"Spoilt for Choice: What is the Right ARM Architecture?"

MICROCONSULT GmbH
Dipl.-Ing. Dieter Volland
E-Mail: d.volland@microconsult.com
# Spoilt for Choice

## Content

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Cortex Processors Overview</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Cortex-M Processors</td>
<td>4</td>
</tr>
<tr>
<td>2.1</td>
<td>Cortex-M Block Diagram</td>
<td>7</td>
</tr>
<tr>
<td>2.2</td>
<td>Register-Set Overview Cortex-M</td>
<td>13</td>
</tr>
<tr>
<td>2.3</td>
<td>Cortex-M Instruction Set</td>
<td>14</td>
</tr>
<tr>
<td>3</td>
<td>Cortex-R Processors</td>
<td>15</td>
</tr>
<tr>
<td>3.1</td>
<td>Cortex-R Block Diagram</td>
<td>19</td>
</tr>
<tr>
<td>3.2</td>
<td>Cortex-R Safety Features</td>
<td>21</td>
</tr>
<tr>
<td>3.3</td>
<td>Cortex-R Register Banks</td>
<td>22</td>
</tr>
<tr>
<td>4</td>
<td>Cortex-A Processors</td>
<td>23</td>
</tr>
<tr>
<td>4.1</td>
<td>Cortex-A Block Diagram</td>
<td>26</td>
</tr>
<tr>
<td>4.2</td>
<td>Pipeline</td>
<td>27</td>
</tr>
<tr>
<td>4.3</td>
<td>MMU</td>
<td>28</td>
</tr>
<tr>
<td>4.4</td>
<td>Cortex-A Register Banks</td>
<td>29</td>
</tr>
<tr>
<td>4.5</td>
<td>Global Interrupt Controller, GIC</td>
<td>30</td>
</tr>
<tr>
<td>4.6</td>
<td>Multiple Cores &amp; Snoop Control Unit, SCU</td>
<td>31</td>
</tr>
<tr>
<td>4.7</td>
<td>TrustZone</td>
<td>32</td>
</tr>
<tr>
<td>4.8</td>
<td>Virtualization- and Large Physical Address Extension</td>
<td>34</td>
</tr>
<tr>
<td>5</td>
<td>64 Bit Cortex-A Processors</td>
<td>35</td>
</tr>
<tr>
<td>6</td>
<td>More Informations</td>
<td>37</td>
</tr>
</tbody>
</table>
1 Cortex Processors Overview

ARMv1: 1985 (ARM1)
ARMv2: 1986 (ARM2), 1989 (ARM3)
ARMv3: 1991 (ARM6), 1993 (ARM7)
ARMv4: 1995 (ARM7TDMI), 1997 (ARM9TDMI)
ARMv5: 2002 (ARM7E, ARM9E)
ARMv6: 2002 (ARM11)
ARMv7: 2004 (Cortex-M), 2005 (Cortex-R, Cortex-A)
ARMv8: 2013 (Cortex-A5x)

ARM7/ 9/10/11, Cortex-R and Cortex-A are very similar and nearly binary compatible. Cortex-M is a different architecture and not binary compatible with the others.
2 Cortex-M Processors

ARM® Cortex® Processors across the Embedded Market

- Cortex®-M processors
  - MCU + DSP
  - Smallest footprint / lowest power

- Cortex®-R processors
  - RTOS
  - Highest performance / real-time

- Cortex®-A processors
  - Rich OS
  - Highest performance

© MicroConsult - MicroElectronics Consulting & Training GmbH
Scalable and Low-Power Technology for any Embedded Market.

The ARM Cortex-M processor family is a range of scalable and compatible, energy efficient, easy to use processors designed to help developers meet the needs of tomorrow’s smart and connected embedded applications. Those demands include delivering more features at a lower cost, increasing connectivity, better code reuse and improved energy efficiency. The Cortex-M family is optimized for cost and power sensitive MCU and mixed-signal devices for applications such as Internet of Things, connectivity, smart metering, human interface devices, automotive and industrial control systems, domestic household appliances, consumer products and medical instrumentation.

2 stage pipeline in Cortex-M0+ Processor for state of the art energy efficiency
3 stage pipeline in Cortex-M0 for a very compact 32-bit embedded processor
3 stage enhanced pipeline in Cortex-M3 and Cortex-M4 processors for high performance embedded system while providing low power advantages
6 stage superscalar pipeline in Cortex-M7 processor for unmatched performance for embedded processors
Cortex-M Processors

ARM® Cortex®-M0+

- Nested Vectored Interrupt Controller
- Wake Up Interrupt Controller Interface
- Memory Protection Unit
- AHB-lite Interface
- Low Latency I/O Interface
- Data Watchpoint
- Breakpoint
- Debug Access Port

ARM® Cortex®-M0

- Nested Vectored Interrupt Controller
- Wake Up Interrupt Controller Interface
- CPU
- AHB-lite Interface
- Data Watchpoint
- Breakpoint
- Debug Access Port

© MicroConsult - MicroElectronics Consulting & Training GmbH
2.1 Cortex-M Block Diagram

Cortex-M0 Block Diagram

NVIC: Nested Vectored Interrupt Controller

3 Stage Pipeline

**Performance Efficiency:** 0.87 / 1.02 / 1.27 DMIPS/MHz (original (K&R) v2.1 of Dhrystone)

The first value abides by all of the “ground rules” laid out in the Dhrystone documentation.

The second value permits inlining of functions, not just the permitted C string libraries.

The third value additionally permits simultaneous ("multi-file") compilation.
Cortex-M0+ Block Diagram

Interrupt Controller

NVIC: Nested Vectored Interrupt Controller

Performance Efficiency: 0.95 / 1.11 / 1.36 DMIPS/MHz

The memory map of a typical system is partitioned into logical regions. Each region may require different memory attributes:

- Access permissions
  - Read/Write permissions for non-privileged/privileged modes
- Memory types
  - Caching/Buffering and access ordering rules for memory accesses

 MPU: Memory Protection Unit

Cortex-M0+ Block Diagram

2 Stage Pipeline
WIC: Wakeup Interrupt Controller
FPU: Floating Point Unit
ETM: Embedded Trace Macrocell
ITM: Instrumentation and Trace Macrocell
MPU: Memory Protection Unit
M4 Performance Efficiency: Without FPU: 1.25 / 1.52 / 1.91 DMIPS/MHz
With FPU: 1.27 / 1.55 / 1.95 DMIPS/MHz

SysTick: integrated System timer

3 Stage Pipeline

M3 Performance Efficiency: 1.25 / 1.50 / 1.89 DMIPS/MHz

M4 Performance Efficiency: Without FPU: 1.25 / 1.52 / 1.91 DMIPS/MHz
With FPU: 1.27 / 1.55 / 1.95 DMIPS/MHz
Cortex-M Processors

- WIC: Wakeup Interrupt Controller
- FPU: Floating Point Unit
- ETM: Embedded Trace Macrocell
- ECC: Error Correction Code
- MPU: Memory Protection Unit
- TCM: Tightly Coupled Memory

- AMBA: Advanced Microcontroller Bus Architecture
- AXI: AMBA Extended Interface
- AHB: AMBA Highspeed Bus

- AXI-M: AXI Master Interface
- AHB-S: AHB Slave Interface
- AHB-P: AHB Peripheral Interface
Cortex-M7 Block Diagram

MPU: Memory Protection Unit
ITCM: Instruction Tightly Coupled Memory
DTCM: Data Tightly Coupled Memory
Instruction and Data Cache

6-stage dual issue pipeline

Performance Efficiency: 2.14 / 2.55 / 3.23 DMIPS/MHz
2.2 Register-Set Overview Cortex-M

All registers are 32 bit wide registers.
2.3 Cortex-M Instruction Set

Cortex-M Instruction Set

- DSP (SIMD, fast MAC)
- Advanced data processing
- Bit field manipulations
- General data processing
- I/O control tasks

Cortex-M0/M0+/M1
Cortex-M3
Cortex-M4/Cortex-M7

© MicroConsult - MicroElectronics Consulting & Training GmbH
3 Cortex-R Processors

The ARM® Cortex®-R real-time processors offer high-performance computing solutions for embedded systems where reliability, high availability, fault tolerance, maintainability and real-time responses are required.

Cortex-R Series

The ARM® Cortex®-R real-time processors offer high-performance computing solutions for embedded systems where reliability, high availability, fault tolerance, maintainability and real-time responses are required. The Cortex-R series processors provide fast time-to-market through proven technology shipped in hundreds of millions of products and leverages the vast ARM Ecosystem and global, local language, 24/7 support services to ensure rapid and low risk development.

There are many applications requiring the key Cortex-R series attributes of:

- **High performance**: Fast processing combined with a high clock frequency
- **Real-time**: Processing meets hard real-time constraints on all occasions
- **Safe**: Dependable, reliable systems with high error resistance
- **Cost effective**: Features for optimal for performance, power and area.
High performance: Fast processing combined with a high clock frequency
Real-time: Processing meets hard real-time constraints on all occasions
Safe: Dependable, reliable systems with high error resistance
Cost effective: Features for optimal for performance, power and area.
Cortex-R5 and Cortex-R7 can be configured as multi-cores.

**Accelerator Coherence Port (ACP)**
The Accelerator Coherency Port (ACP) provides a mechanism for cache coherency with an external data source. Examples of such data sources are 3G/4G modems or a hard disk read channel that write data directly into the processor’s level-2 memory system. By writing this data through the ACP, the processor’s data cache is inspected using a micro-Snoop Control Unit (μSCU) and if the same data is currently in cache it is invalidated so that it is updated when the processor next accesses it. This cache coherency is transparent to the developer, obviating the need to monitor and maintain coherency through additional software overhead. It is estimated that this feature increases effective system performance by up to 25% compared to using a Cortex-R4 processor with software performing cache maintenance, whilst also increasing code reliability by removing the likelihood of software cache maintenance coding errors being introduced into the system.

**Snoop Control Unit (SCU)**
responsible for managing the interconnect, arbitration, communication, cache-to-cache and system memory transfers, cache coherence and other multicore capabilities for all MPCore technology enabled processors.

**Low-Latency Peripheral Port (LLPP)**
The first of these new features is a Low-Latency Peripheral Port (LLPP) which is an additional bus port intended specifically for fast peripheral reads and writes. It is implemented as an AMBA AXI port with an optional AMBA AHB port. By using the LLPP, the processor can always...
guarantee an immediate read or write to peripheral registers in a system where a bounded and deterministic response is required, ensuring that peripheral reads or writes are unaffected by cache refills and/or queued AMBA AXI bus transactions to main

**Low-Latency RAM (LLRAM)**
A key feature of the Cortex-R7 processor is the introduction of a new class of level-2 memory known as Low-Latency RAM (LLRAM). This RAM is connected through a dedicated AMBA3 AXI bus port and is intended to complement the Cortex-R7 processor’s internal TCM. Experience from fast real-time SoC system designs using the Cortex-R4 and Cortex-R5 processors has shown that TCM can limit performance as larger, and therefore slower, RAM arrays introduce wait state cycles. This limitation is exacerbated by the Cortex-R7 processor’s higher clock frequencies. Thus the Cortex-R7 processor’s TCM is organized as high-performance Harvard memory with separate ports for Instruction and Data TCM with RAM size limited to 128 KBytes. Meanwhile the LLRAM port provides for larger, flexible and unified Instruction and Data memory that is not blocked by transactions to the rest of level-2 memory on the main AMBA AXI bus port.
3.1 Cortex-R Block Diagram

VIC: Vectored Interrupt Controller
PMU: Memory Protection Unit
ITCM: Instruction Tightly Coupled Memory
DTCM: Data Tightly Coupled Memory
SCU: Snoop Control Unit
AMBA: ARM Microcontroller Bus Architecture

Cortex-R4 Single Processor, 40nm LP
Maximum clock frequency: Above 800MHz
Performance: 1.68 / 2.03 / 2.45 DMIPS/MHz
Total area (Including Core+RAM+Routing): From 0.45 mm²
Efficiency: From 37 DMIPS/mW

Cortex-R5 Single Processor, 40nm LP
Maximum clock frequency: Above 800MHz
Performance: 1.67 / 2.02 / 2.45 DMIPS/MHz
Total area (Including Core+RAM+Routing): From 0.45 mm²
Efficiency: From 37 DMIPS/mW

Cortex-R7 Single processor systems, 28nm HPM
Maximum Clock frequency: Above 1.5 GHz
Performance: 2.50 / 2.90 / 3.77 DMIPS/MHz
Total area (Including Core+RAM+Routing): From 0.33 mm²
Efficiency: From 46 DMIPS/mW
3.2 Cortex-R Safety Features

- **Lock Step**, CPU Compare Module for Cortex-R4F (CCM-R4F) module compares the output of both Cortex-R4F CPUs running in lock step.

- **ESM**, Error Signaling Module

- **PBIST**, Programmable Built-In Self Test (PBIST) Module

- **STC**, CPU Self Test Controller Module

- **CRC**, Cyclic Redundancy Check Controller Module

- **ECC**, Error Correction Code

- **MPU**, Memory Protection Unit
3.3 Cortex-R Register Banks

- In each mode, the core can access:
  - a particular set of 13 general purpose registers (r0 - r12).
  - a particular r13 - which is typically used as a stack pointer. This will be a different r13 for each mode, so allowing each exception type to have its own stack.
  - a particular r14 - which is used as a link (or return address) register. Again this will be a different r14 for each mode.
  - r15 - whose only use is as the Program counter.

- The CPSR (Current Program Status Register) - this stores additional information about the state of the processor.

- And finally in privileged modes, a particular SPSR (Saved Program Status Register). This stores a copy of the previous CPSR value when an exception occurs. This combined with the link register allows exceptions to return without corrupting processor state.
4 Cortex-A Processors

Cortex-A Series

The ARM® Cortex®-A series of applications processors provide a range of solutions for devices undertaking complex compute tasks, such as hosting a rich Operating System (OS) platform, executing a user interface and supporting software applications. Cortex-A series processors can be found in a range of the highest performing consumer devices, including a spectrum of smartphones from ultra-low-cost to high-end flagship devices, mobil computing platforms, digital TVs, and set-top boxes, but can also be found in enterprise networking, printers and server solutions.
The ARM® Cortex®-A5 processor is the smallest, lowest cost and lowest power ARMv7 application processor, ideal as a stand-alone processor within current and future generations of smart wearable devices.

The ARM® Cortex®-A7 MPCore™ processor is the most power-efficient application processor ARM has ever developed, and dramatically extends ARM’s low-power leadership in entry-level smartphones, tablets, high-end wearables and other advanced mobile devices.

The ARM® Cortex®-A8 processor, based on the ARMv7 architecture, has the ability to scale in speed from 600MHz to greater than 1GHz. The Cortex-A8 processor can meet the requirements for power-optimized mobile devices needing operation in less than 300mW; and performance-optimized consumer applications requiring 2000 Dhrystone MIPS.

The ARM® Cortex®-A9 processor is the power-efficient and popular high performance choice in low power or thermally constrained cost-sensitive devices.

The ARM® Cortex®-A15 MPCore™ processor is today’s high-performance engine for your highly connected device. This processor delivers unprecedented flexibility and processing capability.

The ARM® Cortex®-A17 processor is the most efficient mid-range 32 bit solution targeted at smartphones and tablets and delivers today’s premium user experience in tomorrow’s mid-range mobile and consumer devices.
ARM Cortex-A12 is now also referred to as the ARM Cortex-A17.

**big.LITTLE**
The performance and energy efficiency of ARM Cortex-A series processors is enhanced by ARM big.LITTLE technology. By pairing a high-performance processor with an energy-efficient processor, tasks are instantaneously migrated between them, ensuring that the right processor is selected for the right job. Current big.LITTLE configurations pair the Cortex-A7 with either the Cortex-A15 or Cortex-A17 processors, and the Cortex-A53 with the Cortex-A57 processor.
4.1 Cortex-A Block Diagram

GIC: Generic Interrupt Controller
VIC: Vectored Interrupt Controller
MMU: Memory Management Unit
PMU: Memory Protection Unit
ITCM: Instruction Tightly Coupled Memory
DTCM: Data Tightly Coupled Memory
AMBA: ARM Microcontroller Bus Architecture
4.2 Pipeline
4.3 MMU

CORTEX-A8 has separated data and instruction TLB (Translation Lookaside Buffer). Both use the same table tree.
### 4.4 Cortex-A Register Banks

<table>
<thead>
<tr>
<th>User/Sys</th>
<th>FIQ</th>
<th>IRQ</th>
<th>SVC</th>
<th>Undef</th>
<th>Abort</th>
<th>Mon</th>
<th>Hyp</th>
</tr>
</thead>
<tbody>
<tr>
<td>r0-r7</td>
<td>User mode r0-r12</td>
<td>User mode r0-r12</td>
<td>User mode r0-r12</td>
<td>User mode r0-r12</td>
<td>User mode r0-r12</td>
<td>User mode r0-r12</td>
<td>User mode r0-r12</td>
</tr>
<tr>
<td>r8</td>
<td>r8</td>
<td>r8</td>
<td>r8</td>
<td>r8</td>
<td>r8</td>
<td>r8</td>
<td>r8</td>
</tr>
<tr>
<td>r9</td>
<td>r9</td>
<td>r9</td>
<td>r9</td>
<td>r9</td>
<td>r9</td>
<td>r9</td>
<td>r9</td>
</tr>
<tr>
<td>r10</td>
<td>r10</td>
<td>r10</td>
<td>r10</td>
<td>r10</td>
<td>r10</td>
<td>r10</td>
<td>r10</td>
</tr>
<tr>
<td>r11</td>
<td>r11</td>
<td>r11</td>
<td>r11</td>
<td>r11</td>
<td>r11</td>
<td>r11</td>
<td>r11</td>
</tr>
<tr>
<td>r12</td>
<td>r12</td>
<td>r12</td>
<td>r12</td>
<td>r12</td>
<td>r12</td>
<td>r12</td>
<td>r12</td>
</tr>
<tr>
<td>r13 (sp)</td>
<td>r13 (sp)</td>
<td>r13 (sp)</td>
<td>r13 (sp)</td>
<td>r13 (sp)</td>
<td>r13 (sp)</td>
<td>r13 (sp)</td>
<td>r13 (sp)</td>
</tr>
<tr>
<td>r14 (lr)</td>
<td>r14 (lr)</td>
<td>r14 (lr)</td>
<td>r14 (lr)</td>
<td>r14 (lr)</td>
<td>r14 (lr)</td>
<td>r14 (lr)</td>
<td>r14 (lr)</td>
</tr>
<tr>
<td>r15 (pc)</td>
<td>r15 (pc)</td>
<td>r15 (pc)</td>
<td>r15 (pc)</td>
<td>r15 (pc)</td>
<td>r15 (pc)</td>
<td>r15 (pc)</td>
<td>r15 (pc)</td>
</tr>
<tr>
<td>cpsr</td>
<td>cpsr</td>
<td>cpsr</td>
<td>cpsr</td>
<td>cpsr</td>
<td>cpsr</td>
<td>cpsr</td>
<td>cpsr</td>
</tr>
<tr>
<td>spsr</td>
<td>spsr</td>
<td>spsr</td>
<td>spsr</td>
<td>spsr</td>
<td>spsr</td>
<td>spsr</td>
<td>spsr</td>
</tr>
</tbody>
</table>

- **Security Extension** TrustZone supporting cores introduce an additional “Secure Monitor Mode” Mon.
- **Virtualization Extension** supporting cores introduce an additional “Hypmode” for Hypervisor execution, Hyp.
- additional dedicated Exception Link Register (ELR), stores preferred return address on exception entry
- New instruction –ERET– for exception return from HYP mode
- The temp register is used to hold data for multi-cycle instructions.
- **HB** is the Jazelle X Handler Base Address register for sourcing of the HB instruction.
- Saved Program Status Register (SPSR) are the shadow copy of Current Program Status Register (CPSR) for return from interrupt. CPSR is copied to SPSR upon switching processor mode.
4.5 Global Interrupt Controller, GIC

Used in multi-core systems
- Shared distributor interface
- Individual processor interfaces

Interrupt Distributor controls:
- which core each interrupt targets
- the priority of each interrupt

Processor interface controls:
- Interrupt masking
- Pre-emption masking

**For hardware interrupt:**
- Only one targeted core handles interrupt (1-N model)

**For software interrupts:**
- Each targeted core has to handle the interrupt (N-N model)
4.6 Multiple Cores & Snoop Control Unit, SCU

The SCU is responsible for the cache coherence between the multiple cores. It has a copy of the tag RAMs of each CPU. It supports a local timer and watchdog for each CPU and a global timer. Addresses to the L2 Memory can be filtered between Master0 and Master1 interface. External Masters, like DMA or other CPUs, can be connected to the ACP, in doing so the cache coherency is assured.

CPU0, Primary CPU
CPUx, Secondary CPUs
Global Timer
Private Timers
Private Watchdogs
Interrupt Controller, GIC
Snoop Control Unit, SCU
Master 0 AXI Interface
Master 1 AXI Interface
Accelerator Coherency Port, ACP
4.7 TrustZone

If a hacker infiltrates into your system it can be detected and eliminated before it can spy on your secrets.
Introduces the idea of separate Secure and Normal (Non-secure) 'worlds'

- A normal platform OS and its processes execute in the Normal world
- A small Secure kernel executes in the Secure world, providing services which can be requested from Normal world

Effectively an extra level of protection compared to standard privileged/non-privileged modes

The entry to monitor can be triggered by software executing a dedicated instruction, the Secure Monitor Call (SMC) instruction, or by a subset of the hardware exception mechanisms. The IRQ, FIQ, external Data Abort, and external Prefetch Abort exceptions can all be configured to cause the processor to switch into monitor mode.
4.8 Virtualization- and Large Physical Address Extension

- The complexity of software increases, with it the requirement for multiple software environments to be available on the same physical processor simultaneously.
- Software applications that require separation for reasons of isolation, robustness or differing real-time characteristics need a virtual processor exhibiting the required functionality.
- The ARM Architecture
  - Virtualization Extension and
  - Large Physical Address Extension (LPAE) enable the efficient implementation of virtual machine hypervisors for ARM architecture compliant processors.

![Diagram of Virtualization- and Large Physical Address Extension](image-url)
5 64 Bit Cortex-A Processors

Cortex-A57 Processor
The ARM® Cortex®-A57 processor is ARM’s highest performing processor, designed to further extend the capabilities of future mobile and enterprise computing applications including compute intensive 64-bit applications such as high end computer, tablet and server products. The processor can be implemented individually or paired with the Cortex-A53 processor into an ARM big.LITTLE configuration that enables scalable performance and optimal energy-efficiency.

Cortex-A53 Processor
The ARM® Cortex®-A53 processor is our most power-efficient ARMv8 processor capable of seamlessly supporting 32-bit and 64-bit code. It makes use of a highly efficient 8-stage in-order pipeline balanced with advanced fetch and data access techniques for performance. It fits in a power and area footprint suitable for entry-level smartphones, and is at the same time capable of delivering high aggregate performance in scalable enterprise systems via high core density.

AArch64 64-bit execution state
- 31 * 64-bit general purpose registers
- 64-bit Program Counter (PC)
- 64-bit Stack Pointer (SP)
- 64-bit Exception Link Registers (ELRs)
- ARMv8 exception model

AArch32 32-bit execution state
- backwards-compatible with ARMv7-A
- 13 * 32-bit general purpose registers
- 32-bit Program Counter (PC)
- 32-bit Stack Pointer (SP)
- 32-bit Exception Link Registers (ELRs)
- ARMv7 exception model

Source: www.arm.com
Cortex-A series processors can be found in a range of the highest performing consumer devices. Smartphones, mobile computing platforms, digital TVs, set-top boxes, enterprise networking, printers and server solutions.
6 More Informations

More informations about the Cortex Architecture you can get from: www.microconsult.com

Available seminars for ARM and Cortex:

- ARM7/9/10/11: Architektur und Embedded Programmierung
- Cortex™-M7, M4, M3, M1, M0 (ARM): Architektur und Embedded Programmierung
- Cortex™-R4 (ARM): Architektur und Embedded Programmierung
- Cortex™-A5/7/8/9/15 (ARM): Architektur und Embedded Programmierung

The seminars are available in German and English language.

This presentation can be downloaded from www.microconsult.com.