Customer Training Workshop

Traveo™ II Body High and Cluster 2D CPU Subsystem (CPUSS)
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 High</td>
<td>CYT3BB/4BB</td>
<td>Up to 4160 KB</td>
</tr>
<tr>
<td>Traveo II Automotive Body Controller High</td>
<td>CYT4BF</td>
<td>Up to 8384 KB</td>
</tr>
<tr>
<td>Traveo II Automotive Cluster</td>
<td>CYT3DL</td>
<td>Up to 4160 KB</td>
</tr>
<tr>
<td>Traveo II Automotive Cluster</td>
<td>CYT4DN</td>
<td>Up to 6336 KB</td>
</tr>
</tbody>
</table>
CPU Subsystem Overview

The CPU subsystem (CPUSS) is based on dual 32-bit Arm® Cortex® CPUs.

CPU Subsystem Components

- **Executes application software as the main CPU**
- **System RAM** stores data
- **Data transfer functions such as peripheral to memory, memory to peripheral, and memory to memory**
- **Implements security, safety, and protection features as the secondary CPU**

Code/Work Flash stores programs and data

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

Hardware cryptographic functions

Stores the boot code executed after reset

1. CYT4 series have two Cortex-M7 CPUs, and CYT3 series have single Cortex-M7 CPU
Cortex®-M7 Features Summary

› Main CPU
  - Two Cortex-M7 CPUs 1
  - Execution of application software
  - Up to 350-MHz operation (CYT4BF) 2
  - Up to 320-MHz operation (CYT4DN) 2

› Tightly-coupled memories (ITCM and DTCM) and instruction and data caches3

› System tick (SysTick) timer

› Floating-point unit (FPU)
  - Single and double precision
  - Compliant with the ANSI/IEEE Std 754-2008 IEEE Standard for Binary Floating-Point Arithmetic

Memory protection unit (MPU)
  - Sixteen protection regions
  - Privileged/unprivileged, read/write attributes

Nested vector interrupt controller (NVIC)
  - Eight external system interrupts, eight internal software interrupts, eight interrupt levels, and one non-maskable interrupt
  - Wakeup interrupt controller (WIC) support
  - Vector table relocation (VTOR)

Debug components
  - Supported SWD and JTAG interface (SWJ)
  - Tracing components (ETM/ITM over ETB/TPIU)
  - Cross-triggering component (CTI)

1 CYT4 series have two Cortex-M7 CPUs, and CYT3 series have single Cortex-M7 CPU
2 See the device datasheet for operation frequency of target product.
3 See the device data sheet for supported capacity of ITCM, DTCM and caches.
Cortex-M0+ Features Summary

› Secondary CPU
  - Execution of boot process
  - Secure master in secure system to establish a root-of-trust
  - Up to 100-MHz operation
› SysTick timer
› Memory protection unit (MPU)
  - Eight protection regions
  - Privileged/unprivileged access attributes

› Nested vector interrupt controller (NVIC)
  - Eight external system interrupts, eight internal software interrupts, four interrupt levels, and one non-maskable interrupt
  - Wakeup interrupt controller (WIC) support
  - Vector table relocation (VTOR)
  - Debug components
    - Supported SWD and JTAG interface (SWJ)
    - Tracing component (ETM over MTB)
    - Cross-triggering component (CTI)

Hint Bar

Arm provides additional reference material on their webpage at:
http://infocenter.arm.com

Training section references:
- Interrupts
- Program and Debug Interface
- Protection Units
Combined SWD/JTAG interface (SWJ)
Micro trace buffer (MTB)
Embedded trace macrocell (ETM)
Cross-triggering interface (CTI)

1 For CM4 to operate at 160 MHz, CM0+ is required to operate at a frequency of 80 MHz
2 See the device datasheet for operation frequency of target product
Each bus master has a dedicated master identifier, which is used for:
- Bus arbitration
- IPC lock acquire functionality
- Violation access information by MPU, SMPU, and PPU

<table>
<thead>
<tr>
<th>Master Identifier</th>
<th>Bus Master (CYT4BF)</th>
<th>Bus Master (CYT4DN)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Cortex®-M0+</td>
<td>Cortex®-M0+</td>
</tr>
<tr>
<td>1</td>
<td>Crypto</td>
<td>Crypto</td>
</tr>
<tr>
<td>2</td>
<td>P-DMA 0</td>
<td>P-DMA 0</td>
</tr>
<tr>
<td>3</td>
<td>P-DMA 1</td>
<td>P-DMA 1</td>
</tr>
<tr>
<td>4</td>
<td>M-DMA</td>
<td>M-DMA</td>
</tr>
<tr>
<td>5</td>
<td>SDHC</td>
<td>-</td>
</tr>
<tr>
<td>9</td>
<td>Ethernet 0</td>
<td>Ethernet 0</td>
</tr>
<tr>
<td>10</td>
<td>Ethernet 1</td>
<td>-</td>
</tr>
<tr>
<td>12</td>
<td>-</td>
<td>Video Subsystem</td>
</tr>
<tr>
<td>13</td>
<td>Cortex-M7_1</td>
<td>Cortex-M7_1</td>
</tr>
<tr>
<td>14</td>
<td>Cortex-M7_0</td>
<td>Cortex-M7_0</td>
</tr>
<tr>
<td>15</td>
<td>Test Controller</td>
<td>Test Controller</td>
</tr>
</tbody>
</table>

1 See the device datasheet for master identifier number and regarding peripherals of target product.
CPUSS Bus Infrastructure

› AHB-Lite and AXI bus infrastructure
› Bus clock domains
  – Fast/slow clock domains
  – Each memory interface has both domains

› Bus competition
  – CPUSS has multiple bus masters
  – Design a system that considers bus competition by simultaneous access

Clock Domains

Fast Clock Domain

CM7_0 CPU
- TCM_0
  - Fast Clock
  - Slow Clock

CM7_1 CPU
- TCM_1
  - Fast Clock
  - Slow Clock

CM0+ CPU
- Test Controller (TC)
  - M-DMA
    - Crypto
    - P-DMA0
    - P-DMA1

System Interconnect (Fast Clock)
AXI, AHB-Lite

- Fast Clock
- Slow Clock

Flash

System Interconnect (Slow Clock)
AHB-Lite

- Fast Clock
- Slow Clock

CM7_0/1 Peripheral Interface
CM0+/TC Peripheral Interface
P-DMA0 Peripheral Interface
P-DMA1 Peripheral Interface

Slow Clock Domain

- ROM
- SRAM0,1,2

- Fast Clock
- Slow Clock
CPU Features

- **Cortex-M7**
  - 5 CoreMark/MHz and 2.14 DMIPS/MHz
  - ISA Support (Thumb/Thumb-2)
  - 6-stage superscalar + branch prediction pipeline
  - Interconnect
    - AXI master
    - AHB peripheral port (AHBP/AHBS)
    - Tightly-coupled memory (ITCM/DTCM)
  - Instruction and data cache
  - DSP Extensions
    - Single-cycle 16/32-bit MAC, single-cycle dual 16-bit MAC
    - 8/16-bit SIMD arithmetic
  - Floating Point Unit (FPU)
  - Memory Protection Unit (MPU)
  - Nested vector interrupt controller (NVIC)
    - Vector table relocation (VTOR)
  - Wake-up Interrupt Controller (WIC)
  - SysTick Timer

- **Cortex-M0+**
  - 1.99 CoreMark/MHz and 0.9 DMIPS/MHz
  - ISA Support (Thumb/Thumb-2)
  - 2-stage Pipeline
  - Interconnect
    - AHB - Lite
  - Memory Protection Unit (MPU)
  - Nested vector interrupt controller (NVIC)
    - Vector table relocation (VTOR)
  - Wake-up Interrupt Controller (WIC)
  - SysTick Timer

Arm provides additional reference material on their webpage at: [infocenter.arm.com](http://infocenter.arm.com)
CPU Mode Transition

Both CPUs support two operating modes and two privilege levels

- **Operating mode**
  - Thread mode executes application software
  - Handler mode handles exceptions

- **Privilege levels**
  - Unprivileged: Software has limited access to MSR/MRS instructions (uses CPS instructions), system timer, NVIC, system control block, and memory/peripherals
  - Privileged: Software can use all instructions and has access to all resources
  - Transition from Thread/Privileged to Thread/Unprivileged by CONTROL register\(^1\)
  - Transition from Thread/Unprivileged to Thread/Privileged via SVC\(^2\)

In Handler mode, software is always privileged. After return from Handler mode, the Thread/Privileged level depends on CONTROL

Return to Thread mode when it has finished all exception processing.

Controlled by the CONTROL register, which can be written from privileged software only

\(^1\) The CONTROL register is a CPU-specific register. It defines privileged/unprivileged, stack pointer, and FPU extension.

\(^2\) Supervisor calls are used to request privileged operations
Operation Modes Transition

Use Case: Changing Privileged/Unprivileged Mode

Handler/Privileged → Thread/Privileged → Thread/Unprivileged

- Reset Release
- Operation
- Exception
- Change CONTROL = Unprivileged
- Change Privileged to Unprivileged
- Change CONTROL = Privilege
- Change Unprivileged to Privileged

Arm provides additional reference material on their webpage at: infocenter.arm.com
Memory Protection

› Each CPU has a memory protection unit (MPU)\(^1\) that:
  - Realizes software separation freedom of interface
  - Includes address range, read/write, and privileged/unprivileged attributes
  - Features sixteen protection regions

› Use Case
  - Software partitioning of ASIL and QM in Functional Safety

\(^1\) MPU registers are written by privileged software and are only one part of the protection concept. For details, refer to the Protection Units training section.
Vector Table Relocation

Vector Table Offset Register (VTOR)¹
- Defines the location of the vector table of each core. There is a VTOR for each CM7 and another one for CM0.
- Can be used to relocate the vector table from Flash to SRAM, allowing the interrupt handlers to change dynamically
- VTOR is written from privileged software only
- Use the following registers to set VTOR for both cores:
  - CM0_VECTOR_TABLE_BASE: By default set to beginning of Flash by Boot ROM
  - CM7_0/1_VECTOR_TABLE_BASE: Set by CM0+ application before releasing the CPU core CM7_0/1 from reset
- After boot, each core copies the vector table to SRAM and updates VTOR with the address of the new location

Use Cases
- Execute the program with RAM only, for Flash programming or performance improvement
- Use different vectors depending on software level or system mode, such as normal and reprogramming

¹ The VTOR register is a CPU-specific register
System Tick Generation

› Each CPU supports a SysTick timer to measure time duration, which provides:
  – A 24-bit down counter
  – A selectable internal CPU clock or external clock
  – Active and Sleep mode operation
  – SysTick interrupt generation
› SysTick registers can be written from Privileged software only

› Use Cases
  – RTOS tick timer
  – Alarm timer to alert when an action is not completed within a particular duration
  – Software completion time measurement


**CPU Subsystem Memory Feature Summary**

- **TCM**
  - ITCM and DTCM (instruction/data tightly coupled memory) for CM7
  - Size 16KB: ITCM/16KB: DTCM(CYT4BF)
  - Size 64KB: ITCM/64KB: DTCM(CYT4DN)
  - Error-correction code (ECC) function (SEC/DED)

- **SRAM**
  - Data storage and code execution
  - Size: up to 1024KB(CYT4BF)
  - Size: up to 640KB(CYT4DN)
  - Each CPU sharing
  - ECC function

- **Flash**
  - Code and Work Flash
  - Size: up to 8MB: Code/ 256KB: Work (CYT4BF)
  - Size: up to 6MB: Code/ 128KB: Work (CYT4DN)
  - ECC function
  - Instruction cache for each CPU
  - APIs for Flash programming

- **ROM**
  - Boot code for CM0+
  - API function implementation

---

*1 See the device datasheet for memory size of target product.

*2 No user access to read or modify SROM code.
P-DMA/M-DMA Feature Summary

› Peripheral DMA (P-DMA)
- Single transfer engine shared for all channels
- Focuses on low-latency transfer
- Transfer modes:
  - Single, 1D, 2D, and CRC

› Memory DMA (M-DMA)
- Dedicated transfer engine for each channel
- Focuses on high-memory bandwidth
- Transfer modes:
  - Single, 1D, 2D, Memory Copy, and Scatter

CPU Subsystem Components

- Arm Cortex M4 160 MHz
- eCT Flash 4160 KB Code-flash + 128 KB Work-flash
- SRAM0 256 KB
- SRAM1 256 KB
- P-DMA0 4 Channel
- P-DMA1 44 Channel
- M-DMA0 4 Channel
- SWJ/ETM/ITM/ITM/CTI
- FPU, NVIC, MPU
- SWJ/ETM/ITM/CTI
- MUL, NVIC, MPU
- 100 MHz
- 32 KB
- 4160 KB Code
- 128 KB Work
- 256 KB
- 256 KB
- 8 KB
- 8 KB
- 8 KB
Cryptographic (Crypto) Feature Summary

- Hardware Crypto Functions
  - Symmetric key encryption and decryption
  - Hashing
  - Message authentication
  - Random number generation
  - Cyclic redundancy checking
  - Asymmetric key cryptography

CPU Subsystem Components

1 Access limited to the secure master (CM0+)
Appendix
## Comparison between Families

<table>
<thead>
<tr>
<th>Features</th>
<th>Body Controller Entry (CYT2BL)</th>
<th>Body Controller High (CYT4BF)</th>
<th>Cluster (CYT4DN)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Main CPU</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU</td>
<td>Cortex-M4 CPU</td>
<td>Two Cortex-M7 CPUs</td>
<td></td>
</tr>
<tr>
<td>Operating Frequency</td>
<td>Up to 160 MHz</td>
<td>Up to 350 MHz</td>
<td>Up to 320 MHz</td>
</tr>
<tr>
<td>FPU</td>
<td>Single-precision</td>
<td>Single/double-precision</td>
<td></td>
</tr>
<tr>
<td>Cache</td>
<td>N/A</td>
<td>16KB instruction, 16KB data</td>
<td></td>
</tr>
<tr>
<td>MPU</td>
<td>Cortex-M4: 8 regions</td>
<td>Cortex-M7: 16 regions</td>
<td></td>
</tr>
<tr>
<td>Interrupt Structure</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>System Tick Timer</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Secondary CPU</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU</td>
<td>Cortex-M0+ CPU</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Operating Frequency</td>
<td>Same (Up to 100 MHz)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MPU</td>
<td>Cortex-M0+: 8 regions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Interrupt Structure</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>System Tick Timer</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Flash</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bus Interface</td>
<td>AHB-Lite</td>
<td>AXI, AHB-Lite</td>
<td></td>
</tr>
<tr>
<td>ECC (SEC/DED)</td>
<td>Supported</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bank Modes</td>
<td>Supported</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Size (Code/Work)</td>
<td>4MB / 128KB</td>
<td>8MB / 256KB</td>
<td>6MB / 128KB</td>
</tr>
<tr>
<td><strong>SRAM</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bus Interface</td>
<td>AHB-Lite</td>
<td>AXI, AHB-Lite</td>
<td></td>
</tr>
<tr>
<td>ECC (SEC/DED)</td>
<td>Support</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TCM Size</td>
<td>N/A</td>
<td>16KB ITCM, 16KB DTCM</td>
<td>64KB ITCM, 64KB DTCM</td>
</tr>
<tr>
<td>SRAM Size</td>
<td>512KB</td>
<td>1024KB</td>
<td>640KB</td>
</tr>
<tr>
<td>Boot</td>
<td>Supported</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Device Security with Crypto</td>
<td>Supported</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Direct Memory Access</td>
<td>P-DMA</td>
<td>Supported</td>
<td></td>
</tr>
<tr>
<td></td>
<td>M-DMA</td>
<td>Supported</td>
<td></td>
</tr>
</tbody>
</table>
Part of your life. Part of tomorrow.
### Revision History

<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>6381034</td>
<td>11/12/2018</td>
<td>Initial release</td>
</tr>
<tr>
<td>*A</td>
<td>6633371</td>
<td>7/22/2019</td>
<td>Updated page 2, 3 to 6, 14 to 16, 18. Added page 8.</td>
</tr>
<tr>
<td>*B</td>
<td>7060645</td>
<td>01/06/2021</td>
<td>Updated Slide 2, 4, 6, 14</td>
</tr>
</tbody>
</table>