

Experiment No : 5

Aim : To study about Timer Programming .

Procedure:

- Create a square wave of 50% duty cycle (with equal portions high and low) on the P1.5 bit using timer mode 1 with timer 0 to generate the time delay. Display the output using edsim simulator. Use CPL to transition from hi to lo
- Modify TL and TH in question 1 to get the largest pulse width possible. Use SETB and CLR to transition from hi,lo
- Assume that XTAL = 16 MHz. Write A program for timer 0 to create a pulse width of 5 ms on P2.3 in mode 1.
- Create a square wave (with 75%portion high and 25% portion low) on the P1.5 bit using timer mode 1 with timer 0 to generate the time delay. Display the output using edsim simulator. use CPL to transition from hi to lo
- Assume XTAL = 11.0592 MHz, generate a square wave of 50% duty cycle on P1.0 using timer 1 in mode 2
- Assume XTAL = 11.0592 MHz, generate a time delay of 50 counts,200 times, using timer1 mode2. find the total time delay generated

Programs :

- Program 1**

```
MOV TMOD,#01
```

```
CLR P0.7
```

```
HERE: MOV TL0,#0F2H
```

```
MOV TH0,#0FFH
```

```
CPL P1.5
```

```
ACALL DELAY
```

```
SJMP HERE
```

```
DELAY: SETB TR0
```

```
AGAIN: JNB TF0,AGAIN
```

```
CLR TR0
```

```
CLR TF0
```

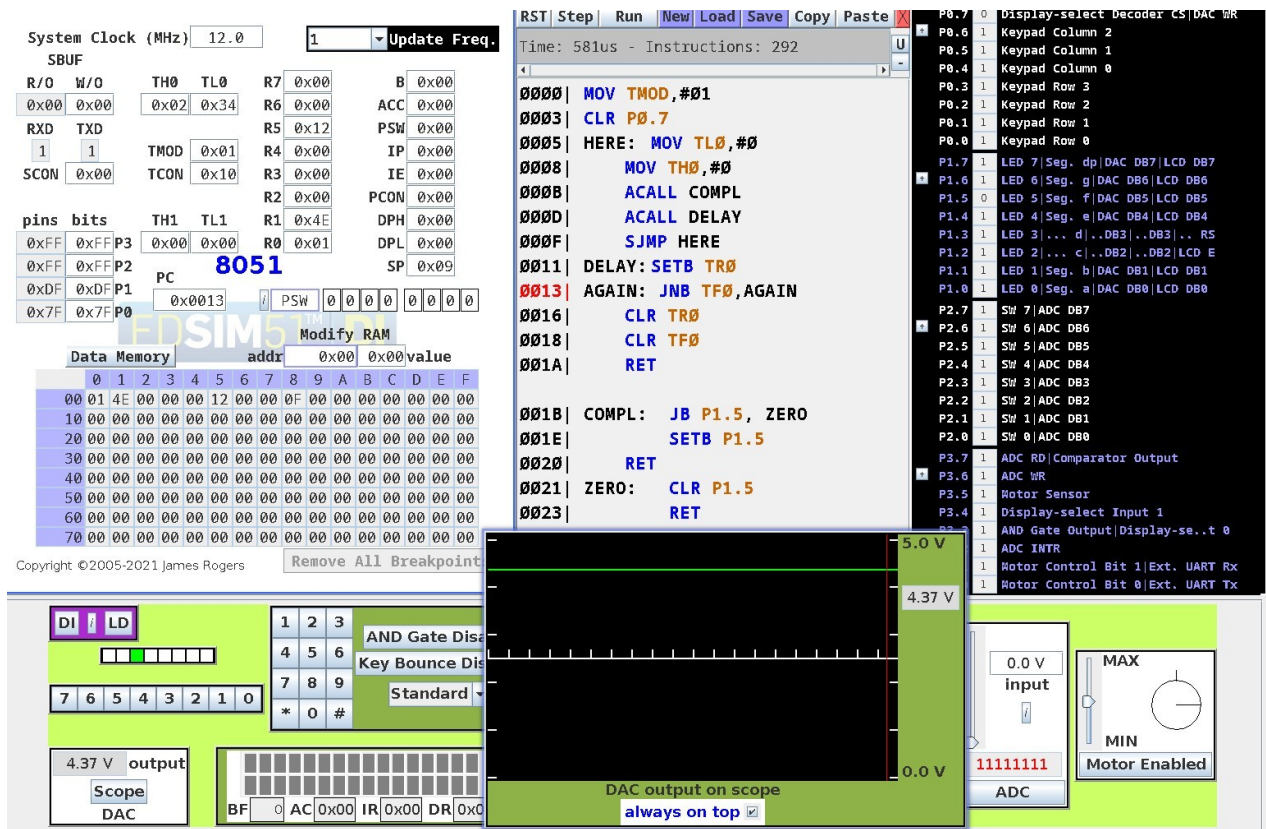
```
RET
```

Output :

The screenshot displays the Proteus simulation environment. On the left, the I/O registers window shows the status of various hardware components, including the 8051 microcontroller registers (R0-R7, ACC, PSW, IP, IE, PCON, DPH, DPL, SP) and the DAC output (P0.7). The central window shows the assembly code being executed, with the instruction pointer (PC) at 0000. The right window shows the pin list for the 8051 microcontroller, including P0.7, P0.6, P0.5, P0.4, P0.3, P0.2, P0.1, P0.0, P1.7, P1.6, P1.5, P1.4, P1.3, P1.2, P1.1, P1.0, P2.7, P2.6, P2.5, P2.4, P2.3, P2.2, P2.1, P2.0, P3.7, P3.6, P3.5, P3.4, P3.3, P3.2, P3.1, P3.0, P4.7, P4.6, P4.5, P4.4, P4.3, P4.2, P4.1, P4.0, P5.7, P5.6, P5.5, P5.4, P5.3, P5.2, P5.1, P5.0, P6.7, P6.6, P6.5, P6.4, P6.3, P6.2, P6.1, P6.0, P7.7, P7.6, P7.5, P7.4, P7.3, P7.2, P7.1, P7.0. The bottom window shows a scope plot of the DAC output, which is a square wave between 0.0 V and 5.0 V. The scope is labeled 'DAC output on scope' and 'always on top'.

- Program 2**
 MOV TMOD,#01
 CLR P0.7
 HERE: MOV TL0,#0
 MOV TH0,#0
 ACALL COMPL
 ACALL DELAY
 SJMP HERE
 DELAY: SETB TR0
 AGAIN: JNB TF0,AGAIN
 CLR TR0
 CLR TF0
 RET
 COMPL: JB P1.5, ZERO
 SETB P1.5
 RET
 ZERO: CLR P1.5
 RET

Output :



- Program 3**
 CLR P2.3
 MOV TMOD,#01
 CLR P0.7
 HERE: MOV TL0,#0F5h
 MOV TH0,#0E5h
 SETB P2.3
 SETB TR0
 ACALL AGAIN
 JMP HERE
 AGAIN: JNB TF0,AGAIN
 CLR P2.3
 CLR TR0
 CLR TF0
 RET

Output :

The screenshot displays the Proteus simulation environment. The top section shows the register window with various registers (R0-R7, B, ACC, PSW, IP, IE, PCON, DPH, DPL, SP) and their current values. The memory window shows the program memory starting at address 0000. The instruction window displays the assembly code being executed, with the current instruction being 'HERE: MOV TL0,#0F5h'. The scope window shows the DAC output on scope, which is currently at 0.0 V. The bottom section shows the hardware components, including a keypad, a display, and a motor.

Register Window:

R/O	W/O	TH0	TL0	R7	B
0x00	0x00	0xE6	0x75	0x00	0x00
R6	0x00	ACC	0x00	R5	0x12
R4	0x00	IP	0x00	R3	0x00
R2	0x00	PCON	0x00	R1	0x4E
R0	0x01	DPL	0x00	SP	0x09

Memory Window:

addr	0x00	0x00	value
0	1	2	3
4	5	6	7
8	9	A	B
C	D	E	F

Instruction Window:

```

0000| CLR P2.3
0002| MOV TMOD,#01
0005| CLR P0.7
0007| HERE: MOV TL0,#0F5h
000A| MOV TH0,#0E5h
000D| SETB P2.3
000F| SETB TR0
0011| ACALL AGAIN
0013| JMP HERE
0015| AGAIN: JNB TF0,AGAIN
0018| CLR P2.3
001A| CLR TF0
001C| RET
  
```

Scope Window:

DAC output on scope
always on top

5.0 V
0.0 V

11111111
ADC

MAX
MIN
Motor Enabled

- Program 4**

CLR P1.5

MOV TMOD,#01

CLR P0.7

HERE: ACALL DELAY75

CPL P1.5

ACALL DELAY25

CPL P1.5

JMP HERE

DELAY75:MOV TL0, #0B4h

MOV TH0, #0FFh

SETB TR0

ACALL REPEAT

RET

DELAY25:MOV TL0, #0E6h

MOV TH0, #0FFh

SETB TR0

ACALL REPEAT

RET

REPEAT: JNB TF0, REPEAT

CLR TF0

CLR TR0

RET

Output :

The screenshot displays the EDISIM51 microcontroller simulator interface. The top panel shows the program execution status with a time of 995us and 527 instructions. The instruction list on the right shows the following code:

```

0000 CLR P1.5
0002 MOV TMOD,#01
0005 CLR P0.7
0007 HERE: ACALL DELAY75
0009 CPL P1.5
000B ACALL DELAY25
000D CPL P1.5
000F JMP HERE
0011 DELAY75:MOV TL0, #0B4h
0014 MOV TH0, #0FFh
0017 SETB TR0
0019 ACALL REPEAT
001B RET
001C DELAY25:MOV TL0, #0E6h
001F MOV TH0, #0FFh
0022 SETB TR0
  
```

The left panel shows the register and memory dump. The register dump includes R0-R7, ACC, PSW, IP, IE, PCON, DPH, DPL, SP, and PC. The memory dump shows the data memory from 0000 to 00FF. The bottom panel shows the hardware components, including a keypad, a display, and an ADC. The ADC output is shown as 11111111, and the DAC output is shown as 4.37 V.

- **Program 5**

MOV TMOD, #20h

CLR P1.0

CLR P0.7

MOV TH1, #0F0h

MOV TL1, #0F0h

HERE:SETB TR1

CALL DELAY

CPL P1.0

JMP HERE

DELAY: JNB TF1, DELAY

CLR TF1

CLR TR1

RET

Output :

The screenshot displays the Proteus 8.0 SP3 IDE with the following components:

- Assembly Code Window:** Shows the assembly code for the DAC program. The code includes instructions like `MOV TMOD, #20h`, `CLR P1.0`, `CLR P0.7`, `MOV TH1, #0F0h`, `MOV TL1, #0F0h`, `HERE: SETB TR1`, `CALL DELAY`, `CPL P1.0`, `JMP HERE`, `DELAY: JNB TF1, DELAY`, `CLR TF1`, `CLR TR1`, and `RET`.
- Register/Variable Window:** Displays the current state of registers and variables. Notable values include `R0 = 0x01`, `R1 = 0x4E`, `R2 = 0x00`, `R3 = 0x00`, `R4 = 0x00`, `R5 = 0x12`, `R6 = 0x00`, `R7 = 0x00`, `B = 0x00`, `ACC = 0x00`, `PSW = 0x00`, `IP = 0x00`, `IE = 0x00`, `PCON = 0x00`, `DPH = 0x00`, `DPL = 0x00`, `SP = 0x09`, `PC = 0x0015`, and `PSW = 00000000`.
- Data Memory Window:** Shows the memory contents. The address `0x0015` is highlighted, containing the value `0x00`.
- Scope Window:** Displays the DAC output on the scope. The output is a square wave between 0.0 V and 5.0 V, labeled "DAC output on scope" and "always on top".
- ADC Window:** Shows the ADC input and output. The input is 0.0 V, and the output is 11111111, labeled "MAX" and "Motor Enabled".
- Motor Control Window:** Shows the motor control status. The motor is enabled, indicated by the "MAX" label and the "Motor Enabled" text.

- Program 6**

```

;Set mode 2 for timer 1
MOV TMOD, #20h
CLR P1.0
; Set initial values for TH and TL
MOV TH1, #0CEh
MOV TL1, #0CEh
; Set number of loops
MOV R2, #200
SETB TR1
;Loop L1 till rollover
L1: JNB TF1, L1
    CLR TF1
;loop L1 till R2 becomes 0
    DJNZ R2, L1
END
  
```

Output :

The screenshot displays the Proteus simulation environment with the following components:

- Assembly Code Window:** Shows the assembly code for Program 6. The code sets timer mode 2, initializes TH1 and TL1 to 0CEh, sets the number of loops to 200, and enters a loop that clears the timer flag and decrements the loop counter until it reaches 0.
- Data Memory Window:** Shows the memory address 0051 (labeled 8051) with a value of 00. The memory is organized into a table with columns for address, data, and labels.
- Hardware Components:**
 - DI / LD:** A switch component.
 - AND Gate Disabled:** A component labeled "AND Gate Disabled".
 - Key Bounce Disabled:** A component labeled "Key Bounce Disabled".
 - Standard:** A component labeled "Standard".
 - 4.37 V output:** A voltage source component.
 - Scope DAC:** A component labeled "Scope DAC".
 - BF, AC, IR, DR:** A set of four components labeled BF, AC, IR, and DR.
 - U:** A component labeled "U".
 - Rx, Tx:** Components labeled "Rx" and "Tx".
 - Rx Reset, Tx Send:** Buttons labeled "Rx Reset" and "Tx Send".
 - 0.0 V Input:** A component labeled "0.0 V Input".
 - ADC:** A component labeled "ADC".
 - MAX, MIN, Motor Enabled:** A component labeled "MAX MIN Motor Enabled".

Total time delay generated:

Let frequency of clock pulse be f and the crystal frequency be C ; then;

$$f = C/12$$

$$= 11.0592/12$$

$$= 921\text{KHz}$$

Let time for one machine cycle be T ; then:

$$T = 1/f$$

$$= 1/921$$

$$= 1.085\mu\text{s}$$

Therefore total time delay(D) generated is

$$D = 50 * 200 * T$$

$$= 10.8\text{ms}$$

Conclusion :

Programs to study Timer were successfully implemented.

Deepraj Bhosale

181105016