



American International University - Bangladesh (AIUB)

Faculty of Engineering

Course Name: Microprocessor and Embedded System

Course Code: EEE 4103

Semester: Spring 2023-24

Term: Mid

Faculty Name: Md. Shaoran Sayem

Assignment #: 02/02

Course Outcome Mapping with Questions

Item	COs	POIs	K	P	A	Marks	Obtained Marks
Q1-4	CO1	P.a.4.C3	K4	P1, P3, P7		5×4	
Total:						20	

CO/ CLO Number	CO/CLO Statement	K	P	A	Assessed Program Outcome Indicator	BNQF Indicator	Teaching- Learning Strategy	Assessment Strategy
CO1	Apply knowledge of microprocessors and microcontrollers to configure different modules of a microprocessor or microcontroller-based system as per given specifications and perform in-depth analysis to optimize the performance of the designed system utilizing the basic concepts and properties of a microcontroller.	K4	P1, P3, P7		P.a.4.C.3	FS.2	Lecture/ Discussion/ Q&A	Midterm Assignment 2

Student Information:

Due Date:	05/03/2024								Submission Date:	
Student Name:	MD. SHOHANUR RAHMAN SHOHAN									
Student ID #:	2	2	-	4	6	0	1	3	-	1
	p	q	-	a	b	c	d	e	-	r
	Department:				CSE			Section:	J	

Marking Rubrics (to be filled by Faculty):

	Excellent [5]	Proficient [4]	Good [3]	Acceptable [2]	Unacceptable [1]	No Response [0]	
Problem #	Detailed unique response explaining the concept properly and the answer is correct with all works clearly shown.	Response with no apparent errors and the answer is correct, but the explanation is not adequate/unique.	The response shows an understanding of the problem, but the final answer may not be correct	The partial problem is solved; the response indicates part of the problem was not understood clearly or not solved.	Unable to clarify the understanding of the problem and method of the problem solving was not correct	No Response/ copied from others/identical submissions with gross errors/ image file printed	Secured Marks
1							
2							
3							
4							
Comments						Total marks (20)	

1. In a biscuit factory, a control system has been designed based on an Arduino Uno microcontroller. This system transfers the biscuit packets at a regular time interval to the conveyor belt from the biscuit-making machine. It was found that the Timer modules in the Arduino microcontroller are perfect candidates to count the number of biscuit packets being transferred to the conveyor belt correctly and appropriately. Compute the number of biscuit packets being transferred using the Timers within the following time intervals (Timer0 or Timer1 may be used as needed).
 - a) $r0$ ms
 - b) abc ms

Assume that the Arduino system clock has been set at $1q$ MHz and the available pre-scalers of the system are 1, 8, 64, 256, 512, and 1024. Comment on why and how the required counts are possible to implement using Timer0 or Timer1. Write down the program codes for the Arduino board.
2. (a) A TV remote controller is to be designed for the selection of channels. The channel selection buttons are required to be debounced, that is, one press must cause the remote to change to the subsequent channel. It was observed that the switches exhibit bounce times well under b ms and the duty cycle is 50%. Design a circuit using the 74HC14 Schmitt trigger IC along with the resistance and capacitance. The worst-case V_{th} of 74HC14 for a signal going LOW is $2.r$ V and that of when going high is $0.pq$ V. Also, consider that the CMOS device leakage current is pa μ A and the gate's best-case switching point is of the order of $0.r$ V. What is the hysteresis voltage?

(b) Write a software code for the Arduino to read the status of a switch connected to pin number p by avoiding the bouncing problem of the switch and then display that switch status in the LED connected to pin number $1r$. If the switch status is HIGH/LOW the LED will be turned ON/OFF respectively. It was observed that the switches exhibit bounce times well under $q0$ ms and the duty cycle is 50%. Use the Timer0 or Timer1 function of the Arduino to compute this delay. Also, draw the flow chart of this program.
3. A biomedical engineer is to design a new laboratory where biologically hazardous materials are being used for experimental research works. As such, during the experimental works are ongoing, no one is allowed to remain inside the laboratory. To display the message that hazardous materials are being used in the laboratory, 2 lights with green and red colors are installed at the entrance door of the laboratory. When the green light is turned on, it indicates that anybody can enter the laboratory but when the red light is turned on then none can enter the laboratory. It is to be ensured that both lights can't remain turned on at the same time and these 2 lights should be controlled using a single mechanical switch inside the laboratory, and they can only light up alternatively when the switch is pressed or depressed. The switch has already been debounced using a hardware circuit. Draw a flowchart to show the logical flow of the program and then write a program for the Arduino Uno microcontroller to operate the door lights based on the switch conditions as per the given constraints.
4. Write an Arduino microcontroller code for an acd ms interval LED blinking system using Timer x ($x = 0-2$) interrupts and a pre-scalar value of 1024. LED is connected to the pin r of the board. Compute the value to be loaded into the OCR($r-1$)A register.

Solution: 1

(a)

According to my id, $p = 1$, $q = 2$, $a = 4$, $b = 6$, $c = 0$.

So Required Delay = 10ms = 10×10^{-3} s
Frequency = 12 MHz = 12×10^6 Hz
Prescaler = 1024

$$\begin{aligned}\text{Time count} &= \frac{\text{Required Delay} \times \text{frequency}}{\text{Prescaler}} - 1 \\ &= \frac{10 \times 10^{-3} \times 12 \times 10^6}{1024} - 1 \\ &= 116.1875 \\ &= 117\end{aligned}$$

With the 10ms time the possible number of biscuit packets that can be counted is 117. Here Timer0 can be used as the maximum count for Timer0 is 255.

Codes for the Arduino Board:

```
#define PIN_USED 10
int milisec = 10;
int prescaler = 1024;
float clock_freq = 12000000.0 / prescaler;
float clock_period = 1.0 / clock_freq;
int count = (milisec * 0.001 / clock_period) - 1;
void setup() {
  pinMode(PIN_USED, OUTPUT);
  // Configure Timer0
  TCCR0A = 0b00000000;
  TCCR0B = 0b00000101;
  TCNT0 = 0;
}
void loop() {
  if (TCNT0 >= count) {
    TCNT0 = 0; // Reset the counter
    digitalWrite(PIN_USED, !digitalRead(PIN_USED));
  }
}
```

(b)

Required Delay = 460ms = 460×10^{-3} s
Frequency = 12 MHz = 12×10^6 Hz
Prescaler = 1024

$$\begin{aligned}\text{Time count} &= \frac{\text{Required Delay} \times \text{frequency}}{\text{Prescaler}} - 1 \\ &= \frac{460 \times 10^{-3} \times 12 \times 10^6}{1024} - 1 \\ &= 5389.625 \\ &= 5390\end{aligned}$$

With the 460ms time the possible number of biscuit packets that can be counted is 5390. Here Timer0 cannot be used as it can only count up to 255. Whereas Timer1 can count up to 65535. So Timer1 is used here.

Codes for the Arduino Board:

```
#define PIN_USED 10
int milisec = 460;
int prescaler = 1024;
float clock_freq = 12000000.0 / prescaler;
float clock_period = 1.0 / clock_freq;
int count = (milisec * 0.001 / clock_period) - 1;
void setup() {
  pinMode(PIN_USED, OUTPUT);
  // Configure Timer1
  TCCR1A = 0b00000000;
  TCCR1B = 0b00000101;
  TCNT1 = 0;
}
void loop() {
  if (TCNT1 >= count) {
    TCNT1 = 0;
    digitalWrite(PIN_USED, !digitalRead(PIN_USED));
  }
}
```

Solution:2

(a)

Hysteresis voltage (V_h) is the difference between the high-going (V_{th+}) and low-going (V_{th-}) threshold voltages. Given that V_{th+} is 0.22 V and V_{th-} is 2.1 V, we can calculate V_h as follows:

$$V_h = V_{th+} - V_{th-}$$

$$V_h = (0.22 - 2.1) \text{ V}$$

$$V_h = -1.88 \text{ V}$$

However, hysteresis voltage should be a positive value. So, the hysteresis voltage is 1.88 V.

(b)

Codes for the Arduino Board:

```
#include <avr/io.h>
#include <avr/interrupt.h>
const int switchPin = 2;
const int ledPin = 10;
volatile bool switchState = LOW; // Initial state of the switch
void setup() {
  pinMode(switchPin, INPUT);
  pinMode(ledPin, OUTPUT);
  // Configure Timer1
  cli(); // Disable global interrupts
  TCCR1A = 0; // Set Timer1 control register A to default values
  TCCR1B = 0; // Set Timer1 control register B to default values
  TCNT1 = 0; // Initialize Timer1 counter value to 0
  OCR1A = 156; // Set the compare value to generate a 10 ms interrupt (12 MHz clock)
  TCCR1B |= (1 << WGM12); // Enable CTC mode

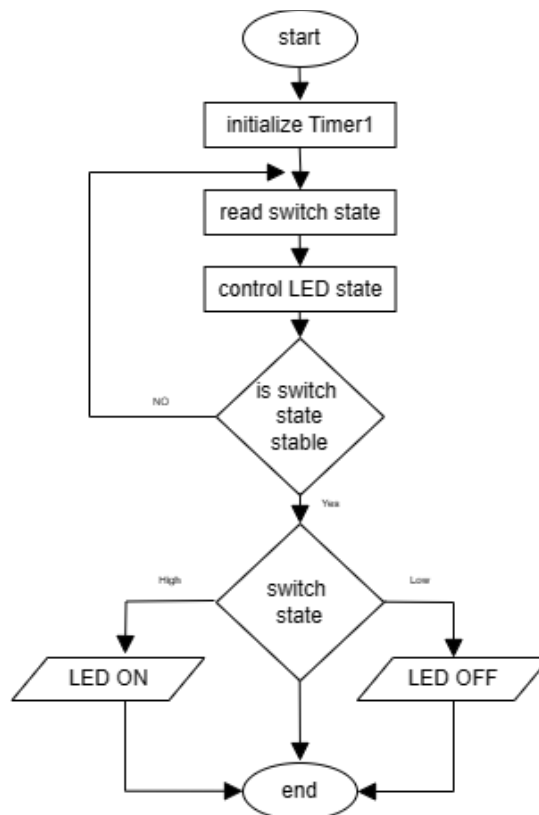
  TCCR1B |= (1 << CS11); // Set prescaler to 8
  TIMSK1 |= (1 << OCIE1A); // Enable Timer1 output compare match A interrupt
  sei(); // Enable global interrupts
```

```

}
void loop() {
// Read the switch state
switchState = digitalRead(switchPin);
// Control the LED based on the debounced switch state
digitalWrite(ledPin, switchState);
}
// Timer1 compare match A interrupt service routine
ISR(TIMER1_COMPA_vect) {
static bool previousSwitchState = LOW;
static int debounceCounter = 0;
// Read the current switch state
bool currentSwitchState = digitalRead(switchPin);
// Check for a stable switch state for 5 consecutive timer interrupts (50 ms)
if (currentSwitchState == previousSwitchState) {
if (debounceCounter < 5) {
debounceCounter++;
} else {
switchState = currentSwitchState;
debounceCounter = 0;
}
} else {
debounceCounter = 0;
}
previousSwitchState = currentSwitchState;
}

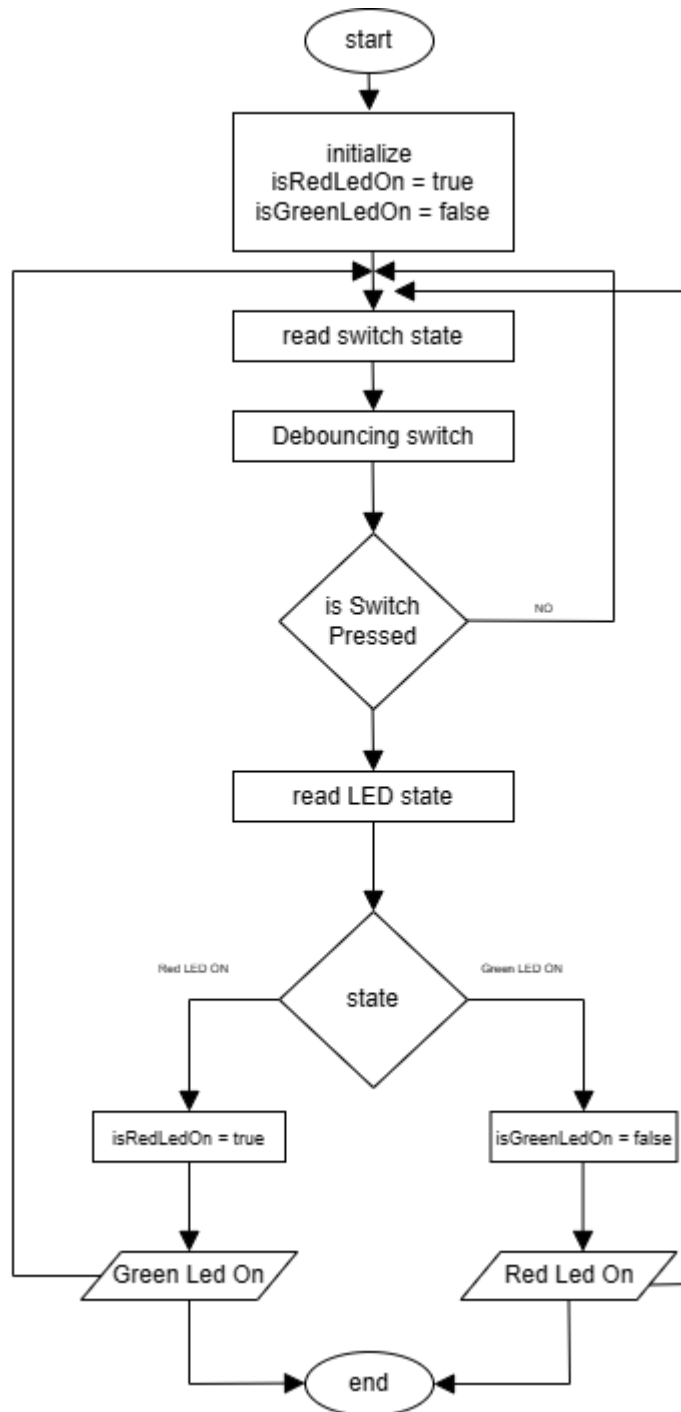
```

Flowchart:



Solution: 3

Flowchart:



Codes for Arduino Board:

```
const int greenLightPin = 2;
const int redLightPin = 3;
const int switchPin = 4;
bool isGreenLightOn = false;
bool isRedLightOn = true;
void setup() {
  pinMode(greenLightPin, OUTPUT);
  pinMode(redLightPin, OUTPUT);
}
```

```

pinMode(switchPin, INPUT_PULLUP);
// Initialize with the green light on
digitalWrite(greenLightPin, HIGH);
digitalWrite(redLightPin, LOW);
}
void loop() {
int switchState = digitalRead(switchPin);
if (switchState == LOW) {
// Switch is pressed
if (isGreenLightOn) {
// Green light is on, turn it off and red light on
digitalWrite(greenLightPin, LOW);
digitalWrite(redLightPin, HIGH);
isGreenLightOn = false;
} else {
// Red light is on, turn it off and green light on
digitalWrite(greenLightPin, HIGH);
digitalWrite(redLightPin, LOW);
isGreenLightOn = true;
}
}
delay(200);
}
}

```

Solution:4

According to my id, p = 1, q = 2, a = 4, b = 6, c = 0, d = 1, r = 1.

Interval = 461 ms = 0.461s

Pre-scalar value = 1024

Frequency = 12 MHz

Timer Frequency = $\frac{\text{frequency}}{\text{pre-scalar}} = \frac{12}{1024} = 11.718\text{KHz}$

OCR1A = (Timer Frequency × interval) - 1 = (11.718 × 0.461) - 1 = 5401

Here Timer1 is to be used as the maximum count for Timer0 and Timer2 is 255 whereas maximum count for Timer1 is 65535. And the value to be loaded in OCR1A register is 5401.

Codes for Arduino Board:

```

#include <avr/io.h>
#include <avr/interrupt.h>
const int ledPin = 1;
void setup() {
pinMode(ledPin, OUTPUT);
noInterrupts();
TCCR1A = 0;
TCCR1B = 0;
TCCR1B |= (1 << CS12) | (1 << CS10);
OCR0A = 5401;
interrupts();
}
void loop() {
}
ISR(TIM1_COMPA_vect) {
digitalWrite(ledPin, !digitalRead(ledPin));
}

```