**BAHRIA UNIVERSITY, (Karachi Campus)**



*Department of Software Engineering*

**Assignment 4 - Fall 2024**

COURSE TITLE: **EMBEDDED SYSTEM DESIGN** COURSE CODE: **CEN-439**

# Class: BSE - V (C) Shift: Morning

Course Instructor: **ENGR.NOMANAHMED** Time Allowed:  **2 Week**  Submission Date: **27-12-2024** Max. Marks: **5 Marks**

# Question No. 1 [CLO3: 1 Mark]

Construct a code to generate a square wave of 1KHz frequency in PORTB5 using STM32F446 microcontroller. Assume XTAL = 1MHz.

**Solution:**

#include "stm32f4xx.h"

void delay\_us(uint32\_t us);

int main() {

RCC->AHB1ENR |= RCC\_AHB1ENR\_GPIOBEN;

GPIOB->MODER |= GPIO\_MODER\_MODER5\_0;

while (1) {

GPIOB->ODR ^= GPIO\_ODR\_OD5;

delay\_us(500);

}

}

void delay\_us(uint32\_t us) {

SysTick->LOAD = (1 \* us) - 1;

SysTick->VAL = 0;

SysTick->CTRL = SysTick\_CTRL\_CLKSOURCE\_Msk | SysTick\_CTRL\_ENABLE\_Msk;

while ((SysTick->CTRL & SysTick\_CTRL\_COUNTFLAG\_Msk) == 0);

SysTick->CTRL = 0;

**Question No. 2 [CLO3: 1 Mark]**

Explain the different I/O registers of STM32F44b microcontroller.

**Solution:**

**GPIOx\_MODER (Mode Register):**

* Configures the mode of each pin: input, output, alternate function, or analog.

**GPIOx\_OTYPER (Output Type Register):**

* Sets the type of output: push-pull or open-drain.

**GPIOx\_OSPEEDR (Output Speed Register):**

* Defines the speed for output pins: low, medium, high, or very high speed.

**GPIOx\_PUPDR (Pull-Up/Pull-Down Register):**

* Configures pull-up or pull-down resistors for each pin.

**GPIOx\_IDR (Input Data Register):**

* Reads the input values of the pins.

**GPIOx\_ODR (Output Data Register):**

* Writes values to the output pins.

**GPIOx\_BSRR (Bit Set/Reset Register):**

* Atomic operations for setting/resetting individual bits.

**GPIOx\_LCKR (Lock Register):**

* Locks the configuration of GPIO pins.

**GPIOx\_AFRL/AFRH (Alternate Function Registers):**

* Configures alternate functions for pins.

# Question No. 3 [CLO3: 1 Mark]

Develop a code to toggle LD2 for 1 second ON and 1 second OFF by writing 0 or 1 to bit 5 of the Port A Output Data Register. The green LED (LD2) is connected to PA5.

**Solution:**

#include "stm32f4xx.h"

void delay\_ms(uint32\_t ms);

int main() {

RCC->AHB1ENR |= RCC\_AHB1ENR\_GPIOAEN;

GPIOA->MODER |= GPIO\_MODER\_MODER5\_0;

while (1) {

GPIOA->ODR ^= GPIO\_ODR\_OD5;

delay\_ms(1000);

}

}

void delay\_ms(uint32\_t ms) {

for (uint32\_t i = 0; i < ms \* 1000; i++) {

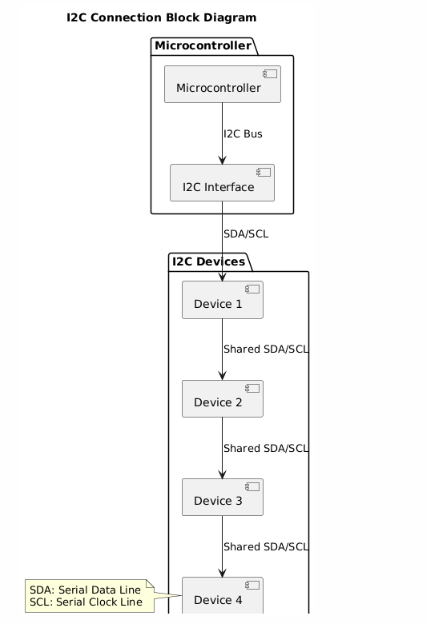
\_\_NOP(); }

}

# Question No. 4 [CLO3: 1 Mark]

Construct a block diagram to connect 4 different devices with microcontroller using I2C protocol.

**Solution:**



# Question No. 5 [CLO3: 1 Mark]

Analog sensor is connected at A0 in Arduino. Develop a code to read the analog data and display it on LCD.

void loop() {

int sensorValue = analogRead(A0);

float voltage = sensorValue \* (5.0 / 1023.0);

lcd.clear();

lcd.print("Sensor Value:");

lcd.setCursor(0, 1);

lcd.print(voltage);

lcd.print(" V");

delay(500);

}

**Solution:**

#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup() {

lcd.begin(16, 2);

analogReference(DEFAULT);