Microcontroller and Microprocessors
Experiment-4
Input Output Ports And Generation Of Waveforms

Varun Singh Inda
16BEE0023

Aim :- To check input output ports using the blinking of LEDs and the to see the formation of square wave and sine wave using microcontroller.

Procedure :-
1. Click on START.
2. Open Keil µvision5.
3. Open PROJECT. Create New µvision project.
4. Open Legacy Device Database.
5. Click on + next to NXP in the application box that opens.
6. Select P89V51RD2 from the list under NXP.
7. Click OK.
8. Proceed further clicking Yes.
9. In the main screen, select blank page icon present under File.
10. A new Text Window opens, where we are to write the program to be executed. 11. Go to File in the menu bar and save the program with the extension .asm.
12. In project window, select target on clicking + and chose Add existing file to source group1 and chose the program to be executed.
13. On clicking + next to SOURCE GROUP, right click, build target.
14. Click on Debug icon.
15. Start debug session.
16. Click on OK when the window pops in.
17. Press RUN or F5.

PROCEDURE FOR FLASH MAGIC:
1. Click on START.
2. Open Flash Magic.
3. Select 89V51RD2, COM1, Baud Rate as 9600.
4. Select Erase All Flash.
5. Chose the hex file.
6. Click on Verify after Programming.
7. Click on Start.

ALGORITHM:
1. MODE 1 of timer0 is selected.
2. Inside loop L, A register is made high.

3. Value of A is moved to port P1.
4. Delay subroutine is called.
5. Register A is made low.
6. Value of A is moved to port P1.
7. Jump to loop L.
8. Delay

Code :-

```
MOV TMOD, #01
L: MOV A, #0FFH
MOV P1, A
ACALL DELAY
MOV A, #00
MOV P1, A
ACALL DELAY
SJMP L
DELAY: MOV TL0,#00
MOV TH0, #00
SETB TR0
L1: JNB TF0, L1
CLR TR0
CLR TF0
RET
```

Algorithm For 1 Glowing LED:

1. MODE 1 of timer0 is selected.
2. Inside loop L, A register is made high.
3. Value of A is moved to port P1.
4. Delay subroutine is called.
5. Register A is made low.
6. Value of A is moved to port P1.
7. Jump to loop L.
8. Delay

Code for 1 LED:-

```
MOV TMOD, #01
L: MOV A, #00000001B
MOV P1, A
ACALL DELAY
MOV A, #00
```

```
MOV P1, A
ACALL DELAY
SJMP L
DELAY: MOV TL0,#00
MOV TH0, #00
SETB TR0
L1: JNB TF0, L1
CLR TR0
CLR TF0
RET
```

ALGORITHM for Rotate Left:

1. MODE 1 of timer0 is selected.
2. Inside loop L, A register is made high.
3. Value of A is moved to port P1.
4. Delay subroutine is called.
5. RL A command makes the bits of Register shift left.
6. Jump to loop L.
7. Delay subroutine is declared.
8. End

For RL, ROTATE LEFT:

```
MOV TMOD, #01
L: MOV A, #0FFH
MOV P1, A
ACALL DELAY
RL A
SJMP L
DELAY: MOV TL0,#00
MOV TH0, #00
SETB TR0
L1: JNB TF0, L1
CLR TR0
CLR TF0
RET
```

ALGORITHM for Rotate Right:

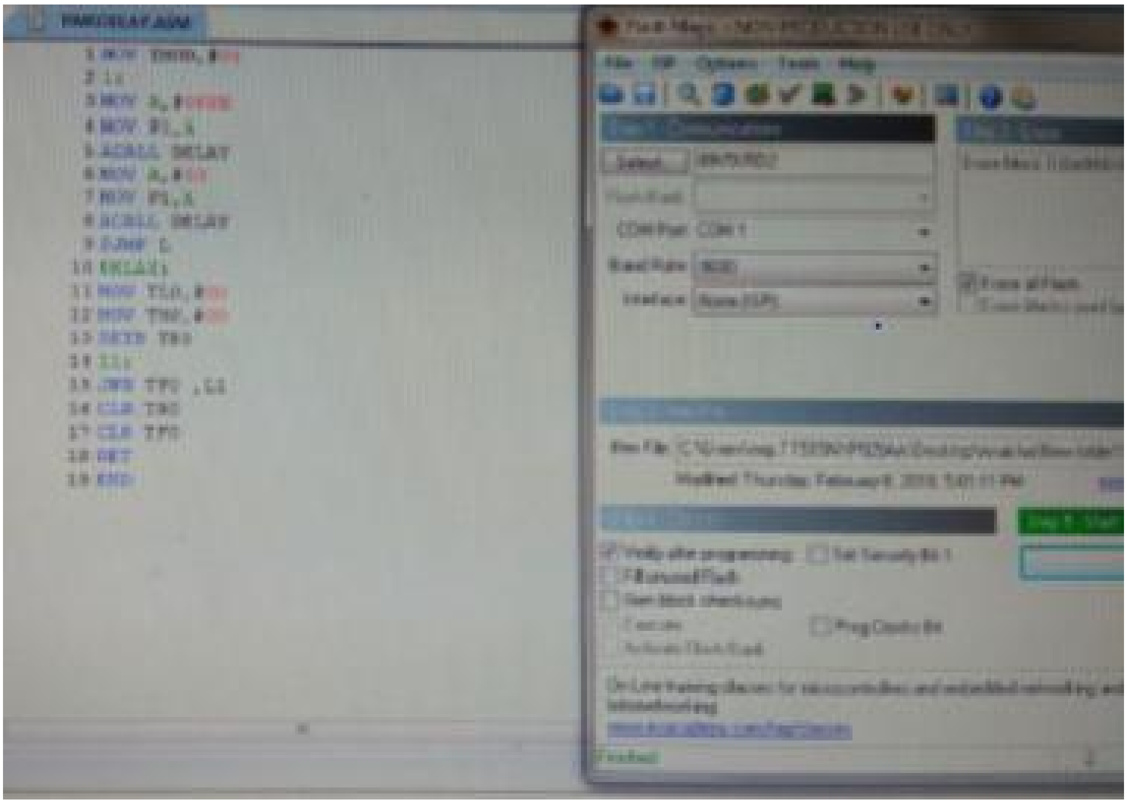
1. MODE 1 of timer0 is selected.
2. Inside loop L, A register is made high.
3. Value of A is moved to port P1.
4. Delay subroutine is called.
5. RL A command makes the bits of Register shift right.

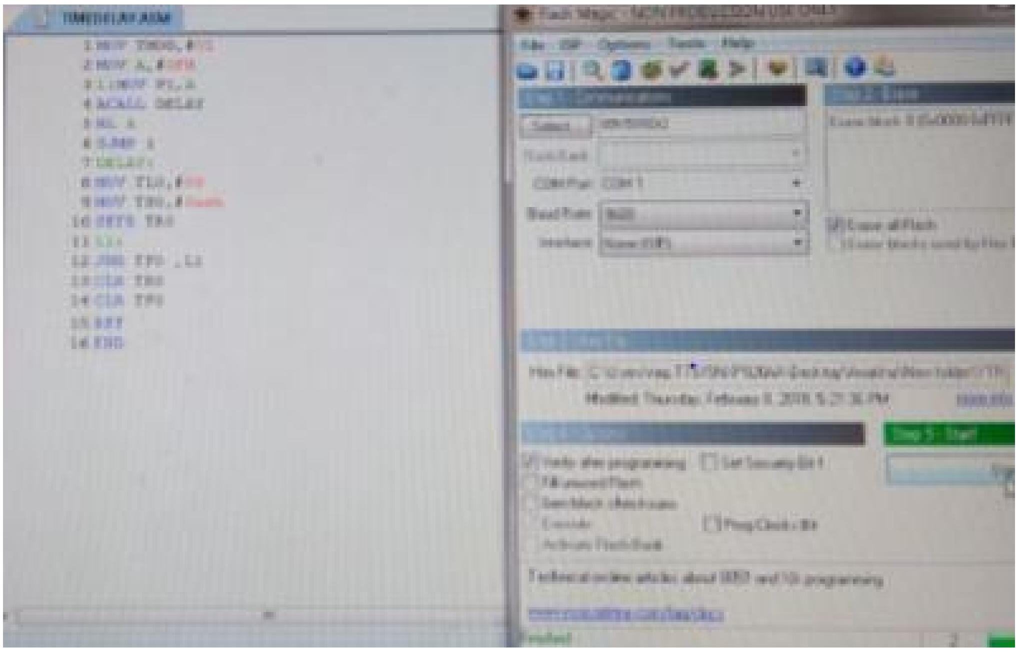
6. Jump to loop L.
7. Delay subroutine is declared.
8. End

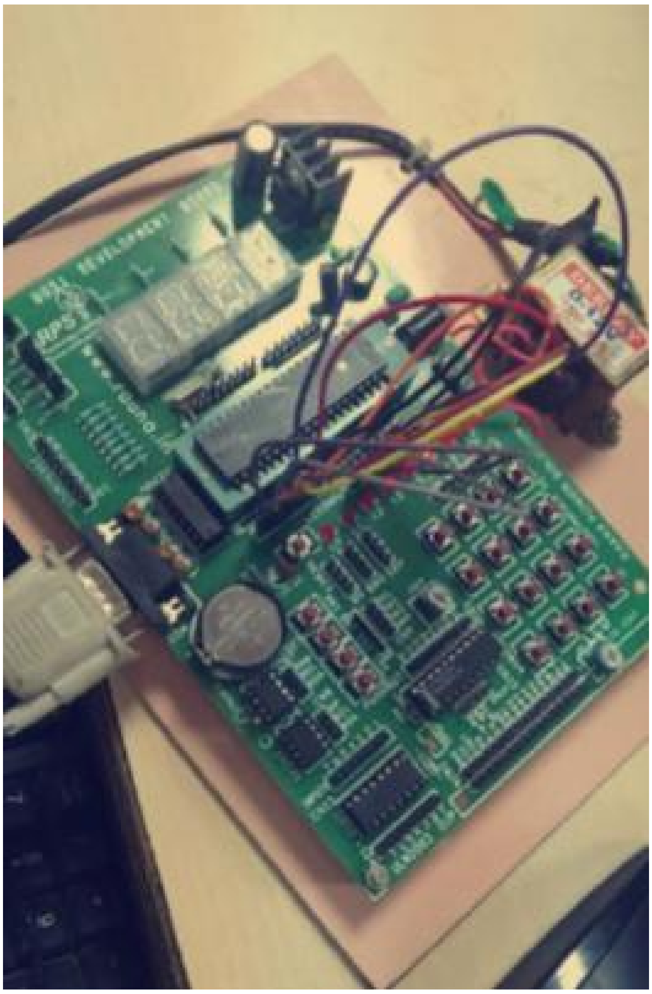
For RR, ROTATE RIGHT:

```
MOV TMOD, #01
L: MOV A, #0FFH
MOV P1, A
ACALL DELAY
RR A
SJMP L
DELAY: MOV TL0,#00
MOV TH0, #00
SETB TR0
L1: JNB TF0, L1
CLR TR0
CLR TF0
RET
```

<u>Pictures</u>




TIMEDELAY.ASM
ACALL DELAY
DELAY:
SETB TR0
JNB TF0 ,L1
CLR TR0
CLR TF0
RET
END
Flash Magic
COM Port:
Baud Rate:
Interface:
Erase all Flash
Verify after programming
Set Security Bit 1
Step 5 - Start!
Technical on-line articles about 8051 and XA programming



Conclusion :- The experiment was successfully performed with the output of blinking of LEDs being noted along with the performance of left shift and right shift of the LEDs.

Waveform Generation

PROCEDURE:

1. Click on START.
2. Open Keil µvision5.
3. Open PROJECT. Create New µvision project.
4. Open Legacy Device Database.
5. Click on + next to NXP in the application box that opens.
6. Select P89V51RD2 from the list under NXP.
7. Click OK.
8. Proceed further clicking Yes.
9. In the main screen, select blank page icon present under File.
10. A new Text Window opens, where we are to write the program to be executed. 11. Go to File in the menu bar and save the program with the extension .asm.
12. In project window, select target on clicking + and chose Add existing file to source group1 and chose the program to be executed.
13. On clicking + next to SOURCE GROUP, right click, build target.
14. Click on Debug icon.
15. Start debug session.

16. Click on OK when the window pops in.
17. Press RUN or F5.

<u>ALGORITHM For Square Wave:</u>

1. Mode 1 of Timer0 is selected.
2. Values of TH0 and TF0 is fed for which the value set is FF00.
3. The input at 0th bit of P1 port is complemented
4. Delay is called.
5. The loop of complementing the bit is repeated for FF00-0000+1 cycles.
6. In the delay subroutine, till TF0 becomes 0, the loop is repeated, followed by clearing of TR0 and TF0 once the agenda is achieved.
7. The program returns to the main loop.
8. End

<u>SQUARE WAVE :</u>

```
MOV TMOD, #01
L: MOV TL0, #00H
MOV TH0, #FFH
CPL P1.0
ACALL DELAY
SJMP L
DELAY: SETB TR0
L1: JNB TF0, L1
CLR TR0
CLR TF0
RET
```

<u>Procedure for Sine Wave :</u>

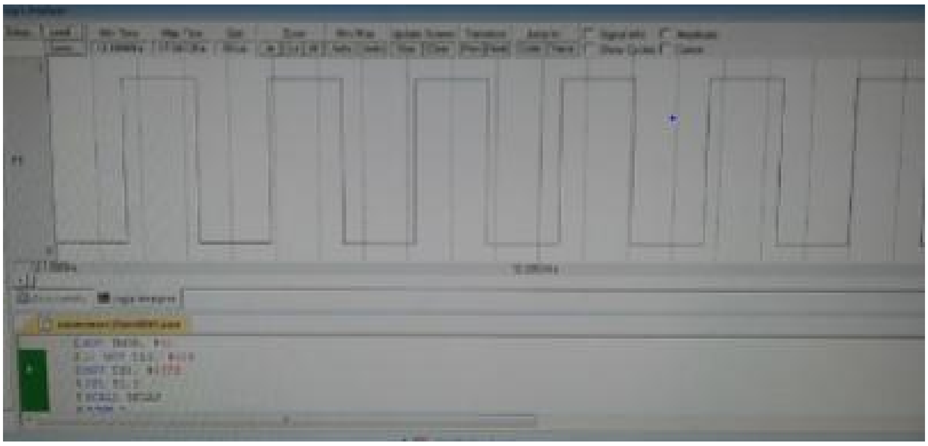
To generate a sine wave, we first need a table whose values represent the magnitude of the sine of angles between 0 and 360 degrees. The values for the sine function vary from -1.0 to +1.0 for 0- to 360-degree angles. Therefore, the table values are integer numbers representing the voltage magnitude for the sine of theta. This method ensures that only integer numbers are output to the DAC by the 8051 microcontroller. Table 13-7 shows the angles, the sine values, the voltage magnitudes, and the integer values representing the voltage magnitude for each angle (with 30-degree increments). To generate Table 13-7, we assumed the full-scale voltage of 10 V for DAC output (as designed in Figure 13-18). Full-scale output of the DAC is achieved when all the data inputs of the DAC are high. Therefore, to achieve the full-scale 10 V output, we use the following equation.

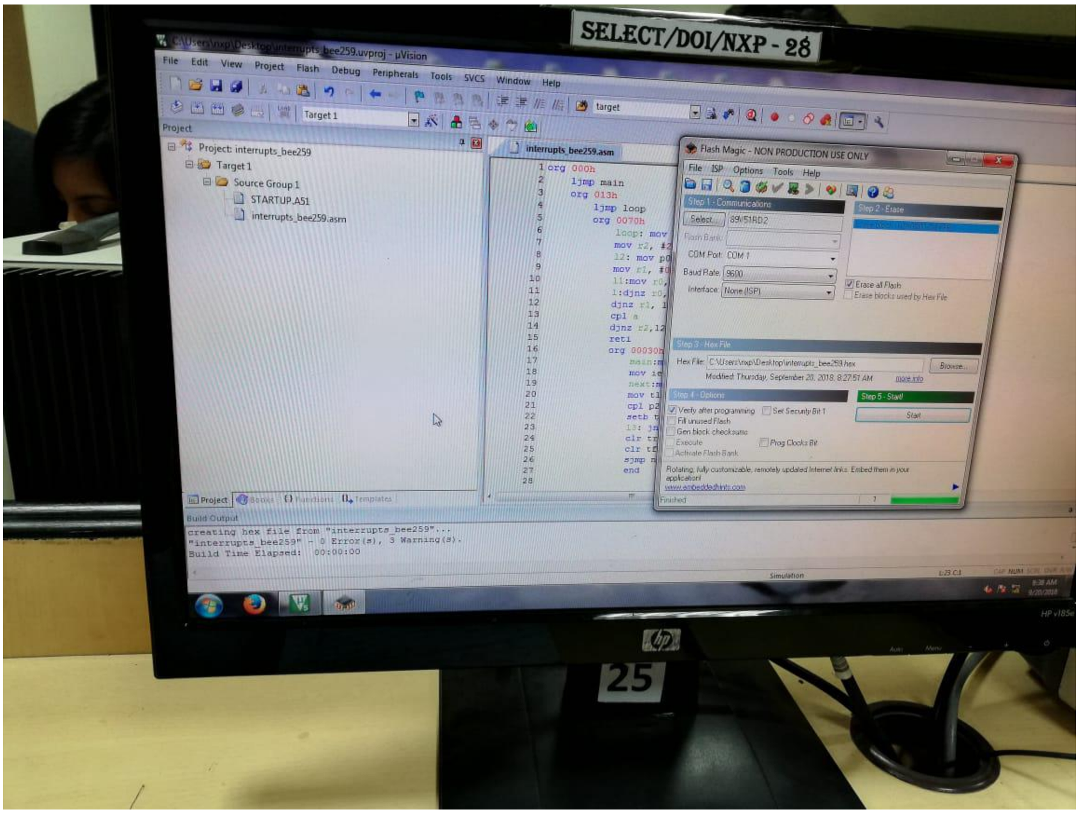
$$V_{out} = 5\text{ V} + (5 \times \sin\theta)$$

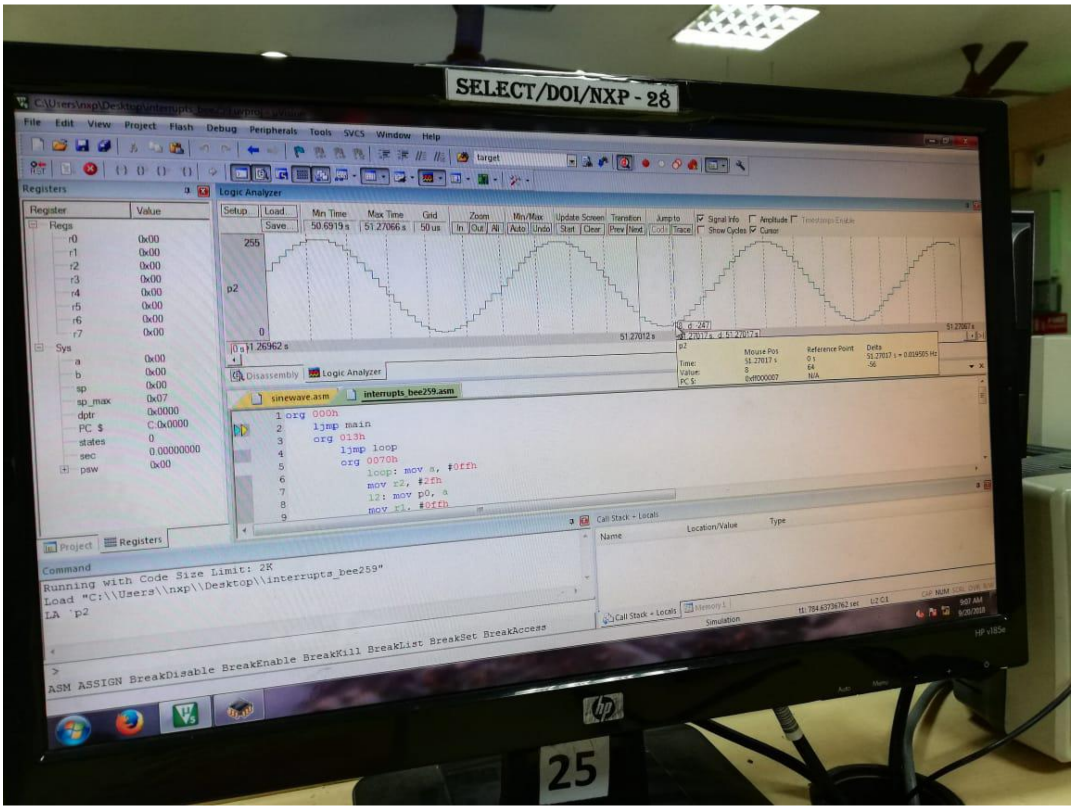
Sine Wave Code :-

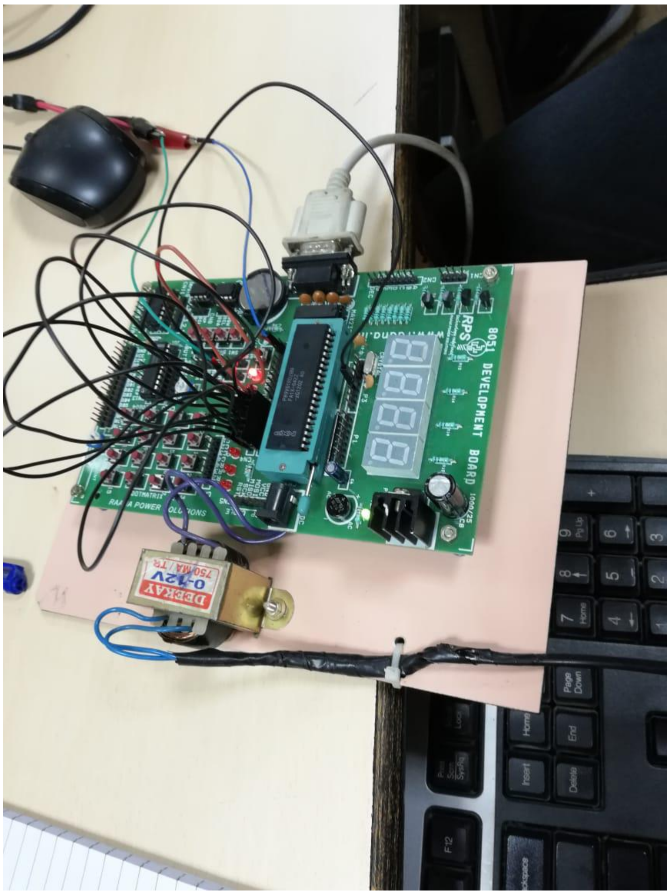
```
ORG 0000H
CLR A
UP : MOV DPTR,#SINE
MOV R0,#36
LABEL: MOVC A, @A+DPTR.
MOV P2,A
CLR A
INC DPTR
DJNZ R0 , LABEL
SJMP UP
ORG 050H
SINE :
DB 128,150,172,192,210,226,239,248,254,255,254,248,239,226,
     172,150,128,106,84,64,46,30,17d 9,2,0,2,8,17,30,46,64,8
     END
```

Picture




SELECT/DOI/NXP - 28
Flash Magic - NON PRODUCTION USE ONLY
creating hex file from "interrupts_bee259"...
"interrupts_bee259" - 0 Error(s), 3 Warning(s).
Build Time Elapsed: 00:00:00
25


SELECT/DOI/NXP - 28
Logic Analyzer
Running with Code Size Limit: 2K
LA `p2
ASM ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess
25



<u>Conclusion :-</u> The waveforms have been generated using the microcontroller 8051 and the observed waveforms are similar to the theoretical reference.