Lab 4: Pulse-width Modulation

Instructor:
Dr. Carl Latino
carl.latino@okstate.edu

Graduate Teaching Assistant:
Francisco E. Fernandes Jr.
feferna@okstate.edu

School of Electrical and Computer Engineering
Oklahoma State University
Fall 2019

Lab Assignment

- **Write an Assembly program that uses the System Timer Interrupts and Pulse-Width Modulation (PWM) to change the brightness of the red LED using the joystick as follow:**
  - **If the MIDDLE button is pressed, the Duty Cycle should be equal to 0% (the red LED will be OFF).**
  - **If the LEFT button is pressed, the Duty Cycle should be equal to 15%.**
  - **If the RIGHT button is pressed, the Duty Cycle should be equal to 40%.**
  - **If the UP button is pressed, the Duty Cycle should be equal to 65%.**
  - **If the DOWN button is pressed, the Duty Cycle should be equal to 100% (the red LED will ON with full brightness).**
- **The AHB clock (external clock) is set to 8 MHz, and a single PWM cycle must be set to exactly 0.02 seconds! It means that every second will contain 50 PWM cycles.**
- **Each PWM cycle should be further divided in 100 equal "chunks" of time. Thus, each chunk of time of a single PWM cycle will take 0.0002 seconds.**

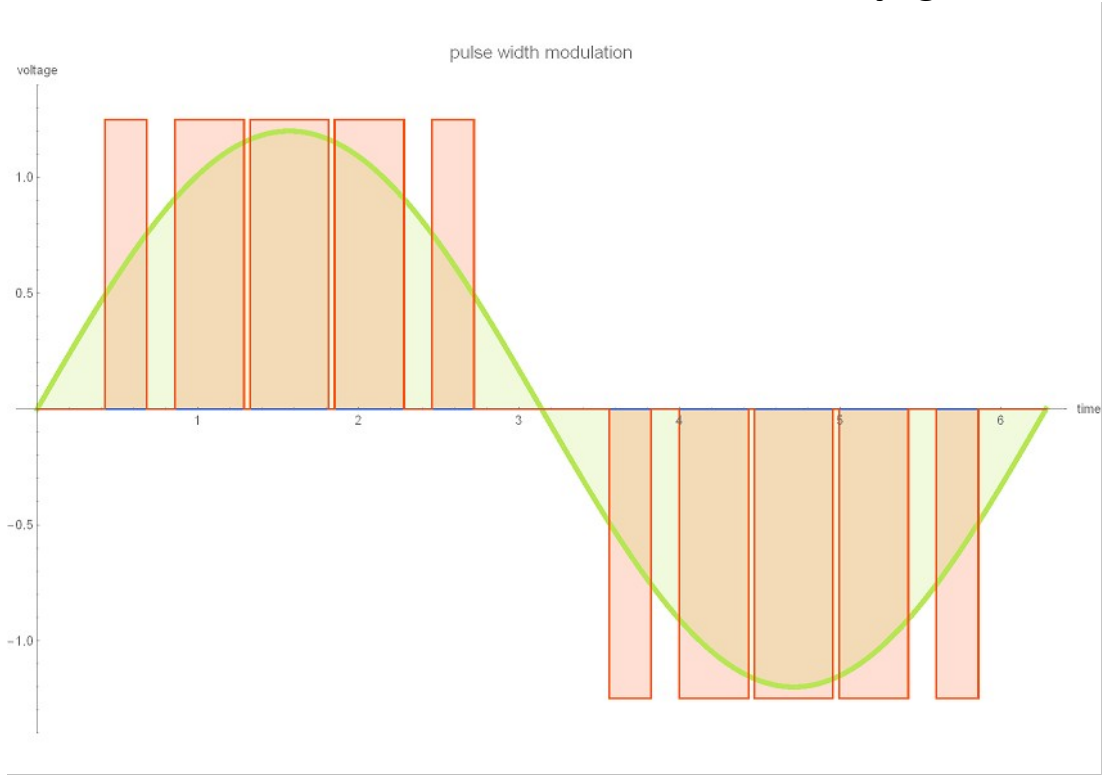
Schedule and Grading

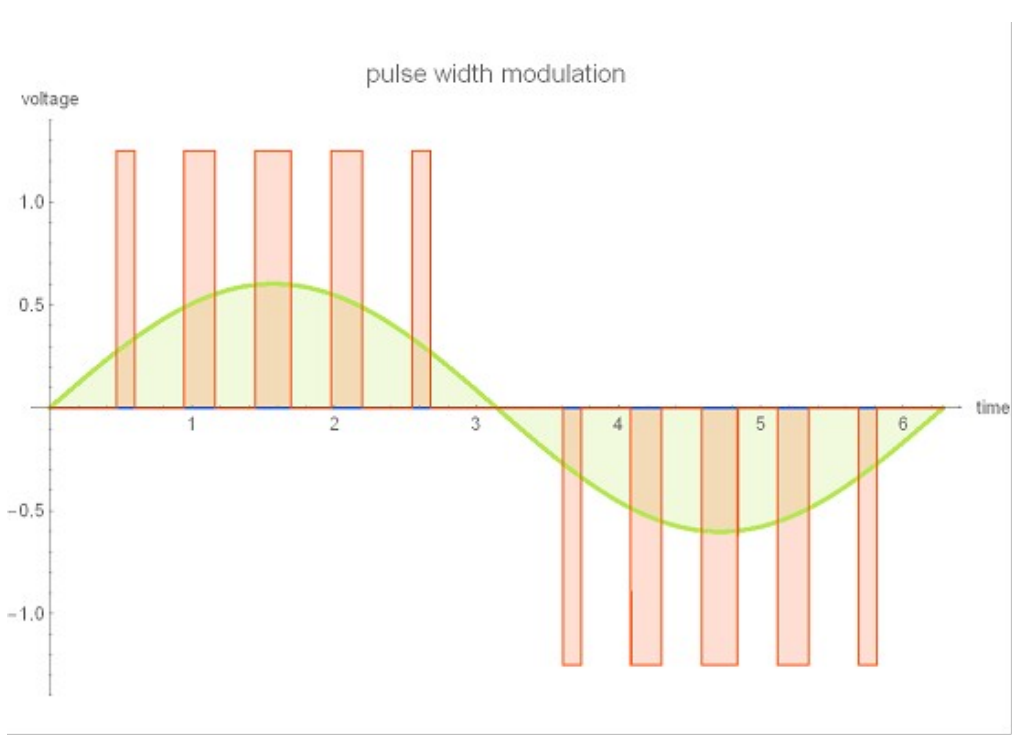
BY THE END OF YOUR LAB SECTION

- **You MUST demo a working LAB 4 on October 07, 2019 as follows:**
  - **Show the five different brightness of the red LED using the SMT32L4 Discovery board.**
  - **Show the five different duty cycles used in this lab with the help of the oscilloscope.**
  - **Show your code.**
- **Grading for LAB 4:**
  - **No pre-lab quiz! Read Chapter 15.3 to have a better understanding of PWMs.**
  - **Functionality and Correctness: 10 points.**
    - **All five brightness settings (0%, 15%, 40%, 65%, 100%) working:** 10 points.
    - **Between two and four brightness settings working:** 4 points.
    - **No PWM implementation:** 0 points.
- **Grading penalization:**
  - Students who disrupt the lecture by talking and not paying attention will lose 2 points in their lab 3's grade!
  - Students who do not follow the lab safety procedures (e.g. coming to lab with shorts and flip flops) will lose 1 points in their lab 3's grade!

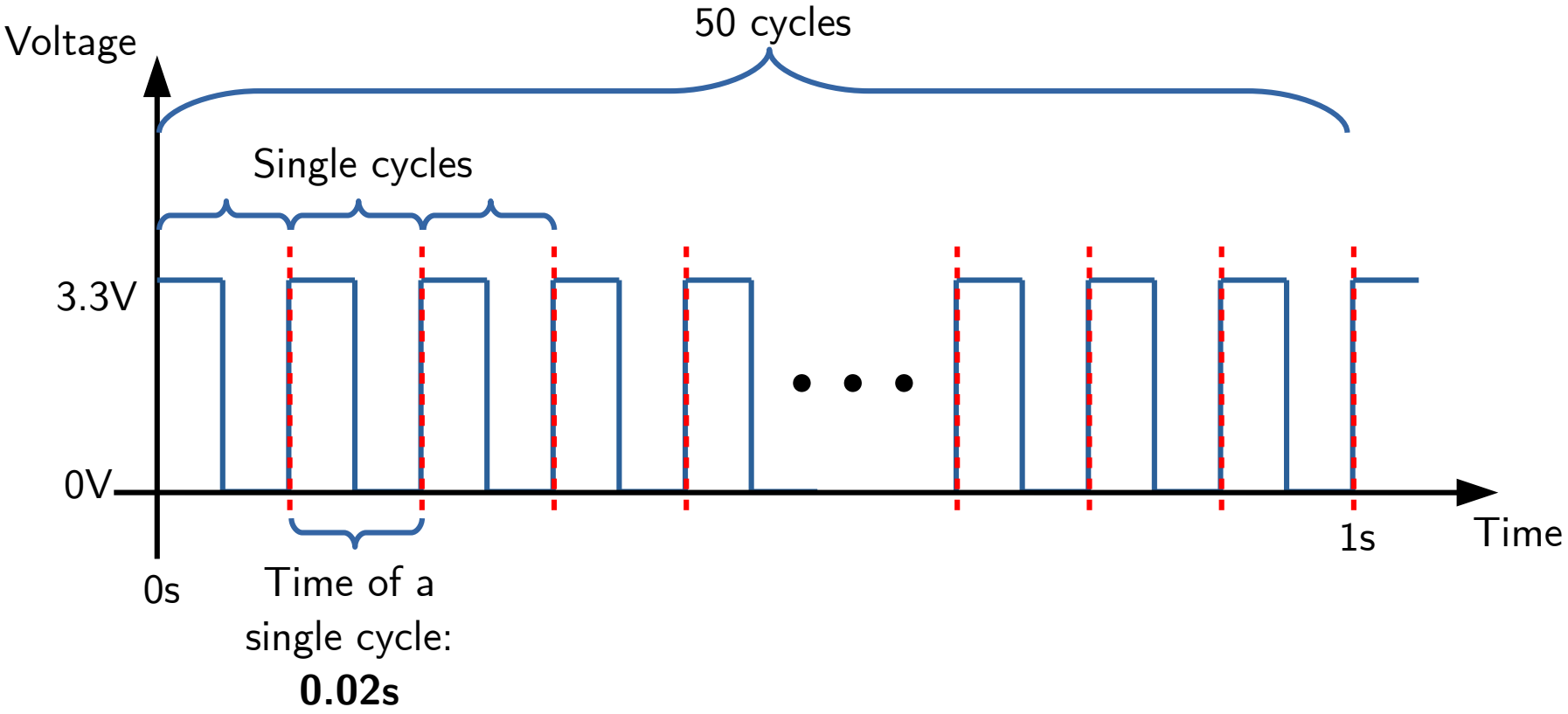
Pulse-Width Modulation (PWM)

- PWM can be used to simulate analog signals with only square waves.
- The STM32L4 Discovery Kit contains hardware to generate PWMs automatically.
- However, in this lab, we will manually generate a PWM to control the brightness of an LED.

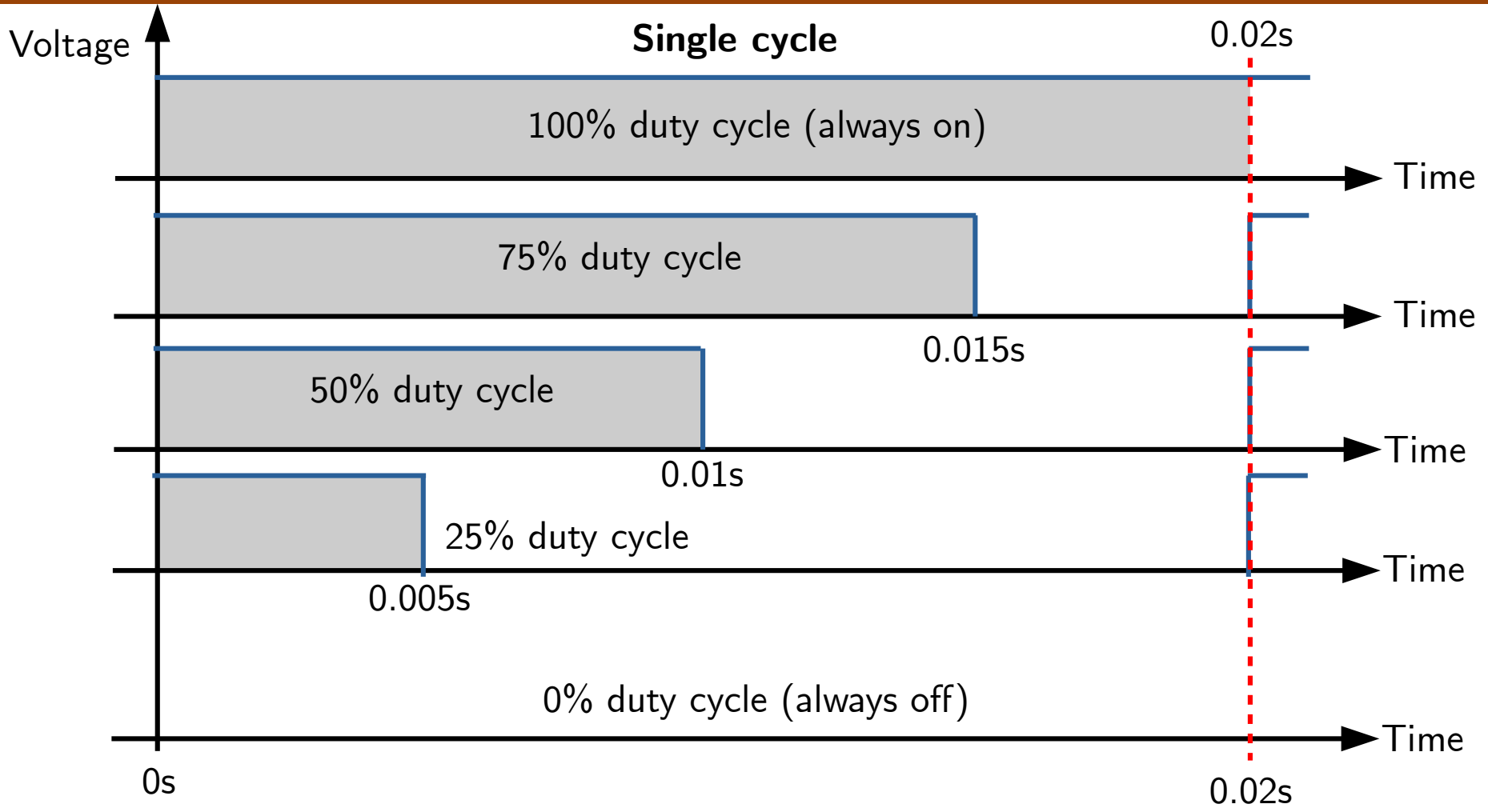




PWM for Lab 4


Voltage
50 cycles
Single cycles
3.3V
0V
0s
Time of a
single cycle:
0.02s
1s
Time

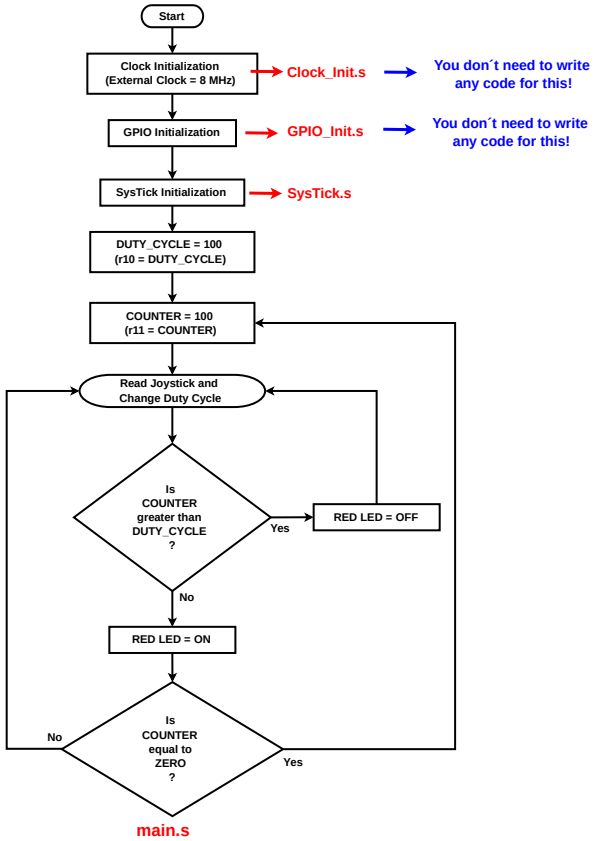
Duty Cycle for Lab 4

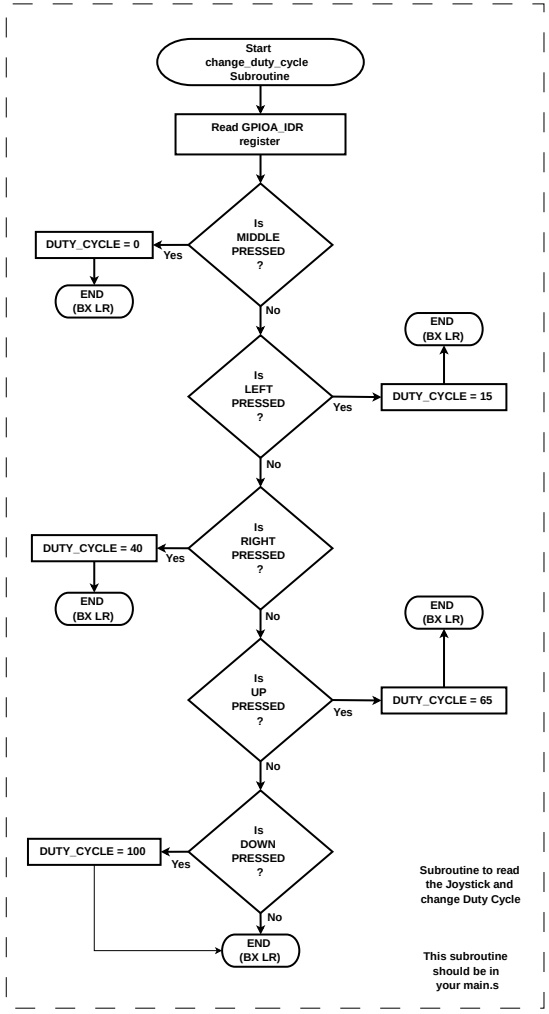


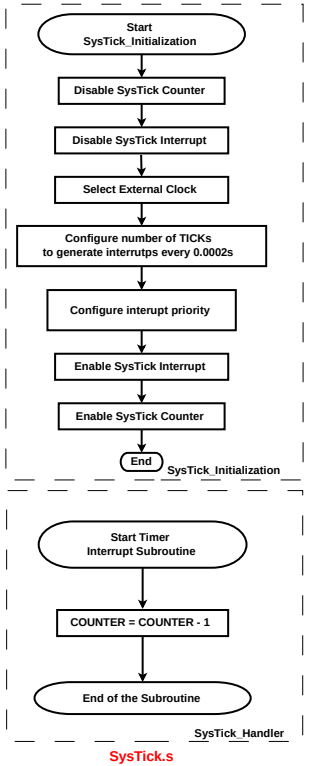
How Frequent Should the Interrupts Be?

- **AHB Clock = 8MHz**
  - **Clock used by the SysTick = 8 MHz/8 = 1 MHz**
- **Each PWM cycle is equal to 0.02 seconds (50Hz).**
- **If we want to be able to generate any duty cycle between 0% and 100% with increments of 1%, we need interrupts at every:**
  - **0.02/100 = 0.0002 seconds**
- **What should be the value of the SysTick_LOAD register (RELOAD value)?**
  - **SysTick_LOAD = (0.0002 x 1 x 10^6) - 1**
  - **SysTick_LOAD = 199**

Lab 4: Flowchart







Lab 4: Startup Code

- A startup code in a zip-file (filename: ***Lab 4 – Startup Code.zip***) is available on ***Canvas***. It contains the following files:
  - ***main.s*** → You have to write all missing code!
  - ***Clock_Init.s*** → You don't need to change anything in this file!
  - ***GPIO_Init.s*** → You don't need to change anything in this file!
  - ***SysTick.s*** → You have to write all missing code!
  - ***stm32l476xx_constants.s*** → You don't need to change anything in this file!
- **Download** and **EXTRACT** the startup code.
- Create a new project from scratch using the **STM32CubeIDE**.
- Move **ALL** files to you project's ***src*** folder, and follow the standard steps when we create new projects.