1. How much memory and FLASH storage does the STM32F072R8 have? (section 1.2)

16 Kbytes of static RAM and 128 Kbytes of FLASH

1. What does the acronym “HAL” stand for? (section 1.3)

Hardware Abstraction Library

1. What is the STM32CubeMX program used for? (section 1.4)

Graphically configuring project parameters to generate a ready to use microvision project

1. Why can’t a “bare-metal” embedded application return from the main function? (section 2.2)

There is no operating system to take over managing low level system operations

1. In the system’s memory table, are the peripheral registers higher or lower in address than the SRAM? (section 2.3)

Higher

1. What information does each of the four main datasheets/manuals used in the labs provide? (section 2.4)

DM00090510 – STM Chip Datasheet: Provides device specific details for the processor. Pin connections and a list of available peripherals.

DM00051352 – Programming & Core manual: Provides information on the ARM core peripherals and assembly instruction sets.

DM00031936 – Peripheral Manual: Provides information on all peripherals available within an STM32F0 device. Our device does not contain every peripheral so this should be used with the chip datasheet.

DM00099401 – Discovery Board Manual: Provides schematics and tables that show the onboard devices and connectors attached to the STM32F0.

1. Why do STM32F0 devices not recognize inputs/outputs on a chip by physical pin numbering? (section 2.4.1)

Different chip packages can have different numbers of pins and the pin ordering between them can be inconsistent

1. What is the name of ST’s header file that defines names for the peripheral registers? (section 2.4.3)

stm32f072xb.h

1. What bitwise operator would you use to set a bit in a register? (section 2.5.1)

OR

1. What peripheral enables the system clock to other peripherals? (section 2.5.2)

Reset and Clock Control (RCC)

1. What peripheral do the HAL library delay functions use? (section 2.5.3)

The SysTick timer

1. Why should you avoid floating-point values on an STM32F0? (section 2.5.4)

It does not have hardware support for floating point math and must emulate it with large and slow code libraries