8086

Arduino

Resistor

Inductor

Capacitor

INdiode

Ground

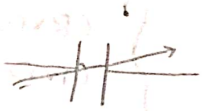
Power Supply

Acpowa Supply

Variable Resistor.

Variable Inductor

Variable capacitor

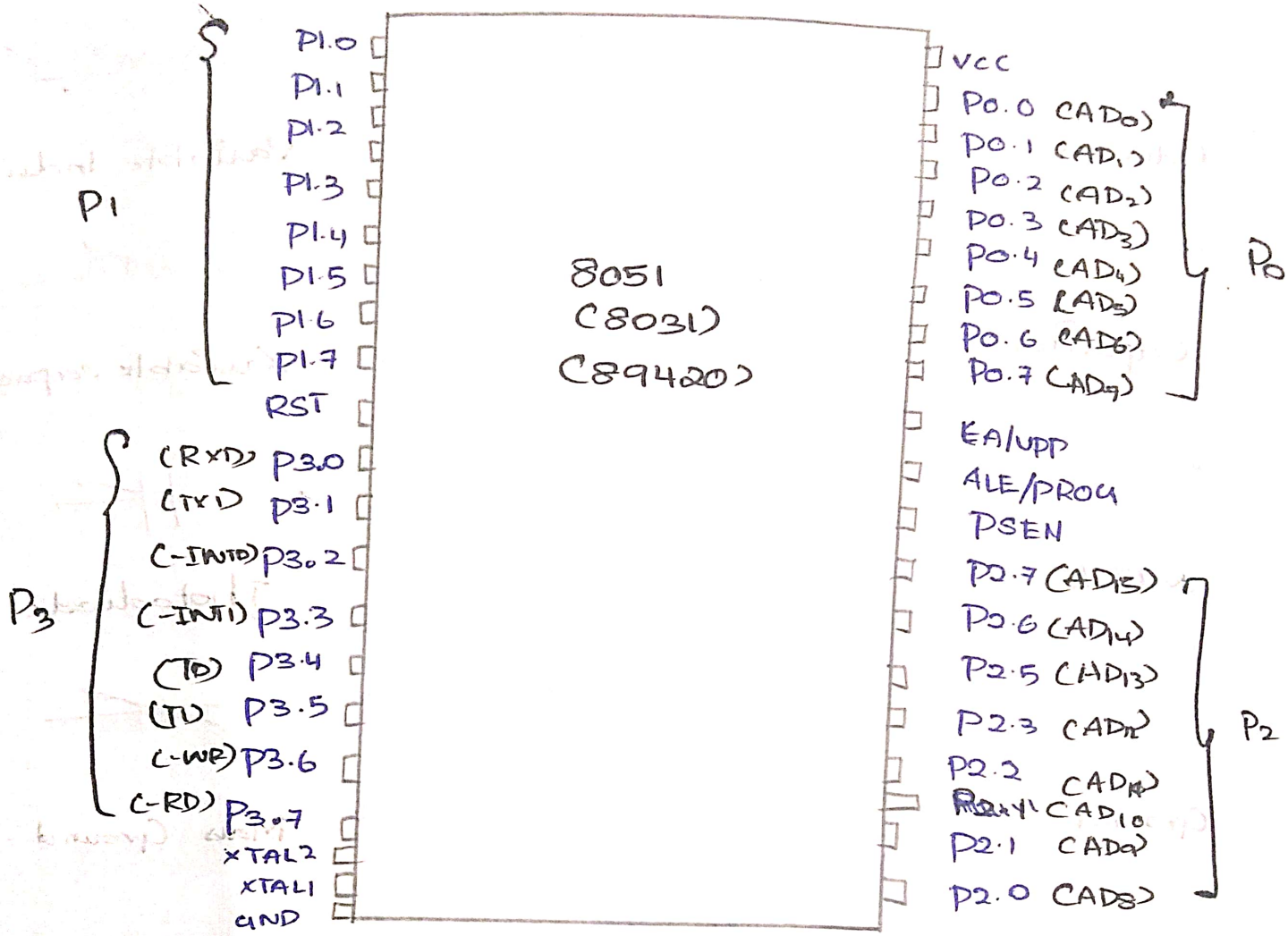


Photodiode

Mains Ground.

Battery (or) DC Power Supply

light



Pin diagram of

8051

Microcontroller

Expt:

21/2/2024

BLINKING OF LED USING 8051 MICROCONTROLLER USING PROTEUS

AIM: To write an assembly language Program to LED blinking using 8051

Software Required:

Proteus Software

Program:

```
        ORG 0000H
UP: SETB P2.0
    ACALL DELAY
    CLR P2.0
    ACALL DELAY
    SJMP UP
DELAY: MOV R4, #35
H1: MOV R3, #255
H2: DJNZ R3, H2
    DJNZ R4, H1
    RET
    END
```

Result:

Thus the program has been executed Successfully

Exp2: LED CHASER USING 8051 USING PROTEUS

2/12/2024

Aim:

Write an assembly language program for LED Chaser using 8051 using Keil and Proteus.

Software Required:

Proteus 8 Software

Program:

```
ORG 0000H
UP: MOV P2,#01H
ACALL DELAY
MOV P2,#02H
ACALL DELAY
MOV P2,#04H
ACALL DELAY
MOV P2,#08H
ACALL DELAY
MOV P2,#10H
ACALL DELAY
MOV P2,#20H
ACALL DELAY
MOV P2,#40H
ACALL DELAY
MOV P2,#80H
ACALL DELAY
SJMP UP
DELAY: MOV R4,#255
H1: DJNZ R4,H1
RET
END
```

Result: Thus the program has been executed successfully.

EXP3 2/12/2024 FADEIN FADEOUT OF LED USING 8051 USING PROTEUS

Aim:

To write an assembly language program for Fade out of LED using 8051 using Keil and Proteus.

Software Required:

Proteus 8 Software

Program:

```
ORG 00H;
MAIN : MOV P2, #00H;
       ACALL FADE_IN;
       ACALL FADE_OUT;
       SJMP MAIN;
;Subroutine to Fade in the LED
FADE_IN:
       MOV R0,#00H;
FADE_IN_LOOP:
       ACALL PWM;
       INC R0;
       CJNE R0,#FFH,FADE_IN_LOOP;
       RET
;Subroutine to fade out the LED
FADE_OUT;
       MOV R0,#FFH;
FADE_OUT_LOOP:
       ACALL PWM;
       DEC R0
       CJNE R0, #00H, FADE_OUT_LOOP;
       RET
;PWM Subroutine.
PWM:
       MOV.A, R0;
       MOV B,#FFH;
       MOV P1,#00H;
PWM_ON_LOOP:
```

```
        DJNZ A,PWM_ON_LOOP;
        MOV P1,#01H ; LEDOFF
PWM_OFF_LOOP:
        DJNZ B,PWM_OFF_LOOP;
        RET
        END
```

OUTPUT:

The brightness of the LED is gradually increasing and decreasing with 1000ms delay.

Result:

Thus the program has been successfully verified and executed.

Exp4
2/12/2024

Generation of square wave using Proteus

Aim: To write an assembly language program to generate square wave using 8051.

Software Required:

Proteus 8 Software.

Program:

```
        ORG 0000H
UP :    SETB P2.0
        ACALL DELAY
        CLR P2.0
        ACALL DELAY
        SJMP UP
DELAY : MOV R4,#35
H1 :    MOV R3,#255
H2 :    DJNZ R3,H2
        DJNZ R4, H1
        RET
        END
```

Result:

Thus the program has been executed successfully.

EXP5
3/12/2021

GENERATION OF TRIANGULAR WAVE USING PROTEUS

AIM:

To write an assembly language program to generate traingular wave using 8051.

Software Required:

Proteus 8 Software

Program:

```
ORG  00H ;
MOV  P2.0,#00H
MOV A,#00H
MOV R0,#00H
UPWARD:
      INCA ;
      MOV P1,A ;
      ACALL DELAY ;
      CJNE A,#0FFH,UPWARD ;
DOWNWARD:
       DECA ;
       MOV P1,A ;
       ACALL DELAY ;
       CJNE A,#00H ,DOWNWARD ;
 SJMP UPWARD ;
:Delay Subroutine
DELAY:
    MOV R1,#255;
DELAY_LOOP1:
    MOV R2,#255;
DELAY_LOOP2:
    DJNZ R2,DELAY_LOOP2 ;
    DJNZ R2,DELAY_LOOP1;
    RET ;
END
```

Result:

Thus the Program has been executed successfully

Exp:6
2/12/2024

ANTICLOCKWISE ROTATION OF STEPPER MOTOR USING 8051 USING PROTEUS

AIM: To write an assembly program to rotate the Stepper motor in anti-clockwise direction in 8051 using proteus.

Software Required:

Proteus & Software

Program:

```
        ORG 00H;
MAIN:   MOV P2, #0F0H;
        ACALL COUNTERCLOCKWISE;
        ACALL DELAY;
        SMP MAIN;
:Subroutine to rotate stepper motor counterclockwise
COUNTERCLOCKWISE:
        MOV A,#08H;
        MOV P2, A
        ACALL DELAY
        MOV A,#04H;
        MOV P2,A
        ACALL DELAY
        MOV A, #02H;
        MOV P2, A
        ACALL DELAY
        RET ;
:Subroutine to generate delay
DELAY:
    MOV R1,#0FFH;
DELAY_LOOP1;
    MOV.R2,#0FFH;
DELAY_LOOP2;
```

```
    DJNZ R2, DELAY_LOOP2;
    DJN2 R1, DELAY_ LOOP1;
    RET;
END
```

OUTPUT:

The stepper motor is rotating in anti clockwise direction in steps.

RESULT:

Thus the program has been executed and verified Successfully

Exp: 7

2/12/2024

CLOCKWISE ROTATION OF STEPPER MOTOR USING 8051 USING PROTEUS

AIM: To write an assembly language Program to rotate the stepper motor in clockwise direction in 8051 using Proteus.

Software Required:

Proteus & Software.

Program:

```
        ORG 0000H.
UP:     MOV P2,#09H.
        ACALL DELAY
        MOV P2,#0CH
        ACALL DELAY
        MOV P2,#06H
        ACALL DELAY
        MOV.P2,#03H
        ACALL DELAY
        SJMP UP
DELAY:  MOV R4,#18
H1:     MOV R3,#255
H2:     DJNZ R3,H2
        DJNZ R4,H1
        RET
        END
```

OUTPUT:

The stepper motor is rotating in clockwise direction in steps

RESULT:

Thus, the Program has been successfully verified and executed.

Exp: 8
2/12/2024

DIGITAL CLOCK ON LCD

AIM: To write an assembly language program to display digital clock on LCD with using proteus

Software Required:

Proteus Software

Program:

```
ORG   8000H;
Mov R7,#00H;
Mov R6,#00H;
Mov R5,#00H;
ACALL INIT_LCD;
MAIN_LOOP:
      ACALL. UPDATE_LCD;
      ACALL DELAY_1_SEC;
      ACALL INCREMENT_TIME;
      SJMP MAIN_LOOP;
:Subroutine to initialize the LCD
INIT_LCD:
      ACALL CMD_WRITE
      DB 38H;
      ACALL CMD_WRITE
      DB 0CH;
      ACALL CMD_WRITE
      DB 06H
      ACALL CMD_WRITE
      DB 01H
      RET
:Subroutine to increment time
INCREMENT_TIME:
      INC R5;
      CJNE R5,#60,DONE_SEC;
      MOV R5,#00H;
      INC R6
```

```
        CJNE R6, #60, DONE_SEC,
        MOV R6, #00H;
        INC R7
        CJNE R7, #24, DONE_SEC;
        MOV R7, #00H;
    DONE_SEC:
        RET

;Subroutine to update the LCD with the current time
UPDATE_LCD:
        ACALL CMD_WRITE
        DB 80H;
        MOV A,R7;
        ACALL DISPLAY_TWO_DIGIT;
        ACALL DISPLAY_COLOMN;
        MOV A,R6;
        ACALL DISPLAY_TWO_DIGIT;
        ACALL DISPLAY_COLON;
        MOV A,R5
        ACALL DISPLAY_TWO-DIGIT;
        RET

;Subroutine to display two-digit number on LCD
DISPLAY_TWO_DIGIT:
        MOV B,#10;
        DIV AB;
        ADD A,#30H
        ACALL DISPLAY_CHAR;
        MOV A,B;
        ADD A,#30H;
        ACALL DISPLAY_CHAR;
        RET
;Subroutine to display colon ':' on the LCD
DISPLAY COLON:
        MOV A,#3AH;
```

```
        ACALL DISPLAY_CHAR;
        RET
;Subroutine to display a character on the LCD
DISPLAY_CHAR:
        MOV P2,A ;
        SETB P3.2 ;
        CLR P3.2;
        SETB P3.4;
        NOP;
        CLR P3.4;
        RET
;Subroutine to write command to the LCD
CMD_WRITE:
        MOV A,R1;
        MOV P2,A
        CLR P3.2
        CLR P3.3
        SETB P3.4
        NOP
        CLR P3.4
        RET

;Subroutine for 1-second delay.
DELAY_1_SEC;
        MOV. R3,#50;
DELAY_LOOP:
        DJNZ R3,DELAY_LOOP;
        RET
END
```

OUTPUT:

* When this program is run, the LCD will display the current time in the format HH:MM.

* Every second, the display will update to increment the seconds value.

* After reaching 59 seconds, the seconds will reset to 00, and the minutes will increment.

* Similarly, when the minutes reach 59 and increment again, they will reset to 00, and hours will increment

* The hours will increment from 00 to 23 in 24-hours format. When the hours reach 23 and the next second occurs, the hours, the minutes and the seconds will all reset to 00:00:00

RESULT:

Thus, the assembly language program to displays digital clock on LCD with the Proteus has been executed.

Exp: 9

2/12/2024

INTERFACING OF RELAY AND LED WITH 8051 USING PROTEUS

AIM: To write an assembly language program to Interface of relay and LED with 8051 using Proteus.

Software Required:

Proteus 8 Software.

Program:

```
        ORG 0000H;
MOV P1,#00H;
MAIN LOOP:
        SETB P1.0;
        ACALL DELAY;
        CLR P1.0;
        ACALL DELAY;
        SJMP MAIN_LOOP;
: Delay Subroutine for blinking Speed
DELAY
    MOV R1,#255;
DELAY 1:
    MOV R2, #255;
DELAY 2:
    DJNZ R2,DELAY2;
    DJNZ R1,DELAY1;
    RET
END
```

Output:

* The LED connected through the relay will blink with a controlled ON and OFF duration

* The relay acts as a switch controlled by the 8051 microcontroller, turning the LED ON when P1.0 is HIGH and OFF when P1.0 is LOW

* The blinking rate of the LED can be adjusted by changing the delay subroutine

Result:

Thus, the program has been successfully verified and executed successfully.

Exp 10

2/12/2024

INTERFACING OF RELAY AND BULB WITH 8051 USING PROTEUS

AIM: To write an assembly language Program to Interface Relay and bulb using 8051 Proteus

Software Required:

Proteus 8 software

Program:

```
        ORG 0000H
UP :    SETB P2.0
        ACALL DELAY
        CLR P2.0
        ACALL DELAY
        SJMP UP
DELAY : MOV R4, #18
H1 :    MOV R3, #255
H2 :    DJNZ R3, H2
        DJNZ R4, H1
        RET
        END
```

OUTPUT: *The bulb connected through the relay will glow at regular intervals with a controlled ON and OFF duration

*The relay acts as a switch controlled by the 8051 microcontroller turning the bulb ON when P1.0 is High and OFF when P1.0 is LOW

*The ON/OFF rate of the buzzer can be adjusted by modifying the delay times in the subroutine

RESULT: Thus, the Program has been successfully Verified and executed.

Expt11:
9/12/2021

7-SEGMENT DISPLAY USING 8051 USING PROTEUS

AIM:

Write an assembly language Program for 7-segment Display using 8051 using Keil and Proteus.

Software Required:

Proteus 8 Software

Program:

```
ORG 000H
UP: MOV P2,#0C0H
ACALL DELAY
MOV P2,#0F9H
ACALL DELAY
MOV P2,#0A4H
ACALL DELAY
MOV P2,#0B0H
ACALL DELAY
MOV P2,#99H
ACALL DELAY
MOV P2,#92H
ACALL DELAY
MOV P2,#82H
ACALL DELAY
MOV P2,#0F8H
ACALL DELAY
MOV P2,#80H
ACALL DELAY
MOV P2,#90H
ACALL DELAY
```

```
DELAY : MOV R5, #10
H1 : MOV R4, #180
H2 : MOV R3, #255
H3 : DSN Z R3, H3
DJNZ R4, H2
DSN Z R5, H1
RET
END
```

RESULT:

Thus the program has been successfully verified and executed.