**T. Y. B. Tech (Electrical and Computer Engineering)**

**Trimester: V Subject: Microcontroller and Applications**

**Name: Shreerang Mhatre Class: TY**

**Roll No: 52 Batch: A3**

**Experiment No: 05**

**Name of the Experiment:** Interfacing of 8-bit DAC with C8051F340



**Performed on: 28/11/2023**

**Submitted on: 07/11/2023**



**Aim:** Write C program for interfacing of 8 bit DAC with C8051F340 to generate

1. Square wave
2. Triangular wave

**Apparatus:** EPBF340 board, DSO, DSO probes, DAC Board

**Theory:**

The digital to analog converter involves translating digital information to equivalent analog information. DAC 0808 is R-2R ladder DAC giving output analog current so need to convert in voltage. I to V converter is used using LF351.DAC and LF351 require dual power supply of +15V &-15V.

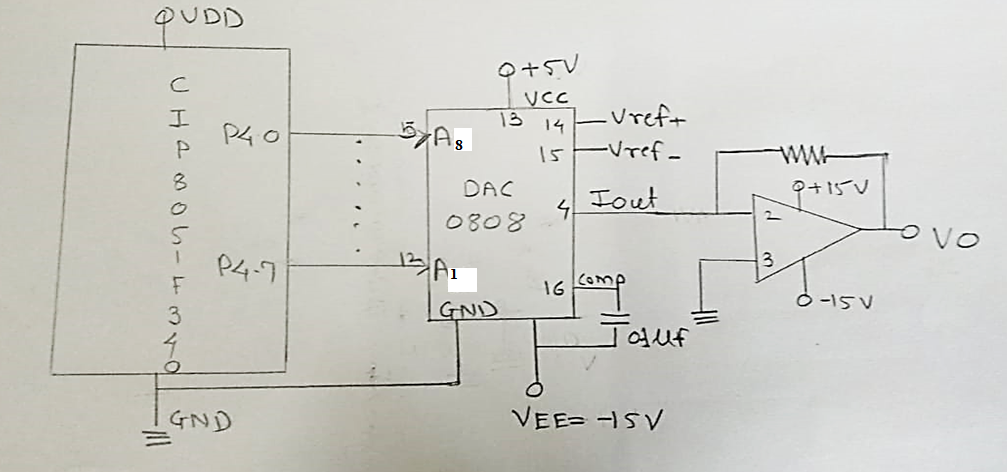
**DAC 0808 features:**

1. 8 bit digital to analog converter
2. Low power consumption 33mW with ± 5V.
3. Power supply voltage range ± 4.5V to ± 18V.
4. Non-inverting digital inputs are TTL and CMOS compatible.
5. 16 pin DIP.
6. High speed multiplying input slew rate: 8mA/μs.
7. Relative accuracy - ± 0.19 % error maximum.
8. Fast settling time: 150 ns typical.
9. Full-scale current match: ± 1 LSB typical.

**Applications:**

1. Programmable power supply.
2. DC motor speed control.
3. Speed synthesis

**Interfacing Diagram:**

****

*Figure 6.1 Interfacing Diagram of 8-bit DAC with C8051F340*

**Hardware Connections:**

Connect dual power supply of 15V to DAC board. Digital data is available on P4 so Connect flat cable between PL6 connector of EPBF340 board to DAC data lines on DAC board.

|  |  |  |  |
| --- | --- | --- | --- |
| **Pin Connection** | **PL6 Connector of EPBF340** | **Pin Connection** | **DAC board data lines socket** |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 | P3.3 |  |  |
| 5 |  |  |  |
| 10 | P4.0 | 3 | D0 |
| 11 | P4.1 | 4 | D1 |
| 12 | P4.2 | 5 | D2 |
| 13 | P4.3 | 6 | D3 |
| 14 | P4.4 | 7 | D4 |
| 15 | P4.5 | 8 | D5 |
| 16 | P4.6 | 9 | D6 |
| 17 | P4.7 | 10 | D7 |
| 18 | 3.3 V |  | NC |
| 19 | 5.0 V | 1 | 5.0 V |
| 20 | GND | **2** | GND |

**Program:** Attach printout of the tested code.

**Result:**

Observe square and triangular wave on DSO.

**Conclusion:**

**---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

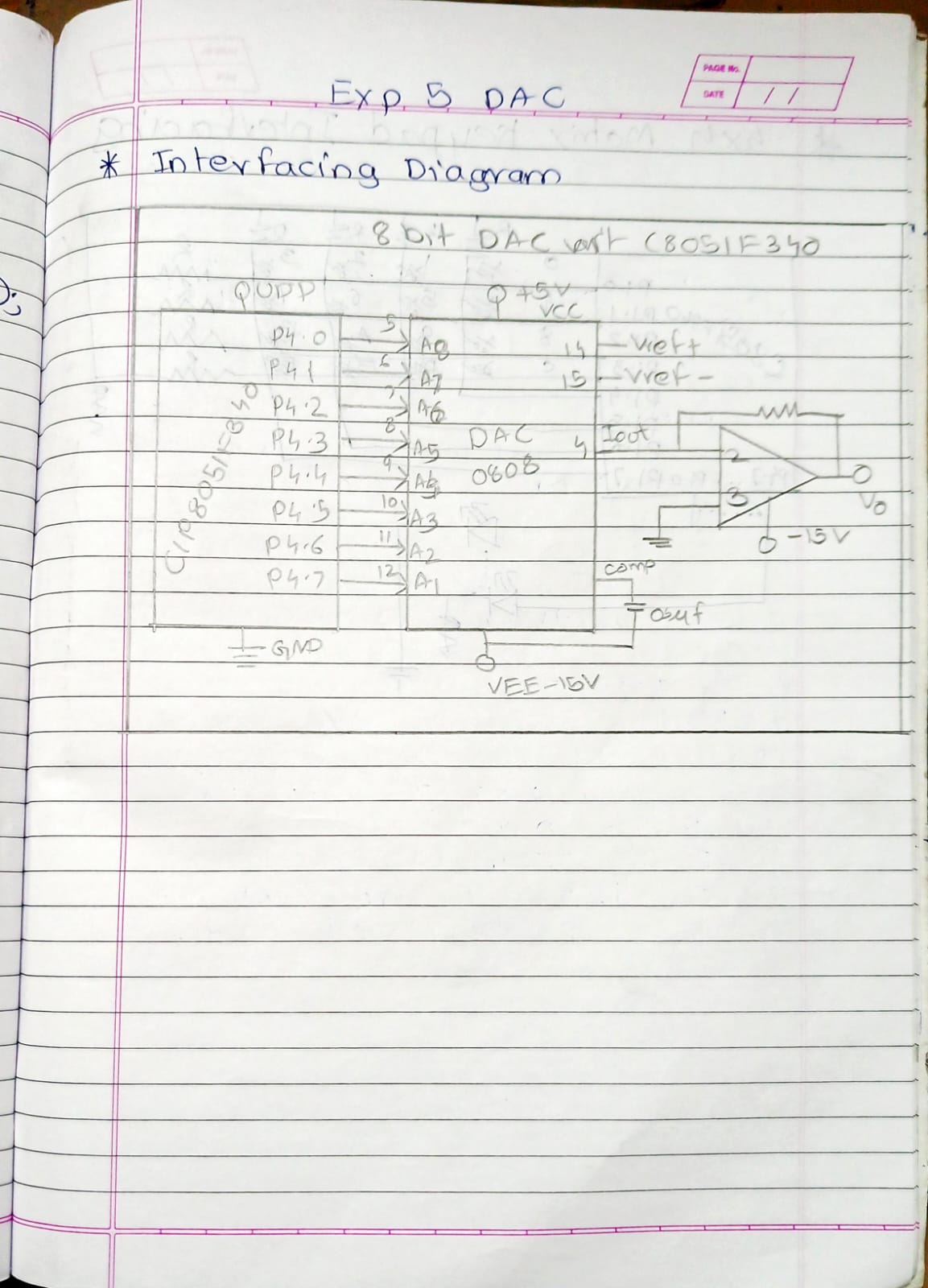
**Study Questions:**

1. Write a program to generate trapezoidal wave using DAC
2. Explain different types of DAC

**Additional Links:**

[*https://nptel.ac.in/courses/112103174/module2/lec8/1.html*](https://nptel.ac.in/courses/112103174/module2/lec8/1.html)

**Interfacing Diagram of 8-bit DAC with C8051F340**



**DAC Interfacing with C8051F340 for Square Waveform:**

// Exp - 5 DAC Interfacing with C8051F340

/\*

Name: Shreerang Mhatre

Rollno: 52

Batch: A3

Class: TY

DAC\_Square Waveform

\*/

#include"c8051f340.h"

void delay(unsigned int Ms);

void main(){

    P4MDOUT=0xff;

    while(1){

        P4=~P4;

        delay(50);

    }

}

void delay(unsigned int Ms){

    unsigned int n;

    unsigned int i;

    for(n=0;n<Ms;n++){

        for(i=0;i<65;i++);

    }

}

**DAC Interfacing with C8051F340 for Triangular Waveform:**

// Exp - 5 DAC Interfacing with C8051F340

/\*

Name: Shreerang Mhatre

Rollno: 52

Batch: A3

Class: TY

DAC\_Triangular Waveform

\*/

#include"c8051f340.h"

void main(){

    int i;

    P4MDOUT=0xff;

    while(1){

        for(i=0; i<=254;i++){

            P4=i;

        }

        for(i=255; i>=1; i--){

            P4=i;

        }

    }

}

**DAC Interfacing with C8051F340 for Sawtooth Waveform:**

// Exp - 5 DAC Interfacing with C8051F340

/\*

Name: Shreerang Mhatre

Rollno: 52

Batch: A3

Class: TY

DAC\_Sawtooth Waveform

\*/

#include"c8051f340.h"

void main(){

    int i;

    P4MDOUT=0xff;

    while(1){

        for(i=0; i<=254;i++){

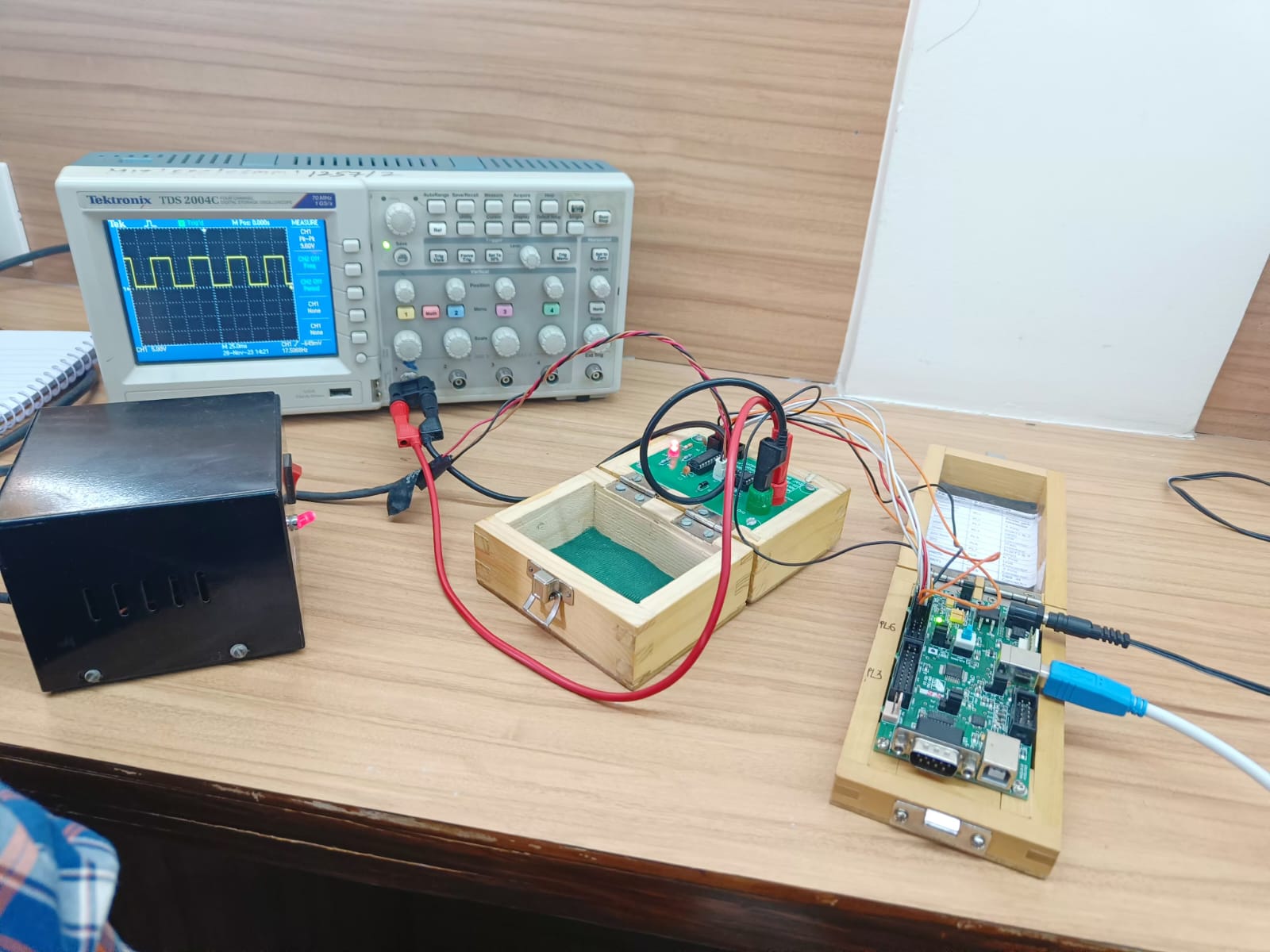
            P4=i;

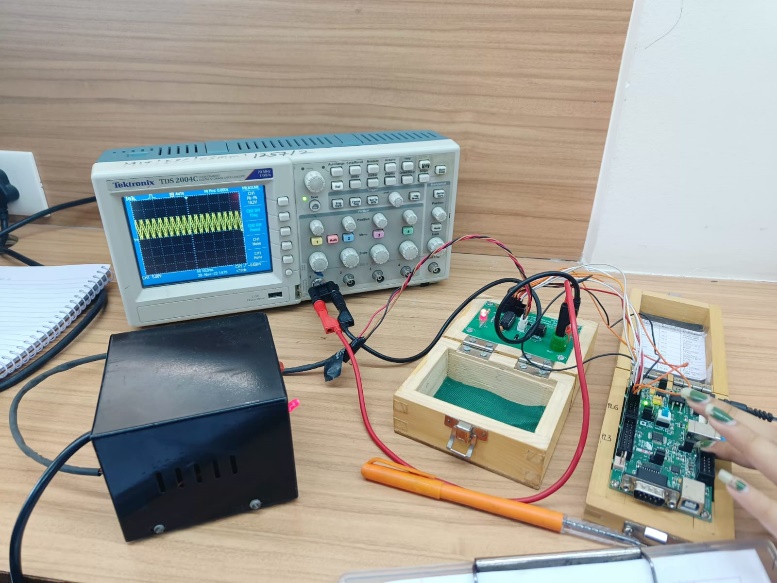
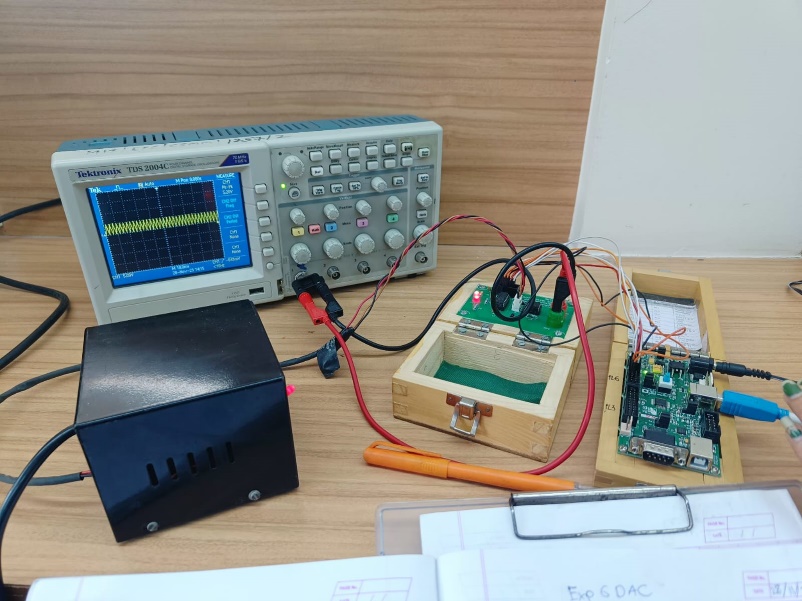
        }

    }

}

**Output for Square Waveform:**



**Output for Triangular Waveform:**

**Output for Sawtooth Waveform:**

