Lecture 4: ARM Cortex-M4 Architecture

Seyed-Hosein Attarzadeh-Niaki

Review

- ARM processors
- ARM Cortex-M processor cores
- STM32 microcontrollers
  - Architecture

Outline

- Cortex M4 core and special registers
- Operating states and modes
- Memory system
- Debug and trace
- Reset and the reset sequence

Introduction to the Architecture

- Cortex-M3&4 are based on ARMv7-M
  - The Architecture Reference Manual is a massive document
  - But, you only need to have a basic understanding of
    - the programmer's model
    - how exceptions (such as interrupts) are handled
    - the memory map
    - how to use the peripherals
    - how to use the software driver library files from the microcontroller vendors

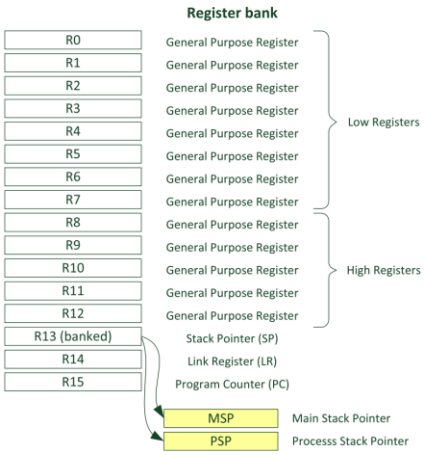
ARM Cortex-M4 Core Registers

R0-R12
- General purpose registers
- R0-R7: low registers
- R8-R12: high registers
- Initial value: undefined

R13
- Stack pointer
- Physically two pointers
  - Main Stack pointer
  - Process Stack Pointer
  - 32-bit aligned (2 low bits: 0)

R14
- Link register
- Return address for subroutine call

R15
- Program Counter
- Read: PC+4
- Write: Branch

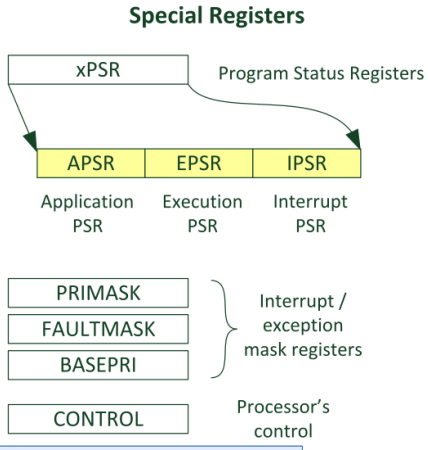


ARM Cortex-M4 Special Registers

Program Status Register
- Composed of three status registers
  - Application PSR (APSR)
  - Execution PSR (EPSR)
  - Interrupt PSR (IPSR)

Interrupt/exception mask registers
- PRIMASK
- FAULTMASK
- BASEPRI

CONTROL register



Access method: Special instructions (Not memory-mapped)

ARM Cortex-M4
Program Status Register

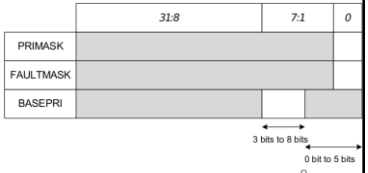
	31	30	29	28	27	26:25	24	23:20	19:16	15:10	9	8	7	6	5	4:0
APSR	N	Z	C	V	Q				GE							
IPSR												Exception Number				
EPSR						ICI/IT	T			ICI/IT						

- Program Status Register (PSR) is three views of same register
  - Application PSR (APSR)
  - Interrupt PSR (IPSR)
  - Execution PSR (EPSR)

Bit	Description
N	Negative flag
Z	Zero flag
C	Carry (or NOT borrow) flag
V	Overflow flag
Q	Sticky saturation flag (not available in ARMv6-M)
GE[3:0]	Greater-Than or Equal flags for each byte lane (ARMv7E-M only; not available in ARMv6-M or Cortex®-M3).
ICI/IT	Interrupt-Continuable Instruction (ICI) bits, IF-THEN instruction status bit for conditional execution (not available in ARMv6-M).
T	Thumb state, always 1; trying to clear this bit will cause a fault exception.
Exception Number	Indicates which exception the processor is handling.

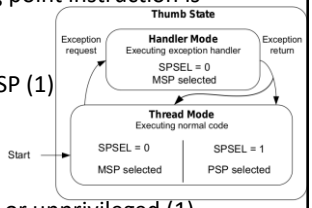
ARM Cortex-M4
Interrupt/exception mask registers

- PRIMASK - Exception mask register
  - Bit 0: PM Flag
    - Set to 1 to prevent activation of all exceptions with configurable priority
  - Access using CPS, MSR and MRS instructions
  - Use to prevent data race conditions with code needing atomicity
- FAULTMASK – HardFault exception mask register
  - Similar to PRIMASK but also blocks HardFault exception
  - Equivalent to raising the current exception priority level to -1
- BASEPRI
  - Mask interrupts based on priority level

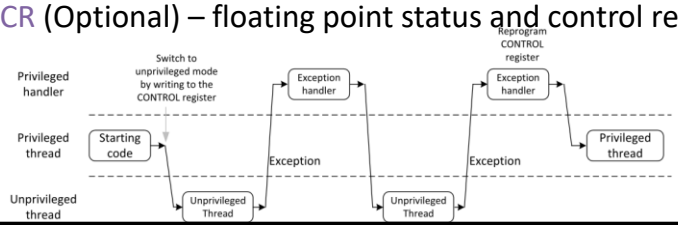


ARM Cortex-M4 Control Register

- CONTROL
  - Bit 2: FPCA flag
    - Floating point context active: not using(0) or need to save floating point registers(1)
    - This bit will be set automatically when floating point instruction is executed, and is 0 by default.
  - Bit 1: SPSEL flag
    - Selects SP when in thread mode: MSP (0) or PSP (1)
    - With OS environment, Threads use PSP
    - OS and exception handlers (ISRs) use MSP
  - Bit 0: nPRIV flag
    - Defines whether thread mode is privileged (0) or unprivileged (1)

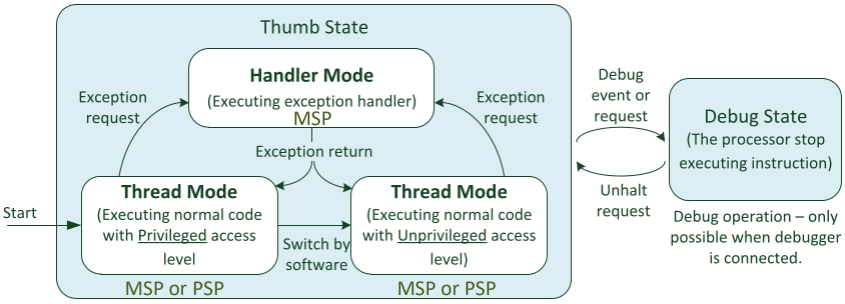


- FPSCR (Optional) – floating point status and control registers



9

Operating States and Modes

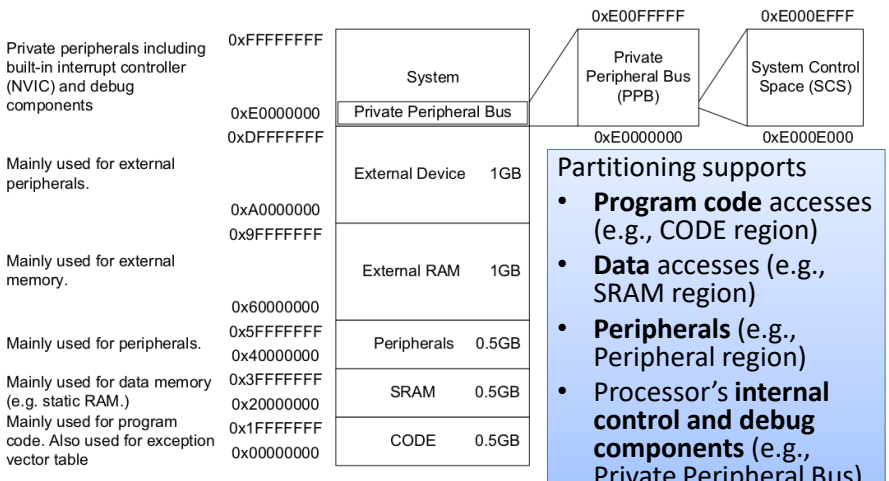


- Which SP is active depends on operating mode, and SPSEL (CONTROL register bit 1)
  - SPSEL == 0: MSP (Main Stack Pointer)
  - SPSEL == 1: PSP (Process Stack Pointer)
- Similarly, the privileged level in Thread mode depends on the nPRIV (bit 0 of CR)
  - nPRIV == 0: privileged level: full access to resources
  - nPRIV == 1: unprivileged level: limited access to resources

Memory System

- 4GB linear address space
- Architecturally defined memory map
  - Allows the processor design to be optimized for performance
- Support for little endian and big endian
- Bit band accesses (optional)
- Write buffer
- Memory Protection Unit (Optional)
  - Defines access permissions for eight programmable regions
- Unaligned transfer support

Memory Map



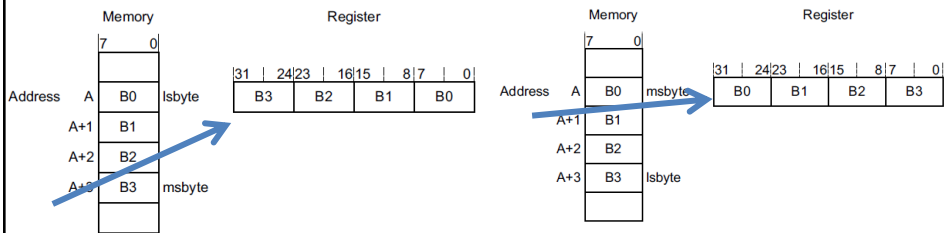
Partitioning supports

- **Program code** accesses (e.g., CODE region)
- **Data** accesses (e.g., SRAM region)
- **Peripherals** (e.g., Peripheral region)
- Processor's **internal control and debug components** (e.g., Private Peripheral Bus)

Reminder: Endianness

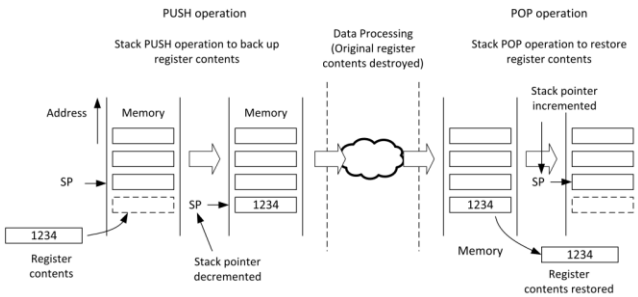
- For a multi-byte value, in what order are the bytes stored?
- Little-Endian: Start with least-significant byte
- Big-Endian: Start with most-significant byte

- Cortex-M4 support both Little-Endianness and Big-Endianness
- Instructions are always little-endian
- Loads and stores to Private Peripheral Bus are always little-endian
- Data: Depends on implementation, or from reset configuration
  - ST processors are little-endian



Stack Memory

- Stack pointers: R13
- Used for temporary storage of registers in functions, passing info to functions, local variables, and hold processor status and register values in case of exception.
- Cortex-M processors use “full-descending stack”
  - PUSH: first decrement the SP, then store the value in the memory location pointed by SP
  - POP: the value of the memory location pointed by SP is read, then the value of SP is incremented

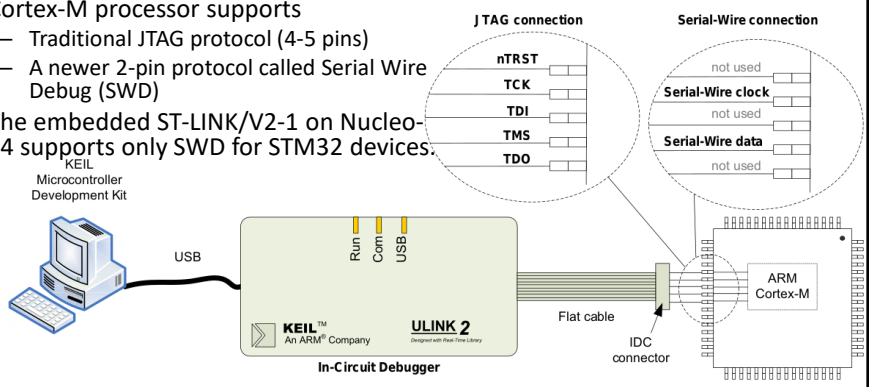


System Control Block (SCB)

- Contains various registers for
  - Controlling processor configurations (e.g., low power modes)
  - Providing fault status information (fault status registers)
  - Vector table relocation (VTOR)
- SCB is memory-mapped
- SCB registers are accessible from the System Control Space (SCS)
  - More details later

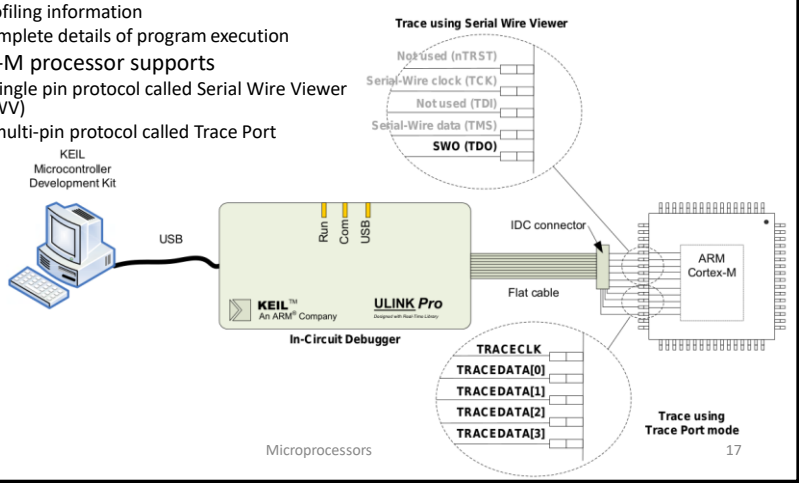
Debug

- Debug interface allows a debug adaptor to connect to a Cortex-M microcontroller to
  - control the debug features
  - access the memory space on the chip
- Cortex-M processor supports
  - Traditional JTAG protocol (4-5 pins)
  - A newer 2-pin protocol called Serial Wire Debug (SWD)
- The embedded ST-LINK/V2-1 on Nucleo-64 supports only SWD for STM32 devices.



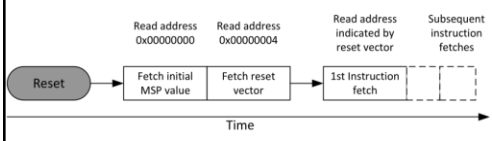
Trace

- Trace interface is used to collect information from the processor during runtime such as
  - Data
  - Event
  - Profiling information
  - Complete details of program execution
- Cortex-M processor supports
  - A single pin protocol called Serial Wire Viewer (SWV)
  - A multi-pin protocol called Trace Port



Reset and The Reset Sequence

- Three types of reset
  - Power on reset: reset everything in the microcontroller
  - System reset: reset just the processor and peripherals, but not the debug support component
  - Processor reset: reset the processor only
- After reset
  - Cortex-M read the first two words from the memory
  - Sets up the MSP and the Program Counter (PC) with these values
- The C startup code will also update the value of the MSP before entering the main program **`main()`**



Initial Stack Pointer value and Initial Program Counter value example