

Course Name:	Microprocessors and Microcontrollers Laboratory	Semester:	IV
Date of Performance:	08 / 04 / 2024	Batch No.:	A - 2
Faculty Name:	Kirti Sawlani	Roll No.:	16014022050
Faculty Sign & Date:		Grade / Marks:	___ / 25

Experiment No.: 6
Title: Generating a Square Wave using Timer of 8051

Aim and Objective of the Experiment:

Aim:

1. Write an 8051 based ALP to generate 5Khz square wave on port P2.0. Assume the oscillator to be running at 11.0592 Mhz.

Objectives:

Study of –

1. Hardware delay.
2. Port of 8051.
3. SFRs related to ports and timer.
4. Using ports in various modes.

COs to be achieved:

CO3: Understand the internal design of 8051 microcontrollers along with its features.

Theory:

8051 microcontroller is designed by Intel in 1981. It is an 8-bit microcontroller. It is built with 40 pins DIP (dual inline package), 4kb of ROM storage and 128 bytes of RAM storage, 2 16-bit timers. It consists of are four parallel 8-bit ports, which are programmable as well as addressable as per the requirement. An on-chip crystal oscillator is integrated in the microcontroller having crystal frequency of 12 MHz.

Hardware timers:

Hardware timers are used for timing and counting operations, allowing the processor to carry on with some other process while the timer process runs. A clock input drives a counting register to measure time or count external events. Its functionality can be extended by using additional registers to store timer values

The 8051 has two timers, Timer 0 and Timer 1. They can be used as timers or as event counters. Both Timer 0 and Timer 1 are 16-bit wide. Since the 8051 follows an 8-bit architecture, each 16 bit is accessed as two separate registers of low-byte and high-byte.

Ports in 8051:

8051 microcontrollers have 4 I/O ports each of 8-bit, which can be configured as input or output. Hence, total 32 input/output pins allow the microcontroller to be connected with the peripheral devices.

Stepwise-Procedure:

1. Write an assembly program to achieve the aim.
2. Build the program using Keil microvision-3 IDE and generate the HEX file.
3. Program the given 8051 device on the Flash programmer.
4. Connect the output pin mentioned in the code to the oscilloscope input.
5. Run the program on the development board and observe the square wave on the oscilloscope.

Circuit diagram / Flowchart / Algorithm:

Start

Initialize SQPIN as output

Initialize Timer0 in mode 1

Loop:

Set SQPIN ON

Call Delay() function

Set SQPIN OFF

Call Delay() function

Endless Loop

Delay() Function:

Set TH0 and TL0 for a 10ms delay

Start Timer0

Wait until Timer0 overflows (TF0 becomes 1)

Stop Timer0

Clear Timer0 overflow flag (TF0)

End Delay() Function

End

Assembly Language Program:

Attach list file with appropriate comments.

C51 COMPILER V9.01 TRY2

04/19/2024 13:13:23

PAGE 1

C51 COMPILER V9.01, COMPILATION OF MODULE TRY2

OBJECT MODULE PLACED IN try2.OBJ

COMPILER INVOKED BY: C:\Keil\C51\BIN\C51.EXE try2.c BROWSE DEBUG

OBJECTEXTEND

line level source

1 #include <reg51.h>

2

3 sbit SQPIN = P2^7; // Assuming SQPIN is at PORT 3 pin 3

4

5 void Delay(void); // Delay function declaration

6

7 void main() // main function

8 {

9 1 SQPIN = 0; // Set SQPIN as output

10 1

11 1 TMOD = 0x01; // Timer0 mode1: 16-bit timer with auto-reload

12 1

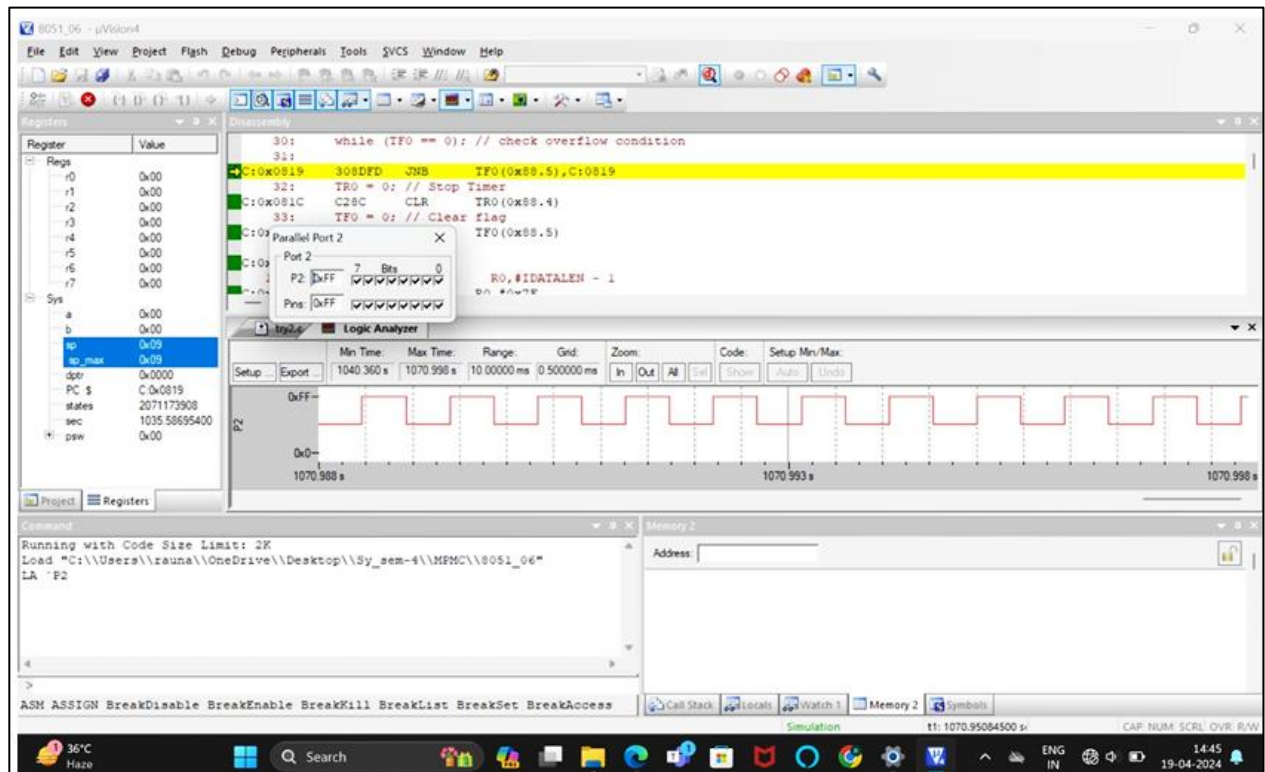
```
13 1    while (1) // infinite loop
14 1    {
15 2        SQPIN = 1; // Set SQPIN ON
16 2        Delay();
17 2
18 2        SQPIN = 0; // Set SQPIN OFF
19 2        Delay();
20 2    }
21 1    }
22
23    void Delay()
24    {
25 1        TH0 = 0xFC; // initial value for approximately 10ms delay
26 1        TL0 = 0x67;
27 1
28 1        TR0 = 1; // timer0 start
29 1
30 1        while (TF0 == 0); // check overflow condition
31 1
32 1        TR0 = 0; // Stop Timer
33 1        TF0 = 0; // Clear flag
34 1    }
```

MODULE INFORMATION: STATIC OVERLAYABLE

```
CODE SIZE      = 33  ---
CONSTANT SIZE  = ----  ---
XDATA SIZE     = ----  ---
PDATA SIZE     = ----  ---
DATA SIZE      = ----  ---
IDATA SIZE     = ----  ---
BIT SIZE       = ----  ---
```

END OF MODULE INFORMATION.**C51 COMPILATION COMPLETE. 0 WARNING(S), 0 ERROR(S)**

Output Screenshots:



Post Lab Subjective/Objective type Questions:

1. Write a C program to generate 5 KHz square wave of 70% duty cycle from port 1.5 using timer 1. Assume oscillator frequency as 11.0592 MHZ.
 Attach C file of the program with suitable comments and snapshot of result.

```
#include <reg51.h>
```

```
#include <reg51.h>
```

```
sbit SQPIN = P1^5; // Output pin for the square wave
```

```
void Delay(void); // Delay function declaration
```

```
void main() // main function
```

```
{
```

```
    SQPIN = 0; // Set SQPIN as output
```



K. J. Somaiya College of Engineering, Mumbai-77
(A Constituent College of Somaiya Vidyavihar University)
Department of Electronics Engineering

Microprocessor and Microcontrollers

Laboratory

Semester: IV Academic Year: 2023-24

Roll no. 16014022042

```
TMOD = 0x10; // Timer1 model: 16-bit timer with auto-reload
```

```
while (1) // infinite loop
```

```
{
```

```
    SQPIN = 1; // Set SQPIN ON
```

```
    Delay();
```

```
    SQPIN = 0; // Set SQPIN OFF
```

```
    Delay();
```

```
}
```

```
}
```

```
void Delay()
```

```
{
```

```
    // Calculate reload value for Timer1 for generating 5 KHz frequency
```

```
    unsigned int reload_value = (11059200UL / (2 * 5 * 1000));
```

```
    TH1 = -(reload_value >> 8); // Set initial value for Timer1 high byte
```

```
    TL1 = -(reload_value & 0xFF); // Set initial value for Timer1 low byte
```

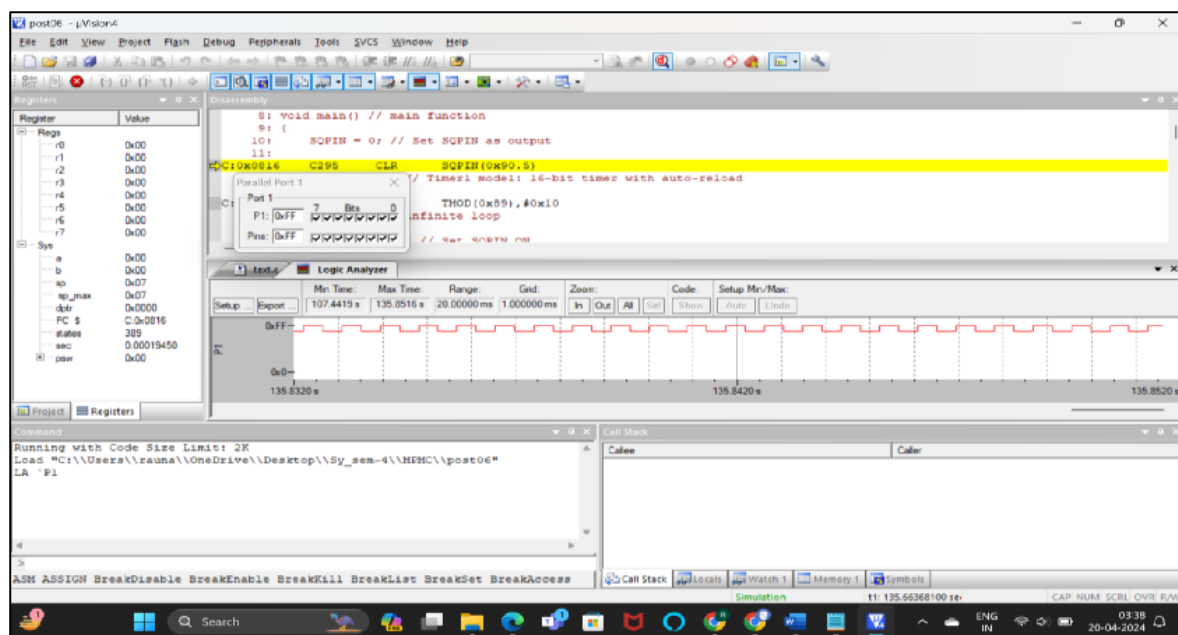
```
    TR1 = 1; // Start Timer1
```

```
    while (!TF1); // Wait until Timer1 overflows
```

```
    TR1 = 0; // Stop Timer1
```

```
    TF1 = 0; // Clear Timer1 overflow flag
```

```
}
```



Conclusion:

In this experiment we used Keil uVision4 and understood its dynamics while writing a code for square wave of given parameters and achieved all objectives i.e., Study of hardware delay, Study of Ports of 8051, SFRs related to Ports and timer and Using ports in various modes.

Signature of faculty in-charge with Date: