

# 14 General Purpose I/O Ports and Peripheral I/O Lines (Ports)

This chapter supplements the family documentation with device specific information for TC35x.

## 14.1 TC35x Specific IP Configuration

The Ports configuration (which Port modules are implemented, their width and functionality) is represented by the device specific register set shown in this chapter.

## 14.2 TC35x Specific Register Set

**Table 75 Register Address Space - Pn**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| P00 | $F003A000_H$ | $F003A0FF_H$ | SPB bus slave interface |
| P02 | $F003A200_H$ | $F003A2FF_H$ | SPB bus slave interface |
| P10 | $F003AA00_H$ | $F003AAFF_H$ | SPB bus slave interface |
| P11 | $F003AB00_H$ | $F003ABFF_H$ | SPB bus slave interface |
| P12 | $F003AC00_H$ | $F003ACFF_H$ | SPB bus slave interface |
| P14 | $F003AE00_H$ | $F003AEFF_H$ | SPB bus slave interface |
| P15 | $F003AF00_H$ | $F003AFFF_H$ | SPB bus slave interface |
| P20 | $F003B400_H$ | $F003B4FF_H$ | SPB bus slave interface |
| P21 | $F003B500_H$ | $F003B5FF_H$ | SPB bus slave interface |
| P22 | $F003B600_H$ | $F003B6FF_H$ | SPB bus slave interface |
| P23 | $F003B700_H$ | $F003B7FF_H$ | SPB bus slave interface |
| P32 | $F003C000_H$ | $F003C0FF_H$ | SPB bus slave interface |
| P33 | $F003C100_H$ | $F003C1FF_H$ | SPB bus slave interface |
| P34 | $F003C200_H$ | $F003C2FF_H$ | SPB bus slave interface |

**Register Overview Tables of Pn**

**Table 76 Register Overview - P00 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P00_OUT | Port 00 Output Register | $000_H$ | U,SV | U,SV,P | Application Reset | 23 |
| P00_OMR | Port 00 Output Modification Register | $004_H$ | U,SV | U,SV,P | Application Reset | 26 |
| P00_ID | Port 00 Identification Register | $008_H$ | U,SV | BE | Application Reset | 31 |
| | Reserved ($004_H$ Byte) | $00C_H$ | BE | BE | | |
| P00_IOCR0 | Port 00 Input/Output Control Register 0 | $010_H$ | U,SV | U,SV,P | See page 32 | 32 |
| P00_IOCR4 | Port 00 Input/Output Control Register 4 | $014_H$ | U,SV | U,SV,P | See page 36 | 36 |

**Table 76 Register Overview - P00 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P00_IOCR8 | Port 00 Input/Output Control Register 8 | $018_H$ | U,SV | U,SV,P | See page 39 | 39 |
| P00_IOCR12 | Port 00 Input/Output Control Register 12 | $01C_H$ | U,SV | U,SV,P | See page 41 | 41 |
| | Reserved ($004_H$ Byte) | $020_H$ | BE | BE | | |
| P00_IN | Port 00 Input Register | $024_H$ | U,SV | BE | Application Reset | 42 |
| | Reserved ($004_H$ Byte) (x=0-5) | $028_H$+x* 4 | BE | BE | | |
| P00_PDR0 | Port 00 Pad Driver Mode Register 0 | $040_H$ | U,SV | SV,E,P | See page 45 | 45 |
| P00_PDR1 | Port 00 Pad Driver Mode Register 1 | $044_H$ | U,SV | SV,E,P | See page 50 | 50 |
| | Reserved ($004_H$ Byte) (x=0-1) | $048_H$+x* 4 | BE | BE | | |
| P00_ESR | Port 00 Emergency Stop Register | $050_H$ | U,SV | SV,E,P | Application Reset | 52 |
| | Reserved ($004_H$ Byte) (x=0-2) | $054_H$+x* 4 | BE | BE | | |
| P00_PDISC | Port 00 Pin Function Decision Control Register | $060_H$ | U,SV | SV,E,P | See page 58 | 58 |
| P00_PCSR | Port 00 Pin Controller Select Register | $064_H$ | U,SV | SV,SE | Application Reset | 63 |
| | Reserved ($004_H$ Byte) (x=0-1) | $068_H$+x* 4 | BE | BE | | |
| P00_OMSR0 | Port 00 Output Modification Set Register 0 | $070_H$ | U,SV | U,SV,P | Application Reset | 69 |
| P00_OMSR4 | Port 00 Output Modification Set Register 4 | $074_H$ | U,SV | U,SV,P | Application Reset | 72 |
| P00_OMSR8 | Port 00 Output Modification Set Register 8 | $078_H$ | U,SV | U,SV,P | Application Reset | 74 |
| P00_OMSR12 | Port 00 Output Modification Set Register 12 | $07C_H$ | U,SV | U,SV,P | Application Reset | 75 |
| P00_OMCR0 | Port 00 Output Modification Clear Register 0 | $080_H$ | U,SV | U,SV,P | Application Reset | 76 |
| P00_OMCR4 | Port 00 Output Modification Clear Register 4 | $084_H$ | U,SV | U,SV,P | Application Reset | 78 |
| P00_OMCR8 | Port 00 Output Modification Clear Register 8 | $088_H$ | U,SV | U,SV,P | Application Reset | 81 |
| P00_OMCR12 | Port 00 Output Modification Clear Register 12 | $08C_H$ | U,SV | U,SV,P | Application Reset | 82 |

**Table 76 Register Overview - P00 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P00_OMSR | Port 00 Output Modification Set Register | $090_H$ | U,SV | U,SV,P | Application Reset | 83 |
| P00_OMCR | Port 00 Output Modification Clear Register | $094_H$ | U,SV | U,SV,P | Application Reset | 86 |
| | Reserved ($004_H$ Byte) (x=0-1) | $098_H$+x* 4 | BE | BE | | |
| | Reserved ($004_H$ Byte) (x=0-13) | $0C0_H$+x* 4 | BE | BE | | |
| P00_ACCEN1 | Port 00 Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | 92 |
| P00_ACCEN0 | Port 00 Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | 94 |

**Table 77 Register Overview - P02 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P02_OUT | Port 02 Output Register | $000_H$ | U,SV | U,SV,P | Application Reset | 24 |
| P02_OMR | Port 02 Output Modification Register | $004_H$ | U,SV | U,SV,P | Application Reset | 28 |
| P02_ID | Port 02 Identification Register | $008_H$ | U,SV | BE | Application Reset | 31 |
| | Reserved ($004_H$ Byte) | $00C_H$ | BE | BE | | |
| P02_IOCR0 | Port 02 Input/Output Control Register 0 | $010_H$ | U,SV | U,SV,P | See page 32 | 32 |
| P02_IOCR4 | Port 02 Input/Output Control Register 4 | $014_H$ | U,SV | U,SV,P | See page 36 | 36 |
| P02_IOCR8 | Port 02 Input/Output Control Register 8 | $018_H$ | U,SV | U,SV,P | See page 39 | 39 |
| | Reserved ($004_H$ Byte) | $020_H$ | BE | BE | | |
| P02_IN | Port 02 Input Register | $024_H$ | U,SV | BE | Application Reset | 43 |
| | Reserved ($004_H$ Byte) (x=0-5) | $028_H$+x* 4 | BE | BE | | |
| P02_PDR0 | Port 02 Pad Driver Mode Register 0 | $040_H$ | U,SV | SV,E,P | See page 45 | 45 |
| P02_PDR1 | Port 02 Pad Driver Mode Register 1 | $044_H$ | U,SV | SV,E,P | See page 51 | 51 |
| | Reserved ($004_H$ Byte) (x=0-1) | $048_H$+x* 4 | BE | BE | | |

**Table 77 Register Overview - P02 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P02_ESR | Port 02 Emergency Stop Register | $050_H$ | U,SV | SV,E,P | Application Reset | 54 |
| | Reserved ($004_H$ Byte) (x=0-2) | $054_H$+x*4 | BE | BE | | |
| P02_PDISC | Port 02 Pin Function Decision Control Register | $060_H$ | U,SV | SV,E,P | See page 60 | 60 |
| P02_PCSR | Port 02 Pin Controller Select Register | $064_H$ | U,SV | SV,SE | Application Reset | 64 |
| | Reserved ($004_H$ Byte) (x=0-1) | $068_H$+x*4 | BE | BE | | |
| P02_OMSR0 | Port 02 Output Modification Set Register 0 | $070_H$ | U,SV | U,SV,P | Application Reset | 69 |
| P02_OMSR4 | Port 02 Output Modification Set Register 4 | $074_H$ | U,SV | U,SV,P | Application Reset | 72 |
| P02_OMSR8 | Port 02 Output Modification Set Register 8 | $078_H$ | U,SV | U,SV,P | Application Reset | 74 |
| P02_OMCR0 | Port 02 Output Modification Clear Register 0 | $080_H$ | U,SV | U,SV,P | Application Reset | 76 |
| P02_OMCR4 | Port 02 Output Modification Clear Register 4 | $084_H$ | U,SV | U,SV,P | Application Reset | 78 |
| P02_OMCR8 | Port 02 Output Modification Clear Register 8 | $088_H$ | U,SV | U,SV,P | Application Reset | 81 |
| P02_OMSR | Port 02 Output Modification Set Register | $090_H$ | U,SV | U,SV,P | Application Reset | 84 |
| P02_OMCR | Port 02 Output Modification Clear Register | $094_H$ | U,SV | U,SV,P | Application Reset | 88 |
| | Reserved ($004_H$ Byte) (x=0-1) | $098_H$+x*4 | BE | BE | | |
| | Reserved ($004_H$ Byte) (x=0-13) | $0C0_H$+x*4 | BE | BE | | |
| P02_ACCEN1 | Port 02 Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | 92 |
| P02_ACCEN0 | Port 02 Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | 94 |

**Table 78 Register Overview - P10 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P10_OUT | Port 10 Output Register | $000_H$ | U,SV | U,SV,P | Application Reset | 24 |
| P10_OMR | Port 10 Output Modification Register | $004_H$ | U,SV | U,SV,P | Application Reset | 28 |
| P10_ID | Port 10 Identification Register | $008_H$ | U,SV | BE | Application Reset | 31 |
| | Reserved ($004_H$ Byte) | $00C_H$ | BE | BE | | |
| P10_IOCR0 | Port 10 Input/Output Control Register 0 | $010_H$ | U,SV | U,SV,P | See page 32 | 32 |
| P10_IOCR4 | Port 10 Input/Output Control Register 4 | $014_H$ | U,SV | U,SV,P | See page 36 | 36 |
| P10_IOCR8 | Port 10 Input/Output Control Register 8 | $018_H$ | U,SV | U,SV,P | See page 39 | 39 |
| | Reserved ($004_H$ Byte) | $020_H$ | BE | BE | | |
| P10_IN | Port 10 Input Register | $024_H$ | U,SV | BE | Application Reset | 43 |
| | Reserved ($004_H$ Byte) (x=0-5) | $028_H$+x* 4 | BE | BE | | |
| P10_PDR0 | Port 10 Pad Driver Mode Register 0 | $040_H$ | U,SV | SV,E,P | See page 45 | 45 |
| P10_PDR1 | Port 10 Pad Driver Mode Register 1 | $044_H$ | U,SV | SV,E,P | See page 51 | 51 |
| | Reserved ($004_H$ Byte) (x=0-1) | $048_H$+x* 4 | BE | BE | | |
| P10_ESR | Port 10 Emergency Stop Register | $050_H$ | U,SV | SV,E,P | Application Reset | 54 |
| | Reserved ($004_H$ Byte) (x=0-2) | $054_H$+x* 4 | BE | BE | | |
| P10_PDISC | Port 10 Pin Function Decision Control Register | $060_H$ | U,SV | SV,E,P | See page 60 | 60 |
| P10_PCSR | Port 10 Pin Controller Select Register | $064_H$ | U,SV | SV,SE | Application Reset | 64 |
| | Reserved ($004_H$ Byte) (x=0-1) | $068_H$+x* 4 | BE | BE | | |
| P10_OMSR0 | Port 10 Output Modification Set Register 0 | $070_H$ | U,SV | U,SV,P | Application Reset | 69 |
| P10_OMSR4 | Port 10 Output Modification Set Register 4 | $074_H$ | U,SV | U,SV,P | Application Reset | 72 |
| P10_OMSR8 | Port 10 Output Modification Set Register 8 | $078_H$ | U,SV | U,SV,P | Application Reset | 74 |

**Table 78 Register Overview - P10 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P10_OMCR0 | Port 10 Output Modification Clear Register 0 | $080_H$ | U,SV | U,SV,P | Application Reset | 76 |
| P10_OMCR4 | Port 10 Output Modification Clear Register 4 | $084_H$ | U,SV | U,SV,P | Application Reset | 78 |
| P10_OMCR8 | Port 10 Output Modification Clear Register 8 | $088_H$ | U,SV | U,SV,P | Application Reset | 81 |
| P10_OMSR | Port 10 Output Modification Set Register | $090_H$ | U,SV | U,SV,P | Application Reset | 84 |
| P10_OMCR | Port 10 Output Modification Clear Register | $094_H$ | U,SV | U,SV,P | Application Reset | 88 |
| | Reserved ($004_H$ Byte) (x=0-1) | $098_H$+x* 4 | BE | BE | | |
| | Reserved ($004_H$ Byte) (x=0-13) | $0C0_H$+x* 4 | BE | BE | | |
| P10_ACCEN1 | Port 10 Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | 92 |
| P10_ACCEN0 | Port 10 Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | 94 |

**Table 79 Register Overview - P11 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P11_OUT | Port 11 Output Register | $000_H$ | U,SV | U,SV,P | Application Reset | 23 |
| P11_OMR | Port 11 Output Modification Register | $004_H$ | U,SV | U,SV,P | Application Reset | 26 |
| P11_ID | Port 11 Identification Register | $008_H$ | U,SV | BE | Application Reset | 31 |
| | Reserved ($004_H$ Byte) | $00C_H$ | BE | BE | | |
| P11_IOCR0 | Port 11 Input/Output Control Register 0 | $010_H$ | U,SV | U,SV,P | See page 32 | 32 |
| P11_IOCR4 | Port 11 Input/Output Control Register 4 | $014_H$ | U,SV | U,SV,P | See page 36 | 36 |
| P11_IOCR8 | Port 11 Input/Output Control Register 8 | $018_H$ | U,SV | U,SV,P | See page 39 | 39 |
| P11_IOCR12 | Port 11 Input/Output Control Register 12 | $01C_H$ | U,SV | U,SV,P | See page 41 | 41 |
| | Reserved ($004_H$ Byte) | $020_H$ | BE | BE | | |
| P11_IN | Port 11 Input Register | $024_H$ | U,SV | BE | Application Reset | 42 |

**Table 79 Register Overview - P11 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| | Reserved ($004_H$ Byte) (x=0-5) | $028_H$+x* 4 | BE | BE | | |
| P11_PDR0 | Port 11 Pad Driver Mode Register 0 | $040_H$ | U,SV | SV,E,P | See page 45 | 45 |
| P11_PDR1 | Port 11 Pad Driver Mode Register 1 | $044_H$ | U,SV | SV,E,P | See page 50 | 50 |
| | Reserved ($004_H$ Byte) (x=0-1) | $048_H$+x* 4 | BE | BE | | |
| P11_ESR | Port 11 Emergency Stop Register | $050_H$ | U,SV | SV,E,P | Application Reset | 52 |
| | Reserved ($004_H$ Byte) (x=0-2) | $054_H$+x* 4 | BE | BE | | |
| P11_PDISC | Port 11 Pin Function Decision Control Register | $060_H$ | U,SV | SV,E,P | See page 58 | 58 |
| P11_PCSR | Port 11 Pin Controller Select Register | $064_H$ | U,SV | SV,SE | Application Reset | 65 |
| | Reserved ($004_H$ Byte) (x=0-1) | $068_H$+x* 4 | BE | BE | | |
| P11_OMSR0 | Port 11 Output Modification Set Register 0 | $070_H$ | U,SV | U,SV,P | Application Reset | 69 |
| P11_OMSR4 | Port 11 Output Modification Set Register 4 | $074_H$ | U,SV | U,SV,P | Application Reset | 72 |
| P11_OMSR8 | Port 11 Output Modification Set Register 8 | $078_H$ | U,SV | U,SV,P | Application Reset | 74 |
| P11_OMSR12 | Port 11 Output Modification Set Register 12 | $07C_H$ | U,SV | U,SV,P | Application Reset | 75 |
| P11_OMCR0 | Port 11 Output Modification Clear Register 0 | $080_H$ | U,SV | U,SV,P | Application Reset | 76 |
| P11_OMCR4 | Port 11 Output Modification Clear Register 4 | $084_H$ | U,SV | U,SV,P | Application Reset | 78 |
| P11_OMCR8 | Port 11 Output Modification Clear Register 8 | $088_H$ | U,SV | U,SV,P | Application Reset | 81 |
| P11_OMCR12 | Port 11 Output Modification Clear Register 12 | $08C_H$ | U,SV | U,SV,P | Application Reset | 82 |
| P11_OMSR | Port 11 Output Modification Set Register | $090_H$ | U,SV | U,SV,P | Application Reset | 83 |
| P11_OMCR | Port 11 Output Modification Clear Register | $094_H$ | U,SV | U,SV,P | Application Reset | 86 |
| | Reserved ($004_H$ Byte) (x=0-1) | $098_H$+x* 4 | BE | BE | | |

**Table 79 Register Overview - P11 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| | Reserved ($004_H$ Byte) (x=0-13) | $0C0_H$+x*4 | BE | BE | | |
| P11_ACCEN1 | Port 11 Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | 92 |
| P11_ACCEN0 | Port 11 Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | 94 |

**Table 80 Register Overview - P12 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P12_OUT | Port 12 Output Register | $000_H$ | U,SV | U,SV,P | Application Reset | 25 |
| P12_OMR | Port 12 Output Modification Register | $004_H$ | U,SV | U,SV,P | Application Reset | 29 |
| P12_ID | Port 12 Identification Register | $008_H$ | U,SV | BE | Application Reset | 31 |
| | Reserved ($004_H$ Byte) | $00C_H$ | BE | BE | | |
| P12_IOCR0 | Port 12 Input/Output Control Register 0 | $010_H$ | U,SV | U,SV,P | See page 32 | 32 |
| | Reserved ($004_H$ Byte) | $020_H$ | BE | BE | | |
| P12_IN | Port 12 Input Register | $024_H$ | U,SV | BE | Application Reset | 44 |
| | Reserved ($004_H$ Byte) (x=0-5) | $028_H$+x*4 | BE | BE | | |
| P12_PDR0 | Port 12 Pad Driver Mode Register 0 | $040_H$ | U,SV | SV,E,P | See page 49 | 49 |
| | Reserved ($004_H$ Byte) (x=0-1) | $048_H$+x*4 | BE | BE | | |
| P12_ESR | Port 12 Emergency Stop Register | $050_H$ | U,SV | SV,E,P | Application Reset | 55 |
| | Reserved ($004_H$ Byte) (x=0-2) | $054_H$+x*4 | BE | BE | | |
| P12_PDISC | Port 12 Pin Function Decision Control Register | $060_H$ | U,SV | SV,E,P | See page 61 | 61 |
| P12_PCSR | Port 12 Pin Controller Select Register | $064_H$ | U,SV | SV,SE | Application Reset | 66 |
| | Reserved ($004_H$ Byte) (x=0-1) | $068_H$+x*4 | BE | BE | | |
| P12_OMSR0 | Port 12 Output Modification Set Register 0 | $070_H$ | U,SV | U,SV,P | Application Reset | 69 |

**Table 80 Register Overview - P12 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P12_OMCR0 | Port 12 Output Modification Clear Register 0 | $080_H$ | U,SV | U,SV,P | Application Reset | 76 |
| P12_OMSR | Port 12 Output Modification Set Register | $090_H$ | U,SV | U,SV,P | Application Reset | 85 |
| P12_OMCR | Port 12 Output Modification Clear Register | $094_H$ | U,SV | U,SV,P | Application Reset | 89 |
| | Reserved ($004_H$ Byte) (x=0-1) | $098_H$+x* 4 | BE | BE | | |
| | Reserved ($004_H$ Byte) (x=0-13) | $0C0_H$+x* 4 | BE | BE | | |
| P12_ACCEN1 | Port 12 Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | 92 |
| P12_ACCEN0 | Port 12 Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | 94 |

**Table 81 Register Overview - P14 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P14_OUT | Port 14 Output Register | $000_H$ | U,SV | U,SV,P | Application Reset | 24 |
| P14_OMR | Port 14 Output Modification Register | $004_H$ | U,SV | U,SV,P | Application Reset | 28 |
| P14_ID | Port 14 Identification Register | $008_H$ | U,SV | BE | Application Reset | 31 |
| | Reserved ($004_H$ Byte) | $00C_H$ | BE | BE | | |
| P14_IOCR0 | Port 14 Input/Output Control Register 0 | $010_H$ | U,SV | U,SV,P | See page 32 | 32 |
| P14_IOCR4 | Port 14 Input/Output Control Register 4 | $014_H$ | U,SV | U,SV,P | See page 36 | 36 |
| P14_IOCR8 | Port 14 Input/Output Control Register 8 | $018_H$ | U,SV | U,SV,P | See page 39 | 39 |
| | Reserved ($004_H$ Byte) | $020_H$ | BE | BE | | |
| P14_IN | Port 14 Input Register | $024_H$ | U,SV | BE | Application Reset | 43 |
| | Reserved ($004_H$ Byte) (x=0-5) | $028_H$+x* 4 | BE | BE | | |
| P14_PDR0 | Port 14 Pad Driver Mode Register 0 | $040_H$ | U,SV | SV,E,P | See page 45 | 45 |
| P14_PDR1 | Port 14 Pad Driver Mode Register 1 | $044_H$ | U,SV | SV,E,P | See page 51 | 51 |

**Table 81 Register Overview - P14 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| | Reserved ($004_H$ Byte) (x=0-1) | $048_H$+x*4 | BE | BE | | |
| P14_ESR | Port 14 Emergency Stop Register | $050_H$ | U,SV | SV,E,P | Application Reset | 54 |
| | Reserved ($004_H$ Byte) (x=0-2) | $054_H$+x*4 | BE | BE | | |
| P14_PDISC | Port 14 Pin Function Decision Control Register | $060_H$ | U,SV | SV,E,P | See page 60 | 60 |
| P14_PCSR | Port 14 Pin Controller Select Register | $064_H$ | U,SV | SV,SE | Application Reset | 64 |
| | Reserved ($004_H$ Byte) (x=0-1) | $068_H$+x*4 | BE | BE | | |
| P14_OMSR0 | Port 14 Output Modification Set Register 0 | $070_H$ | U,SV | U,SV,P | Application Reset | 69 |
| P14_OMSR4 | Port 14 Output Modification Set Register 4 | $074_H$ | U,SV | U,SV,P | Application Reset | 72 |
| P14_OMSR8 | Port 14 Output Modification Set Register 8 | $078_H$ | U,SV | U,SV,P | Application Reset | 74 |
| P14_OMCR0 | Port 14 Output Modification Clear Register 0 | $080_H$ | U,SV | U,SV,P | Application Reset | 76 |
| P14_OMCR4 | Port 14 Output Modification Clear Register 4 | $084_H$ | U,SV | U,SV,P | Application Reset | 78 |
| P14_OMCR8 | Port 14 Output Modification Clear Register 8 | $088_H$ | U,SV | U,SV,P | Application Reset | 81 |
| P14_OMSR | Port 14 Output Modification Set Register | $090_H$ | U,SV | U,SV,P | Application Reset | 84 |
| P14_OMCR | Port 14 Output Modification Clear Register | $094_H$ | U,SV | U,SV,P | Application Reset | 88 |
| | Reserved ($004_H$ Byte) (x=0-1) | $098_H$+x*4 | BE | BE | | |
| | Reserved ($004_H$ Byte) (x=0-13) | $0C0_H$+x*4 | BE | BE | | |
| P14_ACCEN1 | Port 14 Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | 92 |
| P14_ACCEN0 | Port 14 Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | 94 |

**Table 82 Register Overview - P15 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P15_OUT | Port 15 Output Register | $000_H$ | U,SV | U,SV,P | Application Reset | 24 |
| P15_OMR | Port 15 Output Modification Register | $004_H$ | U,SV | U,SV,P | Application Reset | 28 |
| P15_ID | Port 15 Identification Register | $008_H$ | U,SV | BE | Application Reset | 31 |
| | Reserved ($004_H$ Byte) | $00C_H$ | BE | BE | | |
| P15_IOCR0 | Port 15 Input/Output Control Register 0 | $010_H$ | U,SV | U,SV,P | See page 32 | 32 |
| P15_IOCR4 | Port 15 Input/Output Control Register 4 | $014_H$ | U,SV | U,SV,P | See page 36 | 36 |
| P15_IOCR8 | Port 15 Input/Output Control Register 8 | $018_H$ | U,SV | U,SV,P | See page 39 | 39 |
| | Reserved ($004_H$ Byte) | $020_H$ | BE | BE | | |
| P15_IN | Port 15 Input Register | $024_H$ | U,SV | BE | Application Reset | 43 |
| | Reserved ($004_H$ Byte) (x=0-5) | $028_H$+x* 4 | BE | BE | | |
| P15_PDR0 | Port 15 Pad Driver Mode Register 0 | $040_H$ | U,SV | SV,E,P | See page 45 | 45 |
| P15_PDR1 | Port 15 Pad Driver Mode Register 1 | $044_H$ | U,SV | SV,E,P | See page 51 | 51 |
| | Reserved ($004_H$ Byte) (x=0-1) | $048_H$+x* 4 | BE | BE | | |
| P15_ESR | Port 15 Emergency Stop Register | $050_H$ | U,SV | SV,E,P | Application Reset | 54 |
| | Reserved ($004_H$ Byte) (x=0-2) | $054_H$+x* 4 | BE | BE | | |
| P15_PDISC | Port 15 Pin Function Decision Control Register | $060_H$ | U,SV | SV,E,P | See page 60 | 60 |
| P15_PCSR | Port 15 Pin Controller Select Register | $064_H$ | U,SV | SV,SE | Application Reset | 64 |
| | Reserved ($004_H$ Byte) (x=0-1) | $068_H$+x* 4 | BE | BE | | |
| P15_OMSR0 | Port 15 Output Modification Set Register 0 | $070_H$ | U,SV | U,SV,P | Application Reset | 69 |
| P15_OMSR4 | Port 15 Output Modification Set Register 4 | $074_H$ | U,SV | U,SV,P | Application Reset | 72 |
| P15_OMSR8 | Port 15 Output Modification Set Register 8 | $078_H$ | U,SV | U,SV,P | Application Reset | 74 |

**Table 82 Register Overview - P15 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P15_OMCR0 | Port 15 Output Modification Clear Register 0 | $080_H$ | U,SV | U,SV,P | Application Reset | 76 |
| P15_OMCR4 | Port 15 Output Modification Clear Register 4 | $084_H$ | U,SV | U,SV,P | Application Reset | 78 |
| P15_OMCR8 | Port 15 Output Modification Clear Register 8 | $088_H$ | U,SV | U,SV,P | Application Reset | 81 |
| P15_OMSR | Port 15 Output Modification Set Register | $090_H$ | U,SV | U,SV,P | Application Reset | 84 |
| P15_OMCR | Port 15 Output Modification Clear Register | $094_H$ | U,SV | U,SV,P | Application Reset | 88 |
| | Reserved ($004_H$ Byte) (x=0-1) | $098_H$+x* 4 | BE | BE | | |
| | Reserved ($004_H$ Byte) (x=0-13) | $0C0_H$+x* 4 | BE | BE | | |
| P15_ACCEN1 | Port 15 Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | 92 |
| P15_ACCEN0 | Port 15 Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | 94 |

**Table 83 Register Overview - P20 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P20_OUT | Port 20 Output Register | $000_H$ | U,SV | U,SV,P | Application Reset | 23 |
| P20_OMR | Port 20 Output Modification Register | $004_H$ | U,SV | U,SV,P | Application Reset | 26 |
| P20_ID | Port 20 Identification Register | $008_H$ | U,SV | BE | Application Reset | 31 |
| | Reserved ($004_H$ Byte) | $00C_H$ | BE | BE | | |
| P20_IOCR0 | Port 20 Input/Output Control Register 0 | $010_H$ | U,SV | U,SV,P | See page 32 | 32 |
| P20_IOCR4 | Port 20 Input/Output Control Register 4 | $014_H$ | U,SV | U,SV,P | See page 36 | 36 |
| P20_IOCR8 | Port 20 Input/Output Control Register 8 | $018_H$ | U,SV | U,SV,P | See page 39 | 39 |
| P20_IOCR12 | Port 20 Input/Output Control Register 12 | $01C_H$ | U,SV | U,SV,P | See page 41 | 41 |
| | Reserved ($004_H$ Byte) | $020_H$ | BE | BE | | |
| P20_IN | Port 20 Input Register | $024_H$ | U,SV | BE | Application Reset | 42 |

**Table 83 Register Overview - P20 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| | Reserved ($004_H$ Byte) (x=0-5) | $028_H$+x* 4 | BE | BE | | |
| P20_PDR0 | Port 20 Pad Driver Mode Register 0 | $040_H$ | U,SV | SV,E,P | See page 45 | 45 |
| P20_PDR1 | Port 20 Pad Driver Mode Register 1 | $044_H$ | U,SV | SV,E,P | See page 50 | 50 |
| | Reserved ($004_H$ Byte) (x=0-1) | $048_H$+x* 4 | BE | BE | | |
| P20_ESR | Port 20 Emergency Stop Register | $050_H$ | U,SV | SV,E,P | Application Reset | 52 |
| | Reserved ($004_H$ Byte) (x=0-2) | $054_H$+x* 4 | BE | BE | | |
| P20_PDISC | Port 20 Pin Function Decision Control Register | $060_H$ | U,SV | SV,E,P | See page 58 | 58 |
| P20_PCSR | Port 20 Pin Controller Select Register | $064_H$ | U,SV | SV,SE | Application Reset | 63 |
| | Reserved ($004_H$ Byte) (x=0-1) | $068_H$+x* 4 | BE | BE | | |
| P20_OMSR0 | Port 20 Output Modification Set Register 0 | $070_H$ | U,SV | U,SV,P | Application Reset | 69 |
| P20_OMSR4 | Port 20 Output Modification Set Register 4 | $074_H$ | U,SV | U,SV,P | Application Reset | 72 |
| P20_OMSR8 | Port 20 Output Modification Set Register 8 | $078_H$ | U,SV | U,SV,P | Application Reset | 74 |
| P20_OMSR12 | Port 20 Output Modification Set Register 12 | $07C_H$ | U,SV | U,SV,P | Application Reset | 75 |
| P20_OMCR0 | Port 20 Output Modification Clear Register 0 | $080_H$ | U,SV | U,SV,P | Application Reset | 76 |
| P20_OMCR4 | Port 20 Output Modification Clear Register 4 | $084_H$ | U,SV | U,SV,P | Application Reset | 78 |
| P20_OMCR8 | Port 20 Output Modification Clear Register 8 | $088_H$ | U,SV | U,SV,P | Application Reset | 81 |
| P20_OMCR12 | Port 20 Output Modification Clear Register 12 | $08C_H$ | U,SV | U,SV,P | Application Reset | 82 |
| P20_OMSR | Port 20 Output Modification Set Register | $090_H$ | U,SV | U,SV,P | Application Reset | 83 |
| P20_OMCR | Port 20 Output Modification Clear Register | $094_H$ | U,SV | U,SV,P | Application Reset | 86 |
| | Reserved ($004_H$ Byte) (x=0-1) | $098_H$+x* 4 | BE | BE | | |

**Table 83 Register Overview - P20 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| | Reserved ($004_H$ Byte) (x=0-13) | $0C0_H$+x*4 | BE | BE | | |
| P20_ACCEN1 | Port 20 Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | 92 |
| P20_ACCEN0 | Port 20 Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | 94 |

**Table 84 Register Overview - P21 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P21_OUT | Port 21 Output Register | $000_H$ | U,SV | U,SV,P | Application Reset | 26 |
| P21_OMR | Port 21 Output Modification Register | $004_H$ | U,SV | U,SV,P | Application Reset | 30 |
| P21_ID | Port 21 Identification Register | $008_H$ | U,SV | BE | Application Reset | 31 |
| | Reserved ($004_H$ Byte) | $00C_H$ | BE | BE | | |
| P21_IOCR0 | Port 21 Input/Output Control Register 0 | $010_H$ | U,SV | U,SV,P | See page 32 | 32 |
| P21_IOCR4 | Port 21 Input/Output Control Register 4 | $014_H$ | U,SV | U,SV,P | See page 36 | 36 |
| | Reserved ($004_H$ Byte) | $020_H$ | BE | BE | | |
| P21_IN | Port 21 Input Register | $024_H$ | U,SV | BE | Application Reset | 44 |
| | Reserved ($004_H$ Byte) (x=0-5) | $028_H$+x*4 | BE | BE | | |
| P21_PDR0 | Port 21 Pad Driver Mode Register 0 | $040_H$ | U,SV | SV,E,P | See page 45 | 45 |
| | Reserved ($004_H$ Byte) (x=0-1) | $048_H$+x*4 | BE | BE | | |
| P21_ESR | Port 21 Emergency Stop Register | $050_H$ | U,SV | SV,E,P | Application Reset | 56 |
| | Reserved ($004_H$ Byte) (x=0-2) | $054_H$+x*4 | BE | BE | | |
| P21_PDISC | Port 21 Pin Function Decision Control Register | $060_H$ | U,SV | SV,E,P | See page 62 | 62 |
| P21_PCSR | Port 21 Pin Controller Select Register | $064_H$ | U,SV | SV,SE | Application Reset | 67 |
| | Reserved ($004_H$ Byte) (x=0-1) | $068_H$+x*4 | BE | BE | | |

**Table 84 Register Overview - P21 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P21_OMSR0 | Port 21 Output Modification Set Register 0 | $070_H$ | U,SV | U,SV,P | Application Reset | 69 |
| P21_OMSR4 | Port 21 Output Modification Set Register 4 | $074_H$ | U,SV | U,SV,P | Application Reset | 72 |
| P21_OMCR0 | Port 21 Output Modification Clear Register 0 | $080_H$ | U,SV | U,SV,P | Application Reset | 76 |
| P21_OMCR4 | Port 21 Output Modification Clear Register 4 | $084_H$ | U,SV | U,SV,P | Application Reset | 78 |
| P21_OMSR | Port 21 Output Modification Set Register | $090_H$ | U,SV | U,SV,P | Application Reset | 86 |
| P21_OMCR | Port 21 Output Modification Clear Register | $094_H$ | U,SV | U,SV,P | Application Reset | 90 |
| | Reserved ($004_H$ Byte) (x=0-1) | $098_H$+x* 4 | BE | BE | | |
| P21_LPCRx | Port 21 LVDS Pad Control Register x | $0A0_H$+x* 4 | U,SV | SV,E,P | See page 90 | 90 |
| | Reserved ($004_H$ Byte) (x=0-13) | $0C0_H$+x* 4 | BE | BE | | |
| P21_ACCEN1 | Port 21 Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | 92 |
| P21_ACCEN0 | Port 21 Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | 94 |

**Table 85 Register Overview - P22 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P22_OUT | Port 22 Output Register | $000_H$ | U,SV | U,SV,P | Application Reset | 26 |
| P22_OMR | Port 22 Output Modification Register | $004_H$ | U,SV | U,SV,P | Application Reset | 30 |
| P22_ID | Port 22 Identification Register | $008_H$ | U,SV | BE | Application Reset | 31 |
| | Reserved ($004_H$ Byte) | $00C_H$ | BE | BE | | |
| P22_IOCR0 | Port 22 Input/Output Control Register 0 | $010_H$ | U,SV | U,SV,P | See page 32 | 32 |
| P22_IOCR4 | Port 22 Input/Output Control Register 4 | $014_H$ | U,SV | U,SV,P | See page 36 | 36 |
| | Reserved ($004_H$ Byte) | $020_H$ | BE | BE | | |
| P22_IN | Port 22 Input Register | $024_H$ | U,SV | BE | Application Reset | 44 |

**Table 85 Register Overview - P22 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| | Reserved ($004_H$ Byte) (x=0-5) | $028_H$+x*4 | BE | BE | | |
| P22_PDR0 | Port 22 Pad Driver Mode Register 0 | $040_H$ | U,SV | SV,E,P | See page 45 | 45 |
| | Reserved ($004_H$ Byte) (x=0-1) | $048_H$+x*4 | BE | BE | | |
| P22_ESR | Port 22 Emergency Stop Register | $050_H$ | U,SV | SV,E,P | Application Reset | 57 |
| | Reserved ($004_H$ Byte) (x=0-2) | $054_H$+x*4 | BE | BE | | |
| P22_PDISC | Port 22 Pin Function Decision Control Register | $060_H$ | U,SV | SV,E,P | See page 62 | 62 |
| P22_PCSR | Port 22 Pin Controller Select Register | $064_H$ | U,SV | SV,SE | Application Reset | 67 |
| | Reserved ($004_H$ Byte) (x=0-1) | $068_H$+x*4 | BE | BE | | |
| P22_OMSR0 | Port 22 Output Modification Set Register 0 | $070_H$ | U,SV | U,SV,P | Application Reset | 69 |
| P22_OMSR4 | Port 22 Output Modification Set Register 4 | $074_H$ | U,SV | U,SV,P | Application Reset | 72 |
| P22_OMCR0 | Port 22 Output Modification Clear Register 0 | $080_H$ | U,SV | U,SV,P | Application Reset | 76 |
| P22_OMCR4 | Port 22 Output Modification Clear Register 4 | $084_H$ | U,SV | U,SV,P | Application Reset | 78 |
| P22_OMSR | Port 22 Output Modification Set Register | $090_H$ | U,SV | U,SV,P | Application Reset | 86 |
| P22_OMCR | Port 22 Output Modification Clear Register | $094_H$ | U,SV | U,SV,P | Application Reset | 90 |
| | Reserved ($004_H$ Byte) (x=0-1) | $098_H$+x*4 | BE | BE | | |
| | Reserved ($004_H$ Byte) (x=0-13) | $0C0_H$+x*4 | BE | BE | | |
| P22_ACCEN1 | Port 22 Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | 92 |
| P22_ACCEN0 | Port 22 Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | 94 |

**Table 86 Register Overview - P23 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P23_OUT | Port 23 Output Register | $000_H$ | U,SV | U,SV,P | Application Reset | 26 |
| P23_OMR | Port 23 Output Modification Register | $004_H$ | U,SV | U,SV,P | Application Reset | 30 |
| P23_ID | Port 23 Identification Register | $008_H$ | U,SV | BE | Application Reset | 32 |
| | Reserved ($004_H$ Byte) | $00C_H$ | BE | BE | | |
| P23_IOCR0 | Port 23 Input/Output Control Register 0 | $010_H$ | U,SV | U,SV,P | See page 35 | 35 |
| P23_IOCR4 | Port 23 Input/Output Control Register 4 | $014_H$ | U,SV | U,SV,P | See page 36 | 36 |
| | Reserved ($004_H$ Byte) | $020_H$ | BE | BE | | |
| P23_IN | Port 23 Input Register | $024_H$ | U,SV | BE | Application Reset | 44 |
| | Reserved ($004_H$ Byte) (x=0-5) | $028_H$+x* 4 | BE | BE | | |
| P23_PDR0 | Port 23 Pad Driver Mode Register 0 | $040_H$ | U,SV | SV,E,P | See page 45 | 45 |
| | Reserved ($004_H$ Byte) (x=0-1) | $048_H$+x* 4 | BE | BE | | |
| P23_ESR | Port 23 Emergency Stop Register | $050_H$ | U,SV | SV,E,P | Application Reset | 57 |
| | Reserved ($004_H$ Byte) (x=0-2) | $054_H$+x* 4 | BE | BE | | |
| P23_PDISC | Port 23 Pin Function Decision Control Register | $060_H$ | U,SV | SV,E,P | See page 62 | 62 |
| P23_PCSR | Port 23 Pin Controller Select Register | $064_H$ | U,SV | SV,SE | Application Reset | 67 |
| | Reserved ($004_H$ Byte) (x=0-1) | $068_H$+x* 4 | BE | BE | | |
| P23_OMSR0 | Port 23 Output Modification Set Register 0 | $070_H$ | U,SV | U,SV,P | Application Reset | 71 |
| P23_OMSR4 | Port 23 Output Modification Set Register 4 | $074_H$ | U,SV | U,SV,P | Application Reset | 72 |
| P23_OMCR0 | Port 23 Output Modification Clear Register 0 | $080_H$ | U,SV | U,SV,P | Application Reset | 77 |
| P23_OMCR4 | Port 23 Output Modification Clear Register 4 | $084_H$ | U,SV | U,SV,P | Application Reset | 78 |
| P23_OMSR | Port 23 Output Modification Set Register | $090_H$ | U,SV | U,SV,P | Application Reset | 86 |

**Table 86 Register Overview - P23 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P23_OMCR | Port 23 Output Modification Clear Register | $094_H$ | U,SV | U,SV,P | Application Reset | 90 |
| | Reserved ($004_H$ Byte) (x=0-1) | $098_H$+x* 4 | BE | BE | | |
| | Reserved ($004_H$ Byte) (x=0-13) | $0C0_H$+x* 4 | BE | BE | | |
| P23_ACCEN1 | Port 23 Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | 93 |
| P23_ACCEN0 | Port 23 Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | 96 |

**Table 87 Register Overview - P32 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P32_OUT | Port 32 Output Register | $000_H$ | U,SV | U,SV,P | Application Reset | 26 |
| P32_OMR | Port 32 Output Modification Register | $004_H$ | U,SV | U,SV,P | Application Reset | 30 |
| P32_ID | Port 32 Identification Register | $008_H$ | U,SV | BE | Application Reset | 32 |
| | Reserved ($004_H$ Byte) | $00C_H$ | BE | BE | | |
| P32_IOCR0 | Port 32 Input/Output Control Register 0 | $010_H$ | U,SV | U,SV,P | See page 35 | 35 |
| P32_IOCR4 | Port 32 Input/Output Control Register 4 | $014_H$ | U,SV | U,SV,P | See page 38 | 38 |
| | Reserved ($004_H$ Byte) | $020_H$ | BE | BE | | |
| P32_IN | Port 32 Input Register | $024_H$ | U,SV | BE | Application Reset | 44 |
| | Reserved ($004_H$ Byte) (x=0-5) | $028_H$+x* 4 | BE | BE | | |
| P32_PDR0 | Port 32 Pad Driver Mode Register 0 | $040_H$ | U,SV | SV,E,P | See page 48 | 48 |
| | Reserved ($004_H$ Byte) (x=0-1) | $048_H$+x* 4 | BE | BE | | |
| P32_ESR | Port 32 Emergency Stop Register | $050_H$ | U,SV | SV,E,P | Application Reset | 57 |
| | Reserved ($004_H$ Byte) (x=0-2) | $054_H$+x* 4 | BE | BE | | |
| P32_PDISC | Port 32 Pin Function Decision Control Register | $060_H$ | U,SV | SV,E,P | See page 62 | 62 |

**Table 87 Register Overview - P32 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P32_PCSR | Port 32 Pin Controller Select Register | $064_H$ | U,SV | SV,SE | Application Reset | 67 |
| | Reserved ($004_H$ Byte) (x=0-1) | $068_H$+x* 4 | BE | BE | | |
| P32_OMSR0 | Port 32 Output Modification Set Register 0 | $070_H$ | U,SV | U,SV,P | Application Reset | 71 |
| P32_OMSR4 | Port 32 Output Modification Set Register 4 | $074_H$ | U,SV | U,SV,P | Application Reset | 73 |
| P32_OMCR0 | Port 32 Output Modification Clear Register 0 | $080_H$ | U,SV | U,SV,P | Application Reset | 77 |
| P32_OMCR4 | Port 32 Output Modification Clear Register 4 | $084_H$ | U,SV | U,SV,P | Application Reset | 80 |
| P32_OMSR | Port 32 Output Modification Set Register | $090_H$ | U,SV | U,SV,P | Application Reset | 86 |
| P32_OMCR | Port 32 Output Modification Clear Register | $094_H$ | U,SV | U,SV,P | Application Reset | 90 |
| | Reserved ($004_H$ Byte) (x=0-1) | $098_H$+x* 4 | BE | BE | | |
| | Reserved ($004_H$ Byte) (x=0-13) | $0C0_H$+x* 4 | BE | BE | | |
| P32_ACCEN1 | Port 32 Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | 93 |
| P32_ACCEN0 | Port 32 Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | 96 |

**Table 88 Register Overview - P33 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P33_OUT | Port 33 Output Register | $000_H$ | U,SV | U,SV,P | Application Reset | 23 |
| P33_OMR | Port 33 Output Modification Register | $004_H$ | U,SV | U,SV,P | Application Reset | 26 |
| P33_ID | Port 33 Identification Register | $008_H$ | U,SV | BE | Application Reset | 32 |
| | Reserved ($004_H$ Byte) | $00C_H$ | BE | BE | | |
| P33_IOCR0 | Port 33 Input/Output Control Register 0 | $010_H$ | U,SV | U,SV,P | See page 35 | 35 |
| P33_IOCR4 | Port 33 Input/Output Control Register 4 | $014_H$ | U,SV | U,SV,P | See page 38 | 38 |

**Table 88 Register Overview - P33 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P33_IOCR8 | Port 33 Input/Output Control Register 8 | $018_H$ | U,SV | U,SV,P | See page 39 | 39 |
| P33_IOCR12 | Port 33 Input/Output Control Register 12 | $01C_H$ | U,SV | U,SV,P | See page 41 | 41 |
| | Reserved ($004_H$ Byte) | $020_H$ | BE | BE | | |
| P33_IN | Port 33 Input Register | $024_H$ | U,SV | BE | Application Reset | 42 |
| | Reserved ($004_H$ Byte) (x=0-5) | $028_H$+x* 4 | BE | BE | | |
| P33_PDR0 | Port 33 Pad Driver Mode Register 0 | $040_H$ | U,SV | SV,E,P | See page 48 | 48 |
| P33_PDR1 | Port 33 Pad Driver Mode Register 1 | $044_H$ | U,SV | SV,E,P | See page 50 | 50 |
| | Reserved ($004_H$ Byte) (x=0-1) | $048_H$+x* 4 | BE | BE | | |
| P33_ESR | Port 33 Emergency Stop Register | $050_H$ | U,SV | SV,E,P | Application Reset | 58 |
| | Reserved ($004_H$ Byte) (x=0-2) | $054_H$+x* 4 | BE | BE | | |
| P33_PDISC | Port 33 Pin Function Decision Control Register | $060_H$ | U,SV | SV,E,P | See page 58 | 58 |
| P33_PCSR | Port 33 Pin Controller Select Register | $064_H$ | U,SV | SV,SE | Application Reset | 68 |
| | Reserved ($004_H$ Byte) (x=0-1) | $068_H$+x* 4 | BE | BE | | |
| P33_OMSR0 | Port 33 Output Modification Set Register 0 | $070_H$ | U,SV | U,SV,P | Application Reset | 71 |
| P33_OMSR4 | Port 33 Output Modification Set Register 4 | $074_H$ | U,SV | U,SV,P | Application Reset | 73 |
| P33_OMSR8 | Port 33 Output Modification Set Register 8 | $078_H$ | U,SV | U,SV,P | Application Reset | 74 |
| P33_OMSR12 | Port 33 Output Modification Set Register 12 | $07C_H$ | U,SV | U,SV,P | Application Reset | 75 |
| P33_OMCR0 | Port 33 Output Modification Clear Register 0 | $080_H$ | U,SV | U,SV,P | Application Reset | 77 |
| P33_OMCR4 | Port 33 Output Modification Clear Register 4 | $084_H$ | U,SV | U,SV,P | Application Reset | 80 |
| P33_OMCR8 | Port 33 Output Modification Clear Register 8 | $088_H$ | U,SV | U,SV,P | Application Reset | 81 |
| P33_OMCR12 | Port 33 Output Modification Clear Register 12 | $08C_H$ | U,SV | U,SV,P | Application Reset | 82 |

**Table 88 Register Overview - P33 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P33_OMSR | Port 33 Output Modification Set Register | $090_H$ | U,SV | U,SV,P | Application Reset | 83 |
| P33_OMCR | Port 33 Output Modification Clear Register | $094_H$ | U,SV | U,SV,P | Application Reset | 86 |
| | Reserved ($004_H$ Byte) (x=0-1) | $098_H$+x* 4 | BE | BE | | |
| | Reserved ($004_H$ Byte) (x=0-13) | $0C0_H$+x* 4 | BE | BE | | |
| P33_ACCEN1 | Port 33 Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | 93 |
| P33_ACCEN0 | Port 33 Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | 96 |

**Table 89 Register Overview - P34 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P34_OUT | Port 34 Output Register | $000_H$ | U,SV | U,SV,P | Application Reset | 25 |
| P34_OMR | Port 34 Output Modification Register | $004_H$ | U,SV | U,SV,P | Application Reset | 29 |
| P34_ID | Port 34 Identification Register | $008_H$ | U,SV | BE | Application Reset | 32 |
| | Reserved ($004_H$ Byte) | $00C_H$ | BE | BE | | |
| P34_IOCR0 | Port 34 Input/Output Control Register 0 | $010_H$ | U,SV | U,SV,P | See page 35 | 35 |
| | Reserved ($004_H$ Byte) | $020_H$ | BE | BE | | |
| P34_IN | Port 34 Input Register | $024_H$ | U,SV | BE | Application Reset | 44 |
| | Reserved ($004_H$ Byte) (x=0-5) | $028_H$+x* 4 | BE | BE | | |
| P34_PDR0 | Port 34 Pad Driver Mode Register 0 | $040_H$ | U,SV | SV,E,P | See page 49 | 49 |
| | Reserved ($004_H$ Byte) (x=0-1) | $048_H$+x* 4 | BE | BE | | |
| P34_ESR | Port 34 Emergency Stop Register | $050_H$ | U,SV | SV,E,P | Application Reset | 55 |
| | Reserved ($004_H$ Byte) (x=0-2) | $054_H$+x* 4 | BE | BE | | |
| P34_PDISC | Port 34 Pin Function Decision Control Register | $060_H$ | U,SV | SV,E,P | See page 61 | 61 |

**Table 89 Register Overview - P34 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| P34_PCSR | Port 34 Pin Controller Select Register | $064_H$ | U,SV | SV,SE | Application Reset | 69 |
| | Reserved ($004_H$ Byte) (x=0-1) | $068_H$+x* 4 | BE | BE | | |
| P34_OMSR0 | Port 34 Output Modification Set Register 0 | $070_H$ | U,SV | U,SV,P | Application Reset | 71 |
| P34_OMCR0 | Port 34 Output Modification Clear Register 0 | $080_H$ | U,SV | U,SV,P | Application Reset | 77 |
| P34_OMSR | Port 34 Output Modification Set Register | $090_H$ | U,SV | U,SV,P | Application Reset | 85 |
| P34_OMCR | Port 34 Output Modification Clear Register | $094_H$ | U,SV | U,SV,P | Application Reset | 89 |
| | Reserved ($004_H$ Byte) (x=0-1) | $098_H$+x* 4 | BE | BE | | |
| | Reserved ($004_H$ Byte) (x=0-13) | $0C0_H$+x* 4 | BE | BE | | |
| P34_ACCEN1 | Port 34 Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | 93 |
| P34_ACCEN0 | Port 34 Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | 96 |

## 14.3 Pn Registers

### 14.3.1 SPB bus slave interface

**Port 00 Output Register**

The port output register determines the value of a GPIO pin when it is selected by Pn_IOCRx as output. Writing a 0 to a Pn_OUT.Px (x = 0-15) bit position delivers a low level at the corresponding output pin. A high level is output when the corresponding bit is written with a 1. Note that the bits of Pn_OUT.Px can be individually set or cleared by writing appropriate values into the port output modification set register Pn_OMSR or port output modification clear register Pn_OMCR, respectively. The Pn_OUT.Px bits can also be set, cleared or toggled with register Pn_OMR within the same write operation.

**P00_OUT**
**Port 00 Output Register** **($000_H$)** **Application Reset Value: 0000 $0000_H$**
**P11_OUT**
**Port 11 Output Register** **($000_H$)** **Application Reset Value: 0000 $0000_H$**
**P20_OUT**
**Port 20 Output Register** **($000_H$)** **Application Reset Value: 0000 $0000_H$**
**P33_OUT**
**Port 33 Output Register** **($000_H$)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **P15** | **P14** | **P13** | **P12** | **P11** | **P10** | **P9** | **P8** | **P7** | **P6** | **P5** | **P4** | **P3** | **P2** | **P1** | **P0** |
| rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **Px (x=0-15)** | x | rwh | **Output Bit x**<br>This bit determines the level at the output pin Pn.x if the output is selected as GPIO output.<br>Pn.x can also be set or cleared by control bits of the Pn_OMSR, Pn_OMCR or Pn_OMR registers.<br>$0_B$ The output level of Pn.x is 0.<br>$1_B$ The output level of Pn.x is 1. |
| **0** | 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 90 Access Mode Restrictions sorted by descending priority**

Applies to **P00_OUT**
Applies to **P11_OUT**
Applies to **P20_OUT**
Applies to **P33_OUT**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | rwh | Px (x=0-15) | write access for enabled masters |
| Otherwise (default) | rh | Px (x=0-15) | |

**P02_OUT**
**Port 02 Output Register** **($000_H$)** **Application Reset Value: 0000 $0000_H$**
**P10_OUT**
**Port 10 Output Register** **($000_H$)** **Application Reset Value: 0000 $0000_H$**
**P14_OUT**
**Port 14 Output Register** **($000_H$)** **Application Reset Value: 0000 $0000_H$**
**P15_OUT**
**Port 15 Output Register** **($000_H$)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **P11** | **P10** | **P9** | **P8** | **P7** | **P6** | **P5** | **P4** | **P3** | **P2** | **P1** | **P0** |
| r | r | r | r | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **Px (x=0-11)** | x | rwh | **Output Bit x**<br>This bit determines the level at the output pin Pn.x if the output is selected as GPIO output.<br>Pn.x can also be set or cleared by control bits of the Pn_OMSR, Pn_OMCR or Pn_OMR registers.<br>$0_B$ The output level of Pn.x is 0.<br>$1_B$ The output level of Pn.x is 1. |
| **0** | 15, 14, 13, 12, 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 91 Access Mode Restrictions sorted by descending priority**

Applies to **P02_OUT**
Applies to **P10_OUT**
Applies to **P14_OUT**
Applies to **P15_OUT**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | rwh | Px (x=0-11) | write access for enabled masters |
| Otherwise (default) | rh | Px (x=0-11) | |

**P12_OUT**
**Port 12 Output Register** **($000_H$)** **Application Reset Value: 0000 $0000_H$**
**P34_OUT**
**Port 34 Output Register** **($000_H$)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | P3 | P2 | P1 | P0 |
| r | r | r | r | r | r | r | r | r | r | r | r | rwh | rwh | rwh | rwh |

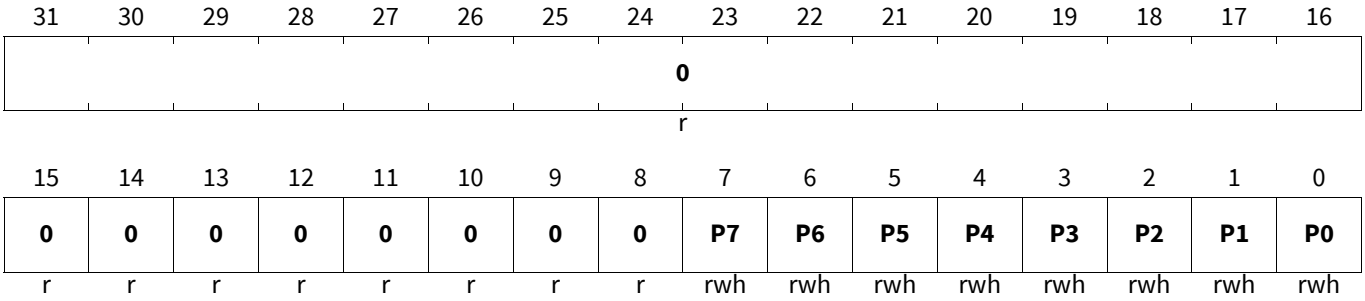
| Field | Bits | Type | Description |
|---|---|---|---|
| **Px (x=0-3)** | x | rwh | **Output Bit x**<br>This bit determines the level at the output pin Pn.x if the output is selected as GPIO output.<br>Pn.x can also be set or cleared by control bits of the Pn_OMSR, Pn_OMCR or Pn_OMR registers.<br>$0_B$ The output level of Pn.x is 0.<br>$1_B$ The output level of Pn.x is 1. |
| **0** | 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 92 Access Mode Restrictions sorted by descending priority**

Applies to **P12_OUT**
Applies to **P34_OUT**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | rwh | Px (x=0-3) | write access for enabled masters |
| Otherwise (default) | rh | Px (x=0-3) | |

**P21_OUT**
**Port 21 Output Register** **($000_H$)** **Application Reset Value: $0000\ 0000_H$**
**P22_OUT**
**Port 22 Output Register** **($000_H$)** **Application Reset Value: $0000\ 0000_H$**
**P23_OUT**
**Port 23 Output Register** **($000_H$)** **Application Reset Value: $0000\ 0000_H$**
**P32_OUT**
**Port 32 Output Register** **($000_H$)** **Application Reset Value: $0000\ 0000_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **Px (x=0-7)** | x | rwh | **Output Bit x**<br>This bit determines the level at the output pin Pn.x if the output is selected as GPIO output.<br>Pn.x can also be set or cleared by control bits of the Pn_OMSR, Pn_OMCR or Pn_OMR registers.<br>$0_B$ The output level of Pn.x is 0.<br>$1_B$ The output level of Pn.x is 1. |
| **0** | 15, 14, 13, 12, 11, 10, 9, 8, 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 93 Access Mode Restrictions sorted by descending priority**

Applies to **P21_OUT**
Applies to **P22_OUT**
Applies to **P23_OUT**
Applies to **P32_OUT**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | rwh | Px (x=0-7) | write access for enabled masters |
| Otherwise (default) | rh | Px (x=0-7) | |

**Port 00 Output Modification Register**

The port output modification register contains control bits that make it possible to individually set, clear or toggle the logic state of a single port line by manipulating the output register.

**P00_OMR**
**Port 00 Output Modification Register** **($004_H$)** **Application Reset Value: 0000 0000$_H$**
**P11_OMR**
**Port 11 Output Modification Register** **($004_H$)** **Application Reset Value: 0000 0000$_H$**
**P20_OMR**
**Port 20 Output Modification Register** **($004_H$)** **Application Reset Value: 0000 0000$_H$**
**P33_OMR**
**Port 33 Output Modification Register** **($004_H$)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **PCL15** | **PCL14** | **PCL13** | **PCL12** | **PCL11** | **PCL10** | **PCL9** | **PCL8** | **PCL7** | **PCL6** | **PCL5** | **PCL4** | **PCL3** | **PCL2** | **PCL1** | **PCL0** |
| w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **PS15** | **PS14** | **PS13** | **PS12** | **PS11** | **PS10** | **PS9** | **PS8** | **PS7** | **PS6** | **PS5** | **PS4** | **PS3** | **PS2** | **PS1** | **PS0** |
| w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PSx (x=0-15)** | x | w0 | **Set Bit x**<br>Setting this bit will set or toggle the corresponding bit in the port output register Pn_OUT. Read as 0. The function of this bit is shown in **Table 95**.<br>$0_B$ No operation<br>$1_B$ Sets or toggles Pn_OUT.Px. |
| **PCLx (x=0-15)** | x+16 | w0 | **Clear Bit x**<br>Setting this bit will clear or toggle the corresponding bit in the port output register Pn_OUT. Read as 0. The function of this bit is shown in **Table 95**.<br>$0_B$ No operation<br>$1_B$ Clears or toggles Pn_OUT.Px. |

**Table 94 Access Mode Restrictions sorted by descending priority**

Applies to **P00_OMR**
Applies to **P11_OMR**
Applies to **P20_OMR**
Applies to **P33_OMR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PCLx (x=0-15), PSx (x=0-15) | write access for enabled masters |
| Otherwise (default) | r0 | PCLx (x=0-15), PSx (x=0-15) | |

*Note: Register Pn_OMR is virtual and does not contain any flip-flop. A read action delivers the value of 0. One 8 or 16-bits write behaves as a 32-bit write padded with zeros.*

**Table 95 Function of the Bits PCLx and PSx**

| PCLx | PSx | Function |
|---|---|---|
| 0 | 0 | Bit Pn_OUT.Px is not changed. |
| 0 | 1 | Bit Pn_OUT.Px is set. |
| 1 | 0 | Bit Pn_OUT.Px is reset. |
| 1 | 1 | Bit Pn_OUT.Px is toggled. |

**P02_OMR**
**Port 02 Output Modification Register** **($004_H$)** **Application Reset Value: $0000\ 0000_H$**
**P10_OMR**
**Port 10 Output Modification Register** **($004_H$)** **Application Reset Value: $0000\ 0000_H$**
**P14_OMR**
**Port 14 Output Modification Register** **($004_H$)** **Application Reset Value: $0000\ 0000_H$**
**P15_OMR**
**Port 15 Output Modification Register** **($004_H$)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **PCL11** | **PCL10** | **PCL9** | **PCL8** | **PCL7** | **PCL6** | **PCL5** | **PCL4** | **PCL3** | **PCL2** | **PCL1** | **PCL0** |
| r | r | r | r | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **PS11** | **PS10** | **PS9** | **PS8** | **PS7** | **PS6** | **PS5** | **PS4** | **PS3** | **PS2** | **PS1** | **PS0** |
| r | r | r | r | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PSx (x=0-11)** | x | w0 | **Set Bit x**<br>Setting this bit will set or toggle the corresponding bit in the port output register Pn_OUT. Read as 0. The function of this bit is shown in **Table 95**.<br>$0_B$ No operation<br>$1_B$ Sets or toggles Pn_OUT.Px. |
| **PCLx (x=0-11)** | x+16 | w0 | **Clear Bit x**<br>Setting this bit will clear or toggle the corresponding bit in the port output register Pn_OUT. Read as 0. The function of this bit is shown in **Table 95**.<br>$0_B$ No operation<br>$1_B$ Clears or toggles Pn_OUT.Px. |
| **0** | 15, 14, 13, 12,<br>31, 30, 29, 28 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 96 Access Mode Restrictions sorted by descending priority**

Applies to **P02_OMR**
Applies to **P10_OMR**
Applies to **P14_OMR**
Applies to **P15_OMR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PCLx (x=0-11), PSx (x=0-11) | write access for enabled masters |
| Otherwise (default) | r0 | PCLx (x=0-11), PSx (x=0-11) | |

**P12_OMR**
**Port 12 Output Modification Register** **($004_H$)** **Application Reset Value: 0000 $0000_H$**
**P34_OMR**
**Port 34 Output Modification Register** **($004_H$)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **PCL3** | **PCL2** | **PCL1** | **PCL0** |
| r | r | r | r | r | r | r | r | r | r | r | r | w0 | w0 | w0 | w0 |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **PS3** | **PS2** | **PS1** | **PS0** |
| r | r | r | r | r | r | r | r | r | r | r | r | w0 | w0 | w0 | w0 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PSx (x=0-3)** | x | w0 | **Set Bit x**<br>Setting this bit will set or toggle the corresponding bit in the port output register Pn_OUT. Read as 0. The function of this bit is shown in **Table 95**.<br>$0_B$ No operation<br>$1_B$ Sets or toggles Pn_OUT.Px. |
| **PCLx (x=0-3)** | x+16 | w0 | **Clear Bit x**<br>Setting this bit will clear or toggle the corresponding bit in the port output register Pn_OUT. Read as 0. The function of this bit is shown in **Table 95**.<br>$0_B$ No operation<br>$1_B$ Clears or toggles Pn_OUT.Px. |
| **0** | 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 97 Access Mode Restrictions sorted by descending priority**

Applies to **P12_OMR**
Applies to **P34_OMR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PCLx (x=0-3), PSx (x=0-3) | write access for enabled masters |
| Otherwise (default) | r0 | PCLx (x=0-3), PSx (x=0-3) | |

**P21_OMR**
**Port 21 Output Modification Register** **($004_H$)** **Application Reset Value: $0000\ 0000_H$**
**P22_OMR**
**Port 22 Output Modification Register** **($004_H$)** **Application Reset Value: $0000\ 0000_H$**
**P23_OMR**
**Port 23 Output Modification Register** **($004_H$)** **Application Reset Value: $0000\ 0000_H$**
**P32_OMR**
**Port 32 Output Modification Register** **($004_H$)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **PCL7** | **PCL6** | **PCL5** | **PCL4** | **PCL3** | **PCL2** | **PCL1** | **PCL0** |
| r | r | r | r | r | r | r | r | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **PS7** | **PS6** | **PS5** | **PS4** | **PS3** | **PS2** | **PS1** | **PS0** |
| r | r | r | r | r | r | r | r | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PSx (x=0-7)** | x | w0 | **Set Bit x**<br>Setting this bit will set or toggle the corresponding bit in the port output register Pn_OUT. Read as 0. The function of this bit is shown in **Table 95**.<br>$0_B$ No operation<br>$1_B$ Sets or toggles Pn_OUT.Px. |
| **PCLx (x=0-7)** | x+16 | w0 | **Clear Bit x**<br>Setting this bit will clear or toggle the corresponding bit in the port output register Pn_OUT. Read as 0. The function of this bit is shown in **Table 95**.<br>$0_B$ No operation<br>$1_B$ Clears or toggles Pn_OUT.Px. |
| **0** | 15, 14, 13, 12, 11, 10, 9, 8,<br>31, 30, 29, 28, 27, 26, 25, 24 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 98 Access Mode Restrictions sorted by descending priority**

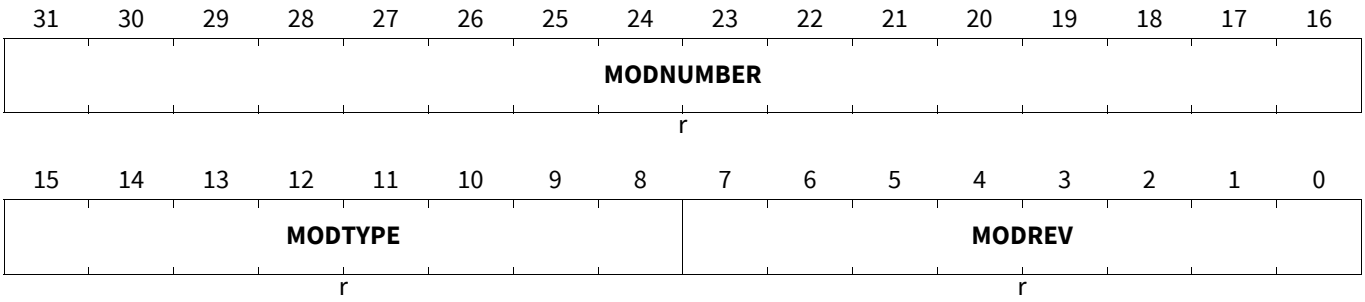
Applies to **P21_OMR**
Applies to **P22_OMR**
Applies to **P23_OMR**
Applies to **P32_OMR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PCLx (x=0-7), PSx (x=0-7) | write access for enabled masters |
| Otherwise (default) | r0 | PCLx (x=0-7), PSx (x=0-7) | |

**Port 00 Identification Register**

The module Identification Register ID contains read-only information about the module version.

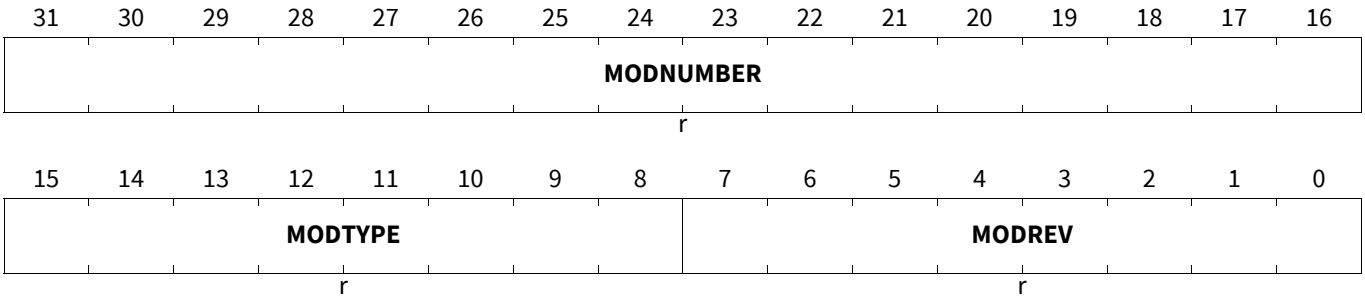
**P00_ID**
**Port 00 Identification Register (008$_H$) Application Reset Value: 00C8 C0XX$_H$**
**P02_ID**
**Port 02 Identification Register (008$_H$) Application Reset Value: 00C8 C0XX$_H$**
**P10_ID**
**Port 10 Identification Register (008$_H$) Application Reset Value: 00C8 C0XX$_H$**
**P11_ID**
**Port 11 Identification Register (008$_H$) Application Reset Value: 00C8 C0XX$_H$**
**P12_ID**
**Port 12 Identification Register (008$_H$) Application Reset Value: 00C8 C0XX$_H$**
**P14_ID**
**Port 14 Identification Register (008$_H$) Application Reset Value: 00C8 C0XX$_H$**
**P15_ID**
**Port 15 Identification Register (008$_H$) Application Reset Value: 00C8 C0XX$_H$**
**P20_ID**
**Port 20 Identification Register (008$_H$) Application Reset Value: 00C8 C0XX$_H$**
**P21_ID**
**Port 21 Identification Register (008$_H$) Application Reset Value: 00C8 C0XX$_H$**
**P22_ID**
**Port 22 Identification Register (008$_H$) Application Reset Value: 00C8 C0XX$_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **MODREV** | 7:0 | r | **Module Revision Number**<br>This bit field indicates the revision number of the TC35x module ($01_H$ = first revision). |

| Field | Bits | Type | Description |
|---|---|---|---|
| **MODTYPE** | 15:8 | r | **Module Type**<br>This bit field is $C0_H$. It defines a 32-bit module |
| **MODNUMBER** | 31:16 | r | **Module Number**<br>This bit field defines the module identification number. The value for the Ports module is $00C8_H$ |

**P23_ID**
**Port 23 Identification Register** **($008_H$)** **Application Reset Value: 00C8 C0XX$_H$**
**P32_ID**
**Port 32 Identification Register** **($008_H$)** **Application Reset Value: 00C8 C0XX$_H$**
**P33_ID**
**Port 33 Identification Register** **($008_H$)** **Application Reset Value: 00C8 C0XX$_H$**
**P34_ID**
**Port 34 Identification Register** **($008_H$)** **Application Reset Value: 00C8 C0XX$_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **MODREV** | 7:0 | r | **Module Revision Number**<br>This bit field indicates the revision number of the TC35x module ($01_H$ = first revision). |
| **MODTYPE** | 15:8 | r | **Module Type**<br>This bit field is $C0_H$. It defines a 32-bit module |
| **MODNUMBER** | 31:16 | r | **Module Number**<br>This bit field defines the module identification number. The value for the Ports module is $00C8_H$ |

### Port 00 Input/Output Control Register 0

The port input/output control registers select the digital output and input driver functionality and characteristics of a GPIO port pin. Port direction (input or output), pull-up, pull-down, or no pull devices for inputs, and push-pull or open-drain functionality for outputs can be selected by the corresponding bit fields PCx (x = 0-15). Each 32-bit wide port input/output control register controls four GPIO port lines:

Register Pn_IOCR0 controls the Pn.[3:0] port lines
Register Pn_IOCR4 controls the Pn.[7:4] port lines
Register Pn_IOCR8 controls the Pn.[11:8] port lines
Register Pn_IOCR12 controls the Pn.[15:12] port lines

The structure with one control bit field for each port pin located in different register bytes offers the possibility to configure the port pin functionality of a single pin with byte-oriented accesses without accessing the other PCx bit fields.

The reset values of 1010 $1010_H$ and 0000 $0000_H$ for Pn_IOCRx registers represents input pull-up and no input pull device (tri-state mode) being activated, respectively. The switching of the intended mode of the device is controlled by HWCFG6.When a cold reset is activated and HWCFG6=1, the port pins except P33.8, P40 and P41 are set to input pull-up mode, P33.8, P40 and P41 are in tri-state mode as long as PORST is activated.If HWCFG6=0, the pins have the default state of tri-state mode. The pad state can also be configured by software through PMSWCR5.TRISTREQ bit. In the event of a warm reset or wake-up from standby mode, PMSWCR5.TRISTREQ is not affected by reset, hence Pn_IOCRx registers have the reset values configured as per the last state of the TRISTREQ bit.

*Note: In LVDS (RX and TX) operation the IOCR register of both pins of the LVDS pair must be configured as output, i.e. $1xxxx_B$. This ensures that the pull devices are disconnected and don't interfere with LVDS operation.*

Register Pn_IOCR0 controls the Pn.[3:0] port lines

**P00_IOCR0**
**Port 00 Input/Output Control Register 0** **($010_H$)** **Reset Value: Table 100**
**P02_IOCR0**
**Port 02 Input/Output Control Register 0** **($010_H$)** **Reset Value: Table 100**
**P10_IOCR0**
**Port 10 Input/Output Control Register 0** **($010_H$)** **Reset Value: Table 100**
**P11_IOCR0**
**Port 11 Input/Output Control Register 0** **($010_H$)** **Reset Value: Table 100**
**P12_IOCR0**
**Port 12 Input/Output Control Register 0** **($010_H$)** **Reset Value: Table 100**
**P14_IOCR0**
**Port 14 Input/Output Control Register 0** **($010_H$)** **Reset Value: Table 100**
**P15_IOCR0**
**Port 15 Input/Output Control Register 0** **($010_H$)** **Reset Value: Table 100**
**P20_IOCR0**
**Port 20 Input/Output Control Register 0** **($010_H$)** **Reset Value: Table 100**
**P21_IOCR0**
**Port 21 Input/Output Control Register 0** **($010_H$)** **Reset Value: Table 100**
**P22_IOCR0**
**Port 22 Input/Output Control Register 0** **($010_H$)** **Reset Value: Table 100**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PC3 | | | | | 0 | | | PC2 | | | | | 0 | | |
| rw | | | | | r | | | rw | | | | | r | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PC1 | | | | | 0 | | | PC0 | | | | | 0 | | |
| rw | | | | | r | | | rw | | | | | r | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PCx (x=0-3)** | 8*x+7:8*x+3 | rw | **Port Control for Pin x**<br>This bit field defines the Port n line x functionality according to **Table 101**. |

| Field | Bits | Type | Description |
|---|---|---|---|
| **0** | 26:24, 18:16, 10:8, 2:0 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 99 Access Mode Restrictions sorted by descending priority**

Applies to **P00_IOCR0**
Applies to **P02_IOCR0**
Applies to **P10_IOCR0**
Applies to **P11_IOCR0**
Applies to **P12_IOCR0**
Applies to **P14_IOCR0**
Applies to **P15_IOCR0**
Applies to **P20_IOCR0**
Applies to **P21_IOCR0**
Applies to **P22_IOCR0**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | rw | PCx (x=0-3) | write access for enabled masters |
| Otherwise (default) | r | PCx (x=0-3) | |

**Table 100 Reset Values**

Applies to **P00_IOCR0**
Applies to **P02_IOCR0**
Applies to **P10_IOCR0**
Applies to **P11_IOCR0**
Applies to **P12_IOCR0**
Applies to **P14_IOCR0**
Applies to **P15_IOCR0**
Applies to **P20_IOCR0**
Applies to **P21_IOCR0**
Applies to **P22_IOCR0**

| Reset Type | Reset Value | Note |
|---|---|---|
| Application Reset | $0000\ 0000_H$ | HWCFG6 is 0 (tri-state mode) |
| Application Reset | $1010\ 1010_H$ | HWCFG6 is 1 (input pull-up mode) |

### Port Control Coding

**Table 101** describes the coding of the PCx bit fields that determine the port line functionality.

**Table 101 PCx Coding**

| PCx[4:0] | I/O | Characteristics | Selected Pull-up / Pull-down / Selected Output Function |
|---|---|---|---|
| $0XX00_B$ | Input | – | No input pull device connected, tri-state mode |
| $0XX01_B$ | | | Input pull-down device connected |
| $0XX10_B$ | | | Input pull-up device connected[1)] |
| $0XX11_B$ | | | No input pull device connected, tri-state mode |
| $10000_B$ | Output | Push-pull | General-purpose output |
| $10001_B$ | | | Alternate output function 1 |
| $10010_B$ | | | Alternate output function 2 |
| $10011_B$ | | | Alternate output function 3 |
| $10100_B$ | | | Alternate output function 4 |
| $10101_B$ | | | Alternate output function 5 |
| $10110_B$ | | | Alternate output function 6 |
| $10111_B$ | | | Alternate output function 7 |
| $11000_B$ | | Open-drain | General-purpose output |
| $11001_B$ | | | Alternate output function 1 |
| $11010_B$ | | | Alternate output function 2 |
| $11011_B$ | | | Alternate output function 3 |
| $11100_B$ | | | Alternate output function 4 |
| $11101_B$ | | | Alternate output function 5 |
| $11110_B$ | | | Alternate output function 6 |
| $11111_B$ | | | Alternate output function 7 |

1) This is the default pull device setting after reset for powertrain applications.

**P23_IOCR0**
**Port 23 Input/Output Control Register 0** **($010_H$)** **Reset Value: Table 103**
**P32_IOCR0**
**Port 32 Input/Output Control Register 0** **($010_H$)** **Reset Value: Table 103**
**P33_IOCR0**
**Port 33 Input/Output Control Register 0** **($010_H$)** **Reset Value: Table 103**
**P34_IOCR0**
**Port 34 Input/Output Control Register 0** **($010_H$)** **Reset Value: Table 103**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PC3 | | | | | 0 | | | PC2 | | | | | 0 | | |
| rw | | | | | r | | | rw | | | | | r | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PC1 | | | | | 0 | | | PC0 | | | | | 0 | | |
| rw | | | | | r | | | rw | | | | | r | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PCx (x=0-3)** | 8*x+7:8*x+3 | rw | **Port Control for Pin x**<br>This bit field defines the Port n line x functionality according to **Table 101**. |
| **0** | 26:24, 18:16, 10:8, 2:0 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 102 Access Mode Restrictions sorted by descending priority**

Applies to **P23_IOCR0**
Applies to **P32_IOCR0**
Applies to **P33_IOCR0**
Applies to **P34_IOCR0**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | rw | PCx (x=0-3) | write access for enabled masters |
| Otherwise (default) | r | PCx (x=0-3) | |

**Table 103 Reset Values**

Applies to **P23_IOCR0**
Applies to **P32_IOCR0**
Applies to **P33_IOCR0**
Applies to **P34_IOCR0**

| Reset Type | Reset Value | Note |
|---|---|---|
| Application Reset | $0000\ 0000_H$ | HWCFG6 is 0 (tri-state mode) |
| Application Reset | $1010\ 1010_H$ | HWCFG6 is 1 (input pull-up mode) |

**Port 00 Input/Output Control Register 4**

Register Pn_IOCR4 controls the Pn.[7:4] port lines

**P00_IOCR4**
**Port 00 Input/Output Control Register 4** **($014_H$)** **Reset Value: Table 105**
**P02_IOCR4**
**Port 02 Input/Output Control Register 4** **($014_H$)** **Reset Value: Table 105**
**P10_IOCR4**
**Port 10 Input/Output Control Register 4** **($014_H$)** **Reset Value: Table 105**
**P11_IOCR4**
**Port 11 Input/Output Control Register 4** **($014_H$)** **Reset Value: Table 105**
**P14_IOCR4**
**Port 14 Input/Output Control Register 4** **($014_H$)** **Reset Value: Table 105**
**P15_IOCR4**
**Port 15 Input/Output Control Register 4** **($014_H$)** **Reset Value: Table 105**
**P20_IOCR4**
**Port 20 Input/Output Control Register 4** **($014_H$)** **Reset Value: Table 105**
**P21_IOCR4**
**Port 21 Input/Output Control Register 4** **($014_H$)** **Reset Value: Table 105**
**P22_IOCR4**
**Port 22 Input/Output Control Register 4** **($014_H$)** **Reset Value: Table 105**
**P23_IOCR4**
**Port 23 Input/Output Control Register 4** **($014_H$)** **Reset Value: Table 105**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **PC7** | | | | | **0** | | | **PC6** | | | | | **0** | | |
| rw | | | | | r | | | rw | | | | | r | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **PC5** | | | | | **0** | | | **PC4** | | | | | **0** | | |
| rw | | | | | r | | | rw | | | | | r | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PCx (x=4-7)** | 8*x-25:8*x-29 | rw | **Port Control for Port 00 Pin x**<br>This bit field defines the Port n line x functionality according to Table 101. |
| **0** | 26:24, 18:16, 10:8, 2:0 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 104 Access Mode Restrictions sorted by descending priority**

Applies to **P00_IOCR4**
Applies to **P02_IOCR4**
Applies to **P10_IOCR4**
Applies to **P11_IOCR4**
Applies to **P14_IOCR4**
Applies to **P15_IOCR4**
Applies to **P20_IOCR4**
Applies to **P21_IOCR4**
Applies to **P22_IOCR4**
Applies to **P23_IOCR4**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | rw | PCx (x=4-7) | write access for enabled masters |
| Otherwise (default) | r | PCx (x=4-7) | |

**Table 105 Reset Values**

Applies to **P00_IOCR4**
Applies to **P02_IOCR4**
Applies to **P10_IOCR4**
Applies to **P11_IOCR4**
Applies to **P14_IOCR4**
Applies to **P15_IOCR4**
Applies to **P20_IOCR4**
Applies to **P21_IOCR4**
Applies to **P22_IOCR4**
Applies to **P23_IOCR4**

| Reset Type | Reset Value | Note |
|---|---|---|
| Application Reset | $0000\ 0000_H$ | HWCFG6 is 0 (tri-state mode) |
| Application Reset | $1010\ 1010_H$ | HWCFG6 is 1 (input pull-up mode) |

**P32_IOCR4**
**Port 32 Input/Output Control Register 4** **($014_H$)** **Reset Value: Table 107**
**P33_IOCR4**
**Port 33 Input/Output Control Register 4** **($014_H$)** **Reset Value: Table 107**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PC7 | | | | | 0 | | | PC6 | | | | | 0 | | |
| rw | | | | | r | | | rw | | | | | r | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PC5 | | | | | 0 | | | PC4 | | | | | 0 | | |
| rw | | | | | r | | | rw | | | | | r | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PCx (x=4-7)** | 8*x-25:8*x-29 | rw | **Port Control for Port 32 Pin x**<br>This bit field defines the Port n line x functionality according to **Table 101**. |
| **0** | 26:24, 18:16, 10:8, 2:0 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 106 Access Mode Restrictions sorted by descending priority**

Applies to **P32_IOCR4**
Applies to **P33_IOCR4**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | rw | PCx (x=4-7) | write access for enabled masters |
| Otherwise (default) | r | PCx (x=4-7) | |

**Table 107 Reset Values**

Applies to **P32_IOCR4**
Applies to **P33_IOCR4**

| Reset Type | Reset Value | Note |
|---|---|---|
| Application Reset | $0000\ 0000_H$ | HWCFG6 is 0 (tri-state mode) |
| Application Reset | $1010\ 1010_H$ | HWCFG6 is 1 (input pull-up mode) |

**Port 00 Input/Output Control Register 8**

Register Pn_IOCR8 controls the Pn.[11:8] port lines

**P00_IOCR8**
**Port 00 Input/Output Control Register 8** **($018_H$)** **Reset Value: Table 109**
**P02_IOCR8**
**Port 02 Input/Output Control Register 8** **($018_H$)** **Reset Value: Table 109**
**P10_IOCR8**
**Port 10 Input/Output Control Register 8** **($018_H$)** **Reset Value: Table 109**
**P11_IOCR8**
**Port 11 Input/Output Control Register 8** **($018_H$)** **Reset Value: Table 109**
**P14_IOCR8**
**Port 14 Input/Output Control Register 8** **($018_H$)** **Reset Value: Table 109**
**P15_IOCR8**
**Port 15 Input/Output Control Register 8** **($018_H$)** **Reset Value: Table 109**
**P20_IOCR8**
**Port 20 Input/Output Control Register 8** **($018_H$)** **Reset Value: Table 109**
**P33_IOCR8**
**Port 33 Input/Output Control Register 8** **($018_H$)** **Reset Value: Table 110**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PC11 | | | | | 0 | | | PC10 | | | | | 0 | | |
| rw | | | | | r | | | rw | | | | | r | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PC9 | | | | | 0 | | | PC8 | | | | | 0 | | |
| rw | | | | | r | | | rw | | | | | r | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PCx (x=8-11)** | 8*x-57:8*x-61 | rw | **Port Control for Port 00 Pin x**<br>This bit field defines the Port n line x functionality according to **Table 101**. |
| **0** | 26:24, 18:16, 10:8, 2:0 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 108 Access Mode Restrictions sorted by descending priority**

Applies to **P00_IOCR8**
Applies to **P02_IOCR8**
Applies to **P10_IOCR8**
Applies to **P11_IOCR8**
Applies to **P14_IOCR8**
Applies to **P15_IOCR8**
Applies to **P20_IOCR8**
Applies to **P33_IOCR8**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | rw | PCx (x=8-11) | write access for enabled masters |
| Otherwise (default) | r | PCx (x=8-11) | |

**Table 109 Reset Values variant 1**

Applies to **P00_IOCR8**
Applies to **P02_IOCR8**
Applies to **P10_IOCR8**
Applies to **P11_IOCR8**
Applies to **P14_IOCR8**
Applies to **P15_IOCR8**
Applies to **P20_IOCR8**

| Reset Type | Reset Value | Note |
|---|---|---|
| Application Reset | 0000 0000$_H$ | HWCFG6 is 0 (tri-state mode) |
| Application Reset | 1010 1010$_H$ | HWCFG6 is 1 (input pull-up mode) |

**Table 110 Reset Values of P33_IOCR8**

| Reset Type | Reset Value | Note |
|---|---|---|
| Application Reset | 0000 0000$_H$ | HWCFG6 is 0 (tri-state mode) |
| Application Reset | 1010 1000$_H$ | HWCFG6 is 1 (input pull-up mode) |

**Port 00 Input/Output Control Register 12**

Register Pn_IOCR12 controls the Pn.[15:12] port lines

**P00_IOCR12**
**Port 00 Input/Output Control Register 12 (01C$_H$) Reset Value: Table 112**
**P11_IOCR12**
**Port 11 Input/Output Control Register 12 (01C$_H$) Reset Value: Table 112**
**P20_IOCR12**
**Port 20 Input/Output Control Register 12 (01C$_H$) Reset Value: Table 112**
**P33_IOCR12**
**Port 33 Input/Output Control Register 12 (01C$_H$) Reset Value: Table 112**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PC15 | | | | | 0 | | | PC14 | | | | | 0 | | |
| rw | | | | | r | | | rw | | | | | r | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PC13 | | | | | 0 | | | PC12 | | | | | 0 | | |
| rw | | | | | r | | | rw | | | | | r | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PCx (x=12-15)** | 8*x-89:8*x-93 | rw | **Port Control for Port 00 Pin x**<br>This bit field defines the Port n line x functionality according to **Table 101**. |
| **0** | 26:24, 18:16, 10:8, 2:0 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 111 Access Mode Restrictions sorted by descending priority**

Applies to **P00_IOCR12**
Applies to **P11_IOCR12**
Applies to **P20_IOCR12**
Applies to **P33_IOCR12**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | rw | PCx (x=12-15) | write access for enabled masters |
| Otherwise (default) | r | PCx (x=12-15) | |

**Table 112 Reset Values**

Applies to **P00_IOCR12**
Applies to **P11_IOCR12**
Applies to **P20_IOCR12**
Applies to **P33_IOCR12**

| Reset Type | Reset Value | Note |
|---|---|---|
| Application Reset | $0000\ 0000_H$ | HWCFG6 is 0 (tri-state mode) |
| Application Reset | $1010\ 1010_H$ | HWCFG6 is 1 (input pull-up mode) |

### Port 00 Input Register

The logic level of a GPIO pin can be read via the read-only port input register Pn_IN.Reading the Pn_IN register always returns the current logical value at the GPIO pin independently whether the pin is selected as input or output.

**P00_IN**
**Port 00 Input Register** **($024_H$)** **Application Reset Value: 0000 XXXX$_H$**
**P11_IN**
**Port 11 Input Register** **($024_H$)** **Application Reset Value: 0000 XXXX$_H$**
**P20_IN**
**Port 20 Input Register** **($024_H$)** **Application Reset Value: 0000 XXXX$_H$**
**P33_IN**
**Port 33 Input Register** **($024_H$)** **Application Reset Value: 0000 XXXX$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| P15 | P14 | P13 | P12 | P11 | P10 | P9 | P8 | P7 | P6 | P5 | P4 | P3 | P2 | P1 | P0 |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **Px (x=0-15)** | x | rh | **Input Bit x**<br>This bit indicates the level at the input pin Pn.x.<br>$0_B$ The input level of Pn.x is 0.<br>$1_B$ The input level of Pn.x is 1. |
| **0** | 31:16 | r | **Reserved**<br>Read as 0. |

**P02_IN**
**Port 02 Input Register** **($024_H$)** **Application Reset Value: 0000 0XXX$_H$**
**P10_IN**
**Port 10 Input Register** **($024_H$)** **Application Reset Value: 0000 0XXX$_H$**
**P14_IN**
**Port 14 Input Register** **($024_H$)** **Application Reset Value: 0000 0XXX$_H$**
**P15_IN**
**Port 15 Input Register** **($024_H$)** **Application Reset Value: 0000 0XXX$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **P11** | **P10** | **P9** | **P8** | **P7** | **P6** | **P5** | **P4** | **P3** | **P2** | **P1** | **P0** |
| r | r | r | r | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **Px (x=0-11)** | x | rh | **Input Bit x**<br>This bit indicates the level at the input pin Pn.x.<br>$0_B$ The input level of Pn.x is 0.<br>$1_B$ The input level of Pn.x is 1. |
| **0** | 15, 14, 13, 12,<br>31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**P12_IN**
**Port 12 Input Register** **($024_H$)** **Application Reset Value: 0000 000X$_H$**
**P34_IN**
**Port 34 Input Register** **($024_H$)** **Application Reset Value: 0000 000X$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | P3 | P2 | P1 | P0 |
| r | r | r | r | r | r | r | r | r | r | r | r | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **Px (x=0-3)** | x | rh | **Input Bit x**<br>This bit indicates the level at the input pin Pn.x.<br>$0_B$ The input level of Pn.x is 0.<br>$1_B$ The input level of Pn.x is 1. |
| **0** | 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**P21_IN**
**Port 21 Input Register** **($024_H$)** **Application Reset Value: 0000 00XX$_H$**
**P22_IN**
**Port 22 Input Register** **($024_H$)** **Application Reset Value: 0000 00XX$_H$**
**P23_IN**
**Port 23 Input Register** **($024_H$)** **Application Reset Value: 0000 00XX$_H$**
**P32_IN**
**Port 32 Input Register** **($024_H$)** **Application Reset Value: 0000 00XX$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | P7 | P6 | P5 | P4 | P3 | P2 | P1 | P0 |
| r | r | r | r | r | r | r | r | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **Px (x=0-7)** | x | rh | **Input Bit x**<br>This bit indicates the level at the input pin Pn.x.<br>$0_B$ The input level of Pn.x is 0.<br>$1_B$ The input level of Pn.x is 1. |

| Field | Bits | Type | Description |
|---|---|---|---|
| **0** | 15, 14, 13, 12, 11, 10, 9, 8, 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Port 00 Pad Driver Mode Register 0**

**P00_PDR0**
**Port 00 Pad Driver Mode Register 0** **($040_H$)** **Reset Value: Table 114**
**P02_PDR0**
**Port 02 Pad Driver Mode Register 0** **($040_H$)** **Reset Value: Table 114**
**P10_PDR0**
**Port 10 Pad Driver Mode Register 0** **($040_H$)** **Reset Value: Table 114**
**P11_PDR0**
**Port 11 Pad Driver Mode Register 0** **($040_H$)** **Reset Value: Table 114**
**P14_PDR0**
**Port 14 Pad Driver Mode Register 0** **($040_H$)** **Reset Value: Table 114**
**P15_PDR0**
**Port 15 Pad Driver Mode Register 0** **($040_H$)** **Reset Value: Table 114**
**P20_PDR0**
**Port 20 Pad Driver Mode Register 0** **($040_H$)** **Reset Value: Table 114**
**P21_PDR0**
**Port 21 Pad Driver Mode Register 0** **($040_H$)** **Reset Value: Table 114**
**P22_PDR0**
**Port 22 Pad Driver Mode Register 0** **($040_H$)** **Reset Value: Table 114**
**P23_PDR0**
**Port 23 Pad Driver Mode Register 0** **($040_H$)** **Reset Value: Table 114**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **PL7** | | **PD7** | | **PL6** | | **PD6** | | **PL5** | | **PD5** | | **PL4** | | **PD4** | |
| rw | | rw | | rw | | rw | | rw | | rw | | rw | | rw | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **PL3** | | **PD3** | | **PL2** | | **PD2** | | **PL1** | | **PD1** | | **PL0** | | **PD0** | |
| rw | | rw | | rw | | rw | | rw | | rw | | rw | | rw | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PDx (x=0-7)** | 4*x+1:4*x | rw | **Pad Driver Mode for Pin x** |
| **PLx (x=0-7)** | 4*x+3:4*x+2 | rw | **Pad Level Selection for Pin x** |

**Table 113 Access Mode Restrictions sorted by descending priority**

Applies to **P00_PDR0**
Applies to **P02_PDR0**
Applies to **P10_PDR0**
Applies to **P11_PDR0**
Applies to **P14_PDR0**
Applies to **P15_PDR0**
Applies to **P20_PDR0**
Applies to **P21_PDR0**
Applies to **P22_PDR0**
Applies to **P23_PDR0**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN and Supervisor Mode and ENDINIT | rw | PDx (x=0-7), PLx (x=0-7) | write access for enabled masters |
| Otherwise (default) | r | PDx (x=0-7), PLx (x=0-7) | |

**Table 114 Reset Values**

Applies to **P00_PDR0**
Applies to **P02_PDR0**
Applies to **P10_PDR0**
Applies to **P11_PDR0**
Applies to **P14_PDR0**
Applies to **P15_PDR0**
Applies to **P20_PDR0**
Applies to **P21_PDR0**
Applies to **P22_PDR0**
Applies to **P23_PDR0**

| Reset Type | Reset Value | Note |
|---|---|---|
| After SSW execution | $2222\ 2222_H$ | Initial value in largest package |
| After SSW execution | $----\ ----_H$ | Initial value package dependent |

### Output Characteristics

The pad structure of the GPIO lines offers the possibility to select the output driver strength and the slew rate. These two parameters are controlled by the PDx bit fields in the pad driver mode registers Pn_PDR0/1 for output modes. The available modes depend on the respective pad type.

**Table 115 Pad Driver Mode Selection for RFast Pads**

| PDx.1 | PDx.0 | Speed Grade | Driver Setting |
|---|---|---|---|
| 0 | 0 | 1 | Strong driver, sharp edge (“ss”) |
| 0 | 1 | 2 | Strong driver, medium edge (“sm”) |
| 1 | 0 | 3 | Medium driver (“m”) |
| 1 | 1 | 4 | RGMII driver. |

**Table 116 Pad Driver Mode Selection for Fast Pads**

| PDx.1 | PDx.0 | Speed Grade | Driver Setting |
|---|---|---|---|
| 0 | 0 | 1 | Strong driver, sharp edge (“ss”) |
| 0 | 1 | 2 | Strong driver, medium edge (“sm”) |
| 1 | 0 | 3 | Medium driver (“m”) |
| 1 | 1 | 4 | TC39x A-Step: Medium driver (“m”)<br>Else: Reserved when operating as output. When operating as input see below “Pad Level Selection for Input Function”. |

**Table 117 Pad Driver Mode Selection for Slow Pads**

| PDx.1 | PDx.0 | Speed Grade | Driver Setting |
|---|---|---|---|
| X | 0 | 1 | Medium driver, sharp edge (“sm”)[1)] |
| X | 1 | 2 | Medium driver (“m”) |

1) This setting is marked “sm” as the electrical characteristics are identical to the strong driver medium edge setting. The Data Sheet contains also only common “sm” tables.

*Note: The Data Sheet describes the DC characteristics of all pad classes.*

### TTL/Automotive Input Selection

The input function can operate with different VIH and VIL levels depending on the pad supply voltage, the pad type and the selection done by the PLx bits of the Pn_PDRx as of **Table 118**. PLx.1 changes additionally the pull-up and pull-down resistors.

**Table 118 Pad Level Selection for Input Function**

| PLx.1 | PLx.0 | Input Levels |
|---|---|---|
| 0 | X | Automotive level “AL”. |
| 1 | 0 | TTL level for 5V pad supply.<br>Degraded TTL level used for CIF when pad supply is 3.3V |
| 1 | 1 | TTL level for 3.3V pad supply. |
| X | X | Only for pads with RGMII input buffer (marked “RGMII_Input” in the pinning table):<br>• when PDx.1=1 and PDx.0=1 the input level RGMII is selected.<br>• for other PDx values the input level is determined by PLx as for all other pads (first three rows of this table). |

### LVDS

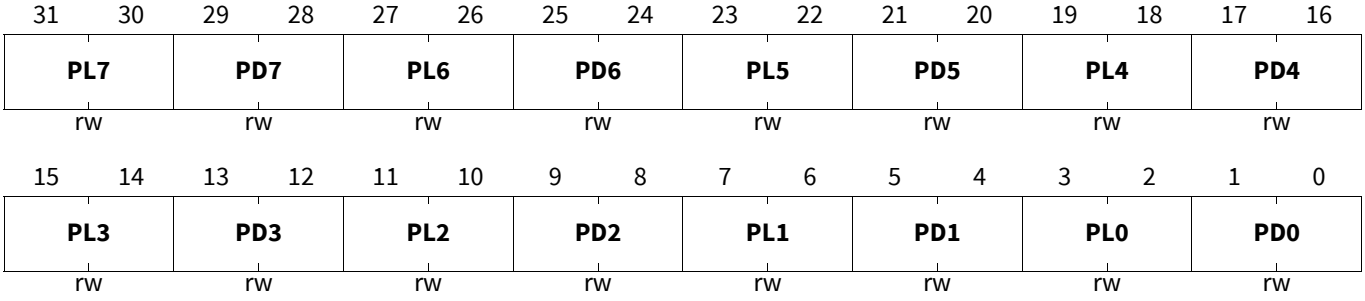
The default CMOS mode can be switched to LVDS mode in LVDS pads through the LPCRx register.

### Pad Driver Mode Registers

This is the general description of the PDR registers. Each port contains its own specific PDR registers, described additionally at each port, that can contain between one and eight PDx fields for PDR0 and PDR1 registers, respectively. Each PDx field controls 1 pin. For coding of PDx, see **Table 115**, **Table 116** and **Table 117**. Similarly, each PLx bit controls 1 pin. For coding of PLx, see **Table 118**.

The boot software configures the reset value of Pn_PDR0 and Pn_PDR1 registers from 0000 0000$_H$ to 2222 2222$_H$ except for analog ports and if the package doesn't make any of the related pins available. The resulting value depends on the implemented port width. The documented value is valid for the largest package.

**P32_PDR0**
**Port 32 Pad Driver Mode Register 0** **(040$_H$)** **Reset Value: Table 120**
**P33_PDR0**
**Port 33 Pad Driver Mode Register 0** **(040$_H$)** **Reset Value: Table 120**




| Field | Bits | Type | Description |
|---|---|---|---|
| **PDx (x=0-7)** | 4*x+1:4*x | rw | **Pad Driver Mode for Pin x** |
| **PLx (x=0-7)** | 4*x+3:4*x+2 | rw | **Pad Level Selection for Pin x** |

**Table 119 Access Mode Restrictions sorted by descending priority**

Applies to **P32_PDR0**
Applies to **P33_PDR0**

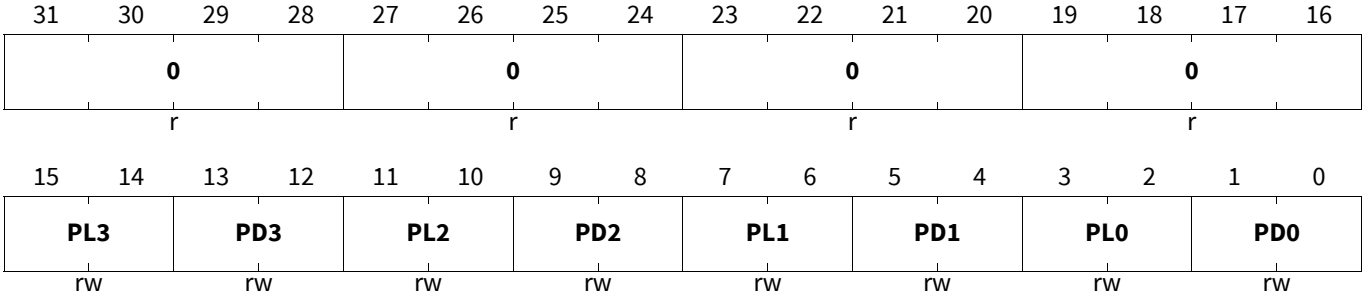
| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN and Supervisor Mode and ENDINIT | rw | PDx (x=0-7), PLx (x=0-7) | write access for enabled masters |
| Otherwise (default) | r | PDx (x=0-7), PLx (x=0-7) | |

**Table 120 Reset Values**

Applies to **P32_PDR0**
Applies to **P33_PDR0**

| Reset Type | Reset Value | Note |
|---|---|---|
| After SSW execution | 2222 2222$_H$ | Initial value in largest package |
| After SSW execution | ---- ----$_H$ | Initial value package dependent |

**P12_PDR0**
**Port 12 Pad Driver Mode Register 0** **($040_H$)** **Reset Value: Table 122**
**P34_PDR0**
**Port 34 Pad Driver Mode Register 0** **($040_H$)** **Reset Value: Table 122**




| Field | Bits | Type | Description |
|---|---|---|---|
| **PDx (x=0-3)** | 4*x+1:4*x | rw | **Pad Driver Mode for Pin x** |
| **PLx (x=0-3)** | 4*x+3:4*x+2 | rw | **Pad Level Selection for Pin x** |
| **0** | 31:28, 27:24, 23:20, 19:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 121 Access Mode Restrictions sorted by descending priority**

Applies to **P12_PDR0**
Applies to **P34_PDR0**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN and Supervisor Mode and ENDINIT | rw | PDx (x=0-3), PLx (x=0-3) | write access for enabled masters |
| Otherwise (default) | r | PDx (x=0-3), PLx (x=0-3) | |

**Table 122 Reset Values**

Applies to **P12_PDR0**
Applies to **P34_PDR0**

| Reset Type | Reset Value | Note |
|---|---|---|
| After SSW execution | 0000 $2222_H$ | Initial value in largest package |
| After SSW execution | 0000 $----_H$ | Initial value package dependent |

### Port 00 Pad Driver Mode Register 1

**P00_PDR1**
**Port 00 Pad Driver Mode Register 1** **($044_H$)** **Reset Value: Table 124**
**P11_PDR1**
**Port 11 Pad Driver Mode Register 1** **($044_H$)** **Reset Value: Table 124**
**P20_PDR1**
**Port 20 Pad Driver Mode Register 1** **($044_H$)** **Reset Value: Table 124**
**P33_PDR1**
**Port 33 Pad Driver Mode Register 1** **($044_H$)** **Reset Value: Table 124**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PL15 | | PD15 | | PL14 | | PD14 | | PL13 | | PD13 | | PL12 | | PD12 | |
| rw | | rw | | rw | | rw | | rw | | rw | | rw | | rw | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PL11 | | PD11 | | PL10 | | PD10 | | PL9 | | PD9 | | PL8 | | PD8 | |
| rw | | rw | | rw | | rw | | rw | | rw | | rw | | rw | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PDx (x=8-15)** | 4*x-31:4*x-32 | rw | **Pad Driver Mode for Pin x** |
| **PLx (x=8-15)** | 4*x-29:4*x-30 | rw | **Pad Level Selection for Pin x** |

**Table 123 Access Mode Restrictions sorted by descending priority**

Applies to **P00_PDR1**
Applies to **P11_PDR1**
Applies to **P20_PDR1**
Applies to **P33_PDR1**

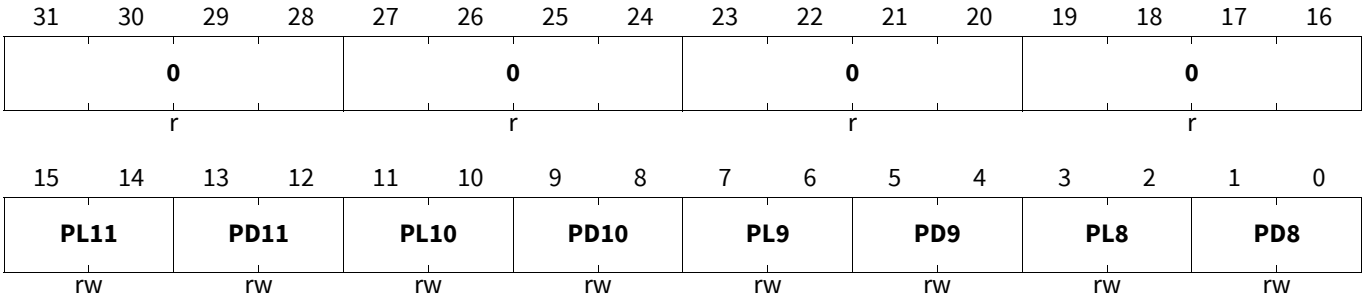
| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN and Supervisor Mode and ENDINIT | rw | PDx (x=8-15), PLx (x=8-15) | write access for enabled masters |
| Otherwise (default) | r | PDx (x=8-15), PLx (x=8-15) | |

**Table 124 Reset Values**

Applies to **P00_PDR1**
Applies to **P11_PDR1**
Applies to **P20_PDR1**
Applies to **P33_PDR1**

| Reset Type | Reset Value | Note |
|---|---|---|
| After SSW execution | $2222\ 2222_H$ | Initial value in largest package |
| After SSW execution | $----\ ----_H$ | Initial value package dependent |

**P02_PDR1**
**Port 02 Pad Driver Mode Register 1** **($044_H$)** **Reset Value: Table 126**
**P10_PDR1**
**Port 10 Pad Driver Mode Register 1** **($044_H$)** **Reset Value: Table 126**
**P14_PDR1**
**Port 14 Pad Driver Mode Register 1** **($044_H$)** **Reset Value: Table 126**
**P15_PDR1**
**Port 15 Pad Driver Mode Register 1** **($044_H$)** **Reset Value: Table 126**




| Field | Bits | Type | Description |
|---|---|---|---|
| **PDx (x=8-11)** | 4*x-31:4*x-32 | rw | **Pad Driver Mode for Pin x** |
| **PLx (x=8-11)** | 4*x-29:4*x-30 | rw | **Pad Level Selection for Pin x** |
| **0** | 31:28, 27:24, 23:20, 19:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 125 Access Mode Restrictions sorted by descending priority**

Applies to **P02_PDR1**
Applies to **P10_PDR1**
Applies to **P14_PDR1**
Applies to **P15_PDR1**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN and Supervisor Mode and ENDINIT | rw | PDx (x=8-11), PLx (x=8-11) | write access for enabled masters |
| Otherwise (default) | r | PDx (x=8-11), PLx (x=8-11) | |

**Table 126 Reset Values**

Applies to **P02_PDR1**
Applies to **P10_PDR1**
Applies to **P14_PDR1**
Applies to **P15_PDR1**

| Reset Type | Reset Value | Note |
|---|---|---|
| After SSW execution | 0000 $2222_H$ | Initial value in largest package |
| After SSW execution | 0000 $----_H$ | Initial value package dependent |

**Port 00 Emergency Stop Register**

**P00_ESR**
**Port 00 Emergency Stop Register** **($050_H$)** **Application Reset Value: 0000 $0000_H$**
**P11_ESR**
**Port 11 Emergency Stop Register** **($050_H$)** **Application Reset Value: 0000 $0000_H$**
**P20_ESR**
**Port 20 Emergency Stop Register** **($050_H$)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| EN15 | EN14 | EN13 | EN12 | EN11 | EN10 | EN9 | EN8 | EN7 | EN6 | EN5 | EN4 | EN3 | EN2 | EN1 | EN0 |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ENx (x=0-15)** | x | rw | **Emergency Stop Enable for Pin x**<br>This bit enables the emergency stop function for all GPIO lines. If the emergency stop condition is met and enabled, the output selection is automatically switched from alternate output function to GPIO input function.<br>$0_B$ Emergency stop function for Pn.x is disabled.<br>$1_B$ Emergency stop function for Pn.x is enabled. |
| **0** | 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 127 Access Mode Restrictions sorted by descending priority**

Applies to **P00_ESR**
Applies to **P11_ESR**
Applies to **P20_ESR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN and Supervisor Mode and ENDINIT | rw | ENx (x=0-15) | write access for enabled masters |
| Otherwise (default) | r | ENx (x=0-15) | |

Most GPIO lines have an emergency stop logic implemented (see Figure “General Structure of a Port Pin” in the Family chapter).

Each of these GPIO lines has its own emergency stop enable bit ENx that is located in the emergency stop register Pn_ESR of Port n. If the emergency stop signal becomes active, one of two states can be selected:

- Emergency stop function disabled (ENx = 0):
  The output line remains connected (alternate function).
- Emergency stop function enabled (ENx = 1):
  The mapped output function is disconnected and the safe state is entered by switching to input function with internal pull-up connected or tri-state, depending on the configured reset value of the corresponding Pn_IOCR register through PMSWCR5.TRISTREQ or setting of HWCFG[6].(the content of the corresponding PCx bit fields in register Pn_IOCR will not be considered).

### Exceptions for Emergency Stop Implementation

The Emergency Stop function is available for all GPIO Ports with the following exceptions:

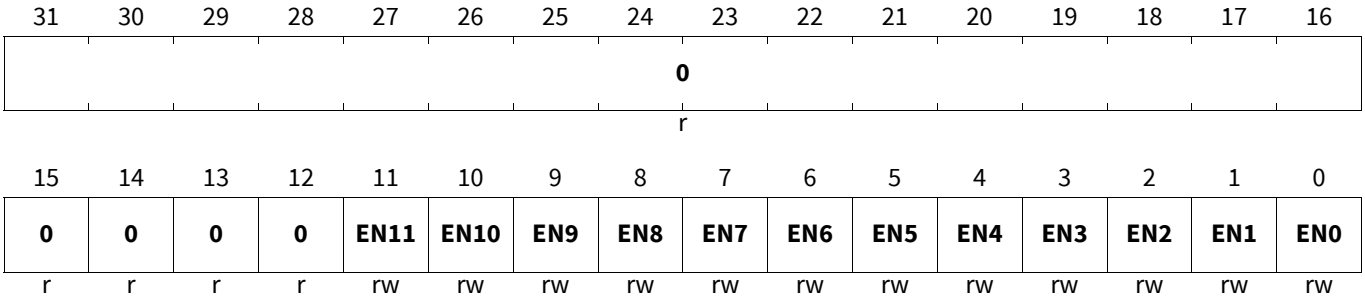
- Not available for P20.2 (General Purpose Input/GPI only, overlayed with Testmode)
- Not available for P40.x and P41.x (analog input ANx overlayed with GPI)
- Not available for P32.0 and P32.1 when using EVRC regulator.
- Not available for P21.2 (used as EMGSTOPB pin).
- Not available for P33.8 (used as EMGSTOPA pin).
- Not available for dedicated I/O without General Purpose Output function (e.g ESRx, TMS, TCK)

The Emergency Stop function can be overruled on the following GPIO Ports:

- P00.x: Emergency Stop can be overruled by the VADC. Overruling can be disabled via the control register P00_PCSR.
- P14.0 and P14.1: Emergency Stop can be overruled in the DXCPL (DAP over CAN physical layer) mode. No Overruling in the DXCM (Debug over can message) mode
- P21.6: Emergency Stop can be overruled in JTAG mode if this pin is used as TDI
- P21.7: Emergency Stop can be overruled in JTAG or Three Pin DAP mode.
- P33.0-7, P33.9-15 and P34.1: Emergency Stop can be overruled by the 8-Bit Standby Controller (SCR), if implemented. Overruling can be disabled via the control register P33_PCSR and P34_PCSR.

On pins with LVDS TX pads the Emergency Stop affects only the CMOS driver not the LVDS driver. Thus only when LPCRx.TX_EN selects CMOS mode the output is switched off. When TX_EN selects LVDS mode the output is not switched off.

**P02_ESR**
**Port 02 Emergency Stop Register** **($050_H$)** **Application Reset Value: 0000 0000$_H$**
**P10_ESR**
**Port 10 Emergency Stop Register** **($050_H$)** **Application Reset Value: 0000 0000$_H$**
**P14_ESR**
**Port 14 Emergency Stop Register** **($050_H$)** **Application Reset Value: 0000 0000$_H$**
**P15_ESR**
**Port 15 Emergency Stop Register** **($050_H$)** **Application Reset Value: 0000 0000$_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **ENx (x=0-11)** | x | rw | **Emergency Stop Enable for Pin x**<br>This bit enables the emergency stop function for all GPIO lines. If the emergency stop condition is met and enabled, the output selection is automatically switched from alternate output function to GPIO input function.<br>$0_B$ Emergency stop function for Pn.x is disabled.<br>$1_B$ Emergency stop function for Pn.x is enabled. |
| **0** | 15, 14, 13, 12, 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 128 Access Mode Restrictions sorted by descending priority**

Applies to **P02_ESR**
Applies to **P10_ESR**
Applies to **P14_ESR**
Applies to **P15_ESR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN and Supervisor Mode and ENDINIT | rw | ENx (x=0-11) | write access for enabled masters |
| Otherwise (default) | r | ENx (x=0-11) | |

**P12_ESR**
**Port 12 Emergency Stop Register** **($050_H$)** **Application Reset Value: 0000 0000$_H$**
**P34_ESR**
**Port 34 Emergency Stop Register** **($050_H$)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | EN3 | EN2 | EN1 | EN0 |
| r | r | r | r | r | r | r | r | r | r | r | r | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ENx (x=0-3)** | x | rw | **Emergency Stop Enable for Pin x**<br>This bit enables the emergency stop function for all GPIO lines. If the emergency stop condition is met and enabled, the output selection is automatically switched from alternate output function to GPIO input function.<br>$0_B$ Emergency stop function for Pn.x is disabled.<br>$1_B$ Emergency stop function for Pn.x is enabled. |
| **0** | 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 129 Access Mode Restrictions sorted by descending priority**

Applies to **P12_ESR**
Applies to **P34_ESR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN and Supervisor Mode and ENDINIT | rw | ENx (x=0-3) | write access for enabled masters |
| Otherwise (default) | r | ENx (x=0-3) | |

**P21_ESR**

**Port 21 Emergency Stop Register** **($050_H$)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

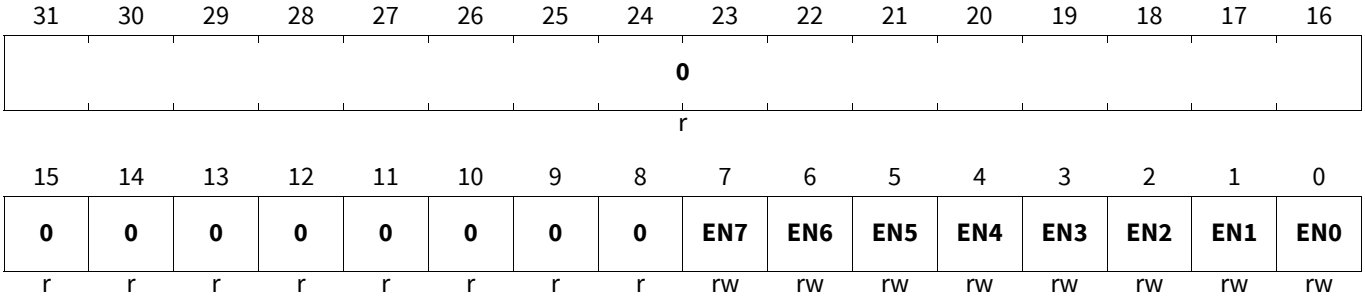
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | EN7 | EN6 | EN5 | EN4 | EN3 | 0 | EN1 | EN0 |
| r | r | r | r | r | r | r | r | rw | rw | rw | rw | rw | r | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ENx (x=0-1,3-7)** | x | rw | **Emergency Stop Enable for Pin x**<br>This bit enables the emergency stop function for all GPIO lines. If the emergency stop condition is met and enabled, the output selection is automatically switched from alternate output function to GPIO input function.<br>$0_B$ Emergency stop function for Pn.x is disabled.<br>$1_B$ Emergency stop function for Pn.x is enabled. |
| **0** | 15, 14, 13, 12, 11, 10, 9, 8, 2, 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 130 Access Mode Restrictions of P21_ESR sorted by descending priority**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN and Supervisor Mode and ENDINIT | rw | ENx (x=0-1,3-7) | write access for enabled masters |
| Otherwise (default) | r | ENx (x=0-1,3-7) | |

**P22_ESR**
**Port 22 Emergency Stop Register** **($050_H$)** **Application Reset Value: 0000 0000$_H$**
**P23_ESR**
**Port 23 Emergency Stop Register** **($050_H$)** **Application Reset Value: 0000 0000$_H$**
**P32_ESR**
**Port 32 Emergency Stop Register** **($050_H$)** **Application Reset Value: 0000 0000$_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **ENx (x=0-7)** | x | rw | **Emergency Stop Enable for Pin x**<br>This bit enables the emergency stop function for all GPIO lines. If the emergency stop condition is met and enabled, the output selection is automatically switched from alternate output function to GPIO input function.<br>$0_B$ Emergency stop function for Pn.x is disabled.<br>$1_B$ Emergency stop function for Pn.x is enabled. |
| **0** | 15, 14, 13, 12, 11, 10, 9, 8, 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 131 Access Mode Restrictions sorted by descending priority**

Applies to **P22_ESR**
Applies to **P23_ESR**
Applies to **P32_ESR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN and Supervisor Mode and ENDINIT | rw | ENx (x=0-7) | write access for enabled masters |
| Otherwise (default) | r | ENx (x=0-7) | |

**P33_ESR**

**Port 33 Emergency Stop Register** **($050_H$)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| EN15 | EN14 | EN13 | EN12 | EN11 | EN10 | EN9 | 0 | EN7 | EN6 | EN5 | EN4 | EN3 | EN2 | EN1 | EN0 |
| rw | rw | rw | rw | rw | rw | rw | r | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ENx (x=0-7,9-15)** | x | rw | **Emergency Stop Enable for Pin x**<br>This bit enables the emergency stop function for all GPIO lines. If the emergency stop condition is met and enabled, the output selection is automatically switched from alternate output function to GPIO input function.<br>$0_B$ Emergency stop function for Pn.x is disabled.<br>$1_B$ Emergency stop function for Pn.x is enabled. |
| **0** | 8,<br>31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 132 Access Mode Restrictions of P33_ESR sorted by descending priority**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN and Supervisor Mode and ENDINIT | rw | ENx (x=0-7,9-15) | write access for enabled masters |
| Otherwise (default) | r | ENx (x=0-7,9-15) | |

### Port 00 Pin Function Decision Control Register

The pad structure of the GPIO lines offers the possibility to disable/enable port pad, select digital input or analog ADC input functionalities. Note that Class S pads have different characteristics than other digital input pads. For analog inputs, setting PDISx to 1 disables the Schmitt trigger input buffer, which would otherwise reduce analog input accuracy. For the ADC diagnostic features “PDD” and “MD” however the corresponding PDISx needs to be 0 to allow activation of their pull resistors. This feature can be controlled by individual bits in the Pn_PDISC register, independently from input/output and pull-up/pull-down control functionality as programmed in the Pn_IOCR register. One Pn_PDISC register is assigned to each port.

*Note: After reset, all Px_PDISC registers have the reset value of $0000\ 0000_H$. The startup software enables only the pads with digital input/output functionality which are available in that package. P40_PDISC and P41_PDISC are configured by the SSW for analog input function (kept disabled). The documented reset value shows the value in the largest package.*

**P00_PDISC**
**Port 00 Pin Function Decision Control Register ($060_H$)** **Reset Value: Table 134**
**P11_PDISC**
**Port 11 Pin Function Decision Control Register ($060_H$)** **Reset Value: Table 134**
**P20_PDISC**
**Port 20 Pin Function Decision Control Register ($060_H$)** **Reset Value: Table 134**
**P33_PDISC**
**Port 33 Pin Function Decision Control Register ($060_H$)** **Reset Value: Table 134**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PDIS15 | PDIS14 | PDIS13 | PDIS12 | PDIS11 | PDIS10 | PDIS9 | PDIS8 | PDIS7 | PDIS6 | PDIS5 | PDIS4 | PDIS3 | PDIS2 | PDIS1 | PDIS0 |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PDISx (x=0-15)** | x | rw | **Pin Function Decision Control for Pin x**<br>This bit selects the function of the port pad.<br>$0_B$ Digital functionality of pad Pn.x is enabled.<br>$1_B$ Digital functionality (including pull resistors) of pad Pn.x is disabled. Analog input function (where this is available) can be used. |
| **0** | 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 133 Access Mode Restrictions sorted by descending priority**

Applies to **P00_PDISC**
Applies to **P11_PDISC**
Applies to **P20_PDISC**
Applies to **P33_PDISC**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN and Supervisor Mode and ENDINIT | rw | PDISx (x=0-15) | write access for enabled masters |
| Otherwise (default) | r | PDISx (x=0-15) | |

**Table 134 Reset Values**

Applies to **P00_PDISC**
Applies to **P11_PDISC**
Applies to **P20_PDISC**
Applies to **P33_PDISC**

| Reset Type | Reset Value | Note |
|---|---|---|
| After SSW execution | 0000 $0000_H$ | Initial value in largest package |
| After SSW execution | 0000 $----_H$ | Initial value package dependent |

**P02_PDISC**
**Port 02 Pin Function Decision Control Register ($060_H$)** **Reset Value: Table 136**
**P10_PDISC**
**Port 10 Pin Function Decision Control Register ($060_H$)** **Reset Value: Table 136**
**P14_PDISC**
**Port 14 Pin Function Decision Control Register ($060_H$)** **Reset Value: Table 136**
**P15_PDISC**
**Port 15 Pin Function Decision Control Register ($060_H$)** **Reset Value: Table 136**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | PDIS11 | PDIS10 | PDIS9 | PDIS8 | PDIS7 | PDIS6 | PDIS5 | PDIS4 | PDIS3 | PDIS2 | PDIS1 | PDIS0 |
| r | r | r | r | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PDISx (x=0-11)** | x | rw | **Pin Function Decision Control for Pin x**<br>This bit selects the function of the port pad.<br>$0_B$ Digital functionality of pad Pn.x is enabled.<br>$1_B$ Digital functionality (including pull resistors) of pad Pn.x is disabled. Analog input function (where this is available) can be used. |
| **0** | 15, 14, 13, 12, 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 135 Access Mode Restrictions sorted by descending priority**

Applies to **P02_PDISC**
Applies to **P10_PDISC**
Applies to **P14_PDISC**
Applies to **P15_PDISC**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN and Supervisor Mode and ENDINIT | rw | PDISx (x=0-11) | write access for enabled masters |
| Otherwise (default) | r | PDISx (x=0-11) | |

**Table 136 Reset Values**

Applies to **P02_PDISC**
Applies to **P10_PDISC**
Applies to **P14_PDISC**
Applies to **P15_PDISC**

| Reset Type | Reset Value | Note |
|---|---|---|
| After SSW execution | 0000 $0000_H$ | Initial value in largest package |
| After SSW execution | 0000 $0---_H$ | Initial value package dependent |

**P12_PDISC**
**Port 12 Pin Function Decision Control Register ($060_H$)** **Reset Value: Table 138**
**P34_PDISC**
**Port 34 Pin Function Decision Control Register ($060_H$)** **Reset Value: Table 138**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | PDIS3 | PDIS2 | PDIS1 | PDIS0 |
| r | r | r | r | r | r | r | r | r | r | r | r | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PDISx (x=0-3)** | x | rw | **Pin Function Decision Control for Pin x**<br>This bit selects the function of the port pad.<br>$0_B$ Digital functionality of pad Pn.x is enabled.<br>$1_B$ Digital functionality (including pull resistors) of pad Pn.x is disabled. Analog input function (where this is available) can be used. |

| Field | Bits | Type | Description |
|---|---|---|---|
| **0** | 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 137 Access Mode Restrictions sorted by descending priority**

Applies to **P12_PDISC**
Applies to **P34_PDISC**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN and Supervisor Mode and ENDINIT | rw | PDISx (x=0-3) | write access for enabled masters |
| Otherwise (default) | r | PDISx (x=0-3) | |

**Table 138 Reset Values**

Applies to **P12_PDISC**
Applies to **P34_PDISC**

| Reset Type | Reset Value | Note |
|---|---|---|
| After SSW execution | $0000\ 0000_H$ | Initial value in largest package |
| After SSW execution | $0000\ 000-_H$ | Initial value package dependent |

**P21_PDISC**
**Port 21 Pin Function Decision Control Register ($060_H$)** **Reset Value: Table 140**
**P22_PDISC**
**Port 22 Pin Function Decision Control Register ($060_H$)** **Reset Value: Table 140**
**P23_PDISC**
**Port 23 Pin Function Decision Control Register ($060_H$)** **Reset Value: Table 140**
**P32_PDISC**
**Port 32 Pin Function Decision Control Register ($060_H$)** **Reset Value: Table 140**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **PDIS7** | **PDIS6** | **PDIS5** | **PDIS4** | **PDIS3** | **PDIS2** | **PDIS1** | **PDIS0** |
| r | r | r | r | r | r | r | r | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PDISx (x=0-7)** | x | rw | **Pin Function Decision Control for Pin x**<br>This bit selects the function of the port pad.<br>$0_B$ Digital functionality of pad Pn.x is enabled.<br>$1_B$ Digital functionality (including pull resistors) of pad Pn.x is disabled. Analog input function (where this is available) can be used. |
| **0** | 15, 14, 13, 12, 11, 10, 9, 8,<br>31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 139 Access Mode Restrictions sorted by descending priority**

Applies to **P21_PDISC**
Applies to **P22_PDISC**
Applies to **P23_PDISC**
Applies to **P32_PDISC**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN and Supervisor Mode and ENDINIT | rw | PDISx (x=0-7) | write access for enabled masters |
| Otherwise (default) | r | PDISx (x=0-7) | |

**Table 140 Reset Values**

Applies to **P21_PDISC**
Applies to **P22_PDISC**
Applies to **P23_PDISC**
Applies to **P32_PDISC**

| Reset Type | Reset Value | Note |
|---|---|---|
| After SSW execution | $0000\ 0000_H$ | Initial value in largest package |
| After SSW execution | $0000\ 00--_H$ | Initial value package dependent |

**Port 00 Pin Controller Select Register**

This register has different functionality in each port:

- In Ports shared with the standby controller (SCR) it selects if the SCR or the Tricore system control data and control functions of these port lines.
- In Ports with analog inputs to the EVADC it enables control of pull by the EVADC for the Pull Down Diagnostics (PDD) / Multiplexer Diagnostics (MD) feature.
- In Ports with Ethernet output it selects between alternate output and fast RGMII/RMII/MII mode.
- In Ports with SMU FSP pin (P33.8) the PCSR.SEL bit enables the SMU to override pad configuration signals. Therefore this bit has the reset value $1_B$ and shall be kept $1_B$ by the application. The SMU override is documented in the SMU chapter (see SMU_PCTL.PCFG and Figure “SMU/PAD Control Interface to the PADs”).

**P00_PCSR**
**Port 00 Pin Controller Select Register** **($064_H$)** **Application Reset Value: 0000 0000$_H$**
**P20_PCSR**
**Port 20 Pin Controller Select Register** **($064_H$)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | **0** | | | | | | | |
| r | | | | | | | | r | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **R15** | **R14** | **R13** | **R12** | **R11** | **R10** | **R9** | **R8** | **R7** | **R6** | **R5** | **R4** | **R3** | **R2** | **R1** | **R0** |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **Rx (x=0-15)** | x | rw | **Reserved**<br>Read as 0; should be written with 0. |
| **0** | 30:16, 31 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 141 Access Mode Restrictions sorted by descending priority**

Applies to **P00_PCSR**
Applies to **P20_PCSR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Supervisor Mode and Safety ENDINIT | r | Rx (x=0-15) | write access only for masters with supervisor mode |
| Otherwise (default) | r | Rx (x=0-15) | |

**P02_PCSR**
**Port 02 Pin Controller Select Register** **($064_H$)** **Application Reset Value: 0000 0000$_H$**
**P10_PCSR**
**Port 10 Pin Controller Select Register** **($064_H$)** **Application Reset Value: 0000 0000$_H$**
**P14_PCSR**
**Port 14 Pin Controller Select Register** **($064_H$)** **Application Reset Value: 0000 0000$_H$**
**P15_PCSR**
**Port 15 Pin Controller Select Register** **($064_H$)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | **0** | | | | | | | |
| r | | | | | | | | r | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **R11** | **R10** | **R9** | **R8** | **R7** | **R6** | **R5** | **R4** | **R3** | **R2** | **R1** | **R0** |
| r | r | r | r | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **Rx (x=0-11)** | x | rw | **Reserved**<br>Read as 0; should be written with 0. |
| **0** | 15, 14, 13, 12, 30:16, 31 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 142 Access Mode Restrictions sorted by descending priority**

Applies to **P02_PCSR**
Applies to **P10_PCSR**
Applies to **P14_PCSR**
Applies to **P15_PCSR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Supervisor Mode and Safety ENDINIT | r | Rx (x=0-11) | write access only for masters with supervisor mode |
| Otherwise (default) | r | Rx (x=0-11) | |

**P11_PCSR**

**Port 11 Pin Controller Select Register (064$_H$) Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | **0** | | | | | | | |
| r | | | | | | | | r | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **R15** | **R14** | **R13** | **R12** | **R11** | **R10** | **R9** | **R8** | **R7** | **SEL6** | **R5** | **SEL4** | **SEL3** | **SEL2** | **SEL1** | **SEL0** |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SELx (x=0-4,6)** | x | rw | **Output Select for Pin x**<br>This bit enables or disables alternate/fast Ethernet output.<br>$0_B$ Ethernet output via ports alternate output of pin x.<br>$1_B$ Ethernet output via fast RGMII/RMII/MII mode of pin x. |
| **Rx (x=5,7-15)** | x | rw | **Reserved**<br>Read as 0; should be written with 0. |
| **0** | 30:16, 31 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 143 Access Mode Restrictions of P11_PCSR sorted by descending priority**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Supervisor Mode and Safety ENDINIT | r | Rx (x=5,7-15) | write access only for masters with supervisor mode |
| | rw | SELx (x=0-4,6) | |
| Otherwise (default) | r | Rx (x=5,7-15), SELx (x=0-4,6) | |

**P12_PCSR**

**Port 12 Pin Controller Select Register ($064_H$) Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | **0** | | | | | | | |
| r | | | | | | | | r | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **R3** | **R2** | **R1** | **R0** |
| r | r | r | r | r | r | r | r | r | r | r | r | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **Rx (x=0-3)** | x | rw | **Reserved**<br>Read as 0; should be written with 0. |
| **0** | 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 30:16, 31 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 144 Access Mode Restrictions of P12_PCSR sorted by descending priority**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Supervisor Mode and Safety ENDINIT | r | Rx (x=0-3) | write access only for masters with supervisor mode |
| Otherwise (default) | r | Rx (x=0-3) | |

**P21_PCSR**
**Port 21 Pin Controller Select Register** **($064_H$)** **Application Reset Value: 0000 0000$_H$**
**P22_PCSR**
**Port 22 Pin Controller Select Register** **($064_H$)** **Application Reset Value: 0000 0000$_H$**
**P23_PCSR**
**Port 23 Pin Controller Select Register** **($064_H$)** **Application Reset Value: 0000 0000$_H$**
**P32_PCSR**
**Port 32 Pin Controller Select Register** **($064_H$)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | **0** | | | | | | | |
| r | | | | | | | | r | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **R7** | **R6** | **R5** | **R4** | **R3** | **R2** | **R1** | **R0** |
| r | r | r | r | r | r | r | r | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **Rx (x=0-7)** | x | rw | **Reserved**<br>Read as 0; should be written with 0. |
| **0** | 15, 14, 13, 12, 11, 10, 9, 8,<br>30:16,<br>31 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 145 Access Mode Restrictions sorted by descending priority**

Applies to **P21_PCSR**
Applies to **P22_PCSR**
Applies to **P23_PCSR**
Applies to **P32_PCSR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Supervisor Mode and Safety ENDINIT | r | Rx (x=0-7) | write access only for masters with supervisor mode |
| Otherwise (default) | r | Rx (x=0-7) | |

**P33_PCSR**

**Port 33 Pin Controller Select Register** **($064_H$)** **Application Reset Value: $0000\ 0100_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **LCK** | | | | | | | | **0** | | | | | | | |
| rh | | | | | | | | r | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **SEL15** | **SEL14** | **SEL13** | **SEL12** | **SEL11** | **SEL10** | **SEL9** | **SEL8** | **SEL7** | **SEL6** | **SEL5** | **SEL4** | **SEL3** | **SEL2** | **SEL1** | **SEL0** |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SELx (x=0-7,9-15)** | x | rw | **Output Select for Pin x**<br>This bit enables or disables SCR control.<br>$0_B$ Tricore selected for data and control of pin x and not SCR.<br>$1_B$ SCR selected for data and control of pin x. |
| **SELx (x=8)** | x | rw | **Output Select for Pin x**<br>This bit enables or disables SMU to override pad configuration.<br>$0_B$ Disable SMU override of pad configuration for FSP pin x.<br>$1_B$ Enable SMU to override pad configuration for FSP pin x. |
| **LCK** | 31 | rh | **Lock Status**<br>This bit indicates if the register can be updated with a new value or if the register is locked due to an ongoing transfer to the SCR and a write action from the bus has no effect.<br>In Ports without SCR overlay this bit is always $0_B$.<br>$0_B$ The register is unlocked and can be updated.<br>$1_B$ The register is locked (a write transfer to SCR is ongoing) and can not be updated. |
| **0** | 30:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 146 Access Mode Restrictions of P33_PCSR sorted by descending priority**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Supervisor Mode and Safety ENDINIT | rh | LCK | write access only for masters with supervisor mode |
| | rw | SELx (x=0-7,9-15), SELx (x=8) | |
| Otherwise (default) | r | SELx (x=0-7,9-15), SELx (x=8) | |
| | rh | LCK | |

**P34_PCSR**

**Port 34 Pin Controller Select Register** **($064_H$)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **LCK** | | | | | | | | **0** | | | | | | | |
| rh | | | | | | | | r | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **R3** | **R2** | **SEL1** | **R0** |
| r | r | r | r | r | r | r | r | r | r | r | r | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SELx (x=1)** | x | rw | **Output Select for Pin x**<br>This bit enables or disables SCR control.<br>$0_B$ Tricore selected for data and control of pin x and not SCR.<br>$1_B$ SCR selected for data and control of pin x. |
| **Rx (x=0,2-3)** | x | rw | **Reserved**<br>Read as 0; should be written with 0. |
| **LCK** | 31 | rh | **Lock Status**<br>This bit indicates if the register can be updated with a new value or if the register is locked due to an ongoing transfer to the SCR and a write action from the bus has no effect.<br>In Ports without SCR overlay this bit is always $0_B$.<br>$0_B$ The register is unlocked and can be updated.<br>$1_B$ The register is locked (a write transfer to SCR is ongoing) and can not be updated. |
| **0** | 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 30:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 147 Access Mode Restrictions of P34_PCSR sorted by descending priority**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Supervisor Mode and Safety ENDINIT | r | Rx (x=0,2-3) | write access only for masters with supervisor mode |
| | rh | LCK | |
| | rw | SELx (x=1) | |
| Otherwise (default) | r | Rx (x=0,2-3), SELx (x=1) | |
| | rh | LCK | |

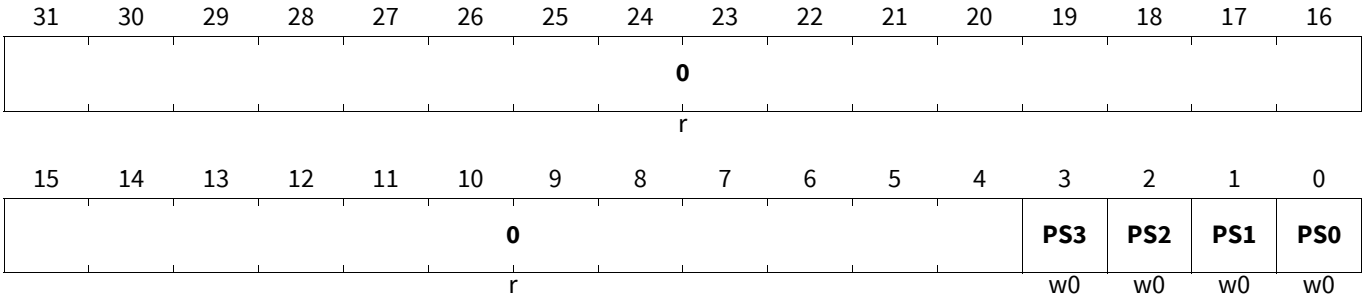
### Port 00 Output Modification Set Register 0

The port output modification set register x, (x = 0, 4, 8, 12) contains control bits to individually set the logic state of a single port line by manipulating the output register.

*Note: Registers Pn_OMSRx (x = 0, 4, 8, 12) are virtual and does not contain any flip-flop. A read action delivers the value of 0. One 8 or 16-bits write behaves as a 32-bit write padded with zeros.*

Register Pn_OMSR0 sets the logic state of Pn.[3:0] port lines

**P00_OMSR0**
**Port 00 Output Modification Set Register 0 ($070_H$) Application Reset Value: 0000 0000$_H$**
**P02_OMSR0**
**Port 02 Output Modification Set Register 0 ($070_H$) Application Reset Value: 0000 0000$_H$**
**P10_OMSR0**
**Port 10 Output Modification Set Register 0 ($070_H$) Application Reset Value: 0000 0000$_H$**
**P11_OMSR0**
**Port 11 Output Modification Set Register 0 ($070_H$) Application Reset Value: 0000 0000$_H$**
**P12_OMSR0**
**Port 12 Output Modification Set Register 0 ($070_H$) Application Reset Value: 0000 0000$_H$**
**P14_OMSR0**
**Port 14 Output Modification Set Register 0 ($070_H$) Application Reset Value: 0000 0000$_H$**
**P15_OMSR0**
**Port 15 Output Modification Set Register 0 ($070_H$) Application Reset Value: 0000 0000$_H$**
**P20_OMSR0**
**Port 20 Output Modification Set Register 0 ($070_H$) Application Reset Value: 0000 0000$_H$**
**P21_OMSR0**
**Port 21 Output Modification Set Register 0 ($070_H$) Application Reset Value: 0000 0000$_H$**
**P22_OMSR0**
**Port 22 Output Modification Set Register 0 ($070_H$) Application Reset Value: 0000 0000$_H$**



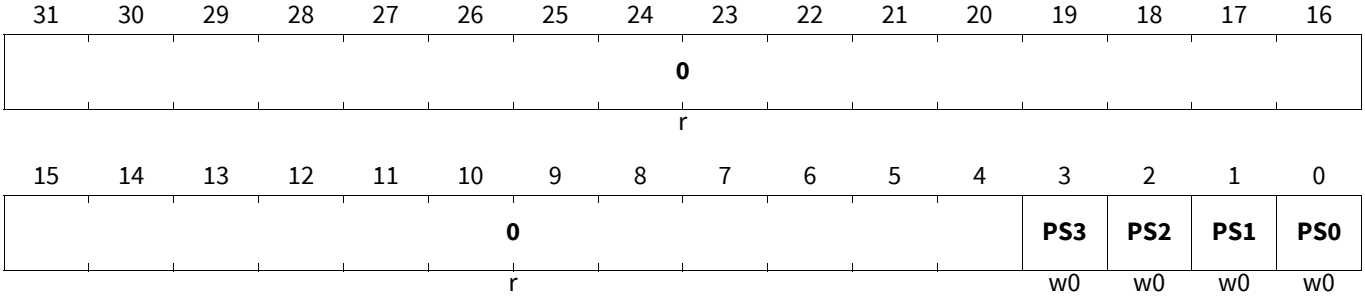

| Field | Bits | Type | Description |
|---|---|---|---|
| **PSx (x=0-3)** | x | w0 | **Set Bit x**<br>Setting this bit will set the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Sets Pn_OUT.Px |
| **0** | 31:4 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 148 Access Mode Restrictions sorted by descending priority**

Applies to **P00_OMSR0**
Applies to **P02_OMSR0**
Applies to **P10_OMSR0**
Applies to **P11_OMSR0**
Applies to **P12_OMSR0**
Applies to **P14_OMSR0**
Applies to **P15_OMSR0**
Applies to **P20_OMSR0**
Applies to **P21_OMSR0**
Applies to **P22_OMSR0**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PSx (x=0-3) | write access for enabled masters |
| Otherwise (default) | r0 | PSx (x=0-3) | |

**P23_OMSR0**
**Port 23 Output Modification Set Register 0** **($070_H$)** **Application Reset Value: 0000 0000$_H$**
**P32_OMSR0**
**Port 32 Output Modification Set Register 0** **($070_H$)** **Application Reset Value: 0000 0000$_H$**
**P33_OMSR0**
**Port 33 Output Modification Set Register 0** **($070_H$)** **Application Reset Value: 0000 0000$_H$**
**P34_OMSR0**
**Port 34 Output Modification Set Register 0** **($070_H$)** **Application Reset Value: 0000 0000$_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **PSx (x=0-3)** | x | w0 | **Set Bit x**<br>Setting this bit will set the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Sets Pn_OUT.Px |
| **0** | 31:4 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 149 Access Mode Restrictions sorted by descending priority**

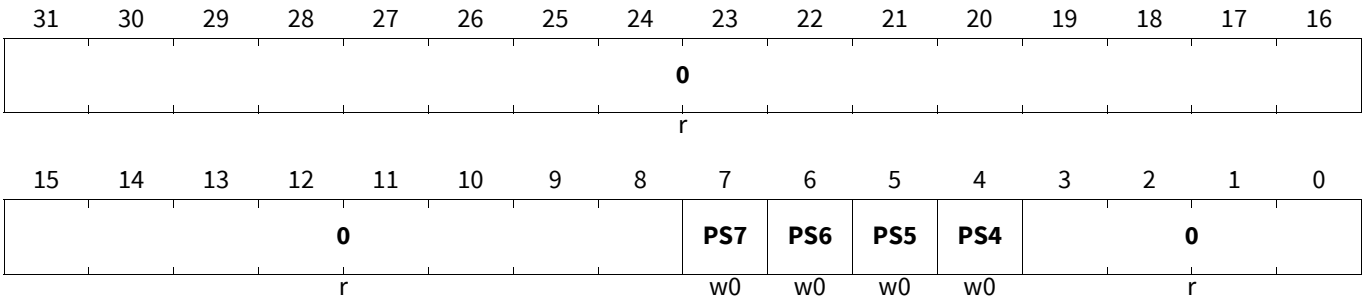
Applies to **P23_OMSR0**
Applies to **P32_OMSR0**
Applies to **P33_OMSR0**
Applies to **P34_OMSR0**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PSx (x=0-3) | write access for enabled masters |
| Otherwise (default) | r0 | PSx (x=0-3) | |

**Port 00 Output Modification Set Register 4**

Register Pn_OMSR4 sets the logic state of Pn.[7:4] port lines

**P00_OMSR4**
**Port 00 Output Modification Set Register 4 ($074_H$) Application Reset Value: $0000\ 0000_H$**
**P02_OMSR4**
**Port 02 Output Modification Set Register 4 ($074_H$) Application Reset Value: $0000\ 0000_H$**
**P10_OMSR4**
**Port 10 Output Modification Set Register 4 ($074_H$) Application Reset Value: $0000\ 0000_H$**
**P11_OMSR4**
**Port 11 Output Modification Set Register 4 ($074_H$) Application Reset Value: $0000\ 0000_H$**
**P14_OMSR4**
**Port 14 Output Modification Set Register 4 ($074_H$) Application Reset Value: $0000\ 0000_H$**
**P15_OMSR4**
**Port 15 Output Modification Set Register 4 ($074_H$) Application Reset Value: $0000\ 0000_H$**
**P20_OMSR4**
**Port 20 Output Modification Set Register 4 ($074_H$) Application Reset Value: $0000\ 0000_H$**
**P21_OMSR4**
**Port 21 Output Modification Set Register 4 ($074_H$) Application Reset Value: $0000\ 0000_H$**
**P22_OMSR4**
**Port 22 Output Modification Set Register 4 ($074_H$) Application Reset Value: $0000\ 0000_H$**
**P23_OMSR4**
**Port 23 Output Modification Set Register 4 ($074_H$) Application Reset Value: $0000\ 0000_H$**



| Field | Bits | Type | Description |
|---|---|---|---|
| **PSx (x=4-7)** | x | w0 | **Set Bit x**<br>Setting this bit will set the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Sets Pn_OUT.Px |
| **0** | 3:0, 31:8 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 150 Access Mode Restrictions sorted by descending priority**

Applies to **P00_OMSR4**
Applies to **P02_OMSR4**
Applies to **P10_OMSR4**
Applies to **P11_OMSR4**
Applies to **P14_OMSR4**
Applies to **P15_OMSR4**
Applies to **P20_OMSR4**
Applies to **P21_OMSR4**
Applies to **P22_OMSR4**
Applies to **P23_OMSR4**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PSx (x=4-7) | write access for enabled masters |
| Otherwise (default) | r0 | PSx (x=4-7) | |

**P32_OMSR4**
**Port 32 Output Modification Set Register 4 ($074_H$) Application Reset Value: $0000\ 0000_H$**
**P33_OMSR4**
**Port 33 Output Modification Set Register 4 ($074_H$) Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | PS7 | PS6 | PS5 | PS4 | 0 | | | |
| r | | | | | | | | w0 | w0 | w0 | w0 | r | | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PSx (x=4-7)** | x | w0 | **Set Bit x**<br>Setting this bit will set the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Sets Pn_OUT.Px |

| Field | Bits | Type | Description |
|---|---|---|---|
| **0** | 3:0,<br>31:8 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 151 Access Mode Restrictions sorted by descending priority**

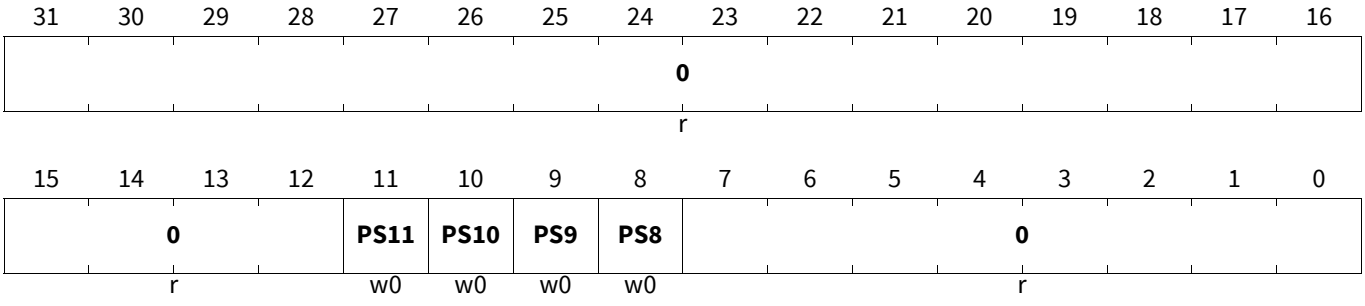
Applies to **P32_OMSR4**
Applies to **P33_OMSR4**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PSx (x=4-7) | write access for enabled masters |
| Otherwise (default) | r0 | PSx (x=4-7) | |

**Port 00 Output Modification Set Register 8**

Register Pn_OMSR8 sets the logic state of Pn.[11:8] port lines

**P00_OMSR8**
**Port 00 Output Modification Set Register 8 ($078_H$) Application Reset Value: 0000 0000$_H$**
**P02_OMSR8**
**Port 02 Output Modification Set Register 8 ($078_H$) Application Reset Value: 0000 0000$_H$**
**P10_OMSR8**
**Port 10 Output Modification Set Register 8 ($078_H$) Application Reset Value: 0000 0000$_H$**
**P11_OMSR8**
**Port 11 Output Modification Set Register 8 ($078_H$) Application Reset Value: 0000 0000$_H$**
**P14_OMSR8**
**Port 14 Output Modification Set Register 8 ($078_H$) Application Reset Value: 0000 0000$_H$**
**P15_OMSR8**
**Port 15 Output Modification Set Register 8 ($078_H$) Application Reset Value: 0000 0000$_H$**
**P20_OMSR8**
**Port 20 Output Modification Set Register 8 ($078_H$) Application Reset Value: 0000 0000$_H$**
**P33_OMSR8**
**Port 33 Output Modification Set Register 8 ($078_H$) Application Reset Value: 0000 0000$_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **PSx (x=8-11)** | x | w0 | **Set Bit x**<br>Setting this bit will set the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Sets Pn_OUT.Px |

| Field | Bits | Type | Description |
|---|---|---|---|
| **0** | 7:0,<br>31:12 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 152 Access Mode Restrictions sorted by descending priority**

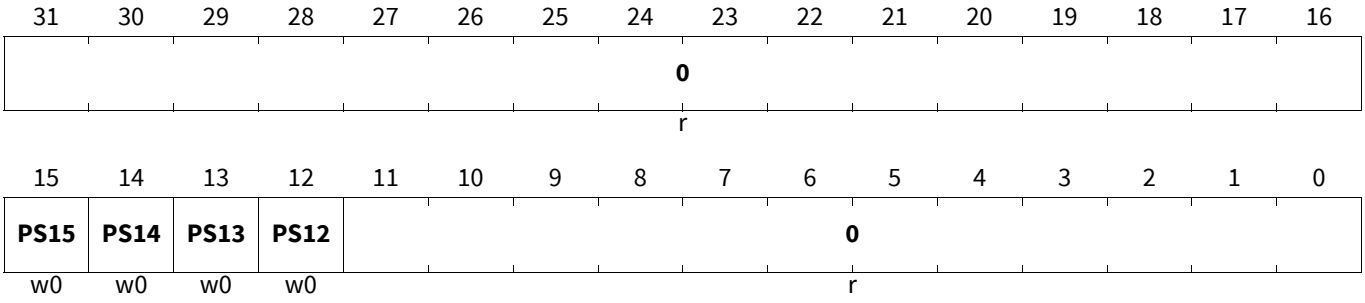
Applies to **P00_OMSR8**
Applies to **P02_OMSR8**
Applies to **P10_OMSR8**
Applies to **P11_OMSR8**
Applies to **P14_OMSR8**
Applies to **P15_OMSR8**
Applies to **P20_OMSR8**
Applies to **P33_OMSR8**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PSx (x=8-11) | write access for enabled masters |
| Otherwise (default) | r0 | PSx (x=8-11) | |

**Port 00 Output Modification Set Register 12**

Register Pn_OMSR12 sets the logic state of Pn.[15:12] port lines

**P00_OMSR12**
**Port 00 Output Modification Set Register 12 ($07C_H$) Application Reset Value: 0000 0000$_H$**
**P11_OMSR12**
**Port 11 Output Modification Set Register 12 ($07C_H$) Application Reset Value: 0000 0000$_H$**
**P20_OMSR12**
**Port 20 Output Modification Set Register 12 ($07C_H$) Application Reset Value: 0000 0000$_H$**
**P33_OMSR12**
**Port 33 Output Modification Set Register 12 ($07C_H$) Application Reset Value: 0000 0000$_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **PSx (x=12-15)** | x | w0 | **Set Bit x**<br>Setting this bit will set the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Sets Pn_OUT.Px |
| **0** | 11:0,<br>31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 153 Access Mode Restrictions sorted by descending priority**

Applies to **P00_OMSR12**
Applies to **P11_OMSR12**
Applies to **P20_OMSR12**
Applies to **P33_OMSR12**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PSx (x=12-15) | write access for enabled masters |
| Otherwise (default) | r0 | PSx (x=12-15) | |

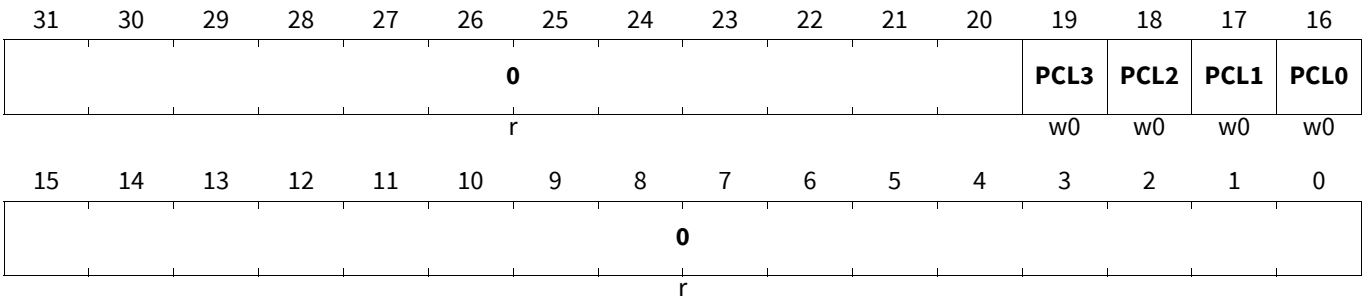
**Port 00 Output Modification Clear Register 0**

The port output modification clear register x, (x = 0, 4, 8, 12) contains control bits to individually clear the logic state of a single port line by manipulating the output register.

*Note: Registers Pn_OMCRx (x = 0, 4, 8, 12) are virtual and does not contain any flip-flop. A read action delivers the value of 0. One 8 or 16-bits write behaves as a 32-bit write padded with zeros.*

Register Pn_OMCR0 clears the logic state of Pn.[3:0] port lines

**P00_OMCR0**
**Port 00 Output Modification Clear Register 0 ($080_H$) Application Reset Value: $0000\ 0000_H$**
**P02_OMCR0**
**Port 02 Output Modification Clear Register 0 ($080_H$) Application Reset Value: $0000\ 0000_H$**
**P10_OMCR0**
**Port 10 Output Modification Clear Register 0 ($080_H$) Application Reset Value: $0000\ 0000_H$**
**P11_OMCR0**
**Port 11 Output Modification Clear Register 0 ($080_H$) Application Reset Value: $0000\ 0000_H$**
**P12_OMCR0**
**Port 12 Output Modification Clear Register 0 ($080_H$) Application Reset Value: $0000\ 0000_H$**
**P14_OMCR0**
**Port 14 Output Modification Clear Register 0 ($080_H$) Application Reset Value: $0000\ 0000_H$**
**P15_OMCR0**
**Port 15 Output Modification Clear Register 0 ($080_H$) Application Reset Value: $0000\ 0000_H$**
**P20_OMCR0**
**Port 20 Output Modification Clear Register 0 ($080_H$) Application Reset Value: $0000\ 0000_H$**
**P21_OMCR0**
**Port 21 Output Modification Clear Register 0 ($080_H$) Application Reset Value: $0000\ 0000_H$**
**P22_OMCR0**
**Port 22 Output Modification Clear Register 0 ($080_H$) Application Reset Value: $0000\ 0000_H$**



| Field | Bits | Type | Description |
|---|---|---|---|
| **PCLx (x=0-3)** | x+16 | w0 | **Clear Bit x**<br>Setting this bit will clear the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Clears Pn_OUT.Px |
| **0** | 15:0, 31:20 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 154 Access Mode Restrictions sorted by descending priority**

Applies to **P00_OMCR0**
Applies to **P02_OMCR0**
Applies to **P10_OMCR0**
Applies to **P11_OMCR0**
Applies to **P12_OMCR0**
Applies to **P14_OMCR0**
Applies to **P15_OMCR0**
Applies to **P20_OMCR0**
Applies to **P21_OMCR0**
Applies to **P22_OMCR0**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PCLx (x=0-3) | write access for enabled masters |
| Otherwise (default) | r0 | PCLx (x=0-3) | |

**P23_OMCR0**
**Port 23 Output Modification Clear Register 0** **($080_H$)** **Application Reset Value: 0000 0000$_H$**
**P32_OMCR0**
**Port 32 Output Modification Clear Register 0** **($080_H$)** **Application Reset Value: 0000 0000$_H$**
**P33_OMCR0**
**Port 33 Output Modification Clear Register 0** **($080_H$)** **Application Reset Value: 0000 0000$_H$**
**P34_OMCR0**
**Port 34 Output Modification Clear Register 0** **($080_H$)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | PCL3 | PCL2 | PCL1 | PCL0 |
| r | | | | | | | | | | | | w0 | w0 | w0 | w0 |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PCLx (x=0-3)** | x+16 | w0 | **Clear Bit x**<br>Setting this bit will clear the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Clears Pn_OUT.Px |
| **0** | 15:0, 31:20 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 155 Access Mode Restrictions sorted by descending priority**

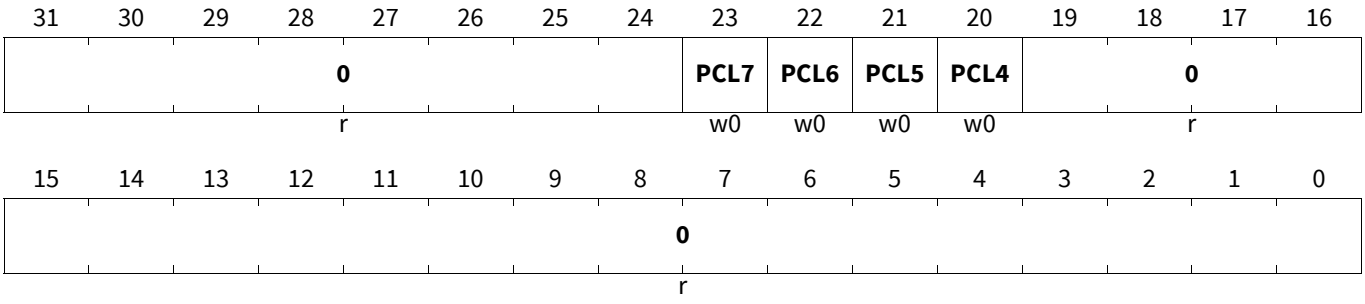
Applies to **P23_OMCR0**
Applies to **P32_OMCR0**
Applies to **P33_OMCR0**
Applies to **P34_OMCR0**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PCLx (x=0-3) | write access for enabled masters |
| Otherwise (default) | r0 | PCLx (x=0-3) | |

### Port 00 Output Modification Clear Register 4

Register Pn_OMCR4 clears the logic state of Pn.[7:4] port lines

**P00_OMCR4**
**Port 00 Output Modification Clear Register 4** **($084_H$)** **Application Reset Value: $0000\ 0000_H$**
**P02_OMCR4**
**Port 02 Output Modification Clear Register 4** **($084_H$)** **Application Reset Value: $0000\ 0000_H$**
**P10_OMCR4**
**Port 10 Output Modification Clear Register 4** **($084_H$)** **Application Reset Value: $0000\ 0000_H$**
**P11_OMCR4**
**Port 11 Output Modification Clear Register 4** **($084_H$)** **Application Reset Value: $0000\ 0000_H$**
**P14_OMCR4**
**Port 14 Output Modification Clear Register 4** **($084_H$)** **Application Reset Value: $0000\ 0000_H$**
**P15_OMCR4**
**Port 15 Output Modification Clear Register 4** **($084_H$)** **Application Reset Value: $0000\ 0000_H$**
**P20_OMCR4**
**Port 20 Output Modification Clear Register 4** **($084_H$)** **Application Reset Value: $0000\ 0000_H$**
**P21_OMCR4**
**Port 21 Output Modification Clear Register 4** **($084_H$)** **Application Reset Value: $0000\ 0000_H$**
**P22_OMCR4**
**Port 22 Output Modification Clear Register 4** **($084_H$)** **Application Reset Value: $0000\ 0000_H$**
**P23_OMCR4**
**Port 23 Output Modification Clear Register 4** **($084_H$)** **Application Reset Value: $0000\ 0000_H$**



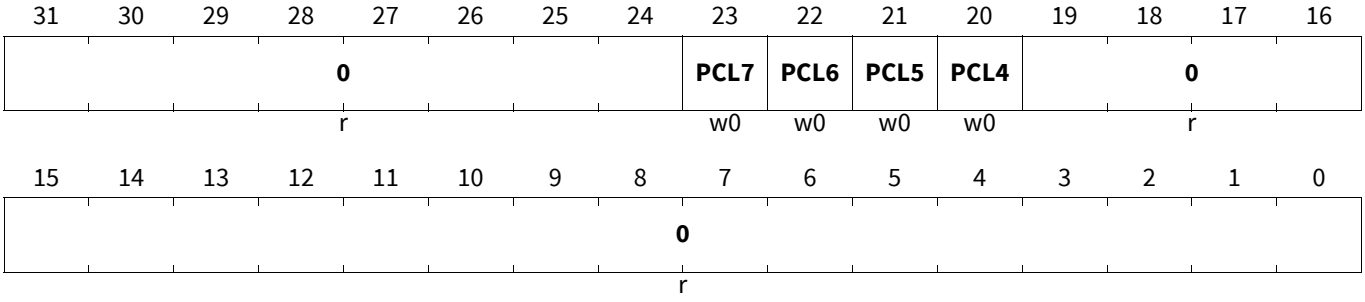

| Field | Bits | Type | Description |
|---|---|---|---|
| **PCLx (x=4-7)** | x+16 | w0 | **Clear Bit x**<br>Setting this bit will clear the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Clears Pn_OUT.Px |
| **0** | 19:0,<br>31:24 | r | **Reserved**<br>Read as 0; should be written with 0 |

**Table 156 Access Mode Restrictions sorted by descending priority**

Applies to **P00_OMCR4**
Applies to **P02_OMCR4**
Applies to **P10_OMCR4**
Applies to **P11_OMCR4**
Applies to **P14_OMCR4**
Applies to **P15_OMCR4**
Applies to **P20_OMCR4**
Applies to **P21_OMCR4**
Applies to **P22_OMCR4**
Applies to **P23_OMCR4**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PCLx (x=4-7) | write access for enabled masters |
| Otherwise (default) | r0 | PCLx (x=4-7) | |

**P32_OMCR4**
**Port 32 Output Modification Clear Register 4 ($084_H$) Application Reset Value: 0000 0000$_H$**
**P33_OMCR4**
**Port 33 Output Modification Clear Register 4 ($084_H$) Application Reset Value: 0000 0000$_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **PCLx (x=4-7)** | x+16 | w0 | **Clear Bit x**<br>Setting this bit will clear the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Clears Pn_OUT.Px |
| **0** | 19:0, 31:24 | r | **Reserved**<br>Read as 0; should be written with 0 |

**Table 157 Access Mode Restrictions sorted by descending priority**

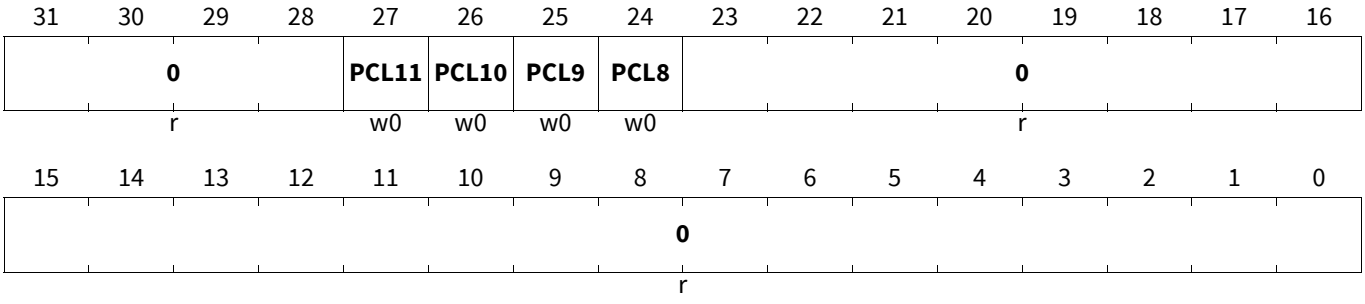
Applies to **P32_OMCR4**
Applies to **P33_OMCR4**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PCLx (x=4-7) | write access for enabled masters |
| Otherwise (default) | r0 | PCLx (x=4-7) | |

**Port 00 Output Modification Clear Register 8**

Register Pn_OMCR8 clears the logic state of Pn.[11:8] port lines

**P00_OMCR8**
**Port 00 Output Modification Clear Register 8 ($088_H$) Application Reset Value: 0000 0000$_H$**
**P02_OMCR8**
**Port 02 Output Modification Clear Register 8 ($088_H$) Application Reset Value: 0000 0000$_H$**
**P10_OMCR8**
**Port 10 Output Modification Clear Register 8 ($088_H$) Application Reset Value: 0000 0000$_H$**
**P11_OMCR8**
**Port 11 Output Modification Clear Register 8 ($088_H$) Application Reset Value: 0000 0000$_H$**
**P14_OMCR8**
**Port 14 Output Modification Clear Register 8 ($088_H$) Application Reset Value: 0000 0000$_H$**
**P15_OMCR8**
**Port 15 Output Modification Clear Register 8 ($088_H$) Application Reset Value: 0000 0000$_H$**
**P20_OMCR8**
**Port 20 Output Modification Clear Register 8 ($088_H$) Application Reset Value: 0000 0000$_H$**
**P33_OMCR8**
**Port 33 Output Modification Clear Register 8 ($088_H$) Application Reset Value: 0000 0000$_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **PCLx (x=8-11)** | x+16 | w0 | **Clear Bit x**<br>Setting this bit will clear the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Clears Pn_OUT.Px |
| **0** | 23:0, 31:28 | r | **Reserved**<br>Read as 0; should be written with 0 |

**Table 158 Access Mode Restrictions sorted by descending priority**

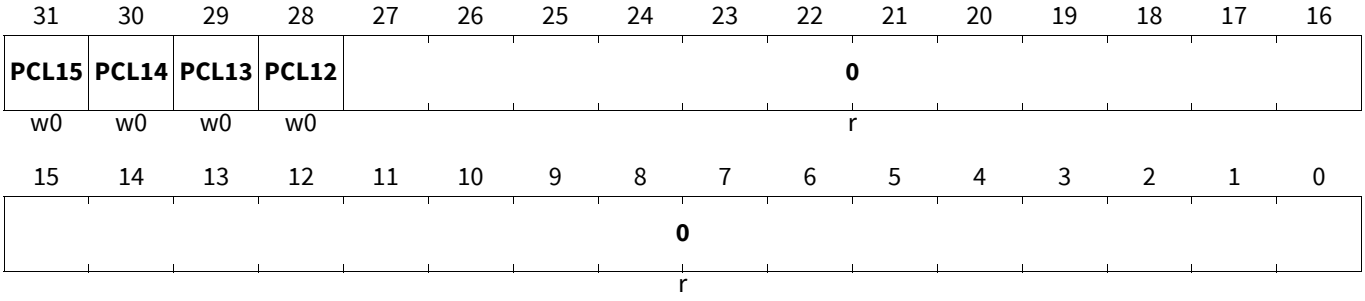
Applies to **P00_OMCR8**
Applies to **P02_OMCR8**
Applies to **P10_OMCR8**
Applies to **P11_OMCR8**
Applies to **P14_OMCR8**
Applies to **P15_OMCR8**
Applies to **P20_OMCR8**
Applies to **P33_OMCR8**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PCLx (x=8-11) | write access for enabled masters |
| Otherwise (default) | r0 | PCLx (x=8-11) | |

**Port 00 Output Modification Clear Register 12**

Register Pn_OMCR12 clears the logic state of Pn.[15:12] port lines

**P00_OMCR12**
**Port 00 Output Modification Clear Register 12 ($08C_H$) Application Reset Value: 0000 0000$_H$**
**P11_OMCR12**
**Port 11 Output Modification Clear Register 12 ($08C_H$) Application Reset Value: 0000 0000$_H$**
**P20_OMCR12**
**Port 20 Output Modification Clear Register 12 ($08C_H$) Application Reset Value: 0000 0000$_H$**
**P33_OMCR12**
**Port 33 Output Modification Clear Register 12 ($08C_H$) Application Reset Value: 0000 0000$_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **PCLx (x=12-15)** | x+16 | w0 | **Clear Bit x**<br>Setting this bit will clear the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Clears Pn_OUT.Px |
| **0** | 27:0 | r | **Reserved**<br>Read as 0; should be written with 0 |

**Table 159 Access Mode Restrictions sorted by descending priority**

Applies to **P00_OMCR12**
Applies to **P11_OMCR12**
Applies to **P20_OMCR12**
Applies to **P33_OMCR12**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PCLx (x=12-15) | write access for enabled masters |
| Otherwise (default) | r0 | PCLx (x=12-15) | |

### Port 00 Output Modification Set Register

The port output modification set register contains control bits that make it possible to individually set the logic state of a single port line by manipulating the output register.

*Note: Register Pn_OMSR is virtual and does not contain any flip-flop. A read action delivers the value of 0. One 8 or 16-bits write behaves as a 32-bit write padded with zeros.*

**P00_OMSR**
**Port 00 Output Modification Set Register** **($090_H$)** **Application Reset Value: 0000 $0000_H$**
**P11_OMSR**
**Port 11 Output Modification Set Register** **($090_H$)** **Application Reset Value: 0000 $0000_H$**
**P20_OMSR**
**Port 20 Output Modification Set Register** **($090_H$)** **Application Reset Value: 0000 $0000_H$**
**P33_OMSR**
**Port 33 Output Modification Set Register** **($090_H$)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PS15 | PS14 | PS13 | PS12 | PS11 | PS10 | PS9 | PS8 | PS7 | PS6 | PS5 | PS4 | PS3 | PS2 | PS1 | PS0 |
| w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PSx (x=0-15)** | x | w0 | **Set Bit x**<br>Setting this bit will set the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Sets Pn_OUT.Px |
| **0** | 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 160 Access Mode Restrictions sorted by descending priority**

Applies to **P00_OMSR**
Applies to **P11_OMSR**
Applies to **P20_OMSR**
Applies to **P33_OMSR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PSx (x=0-15) | write access for enabled masters |
| Otherwise (default) | r0 | PSx (x=0-15) | |

**P02_OMSR**
**Port 02 Output Modification Set Register** **($090_H$)** **Application Reset Value: 0000 0000$_H$**
**P10_OMSR**
**Port 10 Output Modification Set Register** **($090_H$)** **Application Reset Value: 0000 0000$_H$**
**P14_OMSR**
**Port 14 Output Modification Set Register** **($090_H$)** **Application Reset Value: 0000 0000$_H$**
**P15_OMSR**
**Port 15 Output Modification Set Register** **($090_H$)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **PS11** | **PS10** | **PS9** | **PS8** | **PS7** | **PS6** | **PS5** | **PS4** | **PS3** | **PS2** | **PS1** | **PS0** |
| r | r | r | r | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PSx (x=0-11)** | x | w0 | **Set Bit x**<br>Setting this bit will set the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Sets Pn_OUT.Px |
| **0** | 15, 14, 13, 12, 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 161 Access Mode Restrictions sorted by descending priority**

Applies to **P02_OMSR**
Applies to **P10_OMSR**
Applies to **P14_OMSR**
Applies to **P15_OMSR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PSx (x=0-11) | write access for enabled masters |
| Otherwise (default) | r0 | PSx (x=0-11) | |

**P12_OMSR**
**Port 12 Output Modification Set Register (090$_H$) Application Reset Value: 0000 0000$_H$**
**P34_OMSR**
**Port 34 Output Modification Set Register (090$_H$) Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | PS3 | PS2 | PS1 | PS0 |
| r | r | r | r | r | r | r | r | r | r | r | r | w0 | w0 | w0 | w0 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PSx (x=0-3)** | x | w0 | **Set Bit x**<br>Setting this bit will set the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Sets Pn_OUT.Px |
| **0** | 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 162 Access Mode Restrictions sorted by descending priority**

Applies to **P12_OMSR**
Applies to **P34_OMSR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PSx (x=0-3) | write access for enabled masters |
| Otherwise (default) | r0 | PSx (x=0-3) | |

**P21_OMSR**
**Port 21 Output Modification Set Register** **($090_H$)** **Application Reset Value: 0000 0000$_H$**
**P22_OMSR**
**Port 22 Output Modification Set Register** **($090_H$)** **Application Reset Value: 0000 0000$_H$**
**P23_OMSR**
**Port 23 Output Modification Set Register** **($090_H$)** **Application Reset Value: 0000 0000$_H$**
**P32_OMSR**
**Port 32 Output Modification Set Register** **($090_H$)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | PS7 | PS6 | PS5 | PS4 | PS3 | PS2 | PS1 | PS0 |
| r | r | r | r | r | r | r | r | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PSx (x=0-7)** | x | w0 | **Set Bit x**<br>Setting this bit will set the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Sets Pn_OUT.Px |
| **0** | 15, 14, 13, 12, 11, 10, 9, 8, 31:16 | r | **Reserved**<br>Read as 0; should be written with 0. |

**Table 163 Access Mode Restrictions sorted by descending priority**

Applies to **P21_OMSR**
Applies to **P22_OMSR**
Applies to **P23_OMSR**
Applies to **P32_OMSR**

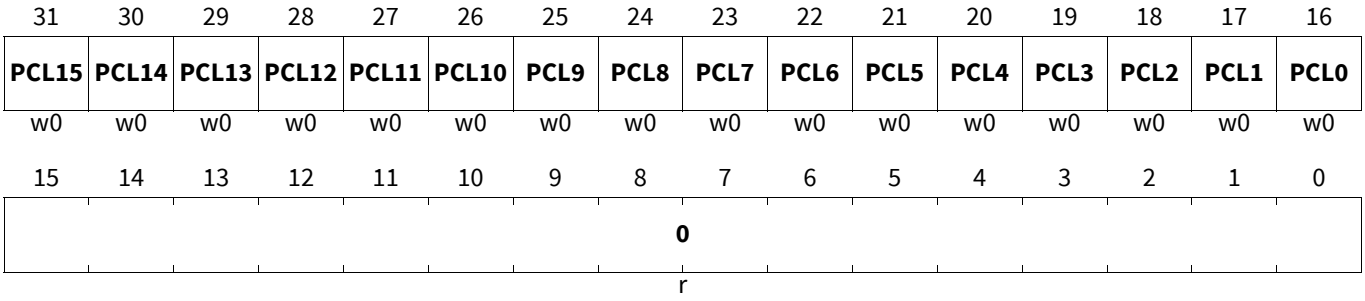
| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PSx (x=0-7) | write access for enabled masters |
| Otherwise (default) | r0 | PSx (x=0-7) | |

**Port 00 Output Modification Clear Register**

The port output modification clear register contains control bits that make it possible to individually clear the logic state of a single port line by manipulating the output register.

*Note: Register Pn_OMCR is virtual and does not contain any flip-flop. A read action delivers the value of 0. One 8 or 16-bits write behaves as a 32-bit write padded with zeros.*

**P00_OMCR**
**Port 00 Output Modification Clear Register** **($094_H$)** **Application Reset Value: $0000\ 0000_H$**
**P11_OMCR**
**Port 11 Output Modification Clear Register** **($094_H$)** **Application Reset Value: $0000\ 0000_H$**
**P20_OMCR**
**Port 20 Output Modification Clear Register** **($094_H$)** **Application Reset Value: $0000\ 0000_H$**
**P33_OMCR**
**Port 33 Output Modification Clear Register** **($094_H$)** **Application Reset Value: $0000\ 0000_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **PCLx (x=0-15)** | x+16 | w0 | **Clear Bit x**<br>Setting this bit will clear the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Clears Pn_OUT.Px. |
| **0** | 15:0 | r | **Reserved**<br>Read as 0; should be written with 0 |

**Table 164 Access Mode Restrictions sorted by descending priority**

Applies to **P00_OMCR**
Applies to **P11_OMCR**
Applies to **P20_OMCR**
Applies to **P33_OMCR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PCLx (x=0-15) | write access for enabled masters |
| Otherwise (default) | r0 | PCLx (x=0-15) | |

**P02_OMCR**
**Port 02 Output Modification Clear Register** **($094_H$)** **Application Reset Value: $0000\ 0000_H$**
**P10_OMCR**
**Port 10 Output Modification Clear Register** **($094_H$)** **Application Reset Value: $0000\ 0000_H$**
**P14_OMCR**
**Port 14 Output Modification Clear Register** **($094_H$)** **Application Reset Value: $0000\ 0000_H$**
**P15_OMCR**
**Port 15 Output Modification Clear Register** **($094_H$)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **PCL11** | **PCL10** | **PCL9** | **PCL8** | **PCL7** | **PCL6** | **PCL5** | **PCL4** | **PCL3** | **PCL2** | **PCL1** | **PCL0** |
| r | r | r | r | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 | w0 |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PCLx (x=0-11)** | x+16 | w0 | **Clear Bit x**<br>Setting this bit will clear the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Clears Pn_OUT.Px. |
| **0** | 15:0, 31, 30, 29, 28 | r | **Reserved**<br>Read as 0; should be written with 0 |

**Table 165 Access Mode Restrictions sorted by descending priority**

Applies to **P02_OMCR**
Applies to **P10_OMCR**
Applies to **P14_OMCR**
Applies to **P15_OMCR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PCLx (x=0-11) | write access for enabled masters |
| Otherwise (default) | r0 | PCLx (x=0-11) | |

**P12_OMCR**
**Port 12 Output Modification Clear Register (094$_H$) Application Reset Value: 0000 0000$_H$**
**P34_OMCR**
**Port 34 Output Modification Clear Register (094$_H$) Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **PCL3** | **PCL2** | **PCL1** | **PCL0** |
| r | r | r | r | r | r | r | r | r | r | r | r | w0 | w0 | w0 | w0 |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

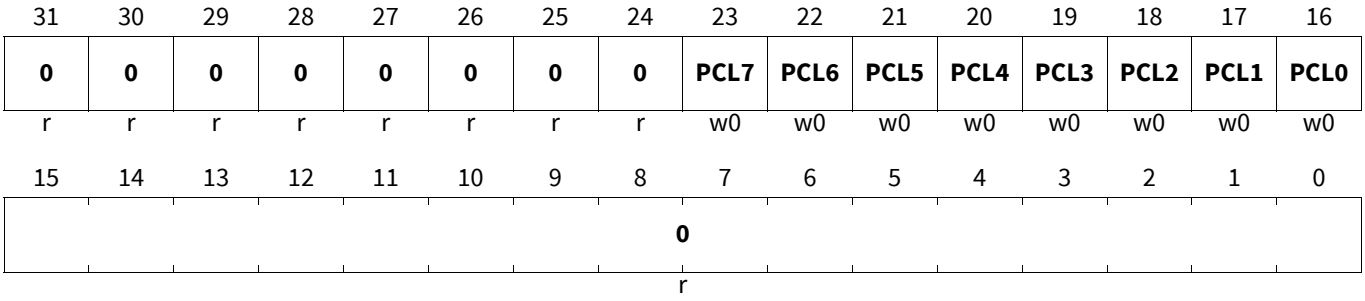
| Field | Bits | Type | Description |
|---|---|---|---|
| **PCLx (x=0-3)** | x+16 | w0 | **Clear Bit x**<br>Setting this bit will clear the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Clears Pn_OUT.Px. |
| **0** | 15:0, 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20 | r | **Reserved**<br>Read as 0; should be written with 0 |

**Table 166 Access Mode Restrictions sorted by descending priority**

Applies to **P12_OMCR**
Applies to **P34_OMCR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PCLx (x=0-3) | write access for enabled masters |
| Otherwise (default) | r0 | PCLx (x=0-3) | |

**P21_OMCR**
**Port 21 Output Modification Clear Register** **($094_H$)** **Application Reset Value: 0000 0000$_H$**
**P22_OMCR**
**Port 22 Output Modification Clear Register** **($094_H$)** **Application Reset Value: 0000 0000$_H$**
**P23_OMCR**
**Port 23 Output Modification Clear Register** **($094_H$)** **Application Reset Value: 0000 0000$_H$**
**P32_OMCR**
**Port 32 Output Modification Clear Register** **($094_H$)** **Application Reset Value: 0000 0000$_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **PCLx (x=0-7)** | x+16 | w0 | **Clear Bit x**<br>Setting this bit will clear the corresponding bit in the port output register Pn_OUT. Read as 0.<br>$0_B$ No operation<br>$1_B$ Clears Pn_OUT.Px. |
| **0** | 15:0, 31, 30, 29, 28, 27, 26, 25, 24 | r | **Reserved**<br>Read as 0; should be written with 0 |

**Table 167 Access Mode Restrictions sorted by descending priority**

Applies to **P21_OMCR**
Applies to **P22_OMCR**
Applies to **P23_OMCR**
Applies to **P32_OMCR**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN | w0 | PCLx (x=0-7) | write access for enabled masters |
| Otherwise (default) | r0 | PCLx (x=0-7) | |

**Port 21 LVDS Pad Control Register x**

The LVDS Pad Control Register controls the RX or TX functions of the LVDS pads. For usage of RX pad, bit field [7:0] are applicable. If used for TX pad, bit field [15:7] apply.

The sleep functionality of the LVDS pads is not controllable via this register. This is exclusively controlled by the HSCT module when this is connected.

The register x controls in general the pad pair 2*x and 2*x+1 of the port n.

Exceptionally when available the pad pair P14.9 and P14.10 is controlled by P14_LPCR5.

***Attention:*** ***The bit field P21_LPCR2.PS configures the pad supply for the LVDS bias distributor for all (not-RIF) LVDS pads and for the oscillator. Therefore even if no LVDS pad is used, this field has to be configured to the correct pad supply level.***

**P21_LPCRx (x=2)**
**Port 21 LVDS Pad Control Register x** **($0A0_H$+x*4)** **Reset Value: Table 169**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | | 0 | 0 | PS | 0 | | 0 | | 0 | 0 | 0 |
| r | r | r | r | r | | r | r | rw | r | | r | | r | r | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **PS** | 7 | rw | **Pad Supply Selection**<br>Selects between 5V or 3.3V supply on $V_{EXT}$ for the pad-pair. Used in RX and TX pads!<br>$0_B$ 3.3V supply<br>$1_B$ 5V supply |
| **0** | 0,<br>1,<br>2,<br>5:3,<br>6,<br>8,<br>9,<br>11:10,<br>12,<br>13,<br>14,<br>15,<br>31:16 | r | **Reserved**<br>Read as 0; should be written with 0 |

**Table 168 Access Mode Restrictions of P21_LPCRx (x=2) sorted by descending priority**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Master enabled in ACCEN and Supervisor Mode and ENDINIT | rw | PS | write access for enabled masters |
| Otherwise (default) | r | PS | |

**Table 169 Reset Values of P21_LPCRx (x=2)**

| Reset Type | Reset Value | Note |
|---|---|---|
| After SSW execution | $0000\ 0080_H$ | Initial value of RX depends on trimming |

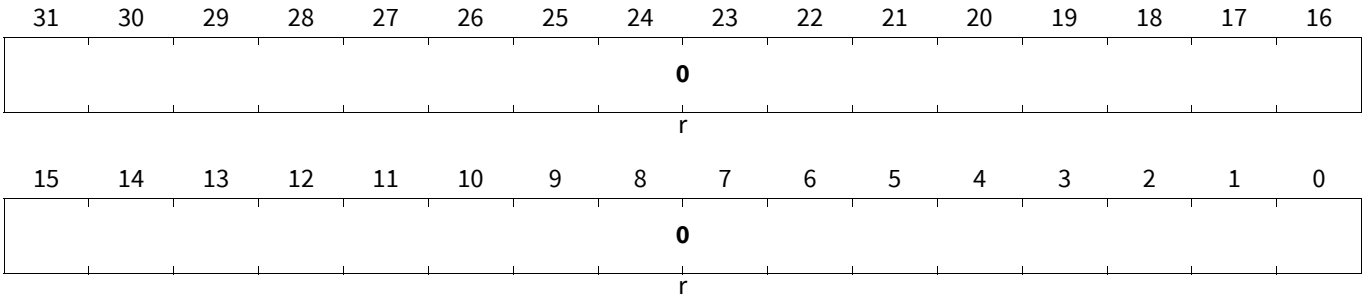
### Port 00 Access Enable Register 1

Each port has its own dedicated ACCEN0 and ACCEN1 registers.

The Access Enable Register 1 controls write[1)] access for transactions with the on chip bus master TAG ID 100000B to 111111B (see On Chip Bus chapter for the products TAG ID <-> master peripheral mapping). The BPI_FPI is prepared for a 6-bit TAG ID. ACCEN1 is not implemented with register bits as the related TAG IDs are not used in this product.

Mapping of TAG IDs to ACCEN1.ENx: EN0 -> TAG ID 100000B, EN1 -> TAG ID 100001B, ... ,EN31 -> TAG ID 111111B.

**P00_ACCEN1**
**Port 00 Access Enable Register 1** **($0F8_H$)** **Application Reset Value: $0000\ 0000_H$**
**P02_ACCEN1**
**Port 02 Access Enable Register 1** **($0F8_H$)** **Application Reset Value: $0000\ 0000_H$**
**P10_ACCEN1**
**Port 10 Access Enable Register 1** **($0F8_H$)** **Application Reset Value: $0000\ 0000_H$**
**P11_ACCEN1**
**Port 11 Access Enable Register 1** **($0F8_H$)** **Application Reset Value: $0000\ 0000_H$**
**P12_ACCEN1**
**Port 12 Access Enable Register 1** **($0F8_H$)** **Application Reset Value: $0000\ 0000_H$**
**P14_ACCEN1**
**Port 14 Access Enable Register 1** **($0F8_H$)** **Application Reset Value: $0000\ 0000_H$**
**P15_ACCEN1**
**Port 15 Access Enable Register 1** **($0F8_H$)** **Application Reset Value: $0000\ 0000_H$**
**P20_ACCEN1**
**Port 20 Access Enable Register 1** **($0F8_H$)** **Application Reset Value: $0000\ 0000_H$**
**P21_ACCEN1**
**Port 21 Access Enable Register 1** **($0F8_H$)** **Application Reset Value: $0000\ 0000_H$**
**P22_ACCEN1**
**Port 22 Access Enable Register 1** **($0F8_H$)** **Application Reset Value: $0000\ 0000_H$**



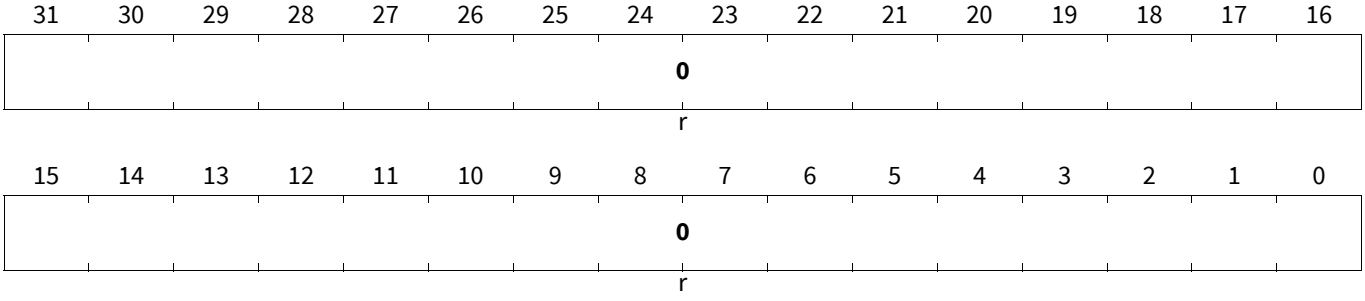

| Field | Bits | Type | Description |
|---|---|---|---|
| **0** | 31:0 | r | **Reserved**<br>Read as 0; should be written with 0 |

**Table 170 Access Mode Restrictions sorted by descending priority**

Applies to **P00_ACCEN1**
Applies to **P02_ACCEN1**
Applies to **P10_ACCEN1**
Applies to **P11_ACCEN1**
Applies to **P12_ACCEN1**
Applies to **P14_ACCEN1**
Applies to **P15_ACCEN1**
Applies to **P20_ACCEN1**
Applies to **P21_ACCEN1**
Applies to **P22_ACCEN1**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Supervisor Mode and Safety ENDINIT | - | See bit field definitions above | write access only for masters with supervisor mode |
| Otherwise (default) | - | See bit field definitions above | |

**P23_ACCEN1**
**Port 23 Access Enable Register 1** **(0F8$_H$)** **Application Reset Value: 0000 0000$_H$**
**P32_ACCEN1**
**Port 32 Access Enable Register 1** **(0F8$_H$)** **Application Reset Value: 0000 0000$_H$**
**P33_ACCEN1**
**Port 33 Access Enable Register 1** **(0F8$_H$)** **Application Reset Value: 0000 0000$_H$**
**P34_ACCEN1**
**Port 34 Access Enable Register 1** **(0F8$_H$)** **Application Reset Value: 0000 0000$_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **0** | 31:0 | r | **Reserved**<br>Read as 0; should be written with 0 |

**Table 171 Access Mode Restrictions sorted by descending priority**

Applies to **P23_ACCEN1**
Applies to **P32_ACCEN1**
Applies to **P33_ACCEN1**
Applies to **P34_ACCEN1**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Supervisor Mode and Safety ENDINIT | - | See bit field definitions above | write access only for masters with supervisor mode |
| Otherwise (default) | - | See bit field definitions above | |

**Port 00 Access Enable Register 0**

Each port has its own dedicated ACCEN0 and ACCEN1 registers.

The Access Enable Register 0 controls write[1)] access for transactions with the on chip bus master TAG ID 000000B to 011111B (see On Chip Bus chapter for the products TAG ID <-> master peripheral mapping). The BPI_FPI is prepared for a 6-bit TAG ID.The registers ACCEN0 and ACCEN1 are providing one enable bit for each possible 6-bit TAG ID encoding.

Mapping of TAG IDs to ACCEN0.ENx: EN0 -> TAG ID 000000B, EN1 -> TAG ID 000001B , ... ,EN31 -> TAG ID 011111B.

1) The BPI_FPI Access Enable functionality controls only write transactions to the kernel registers. Read transactions are not influenced. SW has to take care for destructive/modifying read functionality in kernel registers.

**P00_ACCEN0**
**Port 00 Access Enable Register 0** **(0FC$_H$)** **Application Reset Value: FFFF FFFF$_H$**
**P02_ACCEN0**
**Port 02 Access Enable Register 0** **(0FC$_H$)** **Application Reset Value: FFFF FFFF$_H$**
**P10_ACCEN0**
**Port 10 Access Enable Register 0** **(0FC$_H$)** **Application Reset Value: FFFF FFFF$_H$**
**P11_ACCEN0**
**Port 11 Access Enable Register 0** **(0FC$_H$)** **Application Reset Value: FFFF FFFF$_H$**
**P12_ACCEN0**
**Port 12 Access Enable Register 0** **(0FC$_H$)** **Application Reset Value: FFFF FFFF$_H$**
**P14_ACCEN0**
**Port 14 Access Enable Register 0** **(0FC$_H$)** **Application Reset Value: FFFF FFFF$_H$**
**P15_ACCEN0**
**Port 15 Access Enable Register 0** **(0FC$_H$)** **Application Reset Value: FFFF FFFF$_H$**
**P20_ACCEN0**
**Port 20 Access Enable Register 0** **(0FC$_H$)** **Application Reset Value: FFFF FFFF$_H$**
**P21_ACCEN0**
**Port 21 Access Enable Register 0** **(0FC$_H$)** **Application Reset Value: FFFF FFFF$_H$**
**P22_ACCEN0**
**Port 22 Access Enable Register 0** **(0FC$_H$)** **Application Reset Value: FFFF FFFF$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **EN31** | **EN30** | **EN29** | **EN28** | **EN27** | **EN26** | **EN25** | **EN24** | **EN23** | **EN22** | **EN21** | **EN20** | **EN19** | **EN18** | **EN17** | **EN16** |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **EN15** | **EN14** | **EN13** | **EN12** | **EN11** | **EN10** | **EN9** | **EN8** | **EN7** | **EN6** | **EN5** | **EN4** | **EN3** | **EN2** | **EN1** | **EN0** |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ENx (x=0-31)** | x | rw | **Access Enable for Master TAG ID x**<br>This bit enables write access to the module kernel addresses for transactions with the Master TAG ID n<br>$0_B$ Write access will not be executed<br>$1_B$ Write access will be executed |

**Table 172 Access Mode Restrictions sorted by descending priority**

Applies to **P00_ACCEN0**
Applies to **P02_ACCEN0**
Applies to **P10_ACCEN0**
Applies to **P11_ACCEN0**
Applies to **P12_ACCEN0**
Applies to **P14_ACCEN0**
Applies to **P15_ACCEN0**
Applies to **P20_ACCEN0**
Applies to **P21_ACCEN0**
Applies to **P22_ACCEN0**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Supervisor Mode and Safety ENDINIT | rw | ENx (x=0-31) | write access only for masters with supervisor mode |
| Otherwise (default) | r | ENx (x=0-31) | |

**P23_ACCEN0**
**Port 23 Access Enable Register 0** **(0FC$_H$)** **Application Reset Value: FFFF FFFF$_H$**
**P32_ACCEN0**
**Port 32 Access Enable Register 0** **(0FC$_H$)** **Application Reset Value: FFFF FFFF$_H$**
**P33_ACCEN0**
**Port 33 Access Enable Register 0** **(0FC$_H$)** **Application Reset Value: FFFF FFFF$_H$**
**P34_ACCEN0**
**Port 34 Access Enable Register 0** **(0FC$_H$)** **Application Reset Value: FFFF FFFF$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **EN31** | **EN30** | **EN29** | **EN28** | **EN27** | **EN26** | **EN25** | **EN24** | **EN23** | **EN22** | **EN21** | **EN20** | **EN19** | **EN18** | **EN17** | **EN16** |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **EN15** | **EN14** | **EN13** | **EN12** | **EN11** | **EN10** | **EN9** | **EN8** | **EN7** | **EN6** | **EN5** | **EN4** | **EN3** | **EN2** | **EN1** | **EN0** |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **ENx (x=0-31)** | x | rw | **Access Enable for Master TAG ID x**<br>This bit enables write access to the module kernel addresses for transactions with the Master TAG ID n<br>$0_B$ Write access will not be executed<br>$1_B$ Write access will be executed |

**Table 173 Access Mode Restrictions sorted by descending priority**

Applies to **P23_ACCEN0**
Applies to **P32_ACCEN0**
Applies to **P33_ACCEN0**
Applies to **P34_ACCEN0**

| Mode Name | Access Mode | | Description |
|---|---|---|---|
| Supervisor Mode and Safety ENDINIT | rw | ENx (x=0-31) | write access only for masters with supervisor mode |
| Otherwise (default) | r | ENx (x=0-31) | |

## 14.4 Connectivity

The connectivity of the Ports is documented in the Pinning documentation of each device.

## 14.5 Revision History

**Table 174 Revision History V1.8.20 to the latest revision**

| Reference | Changes to Previous Version | Comment |
|---|---|---|
| **V1.8.20** | | |
| **Page 98** | Revision History entries up to V1.8.19 removed. | |
| **Page 32** | Removed confusing phrase “, only input selection apply.” from register IOCRx from bitfield description of PC. | |
| – | Only cosmetic change: register documentation generator merges more reserved bit fields (e.g. “0” or “1” bit fields). | |
| **V1.8.21** | | |
| – | No content of this Appx changed. TC3Ax Appx added to delivery package and change in Feature List of family chapter. | |

# 15 Safety Management Unit (SMU)

This chapter describes the Safety Management Unit (short SMU) module of the TC35x.

## 15.1 TC35x Specific IP Configuration

See features in family spec.

## 15.2 TC35x Specific Register Set

### SMU_core Specific Register Set

#### Register Address Space Table

**Table 175 Register Address Space - SMU**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| SMU | $F0036800_H$ | $F0036FFF_H$ | FPI slave interface |

#### Register Overview Table

**Table 176 Register Overview - SMU (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SMU_CLC | Clock Control Register | $000_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| SMU_ID | Module Identification Register | $008_H$ | U,SV | BE | Application Reset | See Family Spec |
| SMU_CMD | Command Register | $020_H$ | U,SV | SV,P,32 | Application Reset | See Family Spec |
| SMU_STS | Status Register | $024_H$ | U,SV | SV,P,32 | Application Reset | See Family Spec |
| SMU_FSP | Fault Signaling Protocol | $028_H$ | U,SV | SV,P,SE,32 | PowerOn Reset | See Family Spec |
| SMU_AGC | Alarm Global Configuration | $02C_H$ | U,SV | SV,P,SE,32 | Application Reset | See Family Spec |
| SMU_RTC | Recovery Timer Configuration | $030_H$ | U,SV | SV,P,SE,32 | Application Reset | See Family Spec |
| SMU_KEYS | Key Register | $034_H$ | U,SV | SV,P,SE,32 | Application Reset | See Family Spec |
| SMU_DBG | Debug Register | $038_H$ | U,SV | BE | PowerOn Reset | See Family Spec |

**Table 176 Register Overview - SMU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SMU_PCTL | Port Control | $03C_H$ | U,SV | SV,P,SE,32 | PowerOn Reset | See Family Spec |
| SMU_AFCNT | Alarm and Fault Counter | $040_H$ | U,SV | BE | PowerOn Reset | See Family Spec |
| SMU_RTAC00 | Recovery Timer 0 Alarm Configuration 0 | $060_H$ | U,SV | SV,P,SE,32 | Application Reset | See Family Spec |
| SMU_RTAC01 | Recovery Timer 0 Alarm Configuration 1 | $064_H$ | U,SV | SV,P,SE,32 | Application Reset | See Family Spec |
| SMU_RTAC10 | Recovery Timer 1 Alarm Configuration 0 | $068_H$ | U,SV | SV,P,SE,32 | Application Reset | See Family Spec |
| SMU_RTAC11 | Recovery Timer 1 Alarm Configuration 1 | $06C_H$ | U,SV | SV,P,SE,32 | Application Reset | See Family Spec |
| SMU_AEX | Alarm Executed Status Register | $070_H$ | U,SV | BE | Application Reset | See Family Spec |
| SMU_AEXCLR | Alarm Executed Status Clear Register | $074_H$ | U,SV | SV,P,SE,32 | Application Reset | See Family Spec |
| SMU_AGiCFj (i=0-1;j=0-2) (i=2;j=0-2) (i=3-5;j=0-2) (i=6;j=0-2) (i=7;j=0-2) (i=8;j=0-2) (i=9;j=0-2) (i=10;j=0-2) (i=11;j=0-2) | Alarm Configuration Register | $100_H$+i*12+j*4 | U,SV | SV,P,SE,32 | Application Reset | 5 |
| SMU_AGiFSP (i=0-11) | SMU_core FSP Configuration Register | $190_H$+i*4 | U,SV | SV,P,SE,32 | Application Reset | 10 |
| SMU_AGi (i=0-11) | Alarm Status Register | $1C0_H$+i*4 | U,SV | SV,P,SE,32 | Application Reset | 15 |
| SMU_ADi (i=0-11) | Alarm Debug Register | $200_H$+i*4 | U,SV | BE | PowerOn Reset | 19 |
| SMU_RMCTL | Register Monitor Control | $300_H$ | U,SV | SV,P,SE,32 | Application Reset | See Family Spec |

**Table 176 Register Overview - SMU (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SMU_RMEF | Register Monitor Error Flags | $304_H$ | U,SV | SV,P,SE,32 | Application Reset | See Family Spec |
| SMU_RMSTS | Register Monitor Self Test Status | $308_H$ | U,SV | SV,P,SE,32 | Application Reset | See Family Spec |
| SMU_OCS | OCDS Control and Status | $7E8_H$ | U,SV | SV,P,OEN | Debug Reset | See Family Spec |
| SMU_ACCEN1 | SMU_core Access Enable Register 1 | $7F8_H$ | U,SV | BE | Application Reset | See Family Spec |
| SMU_ACCEN0 | SMU_core Access Enable Register 0 | $7FC_H$ | U,SV | SV,SE | Application Reset | See Family Spec |

**SMU_stdby Specific Register Set**

For SMU_stdby specific register set refer to the Power Management System chapter.

## 15.3 TC35x Specific Registers

## 15.3.1 TC35x Specific Registers

### 15.3.1.1 FPI slave interface

**Alarm Configuration Register**

**SMU_AGiCFj (i=0-1;j=0-2)**
**Alarm Configuration Register** **($100_H$+i*12+j*4)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **CF24** | **CF23** | **CF22** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | rw | rw | rw | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **CF14** | **CF13** | **CF12** | **CF11** | **CF10** | **CF9** | **CF8** | **CF7** | **CF6** | **CF5** | **CF4** | **0** | **CF2** | **CF1** | **CF0** |
| r | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | r | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CFz (z=0-2,4-14,22-24)** | z | rw | **Configuration flag x (x=0-2) for alarm z belonging to alarm group i.**<br>The configuration flags 0, 1 and 2 must be used together to define the behavior of the SMU_core when a fault state is reported by the alarm n belonging to this group.<br>$0_B$ Configuration flag x (x=0-2) is set to 0<br>$1_B$ Configuration flag x (x=0-2) is set to 1 |
| **0** | 31, 30, 29, 28, 27, 26, 25, 21, 20, 19, 18, 17, 16, 15, 3 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGiCFj (i=2;j=0-2)**
**Alarm Configuration Register** **($100_H$+i*12+j*4)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **CF24** | **CF23** | **CF22** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | rw | rw | rw | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **CF14** | **CF13** | **CF12** | **CF11** | **CF10** | **CF9** | **CF8** | **CF7** | **CF6** | **CF5** | **CF4** | **0** | **0** | **CF1** | **0** |
| r | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | r | r | rw | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CFz (z=1,4-14,22-24)** | z | rw | **Configuration flag x (x=0-2) for alarm z belonging to alarm group i.**<br>The configuration flags 0, 1 and 2 must be used together to define the behavior of the SMU_core when a fault state is reported by the alarm n belonging to this group.<br>$0_B$ Configuration flag x (x=0-2) is set to 0<br>$1_B$ Configuration flag x (x=0-2) is set to 1 |
| **0** | 31, 30, 29, 28, 27, 26, 25, 21, 20, 19, 18, 17, 16, 15, 3, 2, 0 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGiCFj (i=3-5;j=0-2)**
**Alarm Configuration Register** **($100_H$+i*12+j*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **0** | 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGiCFj (i=6;j=0-2)**
**Alarm Configuration Register** **($100_H$+i*12+j*4)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **CF25** | **CF24** | **CF23** | **0** | **CF21** | **CF20** | **CF19** | **CF18** | **CF17** | **CF16** |
| r | r | r | r | r | r | rw | rw | rw | r | rw | rw | rw | rw | rw | rw |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CF15** | **CF14** | **CF13** | **0** | **0** | **0** | **0** | **CF8** | **CF7** | **CF6** | **CF5** | **CF4** | **CF3** | **CF2** | **CF1** | **CF0** |
| rw | rw | rw | r | r | r | r | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CFz (z=0-8,13-21,23-25)** | z | rw | **Configuration flag x (x=0-2) for alarm z belonging to alarm group i.**<br>The configuration flags 0, 1 and 2 must be used together to define the behavior of the SMU_core when a fault state is reported by the alarm n belonging to this group.<br>$0_B$ Configuration flag x (x=0-2) is set to 0<br>$1_B$ Configuration flag x (x=0-2) is set to 1 |
| **0** | 31, 30, 29, 28, 27, 26, 22, 12, 11, 10, 9 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGiCFj (i=7;j=0-2)**
**Alarm Configuration Register** **($100_H$+i*12+j*4)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CF31** | **CF30** | **CF29** | **CF28** | **CF27** | **CF26** | **CF25** | **CF24** | **CF23** | **CF22** | **CF21** | **CF20** | **0** | **0** | **CF17** | **CF16** |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | r | r | rw | rw |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CF15** | **CF14** | **CF13** | **CF12** | **0** | **0** | **0** | **CF8** | **CF7** | **CF6** | **CF5** | **CF4** | **CF3** | **CF2** | **CF1** | **CF0** |
| rw | rw | rw | rw | r | r | r | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CFz (z=0-8,12-17,20-31)** | z | rw | **Configuration flag x (x=0-2) for alarm z belonging to alarm group i.**<br>The configuration flags 0, 1 and 2 must be used together to define the behavior of the SMU_core when a fault state is reported by the alarm n belonging to this group.<br>$0_B$ Configuration flag x (x=0-2) is set to 0<br>$1_B$ Configuration flag x (x=0-2) is set to 1 |
| **0** | 19, 18, 11, 10, 9 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGiCFj (i=8;j=0,2)**
**Alarm Configuration Register** **($100_H$+i*12+j*4)** **Application Reset Value: 0001 FC00$_H$**
**SMU_AGiCFj (i=8;j=1)**
**Alarm Configuration Register** **($100_H$+i*12+j*4)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CF31** | **CF30** | **CF29** | **CF28** | **CF27** | **CF26** | **CF25** | **0** | **CF23** | **CF22** | **CF21** | **CF20** | **CF19** | **CF18** | **CF17** | **CF16** |
| rw | rw | rw | rw | rw | rw | rw | r | rw | rw | rw | rw | rw | rw | rw | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **0** | **0** | **0** | **CF12** | **CF11** | **CF10** | **CF9** | **CF8** | **CF7** | **CF6** | **CF5** | **CF4** | **CF3** | **CF2** | **CF1** | **CF0** |
| r | r | r | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CFz (z=0-12,16-23,25-31)** | z | rw | **Configuration flag x (x=0-2) for alarm z belonging to alarm group i.**<br>The configuration flags 0, 1 and 2 must be used together to define the behavior of the SMU_core when a fault state is reported by the alarm n belonging to this group.<br>$0_B$ Configuration flag x (x=0-2) is set to 0<br>$1_B$ Configuration flag x (x=0-2) is set to 1 |
| **0** | 24, 15, 14, 13 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGiCFj (i=9;j=0-2)**
**Alarm Configuration Register** **($100_H$+i*12+j*4)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CF31** | **CF30** | **CF29** | **CF28** | **0** | **0** | **0** | **0** | **CF23** | **CF22** | **CF21** | **CF20** | **0** | **0** | **CF17** | **CF16** |
| rw | rw | rw | rw | r | r | r | r | rw | rw | rw | rw | r | r | rw | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **CF15** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **CF5** | **0** | **CF3** | **0** | **CF1** | **CF0** |
| rw | r | r | r | r | r | r | r | r | r | rw | r | rw | r | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CFz (z=0-1,3,5,15-17,20-23,28-31)** | z | rw | **Configuration flag x (x=0-2) for alarm z belonging to alarm group i.**<br>The configuration flags 0, 1 and 2 must be used together to define the behavior of the SMU_core when a fault state is reported by the alarm n belonging to this group.<br>$0_B$ Configuration flag x (x=0-2) is set to 0<br>$1_B$ Configuration flag x (x=0-2) is set to 1 |

| Field | Bits | Type | Description |
|---|---|---|---|
| **0** | 27, 26, 25, 24, 19, 18, 14, 13, 12, 11, 10, 9, 8, 7, 6, 4, 2 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGiCFj (i=10;j=0)**
**Alarm Configuration Register** **($100_H$+i*12+j*4)** **Application Reset Value: 0000 $0000_H$**
**SMU_AGiCFj (i=10;j=1-2)**
**Alarm Configuration Register** **($100_H$+i*12+j*4)** **Application Reset Value: 0003 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **CF22** | **CF21** | **CF20** | **0** | **CF18** | **CF17** | **CF16** |
| r | r | r | r | r | r | r | r | r | rw | rw | rw | r | rw | rw | rw |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CF15** | **CF14** | **CF13** | **CF12** | **CF11** | **CF10** | **CF9** | **CF8** | **CF7** | **CF6** | **CF5** | **CF4** | **CF3** | **CF2** | **CF1** | **CF0** |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CFz (z=0-18,20-22)** | z | rw | **Configuration flag x (x=0-2) for alarm z belonging to alarm group i.**<br>The configuration flags 0, 1 and 2 must be used together to define the behavior of the SMU_core when a fault state is reported by the alarm n belonging to this group.<br>$0_B$ Configuration flag x (x=0-2) is set to 0<br>$1_B$ Configuration flag x (x=0-2) is set to 1 |
| **0** | 31, 30, 29, 28, 27, 26, 25, 24, 23, 19 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGiCFj (i=11;j=0-2)**
**Alarm Configuration Register** **($100_H$+i*12+j*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **CF13** | **CF12** | **0** | **0** | **CF9** | **0** | **CF7** | **CF6** | **CF5** | **CF4** | **CF3** | **CF2** | **CF1** | **CF0** |
| r | r | rw | rw | r | r | rw | r | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CFz (z=0-7,9,12-13)** | z | rw | **Configuration flag x (x=0-2) for alarm z belonging to alarm group i.**<br>The configuration flags 0, 1 and 2 must be used together to define the behavior of the SMU_core when a fault state is reported by the alarm n belonging to this group.<br>$0_B$ Configuration flag x (x=0-2) is set to 0<br>$1_B$ Configuration flag x (x=0-2) is set to 1 |
| **0** | 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 11, 10, 8 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_core FSP Configuration Register**

**SMU_AGiFSP (i=0-1)**
**SMU_core FSP Configuration Register** **($190_H$+i*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **FE24** | **FE23** | **FE22** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | rw | rw | rw | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **FE14** | **FE13** | **FE12** | **FE11** | **FE10** | **FE9** | **FE8** | **FE7** | **FE6** | **FE5** | **FE4** | **0** | **FE2** | **FE1** | **FE0** |
| r | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | r | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **FEz (z=0-2,4-14,22-24)** | z | rw | **Fault signaling configuration flag for alarm z belonging to alarm group i.**<br>$0_B$ FSP disabled for this alarm event<br>$1_B$ FSP enabled for this alarm event |
| **0** | 31, 30, 29, 28, 27, 26, 25, 21, 20, 19, 18, 17, 16, 15, 3 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGiFSP (i=2)**

**SMU_core FSP Configuration Register** **($190_H$+i*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **FE24** | **FE23** | **FE22** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | rw | rw | rw | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **FE14** | **FE13** | **FE12** | **FE11** | **FE10** | **FE9** | **FE8** | **FE7** | **FE6** | **FE5** | **FE4** | **0** | **0** | **FE1** | **0** |
| r | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | r | r | rw | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **FEz (z=1,4-14,22-24)** | z | rw | **Fault signaling configuration flag for alarm z belonging to alarm group i.**<br>$0_B$ FSP disabled for this alarm event<br>$1_B$ FSP enabled for this alarm event |
| **0** | 31, 30, 29, 28, 27, 26, 25, 21, 20, 19, 18, 17, 16, 15, 3, 2, 0 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGiFSP (i=3-5)**

**SMU_core FSP Configuration Register** **($190_H$+i*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **0** | 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGiFSP (i=6)**

**SMU_core FSP Configuration Register** **($190_H$+i*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **FE25** | **FE24** | **FE23** | **0** | **FE21** | **FE20** | **FE19** | **FE18** | **FE17** | **FE16** |
| r | r | r | r | r | r | rw | rw | rw | r | rw | rw | rw | rw | rw | rw |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **FE15** | **FE14** | **FE13** | **0** | **0** | **0** | **0** | **FE8** | **FE7** | **FE6** | **FE5** | **FE4** | **FE3** | **FE2** | **FE1** | **FE0** |
| rw | rw | rw | r | r | r | r | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **FEz (z=0-8,13-21,23-25)** | z | rw | **Fault signaling configuration flag for alarm z belonging to alarm group i.**<br>$0_B$ FSP disabled for this alarm event<br>$1_B$ FSP enabled for this alarm event |
| **0** | 31, 30, 29, 28, 27, 26, 22, 12, 11, 10, 9 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGiFSP (i=7)**

**SMU_core FSP Configuration Register** **($190_H$+i*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **FE31** | **FE30** | **FE29** | **FE28** | **FE27** | **FE26** | **FE25** | **FE24** | **FE23** | **FE22** | **FE21** | **FE20** | **0** | **0** | **FE17** | **FE16** |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | r | r | rw | rw |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **FE15** | **FE14** | **FE13** | **FE12** | **0** | **0** | **0** | **FE8** | **FE7** | **FE6** | **FE5** | **FE4** | **FE3** | **FE2** | **FE1** | **FE0** |
| rw | rw | rw | rw | r | r | r | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **FEz (z=0-8,12-17,20-31)** | z | rw | **Fault signaling configuration flag for alarm z belonging to alarm group i.**<br>$0_B$ FSP disabled for this alarm event<br>$1_B$ FSP enabled for this alarm event |
| **0** | 19, 18, 11, 10, 9 | r | **Reserved**<br>Read as 0; should be written with 0. |

### SMU_AGiFSP (i=8)

**SMU_core FSP Configuration Register** **($190_H$+i*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **FE31** | **FE30** | **FE29** | **FE28** | **FE27** | **FE26** | **FE25** | **0** | **FE23** | **FE22** | **FE21** | **FE20** | **FE19** | **FE18** | **FE17** | **FE16** |
| rw | rw | rw | rw | rw | rw | rw | r | rw | rw | rw | rw | rw | rw | rw | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **0** | **0** | **0** | **FE12** | **FE11** | **FE10** | **FE9** | **FE8** | **FE7** | **FE6** | **FE5** | **FE4** | **FE3** | **FE2** | **FE1** | **FE0** |
| r | r | r | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **FEz (z=0-12,16-23,25-31)** | z | rw | **Fault signaling configuration flag for alarm z belonging to alarm group i.**<br>$0_B$ FSP disabled for this alarm event<br>$1_B$ FSP enabled for this alarm event |
| **0** | 24, 15, 14, 13 | r | **Reserved**<br>Read as 0; should be written with 0. |

### SMU_AGiFSP (i=9)

**SMU_core FSP Configuration Register** **($190_H$+i*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **FE31** | **FE30** | **FE29** | **FE28** | **0** | **0** | **0** | **0** | **FE23** | **FE22** | **FE21** | **FE20** | **0** | **0** | **FE17** | **FE16** |
| rw | rw | rw | rw | r | r | r | r | rw | rw | rw | rw | r | r | rw | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **FE15** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **FE5** | **0** | **FE3** | **0** | **FE1** | **FE0** |
| rw | r | r | r | r | r | r | r | r | r | rw | r | rw | r | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **FEz (z=0-1,3,5,15-17,20-23,28-31)** | z | rw | **Fault signaling configuration flag for alarm z belonging to alarm group i.**<br>$0_B$ FSP disabled for this alarm event<br>$1_B$ FSP enabled for this alarm event |
| **0** | 27, 26, 25, 24, 19, 18, 14, 13, 12, 11, 10, 9, 8, 7, 6, 4, 2 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGiFSP (i=10)**

**SMU_core FSP Configuration Register** **($190_H$+i*4)** **Application Reset Value: 0003 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **FE22** | **FE21** | **FE20** | **0** | **FE18** | **FE17** | **FE16** |
| r | r | r | r | r | r | r | r | r | rw | rw | rw | r | rw | rw | rw |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **FE15** | **FE14** | **FE13** | **FE12** | **FE11** | **FE10** | **FE9** | **FE8** | **FE7** | **FE6** | **FE5** | **FE4** | **FE3** | **FE2** | **FE1** | **FE0** |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **FEz (z=0-18,20-22)** | z | rw | **Fault signaling configuration flag for alarm z belonging to alarm group i.**<br>$0_B$ FSP disabled for this alarm event<br>$1_B$ FSP enabled for this alarm event |
| **0** | 31, 30, 29, 28, 27, 26, 25, 24, 23, 19 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGiFSP (i=11)**

**SMU_core FSP Configuration Register** **($190_H$+i*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **FE13** | **FE12** | **0** | **0** | **FE9** | **0** | **FE7** | **FE6** | **FE5** | **FE4** | **FE3** | **FE2** | **FE1** | **FE0** |
| r | r | rw | rw | r | r | rw | r | rw | rw | rw | rw | rw | rw | rw | rw |

| Field | Bits | Type | Description |
|---|---|---|---|
| **FEz (z=0-7,9,12-13)** | z | rw | **Fault signaling configuration flag for alarm z belonging to alarm group i.**<br>$0_B$ FSP disabled for this alarm event<br>$1_B$ FSP enabled for this alarm event |
| **0** | 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 11, 10, 8 | r | **Reserved**<br>Read as 0; should be written with 0. |

### Alarm Status Register

Refer to Alarm Status Registers for the conditions to set and reset the status flag by software.

**SMU_AGi (i=0-1)**
**Alarm Status Register** **($1C0_H$+i*4)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **SF24** | **SF23** | **SF22** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | rwh | rwh | rwh | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **SF14** | **SF13** | **SF12** | **SF11** | **SF10** | **SF9** | **SF8** | **SF7** | **SF6** | **SF5** | **SF4** | **0** | **SF2** | **SF1** | **SF0** |
| r | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | r | rwh | rwh | rwh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SFz (z=0-2,4-14,22-24)** | z | rwh | **Status flag for alarm z belonging to alarm group i.**<br>$0_B$ Status flag z does not report a fault condition<br>$1_B$ Status flag z reports a fault condition |
| **0** | 31, 30, 29, 28, 27, 26, 25, 21, 20, 19, 18, 17, 16, 15, 3 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGi (i=2)**
**Alarm Status Register** **($1C0_H$+i*4)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **SF24** | **SF23** | **SF22** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | rwh | rwh | rwh | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **SF14** | **SF13** | **SF12** | **SF11** | **SF10** | **SF9** | **SF8** | **SF7** | **SF6** | **SF5** | **SF4** | **0** | **0** | **SF1** | **0** |
| r | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | r | r | rwh | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SFz (z=1,4-14,22-24)** | z | rwh | **Status flag for alarm z belonging to alarm group i.**<br>$0_B$ Status flag z does not report a fault condition<br>$1_B$ Status flag z reports a fault condition |
| **0** | 31, 30, 29, 28, 27, 26, 25, 21, 20, 19, 18, 17, 16, 15, 3, 2, 0 | r | **Reserved**<br>Read as 0; should be written with 0. |

### SMU_AGi (i=3-5)

**Alarm Status Register** **($1C0_H$+i*4)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **0** | 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0 | r | **Reserved**<br>Read as 0; should be written with 0. |

### SMU_AGi (i=6)

**Alarm Status Register** **($1C0_H$+i*4)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **SF25** | **SF24** | **SF23** | **0** | **SF21** | **SF20** | **SF19** | **SF18** | **SF17** | **SF16** |
| r | r | r | r | r | r | rwh | rwh | rwh | r | rwh | rwh | rwh | rwh | rwh | rwh |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **SF15** | **SF14** | **SF13** | **0** | **0** | **0** | **0** | **SF8** | **SF7** | **SF6** | **SF5** | **SF4** | **SF3** | **SF2** | **SF1** | **SF0** |
| rwh | rwh | rwh | r | r | r | r | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SFz (z=0-8,13-21,23-25)** | z | rwh | **Status flag for alarm z belonging to alarm group i.**<br>$0_B$ Status flag z does not report a fault condition<br>$1_B$ Status flag z reports a fault condition |
| **0** | 31, 30, 29, 28, 27, 26, 22, 12, 11, 10, 9 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGi (i=7)**

**Alarm Status Register** **($1C0_H$+i*4)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **SF31** | **SF30** | **SF29** | **SF28** | **SF27** | **SF26** | **SF25** | **SF24** | **SF23** | **SF22** | **SF21** | **SF20** | **0** | **0** | **SF17** | **SF16** |
| rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | r | r | rwh | rwh |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **SF15** | **SF14** | **SF13** | **SF12** | **0** | **0** | **0** | **SF8** | **SF7** | **SF6** | **SF5** | **SF4** | **SF3** | **SF2** | **SF1** | **SF0** |
| rwh | rwh | rwh | rwh | r | r | r | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SFz (z=0-8,12-17,20-31)** | z | rwh | **Status flag for alarm z belonging to alarm group i.**<br>$0_B$ Status flag z does not report a fault condition<br>$1_B$ Status flag z reports a fault condition |
| **0** | 19, 18, 11, 10, 9 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_AGi (i=8)**

**Alarm Status Register** **($1C0_H$+i*4)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **SF31** | **SF30** | **SF29** | **SF28** | **SF27** | **SF26** | **SF25** | **0** | **SF23** | **SF22** | **SF21** | **SF20** | **SF19** | **SF18** | **SF17** | **SF16** |
| rwh | rwh | rwh | rwh | rwh | rwh | rwh | r | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **SF12** | **SF11** | **SF10** | **SF9** | **SF8** | **SF7** | **SF6** | **SF5** | **SF4** | **SF3** | **SF2** | **SF1** | **SF0** |
| r | r | r | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SFz (z=0-12,16-23,25-31)** | z | rwh | **Status flag for alarm z belonging to alarm group i.**<br>$0_B$ Status flag z does not report a fault condition<br>$1_B$ Status flag z reports a fault condition |
| **0** | 24, 15, 14, 13 | r | **Reserved**<br>Read as 0; should be written with 0. |

### SMU_AGi (i=9)

**Alarm Status Register** **($1C0_H$+i*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **SF31** | **SF30** | **SF29** | **SF28** | **0** | **0** | **0** | **0** | **SF23** | **SF22** | **SF21** | **SF20** | **0** | **0** | **SF17** | **SF16** |
| rwh | rwh | rwh | rwh | r | r | r | r | rwh | rwh | rwh | rwh | r | r | rwh | rwh |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **SF15** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **SF5** | **0** | **SF3** | **0** | **SF1** | **SF0** |
| rwh | r | r | r | r | r | r | r | r | r | rwh | r | rwh | r | rwh | rwh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SFz (z=0-1,3,5,15-17,20-23,28-31)** | z | rwh | **Status flag for alarm z belonging to alarm group i.**<br>$0_B$ Status flag z does not report a fault condition<br>$1_B$ Status flag z reports a fault condition |
| **0** | 27, 26, 25, 24, 19, 18, 14, 13, 12, 11, 10, 9, 8, 7, 6, 4, 2 | r | **Reserved**<br>Read as 0; should be written with 0. |

### SMU_AGi (i=10)

**Alarm Status Register** **($1C0_H$+i*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **SF22** | **SF21** | **SF20** | **0** | **SF18** | **SF17** | **SF16** |
| r | r | r | r | r | r | r | r | r | rwh | rwh | rwh | r | rwh | rwh | rwh |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **SF15** | **SF14** | **SF13** | **SF12** | **SF11** | **SF10** | **SF9** | **SF8** | **SF7** | **SF6** | **SF5** | **SF4** | **SF3** | **SF2** | **SF1** | **SF0** |
| rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SFz (z=0-18,20-22)** | z | rwh | **Status flag for alarm z belonging to alarm group i.**<br>$0_B$ Status flag z does not report a fault condition<br>$1_B$ Status flag z reports a fault condition |
| **0** | 31, 30, 29, 28, 27, 26, 25, 24, 23, 19 | r | **Reserved**<br>Read as 0; should be written with 0. |

### SMU_AGi (i=11)

**Alarm Status Register** **($1C0_H$+i*4)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **SF13** | **SF12** | **0** | **0** | **SF9** | **0** | **SF7** | **SF6** | **SF5** | **SF4** | **SF3** | **SF2** | **SF1** | **SF0** |
| r | r | rwh | rwh | r | r | rwh | r | rwh | rwh | rwh | rwh | rwh | rwh | rwh | rwh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SFz (z=0-7,9,12-13)** | z | rwh | **Status flag for alarm z belonging to alarm group i.**<br>$0_B$ Status flag z does not report a fault condition<br>$1_B$ Status flag z reports a fault condition |
| **0** | 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 11, 10, 8 | r | **Reserved**<br>Read as 0; should be written with 0. |

### Alarm Debug Register

*Note: Writing to this register has no effect*

### SMU_ADi (i=0-1)

**Alarm Debug Register** **($200_H$+i*4)** **PowerOn Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **DF24** | **DF23** | **DF22** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | rh | rh | rh | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **DF14** | **DF13** | **DF12** | **DF11** | **DF10** | **DF9** | **DF8** | **DF7** | **DF6** | **DF5** | **DF4** | **0** | **DF2** | **DF1** | **DF0** |
| r | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | r | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **DFz (z=0-2,4-14,22-24)** | z | rh | **Diagnosis flag for alarm z belonging to alarm group i.**<br>The diagnosis registers make a snapshot of the alarm group status registers when either the executed alarm action is a reset or a state machine transition to FAULT state takes place.<br>$0_B$ Status flag z does not report a fault condition<br>$1_B$ Status flag z reports a fault condition |

| Field | Bits | Type | Description |
|---|---|---|---|
| **0** | 31, 30, 29, 28, 27, 26, 25, 21, 20, 19, 18, 17, 16, 15, 3 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_ADi (i=2)**

**Alarm Debug Register** **($200_H$+i*4)** **PowerOn Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **DF24** | **DF23** | **DF22** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | rh | rh | rh | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **DF14** | **DF13** | **DF12** | **DF11** | **DF10** | **DF9** | **DF8** | **DF7** | **DF6** | **DF5** | **DF4** | **0** | **0** | **DF1** | **0** |
| r | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | r | r | rh | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **DFz (z=1,4-14,22-24)** | z | rh | **Diagnosis flag for alarm z belonging to alarm group i.**<br>The diagnosis registers make a snapshot of the alarm group status registers when either the executed alarm action is a reset or a state machine transition to FAULT state takes place.<br>$0_B$ Status flag z does not report a fault condition<br>$1_B$ Status flag z reports a fault condition |
| **0** | 31, 30, 29, 28, 27, 26, 25, 21, 20, 19, 18, 17, 16, 15, 3, 2, 0 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_ADi (i=3-5)**

**Alarm Debug Register** **($200_H$+i*4)** **PowerOn Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| Field | Bits | Type | Description |
|---|---|---|---|
| **0** | 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_ADi (i=6)**
**Alarm Debug Register** **($200_H$+i*4)** **PowerOn Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **DF25** | **DF24** | **DF23** | **0** | **DF21** | **DF20** | **DF19** | **DF18** | **DF17** | **DF16** |
| r | r | r | r | r | r | rh | rh | rh | r | rh | rh | rh | rh | rh | rh |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **DF15** | **DF14** | **DF13** | **0** | **0** | **0** | **0** | **DF8** | **DF7** | **DF6** | **DF5** | **DF4** | **DF3** | **DF2** | **DF1** | **DF0** |
| rh | rh | rh | r | r | r | r | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **DFz (z=0-8,13-21,23-25)** | z | rh | **Diagnosis flag for alarm z belonging to alarm group i.**<br>The diagnosis registers make a snapshot of the alarm group status registers when either the executed alarm action is a reset or a state machine transition to FAULT state takes place.<br>$0_B$ Status flag z does not report a fault condition<br>$1_B$ Status flag z reports a fault condition |
| **0** | 31, 30, 29, 28, 27, 26, 22, 12, 11, 10, 9 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_ADi (i=7)**
**Alarm Debug Register** **($200_H$+i*4)** **PowerOn Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **DF31** | **DF30** | **DF29** | **DF28** | **DF27** | **DF26** | **DF25** | **DF24** | **DF23** | **DF22** | **DF21** | **DF20** | **0** | **0** | **DF17** | **DF16** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | r | r | rh | rh |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **DF15** | **DF14** | **DF13** | **DF12** | **0** | **0** | **0** | **DF8** | **DF7** | **DF6** | **DF5** | **DF4** | **DF3** | **DF2** | **DF1** | **DF0** |
| rh | rh | rh | rh | r | r | r | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **DFz (z=0-8,12-17,20-31)** | z | rh | **Diagnosis flag for alarm z belonging to alarm group i.**<br>The diagnosis registers make a snapshot of the alarm group status registers when either the executed alarm action is a reset or a state machine transition to FAULT state takes place.<br>$0_B$ Status flag z does not report a fault condition<br>$1_B$ Status flag z reports a fault condition |
| **0** | 19, 18, 11, 10, 9 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_ADi (i=8)**

**Alarm Debug Register** **($200_H$+i*4)** **PowerOn Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **DF31** | **DF30** | **DF29** | **DF28** | **DF27** | **DF26** | **DF25** | **0** | **DF23** | **DF22** | **DF21** | **DF20** | **DF19** | **DF18** | **DF17** | **DF16** |
| rh | rh | rh | rh | rh | rh | rh | r | rh | rh | rh | rh | rh | rh | rh | rh |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **0** | **0** | **0** | **DF12** | **DF11** | **DF10** | **DF9** | **DF8** | **DF7** | **DF6** | **DF5** | **DF4** | **DF3** | **DF2** | **DF1** | **DF0** |
| r | r | r | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **DFz (z=0-12,16-23,25-31)** | z | rh | **Diagnosis flag for alarm z belonging to alarm group i.**<br>The diagnosis registers make a snapshot of the alarm group status registers when either the executed alarm action is a reset or a state machine transition to FAULT state takes place.<br>$0_B$ Status flag z does not report a fault condition<br>$1_B$ Status flag z reports a fault condition |
| **0** | 24, 15, 14, 13 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_ADi (i=9)**

**Alarm Debug Register** **($200_H$+i*4)** **PowerOn Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **DF31** | **DF30** | **DF29** | **DF28** | **0** | **0** | **0** | **0** | **DF23** | **DF22** | **DF21** | **DF20** | **0** | **0** | **DF17** | **DF16** |
| rh | rh | rh | rh | r | r | r | r | rh | rh | rh | rh | r | r | rh | rh |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **DF15** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **DF5** | **0** | **DF3** | **0** | **DF1** | **DF0** |
| rh | r | r | r | r | r | r | r | r | r | rh | r | rh | r | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **DFz (z=0-1,3,5,15-17,20-23,28-31)** | z | rh | **Diagnosis flag for alarm z belonging to alarm group i.**<br>The diagnosis registers make a snapshot of the alarm group status registers when either the executed alarm action is a reset or a state machine transition to FAULT state takes place.<br>$0_B$ Status flag z does not report a fault condition<br>$1_B$ Status flag z reports a fault condition |
| **0** | 27, 26, 25, 24, 19, 18, 14, 13, 12, 11, 10, 9, 8, 7, 6, 4, 2 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_ADi (i=10)**

**Alarm Debug Register** **($200_H$+i*4)** **PowerOn Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **DF22** | **DF21** | **DF20** | **0** | **DF18** | **DF17** | **DF16** |
| r | r | r | r | r | r | r | r | r | rh | rh | rh | r | rh | rh | rh |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **DF15** | **DF14** | **DF13** | **DF12** | **DF11** | **DF10** | **DF9** | **DF8** | **DF7** | **DF6** | **DF5** | **DF4** | **DF3** | **DF2** | **DF1** | **DF0** |
| rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **DFz (z=0-18,20-22)** | z | rh | **Diagnosis flag for alarm z belonging to alarm group i.**<br>The diagnosis registers make a snapshot of the alarm group status registers when either the executed alarm action is a reset or a state machine transition to FAULT state takes place.<br>$0_B$ Status flag z does not report a fault condition<br>$1_B$ Status flag z reports a fault condition |
| **0** | 31, 30, 29, 28, 27, 26, 25, 24, 23, 19 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SMU_ADi (i=11)**
**Alarm Debug Register** **($200_H$+i*4)** **PowerOn Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **DF13** | **DF12** | **0** | **0** | **DF9** | **0** | **DF7** | **DF6** | **DF5** | **DF4** | **DF3** | **DF2** | **DF1** | **DF0** |
| r | r | rh | rh | r | r | rh | r | rh | rh | rh | rh | rh | rh | rh | rh |

| Field | Bits | Type | Description |
|---|---|---|---|
| **DFz (z=0-7,9,12-13)** | z | rh | **Diagnosis flag for alarm z belonging to alarm group i.**<br>The diagnosis registers make a snapshot of the alarm group status registers when either the executed alarm action is a reset or a state machine transition to FAULT state takes place.<br>$0_B$ Status flag z does not report a fault condition<br>$1_B$ Status flag z reports a fault condition |
| **0** | 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 11, 10, 8 | r | **Reserved**<br>Read as 0; should be written with 0. |

## 15.4 TC35x Specific Alarm Mapping

This section defines the mapping between the alarm signals at the input of the SMU in the TC35x and the alarm. For that purpose alarm groups are defined. There is a one-to-one relationship between an alarm group index ALM<n>[index] signal and the alarm configuration and status registers (AG<n>[index]). A group is made of up to 32 alarms; for convenience some entries may be reserved.

### 15.4.1 TC35x Specific Pre-Alarms

There are situations where it is not necessary to implement configuration and status registers for every internal alarm event; a typical case is a module with multiple SRAMs. For that, alarm inputs, called pre-alarms, are combined together with a logical OR internally in SMU. The result is then connected to the alarm group.

### MTU Pre-Alarm Mapping

**Table 177 MTU Pre-Alarm Mapping**

| Alarm Source | Logic | Alarm Index |
|---|---|---|
| CPU0.DMEM - Correctable error<br>CPU0.DLMU - Correctable error<br>CPU0.DMEM1 - Correctable error | OR | ALM0[9] |
| CPU0.DMEM - Uncorrectable Critical error<br>CPU0.DLMU - Uncorrectable Critical error<br>CPU0.DMEM1 - Uncorrectable Critical error | OR | ALM0[10] |
| CPU0.DMEM - Miscellaneous error<br>CPU0.DLMU - Miscellaneous error<br>CPU0.DMEM1 - Miscellaneous error | OR | ALM0[11] |
| CPU1.DMEM - Correctable error<br>CPU1.DLMU - Correctable error<br>CPU1.DMEM1 - Correctable error | OR | ALM1[9] |
| CPU1.DMEM - Uncorrectable Critical error<br>CPU1.DLMU - Uncorrectable Critical error<br>CPU1.DMEM1 - Uncorrectable Critical error | OR | ALM1[10] |
| CPU1.DMEM - Miscellaneous error<br>CPU1.DLMU - Miscellaneous error<br>CPU1.DMEM1 - Miscellaneous error | OR | ALM1[11] |
| CPU2.DMEM - Correctable error<br>CPU2.DLMU - Correctable error | OR | ALM2[9] |
| CPU2.DMEM - Uncorrectable Critical error<br>CPU2.DLMU - Uncorrectable Critical error | OR | ALM2[10] |
| CPU2.DMEM - Miscellaneous error<br>CPU2.DLMU - Miscellaneous error | OR | ALM2[11] |
| LMU_RAM0 - Correctable error<br>LMU_RAM1 - Correctable error<br>FSI_RAM - Correctable error | OR | ALM7[0] |
| LMU_RAM0 - Uncorrectable critical error<br>LMU_RAM1 - Uncorrectable critical error<br>FSI_RAM - Uncorrectable critical error | OR | ALM7[1] |
| LMU_RAM0 - Miscellaneous error<br>LMU_RAM1 - Miscellaneous error<br>FSI_RAM - Miscellaneous error | OR | ALM7[2] |
| DMA - Correctable error<br>MCDS - Correctable error<br>SCR.XRAM - Correctable error<br>SCR.RAMINT - Correctable error<br>GIGETHERNET.RX0 - Correctable error<br>GIGETHERNET.TX0 - Correctable error | OR | ALM6[19] |

**Table 177 MTU Pre-Alarm Mapping** (cont'd)

| Alarm Source | Logic | Alarm Index |
|---|---|---|
| DMA - Uncorrectable Critical error<br>MCDS - Uncorrectable Critical error<br>SCR.XRAM - Uncorrectable critical error<br>SCR.RAMINT - Uncorrectable Critical error<br>GIGETHERNET.RX0 - Uncorrectable Critical error<br>GIGETHERNET.TX0 - Uncorrectable Critical error | OR | **ALM6[20]** |
| DMA - Miscellaneous error<br>MCDS - Miscellaneous error<br>SCR.XRAM - Miscellaneous error<br>SCR.RAMINT - Miscellaneous error<br>GIGETHERNET.RX0 - Miscellaneous error<br>GIGETHERNET.TX0 - Miscellaneous error | OR | **ALM6[21]** |
| EMEM0 - Correctable error<br>EMEM1 - Correctable error<br>EMEM_XTM - Correctable error | OR | **ALM7[3]** |
| EMEM0 - Uncorrectable Critical error<br>EMEM1 - Uncorrectable Critical error<br>EMEM_XTM - Uncorrectable Critical error | OR | **ALM7[4]** |
| EMEM0 - Miscellaneous error<br>EMEM1 - Miscellaneous error<br>EMEM_XTM - Miscellaneous error | OR | **ALM7[5]** |
| SPU_BUFFER0 - Correctable error<br>SPU_BUFFER1 - Correctable error<br>SPU_CONFIG0 - Correctable error<br>SPU_CONFIG1 - Correctable error<br>HSDPM - Correctable error<br>SPU_FFT_RAM0 - Correctable error<br>SPU_FFT_RAM1 - Correctable error<br>SPU_FFT_RAM2 - Correctable error<br>SPU_FFT_RAM3 - Correctable error<br>SPU_FFT_RAM4 - Correctable error<br>SPU_FFT_RAM5 - Correctable error<br>SPU_FFT_RAM6 - Correctable error<br>SPU_FFT_RAM7 - Correctable error | OR | **ALM7[6]** |

**Table 177 MTU Pre-Alarm Mapping** (cont'd)

| Alarm Source | Logic | Alarm Index |
|---|---|---|
| SPU_BUFFER0 - Uncorrectable Critical error<br>SPU_BUFFER1 - Uncorrectable Critical error<br>SPU_CONFIG0 - Uncorrectable Critical error<br>SPU_CONFIG1 - Uncorrectable critical error<br>HSDPM - Uncorrectable Critical error<br>SPU_FFT_RAM0 - Uncorrectable Critical error<br>SPU_FFT_RAM1 - Uncorrectable Critical error<br>SPU_FFT_RAM2 - Uncorrectable Critical error<br>SPU_FFT_RAM3 - Uncorrectable Critical error<br>SPU_FFT_RAM4 - Uncorrectable Critical error<br>SPU_FFT_RAM5 - Uncorrectable Critical error<br>SPU_FFT_RAM6 - Uncorrectable Critical error<br>SPU_FFT_RAM7 - Uncorrectable Critical error | OR | **ALM7[7]** |
| SPU_BUFFER0 - Miscellaneous error<br>SPU_BUFFER1 - Miscellaneous error<br>SPU_CONFIG0 - Miscellaneous error<br>SPU_CONFIG1 - Miscellaneous error<br>HSDPM - Miscellaneous error<br>SPU_FFT_RAM0 - Miscellaneous error<br>SPU_FFT_RAM1 - Miscellaneous error<br>SPU_FFT_RAM2 - Miscellaneous error<br>SPU_FFT_RAM3 - Miscellaneous error<br>SPU_FFT_RAM4 - Miscellaneous error<br>SPU_FFT_RAM5 - Miscellaneous error<br>SPU_FFT_RAM6 - Miscellaneous error<br>SPU_FFT_RAM7 - Miscellaneous error | OR | **ALM7[8]** |
| CAN.MCAN0 - Correctable error<br>CAN.MCAN1 - Correctable error | OR | **ALM6[16]** |
| CAN.MCAN0 - Uncorrectable critical error<br>CAN.MCAN1 - Uncorrectable critical error | OR | **ALM6[17]** |
| CAN.MCAN0 - Miscellaneous error<br>CAN.MCAN1 - Miscellaneous error | OR | **ALM6[18]** |
| ERAY.OBF0 - Correctable error<br>ERAY.TBF_IBF0 - Correctable error<br>ERAY.MBF0 - Correctable error | OR | **ALM6[13]** |
| ERAY.OBF0 - Uncorrectable Critical error<br>ERAY.TBF_IBF0 - Uncorrectable Critical error<br>ERAY.MBF0 - Uncorrectable Critical error | OR | **ALM6[14]** |
| ERAY.OBF0 - Miscellaneous error<br>ERAY.TBF_IBF0 - Miscellaneous error<br>ERAY.MBF0 - Miscellaneous error | OR | **ALM6[15]** |

### Safety Flip-flop Pre-Alarm Mapping

**Table 178 Safety Flip-flop Pre-Alarm Mapping**

| Alarm Source | Logic | Alarm Index |
|---|---|---|
| MTU - Safety flip-flop uncorrectable error<br>IOM - Safety flip-flop uncorrectable error<br>EMEM - Safety flip-flop uncorrectable error<br>IR - Safety flip-flop uncorrectable error<br>SCU - Safety flip-flop uncorrectable error<br>PMS - Safety flip-flop uncorrectable error<br>DMA - Safety flip-flop uncorrectable error<br>SYS_PLL.PER_PLL - Safety flip-flop uncorrectable error<br>CERBERUS - Safety flip-flop uncorrectable error<br>CCU - Safety flip-flop uncorrectable error<br>SMU_core - Safety flip-flop uncorrectable error | OR | **ALM10[21]** |

### LMU Pre-Alarm Mapping

**Table 179 LMU Pre-Alarm Mapping**

| Alarm Source | Logic | Alarm Index |
|---|---|---|
| LMU.RAM0 - Lockstep Comparator error<br>LMU.RAM1 - Lockstep Comparator error<br>LMU.EMEM0 - Lockstep Comparator error<br>LMU.EMEM1 - Lockstep Comparator error | OR | **ALM7[12]** |
| LMU.RAM0 - Lockstep Control error<br>LMU.RAM1 - Lockstep Control error<br>LMU.EMEM0 - Lockstep Control error<br>LMU.EMEM1 - Lockstep Control error | OR | **ALM7[13]** |
| LMU.RAM0 - ECC error<br>LMU.RAM1 - ECC error<br>LMU.EMEM0 - ECC error<br>LMU.EMEM1 - ECC error | OR | **ALM7[14]** |
| LMU.RAM0 - MPU violation<br>LMU.RAM1 - MPU violation<br>LMU.EMEM0 - MPU violation<br>LMU.EMEM1 - MPU violation | OR | **ALM7[15]** |
| LMU.RAM0 - EDC Read Phase Error<br>LMU.RAM1 - EDC Read Phase Error<br>LMU.EMEM0 - EDC Read Phase Error<br>LMU.EMEM1 - EDC Read Phase Error | OR | **ALM7[16]** |

**Table 179 LMU Pre-Alarm Mapping** (cont'd)

| Alarm Source | Logic | Alarm Index |
|---|---|---|
| LMU.RAM0 - SRI Slave Address Phase Error<br>LMU.RAM1 - SRI Slave Address Phase Error<br>LMU.EMEM0 - SRI Slave Address Phase Error<br>LMU.EMEM1 - SRI Slave Address Phase Error | OR | **ALM11[0]** |
| LMU.RAM0 - SRI Slave Write Data Phase Error<br>LMU.RAM1 -SRI Slave Write Data Phase Error<br>LMU.EMEM0 - SRI Slave Write Data Phase Error<br>LMU.EMEM1 - SRI Slave Write Data Phase Error | OR | **ALM11[1]** |

### Module Access Enable Pre-Alarm Mapping

**Table 180 Module Access Enable Pre-Alarm Mapping**

| Alarm Source | Logic | Alarm Index |
|---|---|---|
| IR - Access Enable error<br>HSM - Access Enable error | OR | **ALM10[22]** |

### EMEM Pre-Alarm Mapping

**Table 181 EMEM Pre-Alarm Mapping**

| Alarm Source | Logic | Alarm Index |
|---|---|---|
| EMEM.EMEM0 - Unexpected Write error<br>EMEM.EMEM1 - Unexpected Write error | OR | **ALM9[20]** |
| EMEM.EMEM0 - SEP Control error<br>EMEM.EMEM1 - SEP Control error | OR | **ALM9[21]** |
| EMEM.EMEM0 - Lockstep Control Logic inputs error<br>EMEM.EMEM1 - Lockstep Control Logic inputs error | OR | **ALM9[22]** |

### PMS Pre-Alarm Mapping

**Table 182 PMS Pre-Alarm Mapping**

| Alarm Source | Logic | Alarm Index |
|---|---|---|
| PMS - Uncorrectable error<br>SMU.SMU_stdby - Safety flip-flop Uncorrectable error | OR | **ALM21[7]** |
| HSM.VDD - Under Voltage<br>HSM.VDDP3 - Under Voltage<br>HSM.VEXT - Under Voltage | OR | **ALM9[17]** |
| HSM.VDD - Over Voltage<br>HSM.VDDP3 - Over Voltage<br>HSM.VEXT - Over Voltage | OR | **ALM9[16]** |

**Table 182 PMS Pre-Alarm Mapping** (cont'd)

| Alarm Source | Logic | Alarm Index |
|---|---|---|
| PMS.VDD - Over voltage<br>PMS.VDDPD - Over voltage<br>PMS.VDDP3 - Over voltage<br>PMS.VDDM - Over voltage<br>PMS.VEXT - Over voltage<br>PMS.VEVRSB - Over voltage | OR | ALM9[3] |
| PMS.VDD - Under voltage<br>PMS.VDDPD - Under voltage<br>PMS.VDDP3 - Under voltage<br>PMS.VDDM - Under voltage<br>PMS.VEXT - Under voltage<br>PMS.VEVRSB - Under voltage | OR | ALM9[5] |
| PMS.EVRC - Short to Low<br>PMS.EVRC - Short to High<br>PMS.EVR33 - Short to Low<br>PMS.EVR33 - Short to High | OR | ALM9[15] |

## 15.4.2 TC35x Specific Alarms

The following tables fully specify the mapping between the alarms provided by the safety mechanisms implemented by the microcontroller and the alarm groups.

In the following tables the column "Safety Mechanism & Error Indication" indicates to which safety mechanism the alarm is related. If multiple safety mechanisms are indicated, the alarm corresponds to the detection of an error by one of the listed safety mechanisms.

For some safety mechanisms different terms are used in the microcontroller documents; the following list provides a guideline between the term used in the alarm tables and the other definitions, in bold the definition used in the alarm tables.

- Register Access Protection or alternatively called Safety Register Protection
  - Purpose: Monitors the master identifier of a given bus-master during a write access to a configuration register. The master identifier is a hard-coded information that is provided during any bus access. If the master identifier is not enabled by the Register Access Protection configuration registers (ACCEN0) the write is aborted. Most of the modules do not provide a dedicated alarm for this event and instead will generate a bus error. Therefore the Register Access Protection is only documented where a dedicated alarm is available.
  - Note: for peripherals that implement memory-mapped SRAMs, the write accesses to the memories are monitored as well.
- Bus-level Memory Protection Unit (MPU) or alternatively called Safety Memory Protection
  - Purpose: Monitors the master identifier and the address of a given bus-master during a write access to a local SRAM. The master identifier is a hard-coded information that is provided during any bus access. If the master identifier is enabled by the Bus-level MPU configuration registers and the address is within the valid address range the write is accepted, otherwise the write is aborted and a Bus-level MPU alarm is issued.
  - The SRAMs monitored are the {PSPR, DSPR, DLMU} SRAMs of each CPU and the LMU SRAMs when available in the product.

### Alarm Mapping related to ALM0 group

**Table 183 Alarm Mapping related to ALM0 group**

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM0[0] | cpu_pfi_pfrwb_0 | Safety Mechanism: Lockstep CPU<br>Alarm: CPU0 Lockstep Comparator Error<br>Alarm Type: Pulse<br>Note: For non-Lockstep CPUs (CPUs where no lockstep is implemented or where the lockstep is disabled by the user), this alarm only covers faults that might happen on the access path to the CPU side PFLASH bank. For more details about the CPU lockstep alarm, please refer to the CPU chapter. |
| ALM0[1] | cpu_pfi_pfrwb_0 | Safety Mechanism: Bus-level Memory Protection Unit / Register Access Protection<br>Alarm: CPU0 Bus-level MPU violation / Access Protection violation<br>Alarm Type: Pulse |
| ALM0[2] | cpu_pfi_pfrwb_0 | Safety Mechanism: PFLASH Read Path Monitor<br>Alarm: CPU0 PFLASH0 Read Path Error<br>Alarm Type: Pulse |
| ALM0[3] | Reserved | Reserved |
| ALM0[4] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU0 PCACHE TAG Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM0[5] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU0 PCACHE TAG Miscellaneous error detection<br>Alarm Type: Level |
| ALM0[6] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU0 PSPR/PCACHE Single bit error correction<br>Alarm Type: Level |
| ALM0[7] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU0 PSPR/PCACHE Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM0[8] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU0 PSPR/PCACHE Miscellaneous critical error detection<br>Alarm Type: Level |
| ALM0[9] | **Page 25** | Safety Mechanism: SRAM Monitor<br>Alarm: CPU0 DSPR/DCACHE/DLMU Single bit error correction<br>Alarm Type: Level |
| ALM0[10] | **Page 25** | Safety Mechanism: SRAM Monitor<br>Alarm: CPU0 DSPR/DCACHE/DLMU Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM0[11] | **Page 25** | Safety Mechanism(s): SRAM Monitor<br>Alarm: CPU0 DSPR/DCACHE/DLMU Miscellaneous error detection<br>Alarm Type: Level |

**Table 183 Alarm Mapping related to ALM0 group** (cont'd)

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM0[12] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU0 DCACHE TAG Single bit error correction<br>Alarm Type: Level |
| ALM0[13] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU0 DCACHE TAG Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM0[14] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU0 DCACHE TAG Miscellaneous error detection<br>Alarm Type: Level |
| ALM0[21:15] | Reserved | Reserved |
| ALM0[22] | cpu_pfi_pfrwb_0 | Safety Mechanism: SRI End-to-End EDC<br>Alarm: CPU0 Instruction Fetch SRI Interface EDC Error<br>Alarm Type: Pulse |
| ALM0[23] | cpu_pfi_pfrwb_0 | Safety Mechanism: SRI End-to-End EDC<br>Alarm: CPU0 Data SRI Interface (Load/Store) EDC Error<br>Alarm Type: Pulse |
| ALM0[24] | cpu_pfi_pfrwb_0 | Safety Mechanism: Exception Monitor<br>Alarm: CPU0 exception (interrupt/trap)<br>Alarm Type: Pulse |
| ALM0[31:25] | Reserved | Reserved |

### Alarm Mapping related to ALM1 group

**Table 184 Alarm Mapping related to ALM1 group**

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM1[0] | cpu_pfi_pfrwb_1 | Safety Mechanism: Lockstep CPU<br>Alarm: CPU1 Lockstep Comparator Error<br>Alarm Type: Pulse<br>Note: For non-Lockstep CPUs (CPUs where no lockstep is implemented or where the lockstep is disabled by the user), this alarm only covers faults that might happen on the access path to the CPU side PFLASH bank. Moreover If the CPU side PFLASH bank does not exist, PFLASH read path lockstep still exist. The alarm can be triggered by setting LCL1 bitfield in SCU_LCLTEST Register. For more details about the CPU lockstep alarm, please refer to the CPU chapter. |
| ALM1[1] | cpu_pfi_pfrwb_1 | Safety Mechanism: Bus-level Memory Protection Unit / Register Access Protection<br>Alarm: CPU1 Bus-level MPU violation / Access Protection violation<br>Alarm Type: Pulse |

**Table 184 Alarm Mapping related to ALM1 group** (cont'd)

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM1[2] | cpu_pfi_pfrwb_1 | Safety Mechanism: PFLASH Read Path Monitor<br>Alarm: CPU1 PFLASH1 Read Path Error<br>Alarm Type: Pulse<br>Note: If the CPU side PFLASH bank does not exist, PFLASH read path lockstep still exist. The alarm can be triggered by setting PLCLT1 bitfield in SCU_LCLTEST Register. |
| ALM1[3] | Reserved | Reserved |
| ALM1[4] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU1 PCACHE TAG Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM1[5] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU1 PCACHE TAG Miscellaneous error detection<br>Alarm Type: Level |
| ALM1[6] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU1 PSPR/PCACHE Single bit error correction<br>Alarm Type: Level |
| ALM1[7] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU1 PSPR/PCACHE Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM1[8] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU1 PSPR/PCACHE Miscellaneous critical error detection<br>Alarm Type: Level |
| ALM1[9] | **Page 25** | Safety Mechanism: SRAM Monitor<br>Alarm: CPU1 DSPR/DCACHE/DLMU Single bit error correction<br>Alarm Type: Level |
| ALM1[10] | **Page 25** | Safety Mechanism: SRAM Monitor<br>Alarm: CPU1 DSPR/DCACHE/DLMU Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM1[11] | **Page 25** | Safety Mechanism(s): SRAM Monitor<br>Alarm: CPU1 DSPR/DCACHE/DLMU Miscellaneous error detection<br>Alarm Type: Level |
| ALM1[12] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU1 DCACHE TAG Single bit error correction<br>Alarm Type: Level |
| ALM1[13] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU1 DCACHE TAG Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM1[14] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU1 DCACHE TAG Miscellaneous error detection<br>Alarm Type: Level |
| ALM1[21:15] | Reserved | Reserved |

**Table 184 Alarm Mapping related to ALM1 group** (cont'd)

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM1[22] | cpu_pfi_pfrwb_1 | Safety Mechanism: SRI End-to-End EDC<br>Alarm: CPU1 Instruction Fetch SRI Interface EDC Error<br>Alarm Type: Pulse |
| ALM1[23] | cpu_pfi_pfrwb_1 | Safety Mechanism: SRI End-to-End EDC<br>Alarm: CPU1 Data SRI Interface (Load/Store) EDC Error<br>Alarm Type: Pulse |
| ALM1[24] | cpu_pfi_pfrwb_1 | Safety Mechanism: Exception Monitor<br>Alarm: CPU1 exception (interrupt/trap)<br>Alarm Type: Pulse |
| ALM1[31:25] | Reserved | Reserved |

### Alarm Mapping related to ALM2 group

**Table 185 Alarm Mapping related to ALM2 group**

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM2[0] | Reserved | Reserved |
| ALM2[1] | cpu_2 | Safety Mechanism: Bus-level Memory Protection Unit / Register Access Protection<br>Alarm: CPU2 Bus-level MPU violation / Access Protection violation<br>Alarm Type: Pulse |
| ALM2[2] | Reserved | Reserved |
| ALM2[3] | Reserved | Reserved |
| ALM2[4] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU2 PCACHE TAG Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM2[5] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU2 PCACHE TAG Miscellaneous error detection<br>Alarm Type: Level |
| ALM2[6] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU2 PSPR/PCACHE Single bit error correction<br>Alarm Type: Level |
| ALM2[7] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU2 PSPR/PCACHE Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM2[8] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU2 PSPR/PCACHE Miscellaneous critical error detection<br>Alarm Type: Level |
| ALM2[9] | **Page 25** | Safety Mechanism: SRAM Monitor<br>Alarm: CPU2 DSPR/DCACHE/DLMU Single bit error correction<br>Alarm Type: Level |

**Table 185 Alarm Mapping related to ALM2 group** (cont'd)

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM2[10] | Page 25 | Safety Mechanism: SRAM Monitor<br>Alarm: CPU2 DSPR/DCACHE/DLMU Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM2[11] | Page 25 | Safety Mechanism(s): SRAM Monitor<br>Alarm: CPU2 DSPR/DCACHE/DLMU Miscellaneous error detection<br>Alarm Type: Level |
| ALM2[12] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU2 DCACHE TAG Single bit error correction<br>Alarm Type: Level |
| ALM2[13] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU2 DCACHE TAG Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM2[14] | MTU | Safety Mechanism: SRAM Monitor<br>Alarm: CPU2 DCACHE TAG Miscellaneous error detection<br>Alarm Type: Level |
| ALM2[21:15] | Reserved | Reserved |
| ALM2[22] | cpu_2 | Safety Mechanism: SRI End-to-End EDC<br>Alarm: CPU2 Instruction Fetch SRI Interface EDC Error<br>Alarm Type: Pulse |
| ALM2[23] | cpu_2 | Safety Mechanism: SRI End-to-End EDC<br>Alarm: CPU2 Data SRI Interface (Load/Store) EDC Error<br>Alarm Type: Pulse |
| ALM2[24] | cpu_2 | Safety Mechanism: Exception Monitor<br>Alarm: CPU2 exception (interrupt/trap)<br>Alarm Type: Pulse |
| ALM2[31:25] | Reserved | Reserved |

### Alarm Mapping related to ALM3 group

**Table 186 Alarm Mapping related to ALM3 group**

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM3[0] | Reserved | Reserved |
| ALM3[1] | Reserved | Reserved |
| ALM3[2] | Reserved | Reserved |
| ALM3[3] | Reserved | Reserved |
| ALM3[14:4] | Reserved | Reserved |
| ALM3[21:15] | Reserved | Reserved |
| ALM3[24:22] | Reserved | Reserved |
| ALM3[31:25] | Reserved | Reserved |

### Alarm Mapping related to ALM4 group

**Table 187 Alarm Mapping related to ALM4 group**

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM4[2:0] | Reserved | Reserved |
| ALM4[3] | Reserved | Reserved |
| ALM4[14:4] | Reserved | Reserved |
| ALM4[21:15] | Reserved | Reserved |
| ALM4[24:22] | Reserved | Reserved |
| ALM4[31:25] | Reserved | Reserved |

### Alarm Mapping related to ALM5 group

**Table 188 Alarm Mapping related to ALM5 group**

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM5[2:0] | Reserved | Reserved |
| ALM5[3] | Reserved | Reserved |
| ALM5[14:4] | Reserved | Reserved |
| ALM5[21:15] | Reserved | Reserved |
| ALM5[24:22] | Reserved | Reserved |
| ALM5[31:25] | Reserved | Reserved |

### Alarm Mapping related to ALM6 group

**Table 189 Alarm Mapping related to ALM6 group**

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM6[0] | MTU | Safety Mechanism: Safety Flip-flop<br>Alarm: Safety flip-flop uncorrectable error detected<br>Alarm Type: Level |
| ALM6[1] | IOM | Safety Mechanism: Safety Flip-flop<br>Alarm: Safety flip-flop uncorrectable error detected<br>Alarm Type: Level |
| ALM6[2] | INT | Safety Mechanism: Safety Flip-flop<br>Alarm: Safety flip-flop uncorrectable error detected<br>Alarm Type: Level |
| ALM6[3] | EMEMWRAPPER | Safety Mechanism: Safety Flip-flop<br>Alarm: Safety flip-flop uncorrectable error detected<br>Alarm Type: Level |
| ALM6[4] | SCU | Safety Mechanism: Safety Flip-flop<br>Alarm: Safety flip-flop uncorrectable error detected<br>Alarm Type: Level |
| ALM6[5] | PMS | Safety Mechanism: Safety Flip-flop<br>Alarm: Safety flip-flop uncorrectable error detected<br>Alarm Type: Level |

**Table 189 Alarm Mapping related to ALM6 group** (cont'd)

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM6[6] | DMA | Safety Mechanism: Safety Flip-flop<br>Alarm: Safety flip-flop uncorrectable error detected<br>Alarm Type: Level |
| ALM6[7] | SMU_CORE | Safety Mechanism: Safety Flip-flop<br>Alarm: Safety flip-flop uncorrectable error detected<br>Alarm Type: Level |
| ALM6[8] | CCU | Safety Mechanism: Safety Flip-flop<br>Alarm: SYS_PLL.PER_PLL - Safety flip-flop uncorrectable error detected<br>Alarm Type: Level |
| ALM6[9] | Reserved | Reserved |
| ALM6[12:10] | Reserved | Reserved |
| ALM6[13] | **Page 27** | Safety Mechanism: SRAM Monitor<br>Alarm: ERAY Single bit error correction<br>Alarm Type: Level |
| ALM6[14] | **Page 27** | Safety Mechanism: SRAM Monitor<br>Alarm: ERAY Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM6[15] | **Page 27** | Safety Mechanism: SRAM Monitor<br>Alarm: ERAY Miscellaneous error detection<br>Alarm Type: Level |
| ALM6[16] | **Page 27** | Safety Mechanism: SRAM Monitor<br>Alarm: CAN Single bit error correction<br>Alarm Type: Level |
| ALM6[17] | **Page 27** | Safety Mechanism: SRAM Monitor<br>Alarm: CAN Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM6[18] | **Page 27** | Safety Mechanism: SRAM Monitor<br>Alarm: CAN Miscellaneous error detection<br>Alarm Type: Level |
| ALM6[19] | **Page 25** | Safety Mechanism: SRAM Monitor<br>Alarm: MISC Single bit error correction<br>Alarm Type: Level |
| ALM6[20] | **Page 26** | Safety Mechanism: SRAM Monitor<br>Alarm: MISC Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM6[21] | **Page 26** | Safety Mechanism: SRAM Monitor<br>Alarm: MISC Miscellaneous error detection<br>Alarm Type: Level |
| ALM6[22] | Reserved | Reserved |
| ALM6[23] | CBS | Safety Mechanism: Safety Flip-flop<br>Alarm: Safety flip-flop uncorrectable error detected<br>Alarm Type: Level |

**Table 189 Alarm Mapping related to ALM6 group** (cont'd)

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM6[24] | CCU | Safety Mechanism: Safety Flip-flop<br>Alarm: Safety flip-flop correctable error detected<br>Alarm Type: Level |
| ALM6[25] | CCU | Safety Mechanism: Safety Flip-flop<br>Alarm: Safety flip-flop uncorrectable error detected<br>Alarm Type: Level |
| ALM6[31:26] | Reserved | Reserved |

### Alarm Mapping related to ALM7 group

**Table 190 Alarm Mapping related to ALM7 group**

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM7[0] | **Page 25** | Safety Mechanism: SRAM Monitor<br>Alarm: LMU/FSI_RAM Single bit error correction<br>Alarm Type: Level |
| ALM7[1] | **Page 25** | Safety Mechanism: SRAM Monitor<br>Alarm: LMU/FSI_RAM Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM7[2] | **Page 25** | Safety Mechanism: SRAM Monitor<br>Alarm: LMU/FSI_RAM Miscellaneous error detection<br>Alarm Type: Level |
| ALM7[3] | **Page 26** | Safety Mechanism: SRAM Monitor<br>Alarm: EMEM Single bit error correction<br>Alarm Type: Level |
| ALM7[4] | **Page 26** | Safety Mechanism: SRAM Monitor<br>Alarm: EMEM Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM7[5] | **Page 26** | Safety Mechanism: SRAM Monitor<br>Alarm: EMEM Miscellaneous error detection<br>Alarm Type: Level |
| ALM7[6] | **Page 26** | Safety Mechanism: SRAM Monitor<br>Alarm: SPU/CIF Single bit error correction<br>Alarm Type: Level |
| ALM7[7] | **Page 27** | Safety Mechanism: SRAM Monitor<br>Alarm: SPU/CIF Uncorrectable critical error detection<br>Alarm Type: Level |
| ALM7[8] | **Page 27** | Safety Mechanism: SRAM Monitor<br>Alarm: SPU/CIF Miscellaneous error detection<br>Alarm Type: Level |
| ALM7[11:9] | Reserved | Reserved |
| ALM7[12] | **Page 28** | Safety Mechanism: LMU Lockstep<br>Alarm: Lockstep Comparator Error<br>Alarm Type: Pulse |

**Table 190 Alarm Mapping related to ALM7 group** (cont'd)

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM7[13] | Page 28 | Safety Mechanism: LMU Lockstep<br>Alarm: Lockstep Control Error<br>Alarm Type: Pulse |
| ALM7[14] | Page 28 | Safety Mechanism: SRAM ECC Monitor<br>Alarm: ECC Error<br>Alarm Type: Pulse |
| ALM7[15] | Page 28 | Safety Mechanism: Bus-level MPU<br>Alarm: Bus-level MPU error<br>Alarm Type: Pulse |
| ALM7[16] | Page 28 | Safety Mechanism: LMU Error Detection Code (EDC)<br>Alarm: EDC Read Phase Error<br>Alarm Type: Pulse |
| ALM7[17] | SRI | Safety Mechanism: Built-in SRI Error Detection<br>Alarm: XBAR0 Bus Error Event<br>Alarm Type: Pulse |
| ALM7[18] | Reserved | Reserved |
| ALM7[19] | Reserved | Reserved |
| ALM7[20] | SBCU | Safety Mechanism: Built-in SPB Error Detection<br>Alarm: SPB Bus Error Event<br>Alarm Type: Pulse |
| ALM7[21] | EBCU | Safety Mechanism: Built-in BBB Error Detection<br>Alarm: BBB Bus Error Event<br>Alarm Type: Pulse |
| ALM7[22] | FSI | Safety Mechanism: PFlash ECC<br>Alarm: PFlash Single Bit Error<br>Alarm Type: Level |
| ALM7[23] | FSI | Safety Mechanism: PFlash ECC<br>Alarm: PFlash Double Bit Error<br>Alarm Type: Level |
| ALM7[24] | FSI | Safety Mechanism: PFlash ECC<br>Alarm: Single Bit Correction Tracking Buffer Full<br>Alarm Type: Level |
| ALM7[25] | FSI | Safety Mechanism: PFlash ECC<br>Alarm: Double Bit Correction Tracking Buffer Full<br>Alarm Type: Level |
| ALM7[26] | FSI | Safety Mechanism: PFlash ECC<br>Alarm: Multiple Bit Error Detection Tracking Buffer Full<br>Alarm Type: Level |
| ALM7[27] | FSI | Safety Mechanism: PFlash ECC<br>Alarm: Zero Bit Error Tracking Buffer Full<br>Alarm Type: Level |

**Table 190 Alarm Mapping related to ALM7 group** (cont'd)

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM7[28] | FSI | Safety Mechanism: PFlash ECC Monitor<br>Alarm: PFlash ECC Error<br>Alarm Type: Level |
| ALM7[29] | FSI | Safety Mechanism: PFlash EDC Monitor<br>Alarm: PFlash EDC Error<br>Alarm Type: Level |
| ALM7[30] | FSI | Safety Mechanism: PFlash Configuration Monitor<br>Alarm: CPU FLASHCON Configuration Error<br>Alarm Type: Level |
| ALM7[31] | FSI | Safety Mechanism: PFlash Configuration Monitor<br>Alarm: Flash Stored Configuration Error<br>Alarm Type: Level |

### Alarm Mapping related to ALM8 group

**Table 191 Alarm Mapping related to ALM8 group**

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM8[0] | SCU | Safety Mechanism: Clock Monitor<br>Alarm: OSC clock frequency out of range<br>Alarm Type: Pulse |
| ALM8[1] | CCU | Safety Mechanism: Clock Monitor<br>Alarm: Back-up clock out-of-range alarm<br>Alarm Type: Level |
| ALM8[2] | CCU | Safety Mechanism: Clock Alive Monitor<br>Alarm: Back-up clock alive alarm<br>Alarm Type: Level |
| ALM8[3] | SCU | Safety Mechanism: PLL loss of lock detection<br>Alarm: System PLL DCO loss of lock event<br>Alarm Type: Pulse |
| ALM8[4] | SCU | Safety Mechanism: PLL loss of lock detection<br>Alarm: Peripheral PLL DCO loss of lock event<br>Alarm Type: Pulse |
| ALM8[5] | SCU | Safety Mechanism: LBIST Safe Reset State<br>Alarm: LBIST Alarm<br>Alarm Type: Level |
| ALM8[6] | SCU | Safety Mechanism: External Alarm<br>Alarm: External Request Unit Alarm 0<br>Alarm Type: Pulse |
| ALM8[7] | SCU | Safety Mechanism: External Alarm<br>Alarm: External Request Unit Alarm 1<br>Alarm Type: Pulse |
| ALM8[8] | SCU | Safety Mechanism: External Alarm<br>Alarm: External Request Unit Alarm 2<br>Alarm Type: Pulse |

**Table 191 Alarm Mapping related to ALM8 group** (cont'd)

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM8[9] | SCU | Safety Mechanism: External Alarm<br>Alarm: External Request Unit Alarm 3<br>Alarm Type: Pulse |
| ALM8[10] | SCU | Safety Mechanism: Watchdog<br>Alarm: CPU0 Watchdog Time-out<br>Alarm Type: Pulse |
| ALM8[11] | SCU | Safety Mechanism: Watchdog<br>Alarm: CPU1 Watchdog Time-out<br>Alarm Type: Pulse |
| ALM8[12] | SCU | Safety Mechanism: Watchdog<br>Alarm: CPU2 Watchdog Time-out<br>Alarm Type: Pulse |
| ALM8[13] | Reserved | Reserved |
| ALM8[15:14] | Reserved | Reserved |
| ALM8[16] | SCU | Safety Mechanism: Watchdog<br>Alarm: Safety Watchdog Time-out<br>Alarm Type: Pulse |
| ALM8[17] | SCU | Safety Mechanism: All Watchdogs<br>Alarm: Watchdog Time-out. This alarm is a logical OR over all watchdog time-out alarms<br>Alarm Type: Pulse |
| ALM8[18] | SCU | Safety Mechanism: Lockstep Dual Rail Monitor<br>Alarm: Dual Rail Error<br>Alarm Type: Pulse |
| ALM8[19] | SCU | Safety Mechanism: Emergency Stop<br>Alarm: External Emergency Stop Signal Event<br>Alarm Type: Pulse |
| ALM8[20] | SCU | Safety Mechanism: Pad Monitor<br>Alarm: Pad Heating Alarm<br>Alarm Type: Pulse<br>Note: This alarm is triggered by the pad-heating enable signal of all core supply-pads. It will also be triggered by the enable signal for initialisation of security sensitive RAMs and TCU test enable signals |
| ALM8[21] | SCU | Safety Mechanism: LBIST Test Mode<br>Alarm: LBIST Test Mode Alarm<br>Alarm Type: Level<br>Note: This alarm is also set if TCU related signals are activated in the SCU causing it to fail |
| ALM8[22] | INT | Safety Mechanism: Interrupt Monitor<br>Alarm: EDC Configuration and Data Path Error<br>Alarm Type: Pulse |
| ALM8[23] | DMA | Safety Mechanism: DMA SRI ECC<br>Alarm: DMA SRI ECC Error<br>Alarm Type: Pulse |

**Table 191 Alarm Mapping related to ALM8 group** (cont'd)

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM8[24] | Reserved | Reserved |
| ALM8[25] | IOM | Safety Mechanism: External Alarm<br>Alarm: Pin Mismatch Indication<br>Alarm Type: Level |
| ALM8[26] | SCU | Safety Mechanism: External Alarm<br>Alarm: External Request Unit Alarm 4<br>Alarm Type: Pulse |
| ALM8[27] | SCU | Safety Mechanism: External Alarm<br>Alarm: External Request Unit Alarm 5<br>Alarm Type: Pulse |
| ALM8[28] | SCU | Safety Mechanism: External Alarm<br>Alarm: External Request Unit Alarm 6<br>Alarm Type: Pulse |
| ALM8[29] | SCU | Safety Mechanism: External Alarm<br>Alarm: External Request Unit Alarm 7<br>Alarm Type: Pulse |
| ALM8[30] | SCU | Safety Mechanism: Core Domain Die Temperature Sensor<br>Alarm: Under Temperature Alarm<br>Alarm Type: Level |
| ALM8[31] | SCU | Safety Mechanism: Core Domain Die Temperature Sensor<br>Alarm: Over Temperature Alarm<br>Alarm Type: Level |

### Alarm Mapping related to ALM9 group

**Table 192 Alarm Mapping related to ALM9 group**

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM9[0] | PMS | Safety Mechanism: Die Temperature Sensor<br>Alarm: Temperature Overflow<br>Alarm Type: Level |
| ALM9[1] | PMS | Safety Mechanism: Die Temperature Sensor<br>Alarm: Temperature Underflow<br>Alarm Type: Level |
| ALM9[2] | Reserved | Reserved |
| ALM9[3] | **Page 30** | Safety Mechanism: Voltage Monitor<br>Alarm: Over-voltage Alarm<br>Alarm Type: Level |
| ALM9[4] | Reserved | Reserved |
| ALM9[5] | **Page 30** | Safety Mechanism: Voltage Monitor<br>Alarm: Under-voltage Alarm<br>Alarm Type: Level |
| ALM9[14:6] | Reserved | Reserved |

**Table 192 Alarm Mapping related to ALM9 group** (cont'd)

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM9[15] | **Page 30** | Safety Mechanism: Voltage Monitor<br>Alarm: Short to Low/High Alarm<br>Alarm Type: Level |
| ALM9[16] | **Page 29** | Safety Mechanism: Voltage Monitor<br>Alarm: Over-voltage Alarm<br>Alarm Type: Level |
| ALM9[17] | **Page 29** | Safety Mechanism: Voltage Monitor<br>Alarm: Under-voltage Alarm<br>Alarm Type: Level |
| ALM9[19:18] | Reserved | Reserved |
| ALM9[20] | **Page 29** | Safety Mechanism: EMEM Monitor<br>Alarm: Unexpected Write to EMEM Alarm<br>Alarm Type: Pulse |
| ALM9[21] | **Page 29** | Safety Mechanism: SEP Control Logic Monitor<br>Alarm: SEP Control Logic Alarm<br>Alarm Type: Pulse |
| ALM9[22] | **Page 29** | Safety Mechanism: SPU Lockstep Control Logic Input Monitor<br>Alarm: SPU Configuration Error Alarm<br>Alarm Type: Pulse |
| ALM9[23] | SPULCKSTP | Safety Mechanism: Lockstep<br>Alarm: Lockstep Comparator Alarm<br>Alarm Type: Pulse |
| ALM9[26:24] | Reserved | Reserved |
| ALM9[27] | Reserved | Reserved |
| ALM9[28] | SPU0 | Safety Mechanism: SPU Safety Monitor<br>Alarm: SPU0 Safety Alarm<br>Alarm Type: Pulse |
| ALM9[29] | SPU1 | Safety Mechanism: SPU Safety Monitor<br>Alarm: SPU1 Safety Alarm<br>Alarm Type: Pulse |
| ALM9[30] | RIF0 | Safety Mechanism: RIF Safety Monitor<br>Alarm: RIF0 Safety Alarm<br>Alarm Type: Pulse |
| ALM9[31] | RIF1 | Safety Mechanism: RIF Safety Monitor<br>Alarm: RIF1 Safety Alarm<br>Alarm Type: Pulse |

### Alarm Mapping related to ALM10 group

**Table 193 Alarm Mapping related to ALM10 group**

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM10[0] | Software | Safety Mechanism: Software Monitor<br>Alarm: Software Alarm 0<br>Alarm Type: Pulse |
| ALM10[1] | Software | Safety Mechanism: Software Monitor<br>Alarm: Software Alarm 1<br>Alarm Type: Pulse |
| ALM10[2] | Software | Safety Mechanism: Software Monitor<br>Alarm: Software Alarm 2<br>Alarm Type: Pulse |
| ALM10[3] | Software | Safety Mechanism: Software Monitor<br>Alarm: Software Alarm 3<br>Alarm Type: Pulse |
| ALM10[4] | Software | Safety Mechanism: Software Monitor<br>Alarm: Software Alarm 4<br>Alarm Type: Pulse |
| ALM10[5] | Software | Safety Mechanism: Software Monitor<br>Alarm: Software Alarm 5<br>Alarm Type: Pulse |
| ALM10[6] | Software | Safety Mechanism: Software Monitor<br>Alarm: Software Alarm 6<br>Alarm Type: Pulse |
| ALM10[7] | Software | Safety Mechanism: Software Monitor<br>Alarm: Software Alarm 7<br>Alarm Type: Pulse |
| ALM10[8] | Software | Safety Mechanism: Software Monitor<br>Alarm: Software Alarm 8<br>Alarm Type: Pulse |
| ALM10[9] | Software | Safety Mechanism: Software Monitor<br>Alarm: Software Alarm 9<br>Alarm Type: Pulse |
| ALM10[10] | Software | Safety Mechanism: Software Monitor<br>Alarm: Software Alarm 10<br>Alarm Type: Pulse |
| ALM10[11] | Software | Safety Mechanism: Software Monitor<br>Alarm: Software Alarm 11<br>Alarm Type: Pulse |
| ALM10[12] | Software | Safety Mechanism: Software Monitor<br>Alarm: Software Alarm 12<br>Alarm Type: Pulse |
| ALM10[13] | Software | Safety Mechanism: Software Monitor<br>Alarm: Software Alarm 13<br>Alarm Type: Pulse |

**Table 193 Alarm Mapping related to ALM10 group** (cont'd)

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM10[14] | Software | Safety Mechanism: Software Monitor<br>Alarm: Software Alarm 14<br>Alarm Type: Pulse |
| ALM10[15] | Software | Safety Mechanism: Software Monitor<br>Alarm: Software Alarm 15<br>Alarm Type: Pulse |
| ALM10[16] | SMU_CORE | Safety Mechanism: Recovery Timer 0<br>Alarm: Timer Time-out<br>Alarm Type: Pulse |
| ALM10[17] | SMU_CORE | Safety Mechanism: Recovery Timer 1<br>Alarm: Timer Time-out<br>Alarm Type: Pulse |
| ALM10[18] | FSP | Safety Mechanism: ErrorPin<br>Alarm: ErrorPin Fault State Activation<br>Alarm Type: Pulse |
| ALM10[19] | Reserved | Reserved |
| ALM10[20] | CCU | Safety Mechanism: Safety Flip-flop<br>Alarm: Safety flip-flop correctable error detected<br>Alarm Type: Level |
| ALM10[21] | **Page 28** | Safety Mechanism: Safety Flip-flop<br>Alarm: Safety flip-flop uncorrectable error detected<br>Alarm Type: Level |
| ALM10[22] | **Page 29** | Safety Mechanism: Access Enable Protection<br>Alarm: Access Enable error<br>Alarm Type: Pulse |
| ALM10[31:23] | Reserved | Reserved |

### Alarm Mapping related to ALM11 group

**Table 194 Alarm Mapping related to ALM11 group**

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM11[0] | **Page 29** | Safety Mechanism: SRI Error Detection Code (EDC)<br>Alarm: EDC Address Phase Error<br>Alarm Type: Pulse |
| ALM11[1] | **Page 29** | Safety Mechanism: SRI Error Detection Code (EDC)<br>Alarm: EDC Write Phase Error<br>Alarm Type: Pulse |
| ALM11[2] | SRI | Safety Mechanism: SRI Error Detection Code (EDC)<br>Alarm: EDC Address Phase Error<br>Alarm Type: Pulse |
| ALM11[3] | SRI | Safety Mechanism: SRI Error Detection Code (EDC)<br>Alarm: EDC Write Phase Error<br>Alarm Type: Pulse |

**Table 194 Alarm Mapping related to ALM11 group** (cont'd)

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM11[4] | DMU | Safety Mechanism: SRI Error Detection Code (EDC)<br>Alarm: EDC Address Phase Error<br>Alarm Type: Pulse |
| ALM11[5] | DMU | Safety Mechanism: SRI Error Detection Code (EDC)<br>Alarm: EDC Write Phase Error<br>Alarm Type: Pulse |
| ALM11[6] | SFIBRIDGE2 | Safety Mechanism: SRI Error Detection Code (EDC)<br>Alarm: EDC Address Phase Error<br>Alarm Type: Pulse |
| ALM11[7] | SFIBRIDGE2 | Safety Mechanism: SRI Error Detection Code (EDC)<br>Alarm: EDC Write Phase Error<br>Alarm Type: Pulse |
| ALM11[8] | Reserved | Reserved |
| ALM11[9] | SFIBRIDGE1 | Safety Mechanism: SRI Error Detection Code (EDC)<br>Alarm: EDC Read Phase Error<br>Alarm Type: Pulse |
| ALM11[10] | Reserved | Reserved |
| ALM11[11] | Reserved | Reserved |
| ALM11[12] | converter_0 | Safety Mechanism: Converter<br>Alarm: Phase Synchronizer Error<br>Alarm Type: Level |
| ALM11[13] | SRI | Safety Mechanism: SRI SOTA Monitor<br>Alarm: SOTA Swap Error<br>Alarm Type: Pulse |
| ALM11[31:14] | Reserved | Reserved |

### Alarm Mapping related to ALM20 group

**Table 195 Alarm Mapping related to ALM20 group**

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM20[3:0] | Reserved | Resreved |
| ALM20[4] | PMS | Safety Mechanism: Voltage Monitor<br>Alarm: VDD Over-voltage Alarm<br>Alarm Type: Level |
| ALM20[5] | PMS | Safety Mechanism: Voltage Monitor<br>Alarm: VDDPD Over-voltage Alarm<br>Alarm Type: Level |
| ALM20[6] | PMS | Safety Mechanism: Voltage Monitor<br>Alarm: VDDP3 Over-voltage Alarm<br>Alarm Type: Level |
| ALM20[7] | PMS | Safety Mechanism: Voltage Monitor<br>Alarm: VDDM Over-voltage Alarm<br>Alarm Type: Level |

**Table 195 Alarm Mapping related to ALM20 group** (cont'd)

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM20[8] | PMS | Safety Mechanism: Voltage Monitor<br>Alarm: VEXT Over-voltage Alarm<br>Alarm Type: Level |
| ALM20[9] | PMS | Safety Mechanism: Voltage Monitor<br>Alarm: VEVRSB Over-voltage Alarm<br>Alarm Type: Level |
| ALM20[10] | PMS | Safety Mechanism: Voltage Monitor<br>Alarm: VDD Under-voltage Alarm<br>Alarm Type: Level |
| ALM20[11] | PMS | Safety Mechanism: Voltage Monitor<br>Alarm: VDDPD Under-voltage Alarm<br>Alarm Type: Level |
| ALM20[12] | PMS | Safety Mechanism: Voltage Monitor<br>Alarm: VDDP3 Under-voltage Alarm<br>Alarm Type: Level |
| ALM20[13] | PMS | Safety Mechanism: Voltage Monitor<br>Alarm: VDDM Under-voltage Alarm<br>Alarm Type: Level |
| ALM20[14] | PMS | Safety Mechanism: Voltage Monitor<br>Alarm: VEXT Under-voltage Alarm<br>Alarm Type: Level |
| ALM20[15] | PMS | Safety Mechanism: Voltage Monitor<br>Alarm: VEVRSB Under-voltage Alarm<br>Alarm Type: Level |
| ALM20[31:16] | Reserved | Reserved |

### Alarm Mapping related to ALM21 group

**Table 196 Alarm Mapping related to ALM21 group**

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM21[0] | hsm | Safety Mechanism: Voltage Monitor<br>Alarm: VDD Under-voltage Alarm<br>Alarm Type: Level |
| ALM21[1] | hsm | Safety Mechanism: Voltage Monitor<br>Alarm: VDDP3 Under-voltage Alarm<br>Alarm Type: Level |
| ALM21[2] | hsm | Safety Mechanism: Voltage Monitor<br>Alarm: VEXT Under-voltage Alarm<br>Alarm Type: Level |
| ALM21[3] | hsm | Safety Mechanism: Voltage Monitor<br>Alarm: VDD Over-voltage Alarm<br>Alarm Type: Level |

**Table 196 Alarm Mapping related to ALM21 group** (cont'd)

| Alarm Index | Module | Safety Mechanism & Alarm Indication |
|---|---|---|
| ALM21[4] | hsm | Safety Mechanism: Voltage Monitor<br>Alarm: VDDP3 over-voltage Alarm<br>Alarm Type: Level |
| ALM21[5] | hsm | Safety Mechanism: Voltage Monitor<br>Alarm: VEXT over-voltage Alarm<br>Alarm Type: Level |
| ALM21[6] | Reserved | Reserved |
| ALM21[7] | **Page 29** | Safety Mechanism: Safety Flip-flop<br>Alarm: Safety flip-flop uncorrectable error detected<br>Alarm Type: Level |
| ALM21[8] | PMS | Safety Mechanism: Die Temperature Sensor<br>Alarm: Temperature underflow<br>Alarm Type: Level |
| ALM21[9] | PMS | Safety Mechanism: Die Temperature Sensor<br>Alarm: Temperature overflow<br>Alarm Type: Level |
| ALM21[10] | PMS | Safety Mechanism: Register Access Protection<br>Alarm: Access Protection violation<br>Alarm Type: Pulse |
| ALM21[11] | PMS | Safety Mechanism: Voltage Monitor<br>Alarm: EVRC Short to Low Alarm<br>Alarm Type: Level |
| ALM21[12] | PMS | Safety Mechanism: Voltage Monitor<br>Alarm: EVRC Short to High Alarm<br>Alarm Type: Level |
| ALM21[13] | PMS | Safety Mechanism: Voltage Monitor<br>Alarm: EV33 Short to Low Alarm<br>Alarm Type: Level |
| ALM21[14] | PMS | Safety Mechanism: Voltage Monitor<br>Alarm: EV33 Short to High Alarm<br>Alarm Type: Level |
| ALM21[15] | CCU | Safety Mechanism: Clock Alive Monitor<br>Alarm: PLLx/fSPB Alive Alarm (provided on fBACK clock with x = 0..2)<br>Alarm Type: Level<br>Note: This alarm is also set if TCU related signals are activated in the CCU causing the clocks to fail |
| ALM21[16] | SMU_CORE | Safety Mechanism: SMU_core Alive Monitor<br>Alarm: SMU_core Alive Alarm<br>Alarm Type: Pulse |
| ALM21[31:17] | Reserved | Reserved |

## 15.5 Connectivity

**Table 197 Connections of SMU**

| Interface Signals | connects | | Description |
|---|---|---|---|
| SMU:FSP(0) | to | P33.8:HWOUT(0) | FSP[1..0] Output Signals - Generated by SMU_core |
| SMU:FSP(1) | to | P33.10:HWOUT(0) | FSP[1..0] Output Signals - Generated by SMU_core |
| SMU:FSP_STS(0) | from | P33.8:IN | FSP Status Input - Shows the actual state of the FSP ErroPin |
| SMU:FSP_STS(1) | from | P33.10:IN | FSP Status Input - Shows the actual state of the FSP ErroPin |
| SMU:RUNSTATE | to | SCU:smu_wdt_run | SMU_core RUN state indication |
| SMU:INT(2:0) | to | INT:smu.INT(2:0) | SMU Service Request |

## 15.6 Revision History

**Table 198 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V4.0.17** | | |
| **Page 42** | Updated description of ALM9[22] | |
| **Page 47** | Updated description of ALM21[0] and ALM21[3] | |
| **Page 29** | Updated PMS Pre-Alarm Mapping table | |
| **Page 47** | Added description of ALM21[6] | |
| **V4.0.18** | | |
| **Page 40** | Updated description of ALM8[20] | |
| **Page 34** | Added ALM2[2] in Alarm Mapping table | |
| **Page 17** | Removed bits [18] and [19] from SMU_AG7*** | |
| **Page 1** | Missing blank fixed | |
| **V4.0.19** | | |
| **Page 25**, **Page 38** | Added FSI_RAM Alarms ALM7[0:2] which were not documented in the previous version | |
| **Page 31** | Added Alarm Types in Alarm Mapping Tables | |
| **Page 2** | Typo fixed, no functional change | |
| **Page 49** | Revision History updated | |
| **V4.0.20** | | |
| – | No functional changes. | |
| **V4.0.21** | | |
| – | No functional changes. | |
| **V4.0.22** | | |
| **Page 32**, **Page 34** | Updated description of ALM1[0], ALM1[2], ALM2[0] and ALM2[2] | |

**Table 198 Revision History** (cont'd)

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V4.0.23** | | |
| **Page 36** | Updated description of ALM6[8] | |

# 16 Interrupt Router (IR)

This chapter supplements the family documentation whith device specific information for TC35x.

The Interrupt Router allocates two address ranges

- Interrupt Router System and OTGM register address range: 2 * 256 byte address range covering the Interrupt Router system registers, ICU control registers and OTGM registers (**Chapter 16.2**)
- SRC register address range: 8 KByte address range covering the Service Request Control registers (**Chapter 16.4**)

## 16.1 TC35x Specific Interrupt Router Configuration

**Table 199 TC35x specific configuration of INT**

| Parameter | INT |
|---|---|
| **Number of Interrupt Service Providers** | 4 |
| **Number of SRB groups** | 3 |

**Table 200 TC35x specific configuration of SRC**

| Parameter | SRC |
|---|---|
| **Number of Service Request Nodes** | 1024 |

## 16.2 TC35x Specific Control Registers

This chapter describes the TC35x specific Interrupt Router system, OTGM and ICU registers

### List of used Access Protection Register abbreviations

- P0 -> ACCEN_SRBx, write protection of the related SRBx register. Number of Service Request Broadcast registers (SRB) and the related ACCEN_SRB registers is equal to the number of implemented TriCore CPUs.
- P1 -> ACCEN_CONFIG, write protection of all SRCx[15:0] and ICUx Error Capture registers (ECRx)
- P2 -> ACCEN_SRC_TOSx, write protects bits [31:16] of all SRCs that are mapped to TOSx (SCR.TOS=x). For each implemented Interrupt Control Unit, one ACCEN_SRC_TOSx register is implemented.

*Note: A violation of the access protection will not be executed (e.g. a write to a ´Px´/ACCEN protected register by an SPB access with a disabled Master TAG-ID). In this case an access protection error is signaled to the SMU. Beside this signaling to the SMU, no other error, interrupt or trap is generated.*

### Interrupt Router Module Registers

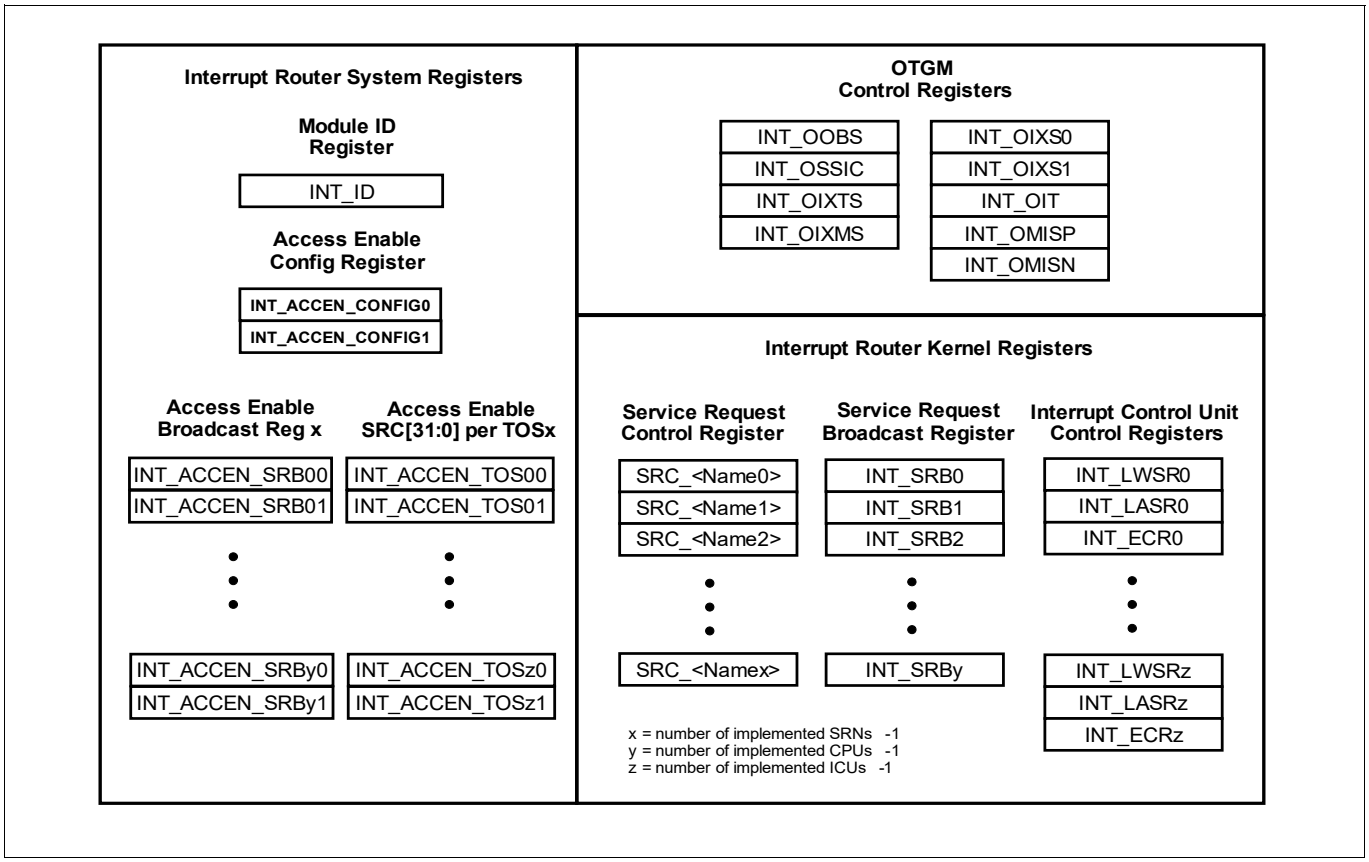



**Figure 5 Interrupt Router module registers (SRC registers are described in Chapter 16.4)**

**Table 201 Register Address Space - INT**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| INT | $F0037000_H$ | $F0037FFF_H$ | IR Status and Control Registers |

**Table 202 Register Overview - INT (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| INT_ID | Module Identification Register | $0008_H$ | U,SV | nBE | Application Reset | See Family Spec |
| INT_SRBx (x=0-2) | Service Request Broadcast Register x | $0010_H$+x *4 | U,SV | SV,P0 | Application Reset | See Family Spec |
| INT_OOBS | OTGM OTGB0/1 Status | $0080_H$ | U,SV | nBE | Application Reset | See Family Spec |
| INT_OSSIC | OTGM SSI Control | $0084_H$ | U,SV | SV | Application Reset | See Family Spec |
| INT_OIXTS | OTGM IRQ MUX Trigger Set Select | $0088_H$ | U,SV | SV | Application Reset | See Family Spec |
| INT_OIXMS | OTGM IRQ MUX Missed IRQ Select | $008C_H$ | U,SV | SV | Application Reset | See Family Spec |
| INT_OIXS0 | OTGM IRQ MUX Select 0 | $0090_H$ | U,SV | SV | Application Reset | See Family Spec |
| INT_OIXS1 | OTGM IRQ MUX Select 1 | $0094_H$ | U,SV | SV | Application Reset | See Family Spec |
| INT_OIT | OTGM IRQ Trace | $00A0_H$ | U,SV | SV | Application Reset | 5 |
| INT_OMISP | OTGM MCDS I/F Sensitivity Posedge | $00A4_H$ | U,SV | SV | Application Reset | See Family Spec |
| INT_OMISN | OTGM MCDS I/F Sensitivity Negedge | $00A8_H$ | U,SV | SV | Application Reset | See Family Spec |
| INT_ACCEN_CONFIG0 | Access Enable covering all INT_ECRx and all SRCy[15:0], Register 0 | $00F0_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| INT_ACCEN_CONFIG1 | Access Enable covering all INT_ECRx and all SRCy[15:0], Register 1 | $00F4_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| INT_ACCEN_SRBx0 (x=0-2) | Access Enable covering SRBx, Register 0 | $0100_H$+x *8 | U,SV | SV,SE | Application Reset | See Family Spec |

**Table 202 Register Overview - INT (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| INT_ACCEN_SRBx1 (x=0-2) | Access Enable covering SRBx, Register 1 | $0104_H$+x*8 | U,SV | SV,SE | Application Reset | See Family Spec |
| INT_ACCEN_SRC_TOSx0 (x=0-3) | Access Enable covering all SRCx[31:16] mapped to ICUx, Register 0 | $0180_H$+x*8 | U,SV | SV,SE | Application Reset | See Family Spec |
| INT_ACCEN_SRC_TOSx1 (x=0-3) | Access Enable covering all SRCx[31:16] mapped to ICUx, Register 1 | $0184_H$+x*8 | U,SV | SV,SE | Application Reset | See Family Spec |
| INT_LWSRx (x=0-3) | Latest Winning Service Request Register x, related to ICUx | $0200_H$+x*$10_H$ | U,SV | nBE | Application Reset | See Family Spec |
| INT_LASRx (x=0-3) | Last Acknowledged Service Request Register x, related to ICUx | $0204_H$+x*$10_H$ | U,SV | nBE | Application Reset | See Family Spec |
| INT_ECRx (x=0-3) | Error Capture Register x, related to ICUx | $0208_H$+x*$10_H$ | U,SV | SV,P1 | Application Reset | See Family Spec |

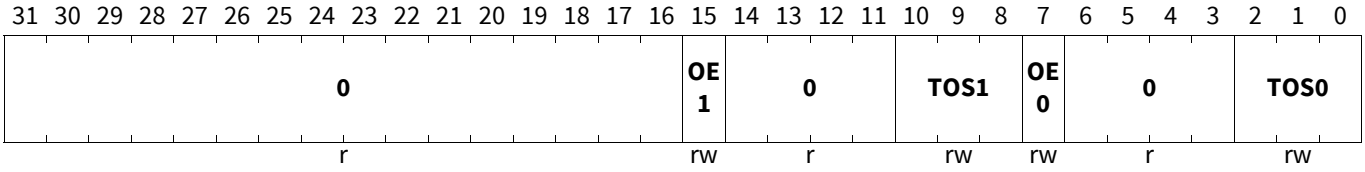
## 16.3 TC35x Specific Registers

### 16.3.1 IR Status and Control Registers

**OTGM IRQ Trace**

**INT_OIT**

**OTGM IRQ Trace** **($00A0_H$)** **Application Reset Value: $0000\ 0000_H$**




| Field | Bits | Type | Description |
|---|---|---|---|
| **TOS0** | 2:0 | rw | **Type of Service for Observation on OTGB0**<br>Trigger Set TS16_SP<br>Family concept encoding, compatible with SRC.TOS<br>$000_B$ CPU0 service is observed<br>$001_B$ DMA service is observed<br>$010_B$ CPU1 service is observed<br>$011_B$ CPU2 service is observed<br>**others**, Reserved (no action) |
| **OE0** | 7 | rw | **Output Enable for OTGB0**<br>$0_B$ Disabled<br>$1_B$ Enabled |
| **TOS1** | 10:8 | rw | **Type of Service for Observation on OTGB1**<br>Trigger Set TS16_SP<br>Family concept encoding, compatible with SRC.TOS<br>$000_B$ CPU0 service is observed<br>$001_B$ DMA service is observed<br>$010_B$ CPU1 service is observed<br>$011_B$ CPU2 service is observed<br>**others**, Reserved (no action) |
| **OE1** | 15 | rw | **Output Enable for OTGB1**<br>$0_B$ Disabled<br>$1_B$ Enabled |
| **0** | 6:3,<br>14:11,<br>31:16 | r | **Reserved**<br>Read as 0; must be written with 0. |

## 16.4 TC35x Specific Service Request Control (SRC) registers

**This chapter describes the TC35x Service Request Control (SRC) registers.**

Table 204 shows all registers associated with the Interrupt Router module in the device. This chapter describes the Service Request Control registers including:

- Mapping of Aurix module interrupt triggers to SRC
- SRC offsets
- The index number of an SRC can be calculated with the SRC Offset: Index(SRC) = <SRC Address Offset> / 4

**List of used Access Protection Register abbreviations**

- P0 -> ACCEN_SRBx, write protection of the related SRBx register. Number of Service Request Broadcast registers (SRB) and the related ACCEN_SRB registers is equal to the number of implemented TriCore CPUs.
- P1 -> ACCEN_CONFIG, write protection of all SRCx[15:0] and ICUx Error Capture registers (ECRx)
- P2 -> ACCEN_SRC_TOSx, write protects bits [31:16] of all SRCs that are mapped to TOSx (SCR.TOS=x). For each implemented Interrupt Control Unit, one ACCEN_SRC_TOSx register is implemented.

*Note: A violation of the access protection will not be executed (e.g. a write to a ´Px´/ACCEN protected register from a disabled master). In this case an access protection error is signaled to the SMU. Beside this signaling to the SMU, no other error, interrupt or trap is generated.*

**Table 203 Register Address Space - SRC**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| SRC | $F0038000_H$ | $F0039FFF_H$ | IR Service Request Control Registers (SRC) |

**Table 204 Register Overview - SRC (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SRC_CPUxSB (x=0-2) | CPUx Software Breakpoint Service Request | $00000_H$+ x*4 | U,SV | SV,P1,P2 | Debug Reset | 10 |
| SRC_BCUSPB | SBCU Service Request (SPB Bus Control Unit) | $00020_H$ | U,SV | SV,P1,P2 | Debug Reset | 10 |
| SRC_BCUBBB | EBCU Service Request (BBB Bus Control Unit, on ED and ADAS devices only) | $00024_H$ | U,SV | SV,P1,P2 | Debug Reset | 10 |
| SRC_AGBT | AGBT Service Request (on ED devices only) | $0002C_H$ | U,SV | SV,P1,P2 | Debug Reset | 10 |
| SRC_XBAR0 | SRI Domain 0 Service Request | $00030_H$ | U,SV | SV,P1,P2 | Debug Reset | 10 |
| SRC_CERBERUSy (y=0-1) | Cerberus Service Request y | $00040_H$+ y*4 | U,SV | SV,P1,P2 | Debug Reset | 10 |
| SRC_ASCLINxTX (x=0-3) | ASCLINx Transmit Service Request | $00050_H$+ x*12 | U,SV | SV,P1,P2 | Application Reset | 10 |
| SRC_ASCLINxRX (x=0-3) | ASCLINx Receive Service Request | $00054_H$+ x*12 | U,SV | SV,P1,P2 | Application Reset | 10 |

**Table 204 Register Overview - SRC (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SRC_ASCLINxERR (x=0-3) | ASCLINx Error Service Request | $00058_H$+ x*12 | U,SV | SV,P1,P2 | Application Reset | 10 |
| SRC_MTUDONE | MTU Done Service Request | $000EC_H$ | U,SV | SV,P1,P2 | Application Reset | 10 |
| SRC_QSPIxTX (x=0-3) | QSPIx Transmit Service Request | $000F0_H$+ x*$14_H$ | U,SV | SV,P1,P2 | Application Reset | 13 |
| SRC_QSPIxRX (x=0-3) | QSPIx Receive Service Request | $000F4_H$+ x*$14_H$ | U,SV | SV,P1,P2 | Application Reset | 13 |
| SRC_QSPIxERR (x=0-3) | QSPIx Error Service Request | $000F8_H$+ x*$14_H$ | U,SV | SV,P1,P2 | Application Reset | 13 |
| SRC_QSPIxPT (x=0-3) | QSPIx Phase Transition Service Request | $000FC_H$+ x*$14_H$ | U,SV | SV,P1,P2 | Application Reset | 13 |
| SRC_QSPIxU (x=0-3) | QSPIx User Defined Service Request | $00100_H$+ x*$14_H$ | U,SV | SV,P1,P2 | Application Reset | 13 |
| SRC_QSPI2HC | QSPI2 High Speed Capture Service Request | $00178_H$ | U,SV | SV,P1,P2 | Application Reset | 13 |
| SRC_QSPI3HC | QSPI3 High Speed Capture Service Request | $0017C_H$ | U,SV | SV,P1,P2 | Application Reset | 13 |
| SRC_I2CxDTR (x=0) | I2Cx Data Transfer Request | $00220_H$ | U,SV | SV,P1,P2 | Application Reset | 13 |
| SRC_I2CxERR (x=0) | I2Cx Error Service Request | $00224_H$ | U,SV | SV,P1,P2 | Application Reset | 13 |
| SRC_I2CxP (x=0) | I2Cx Protocol Service Request | $00228_H$ | U,SV | SV,P1,P2 | Application Reset | 13 |
| SRC_CCU6xSRy (x=0-1;y=0-3) | CCUx Service Request y | $002C0_H$+ x*$10_H$+y*4 | U,SV | SV,P1,P2 | Application Reset | 15 |
| SRC_GPT120CIRQ | GPT120 CAPREL Service Request | $002E0_H$ | U,SV | SV,P1,P2 | Application Reset | 15 |
| SRC_GPT120T2 | GPT120 Timer 2 Service Request | $002E4_H$ | U,SV | SV,P1,P2 | Application Reset | 15 |
| SRC_GPT120T3 | GPT120 Timer 3 Service Request | $002E8_H$ | U,SV | SV,P1,P2 | Application Reset | 15 |
| SRC_GPT120T4 | GPT120 Timer 4 Service Request | $002EC_H$ | U,SV | SV,P1,P2 | Application Reset | 15 |
| SRC_GPT120T5 | GPT120 Timer 5 Service Request | $002F0_H$ | U,SV | SV,P1,P2 | Application Reset | 15 |
| SRC_GPT120T6 | GPT120 Timer 6 Service Request | $002F4_H$ | U,SV | SV,P1,P2 | Application Reset | 15 |

**Table 204 Register Overview - SRC (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SRC_STMxSRy (x=0-2;y=0-1) | System Timer x Service Request y | $00300_H$+ x*8+y*4 | U,SV | SV,P1,P2 | Application Reset | 15 |
| SRC_FCE0 | FCE0 Error Service Request | $00330_H$ | U,SV | SV,P1,P2 | Application Reset | 15 |
| SRC_DMAERRy (y=0-3) | DMA Error Service Request y | $00340_H$+ y*4 | U,SV | SV,P1,P2 | Application Reset | 15 |
| SRC_DMACHy (y=0-63) | DMA Channel y Service Request | $00370_H$+ y*4 | U,SV | SV,P1,P2 | Application Reset | 18 |
| SRC_GETHy (y=0-9) | GETH Service Request y | $00580_H$+ y*4 | U,SV | SV,P1,P2 | Application Reset | 18 |
| SRC_CANxINTy (x=0-1;y=0-15) | CANx Service Request y | $005B0_H$+ x*$40_H$+y*4 | U,SV | SV,P1,P2 | Application Reset | 18 |
| SRC_VADCGxSRy (x=0-1;y=0-3) | EVADC Group x Service Request y | $00670_H$+ x*$10_H$+y*4 | U,SV | SV,P1,P2 | Application Reset | 18 |
| SRC_VADCCGxSRy (x=0-1;y=0-3) | EVADC Common Group x Service Request y | $00750_H$+ x*$10_H$+y*4 | U,SV | SV,P1,P2 | Application Reset | 18 |
| SRC_ERAYxINT0 (x=0) | E-RAY x Service Request 0 | $00800_H$ | U,SV | SV,P1,P2 | Application Reset | 18 |
| SRC_ERAYxINT1 (x=0) | E-RAY x Service Request 1 | $00804_H$ | U,SV | SV,P1,P2 | Application Reset | 18 |
| SRC_ERAYxTINT0 (x=0) | E-RAY x Timer Interrupt 0 Service Request | $00808_H$ | U,SV | SV,P1,P2 | Application Reset | 18 |
| SRC_ERAYxTINT1 (x=0) | E-RAY x Timer Interrupt 1 Service Request | $0080C_H$ | U,SV | SV,P1,P2 | Application Reset | 18 |
| SRC_ERAYxNDAT0 (x=0) | E-RAY x New Data 0 Service Request | $00810_H$ | U,SV | SV,P1,P2 | Application Reset | 18 |
| SRC_ERAYxNDAT1 (x=0) | E-RAY x New Data 1 Service Request | $00814_H$ | U,SV | SV,P1,P2 | Application Reset | 20 |
| SRC_ERAYxMBSC0 (x=0) | E-RAY x Message Buffer Status Changed 0 Service Request | $00818_H$ | U,SV | SV,P1,P2 | Application Reset | 20 |
| SRC_ERAYxMBSC1 (x=0) | E-RAY x Message Buffer Status Changed 1 Service Request | $0081C_H$ | U,SV | SV,P1,P2 | Application Reset | 20 |
| SRC_ERAYxOBUSY (x=0) | E-RAY x Output Buffer Busy | $00820_H$ | U,SV | SV,P1,P2 | Application Reset | 20 |

**Table 204 Register Overview - SRC (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| SRC_ERAYxIBUSY (x=0) | E-RAY x Input Buffer Busy | $00824_H$ | U,SV | SV,P1,P2 | Application Reset | 20 |
| SRC_DMUHOST | DMU Host Service Request | $00860_H$ | U,SV | SV,P1,P2 | Application Reset | 20 |
| SRC_DMUFSI | DMU FSI Service Request | $00864_H$ | U,SV | SV,P1,P2 | Application Reset | 20 |
| SRC_HSMy (y=0-1) | HSM Service Request y | $00870_H$+ y*4 | U,SV | SV,P1,P2 | Application Reset | 20 |
| SRC_SCUERUx (x=0-3) | SCU ERU Service Request x | $00880_H$+ x*4 | U,SV | SV,P1,P2 | Application Reset | 20 |
| SRC_PMSDTS | PMS DTS Service Request | $008AC_H$ | U,SV | SV,P1,P2 | Application Reset | 20 |
| SRC_PMSx (x=0-3) | Power Management System Service Request x | $008B0_H$+ x*4 | U,SV | SV,P1,P2 | Application Reset | 23 |
| SRC_SCR | Stand By Controller Service Request | $008C0_H$ | U,SV | SV,P1,P2 | Application Reset | 23 |
| SRC_SMUy (y=0-2) | SMU Service Request y | $008D0_H$+ y*4 | U,SV | SV,P1,P2 | Application Reset | 23 |
| SRC_HSPDM0BFR | HSPDM0 Buffer Service Request | $00900_H$ | U,SV | SV,P1,P2 | Application Reset | 23 |
| SRC_HSPDM0RAMP | HSPDM0 RAMP Events Service Request | $00904_H$ | U,SV | SV,P1,P2 | Application Reset | 23 |
| SRC_HSPDM0ERR | HSPDM0 Error / RAM Overflow Service Request | $00908_H$ | U,SV | SV,P1,P2 | Application Reset | 23 |
| SRC_RIFxERR (x=0-1) | Radar Interface x Error Service Request | $00970_H$+ x*8 | U,SV | SV,P1,P2 | Application Reset | 23 |
| SRC_RIFxINT (x=0-1) | Radar Interface x Service Request | $00974_H$+ x*8 | U,SV | SV,P1,P2 | Application Reset | 23 |
| SRC_SPUxINT (x=0-1) | SPU x Service Request | $00980_H$+ x*8 | U,SV | SV,P1,P2 | Application Reset | 23 |
| SRC_SPUxERR (x=0-1) | SPU x Error Service Request | $00984_H$+ x*8 | U,SV | SV,P1,P2 | Application Reset | 23 |
| SRC_GPSRxy (x=0-2;y=0-7) | General Purpose Group x Service Request y | $00990_H$+ x*$20_H$+y*4 | U,SV | SV,P1,P2 | Application Reset | 25 |

## 16.5 TC35x Specific Registers

### 16.5.1 IR Service Request Control Registers (SRC)

**CPUx Software Breakpoint Service Request**

**SRC_CPUxSB (x=0-2)**
**CPUx Software Breakpoint Service Request (00000$_H$ + x*4) Debug Reset Value: 0000 0000$_H$**
**SRC_BCUSPB**
**SBCU Service Request [SPB Bus Control Unit) (00020$_H$) Debug Reset Value: 0000 0000$_H$**
**SRC_BCUBBB**
**EBCU Service Request [BBB Bus Control Unit, on ED and ADAS devices only)(00024$_H$) Debug Reset Value: 0000 0000$_H$**
**SRC_AGBT**
**AGBT Service Request [on ED devices only) (0002C$_H$) Debug Reset Value: 0000 0000$_H$**
**SRC_XBAR0**
**SRI Domain 0 Service Request (00030$_H$) Debug Reset Value: 0000 0000$_H$**
**SRC_CERBERUSy (y=0-1)**
**Cerberus Service Request y (00040$_H$+y*4) Debug Reset Value: 0000 0000$_H$**
**SRC_ASCLINxTX (x=0-3)**
**ASCLINx Transmit Service Request (00050$_H$+x*12) Application Reset Value: 0000 0000$_H$**
**SRC_ASCLINxRX (x=0-3)**
**ASCLINx Receive Service Request (00054$_H$+x*12) Application Reset Value: 0000 0000$_H$**
**SRC_ASCLINxERR (x=0-3)**
**ASCLINx Error Service Request (00058$_H$+x*12) Application Reset Value: 0000 0000$_H$**
**SRC_MTUDONE**
**MTU Done Service Request (000EC$_H$) Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | SWSCLR | SWS | IOVCLR | IOV | SETR | CLRR | SRR | 0 | | | ECC | | | | |
| r | w | rh | w | rh | w | w | rh | r | | | rwh | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | TOS | | | SRE | 0 | | SRPN | | | | | | | |
| r | | rw | | | rw | r | | rw | | | | | | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SRPN** | 7:0 | rw | **Service Request Priority Number**<br>The SRPN bit field defines the priority of a service request with respect to service requests with to the same service provider (same SRC.TOS configuration):<br>$00_H$ -> Service request is on lowest priority<br>...<br>$FF_H$ -> Service request is on highest priority<br><br>**Notes**<br>1. *For a CPU $01_H$ is the lowest priority as $00_H$ is never serviced. For a DMA $00_H$ triggers channel 0.*<br>2. *For DMA, SRPN must not be greater than the highest implemented DMA channel number.* |
| **SRE** | 10 | rw | **Service Request Enable**<br>$0_B$ Service request is disabled<br>$1_B$ Service request is enabled |
| **TOS** | 13:11 | rw | **Type of Service Control**<br>The TOS bit field configuration maps a Service Request to an Interrupt Service Provider:<br>$000_B$ CPU0 service is initiated<br>$001_B$ DMA service is initiated<br>$010_B$ CPU1 service is initiated<br>$011_B$ CPU2 service is initiated<br>**Others**, Reserved (no action) |
| **ECC** | 20:16 | rwh | **Error Correction Code**<br>The ECC bit field will be updated by the SRN under the following conditions:<br>• Write or Read-Modify-Write to SRC[31:0]<br>• Write to SRC[15:0] (16-bit write)<br>• Write to SRC[15:8] or write to SRC[7:0] (byte write) |
| **SRR** | 24 | rh | **Service Request Flag**<br>The SRR bit shows the status of the Service Request.<br>$0_B$ No service request is pending<br>$1_B$ A service request is pending |
| **CLRR** | 25 | w | **Request Clear Bit**<br>The CLRR bit is required to reset **SRR**.<br>$0_B$ No action<br>$1_B$ Clear SRR; bit value is not stored; read always returns 0; no action if SETR is set in parallel. |
| **SETR** | 26 | w | **Request Set Bit**<br>The SETR bit is required to set **SRR**.<br>$0_B$ No action<br>$1_B$ Set SRR and SWS; SRR bit value is not stored; read always returns 0; no action if CLRR is set in parallel. |

| Field | Bits | Type | Description |
|---|---|---|---|
| **IOV** | 27 | rh | **Interrupt Trigger Overflow Bit**<br>The IOV bit is set by HW if a new service request was triggered via interrupt trigger or **SETR** bit while the SRN has still an pending service request.<br>$0_B$ No Interrupt Trigger Overflow detected<br>$1_B$ Interrupt Overflow Detected. |
| **IOVCLR** | 28 | w | **Interrupt Trigger Overflow Clear Bit**<br>IOVCLR is required to reset **IOV**.<br>$0_B$ No action<br>$1_B$ Clear IOV; bit value is not stored; read always returns 0. |
| **SWS** | 29 | rh | **SW Sticky Bit**<br>The Software Sticky Bit is set when the **SRR** bit has been set via the **SETR** bit.<br>This bit can be cleared by writing with 1 to **SWSCLR**.<br>Writing to SWS has no effect.<br>$0_B$ No interrupt was initiated via SETR<br>$1_B$ Interrupt was initiated via SETR |
| **SWSCLR** | 30 | w | **SW Sticky Clear Bit**<br>SWSCLR is required to reset **SWS**.<br>$0_B$ No action<br>$1_B$ Clear SWS; bit value is not stored; read always returns 0. |
| **0** | 9:8,<br>15:14,<br>23:21,<br>31 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SRC_QSPIxTX (x=0-3)**
**QSPIx Transmit Service Request** **($000F0_H$+x*$14_H$)** **Application Reset Value: 0000 $0000_H$**
**SRC_QSPIxRX (x=0-3)**
**QSPIx Receive Service Request** **($000F4_H$+x*$14_H$)** **Application Reset Value: 0000 $0000_H$**
**SRC_QSPIxERR (x=0-3)**
**QSPIx Error Service Request** **($000F8_H$+x*$14_H$)** **Application Reset Value: 0000 $0000_H$**
**SRC_QSPIxPT (x=0-3)**
**QSPIx Phase Transition Service Request** **($000FC_H$+x*$14_H$)** **Application Reset Value: 0000 $0000_H$**
**SRC_QSPIxU (x=0-3)**
**QSPIx User Defined Service Request** **($00100_H$+x*$14_H$)** **Application Reset Value: 0000 $0000_H$**
**SRC_QSPI2HC**
**QSPI2 High Speed Capture Service Request** **($00178_H$)** **Application Reset Value: 0000 $0000_H$**
**SRC_QSPI3HC**
**QSPI3 High Speed Capture Service Request** **($0017C_H$)** **Application Reset Value: 0000 $0000_H$**
**SRC_I2C0DTR**
**I2C0 Data Transfer Request** **($00220_H$)** **Application Reset Value: 0000 $0000_H$**
**SRC_I2C0ERR**
**I2C0 Error Service Request** **($00224_H$)** **Application Reset Value: 0000 $0000_H$**
**SRC_I2C0P**
**I2C0 Protocol Service Request** **($00228_H$)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **SWSCLR** | **SWS** | **IOVCLR** | **IOV** | **SETR** | **CLRR** | **SRR** | **0** | | | **ECC** | | | | |
| r | w | rh | w | rh | w | w | rh | r | | | rwh | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | **TOS** | | | **SRE** | **0** | | **SRPN** | | | | | | | |
| r | | rw | | | rw | r | | rw | | | | | | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SRPN** | 7:0 | rw | **Service Request Priority Number**<br>The SRPN bit field defines the priority of a service request with respect to service requests with to the same service provider (same SRC.TOS configuration):<br>$00_H$ -> Service request is on lowest priority<br>...<br>$FF_H$ -> Service request is on highest priority<br><br>**Notes**<br>1. *For a CPU $01_H$ is the lowest priority as $00_H$ is never serviced. For a DMA $00_H$ triggers channel 0.*<br>2. *For DMA, SRPN must not be greater than the highest implemented DMA channel number.* |
| **SRE** | 10 | rw | **Service Request Enable**<br>$0_B$ Service request is disabled<br>$1_B$ Service request is enabled |

| Field | Bits | Type | Description |
|---|---|---|---|
| **TOS** | 13:11 | rw | **Type of Service Control**<br>The TOS bit field configuration maps a Service Request to an Interrupt Service Provider:<br>$000_B$ CPU0 service is initiated<br>$001_B$ DMA service is initiated<br>$010_B$ CPU1 service is initiated<br>$011_B$ CPU2 service is initiated<br>**Others**, Reserved (no action) |
| **ECC** | 20:16 | rwh | **Error Correction Code**<br>The ECC bit field will be updated by the SRN under the following conditions:<br>• Write or Read-Modify-Write to SRC[31:0]<br>• Write to SRC[15:0] (16-bit write)<br>• Write to SRC[15:8] or write to SRC[7:0] (byte write) |
| **SRR** | 24 | rh | **Service Request Flag**<br>The SRR bit shows the status of the Service Request.<br>$0_B$ No service request is pending<br>$1_B$ A service request is pending |
| **CLRR** | 25 | w | **Request Clear Bit**<br>The CLRR bit is required to reset **SRR**.<br>$0_B$ No action<br>$1_B$ Clear SRR; bit value is not stored; read always returns 0; no action if SETR is set in parallel. |
| **SETR** | 26 | w | **Request Set Bit**<br>The SETR bit is required to set **SRR**.<br>$0_B$ No action<br>$1_B$ Set SRR and SWS; SRR bit value is not stored; read always returns 0; no action if CLRR is set in parallel. |
| **IOV** | 27 | rh | **Interrupt Trigger Overflow Bit**<br>The IOV bit is set by HW if a new service request was triggered via interrupt trigger or **SETR** bit while the SRN has still an pending service request.<br>$0_B$ No Interrupt Trigger Overflow detected<br>$1_B$ Interrupt Overflow Detected. |
| **IOVCLR** | 28 | w | **Interrupt Trigger Overflow Clear Bit**<br>IOVCLR is required to reset **IOV**.<br>$0_B$ No action<br>$1_B$ Clear IOV; bit value is not stored; read always returns 0. |
| **SWS** | 29 | rh | **SW Sticky Bit**<br>The Software Sticky Bit is set when the **SRR** bit has been set via the **SETR** bit.<br>This bit can be cleared by writing with 1 to **SWSCLR**.<br>Writing to SWS has no effect.<br>$0_B$ No interrupt was initiated via SETR<br>$1_B$ Interrupt was initiated via SETR |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SWSCLR** | 30 | w | **SW Sticky Clear Bit**<br>SWSCLR is required to reset **SWS**.<br>$0_B$ No action<br>$1_B$ Clear SWS; bit value is not stored; read always returns 0. |
| **0** | 9:8, 15:14, 23:21, 31 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SRC_CCU6xSRy (x=0-1;y=0-3)**
**CCUx Service Request y** **($002C0_H$+x*$10_H$+y*4)** **Application Reset Value: 0000 $0000_H$**
**SRC_GPT120CIRQ**
**GPT120 CAPREL Service Request** **($002E0_H$)** **Application Reset Value: 0000 $0000_H$**
**SRC_GPT120T2**
**GPT120 Timer 2 Service Request** **($002E4_H$)** **Application Reset Value: 0000 $0000_H$**
**SRC_GPT120T3**
**GPT120 Timer 3 Service Request** **($002E8_H$)** **Application Reset Value: 0000 $0000_H$**
**SRC_GPT120T4**
**GPT120 Timer 4 Service Request** **($002EC_H$)** **Application Reset Value: 0000 $0000_H$**
**SRC_GPT120T5**
**GPT120 Timer 5 Service Request** **($002F0_H$)** **Application Reset Value: 0000 $0000_H$**
**SRC_GPT120T6**
**GPT120 Timer 6 Service Request** **($002F4_H$)** **Application Reset Value: 0000 $0000_H$**
**SRC_STMxSRy (x=0-2;y=0-1)**
**System Timer x Service Request y** **($00300_H$+x*8+y*4)** **Application Reset Value: 0000 $0000_H$**
**SRC_FCE0**
**FCE0 Error Service Request** **($00330_H$)** **Application Reset Value: 0000 $0000_H$**
**SRC_DMAERRy (y=0-3)**
**DMA Error Service Request y** **($00340_H$+y*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **SWSCLR** | **SWS** | **IOVCLR** | **IOV** | **SETR** | **CLRR** | **SRR** | **0** | | | **ECC** | | | | |
| r | w | rh | w | rh | w | w | rh | r | | | rwh | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | **TOS** | | | **SRE** | **0** | | **SRPN** | | | | | | | |
| r | | rw | | | rw | r | | rw | | | | | | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SRPN** | 7:0 | rw | **Service Request Priority Number**<br>The SRPN bit field defines the priority of a service request with respect to service requests with to the same service provider (same SRC.TOS configuration):<br>$00_H$ -> Service request is on lowest priority<br>...<br>$FF_H$ -> Service request is on highest priority<br><br>**Notes**<br>1. *For a CPU $01_H$ is the lowest priority as $00_H$ is never serviced. For a DMA $00_H$ triggers channel 0.*<br>2. *For DMA, SRPN must not be greater than the highest implemented DMA channel number.* |
| **SRE** | 10 | rw | **Service Request Enable**<br>$0_B$ Service request is disabled<br>$1_B$ Service request is enabled |
| **TOS** | 13:11 | rw | **Type of Service Control**<br>The TOS bit field configuration maps a Service Request to an Interrupt Service Provider:<br>$000_B$ CPU0 service is initiated<br>$001_B$ DMA service is initiated<br>$010_B$ CPU1 service is initiated<br>$011_B$ CPU2 service is initiated<br>**Others**, Reserved (no action) |
| **ECC** | 20:16 | rwh | **Error Correction Code**<br>The ECC bit field will be updated by the SRN under the following conditions:<br>• Write or Read-Modify-Write to SRC[31:0]<br>• Write to SRC[15:0] (16-bit write)<br>• Write to SRC[15:8] or write to SRC[7:0] (byte write) |
| **SRR** | 24 | rh | **Service Request Flag**<br>The SRR bit shows the status of the Service Request.<br>$0_B$ No service request is pending<br>$1_B$ A service request is pending |
| **CLRR** | 25 | w | **Request Clear Bit**<br>The CLRR bit is required to reset **SRR**.<br>$0_B$ No action<br>$1_B$ Clear SRR; bit value is not stored; read always returns 0; no action if SETR is set in parallel. |
| **SETR** | 26 | w | **Request Set Bit**<br>The SETR bit is required to set **SRR**.<br>$0_B$ No action<br>$1_B$ Set SRR and SWS; SRR bit value is not stored; read always returns 0; no action if CLRR is set in parallel. |

| Field | Bits | Type | Description |
|---|---|---|---|
| **IOV** | 27 | rh | **Interrupt Trigger Overflow Bit**<br>The IOV bit is set by HW if a new service request was triggered via interrupt trigger or **SETR** bit while the SRN has still an pending service request.<br>$0_B$ No Interrupt Trigger Overflow detected<br>$1_B$ Interrupt Overflow Detected. |
| **IOVCLR** | 28 | w | **Interrupt Trigger Overflow Clear Bit**<br>IOVCLR is required to reset **IOV**.<br>$0_B$ No action<br>$1_B$ Clear IOV; bit value is not stored; read always returns 0. |
| **SWS** | 29 | rh | **SW Sticky Bit**<br>The Software Sticky Bit is set when the **SRR** bit has been set via the **SETR** bit.<br>This bit can be cleared by writing with 1 to **SWSCLR**.<br>Writing to SWS has no effect.<br>$0_B$ No interrupt was initiated via SETR<br>$1_B$ Interrupt was initiated via SETR |
| **SWSCLR** | 30 | w | **SW Sticky Clear Bit**<br>SWSCLR is required to reset **SWS**.<br>$0_B$ No action<br>$1_B$ Clear SWS; bit value is not stored; read always returns 0. |
| **0** | 9:8,<br>15:14,<br>23:21,<br>31 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SRC_DMACHy (y=0-63)**
**DMA Channel y Service Request** **($00370_H$+y*4)** **Application Reset Value: 0000 0000$_H$**
**SRC_GETHy (y=0-9)**
**GETH Service Request y** **($00580_H$+y*4)** **Application Reset Value: 0000 0000$_H$**
**SRC_CANxINTy (x=0-1;y=0-15)**
**CANx Service Request y** **($005B0_H$+x*$40_H$+y*4)** **Application Reset Value: 0000 0000$_H$**
**SRC_VADCGxSRy (x=0-1;y=0-3)**
**EVADC Group x Service Request y** **($00670_H$+x*$10_H$+y*4)** **Application Reset Value: 0000 0000$_H$**
**SRC_VADCCGxSRy (x=0-1;y=0-3)**
**EVADC Common Group x Service Request y** **($00750_H$+x*$10_H$+y*4)** **Application Reset Value: 0000 0000$_H$**
**SRC_ERAY0INT0**
**E-RAY 0 Service Request 0** **($00800_H$)** **Application Reset Value: 0000 0000$_H$**
**SRC_ERAY0INT1**
**E-RAY 0 Service Request 1** **($00804_H$)** **Application Reset Value: 0000 0000$_H$**
**SRC_ERAY0TINT0**
**E-RAY 0 Timer Interrupt 0 Service Request** **($00808_H$)** **Application Reset Value: 0000 0000$_H$**
**SRC_ERAY0TINT1**
**E-RAY 0 Timer Interrupt 1 Service Request** **($0080C_H$)** **Application Reset Value: 0000 0000$_H$**
**SRC_ERAY0NDAT0**
**E-RAY 0 New Data 0 Service Request** **($00810_H$)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | SWSCLR | SWS | IOVCLR | IOV | SETR | CLRR | SRR | 0 | | | ECC | | | | |
| r | w | rh | w | rh | w | w | rh | r | | | rwh | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | TOS | | | SRE | 0 | | SRPN | | | | | | | |
| r | | rw | | | rw | r | | rw | | | | | | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SRPN** | 7:0 | rw | **Service Request Priority Number**<br>The SRPN bit field defines the priority of a service request with respect to service requests with to the same service provider (same SRC.TOS configuration):<br>$00_H$ -> Service request is on lowest priority<br>...<br>$FF_H$ -> Service request is on highest priority<br><br>**Notes**<br>1. *For a CPU $01_H$ is the lowest priority as $00_H$ is never serviced. For a DMA $00_H$ triggers channel 0.*<br>2. *For DMA, SRPN must not be greater than the highest implemented DMA channel number.* |
| **SRE** | 10 | rw | **Service Request Enable**<br>$0_B$ Service request is disabled<br>$1_B$ Service request is enabled |

| Field | Bits | Type | Description |
|---|---|---|---|
| **TOS** | 13:11 | rw | **Type of Service Control**<br>The TOS bit field configuration maps a Service Request to an Interrupt Service Provider:<br>$000_B$ CPU0 service is initiated<br>$001_B$ DMA service is initiated<br>$010_B$ CPU1 service is initiated<br>$011_B$ CPU2 service is initiated<br>**Others**, Reserved (no action) |
| **ECC** | 20:16 | rwh | **Error Correction Code**<br>The ECC bit field will be updated by the SRN under the following conditions:<br>• Write or Read-Modify-Write to SRC[31:0]<br>• Write to SRC[15:0] (16-bit write)<br>• Write to SRC[15:8] or write to SRC[7:0] (byte write) |
| **SRR** | 24 | rh | **Service Request Flag**<br>The SRR bit shows the status of the Service Request.<br>$0_B$ No service request is pending<br>$1_B$ A service request is pending |
| **CLRR** | 25 | w | **Request Clear Bit**<br>The CLRR bit is required to reset **SRR**.<br>$0_B$ No action<br>$1_B$ Clear SRR; bit value is not stored; read always returns 0; no action if SETR is set in parallel. |
| **SETR** | 26 | w | **Request Set Bit**<br>The SETR bit is required to set **SRR**.<br>$0_B$ No action<br>$1_B$ Set SRR and SWS; SRR bit value is not stored; read always returns 0; no action if CLRR is set in parallel. |
| **IOV** | 27 | rh | **Interrupt Trigger Overflow Bit**<br>The IOV bit is set by HW if a new service request was triggered via interrupt trigger or **SETR** bit while the SRN has still an pending service request.<br>$0_B$ No Interrupt Trigger Overflow detected<br>$1_B$ Interrupt Overflow Detected. |
| **IOVCLR** | 28 | w | **Interrupt Trigger Overflow Clear Bit**<br>IOVCLR is required to reset **IOV**.<br>$0_B$ No action<br>$1_B$ Clear IOV; bit value is not stored; read always returns 0. |
| **SWS** | 29 | rh | **SW Sticky Bit**<br>The Software Sticky Bit is set when the **SRR** bit has been set via the **SETR** bit.<br>This bit can be cleared by writing with 1 to **SWSCLR**.<br>Writing to SWS has no effect.<br>$0_B$ No interrupt was initiated via SETR<br>$1_B$ Interrupt was initiated via SETR |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SWSCLR** | 30 | w | **SW Sticky Clear Bit**<br>SWSCLR is required to reset **SWS**.<br>$0_B$ No action<br>$1_B$ Clear SWS; bit value is not stored; read always returns 0. |
| **0** | 9:8, 15:14, 23:21, 31 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SRC_ERAY0NDAT1**
**E-RAY 0 New Data 1 Service Request** **($00814_H$)** **Application Reset Value: 0000 0000$_H$**
**SRC_ERAY0MBSC0**
**E-RAY 0 Message Buffer Status Changed 0 Service Request($00818_H$)** **Application Reset Value: 0000 0000$_H$**
**SRC_ERAY0MBSC1**
**E-RAY 0 Message Buffer Status Changed 1 Service Request($0081C_H$)** **Application Reset Value: 0000 0000$_H$**
**SRC_ERAY0OBUSY**
**E-RAY 0 Output Buffer Busy** **($00820_H$)** **Application Reset Value: 0000 0000$_H$**
**SRC_ERAY0IBUSY**
**E-RAY 0 Input Buffer Busy** **($00824_H$)** **Application Reset Value: 0000 0000$_H$**
**SRC_DMUHOST**
**DMU Host Service Request** **($00860_H$)** **Application Reset Value: 0000 0000$_H$**
**SRC_DMUFSI**
**DMU FSI Service Request** **($00864_H$)** **Application Reset Value: 0000 0000$_H$**
**SRC_HSMy (y=0-1)**
**HSM Service Request y** **($00870_H$+y*4)** **Application Reset Value: 0000 0000$_H$**
**SRC_SCUERUx (x=0-3)**
**SCU ERU Service Request x** **($00880_H$+x*4)** **Application Reset Value: 0000 0000$_H$**
**SRC_PMSDTS**
**PMS DTS Service Request** **($008AC_H$)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **SWSCLR** | **SWS** | **IOVCLR** | **IOV** | **SETR** | **CLRR** | **SRR** | **0** | | | **ECC** | | | | |
| r | w | rh | w | rh | w | w | rh | r | | | rwh | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | **TOS** | | | **SRE** | **0** | | **SRPN** | | | | | | | |
| r | | rw | | | rw | r | | rw | | | | | | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SRPN** | 7:0 | rw | **Service Request Priority Number**<br>The SRPN bit field defines the priority of a service request with respect to service requests with to the same service provider (same SRC.TOS configuration):<br>$00_H$ -> Service request is on lowest priority<br>...<br>$FF_H$ -> Service request is on highest priority<br><br>**Notes**<br>1. *For a CPU $01_H$ is the lowest priority as $00_H$ is never serviced. For a DMA $00_H$ triggers channel 0.*<br>2. *For DMA, SRPN must not be greater than the highest implemented DMA channel number.* |
| **SRE** | 10 | rw | **Service Request Enable**<br>$0_B$ Service request is disabled<br>$1_B$ Service request is enabled |
| **TOS** | 13:11 | rw | **Type of Service Control**<br>The TOS bit field configuration maps a Service Request to an Interrupt Service Provider:<br>$000_B$ CPU0 service is initiated<br>$001_B$ DMA service is initiated<br>$010_B$ CPU1 service is initiated<br>$011_B$ CPU2 service is initiated<br>**Others**, Reserved (no action) |
| **ECC** | 20:16 | rwh | **Error Correction Code**<br>The ECC bit field will be updated by the SRN under the following conditions:<br>• Write or Read-Modify-Write to SRC[31:0]<br>• Write to SRC[15:0] (16-bit write)<br>• Write to SRC[15:8] or write to SRC[7:0] (byte write) |
| **SRR** | 24 | rh | **Service Request Flag**<br>The SRR bit shows the status of the Service Request.<br>$0_B$ No service request is pending<br>$1_B$ A service request is pending |
| **CLRR** | 25 | w | **Request Clear Bit**<br>The CLRR bit is required to reset **SRR**.<br>$0_B$ No action<br>$1_B$ Clear SRR; bit value is not stored; read always returns 0; no action if SETR is set in parallel. |
| **SETR** | 26 | w | **Request Set Bit**<br>The SETR bit is required to set **SRR**.<br>$0_B$ No action<br>$1_B$ Set SRR and SWS; SRR bit value is not stored; read always returns 0; no action if CLRR is set in parallel. |

| Field | Bits | Type | Description |
|---|---|---|---|
| **IOV** | 27 | rh | **Interrupt Trigger Overflow Bit**<br>The IOV bit is set by HW if a new service request was triggered via interrupt trigger or **SETR** bit while the SRN has still an pending service request.<br>$0_B$ No Interrupt Trigger Overflow detected<br>$1_B$ Interrupt Overflow Detected. |
| **IOVCLR** | 28 | w | **Interrupt Trigger Overflow Clear Bit**<br>IOVCLR is required to reset **IOV**.<br>$0_B$ No action<br>$1_B$ Clear IOV; bit value is not stored; read always returns 0. |
| **SWS** | 29 | rh | **SW Sticky Bit**<br>The Software Sticky Bit is set when the **SRR** bit has been set via the **SETR** bit.<br>This bit can be cleared by writing with 1 to **SWSCLR**.<br>Writing to SWS has no effect.<br>$0_B$ No interrupt was initiated via SETR<br>$1_B$ Interrupt was initiated via SETR |
| **SWSCLR** | 30 | w | **SW Sticky Clear Bit**<br>SWSCLR is required to reset **SWS**.<br>$0_B$ No action<br>$1_B$ Clear SWS; bit value is not stored; read always returns 0. |
| **0** | 9:8,<br>15:14,<br>23:21,<br>31 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SRC_PMSx (x=0-3)**
**Power Management System Service Request x** **($008B0_H$+x*4)** **Application Reset Value: 0000 0000$_H$**
**SRC_SCR**
**Stand By Controller Service Request** **($008C0_H$)** **Application Reset Value: 0000 0000$_H$**
**SRC_SMUy (y=0-2)**
**SMU Service Request y** **($008D0_H$+y*4)** **Application Reset Value: 0000 0000$_H$**
**SRC_HSPDM0BFR**
**HSPDM0 Buffer Service Request** **($00900_H$)** **Application Reset Value: 0000 0000$_H$**
**SRC_HSPDM0RAMP**
**HSPDM0 RAMP Events Service Request** **($00904_H$)** **Application Reset Value: 0000 0000$_H$**
**SRC_HSPDM0ERR**
**HSPDM0 Error / RAM Overflow Service Request** **($00908_H$)** **Application Reset Value: 0000 0000$_H$**
**SRC_RIFxERR (x=0-1)**
**Radar Interface x Error Service Request** **($00970_H$+x*8)** **Application Reset Value: 0000 0000$_H$**
**SRC_RIFxINT (x=0-1)**
**Radar Interface x Service Request** **($00974_H$+x*8)** **Application Reset Value: 0000 0000$_H$**
**SRC_SPUxINT (x=0-1)**
**SPU x Service Request** **($00980_H$+x*8)** **Application Reset Value: 0000 0000$_H$**
**SRC_SPUxERR (x=0-1)**
**SPU x Error Service Request** **($00984_H$+x*8)** **Application Reset Value: 0000 0000$_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **SWSCLR** | **SWS** | **IOVCLR** | **IOV** | **SETR** | **CLRR** | **SRR** | **0** | | | **ECC** | | | | |
| r | w | rh | w | rh | w | w | rh | r | | | rwh | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | **TOS** | | | **SRE** | **0** | | **SRPN** | | | | | | | |
| r | | rw | | | rw | r | | rw | | | | | | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SRPN** | 7:0 | rw | **Service Request Priority Number**<br>The SRPN bit field defines the priority of a service request with respect to service requests with to the same service provider (same SRC.TOS configuration):<br>$00_H$ -> Service request is on lowest priority<br>...<br>$FF_H$ -> Service request is on highest priority<br><br>**Notes**<br>1. *For a CPU $01_H$ is the lowest priority as $00_H$ is never serviced. For a DMA $00_H$ triggers channel 0.*<br>2. *For DMA, SRPN must not be greater than the highest implemented DMA channel number.* |
| **SRE** | 10 | rw | **Service Request Enable**<br>$0_B$ Service request is disabled<br>$1_B$ Service request is enabled |

| Field | Bits | Type | Description |
|---|---|---|---|
| **TOS** | 13:11 | rw | **Type of Service Control**<br>The TOS bit field configuration maps a Service Request to an Interrupt Service Provider:<br>$000_B$ CPU0 service is initiated<br>$001_B$ DMA service is initiated<br>$010_B$ CPU1 service is initiated<br>$011_B$ CPU2 service is initiated<br>**Others**, Reserved (no action) |
| **ECC** | 20:16 | rwh | **Error Correction Code**<br>The ECC bit field will be updated by the SRN under the following conditions:<br>• Write or Read-Modify-Write to SRC[31:0]<br>• Write to SRC[15:0] (16-bit write)<br>• Write to SRC[15:8] or write to SRC[7:0] (byte write) |
| **SRR** | 24 | rh | **Service Request Flag**<br>The SRR bit shows the status of the Service Request.<br>$0_B$ No service request is pending<br>$1_B$ A service request is pending |
| **CLRR** | 25 | w | **Request Clear Bit**<br>The CLRR bit is required to reset **SRR**.<br>$0_B$ No action<br>$1_B$ Clear SRR; bit value is not stored; read always returns 0; no action if SETR is set in parallel. |
| **SETR** | 26 | w | **Request Set Bit**<br>The SETR bit is required to set **SRR**.<br>$0_B$ No action<br>$1_B$ Set SRR and SWS; SRR bit value is not stored; read always returns 0; no action if CLRR is set in parallel. |
| **IOV** | 27 | rh | **Interrupt Trigger Overflow Bit**<br>The IOV bit is set by HW if a new service request was triggered via interrupt trigger or **SETR** bit while the SRN has still an pending service request.<br>$0_B$ No Interrupt Trigger Overflow detected<br>$1_B$ Interrupt Overflow Detected. |
| **IOVCLR** | 28 | w | **Interrupt Trigger Overflow Clear Bit**<br>IOVCLR is required to reset **IOV**.<br>$0_B$ No action<br>$1_B$ Clear IOV; bit value is not stored; read always returns 0. |
| **SWS** | 29 | rh | **SW Sticky Bit**<br>The Software Sticky Bit is set when the **SRR** bit has been set via the **SETR** bit.<br>This bit can be cleared by writing with 1 to **SWSCLR**.<br>Writing to SWS has no effect.<br>$0_B$ No interrupt was initiated via SETR<br>$1_B$ Interrupt was initiated via SETR |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SWSCLR** | 30 | w | **SW Sticky Clear Bit**<br>SWSCLR is required to reset **SWS**.<br>$0_B$ No action<br>$1_B$ Clear SWS; bit value is not stored; read always returns 0. |
| **0** | 9:8, 15:14, 23:21, 31 | r | **Reserved**<br>Read as 0; should be written with 0. |

**SRC_GPSRxy (x=0-2;y=0-7)**

**General Purpose Group x Service Request y($00990_H$+x*$20_H$+y*4)** **Application Reset Value: 0000 $0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **SWSCLR** | **SWS** | **IOVCLR** | **IOV** | **SETR** | **CLRR** | **SRR** | **0** | | | **ECC** | | | | |
| r | w | rh | w | rh | w | w | rh | r | | | rwh | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | **TOS** | | | **SRE** | **0** | | **SRPN** | | | | | | | |
| r | | rw | | | rw | r | | rw | | | | | | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SRPN** | 7:0 | rw | **Service Request Priority Number**<br>The SRPN bit field defines the priority of a service request with respect to service requests with to the same service provider (same SRC.TOS configuration):<br>$00_H$ -> Service request is on lowest priority<br>...<br>$FF_H$ -> Service request is on highest priority<br><br>**Notes**<br>1. *For a CPU $01_H$ is the lowest priority as $00_H$ is never serviced. For a DMA $00_H$ triggers channel 0.*<br>2. *For DMA, SRPN must not be greater than the highest implemented DMA channel number.* |
| **SRE** | 10 | rw | **Service Request Enable**<br>$0_B$ Service request is disabled<br>$1_B$ Service request is enabled |

| Field | Bits | Type | Description |
|---|---|---|---|
| **TOS** | 13:11 | rw | **Type of Service Control**<br>The TOS bit field configuration maps a Service Request to an Interrupt Service Provider:<br>$000_B$ CPU0 service is initiated<br>$001_B$ DMA service is initiated<br>$010_B$ CPU1 service is initiated<br>$011_B$ CPU2 service is initiated<br>**Others**, Reserved (no action) |
| **ECC** | 20:16 | rwh | **Error Correction Code**<br>The ECC bit field will be updated by the SRN under the following conditions:<br>• Write or Read-Modify-Write to SRC[31:0]<br>• Write to SRC[15:0] (16-bit write)<br>• Write to SRC[15:8] or write to SRC[7:0] (byte write) |
| **SRR** | 24 | rh | **Service Request Flag**<br>The SRR bit shows the status of the Service Request.<br>$0_B$ No service request is pending<br>$1_B$ A service request is pending |
| **CLRR** | 25 | w | **Request Clear Bit**<br>The CLRR bit is required to reset **SRR**.<br>$0_B$ No action<br>$1_B$ Clear SRR; bit value is not stored; read always returns 0; no action if SETR is set in parallel. |
| **SETR** | 26 | w | **Request Set Bit**<br>The SETR bit is required to set **SRR**.<br>$0_B$ No action<br>$1_B$ Set SRR and SWS; SRR bit value is not stored; read always returns 0; no action if CLRR is set in parallel. |
| **IOV** | 27 | rh | **Interrupt Trigger Overflow Bit**<br>The IOV bit is set by HW if a new service request was triggered via interrupt trigger or **SETR** bit while the SRN has still an pending service request.<br>$0_B$ No Interrupt Trigger Overflow detected<br>$1_B$ Interrupt Overflow Detected. |
| **IOVCLR** | 28 | w | **Interrupt Trigger Overflow Clear Bit**<br>IOVCLR is required to reset **IOV**.<br>$0_B$ No action<br>$1_B$ Clear IOV; bit value is not stored; read always returns 0. |
| **SWS** | 29 | rh | **SW Sticky Bit**<br>The Software Sticky Bit is set when the **SRR** bit has been set via the **SETR** bit.<br>This bit can be cleared by writing with 1 to **SWSCLR**.<br>Writing to SWS has no effect.<br>$0_B$ No interrupt was initiated via SETR<br>$1_B$ Interrupt was initiated via SETR |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SWSCLR** | 30 | w | **SW Sticky Clear Bit**<br>SWSCLR is required to reset **SWS**.<br>$0_B$ No action<br>$1_B$ Clear SWS; bit value is not stored; read always returns 0. |
| **0** | 9:8,<br>15:14,<br>23:21,<br>31 | r | **Reserved**<br>Read as 0; should be written with 0. |

## 16.6 Revision History

**Table 205 Revision History**

| Reference | Change to Previous Version | |
|---|---|---|
| **V1.2.6** | | |
| – | No functional changes | |
| **V1.2.7** | | |
| | No functional change | |
| **V1.2.8** | | |
| | Removed connection table. | |
| **V1.2.9** | | |
| | No changes. | |
| **V1.2.10** | | |
| – | No functional changes. | |
| **V1.2.11** | | |
| **Page 6** | Updated bullet list item. | |

# 17 Flexible CRC Engine (FCE)

For the general description of the module and the registers, please refer to the family spec.

## 17.1 TC35x Specific IP Configuration

There are no device specific IP configurations.

## 17.2 TC35x Specific Register Set

**Table 206 Register Address Space - FCE**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| FCE | $F0000000_H$ | $F00001FF_H$ | FPI slave interface |

**Table 207 Register Overview - FCE (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| FCE_CLC | Clock Control Register | $000_H$ | U,SV | E,SV,P | Application Reset | See Family Spec |
| FCE_ID | Module Identification Register | $008_H$ | U,SV | BE | Application Reset | See Family Spec |
| FCE_CHSTS | Channels Status Register | $020_H$ | U,SV | BE | Application Reset | See Family Spec |
| FCE_KRSTCLR | Kernel Reset Status Clear Register | $0EC_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| FCE_KRST1 | Kernel Reset Register 1 | $0F0_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| FCE_KRST0 | Kernel Reset Register 0 | $0F4_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| FCE_ACCEN1 | Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| FCE_ACCEN0 | Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| FCE_IRi (i=0-7) | Input Register i | $100_H$+i*20$_H$ | U,SV | P,U,SV | Application Reset | See Family Spec |
| FCE_RESi (i=0-7) | CRC Result Register i | $104_H$+i*20$_H$ | U,SV | BE | Application Reset | See Family Spec |
| FCE_CFGi (i=0-7) | CRC Configuration Register i | $108_H$+i*20$_H$ | U,SV | P,E,SV | Application Reset | See Family Spec |
| FCE_STSi (i=0-7) | CRC Status Register i | $10C_H$+i*20$_H$ | U,SV | P,U,SV | Application Reset | See Family Spec |

**Table 207 Register Overview - FCE (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| FCE_LENGTHi (i=0-7) | CRC Length Register i | $110_H$+i*20$_H$ | U,SV | P,U,SV | Application Reset | See Family Spec |
| FCE_CHECKi (i=0-7) | CRC Check Register i | $114_H$+i*20$_H$ | U,SV | P,U,SV | Application Reset | See Family Spec |
| FCE_CRCi (i=0-7) | CRC Regsister i | $118_H$+i*20$_H$ | U,SV | P,U,SV | Application Reset | See Family Spec |
| FCE_CTRi (i=0-7) | CRC Test Register i | $11C_H$+i*20$_H$ | U,SV | P,U,SV | Application Reset | See Family Spec |

## 17.3 TC35x Specific Registers

No deviations from the Family Spec

## 17.4 Connectivity

**Table 208 Connections of FCE**

| Interface Signals | connects | | Description |
|---|---|---|---|
| FCE:SRC_FCE | to | INT:fce0.SRC_FCE | FCE Service Request |

## 17.5 Revision History

**Table 209 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V4.2.9** | | |
| | No functional changes. | |

# 18 Direct Memory Access (DMA)

This is the TC35x specific information related to the DMA module of the AURIXTC3XX product family.

## 18.1 TC35x Specific IP Configuration

The TC35x DMA contains 64 DMA channels.

## 18.2 TC35x Specific Register Set

**Table 210 Register Address Space - DMA**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| DMA | $F0010000_H$ | $F0013FFF_H$ | FPI slave interface |

**Table 211 Register Overview - DMA (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| DMA_CLC | DMA Clock Control Register | $0000_H$ | U,SV | SV,E,P00,P01 | Application Reset | See Family Spec |
| DMA_ID | DMA Identification Register | $0008_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_ACCENr0 (r=0-3) | RP r Access Enable Register 0 | $0040_H$+r*8 | U,SV | SV,SE | Application Reset | See Family Spec |
| DMA_ACCENr1 (r=0-3) | RP r Access Enable Register 1 | $0044_H$+r*8 | U,SV | nBE | Application Reset | See Family Spec |
| DMA_EERm (m=0-1) | ME m Enable Error Register | $0120_H$+m*$1000_H$ | U,SV | SV | Application Reset | See Family Spec |
| DMA_ERRSRm (m=0-1) | ME m Error Status Register | $0124_H$+m*$1000_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_CLREm (m=0-1) | ME m Clear Error Register | $0128_H$+m*$1000_H$ | U,SV | SV | Application Reset | See Family Spec |
| DMA_MEmSR (m=0-1) | ME m Status Register | $0130_H$+m*$1000_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_MEm0R (m=0-1) | ME m Read Register 0 | $0140_H$+m*$1000_H$ | U,SV | BE | Application Reset | See Family Spec |

**Table 211 Register Overview - DMA (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| DMA_MEm1R (m=0-1) | ME m Read Register 1 | $0144_H$+m *$1000_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_MEm2R (m=0-1) | ME m Read Register 2 | $0148_H$+m *$1000_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_MEm3R (m=0-1) | ME m Read Register 3 | $014C_H$+ m*$1000_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_MEm4R (m=0-1) | ME m Read Register 4 | $0150_H$+m *$1000_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_MEm5R (m=0-1) | ME m Read Register 5 | $0154_H$+m *$1000_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_MEm6R (m=0-1) | ME m Read Register 6 | $0158_H$+m *$1000_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_MEm7R (m=0-1) | ME m Read Register 7 | $015C_H$+ m*$1000_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_MEmRDCRC (m=0-1) | ME m Channel Read Data CRC Register | $0180_H$+m *$1000_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_MEmSDCRC (m=0-1) | ME m Channel Source and Destination Address CRC Register | $0184_H$+m *$1000_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_MEmSADR (m=0-1) | ME m Channel Source Address Register | $0188_H$+m *$1000_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_MEmDADR (m=0-1) | ME m Channel Destination Address Register | $018C_H$+ m*$1000_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_MEmADICR (m=0-1) | ME m Channel Address and Interrupt Control Register | $0190_H$+m *$1000_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_MEmCHCR (m=0-1) | ME m Channel Control Register | $0194_H$+m *$1000_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_MEmSHADR (m=0-1) | ME m Channel Shadow Address Register | $0198_H$+m *$1000_H$ | U,SV | BE | Application Reset | See Family Spec |

**Table 211 Register Overview - DMA (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| DMA_MEmCHSR (m=0-1) | ME m Channel Status Register | $019C_H$+ m*$1000_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_OTSS | DMA OCDS Trigger Set Select | $1200_H$ | U,SV | SV | See Family Spec | See Family Spec |
| DMA_PRR0 | DMA Pattern Read Register 0 | $1208_H$ | U,SV | SV | Application Reset | See Family Spec |
| DMA_PRR1 | DMA Pattern Read Register 1 | $120C_H$ | U,SV | SV | Application Reset | See Family Spec |
| DMA_TIME | DMA Time Register | $1210_H$ | U,SV | BE | Application Reset | See Family Spec |
| DMA_MODEr (r=0-3) | RP r Mode Register | $1300_H$+r* 4 | U,SV | SV,SE,P00, P01 | Application Reset | See Family Spec |
| DMA_ERRINTRr (r=0-3) | RP r Error Interrupt Set Register | $1320_H$+r* 4 | U,SV | SV,Pr | Application Reset | See Family Spec |
| DMA_HRRc (c=000-63) | DMA Channel c Resource Partition Register | $1800_H$+c *4 | U,SV | SV,SE,P00, P01 | Application Reset | See Family Spec |
| DMA_SUSENRc (c=000-63) | DMA Channel c Suspend Enable Register | $1A00_H$+c *4 | U,SV | SV,E,Pr | See Family Spec | See Family Spec |
| DMA_SUSACRc (c=000-63) | DMA Channel c Suspend Acknowledge Register | $1C00_H$+c *4 | U,SV | BE | See Family Spec | See Family Spec |
| DMA_TSRc (c=000-63) | DMA Channel c Transaction State Register | $1E00_H$+c *4 | U,SV | SV,Pr | Application Reset | See Family Spec |
| DMA_RDCRCRc (c=000-63) | DMARAM Channel c Read Data CRC Register | $2000_H$+c *$20_H$ | U,SV | SV,Pr | Application Reset | See Family Spec |
| DMA_SDCRCRc (c=000-63) | DMARAM Channel c Source and Destination Address CRC Register | $2004_H$+c *$20_H$ | U,SV | SV,Pr | Application Reset | See Family Spec |
| DMA_SADRc (c=000-63) | DMARAM Channel c Source Address Register | $2008_H$+c *$20_H$ | U,SV | SV,Pr | Application Reset | See Family Spec |

**Table 211 Register Overview - DMA (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| DMA_DADRc (c=000-63) | DMARAM Channel c Destination Address Register | $200C_H$+c *$20_H$ | U,SV | SV,Pr | Application Reset | See Family Spec |
| DMA_ADICRc (c=000-63) | DMARAM Channel c Address and Interrupt Control Register | $2010_H$+c *$20_H$ | U,SV | SV,Pr | Application Reset | See Family Spec |
| DMA_CHCFGRc (c=000-63) | DMARAM Channel c Configuration Register | $2014_H$+c *$20_H$ | U,SV | SV,Pr | Application Reset | See Family Spec |
| DMA_SHADRc (c=000-63) | DMARAM Channel c Shadow Address Register | $2018_H$+c *$20_H$ | U,SV | SV,Pr | Application Reset | See Family Spec |
| DMA_CHCSRc (c=000-63) | DMARAM Channel c Control and Status Register | $201C_H$+c *$20_H$ | U,SV | SV,Pr | Application Reset | See Family Spec |

## 18.3 TC35x Specific Registers

No deviations from the Family Spec

## 18.4 Connectivity

**Table 212 Connections of DMA**

| Interface Signals | connects | | Description |
|---|---|---|---|
| DMA:fpi0_sleep_n | from | SCU:scu_syst_sleep_n | Sleep Control |
| DMA:ERR_INT(3:0) | to | INT:dma.ERR_INT(3:0) | DMA Error Service Request |
| DMA:CH_INT(127:0) | to | INT:dma.CH_INT(127:0) | DMA Channel Service Request |

## 18.5 Revision History

**Table 213 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V0.1.15** | | |
| - | Connectivity table updated. | |
| **V0.1.16** | | |
| – | No functional changes. | |
| **V0.1.17** | | |
| – | No functional changes. | |
| **V0.1.18** | | |
| – | No functional changes. | |

# 19 Signal Processing Unit (SPU)

This is the device specific information related to the AURIX™ TC35x version of the SPU.

## 19.1 TC35x Specific IP Configuration

There is no specific configuration of the SPU for this device

## 19.2 TC35x Specific Register Set

**Table 214 Register Address Space - SPU**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| SPU0 | $FA800000_H$ | $FA8007FF_H$ | BBB Slave Interface to the SPU Special Function Registers |
| (SPU0) | $FAA00000_H$ | $FAA0FFFF_H$ | BBB Slave Interface for Accessing the SPU Config RAM |
| SPU1 | $FAC00000_H$ | $FAC007FF_H$ | BBB Slave Interface to the SPU Special Function Registers |
| (SPU1) | $FAE00000_H$ | $FAE0FFFF_H$ | BBB Slave Interface for Accessing the SPU Config RAM |

**Table 215 Register Overview - SPU0 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| SPU0_CLC | Clock Control | $00000_H$ | See Family Spec |
| SPU0_MODID | Module Identification Register | $00004_H$ | See Family Spec |
| SPU0_STAT | Status and Reporting | $00008_H$ | See Family Spec |
| SPU0_ID_CONF | Input DMA Configuration | $00030_H$ | See Family Spec |
| SPU0_ID_CONF2 | Input DMA Configuration 2 | $00034_H$ | See Family Spec |
| SPU0_ID_RM_CONF | Input DMA Configuration: Radar Memory | $00038_H$ | See Family Spec |
| SPU0_ID_RM_ILO | Inner Loop Address Offset | $0003C_H$ | See Family Spec |
| SPU0_ID_RM_OLO | Outer Loop Address Offset | $00040_H$ | See Family Spec |

**Table 215 Register Overview - SPU0 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| SPU0_ID_RM_BLO | Bin Offset Address Configuration | $00044_H$ | See Family Spec |
| SPU0_ID_RM_IOLR | Inner and Outer Loop Repeat | $00048_H$ | See Family Spec |
| SPU0_ID_RM_BLR | Bin Loop Repeat | $0004C_H$ | See Family Spec |
| SPU0_ID_RM_ACFG0 | Spare Configuration Register | $00050_H$ | See Family Spec |
| SPU0_ID_RM_ACFG1 | Spare Configuration Register | $00054_H$ | See Family Spec |
| SPU0_PACTR | Partial-Acquisition Counter | $00058_H$ | See Family Spec |
| SPU0_DPASS_CONF | Double Pass Configuration | $0005C_H$ | See Family Spec |
| SPU0_BEx_LDR_CONF (x=0-1) | Loader Configuration | $00060_H+x*40_H$ | See Family Spec |
| SPU0_BEx_LDR_CONF2 (x=0-1) | Loader Configuration Extended | $00064_H+x*40_H$ | See Family Spec |
| SPU0_BEx_Aj_ANTOFST (j=0-3;x=0-1) | Antenna Offset | $00068_H+x*40_H$ +j*4 | See Family Spec |
| SPU0_BEx_UNLDR_CONF (x=0-1) | Unloader Configuration | $00078_H+x*40_H$ | See Family Spec |
| SPU0_BEx_UNLDR_CONF2 (x=0-1) | Unloader Configuration 2 | $0007C_H+x*40_H$ | See Family Spec |
| SPU0_BEx_UNLDR_ACFG (x=0-1) | Spare Configuration Register | $00080_H+x*40_H$ | See Family Spec |
| SPU0_BEx_ODP_CONF (x=0-1) | Output Data Processor Configuration | $00084_H+x*40_H$ | See Family Spec |

**Table 215 Register Overview - SPU0 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| SPU0_BEx_NCICTRL (x=0-1) | NCI Control | $00088_H$+x*$40_H$ | See Family Spec |
| SPU0_BEx_SUMCTRL (x=0-1) | Summation Unit Control | $0008C_H$+x*$40_H$ | See Family Spec |
| SPU0_BEx_PWRSUM (x=0-1) | Power Summation | $00090_H$+x*$40_H$ | See Family Spec |
| SPU0_BEx_PWRCTRL (x=0-1) | Power Information Channel Control | $00094_H$+x*$40_H$ | See Family Spec |
| SPU0_BEx_CFARCTRL (x=0-1) | CFAR Module Control | $00098_H$+x*$40_H$ | See Family Spec |
| SPU0_BEx_SBCTRL (x=0-1) | Sideband Control | $0009C_H$+x*$40_H$ | See Family Spec |
| SPU0_BINm_REJ (m=0-63) | Bin Rejection Mask | $000E0_H$+m*4 | See Family Spec |
| SPU0_MAGAPPROX | Magnitude Approximation Constants | $001E0_H$ | See Family Spec |
| SPU0_NCISCALAR0 | NCI Antennae Scaling Factor | $001E4_H$ | See Family Spec |
| SPU0_NCISCALAR1 | NCI Antennae Scaling Factor | $001E8_H$ | See Family Spec |
| SPU0_NCISCALAR2 | NCI Antennae Scaling Factor | $001EC_H$ | See Family Spec |
| SPU0_NCISCALAR3 | NCI Antennae Scaling Factor | $001F0_H$ | See Family Spec |
| SPU0_CFARCFG | CFAR Configuration | $001F4_H$ | See Family Spec |
| SPU0_CFARCFG2 | CFAR Configuration 2 | $001F8_H$ | See Family Spec |

**Table 215 Register Overview - SPU0 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| SPU0_CFARCFG3 | CFAR Configuration 3 | $001FC_H$ | See Family Spec |
| SPU0_SCALARADD | Scalar Addition Operand | $00200_H$ | See Family Spec |
| SPU0_SCALARMULT | Scalar Multiplication Operand | $00204_H$ | See Family Spec |
| SPU0_BINREJCTRL | Bin Rejection Unit Control | $00208_H$ | See Family Spec |
| SPU0_LCLMAX | Local Maximum Control | $0020C_H$ | See Family Spec |
| SPU0_ACFG2 | Spare Configuration Register | $00210_H$ | See Family Spec |
| SPU0_REGCRC | Register CRC | $00218_H$ | See Family Spec |
| SPU0_CTRL | SPU Control | $0021C_H$ | See Family Spec |
| SPU0_MDq_METADATA (q=0-1) | Dataset Metadata | $00220_H$+q*$88_H$ | See Family Spec |
| SPU0_MDq_BINCOUNT (q=0-1) | Bin Rejection Unit Tracking | $00224_H$+q*$88_H$ | See Family Spec |
| SPU0_MDq_MASKm_ACCEPT (m=0-31;q=0-1) | Bin Acceptance Mask | $00228_H$+q*$88_H$ +m*4 | See Family Spec |
| SPU0_IDMCNT | Input DMA Count | $00330_H$ | See Family Spec |
| SPU0_IBMCNT | Input Buffer Memory Count | $00334_H$ | See Family Spec |
| SPU0_LDRCNT | Input Buffer Memory Read Count | $00338_H$ | See Family Spec |

**Table 215 Register Overview - SPU0 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| SPU0_FFTWCNT | FFT Load Count | $0033C_H$ | See Family Spec |
| SPU0_FFTRCNT | FFT Unload Count | $00340_H$ | See Family Spec |
| SPU0_ULDRCNT | Output Buffer Memory Write Count | $00344_H$ | See Family Spec |
| SPU0_ODMCNT | Output Buffer Memory Read Count | $00348_H$ | See Family Spec |
| SPU0_BRCNT | Bin Rejection Unit Load Count | $0034C_H$ | See Family Spec |
| SPU0_CFARCNT | CFAR Unit Load Count | $00350_H$ | See Family Spec |
| SPU0_ODMACNTp (p=0-7) | Output DMA Port Write Count | $00354_H$+p*4 | See Family Spec |
| SPU0_CNTCLR | Safety Counter Clear | $00374_H$ | See Family Spec |
| SPU0_MONITOR | SPU Monitor | $00378_H$ | See Family Spec |
| SPU0_SMCTRL | Safety Mechanism Control Functions | $0037C_H$ | See Family Spec |
| SPU0_SMSTAT | Safety Mechanism Status | $00380_H$ | See Family Spec |
| SPU0_SMUSER | Safety Mechanism Control Functions (User) | $00384_H$ | See Family Spec |
| SPU0_DATAd_CRC (d=0-85) | Monitor CRC Register | $00388_H$+d*4 | See Family Spec |
| SPU0_CTRLe_CRC (e=0-24) | Monitor CRC Register | $00500_H$+e*4 | See Family Spec |

**Table 215 Register Overview - SPU0 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| SPU0_USROTC | User OCDS Trace Control | $007E0_H$ | See Family Spec |
| SPU0_ACCEN0 | Access Enable Register 0 | $007E4_H$ | See Family Spec |
| SPU0_ACCEN1 | Access Enable Register 1 | $007E8_H$ | See Family Spec |
| SPU0_OCS | OCDS Control and Status | $007EC_H$ | See Family Spec |
| SPU0_ODA | OCDS Debug Access Register | $007F0_H$ | See Family Spec |
| SPU0_KRST0 | Kernel Reset Register 0 | $007F4_H$ | See Family Spec |
| SPU0_KRST1 | Kernel Reset Register 1 | $007F8_H$ | See Family Spec |
| SPU0_KRSTCLR | Kernel Reset Clear | $007FC_H$ | See Family Spec |

**Table 216 Register Overview - SPU1 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| SPU1_CLC | Clock Control | $00000_H$ | See Family Spec |
| SPU1_MODID | Module Identification Register | $00004_H$ | See Family Spec |
| SPU1_STAT | Status and Reporting | $00008_H$ | See Family Spec |
| SPU1_ID_CONF | Input DMA Configuration | $00030_H$ | See Family Spec |

**Table 216 Register Overview - SPU1 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| SPU1_ID_CONF2 | Input DMA Configuration 2 | $00034_H$ | See Family Spec |
| SPU1_ID_RM_CONF | Input DMA Configuration: Radar Memory | $00038_H$ | See Family Spec |
| SPU1_ID_RM_ILO | Inner Loop Address Offset | $0003C_H$ | See Family Spec |
| SPU1_ID_RM_OLO | Outer Loop Address Offset | $00040_H$ | See Family Spec |
| SPU1_ID_RM_BLO | Bin Offset Address Configuration | $00044_H$ | See Family Spec |
| SPU1_ID_RM_IOLR | Inner and Outer Loop Repeat | $00048_H$ | See Family Spec |
| SPU1_ID_RM_BLR | Bin Loop Repeat | $0004C_H$ | See Family Spec |
| SPU1_ID_RM_ACFG0 | Spare Configuration Register | $00050_H$ | See Family Spec |
| SPU1_ID_RM_ACFG1 | Spare Configuration Register | $00054_H$ | See Family Spec |
| SPU1_PACTR | Partial-Acquisition Counter | $00058_H$ | See Family Spec |
| SPU1_DPASS_CONF | Double Pass Configuration | $0005C_H$ | See Family Spec |
| SPU1_BEx_LDR_CONF (x=0-1) | Loader Configuration | $00060_H+x*40_H$ | See Family Spec |
| SPU1_BEx_LDR_CONF2 (x=0-1) | Loader Configuration Extended | $00064_H+x*40_H$ | See Family Spec |
| SPU1_BEx_Aj_ANTOFST (j=0-3;x=0-1) | Antenna Offset | $00068_H+x*40_H+j*4$ | See Family Spec |

**Table 216 Register Overview - SPU1 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| SPU1_BEx_UNLDR_CONF (x=0-1) | Unloader Configuration | $00078_H$+x*$40_H$ | See Family Spec |
| SPU1_BEx_UNLDR_CONF2 (x=0-1) | Unloader Configuration 2 | $0007C_H$+x*$40_H$ | See Family Spec |
| SPU1_BEx_UNLDR_ACFG (x=0-1) | Spare Configuration Register | $00080_H$+x*$40_H$ | See Family Spec |
| SPU1_BEx_ODP_CONF (x=0-1) | Output Data Processor Configuration | $00084_H$+x*$40_H$ | See Family Spec |
| SPU1_BEx_NCICTRL (x=0-1) | NCI Control | $00088_H$+x*$40_H$ | See Family Spec |
| SPU1_BEx_SUMCTRL (x=0-1) | Summation Unit Control | $0008C_H$+x*$40_H$ | See Family Spec |
| SPU1_BEx_PWRSUM (x=0-1) | Power Summation | $00090_H$+x*$40_H$ | See Family Spec |
| SPU1_BEx_PWRCTRL (x=0-1) | Power Information Channel Control | $00094_H$+x*$40_H$ | See Family Spec |
| SPU1_BEx_CFARCTRL (x=0-1) | CFAR Module Control | $00098_H$+x*$40_H$ | See Family Spec |
| SPU1_BEx_SBCTRL (x=0-1) | Sideband Control | $0009C_H$+x*$40_H$ | See Family Spec |
| SPU1_BINm_REJ (m=0-63) | Bin Rejection Mask | $000E0_H$+m*4 | See Family Spec |
| SPU1_MAGAPPROX | Magnitude Approximation Constants | $001E0_H$ | See Family Spec |
| SPU1_NCISCALAR0 | NCI Antennae Scaling Factor | $001E4_H$ | See Family Spec |
| SPU1_NCISCALAR1 | NCI Antennae Scaling Factor | $001E8_H$ | See Family Spec |

**Table 216 Register Overview - SPU1 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| SPU1_NCISCALAR2 | NCI Antennae Scaling Factor | $001EC_H$ | See Family Spec |
| SPU1_NCISCALAR3 | NCI Antennae Scaling Factor | $001F0_H$ | See Family Spec |
| SPU1_CFARCFG | CFAR Configuration | $001F4_H$ | See Family Spec |
| SPU1_CFARCFG2 | CFAR Configuration 2 | $001F8_H$ | See Family Spec |
| SPU1_CFARCFG3 | CFAR Configuration 3 | $001FC_H$ | See Family Spec |
| SPU1_SCALARADD | Scalar Addition Operand | $00200_H$ | See Family Spec |
| SPU1_SCALARMULT | Scalar Multiplication Operand | $00204_H$ | See Family Spec |
| SPU1_BINREJCTRL | Bin Rejection Unit Control | $00208_H$ | See Family Spec |
| SPU1_LCLMAX | Local Maximum Control | $0020C_H$ | See Family Spec |
| SPU1_ACFG2 | Spare Configuration Register | $00210_H$ | See Family Spec |
| SPU1_REGCRC | Register CRC | $00218_H$ | See Family Spec |
| SPU1_CTRL | SPU Control | $0021C_H$ | See Family Spec |
| SPU1_MDq_METADATA (q=0-1) | Dataset Metadata | $00220_H+q*88_H$ | See Family Spec |
| SPU1_MDq_BINCOUNT (q=0-1) | Bin Rejection Unit Tracking | $00224_H+q*88_H$ | See Family Spec |

**Table 216 Register Overview - SPU1 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| SPU1_MDq_MASKm_ACCEPT (m=0-31;q=0-1) | Bin Acceptance Mask | $00228_H$+q*$88_H$ +m*4 | See Family Spec |
| SPU1_IDMCNT | Input DMA Count | $00330_H$ | See Family Spec |
| SPU1_IBMCNT | Input Buffer Memory Count | $00334_H$ | See Family Spec |
| SPU1_LDRCNT | Input Buffer Memory Read Count | $00338_H$ | See Family Spec |
| SPU1_FFTWCNT | FFT Load Count | $0033C_H$ | See Family Spec |
| SPU1_FFTRCNT | FFT Unload Count | $00340_H$ | See Family Spec |
| SPU1_ULDRCNT | Output Buffer Memory Write Count | $00344_H$ | See Family Spec |
| SPU1_ODMCNT | Output Buffer Memory Read Count | $00348_H$ | See Family Spec |
| SPU1_BRCNT | Bin Rejection Unit Load Count | $0034C_H$ | See Family Spec |
| SPU1_CFARCNT | CFAR Unit Load Count | $00350_H$ | See Family Spec |
| SPU1_ODMACNTp (p=0-7) | Output DMA Port Write Count | $00354_H$+p*4 | See Family Spec |
| SPU1_CNTCLR | Safety Counter Clear | $00374_H$ | See Family Spec |
| SPU1_MONITOR | SPU Monitor | $00378_H$ | See Family Spec |
| SPU1_SMCTRL | Safety Mechanism Control Functions | $0037C_H$ | See Family Spec |

**Table 216 Register Overview - SPU1 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| SPU1_SMSTAT | Safety Mechanism Status | $00380_H$ | See Family Spec |
| SPU1_SMUSER | Safety Mechanism Control Functions (User) | $00384_H$ | See Family Spec |
| SPU1_DATAd_CRC (d=0-85) | Monitor CRC Register | $00388_H$+d*4 | See Family Spec |
| SPU1_CTRLe_CRC (e=0-24) | Monitor CRC Register | $00500_H$+e*4 | See Family Spec |
| SPU1_USROTC | User OCDS Trace Control | $007E0_H$ | See Family Spec |
| SPU1_ACCEN0 | Access Enable Register 0 | $007E4_H$ | See Family Spec |
| SPU1_ACCEN1 | Access Enable Register 1 | $007E8_H$ | See Family Spec |
| SPU1_OCS | OCDS Control and Status | $007EC_H$ | See Family Spec |
| SPU1_ODA | OCDS Debug Access Register | $007F0_H$ | See Family Spec |
| SPU1_KRST0 | Kernel Reset Register 0 | $007F4_H$ | See Family Spec |
| SPU1_KRST1 | Kernel Reset Register 1 | $007F8_H$ | See Family Spec |
| SPU1_KRSTCLR | Kernel Reset Clear | $007FC_H$ | See Family Spec |

## 19.3 TC35x Specific Registers

No deviations from the Family Spec

## 19.4 Connectivity

Connectivity of the SPUs in the AURIX™ TC35x is as follows

**Table 217 Connections of SPU0**

| Interface Signals | connects | | Description |
|---|---|---|---|
| SPU0:SD | to | SPU0:SDI0 | SPU Done Output |
| | | SPU1:SDI0 | |
| SPU0:SDI0 | from | SPU0:SD | Done indication from SPU0 |
| SPU0:SDI1 | from | SPU1:SD | Done Indication from SPU1 |
| SPU0:safety_alarm | to | SMU:safety_alarm=spu.spu0_safety_alarm.safety_alarm | SPU Alarm |
| SPU0:INT | to | INT:spu0.INT | SPU Service Request |
| SPU0:ERR | | INT:spu0.ERR | |

**Table 218 Connections of SPU1**

| Interface Signals | connects | | Description |
|---|---|---|---|
| SPU1:SD | to | SPU0:SDI1 | SPU Done Output |
| | | SPU1:SDI1 | |
| SPU1:SDI0 | from | SPU0:SD | Done indication from SPU0 |
| SPU1:SDI1 | from | SPU1:SD | Done Indication from SPU1 |
| SPU1:safety_alarm | to | SMU:safety_alarm=spu.spu1_safety_alarm.safety_alarm | SPU Alarm |
| SPU1:INT | to | INT:spu1.INT | SPU Service Request |
| SPU1:ERR | | INT:spu1.ERR | |

## 19.5 Revision History

**Table 219 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V1.1.20** | | |
| **Page 12** | Previous versions removed from revision history. | |
| **V1.1.21** | | |
| All | Text Insets updated for new tools and source versions. All tables updated. No functional changes. | |
| **V1.1.22** | | |
| All | Text Insets updated as part of document generation flow. All tables updated. No functional changes. | |
| **V1.1.23** | | |
| **Page 11** | **Section 19.4** Tables updated to fix formatting error in interrupt connections | |
| **V1.1.24** | | |

**Table 219 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| All | Text Insets updated as part of document generation flow. All tables updated. No functional changes. | |
| **V1.1.25** | | |
| All | Text Insets updated as part of document generation flow. All tables updated. No functional changes. | |

# 20 SPU Lockstep Module (SPULCKSTP)

This describes the TC35x specific customisations of the lockstep module for the SPU.

## 20.1 TC35x Specific IP Configuration

There is no device specific customisation

## 20.2 TC35x Specific Register Set

**Table 220 Register Address Space - SPULCKSTP**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| SPULCKSTP | $FA700000_H$ | $FA7000FF_H$ | SPU LOCKSTEP SFR Registers |

**Table 221 Register Overview - SPULCKSTP (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| SPULCKSTP_CLC | Clock Control | $000_H$ | See Family Spec |
| SPULCKSTP_MODID | Module Identification Register | $004_H$ | See Family Spec |
| SPULCKSTP_CTRL | SPU Lockstep Control | $010_H$ | See Family Spec |
| SPULCKSTP_ERROR | Error Monitoring Register | $018_H$ | See Family Spec |
| SPULCKSTP_ERRCLR | Error Clear | $01C_H$ | See Family Spec |
| SPULCKSTP_TEST | Alarm Test Register | $020_H$ | See Family Spec |
| SPULCKSTP_SPUCTRL | SPU Control | $024_H$ | See Family Spec |
| SPULCKSTP_ACCEN0 | Access Enable Register 0 | $0E4_H$ | See Family Spec |
| SPULCKSTP_ACCEN1 | Access Enable Register 1 | $0E8_H$ | See Family Spec |

## 20.3 TC35x Specific Registers

There are no registers specific to the TC35x

## 20.4 Connectivity

Empty section

## 20.5 Revision History

**Table 222 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V1.2.5** | | |
| All | No changes to previous version. | |

# 21 Extended Memory (EMEM)

This is the TC35x specific information related to the EMEM module of the AURIXTC3XX product family.

## 21.1 TC35x Specific IP Configuration

The TC35x EMEM contains 2 Mbyte of extension memory in two instances of the EMEM module.

## 21.2 TC35x Specific Register Set

**Table 223 Register Address Space - EMEM**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| EMEM | $FA006000_H$ | $FA0060FF_H$ | BPI SFF (access to EMEM core registers) |

**Table 224 Register Address Space - EMEM_MPU**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| EMEMMPU0 | $FB000000_H$ | $FB00FFFF_H$ | SRI slave interface 0 (access to EMEM module registers) |
| EMEMMPU1 | $FB010000_H$ | $FB01FFFF_H$ | SRI slave interface 1 (access to EMEM module registers) |

**Table 225 Register Address Space - EMEM_RAM**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| (EMEMRAM0) | $99000000_H$ | $990FFFFF_H$ | BBB slave interface 0 (access to EMEM module RAM, cached segment) |
| | $99000000_H$ | $990FFFFF_H$ | SRI slave interface 0 (access to EMEM module RAM, cached segment) |
| | $B9000000_H$ | $B90FFFFF_H$ | BBB slave interface 0 (access to EMEM module RAM, non-cached segment) |
| | $B9000000_H$ | $B90FFFFF_H$ | SRI slave interface 0 (access to EMEM module RAM, non-cached segment) |
| (EMEMRAM1) | $99100000_H$ | $991FFFFF_H$ | BBB slave interface 1 (access to EMEM module RAM, cached segment) |
| | $99100000_H$ | $991FFFFF_H$ | SRI slave interface 1 (access to EMEM module RAM, cached segment) |
| | $B9100000_H$ | $B91FFFFF_H$ | BBB slave interface 1 (access to EMEM module RAM, non-cached segment) |
| | $B9100000_H$ | $B91FFFFF_H$ | SRI slave interface 1 (access to EMEM module RAM, non-cached segment) |

**Table 226 Register Address Space - XTM**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| (XTM) | $B9400000_H$ | $B947FFFF_H$ | XTM FPI slave interface |

**Table 227 Register Overview - EMEM (ascending Offset Address)**

| Short Name | Description | Offset Address | Access Mode | | Page Number |
|---|---|---|---|---|---|
| | | | Read | Write | |
| EMEM_CLC | EMEM Core Clock Control Register | $0000_H$ | U,SV | SV,E,P | See Family Spec |
| EMEM_ID | EMEM Core Module Identification Register | $0008_H$ | U,SV | BE | See Family Spec |
| EMEM_TILECONFIG | EMEM Core Tile Configuration Register | $0020_H$ | U,SV | U,SV,P | See Family Spec |
| EMEM_TILECC | EMEM Core Tile Control Common Memory Register | $0024_H$ | U,SV | U,SV,P | See Family Spec |
| EMEM_TILECT | EMEM Core Tile Control Trace Memory Register | $0028_H$ | U,SV | U,SV,P | See Family Spec |
| EMEM_TILESTATE | EMEM Core Tile Status Register | $002C_H$ | U,SV | BE | See Family Spec |
| EMEM_SBRCTR | EMEM Core Standby RAM Control Register | $0034_H$ | U,SV | U,SV,P | See Family Spec |
| EMEM_ACCEN1 | EMEM Core Access Enable Register 1 | $00F8_H$ | U,SV | BE | See Family Spec |
| EMEM_ACCEN0 | EMEM Core Access Enable Register 0 | $00FC_H$ | U,SV | SV,SE | See Family Spec |

**Table 228 Register Overview - EMEMMPU0 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| EMEMMPU0_CLC | EMEM Module Clock Control Register | $00000_H$ | SV | SV,E,P | Application Reset | See Family Spec |
| EMEMMPU0_MODID | EMEM Module ID Register | $00008_H$ | SV | R | Application Reset | See Family Spec |
| EMEMMPU0_ACCEN0 | EMEM Module Access Enable Register 0 | $00010_H$ | SV | SV,SE | Application Reset | See Family Spec |

**Table 228 Register Overview - EMEMMPU0 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| EMEMMPU0_ACCEN1 | EMEM Module Access Enable Register 1 | $00014_H$ | SV | SV,SE | Application Reset | See Family Spec |
| EMEMMPU0_MEMCON | EMEM Module Memory Control Register | $00020_H$ | SV | SV,E,P | Application Reset | See Family Spec |
| EMEMMPU0_SCTRL | EMEM Module Safety Control Register | $00024_H$ | SV | SV,SE,P | Application Reset | See Family Spec |
| EMEMMPU0_RGNLAi (i=0-7) | EMEM Module Region i Lower Address Register | $00050_H$+i *$10_H$ | SV | SV,SE,P | Application Reset | See Family Spec |
| EMEMMPU0_RGNUAi (i=0-7) | EMEM Module Region i Upper Address Register | $00054_H$+i *$10_H$ | SV | SV,SE,P | Application Reset | See Family Spec |
| EMEMMPU0_RGNACCENWAi (i=0-7) | EMEM Module Region i Write Access Enable Register 0 | $00058_H$+i *$10_H$ | SV | SV,SE,P | Application Reset | See Family Spec |
| EMEMMPU0_RGNACCENWBi (i=0-7) | EMEM Module Region i Write Access Enable Register 1 | $0005C_H$+i *$10_H$ | SV | SV,SE,P | Application Reset | See Family Spec |
| EMEMMPU0_RGNACCENRAi (i=0-7) | EMEM Module Region i Read Access Enable Register 0 | $000D8_H$+ i*$10_H$ | SV | SV,SE,P | Application Reset | See Family Spec |
| EMEMMPU0_RGNACCENRBi (i=0-7) | EMEM Module Region i Read Access Enable Register 1 | $000DC_H$+ i*$10_H$ | SV | SV,SE,P | Application Reset | See Family Spec |

**Table 229 Register Overview - EMEMMPU1 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| EMEMMPU1_CLC | EMEM Module Clock Control Register | $00000_H$ | SV | SV,E,P | Application Reset | See Family Spec |
| EMEMMPU1_MODID | EMEM Module ID Register | $00008_H$ | SV | R | Application Reset | See Family Spec |
| EMEMMPU1_ACCEN0 | EMEM Module Access Enable Register 0 | $00010_H$ | SV | SV,SE | Application Reset | See Family Spec |

**Table 229 Register Overview - EMEMMPU1 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| EMEMMPU1_ACCEN1 | EMEM Module Access Enable Register 1 | $00014_H$ | SV | SV,SE | Application Reset | See Family Spec |
| EMEMMPU1_MEMCON | EMEM Module Memory Control Register | $00020_H$ | SV | SV,E,P | Application Reset | See Family Spec |
| EMEMMPU1_SCTRL | EMEM Module Safety Control Register | $00024_H$ | SV | SV,SE,P | Application Reset | See Family Spec |
| EMEMMPU1_RGNLAi (i=0-7) | EMEM Module Region i Lower Address Register | $00050_H$+i *$10_H$ | SV | SV,SE,P | Application Reset | See Family Spec |
| EMEMMPU1_RGNUAi (i=0-7) | EMEM Module Region i Upper Address Register | $00054_H$+i *$10_H$ | SV | SV,SE,P | Application Reset | See Family Spec |
| EMEMMPU1_RGNACCENWAi (i=0-7) | EMEM Module Region i Write Access Enable Register 0 | $00058_H$+i *$10_H$ | SV | SV,SE,P | Application Reset | See Family Spec |
| EMEMMPU1_RGNACCENWBi (i=0-7) | EMEM Module Region i Write Access Enable Register 1 | $0005C_H$+i *$10_H$ | SV | SV,SE,P | Application Reset | See Family Spec |
| EMEMMPU1_RGNACCENRAi (i=0-7) | EMEM Module Region i Read Access Enable Register 0 | $000D8_H$+ i*$10_H$ | SV | SV,SE,P | Application Reset | See Family Spec |
| EMEMMPU1_RGNACCENRBi (i=0-7) | EMEM Module Region i Read Access Enable Register 1 | $000DC_H$+ i*$10_H$ | SV | SV,SE,P | Application Reset | See Family Spec |

## 21.3 TC35x Specific Registers

There are no TC35x specific registers in the EMEM.

## 21.4 Connectivity

Nothing included at this release

## 21.5 Revision History

**Table 230 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V1.3.12** | | |
| | Updated to align with EMEM_AURIXTC3XX V1.3.12 specification chapter. | |
| **V1.3.13** | | |
| | No changes. | |
| **V1.3.14** | | |
| **Page 1** | Correction of configuration size in chapter 1.1 | |
| **V1.4.1** | | |
| **Page 2** | Add extra registers TILESTATE1 and TILECONFIG1 to support increased memory size. | |
| **V1.4.2** | | |
| – | No functional changes. | |
| **V1.4.3** | | |
| – | No functional changes. | |
| **V1.4.4** | | |
| – | No functional changes. | |

# 22 Radar Interface (RIF)

This chapter describes the Radar Interface (RIF) module of the TC35x.

## 22.1 TC35x Specific IP Configuration

See features in the family spec.

**Table 231 TC35x specific configuration of RIF**

| Parameter | RIF0 | RIF1 |
|---|---|---|
| **Software Triggered Reset of the Module Kernel** This reset does not affect the bus interfaces and therefore cannot cause a protocol violation. Other outputs are synchronously forced to the idle state | Kernel Reset (software controlled by KRST0-1 registers) | Kernel Reset (software controlled by KRST0-1 registers) |

## 22.2 TC35x Specific Register Set

### 22.2.1 Address Map

**Table 232 Register Address Space - RIF**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| RIF0 | $FA040000_H$ | $FA0401FF_H$ | FPI slave interface |
| RIF1 | $FA040200_H$ | $FA0403FF_H$ | FPI slave interface |

*Note: The absolute register address is calculated as follows: Module Base Address + Offset Address*

A register is addressed word wise.

## 22.3 TC35x Specific Registers

No deviations from the Family Spec

## 22.4 Connectivity

**Table 233 Connections of RIF0**

| Interface Signals | connects | | Description |
|---|---|---|---|
| RIF0:RAMP1B | from | P02.6:IN | External RAMP B input |
| RIF0:safety_alarm | to | SMU:rif0_safety_alarm | RIF Alarm |
| RIF0:ERR | to | INT:rif0.ERR | Radar Interface Service Request |
| RIF0:INT | | INT:rif0.INT | |

**Table 234 Connections of RIF0**

| Interface Signals | connects | | Description |
|---|---|---|---|
| RIF0:CLKN | from | TC35x:P50.4 | LVDS RX Input (inverted Serial Clock) |
| RIF0:CLKP | from | TC35x:P50.5 | LVDS RX Input (Serial Clock) |
| RIF0:D1N | from | TC35x:P50.0 | LVDS RX Input (inverted Data Bits of Channel #0) |
| RIF0:D2N | from | TC35x:P50.2 | LVDS RX Input (inverted Data Bits of Channel #1) |
| RIF0:D3N | from | TC35x:P50.8 | LVDS RX Input (inverted Data Bits of Channel #2) |
| RIF0:D4N | from | TC35x:P50.10 | LVDS RX Input (inverted Data Bits of Channel #3) |
| RIF0:D1P | from | TC35x:P50.1 | LVDS RX Input (Data Bits of Channel #0) |
| RIF0:D2P | from | TC35x:P50.3 | LVDS RX Input (Data Bits of Channel #1) |
| RIF0:D3P | from | TC35x:P50.9 | LVDS RX Input (Data Bits of Channel #2) |
| RIF0:D4P | from | TC35x:P50.11 | LVDS RX Input (Data Bits of Channel #3) |
| RIF0:FRN | from | TC35x:P50.6 | LVDS RX Input (inverted FrameClock) |
| RIF0:FRP | from | TC35x:P50.7 | LVDS RX Input (FrameClock) |

**Table 235 Connections of RIF1**

| Interface Signals | connects | | Description |
|---|---|---|---|
| RIF1:RAMP1B | from | P10.8:IN | External RAMP B input |
| RIF1:safety_alarm | to | SMU:rif1_safety_alarm | RIF Alarm |
| RIF1:ERR | to | INT:rif1.ERR | Radar Interface Service Request |
| RIF1:INT | | INT:rif1.INT | |

**Table 236 Connections of RIF1**

| Interface Signals | connects | | Description |
|---|---|---|---|
| RIF1:CLKN | from | TC35x:P51.4 | LVDS RX Input (inverted Serial Clock) |
| RIF1:CLKP | from | TC35x:P51.5 | LVDS RX Input (Serial Clock) |
| RIF1:D1N | from | TC35x:P51.0 | LVDS RX Input (inverted Data Bits of Channel #0) |
| RIF1:D2N | from | TC35x:P51.2 | LVDS RX Input (inverted Data Bits of Channel #1) |
| RIF1:D3N | from | TC35x:P51.8 | LVDS RX Input (inverted Data Bits of Channel #2) |
| RIF1:D4N | from | TC35x:P51.10 | LVDS RX Input (inverted Data Bits of Channel #3) |
| RIF1:D1P | from | TC35x:P51.1 | LVDS RX Input (Data Bits of Channel #0) |
| RIF1:D2P | from | TC35x:P51.3 | LVDS RX Input (Data Bits of Channel #1) |
| RIF1:D3P | from | TC35x:P51.9 | LVDS RX Input (Data Bits of Channel #2) |

**Table 236 Connections of RIF1** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| RIF1:D4P | from | TC35x:P51.11 | LVDS RX Input (Data Bits of Channel #3) |
| RIF1:FRN | from | TC35x:P51.6 | LVDS RX Input (inverted FrameClock) |
| RIF1:FRP | from | TC35x:P51.7 | LVDS RX Input (FrameClock) |

## 22.5 Revision History

**Table 237 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V1.0.36** | | |
| All | Register and Connectivity Tables updated | |
| **V1.0.37** | | |
| – | No functional changes. | – |
| **V1.0.38** | | |
| – | No functional changes. | – |
| **V1.0.39** | | |
| – | Device specific registers, (RIF0_FLM, RIF1_FLM, RIF0_INTCON, RIF1_INTCON, RIF0_FLAGSSET, RIF1_FLAGSSET) are moved from the family spec to the appendix.<br>No functional changes. | – |
| **V1.0.40** | | |
| – | For registers from ESI to REGCRC, Kernel reset values are added for clarification.<br>Device specific registers, RIF0_FLAGSCL and RIF1_FLAGSCL are moved from the family spec to the appendix. | |
| – | Device specific registers (RIF0_DBGDLY0, RIF1_DBGDLY0, RIF0_DBGDLY1, RIF1_DBGDLY1) are moved from the family spec to the appendix. | |
| – | Device specific registers, RIF0_ESI and RIF1_ESI are moved from the family spec to the appendix. | |
| – | No functional changes. | – |
| **V1.0.41** | | |
| – | No functional changes. | – |
| **V1.0.42** | | |
| – | No functional changes. | – |
| **V1.0.43** | | |
| – | References to TC3Ax are removed | |

# 23 High Speed Pulse Density Modulation Module (HSPDM)

Text with reference to family spec.

## 23.1 TC35x Specific IP Configuration

See features in the family spec.

**Table 238 TC35x specific configuration of HSPDM**

| Parameter | HSPDM |
|---|---|
| **HSPDM ram** | $F0280000_H$ |
| **HSPDM ram size** | $2000_H$ |
| **HSPDM BPI registers** | $F0282000_H$ |
| **HSPDM BPI registers size** | $100_H$ |
| **SRAM size in byte** | 8192 |

## 23.2 TC35x Specific Register Set

There are no specific register set.

### 23.2.1 Address Map

**Table 239 Register Address Space - HSPDM**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| (HSPDM) | $F0280000_H$ | $F0281FFF_H$ | FPI slave interface for SRAM access |
| HSPDM | $F0282000_H$ | $F02820FF_H$ | FPI slave interface for BPI registers access |

*Note: The absolute register address is calculated as follows: Module Base Address + Offset Address*

A register is addressed word wise.

HSPDM RAM takes 8KBytes address space starting from 0xF028 0000 to 0xF028 1FFF.

## 23.3 TC35x Specific Registers

There are no device specific registers for HSPDM in TC35x.

## 23.4 Connectivity

There will be connections to the VADC.

### 23.4.1 Connections Regarding Hardware Run Feature

HSPDM module can be started by a CAN message. The reception (or transmision) of CAN messsage can trigger an interrupt. The interrupt signal can be delayed by a CCU6 timer module for a programmable time interval. The delayed interrupt signal can be used for starting the HSPDM module.

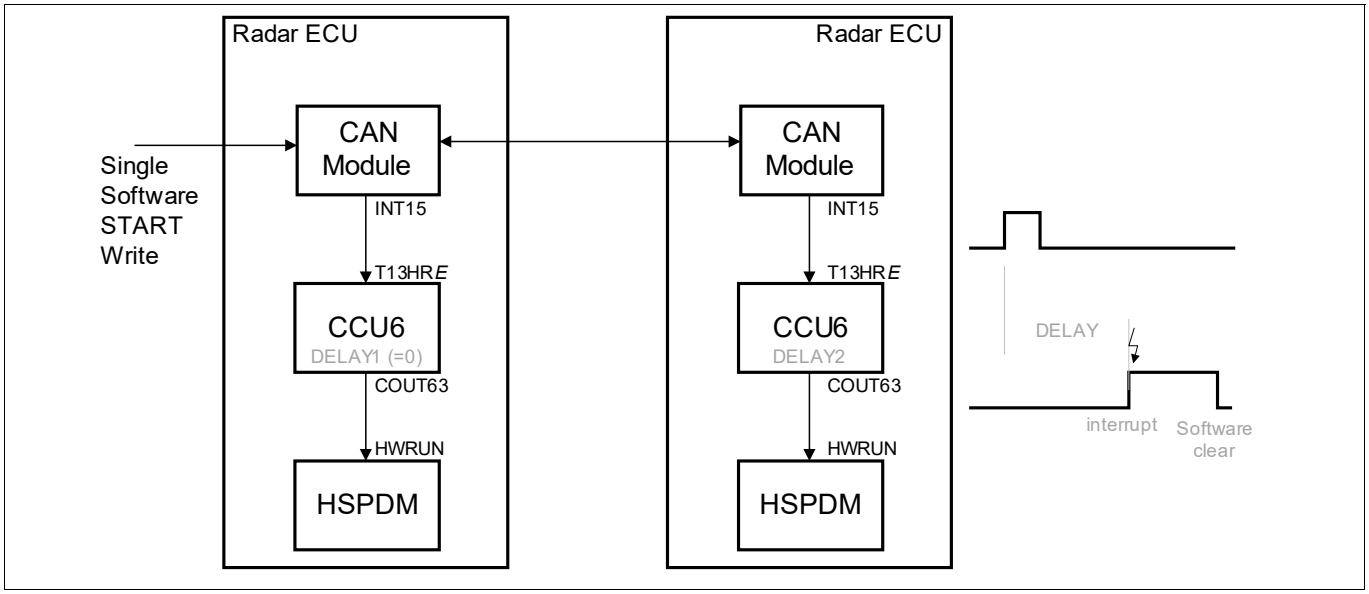



**Figure 6 Hardware Run Connections**

## 23.4.2 Pinning and Layout

The output signals of the HSPDM module are connected to the following pads:

- P22.3 - MUTE
- P22.4 - BS0
- P22.5 - BS1

## 23.5 Revision History

**Table 240 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V0.7.8** | | |
| | No change in appendix | |
| **V0.7.9** | | |
| **Page 2** | Typo “Histrory” fixed. | |

# 24 Camera and ADC Interface (CIF)

This device doesn’t contain a CIF module.

# 25 System Timer (STM)

This chapter describes the device specific details in TC35x.

## 25.1 TC35x Specific IP Configuration

See features in family spec

## 25.2 TC35x Specific Register Set

### Register Address Space Table

The address space for the module registers is defined below

**Table 241 Register Address Space - STM**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| STM0 | $F0001000_H$ | $F00010FF_H$ | FPI slave interface |
| STM1 | $F0001100_H$ | $F00011FF_H$ | FPI slave interface |
| STM2 | $F0001200_H$ | $F00012FF_H$ | FPI slave interface |

### Register Overview Table

There are no product specific register for this module.

## 25.3 TC35x Specific Registers

There are no product specific register for this module.

## 25.4 Connectivity

The tables below list all the connections of STM instances.

**Table 242 Connections of STM0**

| Interface Signals | connects | | Description |
|---|---|---|---|
| STM0:SR0_INT | to | CAN0:STM0.SR0_INT | System Timer Service Request 0 |
| | | CAN1:STM0.SR0_INT | |
| | | INT:stm0.SR0_INT | |
| STM0:SR1_INT | to | CAN0:STM0.SR1_INT | System Timer Service Request 1 |
| | | CAN1:STM0.SR1_INT | |
| | | INT:stm0.SR1_INT | |

**Table 243 Connections of STM1**

| Interface Signals | connects | | Description |
|---|---|---|---|
| STM1:SR0_INT | to | CAN0:STM1.SR0_INT | System Timer Service Request 0 |
| | | CAN1:STM1.SR0_INT | |
| | | INT:stm1.SR0_INT | |

**Table 243 Connections of STM1** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| STM1:SR1_INT | to | CAN0:STM1.SR1_INT | System Timer Service Request 1 |
| | | CAN1:STM1.SR1_INT | |
| | | INT:stm1.SR1_INT | |

**Table 244 Connections of STM2**

| Interface Signals | connects | | Description |
|---|---|---|---|
| STM2:SR0_INT | to | CAN0:STM2.SR0_INT | System Timer Service Request 0 |
| | | CAN1:STM2.SR0_INT | |
| | | INT:stm2.SR0_INT | |
| STM2:SR1_INT | to | CAN0:STM2.SR1_INT | System Timer Service Request 1 |
| | | CAN1:STM2.SR1_INT | |
| | | INT:stm2.SR1_INT | |

## 25.5 Revision History

**Table 245 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V9.2.3** | | |
| **Page 1** | Connection tables updated. | |
| **V9.2.4** | | |
| – | No changes. | |

# 26 Generic Timer Module (GTM)

This device doesn’t contain a GTM.

# 27 Capture/Compare Unit 6 (CCU6)

This chapter describes the specific properties of the product TC35x,
which is a member of the product family TC3XX.

The functionality of the CCU6 is described in the TC3XX family documentation. The complete product description consists of the family documentation and this product-specific appendix.

## 27.1 TC35x Specific Register Set

**Table 246 Register Address Space - CCU6**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| CCU60 | $F0002A00_H$ | $F0002AFF_H$ | FPI slave interface |
| CCU61 | $F0002B00_H$ | $F0002BFF_H$ | FPI slave interface |

*Note: Register MOSEL controls the trigger signals from both CCU6 kernels (CCU60 and CCU61) of the CCU6 module, and is only available in the address space of kernel CCU60.*

**Register Overview Tables of CCU6**

**Table 247 Register Overview - CCU60 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| CCU60_CLC | Clock Control Register | $0000_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| CCU60_MCFG | Module Configuration Register | $0004_H$ | U,SV | U,SV,P | See Family Spec | See Family Spec |
| CCU60_ID | Module Identification Register | $0008_H$ | U,SV | BE | See Family Spec | See Family Spec |
| CCU60_MOSEL | CCU60 Module Output Select Register | $000C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_PISEL0 | Port Input Select Register 0 | $0010_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_PISEL2 | Port Input Select Register 2 | $0014_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_KSCSR | Kernel State Control Sensitivity Register | $001C_H$ | U,SV | U,SV,P,OEN | See Family Spec | See Family Spec |
| CCU60_T12 | Timer T12 Counter Register | $0020_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 247 Register Overview - CCU60 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| CCU60_T12PR | Timer 12 Period Register | $0024_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_T12DTC | Dead-Time Control Register for Timer12 | $0028_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_CC6xR (x=0-2) | Capture/Compare Register for Channel CC6x | $0030_H$+x *4 | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_CC6xSR (x=0-2) | Capture/Compare Shadow Reg. for Channel CC6x | $0040_H$+x *4 | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_T13 | Timer T13 Counter Register | $0050_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_T13PR | Timer 13 Period Register | $0054_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_CC63R | Compare Register for T13 | $0058_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_CC63SR | Compare Shadow Register for T13 | $005C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_CMPSTAT | Compare State Register | $0060_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_CMPMODIF | Compare State Modification Register | $0064_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_T12MSEL | T12 Mode Select Register | $0068_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_TCTR0 | Timer Control Register 0 | $0070_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_TCTR2 | Timer Control Register 2 | $0074_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_TCTR4 | Timer Control Register 4 | $0078_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 247 Register Overview - CCU60 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| CCU60_MODCTR | Modulation Control Register | $0080_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_TRPCTR | Trap Control Register | $0084_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_PSLR | Passive State Level Register | $0088_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_MCMOUTS | Multi-Channel Mode Output Shadow Register | $008C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_MCMOUT | Multi-Channel Mode Output Register | $0090_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_MCMCTR | Multi-Channel Mode Control Register | $0094_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_IMON | Input Monitoring Register | $0098_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_LI | Lost Indicator Register | $009C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_IS | Interrupt Status Register | $00A0_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_ISS | Interrupt Status Set Register | $00A4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_ISR | Interrupt Status Reset Register | $00A8_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_INP | Interrupt Node Pointer Register | $00AC_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_IEN | Interrupt Enable Register | $00B0_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU60_OCS | OCDS Control and Status Register | $00E8_H$ | U,SV | SV,P,OEN | See Family Spec | See Family Spec |

**Table 247 Register Overview - CCU60 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| CCU60_KRSTCLR | Kernel Reset Status Clear Register | $00EC_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| CCU60_KRST1 | Kernel Reset Register 1 | $00F0_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| CCU60_KRST0 | Kernel Reset Register 0 | $00F4_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| CCU60_ACCEN0 | Access Enable Register 0 | $00FC_H$ | U,SV | SV,SE | Application Reset | See Family Spec |

**Table 248 Register Overview - CCU61 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| CCU61_CLC | Clock Control Register | $0000_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| CCU61_MCFG | Module Configuration Register | $0004_H$ | U,SV | U,SV,P | See Family Spec | See Family Spec |
| CCU61_ID | Module Identification Register | $0008_H$ | U,SV | BE | See Family Spec | See Family Spec |
| CCU61_PISEL0 | Port Input Select Register 0 | $0010_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_PISEL2 | Port Input Select Register 2 | $0014_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_KSCSR | Kernel State Control Sensitivity Register | $001C_H$ | U,SV | U,SV,P,OEN | See Family Spec | See Family Spec |
| CCU61_T12 | Timer T12 Counter Register | $0020_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_T12PR | Timer 12 Period Register | $0024_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 248 Register Overview - CCU61 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| CCU61_T12DTC | Dead-Time Control Register for Timer12 | $0028_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_CC6xR (x=0-2) | Capture/Compare Register for Channel CC6x | $0030_H$+x *4 | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_CC6xSR (x=0-2) | Capture/Compare Shadow Reg. for Channel CC6x | $0040_H$+x *4 | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_T13 | Timer T13 Counter Register | $0050_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_T13PR | Timer 13 Period Register | $0054_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_CC63R | Compare Register for T13 | $0058_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_CC63SR | Compare Shadow Register for T13 | $005C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_CMPSTAT | Compare State Register | $0060_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_CMPMODIF | Compare State Modification Register | $0064_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_T12MSEL | T12 Mode Select Register | $0068_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_TCTR0 | Timer Control Register 0 | $0070_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_TCTR2 | Timer Control Register 2 | $0074_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_TCTR4 | Timer Control Register 4 | $0078_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_MODCTR | Modulation Control Register | $0080_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 248 Register Overview - CCU61 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| CCU61_TRPCTR | Trap Control Register | $0084_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_PSLR | Passive State Level Register | $0088_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_MCMOUTS | Multi-Channel Mode Output Shadow Register | $008C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_MCMOUT | Multi-Channel Mode Output Register | $0090_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_MCMCTR | Multi-Channel Mode Control Register | $0094_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_IMON | Input Monitoring Register | $0098_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_LI | Lost Indicator Register | $009C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_IS | Interrupt Status Register | $00A0_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_ISS | Interrupt Status Set Register | $00A4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_ISR | Interrupt Status Reset Register | $00A8_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_INP | Interrupt Node Pointer Register | $00AC_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_IEN | Interrupt Enable Register | $00B0_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| CCU61_OCS | OCDS Control and Status Register | $00E8_H$ | U,SV | SV,P,OEN | See Family Spec | See Family Spec |
| CCU61_KRSTCLR | Kernel Reset Status Clear Register | $00EC_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |

**Table 248 Register Overview - CCU61 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| CCU61_KRST1 | Kernel Reset Register 1 | $00F0_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| CCU61_KRST0 | Kernel Reset Register 0 | $00F4_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| CCU61_ACCEN0 | Access Enable Register 0 | $00FC_H$ | U,SV | SV,SE | Application Reset | See Family Spec |

## 27.2 TC35x Specific Registers

No deviations from the Family Spec

## 27.3 Connectivity

**Table 249 Connections of CCU60**

| Interface Signals | connects | | Description |
|---|---|---|---|
| CCU60:CC60 | to | IOM:MON1(2) | T12 PWM channel 60 |
| | | IOM:REF1(6) | |
| | | P02.0:ALT(7) | |
| | | P02.6:ALT(7) | |
| | | P11.12:ALT(7) | |
| | | P15.6:ALT(7) | |
| | | P34.2:ALT(7) | |
| CCU60:CC61 | to | IOM:MON1(1) | T12 PWM channel 61 |
| | | IOM:REF1(5) | |
| | | P02.2:ALT(7) | |
| | | P02.7:ALT(7) | |
| | | P11.11:ALT(7) | |
| | | P15.5:ALT(7) | |
| CCU60:CC62 | to | IOM:MON1(0) | T12 PWM channel 62 |
| | | IOM:REF1(4) | |
| | | P02.4:ALT(7) | |
| | | P02.8:ALT(7) | |
| | | P11.10:ALT(7) | |
| | | P15.4:ALT(7) | |
| CCU60:CC60INA | from | P02.0:IN | T12 capture input 60 |
| CCU60:CC61INA | from | P02.2:IN | T12 capture input 61 |
| CCU60:CC62INA | from | P02.4:IN | T12 capture input 62 |

**Table 249 Connections of CCU60** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| CCU60:CC60INB | from | P00.1:IN | T12 capture input 60 |
| CCU60:CC61INB | from | P00.3:IN | T12 capture input 61 |
| CCU60:CC62INB | from | P00.5:IN | T12 capture input 62 |
| CCU60:CC60INC | from | P02.6:IN | T12 capture input 60 |
| CCU60:CC61INC | from | P02.7:IN | T12 capture input 61 |
| CCU60:CC62INC | from | P02.8:IN | T12 capture input 62 |
| CCU60:CC60IND | from | PMS:pms_wut_underflow | T12 capture input 60 |
| CCU60:CC62IND | from | SCU:E_PDOUT(4) | T12 capture input 62 |
| CCU60:CCPOS0A | from | P02.6:IN | Hall capture input 0 |
| CCU60:CCPOS1A | from | P02.7:IN | Hall capture input 1 |
| CCU60:CCPOS2A | from | P02.8:IN | Hall capture input 2 |
| CCU60:CCPOS0B | from | CCU61:SR(2) | Hall capture input 0 |
| CCU60:CCPOS0C | from | P10.4:IN | Hall capture input 0 |
| CCU60:CCPOS1C | from | P10.7:IN | Hall capture input 1 |
| CCU60:CCPOS2C | from | P10.8:IN | Hall capture input 2 |
| CCU60:COUT60 | to | SCU:E_REQ0(1) | T12 PWM channel 60 |
| | | IOM:MON1(3) | |
| | | IOM:REF1(3) | |
| | | P02.1:ALT(7) | |
| | | P11.9:ALT(7) | |
| | | P15.7:ALT(7) | |
| | | P34.3:ALT(7) | |
| CCU60:COUT61 | to | IOM:MON1(4) | T12 PWM channel 61 |
| | | IOM:REF1(2) | |
| | | P02.3:ALT(7) | |
| | | P11.6:ALT(7) | |
| | | P15.8:ALT(7) | |
| CCU60:COUT62 | to | IOM:MON1(5) | T12 PWM channel 62 |
| | | IOM:REF1(1) | |
| | | P02.5:ALT(7) | |
| | | P11.3:ALT(7) | |
| | | P14.0:ALT(7) | |

**Table 249 Connections of CCU60** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| CCU60:COUT63 | to | IOM:MON1(6) | T13 PWM channel 63 |
| | | IOM:REF1(0) | |
| | | P00.0:ALT(7) | |
| | | P11.2:ALT(7) | |
| | | P14.1:ALT(7) | |
| | | P32.4:ALT(7) | |
| | | P34.1:ALT(7) | |
| | | PMS:dcdc_sync_ccu6 | |
| CCU60:CTRAPA | from | P00.11:IN | Trap input capture |
| CCU60:CTRAPB | from | CCU60:WHE_N | Trap input capture |
| CCU60:CTRAPD | from | SCU:E_PDOUT(0) | Trap input capture |
| CCU60:SR(0) | to | HSM:EXT_INT(10) | Service request |
| CCU60:SR(1) | to | CCU60:T13HRH | Service request |
| CCU60:SR(2) | to | CCU61:CCPOS0B | Service request |
| | | CCU61:T12HRG | |
| | | CCU61:T13HRG | |
| CCU60:SR(3) | to | EVADC:G0REQTRA | Service request |
| | | EVADC:G1REQTRA | |
| CCU60:T12HRA | from | SCU:scu_cctrig0 | External timer start 12 |
| CCU60:T13HRA | from | SCU:scu_cctrig0 | External timer start 13 |
| CCU60:T12HRB | from | P00.7:IN | External timer start 12 |
| CCU60:T13HRB | from | P00.8:IN | External timer start 13 |
| CCU60:T12HRC | from | P00.9:IN | External timer start 12 |
| CCU60:T13HRC | from | P00.9:IN | External timer start 13 |
| CCU60:T12HRE | from | P00.0:IN | External timer start 12 |
| CCU60:T12HRF | from | GPT120:T6OFL | External timer start 12 |
| CCU60:T13HRF | from | GPT120:T6OFL | External timer start 13 |
| CCU60:T12HRG | from | CCU61:SR(2) | External timer start 12 |
| CCU60:T13HRG | from | CCU61:SR(2) | External timer start 13 |
| CCU60:T12HRH | from | SCU:E_PDOUT(0) | External timer start 12 |
| CCU60:T13HRH | from | CCU60:SR(1) | External timer start 13 |
| CCU60:TRIG(0) | to | EVADC:G0REQGTC | Output select trigger |
| | | EVADC:G1REQGTC | |
| CCU60:TRIG(1) | to | EVADC:G0REQGTD | Output select trigger |
| | | EVADC:G1REQGTD | |
| CCU60:TRIG(2) | to | EVADC:G0REQGTE | Output select trigger |
| | | EVADC:G1REQGTE | |
| CCU60:WHE_N | to | CCU60:CTRAPB | Set wrong hall event negative |

**Table 250 Connections of CCU61**

| Interface Signals | connects | | Description |
|---|---|---|---|
| CCU61:CC60 | to | IOM:MON1(8) | T12 PWM channel 60 |
| | | IOM:REF1(13) | |
| | | P00.1:ALT(7) | |
| | | P00.7:ALT(7) | |
| | | P20.8:ALT(7) | |
| | | P33.13:ALT(7) | |
| CCU61:CC61 | to | IOM:MON1(9) | T12 PWM channel 61 |
| | | IOM:REF1(12) | |
| | | P00.3:ALT(7) | |
| | | P00.8:ALT(7) | |
| | | P20.9:ALT(7) | |
| | | P33.11:ALT(7) | |
| CCU61:CC62 | to | IOM:MON1(10) | T12 PWM channel 62 |
| | | IOM:REF1(11) | |
| | | P00.5:ALT(7) | |
| | | P00.9:ALT(7) | |
| | | P20.10:ALT(7) | |
| | | P33.9:ALT(7) | |
| CCU61:CC60INA | from | P00.1:IN | T12 capture input 60 |
| CCU61:CC61INA | from | P00.3:IN | T12 capture input 61 |
| CCU61:CC62INA | from | P00.5:IN | T12 capture input 62 |
| CCU61:CC60INB | from | P02.0:IN | T12 capture input 60 |
| CCU61:CC61INB | from | P02.2:IN | T12 capture input 61 |
| CCU61:CC62INB | from | P02.4:IN | T12 capture input 62 |
| CCU61:CC60INC | from | P00.7:IN | T12 capture input 60 |
| CCU61:CC61INC | from | P00.8:IN | T12 capture input 61 |
| CCU61:CC62INC | from | P00.9:IN | T12 capture input 62 |
| CCU61:CC60IND | from | PMS:pms_wut_underflow | T12 capture input 60 |
| CCU61:CC61IND | from | CAN0:INT(12) | T12 capture input 61 |
| CCU61:CC62IND | from | SCU:E_PDOUT(5) | T12 capture input 62 |
| CCU61:CCPOS0A | from | P00.7:IN | Hall capture input 0 |
| CCU61:CCPOS1A | from | P00.8:IN | Hall capture input 1 |
| CCU61:CCPOS2A | from | P00.9:IN | Hall capture input 2 |
| CCU61:CCPOS0B | from | CCU60:SR(2) | Hall capture input 0 |
| CCU61:CCPOS0C | from | P33.7:IN | Hall capture input 0 |
| CCU61:CCPOS1C | from | P33.6:IN | Hall capture input 1 |

**Table 250 Connections of CCU61** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| CCU61:CCPOS2C | from | P33.5:IN | Hall capture input 2 |
| CCU61:COUT60 | to | SCU:E_REQ1(1) | T12 PWM channel 60 |
| | | IOM:MON1(11) | |
| | | IOM:REF1(10) | |
| | | P00.2:ALT(7) | |
| | | P20.11:ALT(7) | |
| | | P33.12:ALT(7) | |
| CCU61:COUT61 | to | IOM:MON1(12) | T12 PWM channel 61 |
| | | IOM:REF1(9) | |
| | | P00.4:ALT(7) | |
| | | P20.12:ALT(7) | |
| | | P33.10:ALT(7) | |
| CCU61:COUT62 | to | IOM:MON1(13) | T12 PWM channel 62 |
| | | IOM:REF1(8) | |
| | | P00.6:ALT(7) | |
| | | P20.13:ALT(7) | |
| | | P33.8:ALT(7) | |
| CCU61:COUT63 | to | HSPDM:HWRUN(0) | T13 PWM channel 63 |
| | | IOM:MON1(7) | |
| | | IOM:REF1(7) | |
| | | P00.10:ALT(7) | |
| | | P00.12:ALT(7) | |
| | | P20.7:ALT(7) | |
| CCU61:CTRAPA | from | P00.0:IN | Trap input capture |
| CCU61:CTRAPB | from | CCU61:WHE_N | Trap input capture |
| CCU61:CTRAPC | from | P33.4:IN | Trap input capture |
| CCU61:CTRAPD | from | SCU:E_PDOUT(1) | Trap input capture |
| CCU61:SR(0) | to | HSM:EXT_INT(11) | Service request |
| CCU61:SR(1) | to | CCU61:T13HRH | Service request |
| CCU61:SR(2) | to | CCU60:CCPOS0B | Service request |
| | | CCU60:T12HRG | |
| | | CCU60:T13HRG | |
| CCU61:SR(3) | to | EVADC:G1REQTRB | Service request |
| | | EVADC:G0REQTRB | |
| CCU61:T12HRA | from | SCU:scu_cctrig0 | External timer start 12 |
| CCU61:T13HRA | from | SCU:scu_cctrig0 | External timer start 13 |
| CCU61:T12HRB | from | P02.6:IN | External timer start 12 |
| CCU61:T13HRB | from | P02.7:IN | External timer start 13 |

**Table 250 Connections of CCU61** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| CCU61:T12HRC | from | P02.8:IN | External timer start 12 |
| CCU61:T13HRC | from | P02.8:IN | External timer start 13 |
| CCU61:T12HRE | from | P00.11:IN | External timer start 12 |
| CCU61:T13HRE | from | CAN0:INT(15) | External timer start 13 |
| CCU61:T12HRF | from | GPT120:T6OFL | External timer start 12 |
| CCU61:T13HRF | from | GPT120:T6OFL | External timer start 13 |
| CCU61:T12HRG | from | CCU60:SR(2) | External timer start 12 |
| CCU61:T13HRG | from | CCU60:SR(2) | External timer start 13 |
| CCU61:T12HRH | from | SCU:E_PDOUT(1) | External timer start 12 |
| CCU61:T13HRH | from | CCU61:SR(1) | External timer start 13 |
| CCU61:WHE_N | to | CCU61:CTRAPB | Set wrong hall event negative |

## 27.4 Revision History

**Table 251 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V3.0.0** | | |
| | No change | |

# 28 General Purpose Timer Unit (GPT12)

This chapter describes the specific properties of the product TC35x,
which is a member of the product family TC3XX.

The functionality of the GPT12 is described in the TC3XX family documentation. The complete product description consists of the family documentation and this product-specific appendix.

## 28.1 TC35x Specific Register Set

**Table 252 Register Address Space - GPT12**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| GPT120 | $F0001800_H$ | $F00018FF_H$ | FPI slave interface |

### Register Overview Table

See corresponding AURIX™ TC3xx Platform family specification.

## 28.2 TC35x Specific Registers

No deviations from the Family Spec

## 28.3 Connectivity

**Table 253 Connections of GPT120**

| Interface Signals | connects | | Description |
|---|---|---|---|
| GPT120:CAPINB | from | SCU:E_PDOUT(6) | Trigger input to capture value of timer T5 into CAPREL register |
| GPT120:T2EUDA | from | P00.8:IN | Count direction control input of timer T2 |
| GPT120:T3EUDA | from | P02.7:IN | Count direction control input of core timer T3 |
| GPT120:T4EUDA | from | P00.9:IN | Count direction control input of timer T4 |
| GPT120:T5EUDA | from | P21.6:IN | Count direction control input of timer T5 |
| GPT120:T6EUDA | from | P20.0:IN | Count direction control input of core timer T6 |
| GPT120:T2EUDB | from | P33.6:IN | Count direction control input of timer T2 |
| GPT120:T3EUDB | from | P10.7:IN | Count direction control input of core timer T3 |
| GPT120:T4EUDB | from | P33.5:IN | Count direction control input of timer T4 |
| GPT120:T5EUDB | from | P10.1:IN | Count direction control input of timer T5 |
| GPT120:T6EUDB | from | P10.0:IN | Count direction control input of core timer T6 |
| GPT120:T2INA | from | P00.7:IN | Trigger/gate input of timer T2 |
| GPT120:T3INA | from | P02.6:IN | Trigger/gate input of core timer T3 |
| GPT120:T4INA | from | P02.8:IN | Trigger/gate input of timer T4 |
| GPT120:T5INA | from | P21.7:IN | Trigger/gate input of timer T5 |
| GPT120:T6INA | from | P20.3:IN | Trigger/gate input of core timer T6 |
| GPT120:T2INB | from | P33.7:IN | Trigger/gate input of timer T2 |
| GPT120:T3INB | from | P10.4:IN | Trigger/gate input of core timer T3 |
| GPT120:T4INB | from | P10.8:IN | Trigger/gate input of timer T4 |
| GPT120:T5INB | from | P10.3:IN | Trigger/gate input of timer T5 |
| GPT120:T6INB | from | P10.2:IN | Trigger/gate input of core timer T6 |
| GPT120:T3INC | from | SCU:E_PDOUT(4) | Trigger/gate input of core timer T3 |
| GPT120:T6OFL | to | CCU60:T12HRF | Overflow/underflow signal of timer T6 |
| | | CCU60:T13HRF | |
| | | CCU61:T12HRF | |
| | | CCU61:T13HRF | |
| GPT120:T3OUT | to | SCU:E_REQ4(2) | External output for overflow/underflow detection of core timer T3 |
| | | P10.6:ALT(4) | |
| | | P21.6:ALT(7) | |
| GPT120:T6OUT | to | SCU:E_REQ5(2) | External output for overflow/underflow detection of core timer T6 |
| | | P10.5:ALT(5) | |
| | | P21.7:ALT(7) | |
| GPT120:CIRQ_INT | to | INT:gpt120.CIRQ_INT | GPT120 CAPREL Service Request |
| GPT120:T2_INT | to | INT:gpt120.T2_INT | GPT120 T2 Overflow/Underflow Service Request |

**Table 253 Connections of GPT120** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| GPT120:T3_INT | to | INT:gpt120.T3_INT | GPT120 T3 Overflow/Underflow Service Request |
| GPT120:T4_INT | to | INT:gpt120.T4_INT | GPT120 T4 Overflow/Underflow Service Request |
| GPT120:T5_INT | to | INT:gpt120.T5_INT | GPT120 T5 Overflow/Underflow Service Request |
| GPT120:T6_INT | to | INT:gpt120.T6_INT | GPT120 T6 Overflow/Underflow Service Request |

## 28.4 Revision History

**Table 254 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V2.2.3** | | |
| **Table 254** | Revision history updated | |
| **V3.0.0** | | |
| **Page 2** | Connections table changed (no functional change). | |
| **V3.0.1** | | |
| – | No functional changes. | – |
| **V3.0.2** | | |
| | No functional changes. | |

# 29 Converter Control Block (CONVCTRL)

This chapter describes the specific properties of the product TC35x,
which is a member of the product family TC3xx.

The functionality of the CONVCTRL is described in the TC3xx family documentation. The complete product description consists of the family documentation and this product-specific appendix.

## 29.1 TC35x-Specific IP Configuration

The functional description describes the features and operating modes of the converter control block in a general way. This section summarizes the configuration that is available in a specific product.

**Table 255 TC35x specific configuration of CONVERTER**

| Parameter | CONVCTRL |
|---|---|
| **FPI base address** | $F0025000_H$ |
| **FPI address range** | $100_H$ |
| **Application Reset and Kernel Reset** | Application Reset |
| **Name of the config sector value** | CFS Value |
| **CFS value for register VRCFG** | $000000C3_H$ |

## 29.2 TC35x Specific Register Set

**Table 256 Register Address Space - CONVERTER**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| CONVCTRL | $F0025000_H$ | $F00250FF_H$ | FPI slave interface |

### Register Overview Table

See main family chapter.

## 29.3 TC35x Specific Registers

No deviations from the Family Spec

## 29.4 Connectivity

The CONVCTRL is connected to its environment through a number of input and output signals.

**Table 257 Digital Connections for Product TC35x**

| Signal | Dir. | Source/Destin. | Description |
|---|---|---|---|
| **General** | | | |
| PHSYNC | O | EVADC | Synchronization signal for analog clocks |
| CC_ALARM | O | SMU | Alarm signal from safety logic |

**Table 258 List of CONVERTER Interface Signals**

| Interface Signals | I/O | Description |
|---|---|---|
| PHSYNC | out | **Phase synchronization signal** |
| CC_ALARM | out | **Safety Alarm Signal** |

## 29.5 Revision History

**Table 259 Revision History for the Appendix**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V3.0.0** | | |
| – | No change | |
| **V3.0.1** | | |
| Page 2 | EDSADC removed from digital connections table because TC35x has no EDSADC. | – |

# 30 Enhanced Versatile Analog-to-Digital Converter (EVADC)

This chapter describes the specific properties of the product TC35x,
which is a member of the product family TC3XX.

The functionality of the EVADC is described in the TC3XX family documentation. The complete product description consists of the family documentation and this product-specific appendix.

## 30.1 TC35x-Specific IP Configuration

The functional description describes the features and operating modes of the A/D Converters in a general way. This section summarizes the configuration that is available in a specific product.

Each converter group is equipped with a separate analog converter module and a dedicated analog input multiplexer.

**Table 260 General Converter Configuration TC35x**

| Converter Group | Input Channels | Converter Cluster | Common Service Req. Group | Associated Standard Reference Pins |
|---|---|---|---|---|
| **Primary Groups** | | | | |
| G0 | CH0 … CH7 | Primary | C0 | $V_{AREF1}$, $V_{AGND1}$ |
| G1 | CH0 … CH7 | Primary | C1 | $V_{AREF1}$, $V_{AGND1}$ |

### Synchronization Groups

Both converter kernels can be connected to synchronization groups to achieve parallel conversion of several input channels.

**Table 261** summarizes which kernels can be synchronized for parallel conversions.

**Table 261 Synchronization Groups**

| ADC Kernel | Synchr. Group | Master selected by control input CIx[1] | | | |
|---|---|---|---|---|---|
| | | CI0[2] | CI1 | CI2 | CI3 |
| G0 (Prim.) | A | G0 | G1 | --- | --- |
| G1 (Prim.) | A | G1 | G0 | --- | --- |

1) The control input is selected by bitfield STSEL in register GxSYNCTR.
Select the corresponding ready inputs accordingly by bits EVALRx.

2) Control input CI0 always selects the own control signals of the corresponding ADC kernel. This selection is meant for the synchronization master or for stand-alone operation.

### Major Deviations from the Family Documentation

Since the family documentation covers the whole family of TC3XX products, it contains several sections that do not apply to the TC35x. Be aware of the following restrictions:

- No secondary converter groups
- No fast compare channels
- No channel overlays

**Table 262 TC35x specific configuration of EVADC**

| Parameter | EVADC |
|---|---|
| **Number of available primary groups** | 2 |
| **Number of available secondary groups** | 0 |
| **Number of available Fast Compare channels** | 0 |
| **FPI base address** | $F0020000_H$ |
| **FPI address range** | $4000_H$ |

## 30.2 TC35x Specific Register Set

**Table 263 Register Address Space - EVADC**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| EVADC | $F0020000_H$ | $F0023FFF_H$ | FPI slave interface |

**Register Overview Table**

See main family chapter.

## 30.3 Connectivity

The EVADC is connected to its environment through a number of analog input signals and also digital input and output signals. These connections establish communication with other peripherals, with the system blocks, and with external components. The following tables list:

- **Analog Module Connections**
- **Digital Module Connections**

### 30.3.1 Analog Module Connections

The EVADC module accepts a number of analog input signals. The analog input multiplexers select the input channels to be converted from the signals available in this product.

*Note: If an analog input channel is connected to an I/O port pin, make sure the output driver and/or pull devices and/or the digital input path are disabled during normal operation (Px_PDISC.PDISy = 1). For diagnostic functions (MD, PDD) ports must be configured differently (see family description).*

The exact number of analog input channels and the available connection to port pins depend on the employed product type and package (refer to **Table 260** and to the corresponding Data Sheets).

A summary of channels can be found in the table below.

*Note: The analog input pins of the TC35x are not connected to other channels of the EVADC. Therefore, no connections are listed in column "Overlay"*

**Special Markings**

- Input channels marked "PDD" provide a pull-down device for pull-down diagnostics.
- Input channels marked "MD" can activate the pullup and pulldown devices for multiplexer diagnostics.
- Input channels marked "AltRef" can be selected as an alternate reference voltage for conversions on channels of the same group.
- Input channels marked "FixRef" cannot select an alternate reference voltage, but only the corresponding standard reference voltage.

**Table 264 Analog Input Connections for Product TC35x**

| Signal | Source | Overlay | Description |
|---|---|---|---|
| **Reference Inputs** | | | |
| $V_{AREF}$ | VAREF1 | - | positive analog reference |
| $V_{AGND}$ | VAGND1 | - | negative analog reference |
| **Analog Inputs for Group 0 (Primary)** | | | |
| G0CH0 (AltRef) | AN0 | - | analog input channel 0 of group 0 |
| G0CH1 (MD) | AN1 | - | analog input channel 1 of group 0 |
| G0CH2 (MD) | AN2 | - | analog input channel 2 of group 0 |
| G0CH3 | AN3 | - | analog input channel 3 of group 0 |
| G0CH4 (FixRef) | AN4 | - | analog input channel 4 of group 0 |
| G0CH5 (FixRef) | AN5 | - | analog input channel 5 of group 0 |
| G0CH6 (FixRef) | AN6 | - | analog input channel 6 of group 0 |
| G0CH7 (PDD, FixRef) | AN7 | - | analog input channel 7 of group 0 |

**Table 264 Analog Input Connections for Product TC35x** (cont'd)

| Signal | Source | Overlay | Description |
|---|---|---|---|
| **Analog Inputs for Group 1 (Primary)** | | | |
| G1CH0 (AltRef) | AN8 | - | analog input channel 0 of group 1 |
| G1CH1 (MD) | AN9 | - | analog input channel 1 of group 1 |
| G1CH2 (MD) | AN10 | - | analog input channel 2 of group 1 |
| G1CH3 (PDD) | AN11 | - | analog input channel 3 of group 1 |
| G1CH4 | AN12 | - | analog input channel 4 of group 1 |
| G1CH5 | AN13 | - | analog input channel 5 of group 1 |
| G1CH6 | AN14 | - | analog input channel 6 of group 1 |
| G1CH7 | AN15 | - | analog input channel 7 of group 1 |
| **Common Input Signals (x = 0-1)** | | | |
| GxCH28 | $V_{ANACOMM}$ | - | common reference signal, available to all converters,<br>used for production testing,<br>can be fed to the converters through pin AN11 |
| GxCH29 | $V_{MTS}$ | - | module test signal, comparator supply voltage $V_{DDK}$ |
| GxCH30 | $V_{AGND}$ | - | negative reference voltage |
| GxCH31 | $V_{AREF}$ | - | positive reference voltage |

### 30.3.2 Digital Module Connections

The EVADC module accepts a number of digital input signals and generates a number of output signals. This section summarizes the connection of these signals to other on-chip modules or to external resources via port pins.

**Table 265 Digital Connections for Product TC35x**

| Signal | Dir. | Source/Destin. | Description |
|---|---|---|---|
| **Gate Inputs for Primary Groups (x = 0-1, input line selected via bitfield GTSEL = [yyyy$_B$])** | | | |
| GxREQGTA | I | - | [0000$_B$] Gating input A, group x |
| GxREQGTB | I | - | [0001$_B$] Gating input B, group x |
| GxREQGTC | I | CCU6061 TRIG0 | [0010$_B$] CCU6061 trigger output 0 |
| GxREQGTD | I | CCU6061 TRIG1 | [0011$_B$] CCU6061 trigger output 1 |
| GxREQGTE | I | CCU6061 TRIG2 | [0100$_B$] CCU6061 trigger output 2 |
| GxREQGTF | I | - | [0101$_B$] Gating input F, group x |
| GxREQGTG | I | - | [0110$_B$] Gating input G, group x |
| GxREQGTH | I | - | [0111$_B$] Gating input H, group x |
| GxREQGTI | I | - | [1000$_B$] Gating input I, group x |
| GxREQGTJ | I | - | [1001$_B$] Gating input J, group x |
| GxREQGTK | I | - | [1010$_B$] Gating input K, group x |
| GxREQGTL | I | - | [1011$_B$] Gating input L, group x |
| GxREQGTM | I | eru_pdout_x | [1100$_B$] ERU pattern detection output x |
| GxREQGTN | I | - | [1101$_B$] Gating input N, group x |
| GxREQGTO | I | - | [1110$_B$] Gating input O, group x |
| GxREQGTP | I | [internal] | [1111$_B$] Extend inputs to the selected internal trigger source (see GxTRCTR) |
| GxREQGTySEL | O | GxREQTRyP[1)] | Selected gating signal of the respective source |
| **Trigger Inputs for Primary Groups (x = 0-1, input line selected via bitfield XTSEL = [yyyy$_B$])** | | | |
| GxREQTRA | I | CCU60_SR3 | [0000$_B$] CCU60 service request output 3 |
| GxREQTRB | I | CCU61_SR3 | [0001$_B$] CCU61 service request output 3 |
| GxREQTRC | I | HSPDM_adc_trig | [0010$_B$] HSPDM chirp trigger |
| GxREQTRD | I | - | [0011$_B$] Trigger input D, group x |
| GxREQTRE | I | - | [0100$_B$] Trigger input E, group x |
| GxREQTRF | I | - | [0101$_B$] Trigger input F, group x |
| GxREQTRG | I | - | [0110$_B$] Trigger input G, group x |
| GxREQTRH | I | eru_iout_x | [0111$_B$] ERU interrupt output x |
| GxREQTRI | I | - | [1000$_B$] Trigger input I, group x |
| GxREQTRJ | I | - | [1001$_B$] Trigger input J, group x |
| GxREQTRK | I | - | [1010$_B$] Trigger input K, group x |
| GxREQTRL | I | - | [1011$_B$] Trigger input L, group x |
| GxREQTRM | I | vadc_gxsr1 | [1100$_B$] Service request 1, group x |
| GxREQTRN | I | vadc_c0sr1 | [1101$_B$] Service request 1, common group 0 |

**Table 265 Digital Connections for Product TC35x** (cont'd)

| Signal | Dir. | Source/Destin. | Description |
|---|---|---|---|
| GxREQTRO | I | vadc_c1sr1 | [$1110_B$] Service request 1, common group 1 |
| GxREQTRyP | I | GxREQGTySEL[1)] | [$1111_B$] Extend triggers to selected gating input of the respective source |
| GxREQTRySEL | O | - | Selected trigger signal of the respective source |
| **Global Signals and Service Request Lines For Primary Groups: x = 0-1** | | | |
| GxDATA[20:0] | O | RIF | Result values written to RES15 |
| GxWR | O | RIF | Write signal for GxDATA |
| EMUX00 | O | P02.6, P33.3 | Control of external analog multiplexer interface 0 |
| EMUX01 | O | P02.7, P33.2 | |
| EMUX02 | O | P02.8, P33.1 | |
| EMUX10 | O | P00.6, P33.6 | Control of external analog multiplexer interface 1 |
| EMUX11 | O | P00.7, P33.5 | |
| EMUX12 | O | P00.8, P33.4 | |
| GxSR0 | O | ICU | Service request 0 of group x |
| GxSR1 | O | ICU | Service request 1of group x |
| GxSR2 | O | ICU | Service request 2 of group x |
| GxSR3 | O | ICU | Service request 3 of group x |
| C0SR0 | O | ICU | Service request 0 of common block 0 |
| C0SR1 | O | ICU | Service request 1 of common block 0 |
| C0SR2 | O | ICU | Service request 2 of common block 0 |
| C0SR3 | O | ICU | Service request 3 of common block 0 |
| C1SR0 | O | ICU | Service request 0 of common block 1 |
| C1SR1 | O | ICU | Service request 1 of common block 1 |
| C1SR2 | O | ICU | Service request 2 of common block 1 |
| C1SR3 | O | ICU | Service request 3 of common block 1 |
| **System-Internal Connections (x = 0-1)** | | | |
| PHSYNC | I | Phase synchronizer | Synchronization signal for analog clocks |
| otgb0[15:0] | O | OTGM | Alternate trigger buses for additional trace signals indicating the input signal sample phase (see OCS) |
| otgb1[15:0] | O | OTGM | |

1) Internal signal connection.

## 30.4 Revision History

This is a summary of the modifications that have been applied to this chapter.

**Table 266 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V3.0.0** | | |
| Page 3 | Clarify functionality of channel CH29 (see end of table). | |
| **V3.0.1** | | |
| – | No functional changes. | – |
| **V3.0.2** | | |
| – | No functional changes. | – |
| **V3.0.3** | | |
| – | No functional changes. | – |
| **V3.0.4** | | |
| – | No functional changes. | – |
| **V3.0.5** | | |
| – | No functional changes. | – |

# 31 Enhanced Delta-Sigma Analog-to-Digital Converter (EDSADC)

This device doesn’t contain an EDSADC.

# 32 Inter-Integrated Circuit (I2C)

This chapter describes the Inter-Integrated Circuit (short I2C) Module of the TC35x.

## 32.1 TC35x Specific IP Configuration

See features in family spec.

No product specific configuration for I2C

## 32.2 TC35x Specific Register Set

### Register Address Space Table

The address space for the module registers is defined in **Register Address Space Table**.

**Table 267 Register Address Space - I2C**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| I2C0 | $F00C0000_H$ | $F00D00FF_H$ | FPI slave interface |

### Register Overview Table

There are no product specific register for this module.

## 32.3 TC35x Specific Registers

There are no product specific register for this module.

## 32.4 Connectivity

The tables below list all the connections of I2C instances.

**Table 268 Connections of I2C0**

| Interface Signals | connects | | Description |
|---|---|---|---|
| I2C0:SCL | to | P02.5:ALT(5) | Serial Clock Output |
| | | P15.4:ALT(6) | |
| I2C0:SDA | to | P02.4:ALT(5) | Serial Data Output |
| | | P15.5:ALT(6) | |
| I2C0:SDAA | from | P02.4:IN | Serial Data Input 0 |
| I2C0:SDAC | from | P15.5:IN | Serial Data Input 2 |
| I2C0:SLEEP | from | SCU:scu_syst_sleep_n | Sleep Request |
| I2C0:DTR_INT | to | INT:i2c0.DTR_INT | I2C Data Transfer Request |
| I2C0:ERR_INT | to | INT:i2c0.ERR_INT | I2C Error Service Request |
| I2C0:P_INT | to | INT:i2c0.P_INT | I2C Kernel Service Request |

## 32.5 Revision History

**Table 269 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V2.3.4** | | |
| **Page 1** | No functional changes.<br>Formal changes in Connectivity tables. | |
| **V2.3.5** | | |
| – | No functional changes. | |
| **V2.3.6** | | |
| – | No functional changes. | |

# 33 High Speed Serial Link (HSSL)

This device doesn’t contain a HSSL.

# 34 Asynchronous Serial Interface (ASCLIN)

Text with reference to family spec.

## 34.1 TC35x Specific IP Configuration

No product specific configuration for ASCLIN

## 34.2 TC35x Specific Register Set

### Register Address Space Table

**Table 270 Register Address Space - ASCLIN**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| ASCLIN0 | $F0000600_H$ | $F00006FF_H$ | FPI slave interface |
| ASCLIN1 | $F0000700_H$ | $F00007FF_H$ | FPI slave interface |
| ASCLIN2 | $F0000800_H$ | $F00008FF_H$ | FPI slave interface |
| ASCLIN3 | $F0000900_H$ | $F00009FF_H$ | FPI slave interface |

### Register Overview Table

**Table 271 Register Overview - ASCLIN (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ASCLIN0_CLC | Clock Control Register | $000_H$ | See Family Spec |
| ASCLIN1_CLC | Clock Control Register | $000_H$ | See Family Spec |
| ASCLIN2_CLC | Clock Control Register | $000_H$ | See Family Spec |
| ASCLIN3_CLC | Clock Control Register | $000_H$ | See Family Spec |
| ASCLIN0_IOCR | Input and Output Control Register | $004_H$ | See Family Spec |
| ASCLIN1_IOCR | Input and Output Control Register | $004_H$ | See Family Spec |
| ASCLIN2_IOCR | Input and Output Control Register | $004_H$ | See Family Spec |
| ASCLIN3_IOCR | Input and Output Control Register | $004_H$ | See Family Spec |
| ASCLIN0_ID | Module Identification Register | $008_H$ | See Family Spec |

**Table 271 Register Overview - ASCLIN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ASCLIN1_ID | Module Identification Register | $008_H$ | See Family Spec |
| ASCLIN2_ID | Module Identification Register | $008_H$ | See Family Spec |
| ASCLIN3_ID | Module Identification Register | $008_H$ | See Family Spec |
| ASCLIN0_TXFIFOCON | TX FIFO Configuration Register | $00C_H$ | See Family Spec |
| ASCLIN1_TXFIFOCON | TX FIFO Configuration Register | $00C_H$ | See Family Spec |
| ASCLIN2_TXFIFOCON | TX FIFO Configuration Register | $00C_H$ | See Family Spec |
| ASCLIN3_TXFIFOCON | TX FIFO Configuration Register | $00C_H$ | See Family Spec |
| ASCLIN0_RXFIFOCON | RX FIFO Configuration Register | $010_H$ | See Family Spec |
| ASCLIN1_RXFIFOCON | RX FIFO Configuration Register | $010_H$ | See Family Spec |
| ASCLIN2_RXFIFOCON | RX FIFO Configuration Register | $010_H$ | See Family Spec |
| ASCLIN3_RXFIFOCON | RX FIFO Configuration Register | $010_H$ | See Family Spec |
| ASCLIN0_BITCON | Bit Configuration Register | $014_H$ | See Family Spec |
| ASCLIN1_BITCON | Bit Configuration Register | $014_H$ | See Family Spec |
| ASCLIN2_BITCON | Bit Configuration Register | $014_H$ | See Family Spec |

**Table 271 Register Overview - ASCLIN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ASCLIN3_BITCON | Bit Configuration Register | $014_H$ | See Family Spec |
| ASCLIN0_FRAMECON | Frame Control Register | $018_H$ | See Family Spec |
| ASCLIN1_FRAMECON | Frame Control Register | $018_H$ | See Family Spec |
| ASCLIN2_FRAMECON | Frame Control Register | $018_H$ | See Family Spec |
| ASCLIN3_FRAMECON | Frame Control Register | $018_H$ | See Family Spec |
| ASCLIN0_DATCON | Data Configuration Register | $01C_H$ | See Family Spec |
| ASCLIN1_DATCON | Data Configuration Register | $01C_H$ | See Family Spec |
| ASCLIN2_DATCON | Data Configuration Register | $01C_H$ | See Family Spec |
| ASCLIN3_DATCON | Data Configuration Register | $01C_H$ | See Family Spec |
| ASCLIN0_BRG | Baud Rate Generation Register | $020_H$ | See Family Spec |
| ASCLIN1_BRG | Baud Rate Generation Register | $020_H$ | See Family Spec |
| ASCLIN2_BRG | Baud Rate Generation Register | $020_H$ | See Family Spec |
| ASCLIN3_BRG | Baud Rate Generation Register | $020_H$ | See Family Spec |
| ASCLIN0_BRD | Baud Rate Detection Register | $024_H$ | See Family Spec |

**Table 271 Register Overview - ASCLIN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ASCLIN1_BRD | Baud Rate Detection Register | $024_H$ | See Family Spec |
| ASCLIN2_BRD | Baud Rate Detection Register | $024_H$ | See Family Spec |
| ASCLIN3_BRD | Baud Rate Detection Register | $024_H$ | See Family Spec |
| ASCLIN0_LINCON | LIN Control Register | $028_H$ | See Family Spec |
| ASCLIN1_LINCON | LIN Control Register | $028_H$ | See Family Spec |
| ASCLIN2_LINCON | LIN Control Register | $028_H$ | See Family Spec |
| ASCLIN3_LINCON | LIN Control Register | $028_H$ | See Family Spec |
| ASCLIN0_LINBTIMER | LIN Break Timer Register | $02C_H$ | See Family Spec |
| ASCLIN1_LINBTIMER | LIN Break Timer Register | $02C_H$ | See Family Spec |
| ASCLIN2_LINBTIMER | LIN Break Timer Register | $02C_H$ | See Family Spec |
| ASCLIN3_LINBTIMER | LIN Break Timer Register | $02C_H$ | See Family Spec |
| ASCLIN0_LINHTIMER | LIN Header Timer Register | $030_H$ | See Family Spec |
| ASCLIN1_LINHTIMER | LIN Header Timer Register | $030_H$ | See Family Spec |
| ASCLIN2_LINHTIMER | LIN Header Timer Register | $030_H$ | See Family Spec |

**Table 271 Register Overview - ASCLIN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ASCLIN3_LINHTIMER | LIN Header Timer Register | $030_H$ | See Family Spec |
| ASCLIN0_FLAGS | Flags Register | $034_H$ | See Family Spec |
| ASCLIN1_FLAGS | Flags Register | $034_H$ | See Family Spec |
| ASCLIN2_FLAGS | Flags Register | $034_H$ | See Family Spec |
| ASCLIN3_FLAGS | Flags Register | $034_H$ | See Family Spec |
| ASCLIN0_FLAGSSET | Flags Set Register | $038_H$ | See Family Spec |
| ASCLIN1_FLAGSSET | Flags Set Register | $038_H$ | See Family Spec |
| ASCLIN2_FLAGSSET | Flags Set Register | $038_H$ | See Family Spec |
| ASCLIN3_FLAGSSET | Flags Set Register | $038_H$ | See Family Spec |
| ASCLIN0_FLAGSCLEAR | Flags Clear Register | $03C_H$ | See Family Spec |
| ASCLIN1_FLAGSCLEAR | Flags Clear Register | $03C_H$ | See Family Spec |
| ASCLIN2_FLAGSCLEAR | Flags Clear Register | $03C_H$ | See Family Spec |
| ASCLIN3_FLAGSCLEAR | Flags Clear Register | $03C_H$ | See Family Spec |
| ASCLIN0_FLAGSENABLE | Flags Enable Register | $040_H$ | See Family Spec |

**Table 271 Register Overview - ASCLIN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ASCLIN1_FLAGSENABLE | Flags Enable Register | $040_H$ | See Family Spec |
| ASCLIN2_FLAGSENABLE | Flags Enable Register | $040_H$ | See Family Spec |
| ASCLIN3_FLAGSENABLE | Flags Enable Register | $040_H$ | See Family Spec |
| ASCLIN0_TXDATA | Transmit Data Register | $044_H$ | See Family Spec |
| ASCLIN1_TXDATA | Transmit Data Register | $044_H$ | See Family Spec |
| ASCLIN2_TXDATA | Transmit Data Register | $044_H$ | See Family Spec |
| ASCLIN3_TXDATA | Transmit Data Register | $044_H$ | See Family Spec |
| ASCLIN0_RXDATA | Receive Data Register | $048_H$ | See Family Spec |
| ASCLIN1_RXDATA | Receive Data Register | $048_H$ | See Family Spec |
| ASCLIN2_RXDATA | Receive Data Register | $048_H$ | See Family Spec |
| ASCLIN3_RXDATA | Receive Data Register | $048_H$ | See Family Spec |
| ASCLIN0_CSR | Clock Selection Register | $04C_H$ | See Family Spec |
| ASCLIN1_CSR | Clock Selection Register | $04C_H$ | See Family Spec |
| ASCLIN2_CSR | Clock Selection Register | $04C_H$ | See Family Spec |

**Table 271 Register Overview - ASCLIN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ASCLIN3_CSR | Clock Selection Register | $04C_H$ | See Family Spec |
| ASCLIN0_RXDATAD | Receive Data Debug Register | $050_H$ | See Family Spec |
| ASCLIN1_RXDATAD | Receive Data Debug Register | $050_H$ | See Family Spec |
| ASCLIN2_RXDATAD | Receive Data Debug Register | $050_H$ | See Family Spec |
| ASCLIN3_RXDATAD | Receive Data Debug Register | $050_H$ | See Family Spec |
| ASCLIN0_OCS | OCDS Control and Status | $0E8_H$ | See Family Spec |
| ASCLIN1_OCS | OCDS Control and Status | $0E8_H$ | See Family Spec |
| ASCLIN2_OCS | OCDS Control and Status | $0E8_H$ | See Family Spec |
| ASCLIN3_OCS | OCDS Control and Status | $0E8_H$ | See Family Spec |
| ASCLIN0_KRSTCLR | Kernel Reset Status Clear Register | $0EC_H$ | See Family Spec |
| ASCLIN1_KRSTCLR | Kernel Reset Status Clear Register | $0EC_H$ | See Family Spec |
| ASCLIN2_KRSTCLR | Kernel Reset Status Clear Register | $0EC_H$ | See Family Spec |
| ASCLIN3_KRSTCLR | Kernel Reset Status Clear Register | $0EC_H$ | See Family Spec |
| ASCLIN0_KRST1 | Kernel Reset Register 1 | $0F0_H$ | See Family Spec |

**Table 271 Register Overview - ASCLIN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ASCLIN1_KRST1 | Kernel Reset Register 1 | $0F0_H$ | See Family Spec |
| ASCLIN2_KRST1 | Kernel Reset Register 1 | $0F0_H$ | See Family Spec |
| ASCLIN3_KRST1 | Kernel Reset Register 1 | $0F0_H$ | See Family Spec |
| ASCLIN0_KRST0 | Kernel Reset Register 0 | $0F4_H$ | See Family Spec |
| ASCLIN1_KRST0 | Kernel Reset Register 0 | $0F4_H$ | See Family Spec |
| ASCLIN2_KRST0 | Kernel Reset Register 0 | $0F4_H$ | See Family Spec |
| ASCLIN3_KRST0 | Kernel Reset Register 0 | $0F4_H$ | See Family Spec |
| ASCLIN0_ACCEN1 | Access Enable Register 1 | $0F8_H$ | See Family Spec |
| ASCLIN1_ACCEN1 | Access Enable Register 1 | $0F8_H$ | See Family Spec |
| ASCLIN2_ACCEN1 | Access Enable Register 1 | $0F8_H$ | See Family Spec |
| ASCLIN3_ACCEN1 | Access Enable Register 1 | $0F8_H$ | See Family Spec |
| ASCLIN0_ACCEN0 | Access Enable Register 0 | $0FC_H$ | See Family Spec |
| ASCLIN1_ACCEN0 | Access Enable Register 0 | $0FC_H$ | See Family Spec |

**Table 271 Register Overview - ASCLIN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ASCLIN2_ACCEN0 | Access Enable Register 0 | $0FC_H$ | See Family Spec |
| ASCLIN3_ACCEN0 | Access Enable Register 0 | $0FC_H$ | See Family Spec |

## 34.3 TC35x Specific Registers

No deviations from the Family Spec

## 34.4 Connectivity

**Table 272 Connections of ASCLIN0**

| Interface Signals | connects | | Description |
|---|---|---|---|
| ASCLIN0:ACTSA | from | P14.9:IN | Clear to send input |
| ASCLIN0:ACTSD | from | ASCLIN0:ARTS | Clear to send input |
| ASCLIN0:ARTS | to | P14.7:ALT(2) | Ready to send output |
| | | ASCLIN0:ACTSD | |
| ASCLIN0:ARXA | from | P14.1:IN | Receive input |
| ASCLIN0:ARXB | from | P15.3:IN | Receive input |
| ASCLIN0:ARXD | from | P33.10:IN | Receive input |
| ASCLIN0:ASCLK | to | P14.0:ALT(6) | Shift clock output |
| | | P15.2:ALT(6) | |
| ASCLIN0:ATX | to | IOM:MON2(12) | Transmit output |
| | | IOM:REF2(12) | |
| | | P14.0:ALT(2) | |
| | | P14.1:ALT(2) | |
| | | P15.2:ALT(2) | |
| | | P15.3:ALT(2) | |
| | | P33.9:ALT(6) | |
| ASCLIN0:sleep_n | from | SCU:scu_syst_sleep_n | Negative turn-off request |
| ASCLIN0:TX_INT | to | INT:asclin0.TX_INT | ASCLIN Transmit Service Request |
| ASCLIN0:RX_INT | to | INT:asclin0.RX_INT | ASCLIN Receive Service Request |
| ASCLIN0:ERR_INT | to | INT:asclin0.ERR_INT | ASCLIN Error Service Request |

**Table 273 Connections of ASCLIN1**

| Interface Signals | connects | | Description |
|---|---|---|---|
| ASCLIN1:ACTSA | from | P20.7:IN | Clear to send input |
| ASCLIN1:ACTSB | from | P32.4:IN | Clear to send input |
| ASCLIN1:ACTSD | from | ASCLIN1:ARTS | Clear to send input |
| ASCLIN1:ARTS | to | P20.6:ALT(2) | Ready to send output |
| | | P23.1:ALT(2) | |
| | | ASCLIN1:ACTSD | |
| ASCLIN1:ARXA | from | P15.1:IN | Receive input |
| ASCLIN1:ARXB | from | P15.5:IN | Receive input |
| ASCLIN1:ARXC | from | P20.9:IN | Receive input |
| ASCLIN1:ARXD | from | P14.8:IN | Receive input |
| ASCLIN1:ARXE | from | P11.10:IN | Receive input |
| ASCLIN1:ARXF | from | P33.13:IN | Receive input |
| ASCLIN1:ARXG | from | P02.3:IN | Receive input |
| ASCLIN1:ASCLK | to | P15.0:ALT(6) | Shift clock output |
| | | P20.10:ALT(6) | |
| | | P33.11:ALT(2) | |
| | | P33.12:ALT(4) | |
| ASCLIN1:ASLSO | to | P14.3:ALT(4) | Slave select signal output |
| | | P20.8:ALT(2) | |
| | | P33.10:ALT(4) | |
| ASCLIN1:ATX | to | IOM:MON2(13) | Transmit output |
| | | IOM:REF2(13) | |
| | | P02.2:ALT(2) | |
| | | P11.12:ALT(2) | |
| | | P14.10:ALT(4) | |
| | | P15.0:ALT(2) | |
| | | P15.1:ALT(2) | |
| | | P15.4:ALT(2) | |
| | | P15.5:ALT(2) | |
| | | P20.10:ALT(2) | |
| | | P33.12:ALT(2) | |
| | | P33.13:ALT(2) | |
| ASCLIN1:sleep_n | from | SCU:scu_syst_sleep_n | Negative turn-off request |
| ASCLIN1:TX_INT | to | INT:asclin1.TX_INT | ASCLIN Transmit Service Request |
| ASCLIN1:RX_INT | to | INT:asclin1.RX_INT | ASCLIN Receive Service Request |
| ASCLIN1:ERR_INT | to | INT:asclin1.ERR_INT | ASCLIN Error Service Request |

**Table 274 Connections of ASCLIN2**

| Interface Signals | connects | | Description |
|---|---|---|---|
| ASCLIN2:ACTSA | from | P10.7:IN | Clear to send input |
| ASCLIN2:ACTSB | from | P33.5:IN | Clear to send input |
| ASCLIN2:ACTSD | from | ASCLIN2:ARTS | Clear to send input |
| ASCLIN2:ARTS | to | P10.8:ALT(2) | Ready to send output |
| | | P33.4:ALT(2) | |
| | | ASCLIN2:ACTSD | |
| ASCLIN2:ARXA | from | P14.3:IN | Receive input |
| ASCLIN2:ARXB | from | P02.1:IN | Receive input |
| ASCLIN2:ARXD | from | P10.6:IN | Receive input |
| ASCLIN2:ARXE | from | P33.8:IN | Receive input |
| ASCLIN2:ARXF | from | P32.6:IN | Receive input |
| ASCLIN2:ARXG | from | P02.0:IN | Receive input |
| ASCLIN2:ASCLK | to | P02.4:ALT(2) | Shift clock output |
| | | P10.6:ALT(2) | |
| | | P14.2:ALT(6) | |
| | | P33.7:ALT(2) | |
| | | P33.9:ALT(4) | |
| ASCLIN2:ASLSO | to | P02.3:ALT(2) | Slave select signal output |
| | | P10.5:ALT(6) | |
| | | P33.6:ALT(2) | |
| ASCLIN2:ATX | to | IOM:MON2(14) | Transmit output |
| | | IOM:REF2(14) | |
| | | P02.0:ALT(2) | |
| | | P10.5:ALT(2) | |
| | | P14.2:ALT(2) | |
| | | P14.3:ALT(2) | |
| | | P32.5:ALT(2) | |
| | | P33.8:ALT(2) | |
| | | P33.9:ALT(2) | |
| ASCLIN2:sleep_n | from | SCU:scu_syst_sleep_n | Negative turn-off request |
| ASCLIN2:TX_INT | to | INT:asclin2.TX_INT | ASCLIN Transmit Service Request |
| ASCLIN2:RX_INT | to | INT:asclin2.RX_INT | ASCLIN Receive Service Request |
| ASCLIN2:ERR_INT | to | INT:asclin2.ERR_INT | ASCLIN Error Service Request |

**Table 275 Connections of ASCLIN3**

| Interface Signals | connects | | Description |
|---|---|---|---|
| ASCLIN3:ACTSA | from | P00.12:IN | Clear to send input |
| ASCLIN3:ACTSD | from | ASCLIN3:ARTS | Clear to send input |
| ASCLIN3:ARTS | to | P00.9:ALT(3) | Ready to send output |
| | | ASCLIN3:ACTSD | |
| ASCLIN3:ARXA | from | P15.7:IN | Receive input |
| ASCLIN3:ARXB | from | P11.0:IN | Receive input |
| ASCLIN3:ARXC | from | P20.3:IN | Receive input |
| ASCLIN3:ARXD | from | P32.2:IN | Receive input |
| ASCLIN3:ARXE | from | P00.1:IN | Receive input |
| ASCLIN3:ARXF | from | P21.6:IN | Receive input |
| ASCLIN3:ASCLK | to | P00.0:ALT(2) | Shift clock output |
| | | P00.2:ALT(2) | |
| | | P11.1:ALT(2) | |
| | | P11.4:ALT(2) | |
| | | P15.6:ALT(6) | |
| | | P15.8:ALT(6) | |
| | | P20.0:ALT(3) | |
| | | P21.5:ALT(2) | |
| | | P21.7:ALT(3) | |
| | | P32.3:ALT(4) | |
| | | P33.2:ALT(2) | |
| ASCLIN3:ASLSO | to | P00.3:ALT(2) | Slave select signal output |
| | | P12.1:ALT(2) | |
| | | P14.3:ALT(5) | |
| | | P21.2:ALT(2) | |
| | | P21.6:ALT(2) | |
| | | P33.1:ALT(2) | |

**Table 275 Connections of ASCLIN3** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| ASCLIN3:ATX | to | IOM:MON2(15) | Transmit output |
| | | IOM:REF2(15) | |
| | | P00.0:ALT(3) | |
| | | P00.1:ALT(2) | |
| | | P11.0:ALT(2) | |
| | | P11.1:ALT(3) | |
| | | P15.6:ALT(2) | |
| | | P15.7:ALT(2) | |
| | | P20.0:ALT(2) | |
| | | P20.3:ALT(2) | |
| | | P21.7:ALT(2) | |
| | | P32.2:ALT(2) | |
| | | P32.3:ALT(2) | |
| ASCLIN3:sleep_n | from | SCU:scu_syst_sleep_n | Negative turn-off request |
| ASCLIN3:TX_INT | to | INT:asclin3.TX_INT | ASCLIN Transmit Service Request |
| ASCLIN3:RX_INT | to | INT:asclin3.RX_INT | ASCLIN Receive Service Request |
| ASCLIN3:ERR_INT | to | INT:asclin3.ERR_INT | ASCLIN Error Service Request |

## 34.5 Revision History

**Table 276 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V3.2.6** | | |
| Page 2 | Register tables updated. | |
| | No functional change in connectivity tables. | |
| **V3.2.7** | | |
| – | No functional changes. | |
| **V3.2.8** | | |
| – | No functional changes. | |

# 35 Queued Synchronous Peripheral Interface (QSPI)

## 35.1 TC35x Specific IP Configuration

**Table 277 TC35x specific configuration of QSPI**

| Parameter | QSPI0 | QSPI1 | QSPI2 | QSPI3 |
|---|---|---|---|---|
| QSPI module has HSIC | | | x | x |

## 35.2 TC35x Specific Register Set

### Register Address Space Table

**Table 278 Register Address Space - QSPI**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| QSPI0 | $F0001C00_H$ | $F0001CFF_H$ | Register block QSPI0 |
| QSPI1 | $F0001D00_H$ | $F0001DFF_H$ | Register block QSPI1 |
| QSPI2 | $F0001E00_H$ | $F0001EFF_H$ | Register block QSPI2 |
| QSPI3 | $F0001F00_H$ | $F0001FFF_H$ | Register block QSPI3 |

### Register Overview Tables of QSPI

**Table 279 Register Overview - QSPI0 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| QSPI0_CLC | Clock Control Register | $000_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| QSPI0_PISEL | Port Input Select Register | $004_H$ | U,SV | SV,P | Application Reset | 10 |
| QSPI0_ID | Module Identification Register | $008_H$ | U,SV | BE | Application Reset | See Family Spec |
| QSPI0_GLOBALCON | Global Configuration Register | $010_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| QSPI0_GLOBALCON1 | Global Configuration Register 1 | $014_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| QSPI0_BACON | Basic Configuration Register | $018_H$ | U,SV | BE | Application Reset | See Family Spec |
| QSPI0_ECONz (z=0-7) | Configuration Extension z | $020_H$+z*4 | U,SV | SV,P | Application Reset | See Family Spec |
| QSPI0_STATUS | Status Register | $040_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI0_STATUS1 | Status Register 1 | $044_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 279 Register Overview - QSPI0 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| QSPI0_SSOC | Slave Select Output Control Register | $048_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| QSPI0_FLAGSCLEAR | Flags Clear Register | $054_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI0_XXLCON | Extra Large Data Configuration Register | $058_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI0_MIXENTRY | MIX_ENTRY Register | $05C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI0_BACONENTRY | BACON_ENTRY Register | $060_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI0_DATAENTRYx (x=0-7) | DATA_ENTRY Register x | $064_H$+x* 4 | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI0_RXEXIT | RX_EXIT Register | $090_H$ | U,SV | BE | Application Reset | See Family Spec |
| QSPI0_RXEXITD | RX_EXIT Debug Register | $094_H$ | U,SV | BE | Application Reset | See Family Spec |
| QSPI0_MC | Move Counter Register | $0A4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI0_MCCON | Move Counter control Register | $0A8_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI0_OCS | OCDS Control and Status | $0E8_H$ | U,SV | SV,P | Debug Reset | See Family Spec |
| QSPI0_KRSTCLR | Kernel Reset Status Clear Register | $0EC_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| QSPI0_KRST1 | Kernel Reset Register 1 | $0F0_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| QSPI0_KRST0 | Kernel Reset Register 0 | $0F4_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |

**Table 279 Register Overview - QSPI0 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| QSPI0_ACCEN1 | Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| QSPI0_ACCEN0 | Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | See Family Spec |

**Table 280 Register Overview - QSPI1 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| QSPI1_CLC | Clock Control Register | $000_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| QSPI1_PISEL | Port Input Select Register | $004_H$ | U,SV | SV,P | Application Reset | 11 |
| QSPI1_ID | Module Identification Register | $008_H$ | U,SV | BE | Application Reset | See Family Spec |
| QSPI1_GLOBALCON | Global Configuration Register | $010_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| QSPI1_GLOBALCON1 | Global Configuration Register 1 | $014_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| QSPI1_BACON | Basic Configuration Register | $018_H$ | U,SV | BE | Application Reset | See Family Spec |
| QSPI1_ECONz (z=0-7) | Configuration Extension z | $020_H$+z*4 | U,SV | SV,P | Application Reset | See Family Spec |
| QSPI1_STATUS | Status Register | $040_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI1_STATUS1 | Status Register 1 | $044_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI1_SSOC | Slave Select Output Control Register | $048_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| QSPI1_FLAGSCLEAR | Flags Clear Register | $054_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 280 Register Overview - QSPI1 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| QSPI1_XXLCON | Extra Large Data Configuration Register | $058_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI1_MIXENTRY | MIX_ENTRY Register | $05C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI1_BACONENTRY | BACON_ENTRY Register | $060_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI1_DATAENTRYx (x=0-7) | DATA_ENTRY Register x | $064_H$+x* 4 | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI1_RXEXIT | RX_EXIT Register | $090_H$ | U,SV | BE | Application Reset | See Family Spec |
| QSPI1_RXEXITD | RX_EXIT Debug Register | $094_H$ | U,SV | BE | Application Reset | See Family Spec |
| QSPI1_MC | Move Counter Register | $0A4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI1_MCCON | Move Counter control Register | $0A8_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI1_OCS | OCDS Control and Status | $0E8_H$ | U,SV | SV,P | Debug Reset | See Family Spec |
| QSPI1_KRSTCLR | Kernel Reset Status Clear Register | $0EC_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| QSPI1_KRST1 | Kernel Reset Register 1 | $0F0_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| QSPI1_KRST0 | Kernel Reset Register 0 | $0F4_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| QSPI1_ACCEN1 | Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| QSPI1_ACCEN0 | Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | See Family Spec |

**Table 281 Register Overview - QSPI2 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| QSPI2_CLC | Clock Control Register | $000_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| QSPI2_PISEL | Port Input Select Register | $004_H$ | U,SV | SV,P | Application Reset | 12 |
| QSPI2_ID | Module Identification Register | $008_H$ | U,SV | BE | Application Reset | See Family Spec |
| QSPI2_GLOBALCON | Global Configuration Register | $010_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| QSPI2_GLOBALCON1 | Global Configuration Register 1 | $014_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| QSPI2_BACON | Basic Configuration Register | $018_H$ | U,SV | BE | Application Reset | See Family Spec |
| QSPI2_ECONz (z=0-7) | Configuration Extension z | $020_H$+z*4 | U,SV | SV,P | Application Reset | See Family Spec |
| QSPI2_STATUS | Status Register | $040_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI2_STATUS1 | Status Register 1 | $044_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI2_SSOC | Slave Select Output Control Register | $048_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| QSPI2_FLAGSCLEAR | Flags Clear Register | $054_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI2_XXLCON | Extra Large Data Configuration Register | $058_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI2_MIXENTRY | MIX_ENTRY Register | $05C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI2_BACONENTRY | BACON_ENTRY Register | $060_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 281 Register Overview - QSPI2 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| QSPI2_DATAENTRYx (x=0-7) | DATA_ENTRY Register x | $064_H$+x*4 | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI2_RXEXIT | RX_EXIT Register | $090_H$ | U,SV | BE | Application Reset | See Family Spec |
| QSPI2_RXEXITD | RX_EXIT Debug Register | $094_H$ | U,SV | BE | Application Reset | See Family Spec |
| QSPI2_CAPCON | Capture Control Register | $0A0_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI2_MC | Move Counter Register | $0A4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI2_MCCON | Move Counter control Register | $0A8_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI2_OCS | OCDS Control and Status | $0E8_H$ | U,SV | SV,P | Debug Reset | See Family Spec |
| QSPI2_KRSTCLR | Kernel Reset Status Clear Register | $0EC_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| QSPI2_KRST1 | Kernel Reset Register 1 | $0F0_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| QSPI2_KRST0 | Kernel Reset Register 0 | $0F4_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| QSPI2_ACCEN1 | Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| QSPI2_ACCEN0 | Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | See Family Spec |

**Table 282 Register Overview - QSPI3 (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| QSPI3_CLC | Clock Control Register | $000_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| QSPI3_PISEL | Port Input Select Register | $004_H$ | U,SV | SV,P | Application Reset | 13 |
| QSPI3_ID | Module Identification Register | $008_H$ | U,SV | BE | Application Reset | See Family Spec |
| QSPI3_GLOBALCON | Global Configuration Register | $010_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| QSPI3_GLOBALCON1 | Global Configuration Register 1 | $014_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| QSPI3_BACON | Basic Configuration Register | $018_H$ | U,SV | BE | Application Reset | See Family Spec |
| QSPI3_ECONz (z=0-7) | Configuration Extension z | $020_H$+z*4 | U,SV | SV,P | Application Reset | See Family Spec |
| QSPI3_STATUS | Status Register | $040_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI3_STATUS1 | Status Register 1 | $044_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI3_SSOC | Slave Select Output Control Register | $048_H$ | U,SV | SV,P | Application Reset | See Family Spec |
| QSPI3_FLAGSCLEAR | Flags Clear Register | $054_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI3_XXLCON | Extra Large Data Configuration Register | $058_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI3_MIXENTRY | MIX_ENTRY Register | $05C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI3_BACONENTRY | BACON_ENTRY Register | $060_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 282 Register Overview - QSPI3 (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| QSPI3_DATAENTRYx (x=0-7) | DATA_ENTRY Register x | $064_H$+x*4 | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI3_RXEXIT | RX_EXIT Register | $090_H$ | U,SV | BE | Application Reset | See Family Spec |
| QSPI3_RXEXITD | RX_EXIT Debug Register | $094_H$ | U,SV | BE | Application Reset | See Family Spec |
| QSPI3_CAPCON | Capture Control Register | $0A0_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI3_MC | Move Counter Register | $0A4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI3_MCCON | Move Counter control Register | $0A8_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| QSPI3_OCS | OCDS Control and Status | $0E8_H$ | U,SV | SV,P | Debug Reset | See Family Spec |
| QSPI3_KRSTCLR | Kernel Reset Status Clear Register | $0EC_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| QSPI3_KRST1 | Kernel Reset Register 1 | $0F0_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| QSPI3_KRST0 | Kernel Reset Register 0 | $0F4_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| QSPI3_ACCEN1 | Access Enable Register 1 | $0F8_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| QSPI3_ACCEN0 | Access Enable Register 0 | $0FC_H$ | U,SV | SV,SE | Application Reset | See Family Spec |

## 35.3 TC35x Specific Registers

### 35.3.1 Register block QSPI

#### Port Input Select Register

The PISEL register controls the input signal selection of the SSC module.

**QSPI0_PISEL**

**Port Input Select Register** **($004_H$)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | SLSIS | | | 0 | SCIS | | | 0 | SRIS | | | 0 | MRIS | | |
| r | rw | | | r | rw | | | r | rw | | | r | rw | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **MRIS** | 2:0 | rw | **Master Mode Receive Input Select**<br>MRIS selects one out of eight MRST receive input lines, used in Master Mode. Note that not all inputs are used in every device of the family. Selecting an unused input returns a continuous low value.<br>The following signal sources are available in this product (if supported by the package!)<br>$000_B$ **P20.12_IN**, |
| **SRIS** | 6:4 | rw | **Slave Mode Receive Input Select**<br>SRIS selects one out of eight MTSR receive input lines, used in Slave Mode. Note that not all inputs are used in every device of the family. Selecting an unused input returns a continuous low value.<br>The following signal sources are available in this product (if supported by the package!)<br>$000_B$ **P20.14_IN**,<br>$010_B$ **P22.5_IN**, |
| **SCIS** | 10:8 | rw | **Slave Mode Clock Input Select**<br>SCIS selects one out of eight module kernel SCLK input lines that is used as clock input line in slave mode. Note that not all inputs are used in every device of the family. Selecting an unused input returns a continuous low value.<br>The following signal sources are available in this product (if supported by the package!)<br>$000_B$ **P20.11_IN**, |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SLSIS** | 14:12 | rw | **Slave Mode Slave Select Input Selection**<br>The SLSIS must be programmed properly before the slave mode is set with GLOBALCON.MODE and the module is set to RUN mode.<br>The following signal sources are available in this product (if supported by the package!)<br>$000_B$ no input<br>$001_B$ **P20.13_IN**,<br>$010_B$ **P20.9_IN**, |
| **0** | 3,<br>7,<br>11,<br>31:15 | r | **Reserved**<br>Read as 0; should be written with 0. |

**QSPI1_PISEL**

**Port Input Select Register** **($004_H$)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **SLSIS** | | | **0** | **SCIS** | | | **0** | **SRIS** | | | **0** | **MRIS** | | |
| r | rw | | | r | rw | | | r | rw | | | r | rw | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **MRIS** | 2:0 | rw | **Master Mode Receive Input Select**<br>MRIS selects one out of eight MRST receive input lines, used in Master Mode. Note that not all inputs are used in every device of the family. Selecting an unused input returns a continuous low value.<br>The following signal sources are available in this product (if supported by the package!)<br>$000_B$ **P10.1_IN**,<br>$001_B$ **P11.3_IN**, |
| **SRIS** | 6:4 | rw | **Slave Mode Receive Input Select**<br>SRIS selects one out of eight MTSR receive input lines, used in Slave Mode. Note that not all inputs are used in every device of the family. Selecting an unused input returns a continuous low value.<br>The following signal sources are available in this product (if supported by the package!)<br>$000_B$ **P10.3_IN**,<br>$001_B$ **P11.9_IN**,<br>$010_B$ **P10.4_IN**, |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SCIS** | 10:8 | rw | **Slave Mode Clock Input Select**<br>SCIS selects one out of eight module kernel SCLK input lines that is used as clock input line in slave mode. Note that not all inputs are used in every device of the family. Selecting an unused input returns a continuous low value.<br>The following signal sources are available in this product (if supported by the package!)<br>$000_B$ **P10.2_IN**,<br>$001_B$ **P11.6_IN**, |
| **SLSIS** | 14:12 | rw | **Slave Mode Slave Select Input Selection**<br>The SLSIS must be programmed properly before the slave mode is set with GLOBALCON.MODE and the module is set to RUN mode.<br>The following signal sources are available in this product (if supported by the package!)<br>$000_B$ no input<br>$001_B$ **P11.10_IN**, |
| **0** | 3,<br>7,<br>11,<br>31:15 | r | **Reserved**<br>Read as 0; should be written with 0. |

**QSPI2_PISEL**

**Port Input Select Register** **($004_H$)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **SLSIS** | | | **0** | **SCIS** | | | **0** | **SRIS** | | | **0** | **MRIS** | | |
| r | rw | | | r | rw | | | r | rw | | | r | rw | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **MRIS** | 2:0 | rw | **Master Mode Receive Input Select**<br>MRIS selects one out of eight MRST receive input lines, used in Master Mode. Note that not all inputs are used in every device of the family. Selecting an unused input returns a continuous low value.<br>The following signal sources are available in this product (if supported by the package!)<br>$000_B$ **P15.4_IN**,<br>$001_B$ **P15.7_IN**,<br>$100_B$ **P15.2_IN**, |

| Field | Bits | Type | Description |
|---|---|---|---|
| **SRIS** | 6:4 | rw | **Slave Mode Receive Input Select**<br>SRIS selects one out of eight MTSR receive input lines, used in Slave Mode. Note that not all inputs are used in every device of the family. Selecting an unused input returns a continuous low value.<br>The following signal sources are available in this product (if supported by the package!)<br>$000_B$ **P15.5_IN**,<br>$001_B$ **P15.6_IN**, |
| **SCIS** | 10:8 | rw | **Slave Mode Clock Input Select**<br>SCIS selects one out of eight module kernel SCLK input lines that is used as clock input line in slave mode. Note that not all inputs are used in every device of the family. Selecting an unused input returns a continuous low value.<br>The following signal sources are available in this product (if supported by the package!)<br>$000_B$ **P15.3_IN**,<br>$001_B$ **P15.8_IN**, |
| **SLSIS** | 14:12 | rw | **Slave Mode Slave Select Input Selection**<br>The SLSIS must be programmed properly before the slave mode is set with GLOBALCON.MODE and the module is set to RUN mode.<br>The following signal sources are available in this product (if supported by the package!)<br>$000_B$ no input<br>$001_B$ **P15.2_IN**,<br>$010_B$ **P15.1_IN**, |
| **0** | 3,<br>7,<br>11,<br>31:15 | r | **Reserved**<br>Read as 0; should be written with 0. |

**QSPI3_PISEL**

**Port Input Select Register** **($004_H$)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | | | | | | | | |
| r | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | **SLSIS** | | | **0** | **SCIS** | | | **0** | **SRIS** | | | **0** | **MRIS** | | |
| r | rw | | | r | rw | | | r | rw | | | r | rw | | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **MRIS** | 2:0 | rw | **Master Mode Receive Input Select**<br>MRIS selects one out of eight MRST receive input lines, used in Master Mode. Note that not all inputs are used in every device of the family. Selecting an unused input returns a continuous low value.<br>The following signal sources are available in this product (if supported by the package!)<br>$000_B$ **P02.5_IN**,<br>$001_B$ **P10.7_IN**, |
| **SRIS** | 6:4 | rw | **Slave Mode Receive Input Select**<br>SRIS selects one out of eight MTSR receive input lines, used in Slave Mode. Note that not all inputs are used in every device of the family. Selecting an unused input returns a continuous low value.<br>The following signal sources are available in this product (if supported by the package!)<br>$000_B$ **P02.6_IN**,<br>$001_B$ **P10.6_IN**, |
| **SCIS** | 10:8 | rw | **Slave Mode Clock Input Select**<br>SCIS selects one out of eight module kernel SCLK input lines that is used as clock input line in slave mode. Note that not all inputs are used in every device of the family. Selecting an unused input returns a continuous low value.<br>The following signal sources are available in this product (if supported by the package!)<br>$000_B$ **P02.7_IN**,<br>$001_B$ **P10.8_IN**, |
| **SLSIS** | 14:12 | rw | **Slave Mode Slave Select Input Selection**<br>The SLSIS must be programmed properly before the slave mode is set with GLOBALCON.MODE and the module is set to RUN mode.<br>The following signal sources are available in this product (if supported by the package!)<br>$000_B$ no input<br>$001_B$ **P02.4_IN**, |
| **0** | 3,<br>7,<br>11,<br>31:15 | r | **Reserved**<br>Read as 0; should be written with 0. |

## 35.4 Connectivity

The tables below list all the connections of QSPI instances.

**Table 283 Connections of QSPI0**

| Interface Signals | connects | | Description |
|---|---|---|---|
| QSPI0:MRST | to | IOM:MON2(0) | Slave SPI data output |
| | | IOM:REF2(0) | |
| | | P20.12:ALT(3) | |
| QSPI0:MRSTA | from | P20.12:IN | Master SPI data input |
| QSPI0:MTSR | to | P20.12:ALT(4) | Master SPI data output |
| | | P20.14:ALT(3) | |
| | | P22.5:ALT(4) | |
| QSPI0:MTSRA | from | P20.14:IN | Slave SPI data input |
| QSPI0:MTSRC | from | P22.5:IN | Slave SPI data input |
| QSPI0:SCLK | to | P20.11:ALT(3) | Master SPI clock output |
| | | P20.13:ALT(5) | |
| QSPI0:SCLKA | from | P20.11:IN | Slave SPI clock inputs |
| QSPI0:SLSIA | from | P20.13:IN | Slave select input |
| QSPI0:SLSIB | from | P20.9:IN | Slave select input |
| QSPI0:SLSO(0) | to | P20.8:ALT(3) | Master slave select output |
| QSPI0:SLSO(1) | to | P20.9:ALT(3) | Master slave select output |
| QSPI0:SLSO(2) | to | P20.13:ALT(3) | Master slave select output |
| QSPI0:SLSO(3) | to | P11.10:ALT(3) | Master slave select output |
| QSPI0:SLSO(4) | to | P11.11:ALT(3) | Master slave select output |
| QSPI0:SLSO(5) | to | P11.2:ALT(3) | Master slave select output |
| QSPI0:SLSO(6) | to | P20.10:ALT(3) | Master slave select output |
| QSPI0:SLSO(7) | to | P33.5:ALT(2) | Master slave select output |
| QSPI0:SLSO(8) | to | P20.6:ALT(3) | Master slave select output |
| QSPI0:SLSO(9) | to | P20.3:ALT(3) | Master slave select output |
| QSPI0:SLSO(12) | to | P22.4:ALT(4) | Master slave select output |
| QSPI0:SLSO(13) | to | P15.0:ALT(3) | Master slave select output |
| QSPI0:TX_INT | to | INT:qspi0.TX_INT | QSPI Transmit Service Request |
| QSPI0:RX_INT | to | INT:qspi0.RX_INT | QSPI Receive Service Request |
| QSPI0:ERR_INT | to | INT:qspi0.ERR_INT | QSPI Error Service Request |
| QSPI0:PT_INT | to | INT:qspi0.PT_INT | QSPI Phase Transition Service Request |
| QSPI0:U_INT | to | INT:qspi0.U_INT | QSPI User Defined Service Request |
| QSPI0:HC_INT | to | INT:qspi0.HC_INT | QSPI High Speed Capture Service Request |

**Table 284 Connections of QSPI1**

| Interface Signals | connects | | Description |
|---|---|---|---|
| QSPI1:MRST | to | IOM:MON2(1) | Slave SPI data output |
| | | IOM:REF2(1) | |
| | | P10.1:ALT(3) | |
| | | P10.6:ALT(6) | |
| | | P11.3:ALT(3) | |
| QSPI1:MRSTA | from | P10.1:IN | Master SPI data input |
| QSPI1:MRSTB | from | P11.3:IN | Master SPI data input |
| QSPI1:MTSR | to | P10.1:ALT(2) | Master SPI data output |
| | | P10.3:ALT(3) | |
| | | P10.4:ALT(4) | |
| | | P11.9:ALT(3) | |
| QSPI1:MTSRA | from | P10.3:IN | Slave SPI data input |
| QSPI1:MTSRB | from | P11.9:IN | Slave SPI data input |
| QSPI1:MTSRC | from | P10.4:IN | Slave SPI data input |
| QSPI1:SCLK | to | P10.2:ALT(3) | Master SPI clock output |
| | | P11.6:ALT(3) | |
| QSPI1:SCLKA | from | P10.2:IN | Slave SPI clock inputs |
| QSPI1:SCLKB | from | P11.6:IN | Slave SPI clock inputs |
| QSPI1:SLSIA | from | P11.10:IN | Slave select input |
| QSPI1:SLSO(0) | to | P20.8:ALT(4) | Master slave select output |
| QSPI1:SLSO(1) | to | P20.9:ALT(4) | Master slave select output |
| QSPI1:SLSO(2) | to | P20.13:ALT(4) | Master slave select output |
| QSPI1:SLSO(3) | to | P11.10:ALT(4) | Master slave select output |
| QSPI1:SLSO(4) | to | P11.11:ALT(4) | Master slave select output |
| QSPI1:SLSO(5) | to | P11.2:ALT(4) | Master slave select output |
| QSPI1:SLSO(6) | to | P33.10:ALT(2) | Master slave select output |
| QSPI1:SLSO(7) | to | P33.5:ALT(3) | Master slave select output |
| QSPI1:SLSO(8) | to | P10.4:ALT(3) | Master slave select output |
| QSPI1:SLSO(9) | to | P10.5:ALT(4) | Master slave select output |
| QSPI1:SLSO(10) | to | P10.0:ALT(3) | Master slave select output |
| QSPI1:TX_INT | to | INT:qspi1.TX_INT | QSPI Transmit Service Request |
| QSPI1:RX_INT | to | INT:qspi1.RX_INT | QSPI Receive Service Request |
| QSPI1:ERR_INT | to | INT:qspi1.ERR_INT | QSPI Error Service Request |
| QSPI1:PT_INT | to | INT:qspi1.PT_INT | QSPI Phase Transition Service Request |
| QSPI1:U_INT | to | INT:qspi1.U_INT | QSPI User Defined Service Request |
| QSPI1:HC_INT | to | INT:qspi1.HC_INT | QSPI High Speed Capture Service Request |

**Table 285 Connections of QSPI2**

| Interface Signals | connects | | Description |
|---|---|---|---|
| QSPI2:HSICINA | from | P15.2:IN | Highspeed capture channel |
| QSPI2:HSICINB | from | P15.3:IN | Highspeed capture channel |
| QSPI2:MRST | to | IOM:MON2(2) | Slave SPI data output |
| | | IOM:REF2(2) | |
| | | P15.4:ALT(3) | |
| | | P15.7:ALT(3) | |
| QSPI2:MRSTA | from | P15.4:IN | Master SPI data input |
| QSPI2:MRSTB | from | P15.7:IN | Master SPI data input |
| QSPI2:MRSTE | from | P15.2:IN | Master SPI data input |
| QSPI2:MTSR | to | P15.5:ALT(3) | Master SPI data output |
| | | P15.6:ALT(3) | |
| QSPI2:MTSRA | from | P15.5:IN | Slave SPI data input |
| QSPI2:MTSRB | from | P15.6:IN | Slave SPI data input |
| QSPI2:SCLK | to | P15.3:ALT(3) | Master SPI clock output |
| | | P15.6:ALT(5) | |
| | | P15.8:ALT(3) | |
| | | P33.1:ALT(3) | |
| QSPI2:SCLKA | from | P15.3:IN | Slave SPI clock inputs |
| QSPI2:SCLKB | from | P15.8:IN | Slave SPI clock inputs |
| QSPI2:SLSIA | from | P15.2:IN | Slave select input |
| QSPI2:SLSIB | from | P15.1:IN | Slave select input |
| QSPI2:SLSO(0) | to | P15.2:ALT(3) | Master slave select output |
| QSPI2:SLSO(1) | to | P14.2:ALT(3) | Master slave select output |
| QSPI2:SLSO(2) | to | P14.6:ALT(3) | Master slave select output |
| QSPI2:SLSO(3) | to | P14.3:ALT(3) | Master slave select output |
| QSPI2:SLSO(4) | to | P14.7:ALT(3) | Master slave select output |
| QSPI2:SLSO(5) | to | P15.1:ALT(3) | Master slave select output |
| QSPI2:SLSO(6) | to | P33.13:ALT(4) | Master slave select output |
| QSPI2:SLSO(7) | to | P20.10:ALT(4) | Master slave select output |
| QSPI2:SLSO(8) | to | P20.6:ALT(4) | Master slave select output |
| QSPI2:SLSO(9) | to | P20.3:ALT(4) | Master slave select output |
| QSPI2:SLSO(10) | to | P33.2:ALT(3) | Master slave select output |
| | | P34.3:ALT(4) | |
| QSPI2:SLSO(11) | to | P33.6:ALT(3) | Master slave select output |
| QSPI2:SLSO(12) | to | P32.6:ALT(4) | Master slave select output |
| | | P33.4:ALT(3) | |
| QSPI2:TX_INT | to | INT:qspi2.TX_INT | QSPI Transmit Service Request |

**Table 285 Connections of QSPI2** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| QSPI2:RX_INT | to | INT:qspi2.RX_INT | QSPI Receive Service Request |
| QSPI2:ERR_INT | to | INT:qspi2.ERR_INT | QSPI Error Service Request |
| QSPI2:PT_INT | to | INT:qspi2.PT_INT | QSPI Phase Transition Service Request |
| QSPI2:U_INT | to | INT:qspi2.U_INT | QSPI User Defined Service Request |
| QSPI2:HC_INT | to | INT:qspi2.HC_INT | QSPI High Speed Capture Service Request |

**Table 286 Connections of QSPI3**

| Interface Signals | connects | | Description |
|---|---|---|---|
| QSPI3:HSICINA | from | P33.9:IN | Highspeed capture channel |
| QSPI3:HSICINB | from | P33.10:IN | Highspeed capture channel |
| QSPI3:MRST | to | IOM:MON2(3) | Slave SPI data output |
| | | IOM:REF2(3) | |
| | | P02.5:ALT(3) | |
| | | P10.7:ALT(3) | |
| QSPI3:MRSTA | from | P02.5:IN | Master SPI data input |
| QSPI3:MRSTB | from | P10.7:IN | Master SPI data input |
| QSPI3:MTSR | to | P02.6:ALT(3) | Master SPI data output |
| | | P10.6:ALT(3) | |
| QSPI3:MTSRA | from | P02.6:IN | Slave SPI data input |
| QSPI3:MTSRB | from | P10.6:IN | Slave SPI data input |
| QSPI3:SCLK | to | P02.7:ALT(3) | Master SPI clock output |
| | | P10.8:ALT(3) | |
| QSPI3:SCLKA | from | P02.7:IN | Slave SPI clock inputs |
| QSPI3:SCLKB | from | P10.8:IN | Slave SPI clock inputs |
| QSPI3:SLSIA | from | P02.4:IN | Slave select input |
| QSPI3:SLSO(0) | to | P02.4:ALT(3) | Master slave select output |
| QSPI3:SLSO(1) | to | P02.0:ALT(3) | Master slave select output |
| QSPI3:SLSO(2) | to | P02.1:ALT(3) | Master slave select output |
| QSPI3:SLSO(3) | to | P00.5:ALT(3) | Master slave select output |
| | | P02.2:ALT(3) | |
| QSPI3:SLSO(4) | to | P00.2:ALT(6) | Master slave select output |
| | | P02.3:ALT(3) | |
| QSPI3:SLSO(5) | to | P02.8:ALT(2) | Master slave select output |
| QSPI3:SLSO(6) | to | P00.8:ALT(2) | Master slave select output |
| QSPI3:SLSO(7) | to | P00.9:ALT(2) | Master slave select output |
| QSPI3:SLSO(8) | to | P10.5:ALT(3) | Master slave select output |
| QSPI3:TX_INT | to | INT:qspi3.TX_INT | QSPI Transmit Service Request |
| QSPI3:RX_INT | to | INT:qspi3.RX_INT | QSPI Receive Service Request |

**Table 286 Connections of QSPI3** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| QSPI3:ERR_INT | to | INT:qspi3.ERR_INT | QSPI Error Service Request |
| QSPI3:PT_INT | to | INT:qspi3.PT_INT | QSPI Phase Transition Service Request |
| QSPI3:U_INT | to | INT:qspi3.U_INT | QSPI User Defined Service Request |
| QSPI3:HC_INT | to | INT:qspi3.HC_INT | QSPI High Speed Capture Service Request |

## 35.5 Revision History

**Table 287 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V3.0.20** | | |
| | No functional change. | |

# 36 Micro Second Channel (MSC)

This device doesn’t contain a MSC.

# 37 Single Edge Nibble Transmission (SENT)

This device doesn’t contain a SENT.

# 38 CAN Interface (MCMCAN)

This section describes the MCMCAN Interface specific appendix for the product TC35x.

## 38.1 TC35x Specific IP Configuration

**Table 288 TC35x specific configuration of CAN**

| Parameter | CAN0 | CAN1 |
|---|---|---|
| **Node size in byte** | 1024 | 1024 |
| **Number of CAN Nodes** | 4 | 4 |
| **RAM size in byte** | 32768 | 16384 |
| **Maximum Number of Standard ID Filter Messages per node** | 128 | 128 |
| **Maximum Number of Extended ID Filter Messages per node** | 64 | 64 |
| **Maximum Number of RxFIFO structures per node** | 2 | 2 |
| **Maximum Number of Messages in a Rx buffer per node** | 64 | 64 |
| **Maximum Number of Tx Event Messages per node** | 32 | 32 |
| **Maximum Number of Tx Messages in a Tx Buffer per node** | 32 | 32 |

## 38.2 TC35x Specific Register Set

### Register Address Space Table

**Table 289 Register Address Space - CAN**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| CAN0 | $F0200000_H$ | $F0208FFF_H$ | Bus Interface |
| CAN1 | $F0210000_H$ | $F0218FFF_H$ | Bus Interface |

### Register Overview Table

**Table 290 Register Overview - CAN (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CAN0_RAM | Embedded SRAM for messages ($008000_H$ Byte) | $000000_H$ | |
| CAN1_RAM | Embedded SRAM for messages ($004000_H$ Byte) | $000000_H$ | |
| CAN0_CLC | CAN Clock Control Register | $008000_H$ | See Family Spec |
| CAN1_CLC | CAN Clock Control Register | $008000_H$ | See Family Spec |
| CAN0_ID | Module Identification Register | $008008_H$ | See Family Spec |
| CAN1_ID | Module Identification Register | $008008_H$ | See Family Spec |
| CAN0_MCR | Module Control Register | $008030_H$ | See Family Spec |
| CAN1_MCR | Module Control Register | $008030_H$ | 12 |
| CAN0_BUFADR | Buffer receive address and transmit address | $008034_H$ | See Family Spec |
| CAN0_MECR | Measure Control Register | $008040_H$ | See Family Spec |
| CAN0_MESTAT | Measure Status Register | $008044_H$ | See Family Spec |

**Table 290 Register Overview - CAN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CAN0_ACCENCTR0 | Access Enable Register Control 0 | $0080DC_H$ | See Family Spec |
| CAN1_ACCENCTR0 | Access Enable Register Control 0 | $0080DC_H$ | See Family Spec |
| CAN0_OCS | OCDS Control and Status | $0080E8_H$ | See Family Spec |
| CAN1_OCS | OCDS Control and Status | $0080E8_H$ | See Family Spec |
| CAN0_KRSTCLR | Kernel Reset Status Clear Register | $0080EC_H$ | See Family Spec |
| CAN1_KRSTCLR | Kernel Reset Status Clear Register | $0080EC_H$ | See Family Spec |
| CAN0_KRST1 | Kernel Reset Register 1 | $0080F0_H$ | See Family Spec |
| CAN1_KRST1 | Kernel Reset Register 1 | $0080F0_H$ | See Family Spec |
| CAN0_KRST0 | Kernel Reset Register 0 | $0080F4_H$ | See Family Spec |
| CAN1_KRST0 | Kernel Reset Register 0 | $0080F4_H$ | See Family Spec |
| CAN0_ACCEN0 | Access Enable Register 0 | $0080FC_H$ | See Family Spec |
| CAN1_ACCEN0 | Access Enable Register 0 | $0080FC_H$ | See Family Spec |
| CAN0_ACCENNODEi0 (i=0-3) | Access Enable Register CAN Node i 0 | $008100_H$+i*400 H | See Family Spec |
| CAN1_ACCENNODEi0 (i=0-3) | Access Enable Register CAN Node i 0 | $008100_H$+i*400 H | See Family Spec |

**Table 290 Register Overview - CAN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CAN0_STARTADRi (i=0-3) | Start Address Node i | $008108_H$+i*400 H | See Family Spec |
| CAN1_STARTADRi (i=0-3) | Start Address Node i | $008108_H$+i*400 H | See Family Spec |
| CAN0_ENDADRi (i=0-3) | End Address Node i | $00810C_H$+i*40 $0_H$ | See Family Spec |
| CAN1_ENDADRi (i=0-3) | End Address Node i | $00810C_H$+i*40 $0_H$ | See Family Spec |
| CAN0_ISREGi (i=0-3) | Interrupt Signalling Register i | $008110_H$+i*400 H | See Family Spec |
| CAN1_ISREGi (i=0-3) | Interrupt Signalling Register i | $008110_H$+i*400 H | See Family Spec |
| CAN0_GRINT1i (i=0-3) | Interrupt routing for Groups 1 i | $008114_H$+i*400 H | See Family Spec |
| CAN1_GRINT1i (i=0-3) | Interrupt routing for Groups 1 i | $008114_H$+i*400 H | See Family Spec |
| CAN0_GRINT2i (i=0-3) | Interrupt routing for Groups 2 i | $008118_H$+i*400 H | See Family Spec |
| CAN1_GRINT2i (i=0-3) | Interrupt routing for Groups 2 i | $008118_H$+i*400 H | See Family Spec |
| CAN0_NTCCRi (i=0-3) | Node i Timer Clock Control Register | $008120_H$+i*400 H | See Family Spec |
| CAN1_NTCCRi (i=0-3) | Node i Timer Clock Control Register | $008120_H$+i*400 H | See Family Spec |
| CAN0_NTATTRi (i=0-3) | Node i Timer A Transmit Trigger Register | $008124_H$+i*400 H | See Family Spec |
| CAN1_NTATTRi (i=0-3) | Node i Timer A Transmit Trigger Register | $008124_H$+i*400 H | See Family Spec |

**Table 290 Register Overview - CAN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CAN0_NTBTTRi (i=0-3) | Node i Timer B Transmit Trigger Register | $008128_H$+i*400 H | See Family Spec |
| CAN1_NTBTTRi (i=0-3) | Node i Timer B Transmit Trigger Register | $008128_H$+i*400 H | See Family Spec |
| CAN0_NTCTTRi (i=0-3) | Node i Timer C Transmit Trigger Register | $00812C_H$+i*40 $0_H$ | See Family Spec |
| CAN1_NTCTTRi (i=0-3) | Node i Timer C Transmit Trigger Register | $00812C_H$+i*40 $0_H$ | See Family Spec |
| CAN0_NTRTRi (i=0-3) | Node i Timer Receive Timeout Register | $008130_H$+i*400 H | See Family Spec |
| CAN1_NTRTRi (i=0-3) | Node i Timer Receive Timeout Register | $008130_H$+i*400 H | See Family Spec |
| CAN0_NPCRi (i=0-3) | Node i Port Control Register | $008140_H$+i*400 H | See Family Spec |
| CAN1_NPCRi (i=0-3) | Node i Port Control Register | $008140_H$+i*400 H | See Family Spec |
| CAN0_CRELi (i=0-3) | Core Release Register i | $008200_H$+i*400 H | See Family Spec |
| CAN1_CRELi (i=0-3) | Core Release Register i | $008200_H$+i*400 H | See Family Spec |
| CAN0_ENDNi (i=0-3) | Endian Register i | $008204_H$+i*400 H | See Family Spec |
| CAN1_ENDNi (i=0-3) | Endian Register i | $008204_H$+i*400 H | See Family Spec |
| CAN0_DBTPi (i=0-3) | Data Bit Timing & Prescaler Register i | $00820C_H$+i*40 $0_H$ | See Family Spec |
| CAN1_DBTPi (i=0-3) | Data Bit Timing & Prescaler Register i | $00820C_H$+i*40 $0_H$ | See Family Spec |

**Table 290 Register Overview - CAN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CAN0_TESTi (i=0-3) | Test Register i | $008210_H$+i*400 H | See Family Spec |
| CAN1_TESTi (i=0-3) | Test Register i | $008210_H$+i*400 H | See Family Spec |
| CAN0_RWDi (i=0-3) | RAM Watchdog i | $008214_H$+i*400 H | See Family Spec |
| CAN1_RWDi (i=0-3) | RAM Watchdog i | $008214_H$+i*400 H | See Family Spec |
| CAN0_CCCRi (i=0-3) | CC Control Register i | $008218_H$+i*400 H | See Family Spec |
| CAN1_CCCRi (i=0-3) | CC Control Register i | $008218_H$+i*400 H | See Family Spec |
| CAN0_NBTPi (i=0-3) | Nominal Bit Timing & Prescaler Register i | $00821C_H$+i*40 $0_H$ | See Family Spec |
| CAN1_NBTPi (i=0-3) | Nominal Bit Timing & Prescaler Register i | $00821C_H$+i*40 $0_H$ | See Family Spec |
| CAN0_TSCCi (i=0-3) | Timestamp Counter Configuration i | $008220_H$+i*400 H | See Family Spec |
| CAN1_TSCCi (i=0-3) | Timestamp Counter Configuration i | $008220_H$+i*400 H | See Family Spec |
| CAN0_TSCVi (i=0-3) | Timestamp Counter Value i | $008224_H$+i*400 H | See Family Spec |
| CAN1_TSCVi (i=0-3) | Timestamp Counter Value i | $008224_H$+i*400 H | See Family Spec |
| CAN0_TOCCi (i=0-3) | Timeout Counter Configuration i | $008228_H$+i*400 H | See Family Spec |
| CAN1_TOCCi (i=0-3) | Timeout Counter Configuration i | $008228_H$+i*400 H | See Family Spec |

**Table 290 Register Overview - CAN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CAN0_TOCVi (i=0-3) | Timeout Counter Value i | $00822C_H$+i*40 $0_H$ | See Family Spec |
| CAN1_TOCVi (i=0-3) | Timeout Counter Value i | $00822C_H$+i*40 $0_H$ | See Family Spec |
| CAN0_ECRi (i=0-3) | Error Counter Register i | $008240_H$+i*400 H | See Family Spec |
| CAN1_ECRi (i=0-3) | Error Counter Register i | $008240_H$+i*400 H | See Family Spec |
| CAN0_PSRi (i=0-3) | Protocol Status Register i | $008244_H$+i*400 H | See Family Spec |
| CAN1_PSRi (i=0-3) | Protocol Status Register i | $008244_H$+i*400 H | See Family Spec |
| CAN0_TDCRi (i=0-3) | Transmitter Delay Compensation Register i | $008248_H$+i*400 H | See Family Spec |
| CAN1_TDCRi (i=0-3) | Transmitter Delay Compensation Register i | $008248_H$+i*400 H | See Family Spec |
| CAN0_IRi (i=0-3) | Interrupt Register i | $008250_H$+i*400 H | See Family Spec |
| CAN1_IRi (i=0-3) | Interrupt Register i | $008250_H$+i*400 H | See Family Spec |
| CAN0_IEi (i=0-3) | Interrupt Enable i | $008254_H$+i*400 H | See Family Spec |
| CAN1_IEi (i=0-3) | Interrupt Enable i | $008254_H$+i*400 H | See Family Spec |
| CAN0_GFCi (i=0-3) | Global Filter Configuration i | $008280_H$+i*400 H | See Family Spec |
| CAN1_GFCi (i=0-3) | Global Filter Configuration i | $008280_H$+i*400 H | See Family Spec |

**Table 290 Register Overview - CAN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CAN0_SIDFCi (i=0-3) | Standard ID Filter Configuration i | $008284_H$+i*400 H | See Family Spec |
| CAN1_SIDFCi (i=0-3) | Standard ID Filter Configuration i | $008284_H$+i*400 H | See Family Spec |
| CAN0_XIDFCi (i=0-3) | Extended ID Filter Configuration i | $008288_H$+i*400 H | See Family Spec |
| CAN1_XIDFCi (i=0-3) | Extended ID Filter Configuration i | $008288_H$+i*400 H | See Family Spec |
| CAN0_XIDAMi (i=0-3) | Extended ID AND Mask i | $008290_H$+i*400 H | See Family Spec |
| CAN1_XIDAMi (i=0-3) | Extended ID AND Mask i | $008290_H$+i*400 H | See Family Spec |
| CAN0_HPMSi (i=0-3) | High Priority Message Status i | $008294_H$+i*400 H | See Family Spec |
| CAN1_HPMSi (i=0-3) | High Priority Message Status i | $008294_H$+i*400 H | See Family Spec |
| CAN0_NDAT1i (i=0-3) | New Data 1 i | $008298_H$+i*400 H | See Family Spec |
| CAN1_NDAT1i (i=0-3) | New Data 1 i | $008298_H$+i*400 H | See Family Spec |
| CAN0_NDAT2i (i=0-3) | New Data 2 i | $00829C_H$+i*40 $0_H$ | See Family Spec |
| CAN1_NDAT2i (i=0-3) | New Data 2 i | $00829C_H$+i*40 $0_H$ | See Family Spec |
| CAN0_RXF0Ci (i=0-3) | Rx FIFO 0 Configuration i | $0082A0_H$+i*40 $0_H$ | See Family Spec |
| CAN1_RXF0Ci (i=0-3) | Rx FIFO 0 Configuration i | $0082A0_H$+i*40 $0_H$ | See Family Spec |

**Table 290 Register Overview - CAN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CAN0_RXF0Si (i=0-3) | Rx FIFO 0 Status i | $0082A4_H$+i*40 $0_H$ | See Family Spec |
| CAN1_RXF0Si (i=0-3) | Rx FIFO 0 Status i | $0082A4_H$+i*40 $0_H$ | See Family Spec |
| CAN0_RXF0Ai (i=0-3) | Rx FIFO 0 Acknowledge i | $0082A8_H$+i*40 $0_H$ | See Family Spec |
| CAN1_RXF0Ai (i=0-3) | Rx FIFO 0 Acknowledge i | $0082A8_H$+i*40 $0_H$ | See Family Spec |
| CAN0_RXBCi (i=0-3) | Rx Buffer Configuration i | $0082AC_H$+i*40 $0_H$ | See Family Spec |
| CAN1_RXBCi (i=0-3) | Rx Buffer Configuration i | $0082AC_H$+i*40 $0_H$ | See Family Spec |
| CAN0_RXF1Ci (i=0-3) | Rx FIFO 1 Configuration i | $0082B0_H$+i*40 $0_H$ | See Family Spec |
| CAN1_RXF1Ci (i=0-3) | Rx FIFO 1 Configuration i | $0082B0_H$+i*40 $0_H$ | See Family Spec |
| CAN0_RXF1Si (i=0-3) | Rx FIFO 1 Status i | $0082B4_H$+i*40 $0_H$ | See Family Spec |
| CAN1_RXF1Si (i=0-3) | Rx FIFO 1 Status i | $0082B4_H$+i*40 $0_H$ | See Family Spec |
| CAN0_RXF1Ai (i=0-3) | Rx FIFO 1 Acknowledge i | $0082B8_H$+i*40 $0_H$ | See Family Spec |
| CAN1_RXF1Ai (i=0-3) | Rx FIFO 1 Acknowledge i | $0082B8_H$+i*40 $0_H$ | See Family Spec |
| CAN0_RXESCi (i=0-3) | Rx Buffer/FIFO Element Size Configuration i | $0082BC_H$+i*40 $0_H$ | See Family Spec |
| CAN1_RXESCi (i=0-3) | Rx Buffer/FIFO Element Size Configuration i | $0082BC_H$+i*40 $0_H$ | See Family Spec |

**Table 290 Register Overview - CAN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CAN0_TXBCi (i=0-3) | Tx Buffer Configuration i | $0082C0_H$+i*40 $0_H$ | See Family Spec |
| CAN1_TXBCi (i=0-3) | Tx Buffer Configuration i | $0082C0_H$+i*40 $0_H$ | See Family Spec |
| CAN0_TXFQSi (i=0-3) | Tx FIFO/Queue Status i | $0082C4_H$+i*40 $0_H$ | See Family Spec |
| CAN1_TXFQSi (i=0-3) | Tx FIFO/Queue Status i | $0082C4_H$+i*40 $0_H$ | See Family Spec |
| CAN0_TXESCi (i=0-3) | Tx Buffer Element Size Configuration i | $0082C8_H$+i*40 $0_H$ | See Family Spec |
| CAN1_TXESCi (i=0-3) | Tx Buffer Element Size Configuration i | $0082C8_H$+i*40 $0_H$ | See Family Spec |
| CAN0_TXBRPi (i=0-3) | Tx Buffer Request Pending i | $0082CC_H$+i*40 $0_H$ | See Family Spec |
| CAN1_TXBRPi (i=0-3) | Tx Buffer Request Pending i | $0082CC_H$+i*40 $0_H$ | See Family Spec |
| CAN0_TXBARi (i=0-3) | Tx Buffer Add Request i | $0082D0_H$+i*40 $0_H$ | See Family Spec |
| CAN1_TXBARi (i=0-3) | Tx Buffer Add Request i | $0082D0_H$+i*40 $0_H$ | See Family Spec |
| CAN0_TXBCRi (i=0-3) | Tx Buffer Cancellation Request i | $0082D4_H$+i*40 $0_H$ | See Family Spec |
| CAN1_TXBCRi (i=0-3) | Tx Buffer Cancellation Request i | $0082D4_H$+i*40 $0_H$ | See Family Spec |
| CAN0_TXBTOi (i=0-3) | Tx Buffer Transmission Occurred i | $0082D8_H$+i*40 $0_H$ | See Family Spec |
| CAN1_TXBTOi (i=0-3) | Tx Buffer Transmission Occurred i | $0082D8_H$+i*40 $0_H$ | See Family Spec |

**Table 290 Register Overview - CAN (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| CAN0_TXBCFi (i=0-3) | Tx Buffer Cancellation Finished i | $0082DC_H$+i*40 $0_H$ | See Family Spec |
| CAN1_TXBCFi (i=0-3) | Tx Buffer Cancellation Finished i | $0082DC_H$+i*40 $0_H$ | See Family Spec |
| CAN0_TXBTIEi (i=0-3) | Tx Buffer Transmission Interrupt Enable i | $0082E0_H$+i*40 $0_H$ | See Family Spec |
| CAN1_TXBTIEi (i=0-3) | Tx Buffer Transmission Interrupt Enable i | $0082E0_H$+i*40 $0_H$ | See Family Spec |
| CAN0_TXBCIEi (i=0-3) | Tx Buffer Cancellation Finished Interrupt Enable i | $0082E4_H$+i*40 $0_H$ | See Family Spec |
| CAN1_TXBCIEi (i=0-3) | Tx Buffer Cancellation Finished Interrupt Enable i | $0082E4_H$+i*40 $0_H$ | See Family Spec |
| CAN0_TXEFCi (i=0-3) | Tx Event FIFO Configuration i | $0082F0_H$+i*400 H | See Family Spec |
| CAN1_TXEFCi (i=0-3) | Tx Event FIFO Configuration i | $0082F0_H$+i*400 H | See Family Spec |
| CAN0_TXEFSi (i=0-3) | Tx Event FIFO Status i | $0082F4_H$+i*400 H | See Family Spec |
| CAN1_TXEFSi (i=0-3) | Tx Event FIFO Status i | $0082F4_H$+i*400 H | See Family Spec |
| CAN0_TXEFAi (i=0-3) | Tx Event FIFO Acknowledge i | $0082F8_H$+i*400 H | See Family Spec |
| CAN1_TXEFAi (i=0-3) | Tx Event FIFO Acknowledge i | $0082F8_H$+i*400 H | See Family Spec |

## 38.3 TC35x Specific Registers

### 38.3.1 Bus Interface

**Module Control Register**

The Module Control Register MCR contains basic settings that determine the operation of the MCMCAN module.

The write access to the lowest byte of the MCR register becomes only valid, if and only if, MCR.CCCE and MCR.CI are already set during write access. To switch the clocks on or off, the bits of MCR.CCCE and MCR.CI have to be reset afterwards. Before this sequence hasn't taken place, no write access to the corresponding nodes, can be done.

*Note: If the baud rate logic is supplied from an unstable clock source, or no clock at all, the CAN functionality is not guaranteed.*

To be able to change the clock settings the following programming sequence needs to be met:

uwTemp = CANn_MCR.U;

uwTemp |= (0xC0000000 | CLKSELx);

CANn_MCR.U = uwTemp;

uwTemp &= ~0xC0000000;

CANn_MCR.U = uwTemp;

The clock settings for CAN nodes becomes active.

To be able to start the RAM initialization, the following programming sequence need to be met:

CANn_MCR |= 0xC0000000;

Wait until CANn_MCR.RBUSY is 0b

Set CANn_MCR.RINIT to 0b

Set CANn_MCR.RINIT to 1b

Dummy read CANn_MCR

Wait until CANn_MCR.RBUSY is 0b

Set CANn_MCR.RINIT to 0b

CANn_MCR &= ~0xC0000000;

RAM initialization is finished

**CAN1_MCR**

**Module Control Register** **($008030_H$)** **Application Reset Value: $0000\ 0000_H$**

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CCCE** | **CI** | **RINIT** | **RBUSY** | **0** | | | | **0** | | | | | | | |
| rw | rw | rw | rh | r | | | | r | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | | | | | | | | **CLKSEL3** | | **CLKSEL2** | | **CLKSEL1** | | **CLKSEL0** | |
| r | | | | | | | | rw | | rw | | rw | | rw | |

| Field | Bits | Type | Description |
|---|---|---|---|
| **CLKSEL0** | 1:0 | rw | **Clock Select 0**<br>This bitfield is MCR.CI and MCR.CCCE protected.<br>$00_B$ No clock supplied<br>$01_B$ The asynchronous clock source is switched on<br>$10_B$ The synchronous clock source is switched on<br>$11_B$ Both clock sources are switched on |
| **CLKSEL1** | 3:2 | rw | **Clock Select 1**<br>This bitfield is MCR.CI and MCR.CCCE protected.<br>$00_B$ No clock supplied<br>$01_B$ The asynchronous clock source is switched on<br>$10_B$ The synchronous clock source is switched on<br>$11_B$ Both clock sources are switched on |
| **CLKSEL2** | 5:4 | rw | **Clock Select 2**<br>This bitfield is MCR.CI and MCR.CCCE protected.<br>$00_B$ No clock supplied<br>$01_B$ The asynchronous clock source is switched on<br>$10_B$ The synchronous clock source is switched on<br>$11_B$ Both clock sources are switched on |
| **CLKSEL3** | 7:6 | rw | **Clock Select 3**<br>This bitfield is MCR.CI and MCR.CCCE protected.<br>$00_B$ No clock supplied<br>$01_B$ The asynchronous clock source is switched on<br>$10_B$ The synchronous clock source is switched on<br>$11_B$ Both clock sources are switched on |
| **RBUSY** | 28 | rh | **RAM BUSY**<br>This bit shows that the RAM Initialization is running. This bit is set back to 0b by hardware when the RAM intialization is completed. |
| **RINIT** | 29 | rw | **RAM Init**<br>This bit is MCR.CI and MCR.CCCE protected.<br>This bit starts the initialization of the RAM block to all 0x0.<br>The RAM initialization is started only when this bit is changed from 0b to 1b and also RBUSY is 0b. |
| **CI** | 30 | rw | **Change Init**<br>Needs to be set to enable and disable clocks.<br>$0_B$ Change Init disabled<br>$1_B$ Change Init enabled (takes effect with CCCE:=1) |
| **CCCE** | 31 | rw | **Clock and RAM Change Enable**<br>Needs to be set to enable and disable the clocks.<br>$0_B$ Clock and RAM Change disabled<br>$1_B$ Clock and RAM Change enabled (takes effect with CI:=1) |
| **0** | 23:8,<br>27:24 | r | **Reserved**<br>Shall read 0; shall be written with 0. |

## 38.4 Connectivity

**Table 291 Connections of CAN0**

| Interface Signals | connects | | Description |
|---|---|---|---|
| CAN0:DSTDBG | from | DMU:SCU_ENTERED_DEST_DBG | Destructive Debug entered |
| CAN0:DXSCLK | to | TCU:dxs_clk | DXS Clock, DAP module clock |
| CAN0:INT(5:0) | to | HSM:EXT_INT(18:13) | CAN interrupt request |
| CAN0:INT(12) | to | CCU61:CC61IND | CAN interrupt request |
| CAN0:INT(15) | to | CCU61:T13HRE | CAN interrupt request |
| CAN0:STM0_SR0_INT | from | STM0:SR0_INT | System Timer Service Request 0 |
| CAN0:STM0_SR1_INT | from | STM0:SR1_INT | System Timer Service Request 1 |
| CAN0:STM1_SR0_INT | from | STM1:SR0_INT | System Timer Service Request 0 |
| CAN0:STM1_SR1_INT | from | STM1:SR1_INT | System Timer Service Request 1 |
| CAN0:STM2_SR0_INT | from | STM2:SR0_INT | System Timer Service Request 0 |
| CAN0:STM2_SR1_INT | from | STM2:SR1_INT | System Timer Service Request 1 |
| CAN0:INT(15:0) | to | INT:mcmcan0.INT(15:0) | CAN Service Request |

**Table 292 Connections of CAN00**

| Interface Signals | connects | | Description |
|---|---|---|---|
| CAN00:RXDA | from | P02.1:IN | CAN receive input node 0 |
| CAN00:RXDB | from | P20.7:IN | CAN receive input node 0 |
| CAN00:RXDC | from | P12.0:IN | CAN receive input node 0 |
| CAN00:RXDD | from | P33.12:IN | CAN receive input node 0 |
| CAN00:RXDE | from | P33.7:IN | CAN receive input node 0 |
| CAN00:RXDG | from | P34.2:IN | CAN receive input node 0 |
| CAN00:TXD | to | IOM:MON2(5) | CAN transmit output node 0 |
| | | IOM:REF2(5) | |
| | | P02.0:ALT(5) | |
| | | P12.1:ALT(5) | |
| | | P20.8:ALT(5) | |
| | | P33.8:ALT(5) | |
| | | P33.13:ALT(5) | |
| | | P34.1:ALT(4) | |

**Table 293 Connections of CAN01**

| Interface Signals | connects | | Description |
|---|---|---|---|
| CAN01:RXDA | from | P15.3:IN | CAN receive input node 1 |
| CAN01:RXDB | from | P14.1:IN | CAN receive input node 1 |
| CAN01:RXDD | from | P33.10:IN | CAN receive input node 1 |

**Table 293 Connections of CAN01** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| CAN01:TXD | to | IOM:MON2(6) | CAN transmit output node 1 |
| | | IOM:REF2(6) | |
| | | P14.0:ALT(5) | |
| | | P15.2:ALT(5) | |
| | | P33.9:ALT(5) | |

**Table 294 Connections of CAN1**

| Interface Signals | connects | | Description |
|---|---|---|---|
| CAN1:DSTDBG | from | DMU:SCU_ENTERED_DEST_DBG | Destructive Debug entered |
| CAN1:STM0_SR0_INT | from | STM0:SR0_INT | System Timer Service Request 0 |
| CAN1:STM0_SR1_INT | from | STM0:SR1_INT | System Timer Service Request 1 |
| CAN1:STM1_SR0_INT | from | STM1:SR0_INT | System Timer Service Request 0 |
| CAN1:STM1_SR1_INT | from | STM1:SR1_INT | System Timer Service Request 1 |
| CAN1:STM2_SR0_INT | from | STM2:SR0_INT | System Timer Service Request 0 |
| CAN1:STM2_SR1_INT | from | STM2:SR1_INT | System Timer Service Request 1 |
| CAN1:INT(15:0) | to | INT:mcmcan1.INT(15:0) | CAN Service Request |

**Table 295 Connections of CAN02**

| Interface Signals | connects | | Description |
|---|---|---|---|
| CAN02:RXDA | from | P15.1:IN | CAN receive input node 2 |
| CAN02:RXDB | from | P02.3:IN | CAN receive input node 2 |
| CAN02:RXDC | from | P32.6:IN | CAN receive input node 2 |
| CAN02:RXDD | from | P14.8:IN | CAN receive input node 2 |
| CAN02:RXDE | from | P10.2:IN | CAN receive input node 2 |
| CAN02:TXD | to | IOM:MON2(7) | CAN transmit output node 2 |
| | | IOM:REF2(7) | |
| | | P02.2:ALT(5) | |
| | | P10.3:ALT(6) | |
| | | P14.10:ALT(5) | |
| | | P15.0:ALT(5) | |
| | | P32.5:ALT(6) | |

**Table 296 Connections of CAN03**

| Interface Signals | connects | | Description |
|---|---|---|---|
| CAN03:RXDA | from | P00.3:IN | CAN receive input node 3 |
| CAN03:RXDB | from | P32.2:IN | CAN receive input node 3 |
| CAN03:RXDC | from | P20.0:IN | CAN receive input node 3 |

**Table 296 Connections of CAN03** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| CAN03:RXDD | from | P11.10:IN | CAN receive input node 3 |
| CAN03:RXDE | from | P20.9:IN | CAN receive input node 3 |
| CAN03:TXD | to | IOM:MON2(8) | CAN transmit output node 3 |
| | | IOM:REF2(8) | |
| | | P00.2:ALT(5) | |
| | | P11.12:ALT(5) | |
| | | P20.3:ALT(5) | |
| | | P20.10:ALT(5) | |
| | | P32.3:ALT(5) | |

**Table 297 Connections of CAN10**

| Interface Signals | connects | | Description |
|---|---|---|---|
| CAN10:RXDA | from | P00.1:IN | CAN receive input node 0 |
| CAN10:RXDB | from | P14.7:IN | CAN receive input node 0 |
| CAN10:RXDC | from | P23.0:IN | CAN receive input node 0 |
| CAN10:TXD | to | P00.0:ALT(5) | CAN transmit output node 0 |
| | | P14.9:ALT(4) | |
| | | P23.1:ALT(5) | |

**Table 298 Connections of CAN11**

| Interface Signals | connects | | Description |
|---|---|---|---|
| CAN11:RXDA | from | P02.4:IN | CAN receive input node 1 |
| CAN11:RXDB | from | P00.5:IN | CAN receive input node 1 |
| CAN11:RXDD | from | P11.7:IN | CAN receive input node 1 |
| CAN11:TXD | to | P00.4:ALT(3) | CAN transmit output node 1 |
| | | P02.5:ALT(2) | |
| | | P11.0:ALT(5) | |

**Table 299 Connections of CAN12**

| Interface Signals | connects | | Description |
|---|---|---|---|
| CAN12:RXDA | from | P20.6:IN | CAN receive input node 2 |
| CAN12:RXDB | from | P10.8:IN | CAN receive input node 2 |
| CAN12:RXDC | from | P23.3:IN | CAN receive input node 2 |
| CAN12:RXDD | from | P11.8:IN | CAN receive input node 2 |
| CAN12:TXD | to | P10.7:ALT(6) | CAN transmit output node 2 |
| | | P11.1:ALT(5) | |
| | | P20.7:ALT(5) | |
| | | P23.2:ALT(5) | |

**Table 300 Connections of CAN13**

| Interface Signals | connects | | Description |
|---|---|---|---|
| CAN13:RXDA | from | P14.7:IN | CAN receive input node 3 |
| CAN13:RXDB | from | P33.5:IN | CAN receive input node 3 |
| CAN13:RXDC | from | P22.5:IN | CAN receive input node 3 |
| CAN13:RXDD | from | P11.13:IN | CAN receive input node 3 |
| CAN13:TXD | to | P11.4:ALT(5) | CAN transmit output node 3 |
| | | P14.6:ALT(4) | |
| | | P22.4:ALT(6) | |
| | | P33.4:ALT(7) | |

*Note: For the connectivity of the MCMCAN module to the STM module, please refer to the User Manual, chapter MCMCAN User Interface under CAN Transmit Trigger Inputs section.*

## 38.5 Revision History

**Table 301 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V1.19.8** | | |
| **Page 1** | Update of “specific configuration of CAN” table. | |
| **Page 12** | CAN_MCR register bit field “reserved” fixed. | |
| **V1.19.9** | | |
| | No change. | |
| **V1.19.10** | | |
| **Page 17** | Added note at the end of connections tables. | |
| **V1.19.11** | | |
| – | No functional changes. | |
| **V1.19.12** | | |
| **Page 1** | Update of “specific configuration of CAN” table. | |
| **V1.19.13** | | |
| **Page 12** | Updated information on bit implementation in A-step. | |

# 39 FlexRay™ Protocol Controller (E-Ray)

Text with reference to family spec.

## 39.1 TC35x Specific IP Configuration

No product specific configuration for ERAY

## 39.2 TC35x Specific Register Set

### Register Address Space Table

**Table 302 Register Address Space - ERAY**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| ERAY0 | $F001C000_H$ | $F001CFFF_H$ | FPI slave interface |

### Register Overview Table

**Table 303 Register Overview - ERAY (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ERAY0_CLC | Clock Control Register | $0000_H$ | See Family Spec |
| ERAY0_CUST1 | Busy and Input Buffer Control Register | $0004_H$ | See Family Spec |
| ERAY0_ID | Module Identification Register | $0008_H$ | See Family Spec |
| ERAY0_CUST3 | Customer Interface Timeout Counter Register | $000C_H$ | See Family Spec |
| ERAY0_TEST1 | Test Register 1 | $0010_H$ | See Family Spec |
| ERAY0_TEST2 | Test Register 2 | $0014_H$ | See Family Spec |
| ERAY0_LCK | Lock Register | $001C_H$ | See Family Spec |
| ERAY0_EIR | Error Service Request Select Register | $0020_H$ | See Family Spec |
| ERAY0_SIR | Status Service Request Register | $0024_H$ | See Family Spec |
| ERAY0_EILS | Error Service Request Line Select | $0028_H$ | See Family Spec |

**Table 303 Register Overview - ERAY (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ERAY0_SILS | Status Service Request Line Select | $002C_H$ | See Family Spec |
| ERAY0_EIES | Error Service Request Enable Set | $0030_H$ | See Family Spec |
| ERAY0_EIER | Error Service Request Enable Reset | $0034_H$ | See Family Spec |
| ERAY0_SIES | Status Service Request Enable Set | $0038_H$ | See Family Spec |
| ERAY0_SIER | Status Service Request Enable Reset | $003C_H$ | See Family Spec |
| ERAY0_ILE | Service Request Line Enable | $0040_H$ | See Family Spec |
| ERAY0_T0C | Timer 0 Configuration | $0044_H$ | See Family Spec |
| ERAY0_T1C | Timer 1 Configuration | $0048_H$ | See Family Spec |
| ERAY0_STPW1 | Stop Watch Register 1 | $004C_H$ | See Family Spec |
| ERAY0_STPW2 | Stop Watch Register 2 | $0050_H$ | See Family Spec |
| ERAY0_SUCC1 | SUC Configuration Register 1 | $0080_H$ | See Family Spec |
| ERAY0_SUCC2 | SUC Configuration Register 2 | $0084_H$ | See Family Spec |
| ERAY0_SUCC3 | SUC Configuration Register 3 | $0088_H$ | See Family Spec |
| ERAY0_NEMC | NEM Configuration Register | $008C_H$ | See Family Spec |

**Table 303 Register Overview - ERAY (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ERAY0_PRTC1 | PRT Configuration Register 1 | $0090_H$ | See Family Spec |
| ERAY0_PRTC2 | PRT Configuration Register 2 | $0094_H$ | See Family Spec |
| ERAY0_MHDC | MHD Configuration Register | $0098_H$ | See Family Spec |
| ERAY0_GTUC01 | GTU Configuration Register 1 | $00A0_H$ | See Family Spec |
| ERAY0_GTUC02 | GTU Configuration Register 2 | $00A4_H$ | See Family Spec |
| ERAY0_GTUC03 | GTU Configuration Register 3 | $00A8_H$ | See Family Spec |
| ERAY0_GTUC04 | GTU Configuration Register 4 | $00AC_H$ | See Family Spec |
| ERAY0_GTUC05 | GTU Configuration Register 5 | $00B0_H$ | See Family Spec |
| ERAY0_GTUC06 | GTU Configuration Register 6 | $00B4_H$ | See Family Spec |
| ERAY0_GTUC07 | GTU Configuration Register 7 | $00B8_H$ | See Family Spec |
| ERAY0_GTUC08 | GTU Configuration Register 8 | $00BC_H$ | See Family Spec |
| ERAY0_GTUC09 | GTU Configuration Register 9 | $00C0_H$ | See Family Spec |
| ERAY0_GTUC10 | GTU Configuration Register 10 | $00C4_H$ | See Family Spec |
| ERAY0_GTUC11 | GTU Configuration Register 11 | $00C8_H$ | See Family Spec |

**Table 303 Register Overview - ERAY (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ERAY0_CCSV | Communication Controller Status Vector | $0100_H$ | See Family Spec |
| ERAY0_CCEV | Communication Controller Error Vector | $0104_H$ | See Family Spec |
| ERAY0_SCV | Slot Counter Value | $0110_H$ | See Family Spec |
| ERAY0_MTCCV | Macrotick and Cycle Counter Value | $0114_H$ | See Family Spec |
| ERAY0_RCV | Rate Correction Value | $0118_H$ | See Family Spec |
| ERAY0_OCV | Offset Correction Value | $011C_H$ | See Family Spec |
| ERAY0_SFS | SYNC Frame Status | $0120_H$ | See Family Spec |
| ERAY0_SWNIT | Symbol Window and Network Idle Time Status | $0124_H$ | See Family Spec |
| ERAY0_ACS | Aggregated Channel Status | $0128_H$ | See Family Spec |
| ERAY0_ESIDn (n=01-15) | Even Sync ID Symbol Window n | $0130_H$+(n-1)*4 | See Family Spec |
| ERAY0_OSIDn (n=01-15) | Odd Sync ID Symbol Window n | $0170_H$+(n-1)*4 | See Family Spec |
| ERAY0_NMVx (x=1-3) | Network Management Vector x | $01B0_H$+(x-1)*4 | See Family Spec |
| ERAY0_MRC | Message RAM Configuration | $0300_H$ | See Family Spec |
| ERAY0_FRF | FIFO Rejection Filter | $0304_H$ | See Family Spec |

**Table 303 Register Overview - ERAY (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ERAY0_FRFM | FIFO Rejection Filter Mask | $0308_H$ | See Family Spec |
| ERAY0_FCL | FIFO Critical Level | $030C_H$ | See Family Spec |
| ERAY0_MHDS | Message Handler Status | $0310_H$ | See Family Spec |
| ERAY0_LDTS | Last Dynamic Transmit Slot | $0314_H$ | See Family Spec |
| ERAY0_FSR | FIFO Status Register | $0318_H$ | See Family Spec |
| ERAY0_MHDF | Message Handler Constraints Flags | $031C_H$ | See Family Spec |
| ERAY0_TXRQ1 | Transmission Request Register 1 | $0320_H$ | See Family Spec |
| ERAY0_TXRQ2 | Transmission Request Register 2 | $0324_H$ | See Family Spec |
| ERAY0_TXRQ3 | Transmission Request Register 3 | $0328_H$ | See Family Spec |
| ERAY0_TXRQ4 | Transmission Request Register 4 | $032C_H$ | See Family Spec |
| ERAY0_NDAT1 | New Data Register 1 | $0330_H$ | See Family Spec |
| ERAY0_NDAT2 | New Data Register 2 | $0334_H$ | See Family Spec |
| ERAY0_NDAT3 | New Data Register 3 | $0338_H$ | See Family Spec |
| ERAY0_NDAT4 | New Data Register 4 | $033C_H$ | See Family Spec |

**Table 303 Register Overview - ERAY (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ERAY0_MBSC1 | Message Buffer Status Changed 1 | $0340_H$ | See Family Spec |
| ERAY0_MBSC2 | Message Buffer Status Changed 2 | $0344_H$ | See Family Spec |
| ERAY0_MBSC3 | Message Buffer Status Changed 3 | $0348_H$ | See Family Spec |
| ERAY0_MBSC4 | Message Buffer Status Changed 4 | $034C_H$ | See Family Spec |
| ERAY0_NDIC1 | New Data Interrupt Control 1 | $03A8_H$ | See Family Spec |
| ERAY0_NDIC2 | New Data Interrupt Control 2 | $03AC_H$ | See Family Spec |
| ERAY0_NDIC3 | New Data Interrupt Control 3 | $03B0_H$ | See Family Spec |
| ERAY0_NDIC4 | New Data Interrupt Control 4 | $03B4_H$ | See Family Spec |
| ERAY0_MSIC1 | Message Buffer Status Changed Interrupt Control 1 | $03B8_H$ | See Family Spec |
| ERAY0_MSIC2 | Message Buffer Status Changed Interrupt Control 2 | $03BC_H$ | See Family Spec |
| ERAY0_MSIC3 | Message Buffer Status Changed Interrupt Control 3 | $03C0_H$ | See Family Spec |
| ERAY0_MSIC4 | Message Buffer Status Changed Interrupt Control 4 | $03C4_H$ | See Family Spec |
| ERAY0_CREL | Core Release Register | $03F0_H$ | See Family Spec |
| ERAY0_ENDN | Endian Register | $03F4_H$ | See Family Spec |

**Table 303 Register Overview - ERAY (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ERAY0_WRDSn (n=01-64) | Write Data Section n | $0400_H$+(n-1)*4 | See Family Spec |
| ERAY0_WRHS1 | Write Header Section 1 | $0500_H$ | See Family Spec |
| ERAY0_WRHS2 | Write Header Section 2 | $0504_H$ | See Family Spec |
| ERAY0_WRHS3 | Write Header Section 3 | $0508_H$ | See Family Spec |
| ERAY0_IBCM | Input Buffer Command Mask | $0510_H$ | See Family Spec |
| ERAY0_IBCR | Input Buffer Command Request | $0514_H$ | See Family Spec |
| ERAY0_RDDSn (n=01-64) | Read Data Section n | $0600_H$+(n-1)*4 | See Family Spec |
| ERAY0_RDHS1 | Read Header Section 1 | $0700_H$ | See Family Spec |
| ERAY0_RDHS2 | Read Header Section 2 | $0704_H$ | See Family Spec |
| ERAY0_RDHS3 | Read Header Section 3 | $0708_H$ | See Family Spec |
| ERAY0_MBS | Message Buffer Status | $070C_H$ | See Family Spec |
| ERAY0_OBCM | Output Buffer Command Mask | $0710_H$ | See Family Spec |
| ERAY0_OBCR | Output Buffer Command Request | $0714_H$ | See Family Spec |
| ERAY0_OTSS | OCDS Trigger Set Select | $0870_H$ | See Family Spec |

**Table 303 Register Overview - ERAY (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Page Number |
|---|---|---|---|
| ERAY0_OCS | OCDS Control and Status | $08E8_H$ | See Family Spec |
| ERAY0_KRSTCLR | Kernel Reset Status Clear Register | $08EC_H$ | See Family Spec |
| ERAY0_KRST1 | Kernel Reset Register 1 | $08F0_H$ | See Family Spec |
| ERAY0_KRST0 | Kernel Reset Register 0 | $08F4_H$ | See Family Spec |
| ERAY0_ACCEN0 | Access Enable Register 0 | $08FC_H$ | See Family Spec |

## 39.3 TC35x Specific Registers

No deviations from the Family Spec

## 39.4 Connectivity

**Table 304 Connections of ERAY0**

| Interface Signals | connects | | Description |
|---|---|---|---|
| ERAY0:MT | to | CCU:eray_mt | Macrotick-clock from CC (synchronous to fpi clock) |
| | | SCU:E_REQ2(3) | |
| ERAY0:RXDA0 | from | P14.8:IN | Receive Channel A0 |
| ERAY0:RXDA1 | from | P11.9:IN | Receive Channel A1 |
| ERAY0:RXDA2 | from | P02.1:IN | Receive Channel A2 |
| ERAY0:RXDA3 | from | P14.1:IN | Receive Channel A3 |
| ERAY0:RXDB0 | from | P14.7:IN | Receive Channel B0 |
| ERAY0:RXDB1 | from | P11.10:IN | Receive Channel B1 |
| ERAY0:RXDB2 | from | P02.3:IN | Receive Channel B2 |
| ERAY0:RXDB3 | from | P14.1:IN | Receive Channel B3 |
| ERAY0:STPWT(3:0) | from | SCU:E_PDOUT(3:0) | StoP Watch Trigger signal |
| ERAY0:TINT0 | to | CAN0:ttc_ectt(5) | Timer Interrupt 0 (high-active) |
| ERAY0:TINT1 | to | CAN0:ttc_ectt(6) | Timer Interrupt 1 (high-active) |
| ERAY0:TXDA | to | P02.0:ALT(6) | Transmit Channel A |
| | | P11.3:ALT(4) | |
| | | P14.0:ALT(3) | |
| | | P14.10:ALT(6) | |

**Table 304 Connections of ERAY0** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| ERAY0:TXDB | to | P02.2:ALT(6) | Transmit Channel B |
| | | P11.12:ALT(4) | |
| | | P14.0:ALT(4) | |
| | | P14.5:ALT(6) | |
| ERAY0:TXENA | to | P02.4:ALT(6) | Transmit Enable Channel A |
| | | P11.6:ALT(4) | |
| | | P14.9:ALT(6) | |
| ERAY0:TXENB | to | P02.5:ALT(6) | Transmit Enable Channel B |
| | | P11.6:ALT(2) | |
| | | P11.11:ALT(6) | |
| | | P14.6:ALT(6) | |
| | | P14.9:ALT(5) | |
| ERAY0:sleep_n | from | SCU:scu_syst_sleep_n | turn-off request from processor |
| ERAY0:INT0_INT | to | INT:eray0.INT0_INT | E-RAY Service Request 0 |
| ERAY0:INT1_INT | to | INT:eray0.INT1_INT | E-RAY Service Request 1 |
| ERAY0:TINT0_INT | to | INT:eray0.TINT0_INT | E-RAY Timer Interrupt 0 Service Request |
| ERAY0:TINT1_INT | to | INT:eray0.TINT1_INT | E-RAY Timer Interrupt 1 Service Request |
| ERAY0:NDAT0_INT | to | INT:eray0.NDAT0_INT | E-RAY New Data 0 Service Request |
| ERAY0:NDAT1_INT | to | INT:eray0.NDAT1_INT | E-RAY New Data 1 Service Request |
| ERAY0:MBSC0_INT | to | INT:eray0.MBSC0_INT | E-RAY Message Buffer Status Changed 0 Service Request |
| ERAY0:MBSC1_INT | to | INT:eray0.MBSC1_INT | E-RAY Message Buffer Status Changed 1 Service Request |
| ERAY0:OBUSY | to | INT:eray0.OBUSY | E-RAY Output Buffer Busy Service Request |
| ERAY0:IBUSY_INT | to | INT:eray0.IBUSY_INT | E-RAY Input Buffer Busy Service Request |

## 39.5 Revision History

**Table 305 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V3.2.9** | | |
| **Page 2** | Headline completed by "Specific Register Set". | |
| - | No functional changes in the connectivity tables. | |
| **V3.2.10** | | |
| – | No functional change. | |
| **V3.2.11** | | |
| – | No functional change. | |

# 40 Peripheral Sensor Interface (PSI5)

This device doesn’t contain a PSI5.

# 41 Peripheral Sensor Interface with Serial PHY Connection (PSI5-S)

This device doesn’t contain a PSI5.

# 42 Gigabit Ethernet MAC (GETH)

This document describes the GETH Interface specific appendix for the product TC35x.

## 42.1 TC35x Specific IP Configuration

No product specific configuration for GETH

## 42.2 TC35x Specific Register Set

### Register Address Space Table

The address space for the module registers is defined in **Register Address Space Table**.

**Table 306 Register Address Space - GETH**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| GETH | $F001D000_H$ | $F001F0FF_H$ | FPI bus interface |

### Register Overview Table

**Table 307 Register Overview - GETH (ascending Offset Address)**

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_MAC_CONFIGURATION | MAC Configuration Register | $0000_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_EXT_CONFIGURATION | MAC Extended Configuration Register | $0004_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_PACKET_FILTER | MAC Packet Filter Register | $0008_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_WATCHDOG_TIMEOUT | MAC Watchdog Timeout Register | $000C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_VLAN_TAG_CTRL | MAC VLAN Tag Control Register | $0050_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_VLAN_TAG_DATA | MAC VLAN Tag Data Register | $0054_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_VLAN_TAG_FILTER_i (i=0-7) | MAC VLAN Tag Filter i Register | $0054_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_MAC_VLAN_HASH_TABLE | MAC VLAN Hash Table Register | $0058_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_VLAN_INCL | MAC VLAN Tag Inclusion or Replacement Register | $0060_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_VLAN_INCL_Q_i (i=0-3) | MAC VLAN Tag Inclusion or Replacement Register per Queue | $0060_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_INNER_VLAN_INCL_i (i=0-3) | MAC Inner VLAN Tag Inclusion or Replacement Register | $0064_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_Q0_TX_FLOW_CTRL | MAC Queue 0 TX Flow Control Register | $0070_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_RX_FLOW_CTRL | MAC Receive Flow Control Register | $0090_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_RXQ_CTRL4 | MAC Receive Queue Control 4 register | $0094_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_RXQ_CTRL0 | MAC Receive Queue Control 0 Register | $00A0_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_RXQ_CTRL1 | MAC Receive Queue Control 1 Register | $00A4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_RXQ_CTRL2 | MAC Receive Queue Control 2 Register | $00A8_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_INTERRUPT_STATUS | MAC Interrupt Status Register | $00B0_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_INTERRUPT_ENABLE | MAC Interrupt Enable Register | $00B4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_RX_TX_STATUS | MAC Receive Transmit Status Register | $00B8_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_PMT_CONTROL_STATUS | MAC PMT Control and Status Register | $00C0_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_MAC_RWK_PACKET_FILTER | MAC Wake-up Packet Filter Register | $00C4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RWK_FILTER_COMMAND_0 | MAC Wake-up Filter Command 0 Register | $00C4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RWK_FILTER_OFFSET_0 | MAC Wake-up Filter Offset 0 Register | $00C4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RWK_FILTER_CRC_i (i=0-1) | MAC Wake-up Filter CRC i Register | $00C4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RWK_FILTER_BYTE_MASK_i (i=0-3) | MAC Wake-up i Filter Byte Mask register | $00C4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_LPI_CONTROL_STATUS | MAC LPI Control and Status Register | $00D0_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_LPI_TIMERS_CONTROL | MAC LPI Timers Control Register | $00D4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_LPI_ENTRY_TIMER | MAC LPI Entry Timer Register | $00D8_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_1US_TIC_COUNTER | MAC One Microsecond Tic Counter Register | $00DC_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_PHYIF_CONTROL_STATUS | MAC PHY Interface Control and Status Register | $00F8_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_VERSION | MAC Version Register | $0110_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_DEBUG | MAC Debug Register | $0114_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_HW_FEATURE0 | MAC Hardware Feature Register 0 | $011C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_HW_FEATURE1 | MAC Hardware Feature Register 1 | $0120_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_MAC_HW_FEATURE2 | MAC Hardware Feature Register 2 | $0124_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_HW_FEATURE3 | MAC Hardware Feature Register 3 | $0128_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_MDIO_ADDRESS | MAC MDIO Address Register | $0200_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_MDIO_DATA | MAC MDIO Data Register | $0204_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_CSR_SW_CTRL | MAC CSR Software Controls Register | $0230_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_EXT_CFG1 | MAC Extended Configuration Register 1 | $0238_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_ADDRESS0_HIGH | MAC Address 0 High Register | $0300_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_ADDRESS0_LOW | MAC Address 0 Low Register | $0304_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_ADDRESSi_HIGH (i=1-31) | MAC Address i High Register | $0308_H$+(i-1)*8 | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_ADDRESSi_LOW (i=1-31) | MAC Address i Low Register | $030C_H$+(i-1)*8 | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MMC_CONTROL | MMC Control Register | $0700_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MMC_RX_INTERRUPT | MMC Receive Interrupts Register | $0704_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MMC_TX_INTERRUPT | MMC Transmit Interrupts Register | $0708_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MMC_RX_INTERRUPT_MASK | MMC Receive Interrupts Mask Register | $070C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_MMC_TX_INTERRUPT_MASK | MMC Transmit Interrupts Mask Register | $0710_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_OCTET_COUNT_GOOD_BAD | Good And Bad Transmitted Octet Count Register | $0714_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_PACKET_COUNT_GOOD_BAD | Good And Bad Transmitted Packets Count Register | $0718_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_BROADCAST_PACKETS_GOOD | Good Transmitted Broadcast Packets Count Register | $071C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_MULTICAST_PACKETS_GOOD | Good Transmitted Multicast Packets Count Register | $0720_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_64OCTETS_PACKETS_GOOD_BAD | Good And Bad 64 Octets Packets Transmitted Count Register | $0724_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_65TO127OCTETS_PACKETS_GOOD_BAD | Good And Bad 65to127 Octets Packets Transmitted Count Register | $0728_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_128TO255OCTETS_PACKETS_GOOD_BAD | Good And Bad 128to255 Octets Packets Transmitted Count Register | $072C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_256TO511OCTETS_PACKETS_GOOD_BAD | Good And Bad 256to511 Octets Packets Transmitted Count Register | $0730_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_512TO1023OCTETS_PACKETS_GOOD_BAD | Good And Bad 512to1023 Octets Packets Transmitted Count Register | $0734_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_1024TOMAXOCTETS_PACKETS_GOOD_BAD | Good And Bad 1024toMax Octets Packets Transmitted Count Register | $0738_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_UNICAST_PACKETS_GOOD_BAD | Good Transmitted Unicat Packets Count Register | $073C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_MULTICAST_PACKETS_GOOD_BAD | Good And Bad Transmitted Multicast Packets Count Register | $0740_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_TX_BROADCAST_PACKETS_GOOD_BAD | Good And Bad Transmitted Broadcast Packets Count Register | $0744_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_UNDERFLOW_ERROR_PACKETS | Transmitted Underflow Error Packets Count Register | $0748_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_SINGLE_COLLISION_GOOD_PACKETS | Good Transmitted Single Collision Count Register | $074C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_MULTIPLE_COLLISION_GOOD_PACKETS | Transmitted Multiple Collision Count Register | $0750_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_DEFERRED_PACKETS | Transmitted Deferred Packets Count Register | $0754_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_LATE_COLLISION_PACKETS | Transmitted Late Collision Packets Count Register | $0758_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_EXCESSIVE_COLLISION_PACKETS | Transmitted Excessive Collision Packets Count Register | $075C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_CARRIER_ERROR_PACKETS | Transmitted Carrier Error Packets Count Register | $0760_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_OCTET_COUNT_GOOD | Good Transmitted Octet Count Register | $0764_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_PACKET_COUNT_GOOD | Good Transmitted Packet Count Register | $0768_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_EXCESSIVE_DEFERRAL_ERROR | Transmitted Excessive Deferral Error Count Register | $076C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_PAUSE_PACKETS | Transmitted Pause Packets Count Register | $0770_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_VLAN_PACKETS_GOOD | Good Transmitted VLAN Packets Count Register | $0774_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_OSIZE_PACKETS_GOOD | Good Transmitted Osize Packets Count Register | $0778_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_RX_PACKETS_COUNT_GOOD_BAD | Good And Bad Received Packets Count Register | $0780_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_OCTET_COUNT_GOOD_BAD | Good And Bad Received Octet Count Register | $0784_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_OCTET_COUNT_GOOD | Good Received Octet Count Register | $0788_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_BROADCAST_PACKETS_GOOD | Good Received Broadcast Packets Count Register | $078C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_MULTICAST_PACKETS_GOOD | Good Received Multicast Packets Count Register | $0790_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_CRC_ERROR_PACKETS | Received CRC Error Packets Count Register | $0794_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_ALIGNMENT_ERROR_PACKETS | Received Alignment Error Count Register | $0798_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_RUNT_ERROR_PACKETS | Received Runtime Error Count Register | $079C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_JABBER_ERROR_PACKETS | Received Jabber Error Count Register | $07A0_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_UNDERSIZE_PACKETS_GOOD | Good Received Undersized Packets Count Register | $07A4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_OVERSIZE_PACKETS_GOOD | Good Received Oversized Packets Count Register | $07A8_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_64OCTETS_PACKETS_GOOD_BAD | Good And Bad 64 Octets Packets Received Count Register | $07AC_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_65TO127OCTETS_PACKETS_GOOD_BAD | Good And Bad 65to127 Octets Packets Received Count Register | $07B0_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_128TO255OCTETS_PACKETS_GOOD_BAD | Good And Bad 128to255 Octets Packets Received Count Register | $07B4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_RX_256TO511OCTETS_PACKETS_GOOD_BAD | Good And Bad 256to511 Octets Packets Received Count Register | $07B8_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_512TO1023OCTETS_PACKETS_GOOD_BAD | Good And Bad 512to1023 Octets Packets Received Count Register | $07BC_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_1024TOMAXOCTETS_PACKETS_GOOD_BAD | Good And Bad 1024toMax Octets Packets Received Count Register | $07C0_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_UNICAST_PACKETS_GOOD | Good Received Unicat Packets Count Register | $07C4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_LENGTH_ERROR_PACKETS | Received Length Error Packets Count Register | $07C8_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_OUT_OF_RANGE_TYPE_PACKETS | Received Out Of Range Type Count Register | $07CC_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_PAUSE_PACKETS | Received Pause Packets Count Register | $07D0_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_FIFO_OVERFLOW_PACKETS | Received FIFO Overflow Count Register | $07D4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_VLAN_PACKETS_GOOD_BAD | Good And Bad Received VLAN Packets Count Registerv | $07D8_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_WATCHDOG_ERROR_PACKETS | Received Watchdog Error Count Register | $07DC_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_RECEIVE_ERROR_PACKETS | Received Receive Error Count Register | $07E0_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_CONTROL_PACKETS_GOOD | Good Received Control Packets Count Register | $07E4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_TX_LPI_USEC_CNTR | Transmitted LPI Microseconds Count Register | $07EC_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_TX_LPI_TRAN_CNTR | Transmitted LPI Transition Count Register | $07F0_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_LPI_USEC_CNTR | Received Microseconds LPI Count Register | $07F4_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RX_LPI_TRAN_CNTR | Received LPI Transition Count Register | $07F8_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MMC_IPC_RX_INTERRUPT_MASK | MMC IPC Receive Interrupts Mask Register | $0800_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MMC_IPC_RX_INTERRUPT | MMC IPC Receive Interrupts Register | $0808_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXIPV4_GOOD_PACKETS | Good Received RxIPv4 Packets Count Register | $0810_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXIPV4_HEADER_ERROR_PACKETS | Received IPv4 Header Error Packets Count Register | $0814_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXIPV4_NO_PAYLOAD_PACKETS | Received IPv4 No Payload Packets Count Register | $0818_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXIPV4_FRAGMENTED_PACKETS | Received IPv4 Fragmented Packets Count Register | $081C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXIPV4_UDP_CHECKSUM_DISABLED_PACKETS | Received IPv4 UPD Checksum Disabled Packets Count Register | $0820_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXIPV6_GOOD_PACKETS | Good Received RxIPv6 Packets Count Register | $0824_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXIPV6_HEADER_ERROR_PACKETS | Received IPv6 Header Error Packets Count Register | $0828_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXIPV6_NO_PAYLOAD_PACKETS | Received IPv6 No Payload Packets Count Register | $082C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXUDP_GOOD_PACKETS | Good Received UDP Packets Count Register | $0830_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_RXUDP_ERROR_PACKETS | Received UDP Error Packets Count Register | $0834_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXTCP_GOOD_PACKETS | Good Received TCP Packets Count Register | $0838_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXTCP_ERROR_PACKETS | Received TCP Error Packets Count Register | $083C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXICMP_GOOD_PACKETS | Good Received ICMP Packets Count Register | $0840_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXICMP_ERROR_PACKETS | Received ICMP Error Packets Count Register | $0844_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXIPV4_GOOD_OCTETS | Good Received IPV4 Octets Count Register | $0850_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXIPV4_HEADER_ERROR_OCTETS | Received IPV4 Header Error Octets Count Register | $0854_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXIPV4_NO_PAYLOAD_OCTETS | Received IPV4 No Payload Octets Count Register | $0858_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXIPV4_FRAGMENTED_OCTETS | Received IPV4 Fragmented Octets Count Register | $085C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXIPV4_UDP_CHECKSUM_DISABLE_OCTETS | Received IPV4 UPD Checksum Disabled Octets Count Register | $0860_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXIPV6_GOOD_OCTETS | Good Received IPV6 Octets Count Register | $0864_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXIPV6_HEADER_ERROR_OCTETS | Received IPV6 Header Error Octets Count Register | $0868_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXIPV6_NO_PAYLOAD_OCTETS | Received IPV6 No Payload Octets Count Register | $086C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXUDP_GOOD_OCTETS | Good Received UDP Octets Count Register | $0870_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_RXUDP_ERROR_OCTETS | Received UDP Error Octets Count Register | $0874_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXTCP_GOOD_OCTETS | Good Received TCP Octets Count Register | $0878_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXTCP_ERROR_OCTETS | Received TCP Error Octets Count Register | $087C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXICMP_GOOD_OCTETS | Good Received ICMP Octets Count Register | $0880_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_RXICMP_ERROR_OCTETS | Received ICMP Error Octets Count Register | $0884_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_TIMESTAMP_CONTROL | MAC Timestamp Control Register | $0B00_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_SUB_SECOND_INCREMENT | MAC Sub-Second Increment Register | $0B04_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_SYSTEM_TIME_SECONDS | MAC System Time Seconds Register | $0B08_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_SYSTEM_TIME_NANOSECONDS | MAC System Time Nanoseconds Register | $0B0C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_SYSTEM_TIME_SECONDS_UPDATE | MAC System Time Seconds Update Register | $0B10_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_SYSTEM_TIME_NANOSECONDS_UPDATE | MAC System Time Nanoseconds Update Register | $0B14_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_TIMESTAMP_ADDEND | MAC Timestamp Addend Register | $0B18_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_SYSTEM_TIME_HIGHER_WORD_SECONDS | MAC System Time Higher Word Seconds Register | $0B1C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_MAC_TIMESTAMP_STATUS | MAC Timestamp Status Register | $0B20_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_TX_TIMESTAMP_STATUS_NANOSECONDS | MAC Transmit Timestamp Nanoseconds Status Register | $0B30_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_TX_TIMESTAMP_STATUS_SECONDS | MAC Transmit Timestamp Seconds Status Register | $0B34_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_TIMESTAMP_INGRESS_ASYM_CORR | MAC Timestamp Ingress Asymmetry Correction Register | $0B50_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_TIMESTAMP_EGRESS_ASYM_CORR | MAC Timestamp Egress Asymmetry Correction Register | $0B54_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_TIMESTAMP_INGRESS_CORR_NANOSECOND | MAC Timestamp Ingress Correction Nanoseconds Register | $0B58_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_TIMESTAMP_EGRESS_CORR_NANOSECOND | MAC Timestamp Egress Correction Nanoseconds Register | $0B5C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_TIMESTAMP_INGRESS_CORR_SUBNANOSEC | MAC Timestamp Ingress Correction Subnanoseconds Register | $0B60_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_TIMESTAMP_EGRESS_CORR_SUBNANOSEC | MAC Timestamp Egress Correction Subnanoseconds Register | $0B64_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_PPS_CONTROL | MAC PPS Control Register | $0B70_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_PPS0_TARGET_TIME_SECONDS | MAC PPS 0 Target Time Seconds Register | $0B80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_PPS0_TARGET_TIME_NANOSECONDS | MAC PPS 0 Target Time Nanoeconds Register | $0B84_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_MAC_PPS0_INTERVAL | MAC PPS 0 Interval Register | $0B88_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MAC_PPS0_WIDTH | MAC PPS 0 Width Register | $0B8C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_OPERATION_MODE | MTL Operation Mode Register | $0C00_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_INTERRUPT_STATUS | MTL Interrupt Status Register | $0C20_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_RXQ_DMA_MAP0 | MTL Receive Queue and DMA Channel Mapping 0 Register | $0C30_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_TXQ0_OPERATION_MODE | MTL Queue 0 Transmit Operation Mode Register | $0D00_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_TXQ0_UNDERFLOW | MTL Queue 0 Transmit Underflow Counter Register | $0D04_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_TXQ0_DEBUG | MTL Queue 0 Transmit Debug Register | $0D08_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_TXQ0_ETS_STATUS | MTL Queue 0 Transmit Status Register | $0D14_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_TXQ0_QUANTUM_WEIGHT | MTL Queue 0 Transmit Quantum or Weights Register | $0D18_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_Q0_INTERRUPT_CONTROL_STATUS | MTL Queue 0 Interrupt Control Status Register | $0D2C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_RXQ0_OPERATION_MODE | MTL Queue 0 Receive Operation Mode Register | $0D30_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_RXQ0_MISSED_PACKET_OVERFLOW_CNT | MTL Queue 0 Receive Missed Packet and Overflow Counter Register | $0D34_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_RXQ0_DEBUG | MTL Queue 0 Receive Debug Register | $0D38_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_MTL_RXQ0_CONTROL | MTL Queue 0 Receive Control Register | $0D3C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_TXQi_OPERATION_MODE (i=1-3) | MTL Queue i Transmit Operation Mode Register | $0D40_H+(i-1)*40_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_TXQi_UNDERFLOW (i=1-3) | MTL Queue i Transmit Underflow Counter Register | $0D44_H+(i-1)*40_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_TXQi_DEBUG (i=1-3) | MTL Queue i Transmit Debug Register | $0D48_H+(i-1)*40_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_TXQi_ETS_CONTROL (i=1-3) | MTL Queue i Transmit ETS Control Register | $0D50_H+(i-1)*40_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_TXQi_ETS_STATUS (i=1-3) | MTL Queue i Transmit ETS Status Register | $0D54_H+(i-1)*40_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_TXQi_QUANTUM_WEIGHT (i=1-3) | MTL Queue i Transmit Quantum or Weights Register | $0D58_H+(i-1)*40_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_TXQi_SENDSLOPECREDIT (i=1-3) | MTL Queue i Transmit SendSlopeCredit Register | $0D5C_H+(i-1)*40_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_TXQi_HICREDIT (i=1-3) | MTL Queue i Transmit HiCredit Register | $0D60_H+(i-1)*40_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_TXQi_LOCREDIT (i=1-3) | MTL Queue i Transmit LoCredit Register | $0D64_H+(i-1)*40_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_Qi_INTERRUPT_CONTROL_STATUS (i=1-3) | MTL Queue i Interrupt Status Register | $0D6C_H+(i-1)*40_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_RXQi_OPERATION_MODE (i=1-3) | MTL Queue i Receive Operation Mode Register | $0D70_H+(i-1)*40_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_MTL_RXQi_MISSED_PACKET_OVERFLOW_CNT (i=1-3) | MTL Queue i Receive Missed Packet and Overflow Counter Register | $0D74_H$+(i-1)*$40_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_RXQi_DEBUG (i=1-3) | MTL Queue i Receive Debug Register | $0D78_H$+(i-1)*$40_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_MTL_RXQi_CONTROL (i=1-3) | MTL Queue i Receive Control Register | $0D7C_H$+(i-1)*$40_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_MODE | DMA Bus Mode Register | $1000_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_SYSBUS_MODE | DMA System Bus Mode Register | $1004_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_INTERRUPT_STATUS | DMA Interrupt Status Register | $1008_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_DEBUG_STATUS0 | DMA Debug Status 0 Register | $100C_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_DEBUG_STATUS1 | DMA Debug Status 1 Register | $1010_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_CHi_CONTROL (i=0-3) | DMA Channel i Control Register | $1100_H$+i*$80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_CHi_TX_CONTROL (i=0-3) | DMA Channel i Transmit Control Register | $1104_H$+i*$80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_CHi_RX_CONTROL (i=0-3) | DMA Channel i Receive Control Register | $1108_H$+i*$80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_CHi_TXDESC_LIST_ADDRESS (i=0-3) | DMA Channel i Transmit Descriptor List Address Register | $1114_H$+i*$80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_CHi_RXDESC_LIST_ADDRESS (i=0-3) | DMA Channel i Receive Descriptor List Address Register | $111C_H$+i*$80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_DMA_CHi_TXDESC_TAIL_POINTER (i=0-3) | DMA Channel i Transmit Descriptor Tail Pointer Register | $1120_H$+i* $80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_CHi_RXDESC_TAIL_POINTER (i=0-3) | DMA Channel i Recieve Descriptor Tail Pointer Register | $1128_H$+i* $80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_CHi_TXDESC_RING_LENGTH (i=0-3) | DMA Channel i Transmit Descriptor Ring Length Register | $112C_H$+i* $80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_CHi_RXDESC_RING_LENGTH (i=0-3) | DMA Channel i Recieve Descriptor Ring Length Register | $1130_H$+i* $80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_CHi_INTERRUPT_ENABLE (i=0-3) | DMA Channel i Interrupt Enable Register | $1134_H$+i* $80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_CHi_RX_INTERRUPT_WATCHDOG_TIMER (i=0-3) | DMA Channel i Recieve Interrupt Watchdog Timer Register | $1138_H$+i* $80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_CHi_SLOT_FUNCTION_CONTROL_STATUS (i=0-3) | DMA Channel i Slot Function Control and Status Register | $113C_H$+i* $80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_CHi_CURRENT_APP_TXDESC (i=0-3) | DMA Channel i Current Application Transmit Descriptor Register | $1144_H$+i* $80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_CHi_CURRENT_APP_RXDESC (i=0-3) | DMA Channel i Current Application Receive Descriptor Register | $114C_H$+i* $80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_CHi_CURRENT_APP_TXBUFFER (i=0-3) | DMA Channel i Current Application Transmit Buffer Address Register | $1154_H$+i* $80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_DMA_CHi_CURRENT_APP_RXBUFFER (i=0-3) | DMA Channel i Current Application Receive Buffer Address Register | $115C_H$+i*$80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_CHi_STATUS (i=0-3) | DMA Channel i Status Register | $1160_H$+i*$80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_DMA_CHi_MISS_FRAME_CNT (i=0-3) | DMA Channel i Missed Frames Count Register | $1164_H$+i*$80_H$ | U,SV | U,SV,P | Application Reset | See Family Spec |
| GETH_CLC | Clock Control Register | $2000_H$ | SV,U | SV,E,P | Application Reset | See Family Spec |
| GETH_ID | Module Identification Register | $2004_H$ | SV,U | BE | Application Reset | See Family Spec |
| GETH_GPCTL | General Purpose Control Register | $2008_H$ | SV,U | SV,P | Application Reset | See Family Spec |
| GETH_ACCEN0 | Access Enable Register 0 | $200C_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| GETH_ACCEN1 | Access Enable Register 1 | $2010_H$ | U,SV | SV,SE | Application Reset | See Family Spec |
| GETH_KRST0 | Kernel Reset Register 0 | $2014_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| GETH_KRST1 | Kernel Reset Register 1 | $2018_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| GETH_KRSTCLR | Kernel Reset Status Clear Register | $201C_H$ | U,SV | SV,E,P | Application Reset | See Family Spec |
| GETH_ACCEN0Dx (x=0-3) | Access Enable Register 0 for DMAx | $2020_H$+x*8 | U,SV | SV,SE | Application Reset | See Family Spec |

**Table 307 Register Overview - GETH (ascending Offset Address)** (cont'd)

| Short Name | Long Name | Offset Address | Access Mode | | Reset | Page Number |
|---|---|---|---|---|---|---|
| | | | Read | Write | | |
| GETH_ACCEN1Dx (x=0-3) | Access Enable Register 1 for DMAx | $2024_H$+x *8 | U,SV | SV,SE | Application Reset | See Family Spec |
| GETH_SKEWCTL | Skew Control Register | $2040_H$ | SV,U | SV,P | Application Reset | See Family Spec |

## 42.3 TC35x Specific Registers

No deviations from the Family Spec

## 42.4 Connectivity

If for one product no signal is connected to an alternate input, it is connected to GND internally at module entity level. This allows to leave some signals unconnected in the application (i.e. RXER, CRS, COL) and save pins and external connection to GND. The tables below list all the connections of the instances.

**Table 308 Connections of GETH**

| Interface Signals | connects | | Description |
|---|---|---|---|
| GETH:COLA | from | P11.15:IN | Collision MII |
| GETH:CRSA | from | P11.14:IN | Carrier Sense MII |
| GETH:CRSB | from | P11.11:IN | Carrier Sense MII |
| GETH:CRSDVA | from | P11.11:IN | Carrier Sense / Data Valid combi-signal for RMII |
| GETH:CRSDVB | from | P11.14:IN | Carrier Sense / Data Valid combi-signal for RMII |
| GETH:GREFCLK | from | TC35x:P11.5 | Gigabit Reference Clock input for RGMII (125 MHz high precission) |
| GETH:MDC | to | P02.8:ALT(6) | MDIO clock |
| | | P12.0:ALT(6) | |
| | | P21.2:ALT(5) | |
| GETH:MDIO | to | P00.0:HWOUT(0) | MDIO Output |
| | | P12.1:HWOUT(0) | |
| | | P21.3:HWOUT(0) | |
| GETH:MDIOA | from | P00.0:IN | MDIO Input |
| GETH:MDIOC | from | P12.1:IN | MDIO Input |
| GETH:MDIOD | from | P21.3:IN | MDIO Input |
| GETH:PPS | to | P14.4:ALT(6) | Pulse Per Second |
| GETH:RCTLA | from | P11.11:IN | Receive Control for RGMII |
| | | TC35x:P11.11 | |
| GETH:REFCLKA | from | P11.12:IN | Reference Clock input for RMII (50 MHz) |

**Table 308 Connections of GETH** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| GETH:RXCLKA | from | P11.12:IN | Receive Clock MII and RGMII |
| | | TC35x:P11.12 | |
| GETH:RXCLKB | from | P11.4:IN | Receive Clock MII and RGMII |
| GETH:RXCLKC | from | P12.0:IN | Receive Clock MII and RGMII |
| GETH:RXD0A | from | P11.10:IN | Receive Data 0 MII, RMII and RGMII (RGMII can use RXD0A only) |
| | | TC35x:P11.10 | |
| GETH:RXD1A | from | P11.9:IN | Receive Data 1 MII, RMII and RGMII (RGMII can use RXD1A only) |
| | | TC35x:P11.9 | |
| GETH:RXD2A | from | P11.8:IN | Receive Data 2 MII and RGMII (RGMII can use RXD2A only) |
| | | TC35x:P11.8 | |
| GETH:RXD3A | from | P11.7:IN | Receive Data 3 MII and RGMII (RGMII can use RXD3A only) |
| | | TC35x:P11.7 | |
| GETH:RXDVA | from | P11.11:IN | Receive Data Valid MII |
| GETH:RXDVB | from | P11.14:IN | Receive Data Valid MII |
| GETH:RXERA | from | P11.13:IN | Receive Error MII |
| GETH:RXERB | from | P21.7:IN | Receive Error MII |
| GETH:RXERC | from | P10.0:IN | Receive Error MII |
| GETH:TRIGO(9:0) | to | INT:eth.TRIGO(9:0) | Ethernet Service Request |
| GETH:TCTL | to | TC35x:P11.6 | Transmit Control for RGMII |
| GETH:TXCLK | to | TC35x:P11.4 | Transmit Clock Output for MII and RGMII |
| GETH:TXCLKA | from | P11.5:IN | Transmit Clock Input for MII |
| GETH:TXCLKB | from | P11.12:IN | Transmit Clock Input for MII |
| GETH:TXD(0) | to | TC35x:P11.3 | Transmit Data |
| GETH:TXD(1) | to | TC35x:P11.2 | Transmit Data |
| GETH:TXD(2) | to | TC35x:P11.1 | Transmit Data |
| GETH:TXD(3) | to | TC35x:P11.0 | Transmit Data |
| GETH:TXEN | to | TC35x:P11.6 | Transmit Enable MII and RMII |
| GETH:TXER | to | P11.4:ALT(6) | Transmit Error MII |

## 42.5 DMA Burst Lengths Limitations by the System

Not all burst lengths of the IP are supported by the system.

The GETH kernel IP supports various burst length of 1 up to 32 beats as defined in DMA_CHi_TX_CONTROL.TxPBL and GETH_DMA_CHi_RX_CONTROL.RxPBL. They can be multiplied by 8 by setting DMA_CH(#i)_Control.PBLx8.

Other than specified in the IP only the following burst lengths are supported by the system: SINGLE, INCR4, INCR8. Note that DMA_CH(#i)_Control.PBLx8 must not be set with PBL values higher than 1.

## 42.6 Buffer and Descriptor Alignment

The GETH is implemented as a 32 bit peripheral. Nevertheless it is connected to 64 bit wide bus (SRI). To make full use of the possible performance of SRI and its bridges, the data buffers and the descriptors need to be aligned to 64 bit addresses.

## 42.7 Embedded FIFOs

The GETH uses two embedded FIFOs. The TX FIFO has a size of 4 kByte, the RX FIFO has a size of 8 kByte.

## 42.8 Master TAG ID

The module has 4 DMA Channels that share one master interface connecting them to the SRI bus. In order to distinguish the 4 DMAs from each other in the system, the master tag ID will dynamically be changed depending on the currently active DMA. **Table 309** details which ID is presented for each DMA.

**Table 309 Master TAG IDs for the Gigabit Ethernet MAC**

| DMA | Master TAG ID |
|---|---|
| DMA0 | $0x28_H$ |
| DMA1 | $0x29_H$ |
| DMA2 | $0x2A_H$ |
| DMA3 | $0x2B_H$ |

## 42.9 Interrupt Service Requests

The module has 10 Service Request Nodes connecting it to the interrupt system. The interrupt request lines are connected to the interrupt controller as shown in **Table 310**.

**Table 310 Service Request Lines of Ethernet MAC**

| IR SRC | GETH IP signal | GETH IP function | Description |
|---|---|---|---|
| SRC_GETH0 | GETH_TRIGO0 | GETH_INTR | DMA functions (sbd_intr_o), this internal line is connected via OR gate to GETH.SR0<br>wake up on LAN (pmt_intr_o), this internal line is connected via OR gate to GETH.SR0<br>wake up on EEE - LPI (lpi_intr_o), this internal line is connected via OR gate to GETH.SR0 |
| SRC_GETH1 | GETH_TRIGO1 | GETH_PPS | Pulse Per Second signal from Precision Time Protocol (ptp_pps_o) |
| SRC_GETH2 | GETH_TRIGO2 | GETH_TX_DMA0 | TX interrupt from DMA 0 (sbd_perch_tx_intr_o[0]) |
| SRC_GETH3 | GETH_TRIGO3 | GETH_TX_DMA1 | TX interrupt from DMA 1(sbd_perch_tx_intr_o[1]) |
| SRC_GETH4 | GETH_TRIGO4 | GETH_TX_DMA2 | TX interrupt from DMA 2(sbd_perch_tx_intr_o[2]) |
| SRC_GETH5 | GETH_TRIGO4 | GETH_TX_DMA3 | TX interrupt from DMA 3(sbd_perch_tx_intr_o[3]) |
| SRC_GETH6 | GETH_TRIGO6 | GETH_RX_DMA0 | RX interrupt from DMA 0 (sbd_perch_rx_intr_o[0]) |
| SRC_GETH7 | GETH_TRIGO7 | GETH_RX_DMA1 | RX interrupt from DMA 1(sbd_perch_rx_intr_o[1]) |
| SRC_GETH8 | GETH_TRIGO8 | GETH_RX_DMA2 | RX interrupt from DMA 2(sbd_perch_rx_intr_o[2]) |
| SRC_GETH9 | GETH_TRIGO9 | GETH_RX_DMA3 | RX interrupt from DMA 3(sbd_perch_rx_intr_o[3]) |

## 42.10 Clocks

The module has multiple clock inputs and outputs connecting it to the system. They are connected to the system as shown in **Table 311**.

If the application wants to use the IP in RGMII mode the appliction has to execute the following steps:

- Prior to the application reset the application must switch on $f_{GETH}$ (by configuring CCUCON5.GETHDIV)
- Attach an external 125 MHz clock to input GREFCLK
- Activate the application reset
- Wait for 10 µs

**Table 311 Clock Lines of Ethernet MAC**

| Clock Line | Connected to | Description |
|---|---|---|
| hclk_i / $f_{AHB}$ | $f_{GETH}$ | AHB master interface clock |
| clk_csr_i / $f_{CSR}$ | $f_{SPB}$ | AHB slave interface clock |
| clk_tx_i | GETH_TXCLK (port pin) | MII transmit clock Input from PHY (10/100 MBit/s) The external PHY or oscillator provides this transmission clock. TXD[3:0] is synchronous to this clock. This is 25 MHz in 100 Mbps mode and 2.5 MHz in 10 Mbps mode. Needs a clock input during SW reset (GETH_BUS_MODE.SWR) |
| clk_gref_i | GETH_GREFCLK (port pin) | RGMII transmit clock **Reference Input** from PHY (1000 MBit/s). The external PHY or oscillator provides this clock as reference. TXD[3:0] is not neccessarily synchronous to this clock!<br>This is always 125 MHz in 1000/100/10 MBit/s mode. Needs a clock input during SW reset (GETH_BUS_MODE.SWR) |
| clk_tx_o | GETH_TXCLK (port pin) | RGMII transmission clock **Output** to PHY (1000 MBit/s) . TXD[3:0] is synchronous (DDR) to this 125 MHz clock to the PHY. If RGMII is downgraded to 100 MBit/s it is divided internally to 25 MHz and for 10 MBit/s to 2.5 MHz. In each case it is generated from clk_gref_i by respective division. This division is controlled automatically by evaluation of mac_speed_o[1:0] and phy_intf_sel_i[2:0].<br>TXCLK has the same on chip delay and output pad delay as the TXD signals and TCTL unless clock skew is selected in SKEW. |
| clk_rx_i | GETH_RXCLK (port pin) | Receive clock from Phy. The external PHY provides this receive clock for RGMII, GMII, MII, and RMII interfaces. This clock is 125 MHz in 1 Gbps mode, 25 MHz in 100 Mbps mode, 2.5 MHz in 10 Mbps mode. Needs a clock input during SW reset (GETH_BUS_MODE.SWR) |
| clk_rmii_i | GETH_REFCLK (port pin) | 50-MHz clock used by the RMII from PHY. In 100 MBit/s Mode, it is divided internally to 25 MHz and provided to the internal MII interface. If RMII is downgraded to 10 MBit/s it is divided internally to 2,5 MHz. This division is controlled automatically by evaluation of mac_speed_o[1:0] and phy_intf_sel_i[2:0]. Needs a clock input during SW reset (GETH_BUS_MODE.SWR). |
| clk_ptp_ref_i | $f_{GETH}$ | Reference Clock for the Time Stamp Update Logic |

## 42.11 Revision History

**Table 312 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V1.3.10** | | |
| **Page 22** | Previous versions removed from revision history. | – |
| **Page 18** | Connections table changed (no functional changes). | – |
| **V1.3.11** | | |
| – | No functional change. | – |
| **V1.3.12** | | |
| **Page 21** | $f_{SRI}$ changed to $f_{GETH}$ as connection of clk_ptp_ref_i. | |
| **Page 18** | Updated connection of MII and RGMII in **Connectivity**. | |
| **V1.3.13** | | |
| – | No functional changes. | – |
| **V1.3.14** | | |
| – | No functional changes. | – |
| **V1.3.15** | | |
| – | No functional changes. | – |

# 43 External Bus Unit (EBU)

This device doesn’t contain an EBU module.

# 44 SD- and eMMC Interface (SDMMC)

This device doesn’t contain an SDMMC module.

# 45 Hardware Security Module (HSM)

The HSM is a separate processor subsystem dedicated for security tasks. It is connected as master and slave to the SPB bus.

For security reasons this module is described in a separate documentation. Please contact your Infineon representative for further information.

# 46 Input Output Monitor (IOM)

This document describes the IOM specific appendix for the product TC35x.

## 46.1 TC35x Specific IP Configuration

**Table 313 TC35x specific configuration of IOM**

| Parameter | IOM |
|---|---|
| **Number of FPC channels** | 16 |
| **Number of GTM inputs** | 8 |
| **Number of LAM** | 16 |
| **Number of ECM** | 1 |

## 46.2 TC35x Specific Register Set

### Register Address Space Table

**Table 314 Register Address Space - IOM**

| Module | Base Address | End Address | Note |
|---|---|---|---|
| IOM | $F0035000_H$ | $F00351FF_H$ | FPI slave interface |

### Register Overview Table

There are no product specific register for this module.

## 46.3 TC35x Specific Registers

There are no product specific register for this module.

## 46.4 Connectivity

This section describes the connectivity of the IOMmodule.

**Table 315 Connections of IOM**

| Interface Signals | connects | | Description |
|---|---|---|---|
| IOM:MON1(0) | from | CCU60:CC62 | Monitor input 1 |
| IOM:MON1(1) | from | CCU60:CC61 | Monitor input 1 |
| IOM:MON1(2) | from | CCU60:CC60 | Monitor input 1 |
| IOM:MON1(3) | from | CCU60:COUT60 | Monitor input 1 |
| IOM:MON1(4) | from | CCU60:COUT61 | Monitor input 1 |
| IOM:MON1(5) | from | CCU60:COUT62 | Monitor input 1 |
| IOM:MON1(6) | from | CCU60:COUT63 | Monitor input 1 |
| IOM:MON1(7) | from | CCU61:COUT63 | Monitor input 1 |
| IOM:MON1(8) | from | CCU61:CC60 | Monitor input 1 |

**Table 315 Connections of IOM** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| IOM:MON1(9) | from | CCU61:CC61 | Monitor input 1 |
| IOM:MON2(0) | from | QSPI0:MRST | Monitor input 2 |
| IOM:MON2(1) | from | QSPI1:MRST | Monitor input 2 |
| IOM:MON2(2) | from | QSPI2:MRST | Monitor input 2 |
| IOM:MON2(3) | from | QSPI3:MRST | Monitor input 2 |
| IOM:MON2(5) | from | CAN00:TXD | Monitor input 2 |
| IOM:MON2(6) | from | CAN01:TXD | Monitor input 2 |
| IOM:MON2(7) | from | CAN02:TXD | Monitor input 2 |
| IOM:MON2(8) | from | CAN03:TXD | Monitor input 2 |
| IOM:MON1(10) | from | CCU61:CC62 | Monitor input 1 |
| IOM:MON1(11) | from | CCU61:COUT60 | Monitor input 1 |
| IOM:MON1(12) | from | CCU61:COUT61 | Monitor input 1 |
| IOM:MON1(13) | from | CCU61:COUT62 | Monitor input 1 |
| IOM:MON2(12) | from | ASCLIN0:ATX<br>ASCLIN0:ATXP | Monitor input 2 |
| IOM:MON2(13) | from | ASCLIN1:ATX<br>ASCLIN1:ATXP | Monitor input 2 |
| IOM:MON2(14) | from | ASCLIN2:ATX<br>ASCLIN2:ATXP | Monitor input 2 |
| IOM:MON2(15) | from | ASCLIN3:ATX<br>ASCLIN3:ATXP | Monitor input 2 |
| IOM:PIN(0) | from | P33.0:IN | GPIO pad input to FPC |
| IOM:PIN(1) | from | P33.1:IN | GPIO pad input to FPC |
| IOM:PIN(2) | from | P33.2:IN | GPIO pad input to FPC |
| IOM:PIN(3) | from | P33.3:IN | GPIO pad input to FPC |
| IOM:PIN(4) | from | P33.4:IN | GPIO pad input to FPC |
| IOM:PIN(5) | from | P33.5:IN | GPIO pad input to FPC |
| IOM:PIN(6) | from | P33.6:IN | GPIO pad input to FPC |
| IOM:PIN(7) | from | P33.7:IN | GPIO pad input to FPC |
| IOM:PIN(8) | from | P33.8:IN | GPIO pad input to FPC |
| IOM:PIN(9) | from | P33.9:IN | GPIO pad input to FPC |
| IOM:PIN(10) | from | P33.10:IN | GPIO pad input to FPC |
| IOM:PIN(11) | from | P33.11:IN | GPIO pad input to FPC |
| IOM:PIN(12) | from | P33.12:IN | GPIO pad input to FPC |
| IOM:PIN(13) | from | P20.12:IN | GPIO pad input to FPC |
| IOM:PIN(14) | from | P20.13:IN | GPIO pad input to FPC |
| IOM:PIN(15) | from | P20.14:IN | GPIO pad input to FPC |
| IOM:REF1(0) | from | CCU60:COUT63 | Reference input 1 |
| IOM:REF1(1) | from | CCU60:COUT62 | Reference input 1 |

**Table 315 Connections of IOM** (cont'd)

| Interface Signals | connects | | Description |
|---|---|---|---|
| IOM:REF1(2) | from | CCU60:COUT61 | Reference input 1 |
| IOM:REF1(3) | from | CCU60:COUT60 | Reference input 1 |
| IOM:REF1(4) | from | CCU60:CC62 | Reference input 1 |
| IOM:REF1(5) | from | CCU60:CC61 | Reference input 1 |
| IOM:REF1(6) | from | CCU60:CC60 | Reference input 1 |
| IOM:REF1(7) | from | CCU61:COUT63 | Reference input 1 |
| IOM:REF1(8) | from | CCU61:COUT62 | Reference input 1 |
| IOM:REF1(9) | from | CCU61:COUT61 | Reference input 1 |
| IOM:REF2(0) | from | QSPI0:MRST | Reference input 2 |
| IOM:REF2(1) | from | QSPI1:MRST | Reference input 2 |
| IOM:REF2(2) | from | QSPI2:MRST | Reference input 2 |
| IOM:REF2(3) | from | QSPI3:MRST | Reference input 2 |
| IOM:REF2(5) | from | CAN00:TXD | Reference input 2 |
| IOM:REF2(6) | from | CAN01:TXD | Reference input 2 |
| IOM:REF2(7) | from | CAN02:TXD | Reference input 2 |
| IOM:REF2(8) | from | CAN03:TXD | Reference input 2 |
| IOM:REF1(10) | from | CCU61:COUT60 | Reference input 1 |
| IOM:REF1(11) | from | CCU61:CC62 | Reference input 1 |
| IOM:REF1(12) | from | CCU61:CC61 | Reference input 1 |
| IOM:REF1(13) | from | CCU61:CC60 | Reference input 1 |
| IOM:REF2(12) | from | ASCLIN0:ATX<br>ASCLIN0:ATXP | Reference input 2 |
| IOM:REF2(13) | from | ASCLIN1:ATX<br>ASCLIN1:ATXP | Reference input 2 |
| IOM:REF2(14) | from | ASCLIN2:ATX<br>ASCLIN2:ATXP | Reference input 2 |
| IOM:REF2(15) | from | ASCLIN3:ATX<br>ASCLIN3:ATXP | Reference input 2 |

## 46.5 Revision History

**Table 316 Revision History**

| Reference | Change to Previous Version | Comment |
|---|---|---|
| **V2.1.15** | | |
| – | No changes. | |

# 47 8-Bit Standby Controller (SCR)

The description of the SCR for all devices is covered by the family specification.

# Revision history

| Document version | Date of release | Description of changes |
|---|---|---|
| V2.0.0 | 2021-02 | • Version comparison table updated.<br>• For further changes see respective revision history of each chapter. The version comparison table below gives an overview. |
| V1.6.0 | 2020-08 | • Version comparison table updated.<br>• For further changes see respective revision history of each chapter. The version comparison table below gives an overview.<br>• Removed device TC3Ax from set of documentation. |
| V1.5.0 | 2020-04 | • Version comparison table updated.<br>• For further changes see respective revision history of each chapter. The version comparison table below gives an overview. |
| V1.4.0 | 2019-12 | • Added TC3Ax appendix as target specification.<br>• Version comparison table updated.<br>• For further changes see respective revision history of each chapter. The version comparison table below gives an overview. |
| V1.3.0 | 2019-09 | • Added additional device TC3Ax to AURIX™ TC3xx set of documentation.<br>• Version comparison table updated.<br>• For further changes see respective revision history of each chapter. The version comparison table below gives an overview. |
| V1.2.0 | 2019-04 | • Added additional device TC3Ex to AURIX™ TC3xx set of documentation.<br>• Version comparison table updated.<br>• For further changes see respective revision history of each chapter. The version comparison table below gives an overview. |
| V1.1.0 | 2019-01 | • Power Management System for Low-End (PMSLE) added.<br>• TC33x and TC33xEXT added.<br>• Changes in connectivity tables.<br>• Version comparison table new.<br>• Detailed Revision History contained in each chapter. |
| V1.0.0 | 2018-08 | • First revision of the User's Manual.<br>• Detailed OCDS information not contained. Available under NDA.<br>• Detailed Revision History contained in each chapter. |

**Version comparison table for AURIX™ TC35x appendix**

| Chapter name | UM V1.6.0 chapter version | UM V2.0.0 chapter version | Content changes |
|---|---|---|---|
| Introduction | V1.0.0 | V1.0.0 | No |
| MEMMAP | V0.1.20 | V0.1.21 | **Yes, see chapter revision history** |
| FW | V1.1.0.1.17 | V1.1.0.1.18 | No functional changes |
| SRI Fabric | V1.1.16 | V1.1.17 | No functional changes |

| Chapter name | UM V1.6.0 chapter version | UM V2.0.0 chapter version | Content changes |
|---|---|---|---|
| • SBCU, EBCU | V1.2.8 | V1.2.9 | No functional changes |
| CPU | V1.1.20 | V1.1.21 | No functional changes |
| NVM Subsystem | V2.0.7 | V2.0.7 | No |
| • DMU | V2.0.11 | V2.0.12 | No functional changes |
| • NVM | V2.0.6 | V2.0.6 | No |
| LMU | V3.1.16 | V3.1.16 | No |
| DAM | n/a | n/a | – |
| SCU | V2.1.26 | V2.1.27 | No functional changes |
| CCU | see SCU | see SCU | – |
| PMS | V2.2.33 | V2.2.34 | No functional changes |
| PMSLE | n/a | n/a | – |
| MTU | V7.4.12 | V7.4.13 | **Yes, see chapter revision history** |
| PORTS | V1.8.21 | V1.8.21 | No |
| SMU | V4.0.22 | V4.0.23 | **Yes, see chapter revision history** |
| INT | V1.2.11 | V1.2.11 | No |
| FCE | V4.2.9 | V4.2.9 | No |
| DMA | V0.1.18 | V0.1.18 | No |
| SPU | V1.1.24 | V1.1.25 | No functional changes |
| SPU2 | n/a | n/a | – |
| BITMGR | n/a | n/a | – |
| SPULCKSTP | V1.2.5 | V1.2.5 | No |
| EMEM | V1.4.4 | V1.4.4 | No |
| RIF | V1.0.40 | V1.0.43 | **Yes, see chapter revision history** |
| HSPDM | V0.7.9 | V0.7.9 | No |
| CIF | n/a | n/a | – |
| STM | V9.2.4 | V9.2.4 | No |
| GTM | n/a | n/a | – |
| CCU6 | V3.0.0 | V3.0.0 | No |
| GPT12 | V3.0.2 | V3.0.2 | No |
| CONVCTRL | V3.0.1 | V3.0.1 | No |
| EVADC | V3.0.4 | V3.0.5 | No functional changes |
| EDSADC | n/a | n/a | – |
| I2C | V2.3.6 | V2.3.6 | No |
| HSSL | n/a | n/a | – |
| • HSCT | n/a | n/a | – |
| ASCLIN | V3.2.8 | V3.2.8 | No |
| QSPI | V3.0.20 | V3.0.20 | No |
| MSC | n/a | n/a | – |

| Chapter name | UM V1.6.0 chapter version | UM V2.0.0 chapter version | Content changes |
|---|---|---|---|
| SENT | n/a | n/a | – |
| MCMCAN | V1.19.13 | V1.19.13 | No |
| E-Ray | V3.2.10 | V3.2.11 | No functional changes |
| PSI5 | n/a | n/a | – |
| PSI5-S | n/a | n/a | – |
| GETH | V1.3.14 | V1.3.15 | No functional changes |
| EBU | n/a | n/a | – |
| SDMMC | n/a | n/a | – |
| HSM | n/a | n/a | – |
| IOM | V2.1.15 | V2.1.15 | No |
| SCR | n/a | n/a | – |

**Edition 2021-02**
**Published by**
**Infineon Technologies AG**
**81726 Munich, Germany**



**Do you have a question about any aspect of this document?**
**Email: erratum@infineon.com**

**Document reference**

**IMPORTANT NOTICE**

The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics ("Beschaffenheitsgarantie").

With respect to any examples, hints or any typical values stated herein and/or any information regarding the application of the product, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

In addition, any information given in this document is subject to customer's compliance with its obligations stated in this document and any applicable legal requirements, norms and standards concerning customer's products and any use of the product of Infineon Technologies in customer's applications.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on technology, delivery terms and conditions and prices, please contact the nearest Infineon Technologies Office (**www.infineon.com**).

**WARNINGS**

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.