Customer Training Workshop
Traveo™ II Protection Units

Q4 2020
# Target Products

- **Target product list for this training material**

<table>
<thead>
<tr>
<th>Family Category</th>
<th>Series</th>
<th>Code Flash Memory Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traveo™ II Automotive Body Controller Entry</td>
<td>CYT2B6</td>
<td>Up to 576KB</td>
</tr>
<tr>
<td>Traveo II Automotive Body Controller Entry</td>
<td>CYT2B7</td>
<td>Up to 1088KB</td>
</tr>
<tr>
<td>Traveo II Automotive Body Controller Entry</td>
<td>CYT2B9</td>
<td>Up to 2112KB</td>
</tr>
<tr>
<td>Traveo II Automotive Body Controller Entry</td>
<td>CYT2BL</td>
<td>Up to 4160KB</td>
</tr>
<tr>
<td>Traveo II Automotive Body Controller High</td>
<td>CYT3BB/4BB</td>
<td>Up to 4160KB</td>
</tr>
<tr>
<td>Traveo II Automotive Body Controller High</td>
<td>CYT4BF</td>
<td>Up to 8384KB</td>
</tr>
<tr>
<td>Traveo II Automotive Cluster</td>
<td>CYT3DL</td>
<td>Up to 4160KB</td>
</tr>
<tr>
<td>Traveo II Automotive Cluster</td>
<td>CYT4DN</td>
<td>Up to 6336KB</td>
</tr>
</tbody>
</table>
Introduction to Traveo II Body Controller Entry

Protection units (MPU/SMPU/PPU) are in CPUSS and peripheral interconnect.

Review TRM chapter 6 and the Register TRM for additional details.
Protection units (MPU/SMPU/PPU) are in CPUSS and peripheral interconnect

Review TRM chapter 6 and the Register TRM for additional details.
Introduction to Traveo II Cluster

Protection units (MPU/SMPU/PPU) are in CPUSS and peripheral interconnect

CPU Subsystem
- Arm Cortex-M7 320 MHz
- eCT Flash 6306 KB Code-flash + 128 KB Work-flash
- SRAM0 256 KB
- SRAM1 256 KB
- SRAM2 128 KB
- SSRAM
- P RAM
- CRYPTO AES, SHA, CRC, TRNG, RSA, ECC
- CRYPT0 Initiator-MUO
- SW/MTB/CTI
- ROM 64 KB
- ROM Controller

GFX Subsystem
- VRAM 4096 KB
- Voice Codec

Peripheral Interconnect (MMIO, PPU)
- IOSS GPIO
- PCLK
- 52x GPIO_STD, 8x GPIO_ENH, 26x GPIO_SMC, 70x HSIO_STD, 22x HSIO_ENH, 4x HSIO_ENG_DIFF

System Interconnect (Multi Layer AXI/AHB, IPC, MPU/SMPU)

Review TRM chapter 6 and the Register TRM for additional details

Copyright © Infineon Technologies AG 2020. All rights reserved.
Protection Unit Overview

› Overview
  - Allows/restricts bus transfers on the bus infrastructure
  - Includes four protection units (MPU/SMPU/PPU/SWPU)

› Features
  - Supports various access attributes
    - Address range (Start Address and Region Size)
    - Read/Write
    - Execute (Code or Data)
    - Privileged/Unprivileged
    - Secure/Non-secure
    - Protection context (PC)
  - Protects protection structures
  - Captures protection violations in fault report structure
Protection Context

- Protection Context (PC) is an access attribute of the protection units
- The PC prevents erroneous writing from the access with an unintended PC to memory and peripheral
- The PC helps to apply different protection attributes without changing the SMPU, PPU, and SWPU settings
  - Traveo II supports eight PCs
  - Used as the PC attribute for bus transfers
  - Access to memory and peripheral is allowed or restricted by SMPU, PPU and SWPU
  - Changed by reprogramming the PC field in MSx_CTL
  - PC0 and PC1 are hardware-controlled and not available to the user; PC2 to PC7 are available

Hint Bar
Review TRM chapter 6 and the Register TRM for additional details
Protection Context Programming and Restrictions

› PC Programming
  - Main CPU\(^1\), secondary CPU\(^2\), and test controller bus master have a PC field\(^3\)
  - A PC attribute is changed by reprogramming the PC field
  - The PC is used as the access attribute for all bus transfers by the master
  - The SMPUs, PPUs, and SWPUs allow/restrict bus transfers based on the PC attribute

› PC Programming Restrictions
  - Changes to the PC can be restricted by the PC\_MASK\(^4\) field
  - Controlled by secure CPU (CM0+)

PC\_MASK Example\(^5\)

<table>
<thead>
<tr>
<th>PC_MASK_15_TO_1 Field</th>
<th>PC_MASK_0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 1 0 1 0 0 0 0</td>
<td></td>
</tr>
</tbody>
</table>

\(^1\) The main CPU refers to CM4 or CM7 CPU in the MCU.
\(^2\) The secondary CPU refers to CM0+ CPU in the MCU.
\(^3\) The PC field is located in the MPU\_MSx_CTL register.
\(^4\) PC\_MASK field is located in the SMPU\_MSx_CTL register. PC\_MASK_0 is always ‘0’ and cannot set any master.
\(^5\) PCs 4 and 6 are configurable; other PCs are nonconfigurable.
Use Cases:
Protection Context

Software separation between ASIL\(^1\) and QM\(^2\)

1. ASIL: Automotive Safety Integrity Level. This software function has safety requirements.
2. QM: Quality Management. This software function does not have safety requirements.
Use Cases:
Protection Context

› Software separation between ASIL and QM

› Assign PC[4] to ASIL software

Main CPU

- ASIL Software
  - Privileged

- QM Software
  - Unprivileged

PC Field for Main CPU

- PC[4]
  - Transfer Attribute: Privileged Read

- Attribute of PC[4] is Given for Bus Transfer After PC Field Setting

SMPU

- Region 0
  - Attributes
    - PC[4]
    - Privileged Read/Write
    - Unprivileged Read/Write

- Region 1
  - Attributes
    - PC[5]
    - Privileged Read/Write
    - Unprivileged Read/Write

- Region 2
  - Attributes
    - Privileged Read/Write
    - Unprivileged Read/Write

Memory

- Region 0

- Region 1

- Region 2
Use Cases:

Protection Context

- Software separation between ASIL and QM
- Assign PC[4] to ASIL software
- Restrict memory access between PCs with SMPU
Use Cases:

Protection Context

Software separation between ASIL and QM

- **Main CPU**
  - ASIL Software: Privileged
  - QM Software: Unprivileged
  - QM Software sets PC field to PC[5] before memory access

- **PC Field for Main CPU**
  - PC[5]

- **SMPU**
  - Region 0
    - Attributes
      - PC[4]
      - Privileged Read/Write
      - Unprivileged Read/Write
  - Region 1
    - Attributes
      - PC[5]
      - Privileged Read/Write
      - Unprivileged Read/Write
  - Region 2
    - Attributes
      - Privileged Read/Write
      - Unprivileged Read/Write

- **Memory**
  - Region 0
  - Region 1
  - Region 2
Use Cases:

Protection Context

- Software separation between ASIL and QM
- Assign PC[5] to QM software
Use Cases:

Protection Context

› Software separation between ASIL and QM
› Assign PC[5] to QM software
› Restrict memory access between PCs with SMPU

Main CPU

PC Field for Main CPU

SMPU

Memory

› Region 0
  › Attributes
    - PC[4]
    - Privileged Read/Write
    - Unprivileged Read/Write

QM Software is not Permitted in Region 0 and Region 2

› Region 1
  › Attributes
    - PC[5]
    - Privileged Read/Write
    - Unprivileged Read/Write

QM Software is Permitted in Region 1

› Region 2
  › Attributes
    - Privileged Read/Write
    - Unprivileged Read/Write

QM Software is not Permitted in Region 0 and Region 2
Protection unit consists of the following components:
Memory Protection Unit Block Diagram

- **MPU**
  - Provides memory protection
  - Associated with a single master
    - Each CPU can independently give different roles
    - P-DMA, M-DMA, and Crypto inherit the attribute of programmed transfer
  - Up to sixteen regions
  - Two types of MPU:
    - Arm Cortex-M4/7 and Cortex-M0+ CPU
    - Bus infrastructure\(^1\) for the test controller\(^2\)
  - Access restriction:
    - Address range, Read/Write, Execute, and Privileged/Unprivileged
    - The region is equally divided into eight subregions

---

\(^1\) The definition of MPU aspect of bus infrastructure follows the Arm MPU definition.

\(^2\) The test controller is connected to the external debugger.
Use Case 1

Software (func1) of both CPUs accesses to memory with privileged write

1 Main CPU_0 is CM4 or CM7_0 in CYT2 (Entry).
2 Main CPU_1 is CM7_1 in CYT4BF (High).
Use Case 1
- Software (func1) of Main CPU_0 is allowed access to Region 0 and Region 1
- Software (func1) of Main CPU_1 or secondary CPU is allowed access to Region 0 and Region 2
Use Case 2
- Software (func2) of both CPUs accesses to memory with unprivileged read
Use Case 2

- Software (func2) of Main CPU_0 is allowed access to Region 1
- Software (func2) of Main CPU_1 or secondary CPU is allowed access to all regions
P-DMA/M-DMA/Crypto Access Attribute

- Inherit the access control attributes of the programmed bus transfer for these bus transfer attributes

Main CPU

DMA CH0 Setting
- Unprivileged
- PC[7]
- Non-secure

P-DMA

P-DMA Channel Setting

P-DMA CH0 Setting
- Unprivileged
- PC[7]
- Non-secure

Inherit the Access Attributes

Activation Trigger

P-DMA

Unprivileged, Write, PC[7], Non-secure

SMPU

Memory

Unprivileged, Read, PC[7], Non-secure

Peripheral

PPU

Hint Bar

Review TRM chapter 6 for additional details
Shared Memory Protection Unit Block Diagram

- SMPU
  - Provides memory protection
  - Shared by all bus masters
  - Includes 16 regions
  - Access restriction to:
    - Address range
    - Read/Write
    - Execute
    - Privileged/Unprivileged
    - Secure/Non-secure
    - Protection Context
  - The region is equally divided into eight subregions
  - Protection pair: slave and master protection structures

Hint Bar
- Review TRM chapter 6 and the Register TRM for additional details
- Training section references: CPUSS

Copyright © Infineon Technologies AG 2020. All rights reserved.
SMPU Protection

Use Case 1
- Main CPU_0 software accesses to memory with privileged write, non-secure, and PC[5]
SMPU Protection

› Use Case 1
  - Main CPU_0 software is allowed access to Region 1 and Region 3
SMPU Protection

Use Case 2
- Main CPU_1 software accesses to memory with unprivileged write, non-secure, and PC[6]

<table>
<thead>
<tr>
<th>Main CPU_0</th>
<th>SMPU</th>
<th>Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Privileged</td>
<td>Region 0</td>
<td>Region 0</td>
</tr>
<tr>
<td>Write</td>
<td>- Attribute:</td>
<td></td>
</tr>
<tr>
<td>Non-secure</td>
<td>- PC[4]</td>
<td></td>
</tr>
<tr>
<td>PC[5]</td>
<td>- Privileged Read/Write</td>
<td>Region 0</td>
</tr>
<tr>
<td></td>
<td>- Unprivileged No Access</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Secure</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Region 1</td>
<td>Region 1</td>
</tr>
<tr>
<td></td>
<td>- Attribute:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Privileged Read/Write</td>
<td>Region 1</td>
</tr>
<tr>
<td></td>
<td>- Unprivileged Read only</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Non-secure</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Region 2</td>
<td>Region 2</td>
</tr>
<tr>
<td></td>
<td>- Attribute:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Privileged Read/Write</td>
<td>Region 2</td>
</tr>
<tr>
<td></td>
<td>- Unprivileged No Access</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Secure</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Region 3</td>
<td>Region 3</td>
</tr>
<tr>
<td></td>
<td>- Attribute:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- PC[4], [5] and [6]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Privileged Read/Write</td>
<td>Region 3</td>
</tr>
<tr>
<td></td>
<td>- Unprivileged Read/Write</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Non-secure</td>
<td></td>
</tr>
</tbody>
</table>

Secondary CPU

Main CPU_1

Unprivileged
Write
Non-secure
PC[6]
Use Case 2
- Main CPU_1 software is allowed access to Region 3
SMPU Protection

- **Use Case 3**
  - Secondary CPU software accesses to memory with privileged write, Secure, and PC[4]
### Use Case 3

- Secondary CPU software is allowed access to Region 0, Region 2, and Region 3.
Peripheral Protection Unit Block Diagram

- **PPU**
  - Provides peripheral protection
  - Shared by all bus masters
  - Two PPU types:
    - Fixed PPU structure
    - Programmable PPU structure
  - Access restriction to:
    - Address range (restricted to programmable PPU)
    - Read/Write
    - Privileged/Unprivileged
    - Secure/Non-secure
    - Protection Context
    - Applies different setting for each PC
  - Protection pair: slave and master protection structures
  - Protection information uses a single SRAM with ECC

1. Protects address range of known resources
2. Protects address range of specific resources

---

**Hint Bar**

Review TRM chapter 6 and Register TRM for additional details

Review TRM section 6.4.7 for details of SRAM and ECC

Fixed structures take precedence over programmable structures

---

Copyright © Infineon Technologies AG 2020. All rights reserved.
Use Cases:

PPU Protection

- Execute system settings such as clock and initial setting of peripheral in startup software at power-up
- Startup software: Privileged, PC[4] access to peripheral
Use Cases:

PPU Protection

› Transfer to normal operation software after initial setting and startup software is not executed

› Normal operation software: Privileged, PC[6] access to peripheral

Main CPU

PC Field for main CPU

PPU

Peripheral

Peripherals

Control

After PC[6] Setting

Startup Software

Privileged

PC[4]

Normal Operation Software

Privileged

PC[6]

Transfer

Attribute is Privileged, Write, PC[6]

Transfer

Attribute

PC[6]

Region 0

Attribute:

- PC[4]
  - Privileged Read/Write
  - Unprivileged Read/Write

- PC[6]
  - Privileged No Access
  - Unprivileged No Access

System Control (Clock, Interrupt, etc.)

Peripheral Control/ Data Register

Hint Bar

Advantage:

Prevents erroneous change of system setting by applying different access attributes between initial setting and normal

Copyright © Infineon Technologies AG 2020. All rights reserved.
SWPU

- SWPU is used to access restrictions to flash (write) and eFuse (read/write)
  - SWPU prevents malicious or unintended access of flash or eFuse.
  - SWPU has three Protection Units:
    - FWPU: Flash Write Protection Unit. (Up to 16 regions)
    - ERPU: eFuse Read Protection Unit. (Up to 4 regions)
    - EWPU: eFuse Write Protection Unit. (Up to 4 regions)
  - SWPU has two configuration parts:
    - Boot protection: It cannot be updated.
    - Application protection: It is additional access restrictions specific to the application.
  - SWPU is read during boot process from SFlash, and stores them in RAM
  - Access restriction to:
    - Address range
    - Read/Write
    - Privileged/Unprivileged
    - Secure/Non-secure
    - Protection Context
  - Protection pair: slave and master protection structures

* Program and Erase
### SWPU Structure (Application Protection)

#### SWPU is stored in SFlash

<table>
<thead>
<tr>
<th>SWPU</th>
<th>Field Name</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Object Size</td>
<td>4 bytes</td>
<td>Number of configured elements</td>
</tr>
<tr>
<td>FWPU</td>
<td>N_FWPU</td>
<td>4 bytes</td>
<td>Number of FWPU objects. Up to 16 regions</td>
</tr>
<tr>
<td></td>
<td>FWP0_SL_ADDR</td>
<td>4 bytes</td>
<td>Configures the FWP0 base address</td>
</tr>
<tr>
<td></td>
<td>FWP0_SL_SIZE</td>
<td>4 bytes</td>
<td>Configures the FWP0 region size and FWP0 enable</td>
</tr>
<tr>
<td></td>
<td>FWP0_SL_ATT</td>
<td>4 bytes</td>
<td>Configures the FWP0 slave attribute</td>
</tr>
<tr>
<td></td>
<td>FWP0_MS_ATT</td>
<td>4 bytes</td>
<td>Configures the FWP0 master attribute</td>
</tr>
<tr>
<td>ERPU</td>
<td>N_ERPU</td>
<td>4 bytes</td>
<td>Number of ERPU objects. ERPU has up to four regions.</td>
</tr>
<tr>
<td></td>
<td>ERPU0_SL_OFFSET</td>
<td>4 bytes</td>
<td>Configures the ERPU0 base address offset</td>
</tr>
<tr>
<td></td>
<td>ERPU0_SL_SIZE</td>
<td>4 bytes</td>
<td>Configures the ERPU0 region size and ERPU0 enable</td>
</tr>
<tr>
<td></td>
<td>ERPU0_SL_ATT</td>
<td>4 bytes</td>
<td>Configures the ERPU0 slave attribute</td>
</tr>
<tr>
<td></td>
<td>ERPU0_MS_ATT</td>
<td>4 bytes</td>
<td>Configures the ERPU0 master attribute</td>
</tr>
<tr>
<td>EWPU</td>
<td>N_EWPU</td>
<td>4 bytes</td>
<td>Number of EWPU objects. EWPU has up to four regions.</td>
</tr>
<tr>
<td></td>
<td>EWPU0_SL_OFFSET</td>
<td>4 bytes</td>
<td>Configures the EWPU0 base address offset</td>
</tr>
<tr>
<td></td>
<td>EWPU0_SL_SIZE</td>
<td>4 bytes</td>
<td>Configures the EWPU0 region size and EWPU0 enable</td>
</tr>
<tr>
<td></td>
<td>EWPU0_SL_ATT</td>
<td>4 bytes</td>
<td>Configures the EWPU0 slave attribute</td>
</tr>
<tr>
<td></td>
<td>WPU0_MS_ATT</td>
<td>4 bytes</td>
<td>Configures the EWPU0 master attribute</td>
</tr>
</tbody>
</table>
Protection Pair

- SMPU, PPU, and SWPU have a protection pair for protection of protection structures
  - Master structure: Protection of slave structure
  - Slave structure: Protection of resources
  - Change slave setting attributes by master setting attributes
Protection Violation

- MPU, SMPU, and PPU detect a restricted transfer; the bus transfer results in a bus error
  - Access is evaluated by each protection unit in the following order:
    - MPU (High) > SMPU > PPU (Low)
  - The slave address regions in programmable PPU may overlap with other slave address regions. In this case, access is evaluated by the PPU in the following order:
    - PPU master structure (High) > Programmable slave structures > Fixed slave structure (Low)
  - The bus transfer will not reach its target

- Violation information is reported to the fault reporting structure
  - Violation address
  - Bus transfer attribute
  - Accessed bus master

Hint Bar

Training section references: Fault Structure

Advantage:
Prevents erroneous writing by not allowing transfer

Possible analysis of violation transfer
Part of your life. Part of tomorrow.
<table>
<thead>
<tr>
<th>Revision</th>
<th>ECN</th>
<th>Submission Date</th>
<th>Description of Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>**</td>
<td>6138645</td>
<td>04/17/2018</td>
<td>Initial release</td>
</tr>
<tr>
<td>*A</td>
<td>6346788</td>
<td>12/10/2018</td>
<td>Added slide 2. Updated slides 3, 4, and 5. Revised description on slides 10 and 11 from Privileged to Unprivileged. In all slides, changed CM4 and CM0+ to main CPU and secondary CPU, and added notes 4 and 5 on slide 8. Changed slides 22 – 27.</td>
</tr>
<tr>
<td>*B</td>
<td>6633371</td>
<td>7/22/2019</td>
<td>Added slide 5. Updated slide 2, 18.</td>
</tr>
<tr>
<td>*C</td>
<td>6825576</td>
<td>03/06/2020</td>
<td>Changed document title from PROTECTION UNITS (M4CPUSS_VER2/M7CPUSS) to TRAVEO(TM) II PROTECTION UNITS (M4CPUSS_VER2/M7CPUSS)</td>
</tr>
<tr>
<td>*D</td>
<td>7060711</td>
<td>01/06/2021</td>
<td>Updated Slide 2, 3, 7, 8, 16, 29, 34. Added slide 32, 33</td>
</tr>
</tbody>
</table>