

# **Intro to ARM Assembly on Raspberry Pi 3**

## **Lab 1 Report TAMU CSCE 462**

Zach Hudnall (502) | Todd Christian  
(503)

### **Questions to Explore** (provided as prompt)

1. What is the range of voltage that represent logic low on raspberry Pi?
2. Which instruction set architecture is used in Raspberry Pi?
3. In this lab, we used assembly language to control the output to the GPIO pin on the Raspberry Pi in order to blink the LED. Do you have an alternative way to blink this external LED? If so, write down your answer.
4. What can you add to or change in the code in section 3.2 – 3.3 if you want to control the LED on/off time to different numbers? (For example LED on for 3 second and off for 4 second, repeat.) Can we replace function delay in our code?
5. Now, if we replace the external LED in this lab by an analog-in buzzer. Can we still produce music by this buzzer using the same way we do to the LED? If yes, provide your solution. If no, what needs to be changed in order to produce music?

### **Responses**

1. 0 - 1.19 V
2. 32-bit wordsize ARMv7
3. 555 timer with resistors and capacitors to provide a discrete synchronous function.
4. Yes, in the assembly code, changing the values for **delay\_time** function to a named variable vector. Iterating the vector with each individual numbers through the loop function, providing a delay of varying, but repetitive values.
5. Playing music isn't the same as turning a buzzer on and off. In order to operate the buzzer to play music would require specific values to adjust, not just the time, but the pitch of the buzzer (music requires pitch and timing). Using a PWM control on the GPIO port would allow this pitch change.

### **Source Code**

(See Git repository for latest version of source code)

<https://github.com/zhudnall/CSCE462/blob/master/Lab1/>