XC161

16-Bit Single-Chip Microcontroller with 166SV2 Core

16bit

Microcontrollers

Infineon technologies

Never stop thinking.

Edition 2002-02

Published by Infineon Technologies AG,
St.-Martin-Strasse 53,
D-81541 München, Germany

Attention please!

The information herein is given to describe certain components and shall not be considered as warranted characteristics.

Terms of delivery and rights to technical change reserved.

We hereby disclaim any and all warranties, including but not limited to warranties of non-infringement, regarding circuits, descriptions and charts stated herein.

Infineon Technologies is an approved CECC manufacturer.

Information

For further information on technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies Office in Germany or our Infineon Technologies Representatives worldwide (see address list).

Warnings

Due to technical requirements components may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies Office.

Infineon Technologies Components may only be used in life-support devices or systems with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system, or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body, or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.

XC161

16-Bit Single-Chip Microcontroller with 166SV2 Core

Microcontrollers

Infineon technologies

Never stop thinking.

Revision History:	**2002-02**	V1.0
Previous Version:	---	

Page	**Subjects (major changes since last revision)**

Controller Area Network (CAN): License of Robert Bosch GmbH

We Listen to Your Comments

Any information within this document that you feel is wrong, unclear or missing at all? Your feedback will help us to continuously improve the quality of this document. Please send your proposal (including a reference to this document) to:

mcdocu.comments@infineon.com

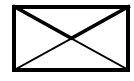


Table of Contents

		Page
1	**On-Chip System Architecture**	1
2	**The Core System Resources**	3
2.1	Central Processing Unit (CPU)	3
2.2	Program Management Unit (PMU)	6
2.3	Data Management Unit (DMU)	6
2.4	Interrupt and PEC Control	6
2.5	External Bus Controller (EBC)	9
3	**XC161 Memory Organization**	11
3.1	Overview	11
3.2	On-chip Program Flash	11
4	**System Control**	13
4.1	Startup Memory and Startup Process	13
4.2	The System Control Unit	13
5	**Debug System**	15
6	**The On-Chip Peripheral System**	16
6.1	Asynchronous/Synchronous Serial Interfaces (ASC0/1)	17
6.2	High Speed Synchronous Serial Channels (SSC0/1)	18
6.3	Serial Data Link Module (SDLM)	19
6.4	The IIC Bus Module	20
6.5	General Purpose Timer (GPT12E) Unit	20
6.6	Capture/Compare Units (CAPCOM1/2)	21
6.7	A/D Converter	23
6.8	Real Time Clock	24
6.9	The TwinCAN Module	25
6.10	Parallel Ports	29
7	**Pin Definitions and Port Functions**	31
7.1	XC161 Logic Symbol	31
7.2	Pin Configuration of XC161	32
8	**Keyword Index**	44

1 On-Chip System Architecture

The on-chip system of the XC161 is built around the synthezisable C166S V2 Core, including the CPU, the Program Management Unit PMU, the Data Management Unit DMU, the Interrupt&PEC controller, the External Bus Controller EBC and part of System Control Unit SCU. Additionally, the on-chip system is characterized by:

- Up to 2K dual port RAM for register sets and system stack
- Up to 4K data RAM, controlled by DMU
- Internal instruction memory block IMB, controlled by PMU, including a program memory interface PMI, the 128KByte Program Flash, up to 2K Program RAM and 8K Start up ROM
- Intelligent on-chip peripheral subsystems
- Automotive specific peripherals (ADC, TwinCAN, SDLM)
- Debug and emulation control blocks

The following figure shows the block diagram of the XC161.

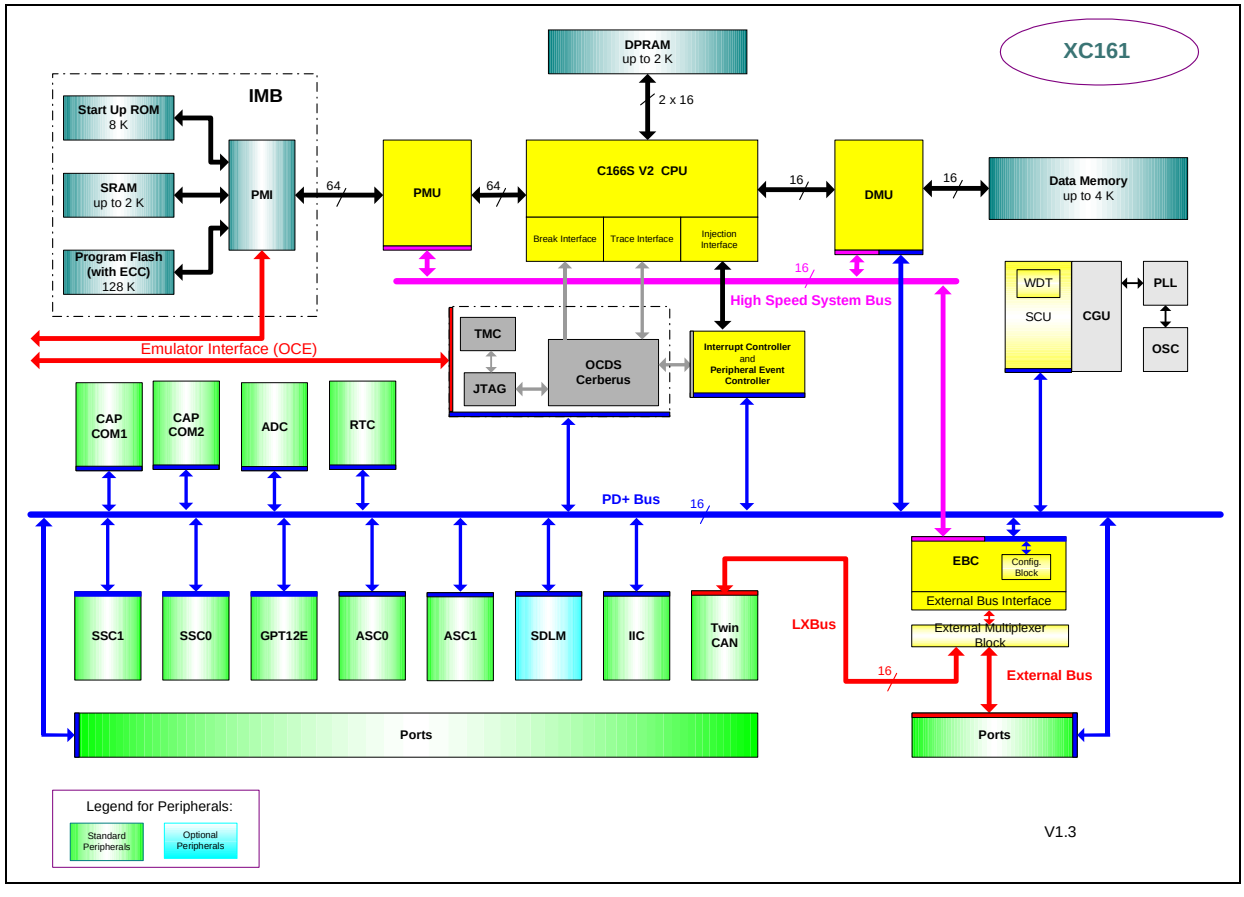


Figure 1 XC161 Block Diagram

Note: The block diagram above shows the fully equipped XC161 in 144-pin package, as described in this document, including the optional SDLM and supporting a separate JTAG and debug interface*.*

The XC161 on-chip system is addtionally characterized by the following system buses and interfaces:

- 16-bit peripheral bus (PD+Bus) with 2 unidirectional data buses, dedicated to the SFR and ESFR space as well as to the new 4K XSFR space
- 16-bit on-chip system bus (LXBus) with 2 unidirectional data buses (similar to the well known XBUS), used for TwinCAN connection
- Additional buses of the core subsystem: 64-bit pipelined two-cycle program memory bus, 16-bit single-cycle dual-ported memory bus and 16-bit single-cycle data memory bus.

2 The Core System Resources

The XC161 controller provides a number of powerful system resources designed within and around the SV2 Core Macro. The combination of the C166S V2 Core and the system resources results in the high performance of the XC161. In the following, overview chapters are provided for all system resources and units.

2.1 Central Processing Unit (CPU)

The C166S V2 CPU represents the third generation of the well known C166 core family. It combines many powerful enhancements with compatibility to the C166 Family. The high performance CPU is characterized by its single clock cycle execution of instructions what doubles the performance at the same CPU frequency (relative to the performance of the C166). Impressive DSP performance and C166 code compatibility enable re-use of existing code with optimized DSP support for critical DSP tasks.

Summary of CPU Features

- Opcode fully upward compatible with C166 Family
- 5-stage execution pipeline
- 2-stage instruction fetch pipeline with FIFO for instruction pre-fetching
- Pipeline with forwarding that controls data dependencies in hardware
- Multiple high bandwidth buses for data and instructions
- Linear address space for code and data (von Neumann architecture)
- C166 Family compatible on-chip special function register area
- Fast multiplication (16-bit x 16-bit) in one CPU clock cycle
- Fast background execution of division (32-bit/16-bit) in 21 CPU clock cycles
- Nearly all instructions executed in one CPU clock cycle
- Enhanced boolean bit manipulation facilities
- Zero cycle jump execution
- Register-based design with multiple variable register banks
- Two additional fast register banks
- General purpose register architecture
- 16 General-purpose registers (GPRs) for byte operands
- 16 General-purpose registers (GPRs) for integer operands
- Overlapping 8-bit and 16-bit registers
- Variable stack with automatic stack overflow/underflow detection
- High performance branch-, call- and loop processing
- Extremely short interrupt response time
- "Fast interrupt" and "Fast context switch" features
- Built-in advanced MAC (Multiply Accumulate) Unit, supporting:
  - Single cycle MAC with zero cycle latency including a 16*16 multiplier plus 40-bit barrel shifter; single clock multiplication is ten times faster than C166 at the same CPU clock

– 40-bit accumulator to handle overflows
– Automatic saturation to 32 bit or rounding included with the MAC instruction
– Fractional numbers supported directly
– One Finite Impulse Response Filter (FIR) tap per cycle with no circular buffer management

Structure of CPU

The new CPU architecture results in high CPU performance, fast and efficient access to different kinds of memories, and proficient peripheral units integration. The following block diagram shows the structure of CPU:

.

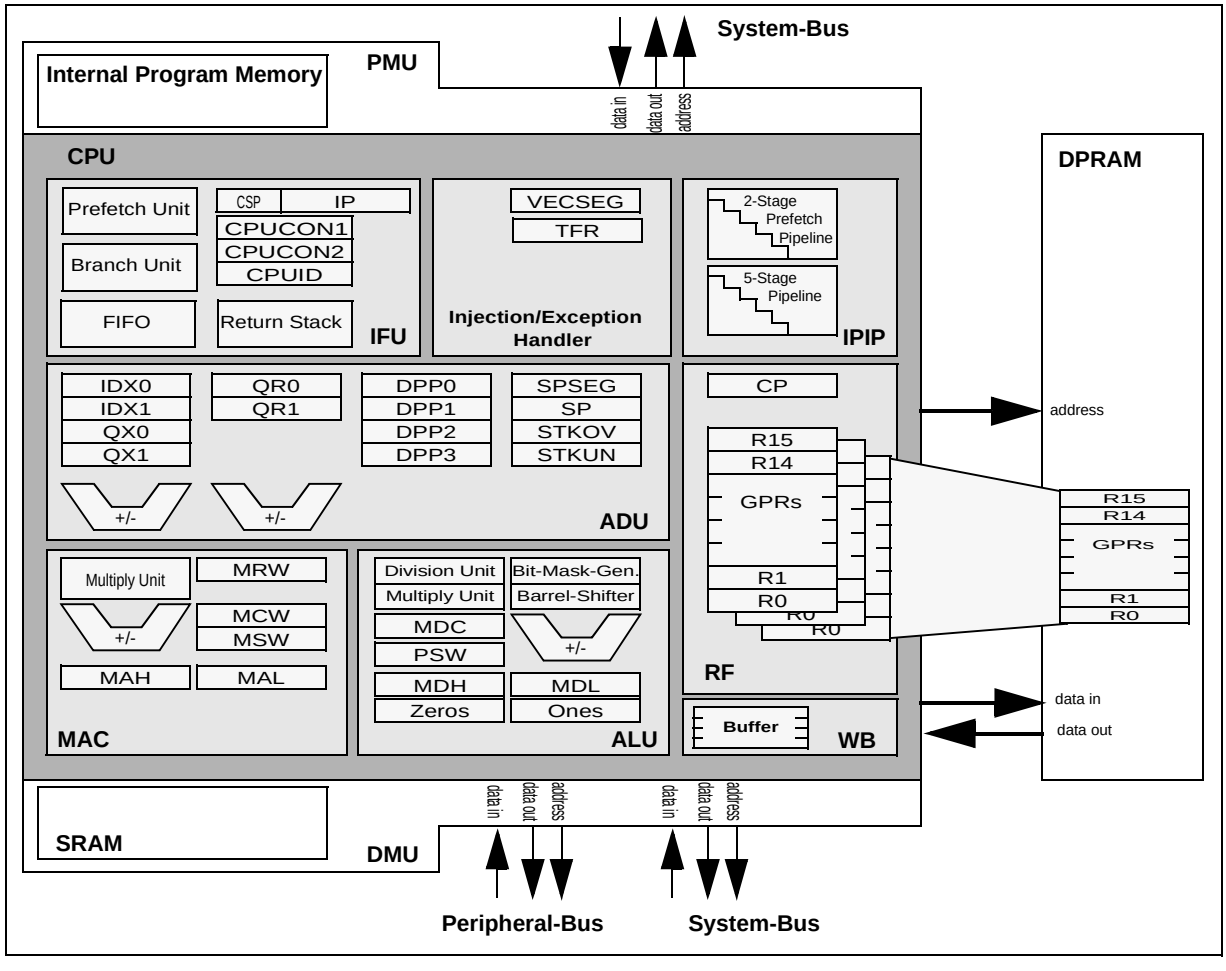


Figure 2 CPU Architecture

The new CPU architecture of the C166SV2 Core results in higher CPU clock frequencies and reduces the number of clock cycles per executed instruction by half, compared to the C166 core. The CPU has eight main units that are listed below. All of these units have been optimized to achieve maximum performance and flexibility.

- High Performance **I**nstruction **F**etch **U**nit (**IFU**)
  - High Bandwidth Fetch Interface
  - Instruction FIFO
  - High Performance Branch-, Call-, and Loop-Processing with instruction flow prediction
  - Return Stack
- Injection/Exception Handler
  - Handling of Interrupt Requests
  - Handling of Hardware Traps
- **I**nstruction **Pip**eline (**IPIP**)
  - Bypassable 2-stage Prefetch Pipeline
  - 5-stage Execution Pipeline
- **A**ddress and **D**ata **U**nit (**ADU**)
  - 16-bit arithmetic unit for address generation
  - DSP address unit with a set of dedicated address- and offset pointers
- **A**rithmetic and **L**ogic **U**nit (**ALU**)
  - 8-bit and 16-bit Arithmetic Unit
  - 16-bit Barrel Shifter
  - Multiplication and Division Unit
  - 8-bit and 16-bit Logic Unit
  - Bit manipulation Unit
- **M**ultiply and **AC**cumulate Unit (**MAC**)
  - 16-bit multiplier with 32-bit result generation[1)]
  - 40-bit Accumulator with 40-bit Barrel Shifter
  - Repeat Control Unit
- **R**egister **F**ile (**RF**)
  - 5-port Register File with three independent register banks
- **W**rite **B**ack **B**uffer (**WB**)
  - 3-entries buffer

1) The same hardware-multiplier is used in the ALU and in the MAC Unit.

2.2 Program Management Unit (PMU)

The program management unit has the task to access the internal program memory block (IMB) and to control the instruction fetch from external memory. Furthermore, the data access from the CPU targeting memory locations in the IMB has to be supported. This type of access is initiated by the CPU (data side) and passes through the data management unit DMU.

The interface between the PMU and the CPU delivers the instruction words (64 bit wide), which are requested by the CPU. The PMU decides whether the requested instruction word has to be fetched from the IMB or from external memory.

2.3 Data Management Unit (DMU)

The data management unit DMU controls the 16 bit wide data flow in the system. Data accesses by the CPU are distributed to the appropriate buses according to the defined address map.

The data can be transferred to/from the CPU from/to the peripherals on the PD+Bus, the peripherals on the LXBus (controlled by the external bus controller EBC), the internal memory block IMB (via PMU), the external world via the EBC or the fast DSRAM block.

Furthermore, the DMU enables the PMU to access the EBC for external instruction fetches, while the CPU can do data accesses internally on the PD+Bus or to the IMB.

The fast data SRAM block located on DMU side can only be accessed by data moves from the CPU and not for instruction fetches by the PMU. As a result, this memory block can be used for very fast data storage. The data access to the IMB takes more time, because the data has to pass through the DMU, over the internal system bus (data part), through the PMU to the IMB.

2.4 Interrupt and PEC Control

The Interrupt&Exception control system is responsible for managing all system and core exceptions. Four kinds of exceptions are differenciated:

- Interrupts generated by the Interrupt controller (ITC)
- DMA transfers issued from the Peripheral Event Controller (PEC)
- Software Traps caused by the TRAP instruction
- Hardware Traps issued by faults or specific system states

The Interrupt&Exception control system of XC161 is especially characterized by the following extensions compared to the standard C16x interrupt structure:

- Up to 74 interrupt sources on 16 priority levels and 8 group levels
- 8 PEC channels with 24-bit source and destination pointers
- PEC source pointers and PEC destination pointers can be simultaneously modified, supporting memory to memory transfer
- Independent programmable PEC transfer request level and "End of PEC" interrupt

Interrupt and PEC System Structure

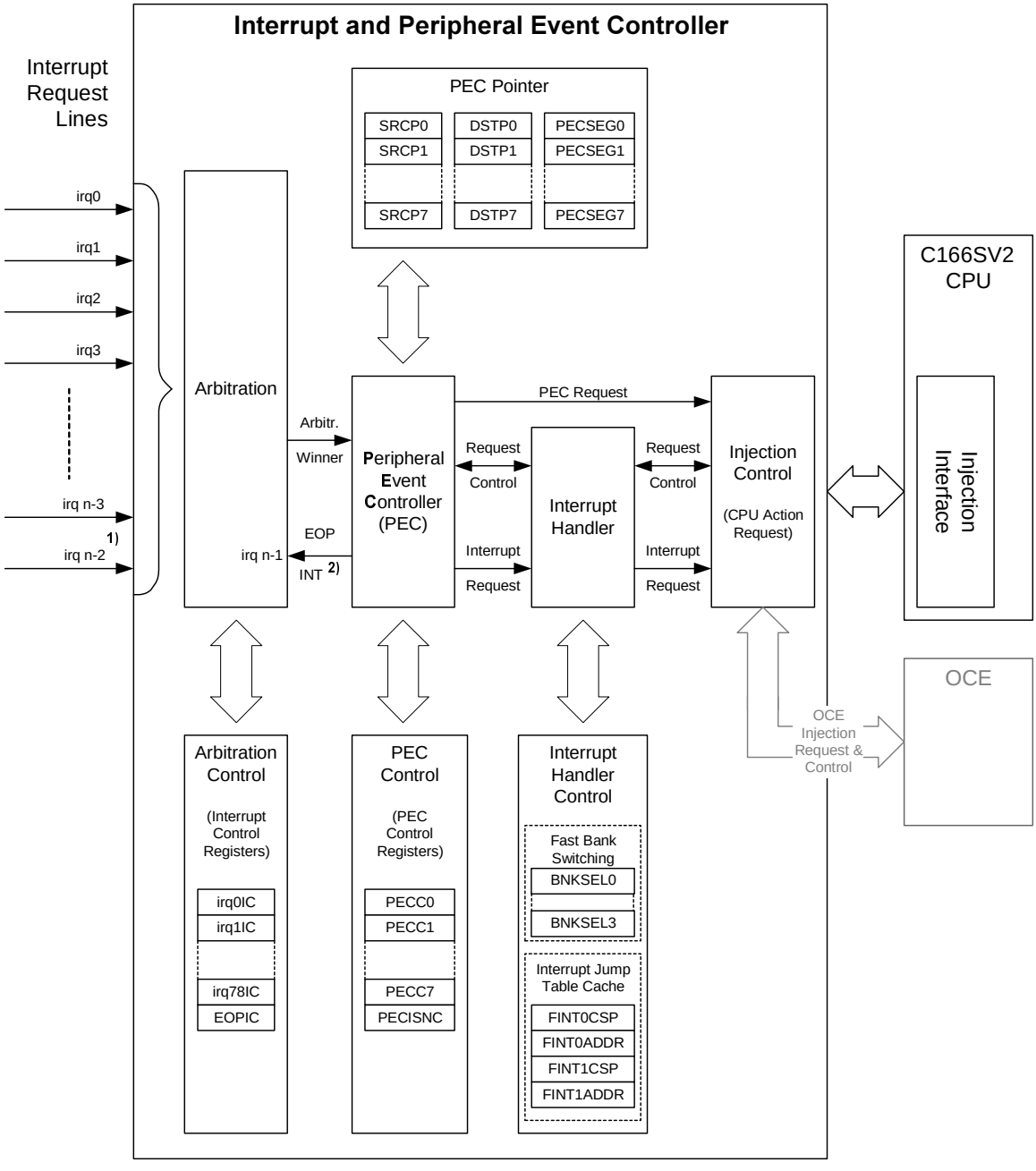
The XC161 provides up to 74 separate interrupt nodes that may be assigned to separate arbitration priority levels with sixteen interrupt priority groups and eight priorities inside each group. In order to support modular and consistent software design techniques, most sources of an interrupt or PEC request are supplied with separate interrupt control registers and interrupt vectors. The control register contains an interrupt request flag, an interrupt enable bit, and an interrupt priority of the associated source. Each source request is activated by one specific event, depending on the selected operating mode of the respective device. In some cases, the multi-source interrupt nodes are incorporated for efficient use of system resources. These nodes can be activated by several source requests and are controlled via interrupt subnode control registers.

XC161 provides a vectored interrupt system. This system reserves specific vector locations in the memory space for the reset, trap, and interrupt service functions. Whenever a request occurs, the CPU branches to the location associated with the respective interrupt source. The reserved vector locations build a jump table in the CPU’s address space.

The arbitration winner is sent to the CPU together with its priority level and action request. The CPU triggers the corresponding action based on the required functionality (normal interrupt, PEC, jump table cache, etc.) of the arbitration winner.

An action request will be accepted by the CPU if the requesting source has a higher priority than the current CPU priority level and if interrupts are globally enabled. If the requesting source has a lower (or equal) interrupt level priority than the current CPU task, it stays pending.

The basic functionality of the interrupt and peripheral event controller can be seen in the following figure:



1) Number of interrupt nodes n

2) End of PEC Interrupt (EOPINT) is connected to interrupt request line irq n-1. Therefore, only n-1 interrupt lines (irq n-2...irq 0) are available for peripheral request handling.

Figure 3 Block Diagram of the Interrupt and Peripheral Event Controller

2.5 External Bus Controller (EBC)

All external memory accesses are performed by a particular on-chip External Bus Controller (EBC). It can be programmed either to Single Chip Mode when no external memory is required, or dynamically (depending on the selected address range, belonging to a chip-select signal) to one of four different external memory access modes, which are as follows:

- 16/17/18/19...24-bit Addresses, 16-bit Data, Demultiplexed
- 16/17/18/19...24-bit Addresses, 16-bit Data, Multiplexed
- 16/17/18/19...24-bit Addresses, 8-bit Data, Multiplexed
- 16/17/18/19...24-bit Addresses, 8-bit Data, Demultiplexed

In the demultiplexed bus modes, addresses are output on PORT1 and data is input/ output on PORT0. In the multiplexed bus modes both addresses and data use PORT0 for input/output. High order address (segment) lines use Port 4. For applications which do not use all address lines for external devices, the external address space can be restricted to 8 MByte, 4 MByte, 2 MByte, 1 MByte, 512 KByte, 256 KByte, 128 KByte or 64 KByte. Up to 5 external CS signals can be generated in order to save external glue logic. Access to very slow memories is supported via a particular ‘Ready’ function. A HOLD/HLDA protocol is available for bus arbitration.

The XC161 External Bus Controller (EBC) allows access to external peripherals and volatile and non-volatile memories. ’External’ means off-chip. However, one module, the TwinCAN module, is connected on-chip to the EBC via the so called LXBus. This module is external from the EBC point of view, but is located on-chip. The LXBus is an internal representation of the ExtBus and allows to access integrated peripherals and modules in the same way as external components. Because some ExtBus control signals are generally configurable, related additional control signals are necessary for the internal LXBus to support its maybe different configuration.

The function of the EBC is controlled via a set of configuration registers. The basic and general behaviour is programmed via the mode-selection registers EBCMOD0 and EBCMOD1.

The XC161-EBC supports 5 external bus chip-select channels and 1 LXBus chip select channel (both together handled as ’external’ chip select channels). With one exception, each of these chip-select signals is programmable via a set of registers. The Function CONtrol register for CSx (FCONCSx) register specifies the external bus/LXBus cycles in terms of address (multiplexed/demultiplexed), data (16-bit/8-bit), chip-select enable, and READY control. The timing of the bus access is controlled by the Timing CONfiguration registers for CSx (TCONCSx), which specify the length of the different access phases. All these parameters are used for accesses within a specific address area that is defined via the corresponding ADDRess SELect register ADDRSELx.

The five register sets (FCONCS1/TCONCS1/ADDRSEL1 through FCONCS5/ TCONCS5/ADDRSEL5) define five independent and programmable "address windows", whereas all external accesses outside these windows are controlled via registers

FCONCS0 and TCONCS0. Chip Select signals CS0...CS4 belong to accesses on external bus, Chip Select CS7 is used for access to the internal TwinCAN module on LXBus.

The external bus timing is related to the reference CLocK OUTput (CLKOUT). All bus signals are generated in relation to the rising edge of this clock. The external bus protocol is compatible with those of the standard C166 Family. However, the external bus timing is improved in terms of wait-state granularity and signal flexibility.

These improvements are configured via an enhanced register set (see above) in comparison to C166 Family. The C16x registers SYSCON and BUSCONx are no longer used. But because the configuration of the external bus controller is done during the initialization of application, only some initialization code has to be adapted for using the new EBC module instead of the C16x external bus controller.

3 XC161 Memory Organization

3.1 Overview

The memory space of the XC161 CPU is configured in a “Von Neumann” architecture. This means that code and data are accessed within the same linear address space. All of the physically separated memory areas, including internal Flash, internal RAM, internal Special Function Register Areas (SFRs and ESFRs), and external memory are mapped into a single common address space.

The XC161 core provides a total addressable memory space of 16 MBytes. This address space is arranged as 256 segments of 64 KBytes each. Each segment is again subdivided into four data pages of 16 KBytes each.

Most internal data memory areas are located in the system segment, segment 0. The upper 4 KBytes of segment 0 ($00'F000_H$...$00'FFFF_H$) hold the Special Function Register Areas (SFR and ESFR) and the DPRAM areas. The 4KBytes range below this area within segment 0 ($00'E000_H$...$00'EFFF_H$) is reserved for the XSFR space.

Data may be stored in any part of the internal memory areas. Code may be stored in any part of the internal memory areas except the SFR blocks, the DPRAM, and Internal data SRAM and internal IO area as these areas may be used for control/data, but not for instructions.

The 64 KByte memory area of segment 191 ($BF'0000_H$...$BF'FFFF_H$) cannot be used to store code and data. It is reserved for startup ROM access.

3.2 On-chip Program Flash

The XC161 incorporates 128 KByte of embedded Flash memory for code or constant data. It is connected to the Core-CPU via the Program Memory Unit and it’s powerful PMI Interface. The Flash module combines the advantages of a very fast read access of 64 bit in one access cycle with protected but simple writing algorithm for programming and erase. The 64-bit read accesses of code are realising the highest CPU performance with fetch of two double word instructions (or four single word instructions) in a single access cycle. Thus, program execution out of internal Flash results in a significant increase of performance compared to execution out of external memory. Due to dynamic error correction, the Flash module provides extremely high read data security for all read accesses.

Summary of Features and Functions

- 128 KByte on-chip Flash module
- Any use for instruction code or constant data
- 64 bit read interface
- Dynamic correction of single bit errors during read access
- Optional read protection for whole Flash - includes protection against write

- Sector architecture
  - Four 8K, one 32K and one 64 KByte sector
  - Each sector separately erasable
  - Sector protection: Sectors can be locked (and unlocked) against erase and programming (two 8K sectors are combined to one lockable 16K sector)
- Comfortable password checking for temporary disable of write or read protection
- Efficient programming operation with 128 byte pages to be written in one step
- Programming time: typical 2 msec per page (128 byte); max.: 5 msec
- Margin check for in-system detection and refreshing of problematic Flash bits
- Fast erase per sector: typical 200 msec; max.: 500 msec
- Endurance = 100; minimum 100 programming/erase cycles per sector or wordline

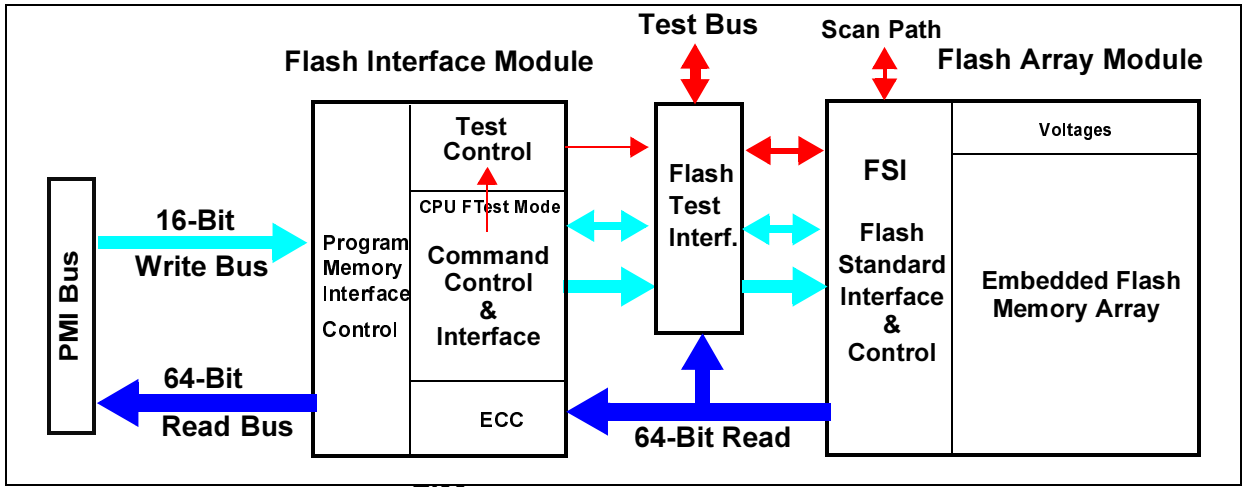


Figure 4 Blocks and Interfaces of Flash Module

4 System Control

4.1 Startup Memory and Startup Process

During the reset period, the first instructions are fetched out of the startup memory in Internal Memory Block IMB which is controlled by the Program Memory Unit PMU. This process is not visible to the customer's application.

The program in the startup memory is used to change the controller's register from default reset state into startup state, i.e. the state that is needed to start user code execution.

Typically, the defaults for the ports, the external bus, the clock system, and the startup vector are set up in this phase. Thereby, the startup address is configurable (normal start, alternate start) via external pins ($\overline{EA}$, $\overline{RD}$, ALE).

At the end of the startup program, the boot routine may be executed, if the system is started in boot mode. If the system is started in user mode, the startup memory is disabled and the CPU executes user code from address $00'0000_H$ if $\overline{EA}$=0, from $C0'0000_H$ if $\overline{EA}$=1 (internal normal start) or from internal address $C1'0000_H$ in case of alternate start.

4.2 The System Control Unit

The XC161 System Control Unit (SCU) summarizes a number of central control tasks and product specific features. The SCU's sub-modules provide the following functions:

- The **Reset Control Block** handles the reset and startup behavior (internal initialization) of the chip. It controls the reset triggers as well as the reset timing.
- The **Central System Control Block** handles all central control tasks like security level selection and clock control. The frequency of the CPU clock is programmable according to the application requirements. Furthermore, the clock generation status is indicated. Depending on the application state, different security levels (like protected and unprotected mode) are supported by the security level control state machine.
  A special register controls the operation of the Instruction Memory Block (IMB).
- The **Power Management Control Block** supports the power management of the XC161, including power reduction modes (idle mode, sleep mode, power down mode) and CPU clock frequency control. The Flexible Peripheral Management permits to disable/enable each peripheral via register SYSCON3.
- The **Watchdog Timer (WDT)** represents one of the fail-safe mechanisms which have been implemented to prevent the controller from malfunctioning. The WDT can detect long term malfunctions. The WDT is always enabled after chip initialization (after internal startup sequence is finished) and is restarted by executing the special SRVWDT instruction. Two modes can be selected for controlling the WDT:
  – **Compatible mode**: the WDT can only be disabled in the time interval until the EINIT (end of initialization) instruction has been executed (compatible mode).

  - **Enhanced WDT mode**: the WDT can be disabled (DISWDT) and enabled (ENWDT) independent from execution of the EINIT instruction in this new WDT mode.
- The **External Interrupt Control Block** handles the eight asynchronous fast external interrupt inputs of the XC161 and triggers internal interrupt requests as programmed. The external interrupt inputs are sampled with every peripheral clock cycle[1)]. The trigger transition (rising, falling or both) can be selected for each interrupt input. The interrupt input lines can be switched to alternate input pins, e.g. in order to monitor serial interfaces while these are disabled (react on an incoming message).
- The **Identification Control Block** supports a set of six identification registers for identification of the most important silicon parameters (chip manufacturer, chip type and its properties). This information can be used for automatic test selection.

Clock Generation Unit CGU

The Clock Generation Unit (CGU) is directly controlled by the clock management and power management functions within the System Control Unit SCU. A programmable on-chip PLL adds high flexibility to clock generation for the XC161 (see SCU description). The EBC and the interupt&PEC controller work with the same system clock as the CPU. The clock for the PD+Bus peripherals is configurable. In case of high-frequency system (CPU) clock, the PD+Bus clock and thus the clock of all peripherals (one exception: TwinCAN) can be divided by 2 or 4. The CGU includes the following functional blocks:

- Main and auxiliary oscillators
- Programmable PLL with pre- and post-divider and with emergency clock function and PLL Unlock interrupt
- Clock drivers
- Oscillator watchdog for direct drive or prescaler mode
  - OWD interrupt for incorrect clock input
  - Switch to PLL base clock in case of clock input failure

1) The peripheral clock is derived from the CPU clock via a divider controlled by bit CPPD.

5 Debug System

The XC161 includes an innovative debug system, which provides comfortable on chip debug support (OCDS) to be controlled directly via dedicated debug interface pins, and which supports high-end emulation devices via an on chip emulator and trace interface for a carrier chip, based on the “New Emulation Strategy NET”.

On Chip Debug Support (OCDS)

The On Chip Debug Support system provides a broad range of debug and emulation features on the product chip (XC161) itself without using the carrier chip. It allows to set breakpoints and to trace memory locations. Typical application of OCDS is to debug the user software running on the XC161 in the customer’s system environment.
The OCDS is controlled by an external debugging device via the debug interface, including the independent JTAG interface and a break interface. The debugger manages the debugging tasks through a set of OCDS registers accessible via the JTAG interface. Addditionally, the OCDS system is controlled by the CPU, e.g. by the monitor program. The OCDS interacts with the core also through an injection interface to allow execution of OCDS generated instructions, and through a break port.

The OCDS provides the following basic features:

- Hardware, software and external pin breakpoints.
- Reaction on break with CPU-Halt, monitor call, data transfer or/and external signal
- Read/write access to whole internal address space
- Single stepping
- Dedicated Debug Interface pins for JTAG interface and break interface
- Injection of arbitrary instructions via JTAG interface
- Fast tracing through transfer to external bus.
- Analysis and status registers

The OCDS system functions are represented and controlled by the debug interface, the OCDS function control and by the debug IO control (called Cerberus) which provides all the functionality necessary to interact with the external debugger.

Emulation Device

The New Emulation Technology (NET) is supported in XC161 to build an Emulation Device (EDEV) with the XC161 mass production chip and the NET Carrier Chip. For connection to the carrier chip, the XC161 provides a broad emulator interface including all important internal buses for tracing, emulation control and status information as well as the external pin (pad) signals. The XC161 mass production chip is mounted on the emulation carrier chip upside down and it’s emulator interface signals are connected across flip chip pads. On the functional and system level, the NET based emulation device is similar to a bondout. The superiority of the NET based emulation device is the high-end emulation functionality with full visibility, and with exactly the same functionality as the mass production chip (emulation system uses the identical silicon).

6 The On-Chip Peripheral System

The on-chip peripherals are controlled either via the PD+Bus or the LXBus. Peripherals are controlled by data written to the proprietary Special Function Registers (SFRs). These SFRs are located either within the standard SFR area ($00'FE00_H$...$00'FFFF_H$) or within the extended ESFR area ($00'F000_H$...$00'F1FF_H$), the internal XSFR area ($00'E000_H$...$00'EFFF_H$), or in case of XBUS peripherals in the (external) memory address space (XLOC). The SFRs serve as control/status and data registers for the peripherals. Interrupt requests are generated by the peripherals based on specific events which occur during their operation.

Each peripheral has an associated set of status flags, indicated in SFR, ESFR or memory locations. Each peripheral can be disabled by power management control.

Byte write operations to word wide SFRs via indirect or direct 16-bit (mem) addressing or byte transfers via the PEC force zeros in the non-addressed byte. Byte write operations via short 8-bit (reg) addressing can only access the low byte of an SFR and force zeros in the high byte. It is therefore recommended, to use the bit field instructions (BFLDL and BFLDH) to write to any number of bits in either byte of an SFR without disturbing the non-addressed byte and the unselected bits.

In the subsequent sections, both the bus systems and their respective interfaces are described. This is followed by short introductions to each peripheral.

The on-chip peripherals attached to the PD+Bus are :

- Asynchronous / Synchronous Serial Interfaces (ASC0 and ASC1)
- High Speed Serial Interfaces (SSC0/1)
- Serial Data Link Module (SDLM), optional
- IIC Bus Module
- General Purpose Timer Unit (GPT12E)
- Capture/Compare Units 1/2 (CAPCOM1/2)
- Analog/Digital Converter (ADC) with 16 or 12 channels
- Real Timer Clock (RTC)
- Parallel Ports

The TwinCAN is the only on-chip peripheral connected to the LXBus.

6.1 Asynchronous/Synchronous Serial Interfaces (ASC0/1)

The Asynchronous/Synchronous Serial Interfaces (ASC0 and ASC1) provide serial communication between the host microcontroller and other microcontrollers, microprocessors or external peripherals.

The ASC0 supports full-duplex asynchronous communication and half-duplex synchronous communication. In asynchronous mode, 8 or 9 bit data transfer, parity generation and the number of stop bits can be selected. Parity, framing and overrun error detection is provided to increase the reliability of data transfers. In synchronous mode, data are transmitted or received synchronous to a shift clock which is generated by the ASC0 internally. In both modes, transmission and reception of data is FIFO-buffered. For multiprocessor communication, a mechanism to distinguish address from data bytes is included. Testing is supported by a loop-back option. A 13-bit baud rate generator provides the ASC0 with a separate serial clock signal, which can be very accurately adjusted by a prescaler implemented as fractional divider. In a special asynchronous mode, the ASC0 supports IrDA data transmissions up to 115.2KBaud with fixed or programmable IrDA pulse width. An autobaud detection unit allows to detect asynchronous data frames with its baudrate and mode with automatic initialization of the baudrate generator and the mode control bits.

Summary of Features

- Full duplex asynchronous operating modes
  - 8- or 9-bit data frames, LSB first
  - Parity bit generation/checking
  - One or two stop bits
  - Baudrate from 2.5 MBaud to 0.6 Baud (@ 40 MHz clock)
  - Multiprocessor mode for automatic address/data byte detection
  - Loop-back capability
  - Support for IrDA data transmission/reception up to max. 115.2 KBaud
- Half-duplex 8-bit synchronous operating mode
  - Baudrate from 5 MBaud to 406.9 Baud (@ 40 MHz clock)
- Autobaud rate detection
- Buffered transmitter/receiver with FIFO support (8 byte per direction)
- Interrupt generation
  - on a transmitter buffer empty condition
  - on a transmit last bit of a frame condition
  - on a receiver buffer full condition
  - on an error condition (frame, parity, overrun error)
  - on the start and the end of an autobaud detection

The ASC1 is a second USART which is functionally identical with the ASC0. The ASC1 is also accessed via the PD+Bus and supports also 5 interrupt vectors. The port line handling is slightly different from that of ASC0.

6.2 High Speed Synchronous Serial Channels (SSC0/1)

The High Speed Synchronous Serial Channels (SSC0 and SSC1) support full-duplex and half-duplex serial synchronous communication with up to 20 MBaud (@ 40 MHz clock). Data width, shift direction, clock polarity and phase are programmable. Both SSCs are identical and they are accessed via the PD+Bus.

A dedicated baud rate generator allows to set up all standard baud rates without subsystem clock tuning. For transmission, reception, and erroneous reception three separate interrupt requests are provided.

The SSCx transmits or receives characters of 2...16 bits length synchronously to a shift clock which can be generated by the SSCx (master mode) or by an external master (slave mode). The SSCx can start shifting with LSB or with MSB. Full SPI functionality is supported. A loop back option is available for testing purposes.

A number of optional hardware error detection capabilities have been included to increase the reliability of data transfers. Transmit and receive error detection supervise the correct handling of the data buffers. Phase and baudrate error detect incorrect serial data.

Summary of Features

- Master and slave mode operation
  - Full-duplex or half-duplex operation
- Flexible data format
  - Programmable number of data bits : 2 to 16 bit
  - Programmable shift direction : LSB or MSB shift first
  - Programmable clock polarity : idle low or high state for the shift clock
  - Programmable clock/data phase : data shift with leading or trailing edge of the shift clock
- Baudrate generation from 20 MBaud to 305.18 Baud (@ 40 MHz clock)
- Comfortable Interrupt generation
  - on a transmitter empty condition
  - on a receiver full condition
  - on an error condition (receive, phase, baudrate, transmit error)
- Three pin interface with flexible SSC pin configuration

6.3 Serial Data Link Module (SDLM)

The optional Serial Data Link Module (SDLM) provides serial communication through a J1850 based multiplexed serial bus via an external J1850 bus transceiver. The module conforms to the SAE Class B J1850 specification for variable pulse width modulation (VPW).

General SDLM Features

- Compliant to the SAE Class B J1850 specification (VPW), Class 2 protocol fully supported
- Variable Pulse Width (VPW) operation at 10.4 KBaud, High Speed 4X operation at 41.6 KBaud
- Programmable Normalization Bit, programmable delay for transceiver interface
- Digital Noise Filter
- Power Down mode with automatic wakeup support upon bus activity
- Single Byte Header and Consolidated Header supported
- CRC generation and checking
- Receive and transmit Block Mode

Data Link Operation Features

- 11 Byte Transmit Buffer and double buffered 11 Byte receive buffer (optional overwrite enable)
- Support for In Frame Response (IFR) types 1, 2 and 3
- Transmit and Receiver Message Buffers configurable for either FIFO or Byte mode
- Advanced Interrupt Handling with 8 separately enabled sources
- Automatic IFR transmission (Types 1 and 2) for 3-Byte consolidated headers
- User configurable clock divider
- Bus status flags (IDLE, EOF, EOD, SOF, Tx and Rx in progress)

6.4 The IIC Bus Module

The integrated IIC Module handles the transmission and reception of frames over the two-line IIC bus in accordance with the IIC Bus specification. The on-chip IIC Module can receive and transmit data using 7-bit or 10-bit addressing and it can operate in slave mode, in master mode or in multi-master mode.

Several physical interfaces (port pins) can be established under software control. Data can be transferred at speeds up to 400 Kbit/sec.

Two interrupt nodes dedicated to the IIC module allow efficient interrupt service and also support operation via PEC transfers.

The port pins associated with the IIC interfaces feature standard open drain drivers.

Features

- Extended buffer allows up to 4 send/receive data bytes to be stored.
- Selectable baud rate generation.
- Support of standard 100 KBaud and extended 400 KBaud data rates.
- Operation in 7-bit addressing mode or 10-bit addressing mode.
- Flexible control via interrupt service routines or by polling.
- Dynamic access to up to 3 physical IIC buses.

6.5 General Purpose Timer (GPT12E) Unit

The GPT12E unit represents a very flexible multifunctional timer/counter structure which may be used for many different time related tasks such as event timing and counting, pulse width and duty cycle measurements, pulse generation, or pulse multiplication. Its expanded functionality includes enhanced incremental interface modes and clock prescaler support.

The GPT12E unit incorporates five 16-bit timers which are organized in two separate modules, GPT1 and GPT2. Each timer in each module may operate independently in a number of different modes, or may be concatenated with another timer of the same module.

Each timer can be configured individually for one of four basic modes of operation, which are Timer, Gated Timer, Counter Mode and Incremental Interface Mode (GPT1 timers). In Timer Mode the input clock for a timer is derived from the internal CPU clock divided by a programmable prescaler, while Counter Mode allows a timer to be clocked in reference to external events (via TxIN).
Pulse width or duty cycle measurement is supported in Gated Timer Mode where the operation of a timer is controlled by the ‘gate’ level on its external input pin TxIN.
In Incremental Interface Mode the GPT1 timers (T2, T3, T4) can be directly connected to the incremental position sensor signals A and B via the respective inputs TxIN and TxEUD. Direction and count signals are internally derived from these two input signals,

so the contents of timer Tx corresponds to the sensor position. The third position sensor signal TOP0 can be connected to an interrupt input.

The count direction (up/down) for each timer is programmable by software or may additionally be altered dynamically by an external signal (TxEUD) to facilitate e.g. position tracking.

The core timers T3 and T6 have output toggle latches (TxOTL) which change their state on each timer over-flow/underflow. The state of these latches may be output on port pins (TxOUT) or may be used internally to clock other timers (T3 for T2, T4, and T6 for T5 or for CAPCOM timers T0 and T1) for measuring long time periods with high resolution.

Various reload or capture functions can be selected to reload timers or capture a timer's contents triggered by an external signal or a selectable transition of toggle latch TxOTL.

The maximum resolution of the timers in module GPT1 is 8 CPU clock cycles. With their maximum resolution of 4 CPU clock cycles the GPT2 timers provide precise event control and time measurement.

Summary of features

- Timer Block GPT1:
  - $f_{PD+Bus}/4$ maximum resolution
  - clock prescaler support
  - 3 independent timers/counters (T2, T3, T4).
  - Timers/counters can be concatenated.
  - 4 operating modes (timer, gated timer, counter, incremental).
  - enhanced incremental interface modes
  - Separate interrupt request lines.

- Timer Block GPT2:
  - $f_{PD+Bus}/2$ maximum resolution
  - clock prescaler support
  - 2 independent timers/counters (T5, T6).
  - Timers/counters can be concatenated.
  - 3 operating modes (timer, gated timer, counter).
  - Extended capture/reload functions via 16-bit Capture/Reload register CAPREL
  - Separate interrupt request lines.

6.6 Capture/Compare Units (CAPCOM1/2)

The CAPCOM units support generation and control of timing sequences on up to 32 channels with a maximum resolution of 16 TCL. The CAPCOM units are typically used to handle high speed I/O tasks such as pulse and waveform generation, pulse width modulation (PMW), Digital to Analog (D/A) conversion, software timing, or time recording

relative to external events. In XC161, all channels of CAPCOM1 and 12 channels of CAPCOM2 are available for such I/O tasks, the remaining channels can be used for internal tasks without pin connection.

Four 16-bit timers (T0/T1 in CAPCOM1, T7/T8 in CAPCOM2) with reload registers provide two independent time bases for the capture/compare register array.

The input clock for the timers is programmable to several prescaled values of the internal system clock, or may be derived from an overflow/underflow of timer T6 in module GPT2. This provides a wide range of variation for the timer period and resolution and allows precise adjustments to the application specific requirements. In addition, external count inputs for CAPCOM timers T0 and T7 allow event scheduling for the capture/compare registers relative to external events.

Both of the two capture/compare register arrays contain 16 dual purpose capture/compare registers, each of which may be individually allocated to either CAPCOM timer T0 or T1 (T7 or T8, respectively), and programmed for capture or compare function. All 16 registers of CAPCOM1 module and 12 registers of CAPCOM2 module have each one port pin associated with it which serves as an input pin for triggering the capture function, or as an output pin to indicate the occurrence of a compare event.

When a capture/compare register has been selected for capture mode, the current contents of the allocated timer will be latched (‘capture’d) into the capture/compare register in response to an external event at the port pin which is associated with this register. In addition, a specific interrupt request for this capture/compare register is generated. Either a positive, a negative, or both a positive and a negative external signal transition at the pin can be selected as the triggering event.

The contents of all registers which have been selected for one of the five compare modes are continuously compared with the contents of the allocated timers. When a match occurs between the timer value and the value in a capture/compare register, specific actions will be taken based on the selected compare mode.

Table 1 Compare Modes in CAPCOM1/2

Compare Modes	Function
Mode 0	Interrupt-only compare mode; several compare interrupts per timer period are possible
Mode 1	Pin toggles on each compare match; several compare events per timer period are possible
Mode 2	Interrupt-only compare mode; only one compare interrupt per timer period is generated
Mode 3	Pin set ‘1’ on match; pin reset ‘0’ on compare time overflow; only one compare event per timer period is generated

6.7 A/D Converter

For analog signal measurement, the XC161 is equipped with a 10-bit A/D Converter with 12 (optional 16) multiplexed input channels including sample and hold functionality. It uses the method of successive approximation. The on chip A/D Converter module combines a powerful control logic and top analog technique in order to fulfill the enhanced requirements of embedded control applications.

An automatic self-calibration adjusts the A/D Converter module to changing temperatures or process variations. The sample time for loading the capacitors and the conversion time is programmable in order to adjust the A/D Converter to the external circuitry. Sample as well as conversion time programming is performed either fully compatible to standard C16x-ADC using the dedicated bitfields in register ADCON, or more advanced with a new control register ADCON1.

Overrun error detection/protection is provided for the conversion result register ADDAT. Either an interrupt request will be generated when the result of a previous conversion has not been read from the result register at the time the next conversion is complete, or the next conversion is suspended in such a case until the previous result has been read.

For applications which require less analog input channels, the remaining channel inputs can be used as digital input port pins.

The A/D converter supports four different conversion modes. In Fixed Channel Single Conversion mode, the analog level on a specified channel is sampled once and converted to a digital result. In Fixed Channel Continuous Conversion mode, the analog level on a specified channel is repeatedly sampled and converted without software intervention. In Auto Scan Single Conversion mode, the analog levels on a prespecified number of channels are sequentially sampled and converted. In the Auto Scan Continuous Conversion mode, the number of prespecified channels is repeatedly sampled and converted. In addition, the conversion of a specific channel can be inserted (injected) into a running sequence without disturbing this sequence. This is called Channel Injection Mode.

The Peripheral Event Controller (PEC) may be used to automatically store the conversion results into a table in memory for later evaluation, without requiring the overhead of entering and exiting interrupt routines for each data transfer.

Summary of Features

- Use of Successive Approximation Method
- Integrated Sample and Hold Functionality
- Analog Input Voltage Range from 0V to 5V
- 12 (optional 16) Analog Input Channels
- 8-bit or 10-Bit Resolution
- Minimum Conversion Time: 2.85 µs @ 10-Bit
- Total Unadjusted Error (TUE): ±1 LSB @ 8-Bit, ± 2 LSB @10-Bit

- Support of Different Conversion Modes
  - Fixed Channel Single Conversion
  - Fixed Channel Continuous Conversion
  - Auto Scan Single Conversion
  - Auto Scan Continuous Conversion
  - Wait for Result Read and Start Next Conversion
  - Channel Injection during Group Conversion
- Flexible Programmable Conversion and Sample Timing Scheme
- DMA (PEC) Support for Result Transfer to Memory Location
- Automatic Self-Calibration to changing temperatures or process variations
- Auto-Power-Down Feature of the A/D Converter
  Power Consumption below 20 mW

6.8 Real Time Clock

The Real Time Clock (RTC) module of XC161 serves for the following purposes:

- System clock to determine the current time and date,
  even during idle mode and power down mode (optionally)
- Cyclic time based interrupt, e.g. to provide a system time tick independent of the CPU frequency without loading the general purpose timers, or to wake up regularly from idle mode.
- 48-bit timer for long term measurements,
  the maximum usable timespan is more than 100 years.
- Alarm interrupt for wakeup on a defined time

The RTC module consists of a chain of 3 divider blocks, a fixed 8:1 divider, the reloadable 16-bit timer T14 and the reloadable 32-bit RTC timer (accessible via 16-bit registers). Both timers count up. An internal interrupt subnode register is provided to indicate up to five different time-triggered interrupts. The RTC is supplied with the dedicated (auxiliary) oscillator frequency or optionally by the main oscillator. The base frequency of the RTC can be programmed via the reload value for timer T14.

6.9 The TwinCAN Module

The TwinCAN module is connected to the LXBus, which is functionally identical to the External Bus. It contains one, optionally two Full-CAN nodes operating independently or exchanging data and remote frames via a gateway function. Transmission and reception of CAN frames is handled in accordance to CAN specification V2.0 part B (active). Each of the two Full-CAN nodes can receive and transmit standard frames with 11-bit identifiers as well as extended frames with 29-bit identifiers.

Both CAN nodes share the TwinCAN module's resources in order to optimize the CAN bus traffic handling and to minimize the CPU load. The flexible combination of Full-CAN functionality and FIFO architecture reduces the efforts to fulfill the real-time requirements of complex embedded control applications. Improved CAN bus monitoring functionality as well as the increased number of message objects permit precise and comfortable CAN bus traffic handling.

Depending on the application, each of the 32 message objects can be individually assigned to one of the two CAN nodes. Gateway functionality allows automatic data exchange between two separate CAN bus systems, which reduces CPU load and improves the real time behavior of the entire system.

The bit timings for both CAN nodes are derived from the peripheral clock (f_{CAN}) and are programmable up to a data rate of 1 MBaud. A pair of receive and transmit pins connect each CAN node to a bus transceiver.

Features

- CAN functionality according to CAN specification V2.0 B active.
- Dedicated control registers are provided for each CAN node.
- A data transfer rate up to 1MBaud is supported.
- Flexible and powerful message transfer control and error handling capabilities are implemented.
- Full-CAN functionality: 32 message objects can be individually
  - assigned to one of the two CAN nodes,
  - configured as transmit or receive object,
  - participate in a 2,4,8,16 or 32 message buffer with FIFO algorithm,
  - setup to handle frames with 11 bit or 29 bit identifiers,
  - provided with programmable acceptance mask register for filtering,
  - monitored via a frame counter,
  - configured to Remote Monitoring Mode.
- Up to eight individually programmable interrupt nodes can be used.
- CAN Analyzer Mode for bus monitoring is implemented.

The following figure 5 shows the functional units of the TwinCAN module. In this figure, node B with its port and interrupt control is optional.

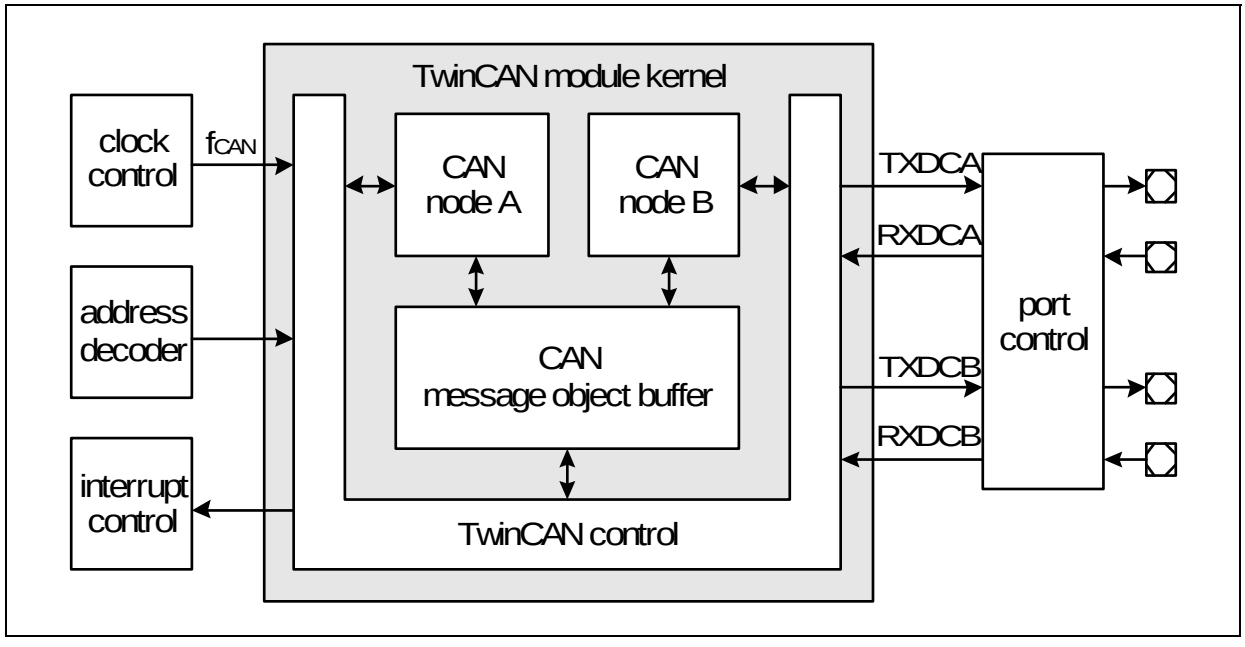


Figure 5 General Block Diagram of the TwinCAN Module

The TwinCAN kernel (figure 6) is split into

- A global control shell, subdivided into the initialization logic, the global control and status logic and the interrupt request compressor.
  - The initialization logic sets up all submodules after power-on or reset. After finishing the initialization of the node control logic and its associated message objects, the respective CAN node is synchronized with the connected CAN bus.
  - The global control and status logic informs the CPU about pending object transmit and receive interrupts and about the recent transfer history.
  - The interrupt request compressor condenses the interrupt requests from 72 sources, belonging to CAN node A and B, to 8 interrupt nodes.
- A message buffer unit, containing the message buffers, the FIFO buffer management, the gateway control logic and a message-based interrupt request generation unit.
  - The message buffer unit stores up to 32 message objects of 8 bytes maximum data length. Each object has an identifier and its own set of control and status bits. After initialization, the message buffer unit can handle reception and transmission of data without CPU supervision.
  - The FIFO buffer management stores the incoming and outgoing messages in a circular buffer and determines the next message to be processed by the CAN controller.

  - The gateway control logic transfers a message from CAN node A to CAN node B or vice versa.
  - The interrupt request generation unit indicates message-specifically the reception or transmission of an object.
- Two (one of them is optional) separate CAN nodes, subdivided into a bit stream processor, a bit timing control unit, an error handling logic, an interrupt request generation unit and a node control logic:
  - The bit stream processor performs data, remote, error and overload frames according to the ISO-DIS 11898 standard. The serial data flow between the CAN bus line, the input/output shift register and the CRC register is controlled as well as the parallel data flow between the I/O shift register and the message buffer unit.
  - The bit timing control unit defines the sampling point in respect to propagation time delays and phase shift errors and performs the resynchronization.
  - The error handling control logic manages the receive and the transmit error counter. According the contents in both timers, the CAN controller is set into an error-active, error-passive or bus-off state.
  - The interrupt request generation unit signals globally the successful end of a message transmit or receive operation, all kinds of transfer problems like bit stuffing errors, format, acknowledge, CRC or bit state errors, every change of the CAN bus warning level or of the bus-off state.
  - The node control logic enables and disables the node specific interrupt sources, enters the CAN analyzer mode and administrates a global frame counter

.

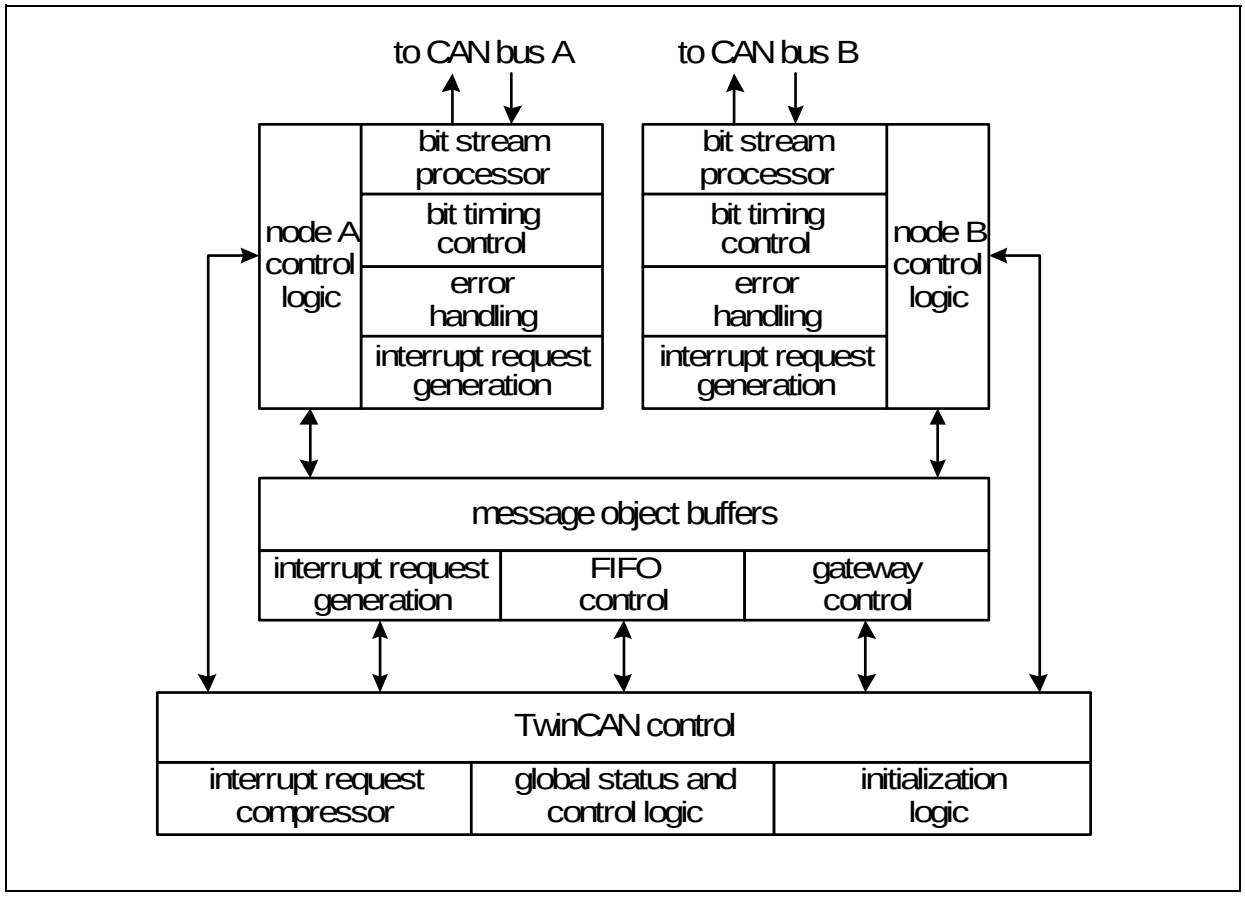


Figure 6 Detailed Block Diagram of the TwinCAN Kernel

6.10 Parallel Ports

The XC161 provides up to 103 I/O lines which are organized into nine input/output ports and one input port. The figure below shows the ports which are available.

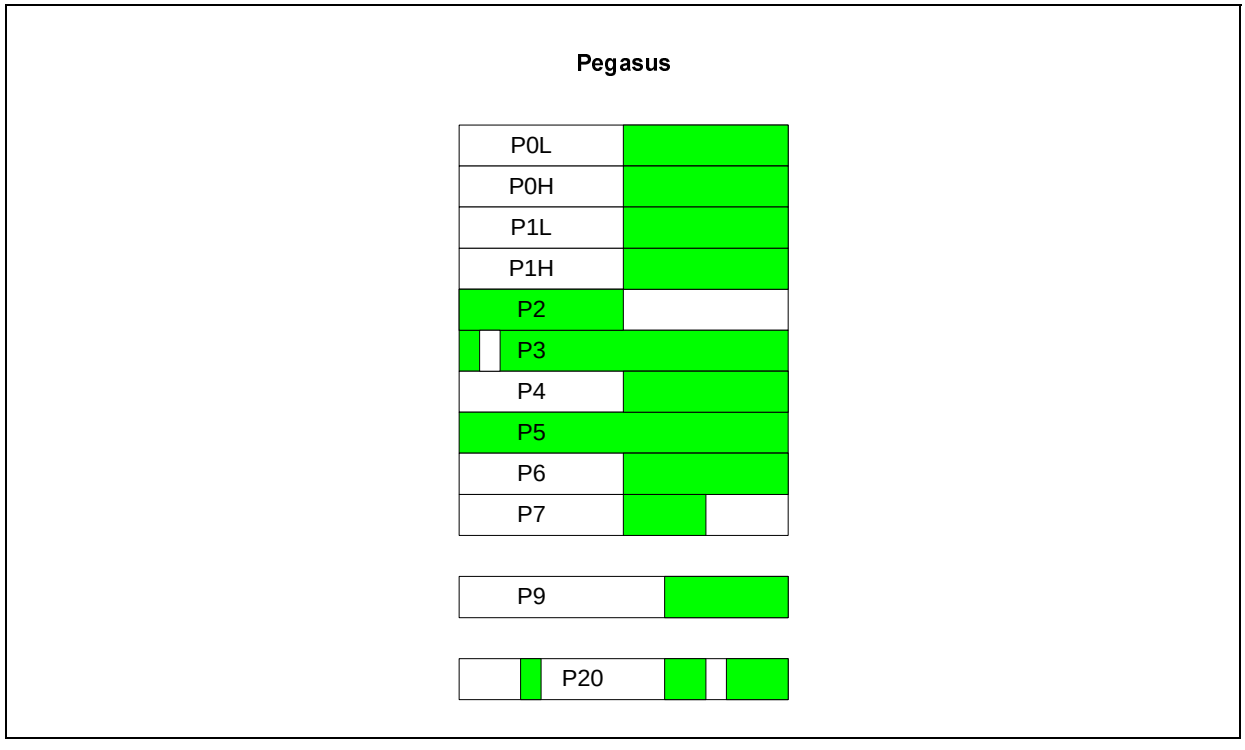


Figure 7 Ports available in XC161

All port lines are bit-addressable, and all input/output lines are individually (bit-wise) programmable as inputs or outputs via direction registers. The I/O ports are true bidirectional ports which are switched to high impedance state when configured as inputs. The output drivers of the I/O ports – Port 2, Port 3, Port 4, Port 6, Port 7, Port 9 and Port 20 - can be configured (pin by pin) for push/pull operation or open-drain operation via the Open Drain Mode Selection Registers (ODPx). During internal reset, all port pins are configured as inputs (exception: pin $\overline{\text{RSTOUT}}$).

The edge characteristics (transition time) and driver characteristics (output current) of the port drivers can be selected via the Port Output Control registers (POCONx).

The input threshold of Port 2, Port 3, Port 4, Port 6, Port 7 and Port 9 is selectable (standard or special), where the special CMOS like input threshold reduces noise sensitivity due to the input hysteresis. The input threshold may be selected individually for each byte of the respective ports.

All port lines have programmable alternate input or output functions associated with them. All port lines that are not used for these alternate functions may be used as general purpose IO lines.

Port 0 and Port 1 may be used as address and data lines when accessing external memory, while Port 4 outputs the additional segment address bits A23/19/17...A16 in systems which access more than 64 KByte of memory.
Port 2, Port 7, Port 9 and parts of Port 1 are associated with the capture inputs or compare outputs of the CAPCOM units.
Port 6 provides optional bus arbitration signals ($\overline{\text{BREQ}}$, $\overline{\text{HLDA}}$, $\overline{\text{HOLD}}$) and chip select signals.
Port 3 includes alternate functions of timers, serial interfaces, the optional bus control signal $\overline{\text{BHE}}$, and the system clock output CLKOUT (or the programmable frequency output FOUT).
Port 5 is used for the analog input channels to the A/D converter or for timer control signals.

Port 9 is also associated to the IIC bus interface signals.

Port 20 combines the dedicated pins to increase flexibility to use them.

For details please refer to the pin description in chapter 7

7 Pin Definitions and Port Functions

7.1 XC161 Logic Symbol

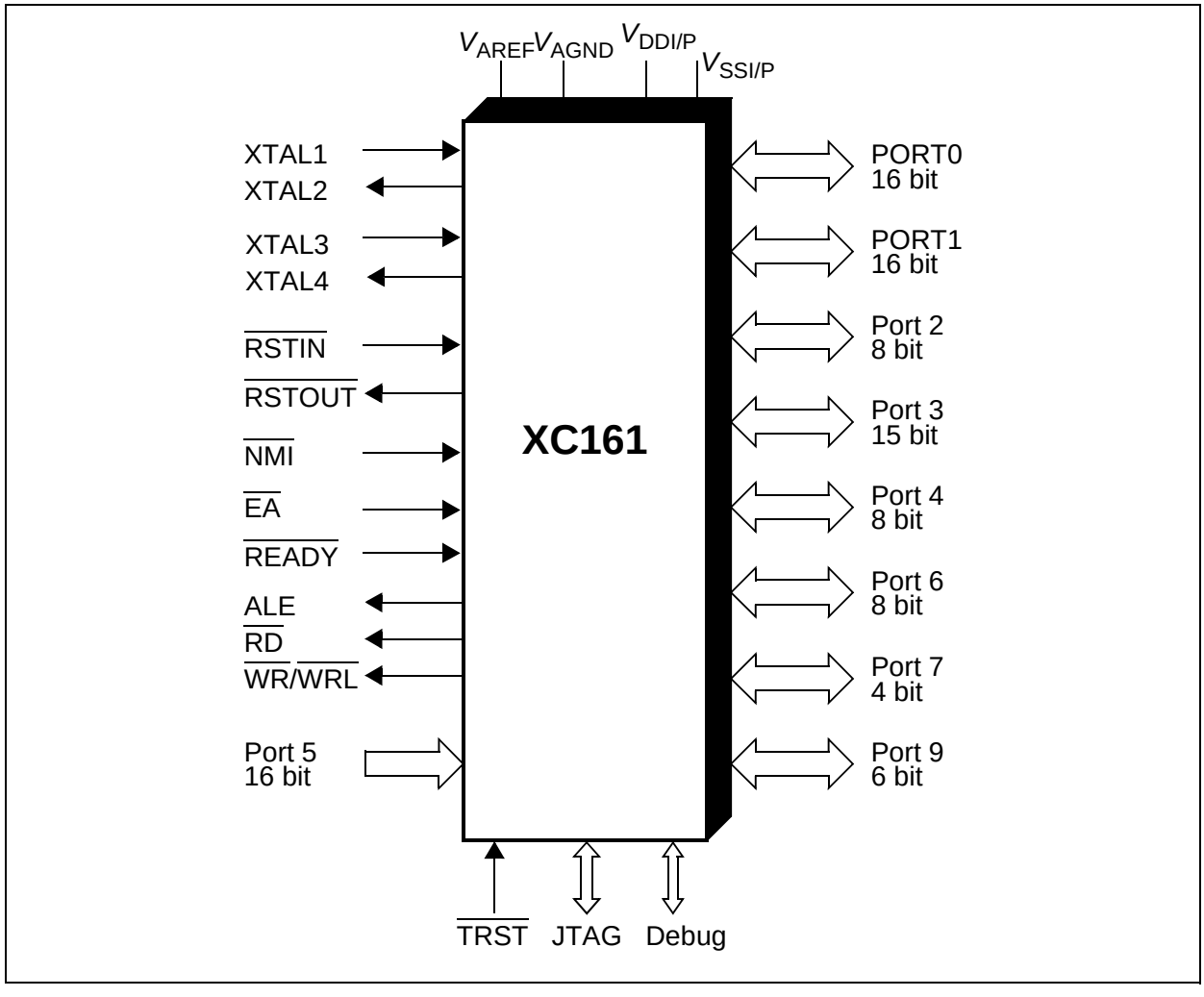


Figure 8 Logic Symbol of XC161 in 144-pin package

7.2 Pin Configuration of XC161

Note: Four ADC pins on port P5 are optional.

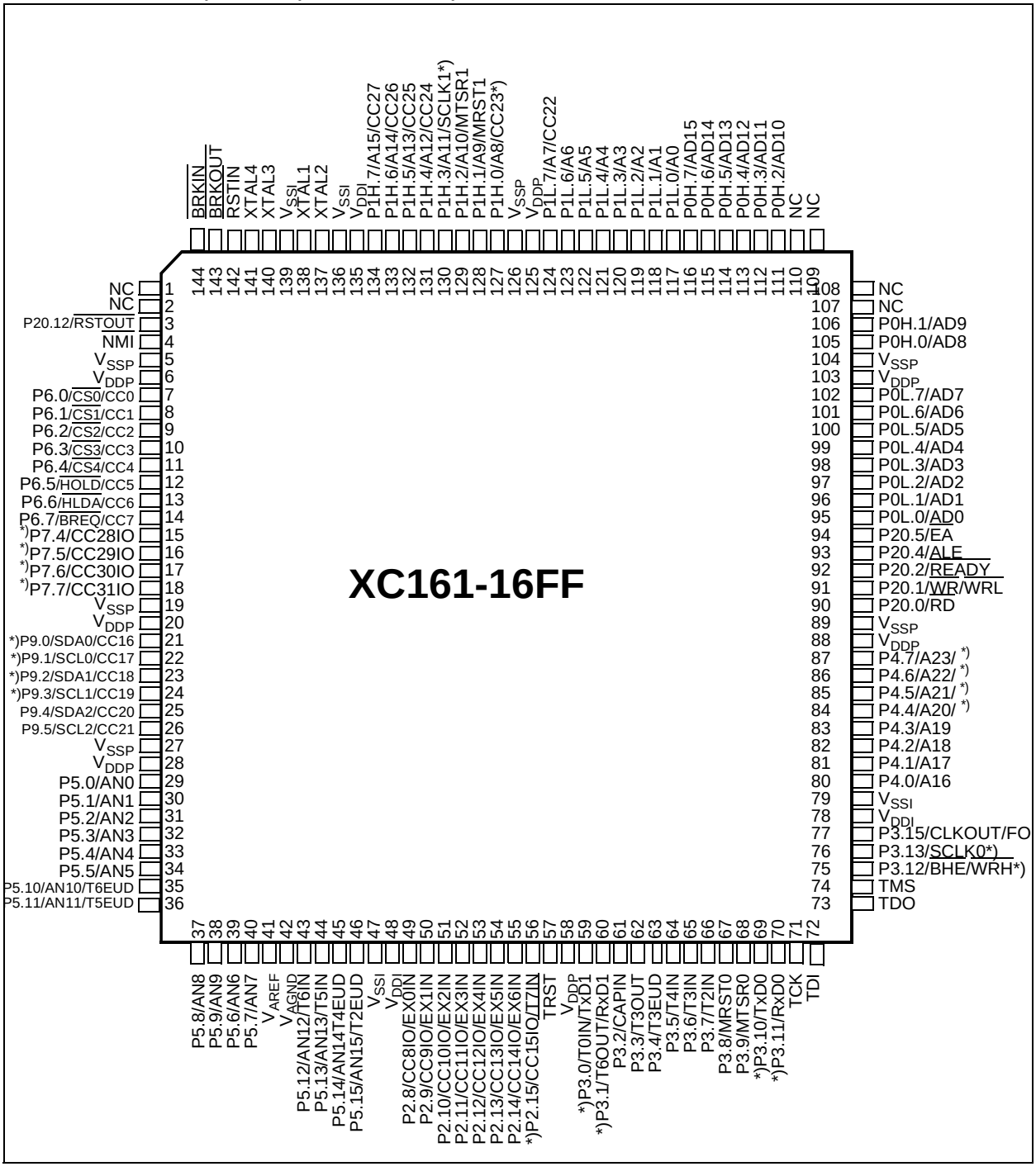


Figure 9 Pin Diagram (Footprint) of XC161 in TQFP-144 Package (top view)

*) Also a CAN/SDLM interface line or/and an external interrupt input can be assigned to this port pin

Table 2 Pin Definitions and Functions

Symbol	Pin Num.	Input Outp.	Function
–	(3)	–	*Note: Please refer to the description of **P20**.*
NMI	4	I	Non-Maskable Interrupt Input. A high to low transition at this pin causes the CPU to vector to the NMI trap routine. When the PWRDN (power down) instruction is executed, the NMI pin must be low in order to force the XC161 into power down mode. If NMI is high, when PWRDN is executed, the part will continue to run in normal mode. If not used, pin NMI should be pulled high externally.
P6		IO	Port 6 is an 8-bit bidirectional I/O port. It is bit-wise programmable for input or output via direction bits. For a pin configured as input, the output driver is put into high-impedance state. Port 6 outputs can be configured as push/pull or open drain drivers. The Port 6 pins also serve for alternate functions:
P6.0	7	O	CS0 Chip Select 0 Output,
		IO	CC0IO CAPCOM1: CC0 Capture Inp./Compare Output
P6.1	8	O	CS1 Chip Select 1 Output,
		IO	CC1IO CAPCOM1: CC1 Capture Inp./Compare Output
P6.2	9	O	CS2 Chip Select 2 Output,
		IO	CC2IO CAPCOM1: CC2 Capture Inp./Compare Output
P6.3	10	O	CS3 Chip Select 3 Output,
		IO	CC3IO CAPCOM1: CC3 Capture Inp./Compare Output
P6.4	11	O	CS4 Chip Select 4 Output,
		IO	CC4IO CAPCOM1: CC4 Capture Inp./Compare Output
P6.5	12	I	HOLD External Master Hold Request Input,
		IO	CC5IO CAPCOM1: CC5 Capture Inp./Compare Output
P6.6	13	I/O	HLDA Hold Acknowledge Output (master mode) or Input (slave mode),
		IO	CC6IO CAPCOM1: CC6 Capture Inp./Compare Output
P6.7	14	O	BREQ Bus Request Output,
		IO	CC7IO CAPCOM1: CC7 Capture Inp./Compare Output

Table 2 Pin Definitions and Functions (cont'd)

Symbol	Pin Num.	Input Outp.	Function
P7		IO	Port 7 is a 4-bit bidirectional I/O port. It is bit-wise programmable for input or output via direction bits. For a pin configured as input, the output driver is put into high-impedance state. Port 7 outputs can be configured as push/pull or open drain drivers. The input threshold of Port 7 is selectable (standard or special). Port 7 pins provide inputs/outputs for CAPCOM2 and serial interface lines.[1)]
P7.4	15	I/O	CC28IO CAPCOM2: CC28 Capture Inp./Compare Outp.,
		I	CAN2_RxD CAN Node 2 Receive Data Input,
		I	EX7IN Fast External Interrupt 7 Input (alternate pin B)
P7.5	16	I/O	CC29IO CAPCOM2: CC29 Capture Inp./Compare Outp.,
		O	CAN2_TxD CAN Node 2 Transmit Data Output,
		I	EX6IN Fast External Interrupt 6 Input (alternate pin B)
P7.6	17	I/O	CC30IO CAPCOM2: CC30 Capture Inp./Compare Outp.,
		I	CAN1_RxD CAN Node 1 Receive Data Input,
		O	SDL_TxD SDLM Transmit Data Output,
		I	EX7IN Fast External Interrupt 7 Input (alternate pin A)
P7.7	18	I/O	CC31IO CAPCOM2: CC31 Capture Inp./Compare Outp.,
		O	CAN1_TxD CAN Node 1 Transmit Data Output,
		I	SDL_RxD SDLM Receive Data Input,
		I	EX6IN Fast External Interrupt 6 Input (alternate pin A)

Table 2 Pin Definitions and Functions (cont'd)

Symbol	Pin Num.	Input Outp.	Function
P9		IO	Port 9 is a 6-bit bidirectional I/O port. It is bit-wise programmable for input or output via direction bits. For a pin configured as input, the output driver is put into high-impedance state. Port 9 outputs can be configured as push/pull or open drain drivers.
			The following Port 9 pins also serve for alternate functions:[1)]
P9.0	21	I/O	CC16IO CAPCOM2: CC16 Capture Inp./Compare Outp.,
		I	CAN2_RxD CAN Node 2 Receive Data Input,
		I/O	SDA0 IIC Bus Data Line 0
P9.1	22	I/O	CC17IO CAPCOM2: CC17 Capture Inp./Compare Outp.,
		O	CAN2_TxD CAN Node 2 Transmit Data Output,
		I/O	SCL0 IIC Bus Clock Line 0
P9.2	23	I/O	CC18IO CAPCOM2: CC18 Capture Inp./Compare Outp.,
		I	CAN1_RxD CAN Node 1 Receive Data Input,
		O	SDL_TxD SDLM Transmit Data Output,
		I/O	SDA1 IIC Bus Data Line 1
P9.3	24	I/O	CC19IO CAPCOM2: CC19 Capture Inp./Compare Outp.,
		O	CAN1_TxD CAN Node 1 Transmit Data Output,
		I	SDL_RxD SDLM Receive Data Input,
		I/O	SCL1 IIC Bus Clock Line 1
P9.4	25	I/O	CC20IO CAPCOM2: CC20 Capture Inp./Compare Outp.,
		I/O	SDA2 IIC Bus Data Line 2
P9.5	26	I/O	CC21IO CAPCOM2: CC21 Capture Inp./Compare Outp.,
		I/O	SCL2 IIC Bus Clock Line 2

Table 2 Pin Definitions and Functions (cont'd)

Symbol	Pin Num.	Input Outp.	Function
P5		I	Port 5 is a 12-bit/16-bit input-only port with Schmitt-Trigger characteristic. The pins of Port 5 also serve as analog input channels for the A/D converter, or they serve as timer inputs:
P5.0	29	I	AN0
P5.1	30	I	AN1
P5.2	31	I	AN2
P5.3	32	I	AN3
P5.4	33	I	AN4
P5.5	34	I	AN5
P5.10	35	I	AN10, T6EUD GPT2 Timer T6 Ext. Up/Down Ctrl. Inp.
P5.11	36	I	AN11, T5EUD GPT2 Timer T5 Ext. Up/Down Ctrl. Inp.
P5.8	37	I	AN8
P5.9	38	I	AN9
P5.6	39	I	AN6
P5.7	40	I	AN7
P5.12	43	I	AN12, T6IN GPT2 Timer T6 Count/Gate Input
P5.13	44	I	AN13, T5IN GPT2 Timer T5 Count/Gate Input
P5.14	45	I	AN14, T4EUD GPT1 Timer T4 Ext. Up/Down Ctrl. Inp.
P5.15	46	I	AN15, T2EUD GPT1 Timer T2 Ext. Up/Down Ctrl. Inp.
			Note: Pins P5.8-11 are not available in all derivatives.

Table 2 Pin Definitions and Functions (cont'd)

Symbol	Pin Num.	Input Outp.	Function
P2		IO	Port 2 is an 8-bit bidirectional I/O port. It is bit-wise programmable for input or output via direction bits. For a pin configured as input, the output driver is put into high-impedance state. Port 2 outputs can be configured as push/pull or open drain drivers. The input threshold of Port 2 is selectable (standard or special). The following Port 2 pins also serve for alternate functions:
P2.8	49	I/O I	CC8IO CAPCOM1: CC8 Capture Inp./Compare Output, EX0IN Fast External Interrupt 0 Input (default pin)
P2.9	50	I/O I	CC9IO CAPCOM1: CC9 Capture Inp./Compare Output, EX1IN Fast External Interrupt 1 Input (default pin)
P2.10	51	I/O I	CC10IO CAPCOM1: CC10 Capture Inp./Compare Outp., EX2IN Fast External Interrupt 2 Input (default pin)
P2.11	52	I/O I	CC11IO CAPCOM1: CC11 Capture Inp./Compare Outp., EX3IN Fast External Interrupt 3 Input (default pin)
P2.12	53	I/O I	CC12IO CAPCOM1: CC12 Capture Inp./Compare Outp., EX4IN Fast External Interrupt 4 Input (default pin)
P2.13	54	I/O I	CC13IO CAPCOM1: CC13 Capture Inp./Compare Outp., EX5IN Fast External Interrupt 5 Input (default pin)
P2.14	55	I/O I	CC14IO CAPCOM1: CC14 Capture Inp./Compare Outp., EX6IN Fast External Interrupt 6 Input (default pin)
P2.15	56	I/O I I	CC15IO CAPCOM1: CC15 Capture Inp./Compare Outp., EX7IN Fast External Interrupt 7 Input (default pin), T7IN CAPCOM2: Timer T7 Count Input
$\overline{\text{TRST}}$	57	I	Test-System Reset Input. A high level at this pin activates the XC161's debug system. *Note: For normal system operation, pin $\overline{\text{TRST}}$ should be held low.*

Table 2 Pin Definitions and Functions (cont'd)

Symbol	Pin Num.	Input Outp.	Function	
P3		IO	Port 3 is a 15-bit bidirectional I/O port. It is bit-wise programmable for input or output via direction bits. For a pin configured as input, the output driver is put into high-impedance state. Port 3 outputs can be configured as push/pull or open drain drivers. The input threshold of Port 3 is selectable (standard or special). The following Port 3 pins also serve for alternate functions:	
P3.0	59	I	T0IN	CAPCOM1 Timer T0 Count Input,
		O	TxD1	ASC1 Clock/Data Output (Async./Sync),
		I	EX1IN	Fast External Interrupt 1 Input (alternate pin B)
P3.1	60	O	T6OUT	GPT2 Timer T6 Toggle Latch Output,
		I/O	RxD1	ASC1 Data Input (Async.) or Inp./Outp. (Sync.),
		I	EX1IN	Fast External Interrupt 1 Input (alternate pin A)
P3.2	61	I	CAPIN	GPT2 Register CAPREL Capture Input
P3.3	62	O	T3OUT	GPT1 Timer T3 Toggle Latch Output
P3.4	63	I	T3EUD	GPT1 Timer T3 External Up/Down Control Input
P3.5	64	I	T4IN	GPT1 Timer T4 Count/Gate/Reload/Capture Inp
P3.6	65	I	T3IN	GPT1 Timer T3 Count/Gate Input
P3.7	66	I	T2IN	GPT1 Timer T2 Count/Gate/Reload/Capture Inp
P3.8	67	I/O	MRST0	SSC0 Master-Receive/Slave-Transmit In/Out.
P3.9	68	I/O	MTSR0	SSC0 Master-Transmit/Slave-Receive Out/In.
P3.10	69	O	TxD0	ASC0 Clock/Data Output (Async./Sync.),
		I	EX2IN	Fast External Interrupt 2 Input (alternate pin B)
P3.11	70	I/O	RxD0	ASC0 Data Input (Async.) or Inp./Outp. (Sync.),
		I	EX2IN	Fast External Interrupt 2 Input (alternate pin A)
P3.12	75	O	$\overline{\text{BHE}}$	External Memory High Byte Enable Signal,
		O	$\overline{\text{WRH}}$	External Memory High Byte Write Strobe,
		I	EX3IN	Fast External Interrupt 3 Input (alternate pin B)
P3.13	76	I/O	SCLK0	SSC0 Master Clock Output / Slave Clock Input.,
		I	EX3IN	Fast External Interrupt 3 Input (alternate pin A)
P3.15	77	O	CLKOUT	System Clock Output (=CPU Clock),
		O	FOUT	Programmable Frequency Output
TCK	71	I	Debug System: JTAG Clock Input	
TDI	72	I	Debug System: JTAG Data In	
TDO	73	O	Debug System: JTAG Data Out	
TMS	74	I	Debug System: JTAG Test Mode Selection	

Table 2 Pin Definitions and Functions (cont'd)

Symbol	Pin Num.	Input Outp.	Function	
P4		IO	Port 4 is an 8-bit bidirectional I/O port. It is bit-wise programmable for input or output via direction bits. For a pin configured as input, the output driver is put into high-impedance state. The Port 4 outputs can be configured as push/pull or open drain drivers. The input threshold of Port 4 is selectable (standard or special). Port 4 can be used to output the segment address lines, the optional chip select lines, and for serial interface lines:[1)]	
P4.0	80	O	A16	Least Significant Segment Address Line
P4.1	81	O	A17	Segment Address Line
P4.2	82	O	A18	Segment Address Line
P4.3	83	O	A19	Segment Address Line
P4.4	84	O	A20	Segment Address Line,
		I	CAN2_RxD	CAN Node 2 Receive Data Input,
		I	SDL_RxD	SDLM Receive Data Input,
		I	EX5IN	Fast External Interrupt 5 Input (alternate pin B)
P4.5	85	O	A21	Segment Address Line,
		I	CAN1_RxD	CAN Node 1 Receive Data Input,
		I	EX4IN	Fast External Interrupt 4 Input (alternate pin B)
P4.6	86	O	A22	Segment Address Line,
		O	CAN1_TxD	CAN Node 1 Transmit Data Output,
		I	SDL_RxD	SDLM Receive Data Input,
		I	EX5IN	Fast External Interrupt 5 Input (alternate pin A)
P4.7	87	O	A23	Most Significant Segment Address Line,
		I	CAN1_RxD	CAN Node 1 Receive Data Input,
		O	CAN2_TxD	CAN Node 2 Transmit Data Output,
		I	SDL_TxD	SDLM Transmit Data Output,
		I	EX4IN	Fast External Interrupt 4 Input (alternate pin A)

Table 2 Pin Definitions and Functions (cont'd)

Symbol	Pin Num.	Input Outp.	Function
P20		IO	Port 20 is a 6-bit bidirectional I/O port. It is bit-wise programmable for input or output via direction bits. For a pin configured as input, the output driver is put into high-impedance state. The following Port 20 pins also serve for alternate functions:
P20.0	90	O	$\overline{RD}$ External Memory Read Strobe, activated for every external instruction or data read access.
P20.1	91	O	$\overline{WR}$/$\overline{WRL}$ External Memory Write Strobe. In $\overline{WR}$-mode this pin is activated for every external data write access. In $\overline{WRL}$-mode this pin is activated for low byte data write accesses on a 16-bit bus, and for every data write access on an 8-bit bus.
P20.2	92	I	READY Ready Input. When the Ready function is enabled, memory cycle time waitstates can be forced via this pin during an external access.
P20.4	93	O	ALE Address Latch Enable Output. Can be used for latching the address into external memory or an address latch in the multiplexed bus modes.
P20.5	94	I	$\overline{EA}$ External Access Enable pin. **A low level** at this pin during and after Reset forces the XC161 to latch the configuration from PORT0 and pin $\overline{RD}$, and to begin instruction execution out of external memory. **A high level** forces the XC161 to latch the configuration from pins $\overline{RD}$, ALE, and $\overline{WR}$, and to begin instruction execution out of the internal program memory. "ROMless" versions must have this pin tied to '0'.
P20.12	3	O	$\overline{RSTOUT}$ Internal Reset Indication Output. **Is activated** asynchronously with an external hardware reset. It may also be activated (selectable) synchronously with an internal software or watchdog reset. **Is deactivated** upon the execution of the EINIT instruction, optionally at the end of reset, or at any time (before EINIT) via user software.

Table 2 Pin Definitions and Functions (cont'd)

Symbol	Pin Num.	Input Outp.	Function
PORT0		IO	PORT0 consists of the two 8-bit bidirectional I/O ports P0L and P0H. It is bit-wise programmable for input or output via direction bits. For a pin configured as input, the output driver is put into high-impedance state. In case of an external bus configuration, PORT0 serves as the address (A) and address/data (AD) bus in multiplexed bus modes and as the data (D) bus in demultiplexed bus modes.
P0L.0-7	95 - 102		**Demultiplexed bus modes:** Data Path Width: 8-bit 16-bit
P0H.0-1	105 - 106		P0L.0 – P0L.7: D0 – D7 D0 - D7 P0H.0 – P0H.7: I/O D8 - D15
P0H.2-7	111 - 116		**Multiplexed bus modes:** Data Path Width: 8-bit 16-bit P0L.0 – P0L.7: AD0 – AD7 AD0 - AD7 P0H.0 – P0H.7: A8 - A15 AD8 - AD15 *Note: At the end of an external reset ($\overline{EA}$ = '0') PORT0 also may input configuration values*
PORT1		IO	PORT1 consists of the two 8-bit bidirectional I/O ports P1L and P1H. It is bit-wise programmable for input or output via direction bits. For a pin configured as input, the output driver is put into high-impedance state. PORT1 is used as the 16-bit address bus (A) in demultiplexed bus modes (also after switching from a demultiplexed to a multiplexed bus mode). The following PORT1 pins also serve for alt. functions:
P0L.0-6	117 - 123	O	(A0-6) Address output only
P0L.7	124	I/O	CC22IO CAPCOM2: CC22 Capture Inp./Compare Outp.
P1H.0	127	I/O O	CC23IO CAPCOM2: CC23 Capture Inp./Compare Outp., EX0IN Fast External Interrupt 0 Input (alternate pin B)
P1H.1	128	O	MRST1 SSC1 Master-Receive/Slave-Transmit In/Outp.
P1H.2	129	I/O	MTSR1 SSC1 Master-Transmit/Slave-Receive Out/Inp.
P1H.3	130	I/O I	SCLK1 SSC1 Master Clock Output / Slave Clock Input, EX0IN Fast External Interrupt 0 Input (alternate pin A)
P1H.4	131	I/O	CC24IO CAPCOM2: CC24 Capture Inp./Compare Outp.
P1H.5	132	I/O	CC25IO CAPCOM2: CC25 Capture Inp./Compare Outp.
P1H.6	133	I/O	CC26IO CAPCOM2: CC26 Capture Inp./Compare Outp.
P1H.7	134	I/O	CC27IO CAPCOM2: CC27 Capture Inp./Compare Outp.

Table 2 Pin Definitions and Functions (cont'd)

Symbol	Pin Num.	Input Outp.	Function
XTAL2 XTAL1	137 138	O I	XTAL2: Output of the oscillator amplifier circuit. XTAL1: Input to the oscillator amplifier and input to the internal clock generator To clock the device from an external source, drive XTAL1, while leaving XTAL2 unconnected. Minimum and maximum high/low and rise/fall times specified in the AC Characteristics must be observed.
XTAL3 XTAL4	140 141	I O	XTAL3: Input to the 32-kHz oscillator amplifier and input to the internal clock generator XTAL4: Output of the oscillator amplifier circuit. To clock the device from an external source, drive XTAL3, while leaving XTAL4 unconnected. Minimum and maximum high/low and rise/fall times specified in the AC Characteristics must be observed.
$\overline{\text{RSTIN}}$	142	I	Reset Input with Schmitt-Trigger characteristics. A low level at this pin while the oscillator is running resets the XC161. An internal pullup resistor permits power-on reset using only a capacitor connected to V_{SS}. A spike filter suppresses input pulses <10 ns. Input pulses >100 ns safely pass the filter. The minimum duration for a safe recognition should be 100 ns + 2 CPU clock cycles. *Note: To let the reset configuration of PORT0 settle and to let the PLL lock a reset duration of ca. 1 ms is recommended.*
$\overline{\text{BRK OUT}}$	143	O	Debug System: Break Out
$\overline{\text{BRKIN}}$	144	I	Debug System: Break In
NC	1, 2, 107 - 110	-	No connection. It is recommended not to connect these pins to the PCB.
V_{AREF}	41	-	Reference voltage for the A/D converter.
V_{AGND}	42	-	Reference ground for the A/D converter.
V_{DDI}	48, 78, 135	-	Digital Core Supply Voltage (On-Chip Modules): +2.5 V during normal operation and idle mode.

Table 2 Pin Definitions and Functions (cont'd)

Symbol	Pin Num.	Input Outp.	Function
V_{DDP}	6, 20, 28, 58, 88, 103, 125	-	Digital Pad Supply Voltage (Pin Output Drivers): +5 V during normal operation and idle mode.
V_{SSI}	47, 79, 136, 139	-	**Digital Ground.** Connect decoupling capacitors to adjacent V_{DD}/V_{SS} pin pairs as close as possible to the pins. All V_{SS} pins must be connected to the ground-line or ground-plane.
V_{SSP}	5, 19, 27, 89, 104, 126	-	

1) The CAN/SDLM interface lines are assigned to ports P4, P7, and P9 under software control.

Note: The different power supply pins (V_{DDI}, V_{DDP}) refer to 2.5V for internal logic supply (V_{DDI}) and 5V for pad supply (V_{DDP}) voltage.

8 Keyword Index

This section lists a number of keywords which refer to specific details of the XC161 in terms of its architecture, its functional units or functions. This helps to quickly find the answer to specific questions about the XC161.

A

ADC 23
Analog/Digital Converter 23
ASC
  Features 17

B

Block Diagram ITC / PEC 8
Block Diagram of Pegasus **1**
Bus
  CAN 25
  I2C 19, 20

C

CAN Interface 25
CAPCOM 21
Capture/Compare Units 21
Central Processing Unit (CPU) 3, **3**
Central System Control Block 13
CGU 14
Clock Generation Unit 14
Control
  reset 13
  unit, system 13
CPU Architecture 4

D

Data Management Unit (Introduction) 6

E

External
  interrupts control 14

F

Flash
  Block diagram 12
  Features 11
  Introduction 11

G

General Purpose Timer 20
GPT 20
GPT12E
  Features 21

I

I2C Interface 20
Identification Control Block 14
Interface
  CAN 25
  I2C 20
  J1850 19
Interrupt
  System 7

J

J1850 Interface (->SDLM) 19

M

Memory Organization 11

O

On-Chip Peripheral System 16
On-chip Program Flash 11

P

Pins 32
Power Saving Control Block 13
Program Flash 11
Program Management Unit (Introduction) 6

R

Real Time Clock 24
Reset
    control block 13
RTC 24

S

SDLM 19
Serial Interface
    CAN 25
    I2C 20
    J1850 19
Sleep mode 13
SSC
    Features 18
Startup Process 13
System
    control unit 13
System Architecture 1
System Resources 3

T

Timer 20

W

Watchdog Timer 13
WDT 13

X

XC164CS Memory Organization 11

Infineon goes for Business Excellence

“Business excellence means intelligent approaches and clearly defined processes, which are both constantly under review and ultimately lead to good operating results.
Better operating results and business excellence mean less idleness and wastefulness for all of us, more professional success, more accurate information, a better overview and, thereby, less frustration and more satisfaction.”

Dr. Ulrich Schumacher

http://www.infineon.com

Published by Infineon Technologies AG