#HW, cec32x_20_HW1_ARM_architecture, 110 points.

# HW 1. ARM architecture

Prob 1-1 (30 points). There are three kinds of demo boards from ST Microelectronics for Cortex-M based MCUs: Nucleo boards, Discovery boards, and Eval boards. Compare the pros and cons of these boards in terms of functionalities, cost, number of onboard hardware, and compatibility to other boards.

Prob 1-2 (30 points). Suppose a specific MCU has the following size of memories: 2 M byte of flash, starting from 0x0800_0000, 256 k byte of SRAM starting from 0x2000_0000, and 8 k byte registers for GPIOs, start at 0x4001_0000. (Note that 0x is the prefix for hexadecimal numbers.) Draw the memory map similar to Fig 1-9 of the book. Note that you need to clearly label the starting and ending addresses of each memory/register block.

Prob 1-3 (10 points). Prob 6 of chapter 1 of the textbook.

Prob 1-4 (10 points). Prob 9 of chapter 1 of the textbook.

Prob 1-5 (30 points). Prob 12 of chapter 1 of the textbook.

Soln to Prob 1-1.

This problem is an open problem, but we are looking to answers covering the following:

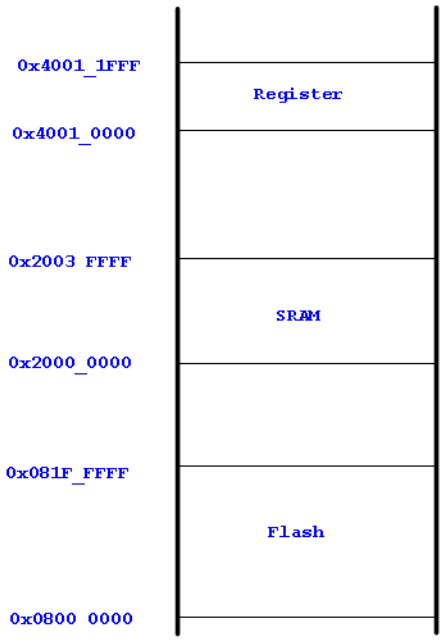
* Functionality:
  - Eval boards are the best with the most functionalities builtin on the boards.
  - Discovery boards have many functionalities, just less than that of Eval boards.
  - Nucleo boards only provide the most basic functionalities.

* Cost:
  - Nucleo boards are very affordable.
  - Discovery boards have similar prices as the Nucleo boards.
  - Eval boards are much more expensive.

* Number of onboard hardware components: The number of onboard hardware components is directly related to the functionality of the board. As such,
  - Eval boards have the most onboard hardware components. Many of the major functionalities of the MCU can be evaluated via some onboard components.
  - Discovery boards have many onboard hardware components as well, just not as many as Eval boards.
  - Nucleo boards does not provide many onboard hardware components except the very basic one like a push button and a few LEDs.

* compatibility to other boards:
  - Nucleo boards are compatible with Arduino, and hence they have the best compatibility.
  - Discovery boards are somehow compatible with other similar Discovery boards only.
  - Eval boards are very specific to the special functionalities of certain MCUs and have the worst compatibility.

Soln to Prob 1-2:




(If you want to see how the numbers are calculated, here are the details:
1. 2 MB => 2 * 2^(20) = 2 * 2^4 * 2^(16) => 0x20_0000 in Hex. The address of the last memory in this block is then 0x0800_0000 + 0x20_0000 - 1 = 0x081F_FFFF.
2. 256 kB => 2^8 * 2^(10) = 2^2 * 2^(16) => 0x4_0000 in Hex. The address of the last memory in this block is then 0x2000_0000 + 0x4_0000 - 1 = 0x2003_FFFF.
3. 8 kB => 2^3 * 2^(10) = 2 * 2^(12) => 0x2000 in Hex. The address of the last memory in this block is then 0x4001_0000 + 0x2000 - 1 = 0x4001_1FFF.
)

Prob 1-3.

6. What are the advantages and disadvantages of Von Neumann architecture and Harvard architecture?

Soln to Prob 1-3

* Von Neumann:
  - Advantages:
    + Instruction memory and data memory share the same data bus so that the hardware is simple.
  - Disadvantages:
    + Loading of the instruction and data has to be done separately, and hence slows down the execution for the same clock rate.

* Harvard:
  - Advantages:
    + Instruction memory and data memory use different data buses and hence Loading of the instruction and data can be done simultaneously, leading to faster the execution for the same clock rate.
  - Disadvantages:
    + Two buses are used which makes the hardware more complex.

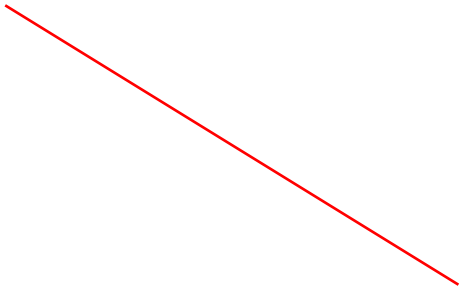
Just a note: Modern ARM-based MCUs use multiple buses (much more than TWO) in the system to improve the performance.

Prob 1-4.

9. Why can instruction memory and data memory share the same address bus in Cortex-M3, Cortex-M4, and Cortex-M7? In other words, why can we put the instruction memory and data memory in the same memory address space?

Soln to Prob 1-4:

All these MCUs use Harvard architecture in their core processor. They all use the same memory space for both the instruction memory and the data memory since the memory space is huge, 4 Gbyte in total, and we can put the instruction memory and data memory in different addresses. When these memories are in different addresses, we can use a hardware logic circuit to determine if the address is from the range of a instruction memory or data memory.



Prob 1-5.

12. Suppose a pipeline processor has three stages, as shown in Figure 1-14. Assume in a perfect scenario (such as no branch instructions, no data dependence between instructions, and no memory I/O waiting).

    a) How many clock cycles does it take to execute 10 instructions? What is the throughput (measured in instructions per cycle)?
    b) How many clock cycles does it take to execute $n$ instructions? If $n$ is sufficiently large, what is the throughput in terms of instructions per cycle?

Soln to Prob 1-5:

a) To execute 10 instructions we need 12 cycles, and the throughput is 10/12 = 0.833 instruction per cycle.

b) To execute n instructions we need n + 2 cycles, and the throughput is n / (n + 2) instruction per cycle.