University of Texas at El Paso
Electrical and Computer Engineering Department

EE4376 Laboratory for Microprocessors II

LAB 03

Peripherals and Queues: ADC and PWM (LEDC)

Goals:

- With the help of the ADC and the LEDC API's, initialize the peripherals and create a task to perform the readings of the ADC every 100 millisecond.
- This task should feed the information into a queue.
- Create a task that synchronous to the queue to update the PWM's duty cycle based on the value read by the ADC.
- Modify the program in listing 1 to achieve these goals.

Bonus:

Add a port interrupt to stop and start the PWM signal. **+10**

Pre-Lab Questions:

- What is the function use to create a queue?
- What is the function use to send information into the

Written by Hector Mota. Modified by Dr. Erives & Mirza Elahi September 2021

queue?

- How do you read information from the queue?

```
#include <stdio.h>
#include "sdkconfig.h"
#include "freertos/FreeRTOS.h"
#include "freertos/task.h"
#include "freertos/queue.h"
#include <driver/ledc.h>
#include <driver/adc.h>

static xQueueHandle duty_queue = NULL;

void ADCtask(void *pvParameter)
{
        while(1)
        {
           vTaskDelay(100/portTICK_PERIOD_MS);
        }
}

void PWMtask(void *pvParameter)
{
        while(1)
        {
                ledc_set_duty();
                ledc_update_duty();
        }
}


void setADC()
{
  adc1_config_width();
  adc1_config_channel_atten(, );
}

void setPWM()
{
        ledc_timer_config_t timerConfig;
        timerConfig.duty_resolution = ;
        timerConfig.timer_num = ;
        timerConfig.freq_hz = ;
        timerConfig.speed_mode = ;
        ledc_timer_config(&timerConfig);

        ledc_channel_config_t tChaConfig;
        tChaConfig.gpio_num = ;
        tChaConfig.speed_mode = ;
        tChaConfig.channel = ;
        tChaConfig.intr_type = ;
```

```
	tChaConfig.timer_sel = ;
	tChaConfig.duty = ;
	ledc_channel_config(&tChaConfig);
}


void app_main()
{
	setADC();
	setPWM();
	duty_queue = xQueueCreate(10, sizeof(int));
	xTaskCreate(&ADCtask,"ADCtask",2048,NULL,5,NULL);
	xTaskCreate(&PWMtask,"PWMtask",2048,NULL,5,NULL);

}
```

Listing 1. Program template for Lab 3.