PRESETTABLE ALARM

Micro Processor & Micro Controller (2EC404)

Special Assignment Report

Submitted by:

21BEC111 Dhairya Senghani

21BEC123 Tanishk Jain

Submitted to Dr. Jayesh Patel

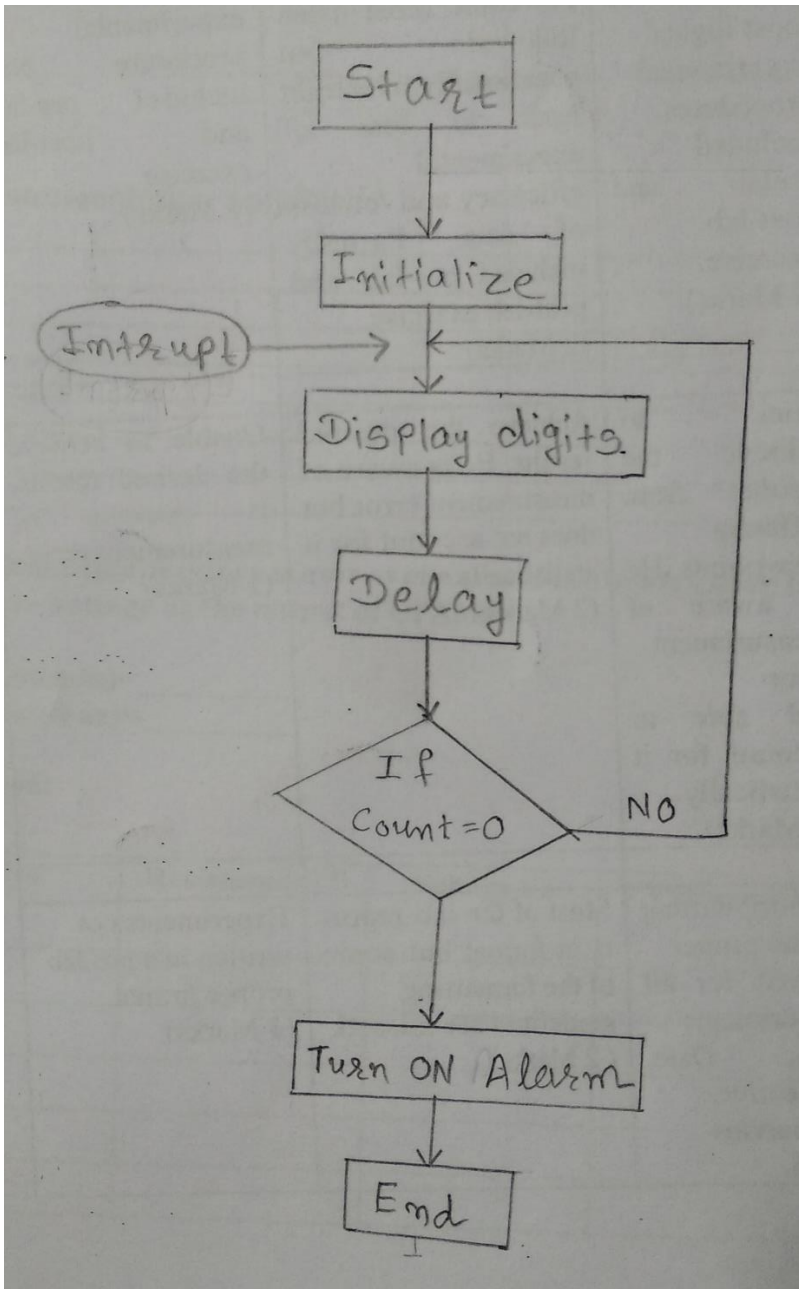
INTRODUCTION

A presettable alarm system is a popular application of any microcontroller. The 8051 microcontroller is a widely used microcontroller in embedded systems and has various features that make it suitable for designing such systems. The presettable alarm system is designed to trigger an alarm at a specific time and date, which can be set by the user.

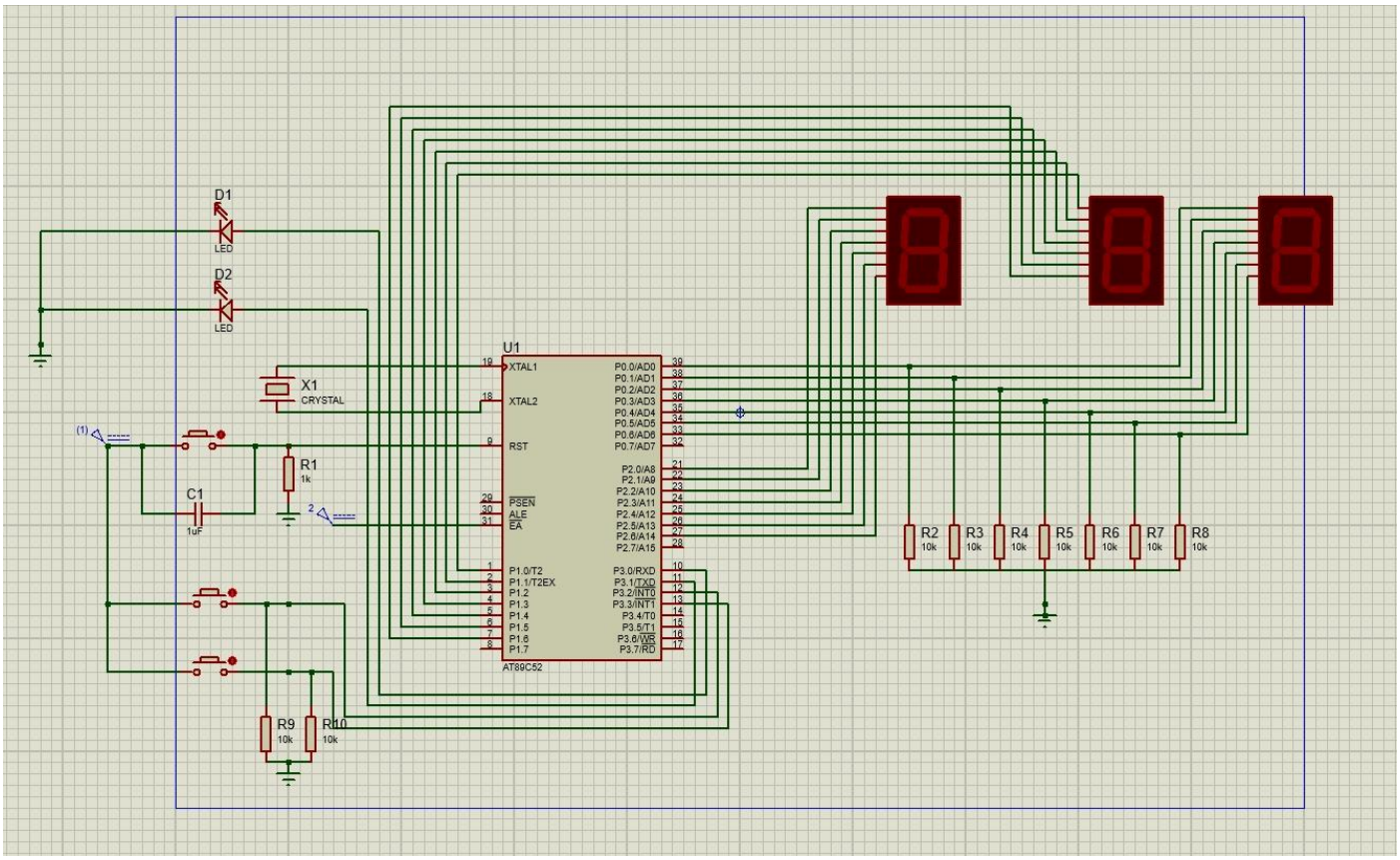
Here we have used **AT89C52** microcontroller IC to implement an alarm. Output is displayed on 3 seven segment displays. Alarm can be set by using push buttons. LED and BUZZER gets turn ON when alarm triggers.

This alarm is one kind of stop watch which count down the numbers(minutes and hours). When count reaches to zero it triggers the alarm. Timer 0 is used in mode 1 to generate the delay of 1 second. Also some registors are involved to generate such large delay.INT0 is uses to decrement minute count and INT1 is used to decrement hour count. Hour count shows hexadecimal value starts from F(15) and goes down to 0. Crystal frequency is 11.0592MHz.

FLOW CHART



CIRCUIT DIAGRAM



WORKING OF ALARM

1. **Initialization**: When the system is powered on, the microcontroller initializes the minute to **59** and hour to **F (15).**

2. **Setting the alarm:** The user can set the alarm by pressing the buttons. The microcontroller reads the input and updates the count.

3. **Comparison:** The microcontroller continuously compares the current with **0 hour** and **00 minute**. Once the present count reaches to **0 hour and 00 minute**, the microcontroller triggers an alarm.

4. **Alarm triggering:** When the alarm triggers, alarm output pin(P3.1) blink an LED (OR buzzer). The type of alarm triggering can be customized, such as a buzzer, flashing light, or any other suitable device that can alert the user.

CODE (Assembly Language)

```
ORG 0000H
        LJMP MAIN
    ORG 0003H       ;ISR for INT0
   DEC R3  ;decrement the count for minute (tenth_digit)
    MOV A,R3        ;display minute(tenth_digit)
    ADD A,R0
    MOV R1,A
    MOV P1,@R1
    RETI

    ORG 0013H       ;ISR for INT1
    DEC R4          ;decrement the count for hour
    MOV A,R4        ;display hour
    ADD A,R0
    MOV R1,A
    MOV P2,@R1
RETI

    ORG 0030H
    MAIN:       ;main program
    MOV R0,#30H     ;starting address of lookup table
            ;look up table for 7 segment display
    MOV @R0,#00H    ;turn OFF display
    INC R0
    MOV @R0,#3FH    ;digit_0
    INC R0
    MOV @R0,#06H    ;digit_1
    INC R0
    MOV @R0,#5BH    ;digit_2
    INC R0
    MOV @R0,#4FH        ;digit_3
    INC R0
    MOV @R0,#66H        ;digit_4
    INC R0
    MOV @R0,#6DH        ;digit_5
    INC R0
    MOV @R0,#7DH        ;digit_6
    INC R0
    MOV @R0,#07H        ;digit_7
    INC R0
    MOV @R0,#7FH        ;digit_8
    INC R0
    MOV @R0,#6FH        ;digit_9
    INC R0
    MOV @R0,#77H        ;digit_A
    INC R0
    MOV @R0,#7CH        ;digit_B
    INC R0
    MOV @R0,#39H        ;digit_C
    INC R0
    MOV @R0,#5EH        ;digit_D
    INC R0
    MOV @R0,#79H        ;digit_E
    INC R0
    MOV @R0,#71H        ;digit_F

    MOV R0,# 30H    ;starting adrees of look-up table
```

```
MOV TMOD,#01H          ;timer-0 in mode-1
CLR P3.1           ;Alarm pin1(output)
CLR P3.4
MOV IE,#10000101B      ;enable INT0 & INT1
SETB TCON.0            ;INT0 in edge triggered
SETB TCON.2       ;INT1 in edge triggered
CLR C
;initialization of ports and registers
MOV P0,#6FH
MOV P1,#6FH
MOV P2,#6FH
MOV R2,#0AH
MOV R3,#06H
MOV R4,#10H

HOUR:MOV A,R4
ADD A,R0
MOV R1,A
MOV P2,@R1             ;display hour

MOV R3,#06H
MINUTE_1:MOV A,R3
ADD A,R0
MOV R1,A
MOV P1,@R1     ;display minute(tenth digit)

MOV R2,#0AH
MINUTE_0:MOV A,R2
ADD A,R0
MOV R1,A
MOV P0,@R1             ;display minute(unit)

MOV R6,#78H      ;delay generation of 1 minute
TMR2:MOV R7,#07H
TMR1:
MOV TL0,#00H
MOV TH0,#00H
SETB TR0
BACK:JNB TF0,BACK
CLR TR0
CLR TF0
DJNZ R7,TMR1
CPL P3.0
DJNZ R6,TMR2

DJNZ R2,MINUTE_0   ;jump to display new digits
DJNZ R3,MINUTE_1
DJNZ R4,HOUR

SETB P3.4          ;turn on alarm
ALARM:
CPL P3.1      ;blink alarm LED with 500ms delay
MOV R5,#03H
REP3:MOV R6,#0FFH
REP2:MOV R7,#0FFH
REP1:DJNZ R7,REP1
DJNZ R6,REP2
DJNZ R5,REP3
SJMP ALARM
        END
```

BILL OF MATERIAL

• AT89C52 Microcontroller	1 x 72 Rs.	= 72 Rs.
• Crystal Oscillator (11.0592MHz)	1 x 8 Rs.	= 8 Rs.
• 7 Segment displays (common cathod)	3 x 7 Rs.	= 21 Rs.
• Resistors (10k, 1k,470 ohm)	13 x 0.5 Rs.	= 6.5 Rs.
• Capacitor (1uF)	1 x 3 Rs.	= 3 Rs.
• LED (White)	2 x 1 Rs.	= 2 Rs.
• Push Button Switch 4 Pin	3 x 5 Rs.	= 15 Rs.
• Buzzer (5V)	1 x 10 Rs.	= 10 Rs.
• Jumper wires	35 x 0.5 Rs.	= 17.5 Rs.
• Breadboards	1 x 40 Rs.	= 40 Rs.
• power supply (5V)		
• **TOTAL**		**195 Rs.**

APPLICATIONS

1. **Personal alarms:** Presettable alarms are often found to be used to wake up. It is also used in daily life to get reminder to do any task.
2. **Industrial alarms:** Presettable alarms are used in industrial settings to alert workers to specific events or milestones. For example, an alarm may be set to sound when a production run is complete.
3. **Medical devices:** Presettable alarms are often built into medical devices such as infusion pumps or glucose monitors to remind patients or healthcare professionals for specific events, such as medication needs to be administered.
4. **Environmental monitoring:** Presettable alarms can be used in environmental monitoring systems to remind researchers or technicians for specific events, such as monitoring of level of pollution.

SUMMARY

The presettable alarm system can be easily implemented using the 8051 microcontroller. We can program the microcontroller as per the requirement of user. Since very less hardware is required, implementation is easy and cheap. Multiplexing concept can also be used to reduce the connections. We learnt a lot about software and hardware interfacing of 8051.