**Department of Electrical Engineering and   
Computer Science**

**Faculty Member:** Dr. Arbab Latif  **Dated:** 23/05/2022

**Semester:** 4th **Section:** BEE 12C

**EE-222: Microprocessor Systems**

Lab 13: 8051 Assembly

Group Members

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Reg. No** | **Report**  **Marks / 10** | **Viva**  **Marks / 5** | **Total**  **Marks / 15** |
| Danial Ahmad | 331388 |  |  |  |
| Muhammad Umer | 345834 |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Experiment

## Objectives

1. Write an assembly language code for x86 architecture
2. Create .hex file using Keil µVision IDE
3. Burn your code onto 8051 microcontroller to run your application successfully.

## Equipment

Software

* *Keil µVision*

## Introduction

Atmel Studio IDE is a free development environment for programming Atmel MCUs, sourced by Microchip Technology Inc. It provides us with the means to simulate assembly language codes on specific Microcontrollers and provides an easy and intuitive way of producing .HEX files, which are what makes burning the code on the hardware possible.

In this specific lab, we familiarize ourselves with the extension of application of timers present within the ATmega16A. We use specific values of Output Compare Register (OCR) to generate a pulse with desired duty cycle. To implement a practical application of PWM, we use a potentiometer (control voltage) to directly change the brightness of an LED.

## Deliverable

You are required to submit:

* Appropriately Commented Code
* Explicit Calculations for Timer Values
* Issues in Developing the Solution and your Response

## Hardware Resources

This is a Proteus based simulation lab and does not require any hardware

# Lab Tasks

## Task A (LED Interfacing with 8051)

**ORG 00H ; PROGRAM STORE LOCATION**

**MOV P0,#00H ; REFRESHING PORT-0**

**MOV P1,#00H ; REFRESHING PORT-1**

**MOV P2,#00H ; REFRESHING PORT-2**

**MOV P3,#00H ; REFRESHING PORT-3**

**AGAIN:**

**CLR P0.7 ; PORT-0 PIN-7 LOW (LED1-ON)**

**SETB P2.0 ; PORT-2 PIN-0 HIGH (LED2-ON)**

**LCALL DELAY ; CALL DELAY SUBROUTINE**

**SETB P0.7 ; PORT-0 PIN-7 HIGH (LED1-OFF)**

**CLR P2.0 ; PORT-2 PIN-0 LOW (LED2-OFF)**

**LCALL DELAY ; CALL DELAY SUBROUTINE**

**LJMP AGAIN ; LOOP FOREVER TO LABEL AGAIN**

**DELAY: ; DELAY SUBROUTINE**

**MOV R7,#50**

**L3 : MOV R6,#100**

**L2 : MOV R5,#100**

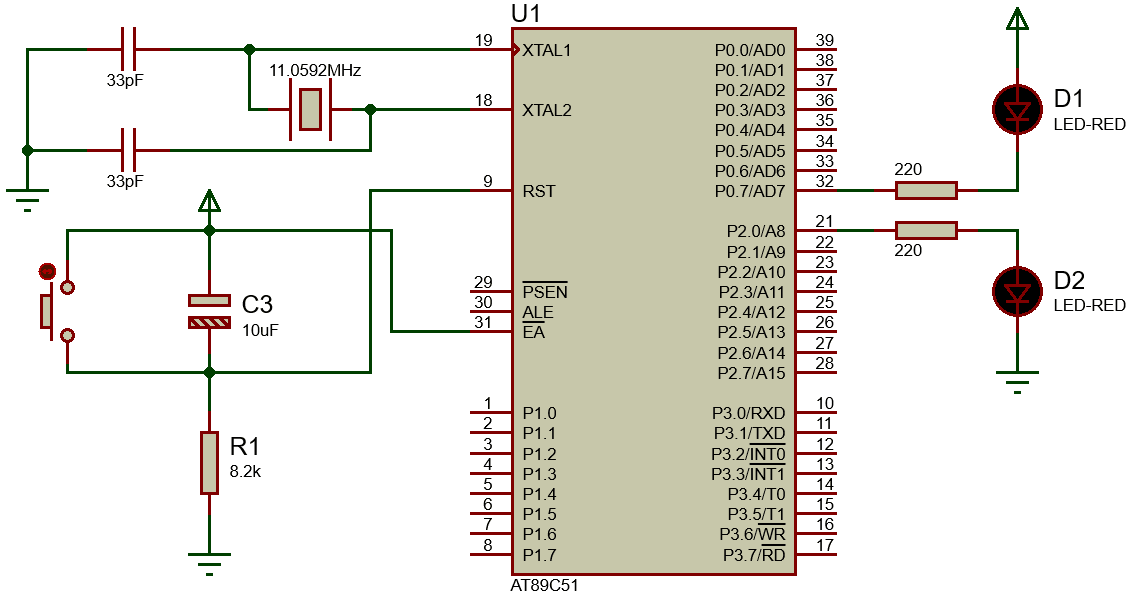
**L1 : DJNZ R5,L1**

**DJNZ R6,L2**

**DJNZ R7,L3**

**RET ; END DELAY SUBROUTINE**

**END**



Simulation

## Task B (ADC Interfacing with 8051)

**ORG 000H ;initial starting address**

**MOV DPTR,#CC\_PATTERNS ; loads the adress Lookup Table**

**MOV P1, #0H**

**MOV P3, #0H**

**START:**

**MOV A,#00H ; initial value of accumulator**

**MOV B,A**

**MOV R0,#0AH ;Register R0 initialized as counter which counts from 10 to 0**

**LOOP:**

**MOV A,B**

**INC A**

**MOV B,A**

**ACALL DISPLAY ;**

**MOV P2,A**

**ACALL DELAY1 ; calls the delay of the timer**

**DEC R0 ;Counter R0 decremented by 1**

**MOV A,R0**

**JZ START**

**SJMP LOOP**

**DELAY1:**

**MOV R5, #250D**

**LABEL:**

**ACALL DELAY**

**ACALL DELAY**

**ACALL DELAY**

**ACALL DELAY**

**DJNZ R5,LABEL**

**RET**

**DELAY:**

**MOV R6,#250D**

**MOV R7,#250D**

**LABEL1:**

**DJNZ R6,LABEL1**

**LABEL2:**

**DJNZ R7,LABEL2**

**RET**

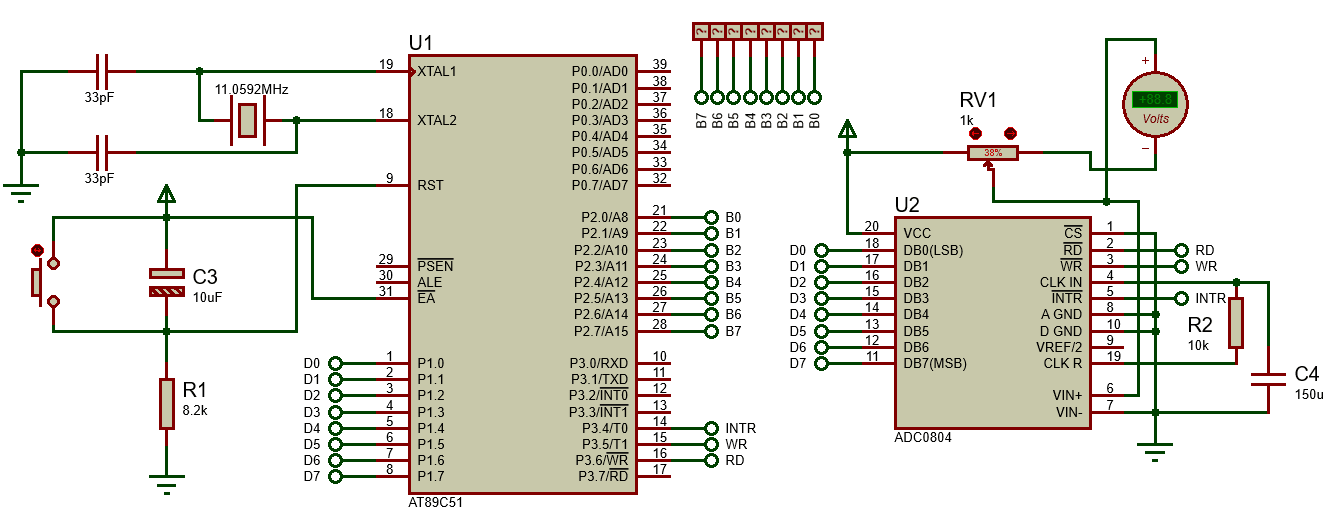
**DISPLAY:**

**MOVC A,@A+PC**

**CC\_PATTERNS:**

**DB 3FH, 06H, 5BH, 4FH, 66H, 6DH, 7DH, 07H, 7FH, 6FH, 0**

**END**



Simulation

## Conclusion

After the conduction of this lab, we have learnt and familiarized ourselves with the assembly language propramming of the 8051 microcontroller. Along with its programming, we also learnt how to get around a newly introduced IDE, Keil Vision, and generate the HEX file of a target build file to use it in a Proteus simulation.