Date: 08/08/2025

EXPERIMENT-2

AIM:

a). Write an Assembly language program for 8051 micro controller to complement the data in port-2 and external memory location 0FFC0H with time delay of 1 sec using timer 0 in mode 1 for crystal frequency of 11.0592MHz .

b). Write an Assembly language program for 8051 micro controller to complement the data in port-2 and external memory location 0FFC0H with time delay of 2 sec using timer 1 in mode 2 for crystal frequency of 11.0592MHz .

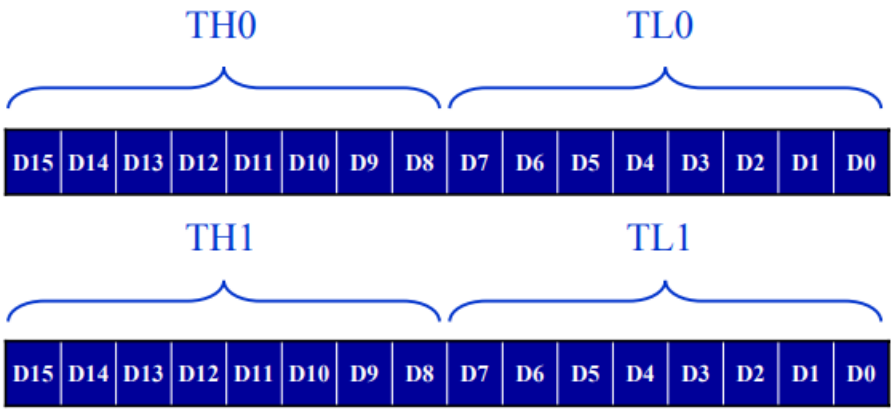
TOOLS REQUIRED: PC, Keil µvision5

THEORY:

- The 8051 has two timers/counters, they can be used either as
  - Timers to generate a time delay or as
  - Events counters to count events happening outside the microcontroller.
- Both Timer 0 and Timer 1 are 16 bits wide
  - Since 8051 has an 8-bit architecture, each 16-bit timer is accessed as two separate registers of low byte and high byte

- Accessed as low byte and high byte
  - The low byte register is called TL0/TL1 and
  - The high byte register is called TH0/TH1
  - Accessed like any other register

    - MOV TL0, #4FH
    - MOV R5, TH0



- Both timers 0 and 1 use the same register, called TMOD (timer mode), to set the various timer operation modes
- TMOD is 8-bit register
  - The lower 4 bits are for Timer 0
  - The upper 4 bits are for Timer 1
  - In each case

- The lower 2 bits are used to set the timer mode
- The upper 2 bits to specify the operation

PROCEDURE:

1. Turn on the computer, create a folder on D drive saved with Register Number.
2. Open Keil uVision5 in desktop, or windows start menu -> all programs->open Keil uVision5.

Creating Project:

3. Go to project ->click on new uVision project -> create a new folder saved with experiment number within the already existed register number folder in D drive mentioned in step 1, -> enter the project name -> click on save.
4. Select the device for target -> In devices ->Enter P89C51RD2XX in Search toolbar -> click on ok -> select **No** for dialog box message "Copy STARTUP.A51 to project folder and add files to project".

(or)

Choose NXP -> to select the device P89C51RD2XX -> click on ok -> select **No** for Copy STARTUP.A51 to project folder.

Creating Coding File:

5. Go to file -> click on new->go to save (choose the path to save the file, It is saved within the name of experiment number folder mentioned in step 3)-> enter a file name with extension **.asm** ->save the file.

Linking the Coding File to Project :

6. Right-click on Source group1 in project bar-> Add existing files to source group1-> choose the experiment number folder path and select all files in the folder -> select .asm code file ->click on add-> click on close.
7. Write the assembly language program in .asm code file and save it.

Executing the Code File:

8. Right-click on .asm code file->Click on Build target to check the errors (i.e 0-Errors,0-Warning)
9. Go to debug->Click on Start/Stop Debug Session -> click on ok for dialog box message "running code size limit 2K" -> and Click on RUN in debug label
10. Observe the output in Register windows, Memory windows, Serial window.

CALCULATIONS:

a) TIMER 0 MODE 1, 1SEC TIME DELAY

- Since XTAL = 11.0592 MHz
- 12 Machine Cycles = 1 Clock Cycles
- Single clock period = 11.0592MHz/12 = 921.6KHz
  = 1/921.6 = 1.085us
- Required delay = number of clocks X single clock period
  1sec = number of clocks X 1.085us
  Number of clocks = 9,21,658
- Mode 1 is 16 bit timer, Maximum clocks 65536
- Maximum time delay from 0000H state to FFFFH state is
  = 65536 X 1.085us = 71.1ms
- 1 sec = loop count X 71.1ms
  Loop count = 14.06 = 0EH
- Therefore, we have initial values in TH = #00H and TL = #00H, TMOD = #01H

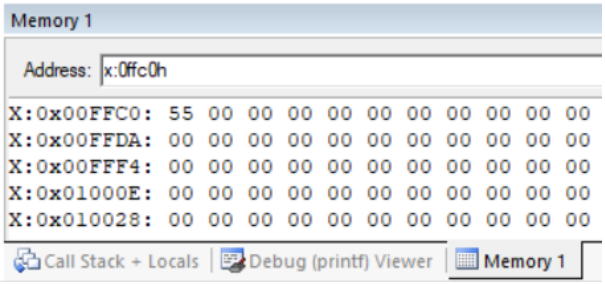
b) TIMER 1 MODE 2, 2SEC TIME DELAY

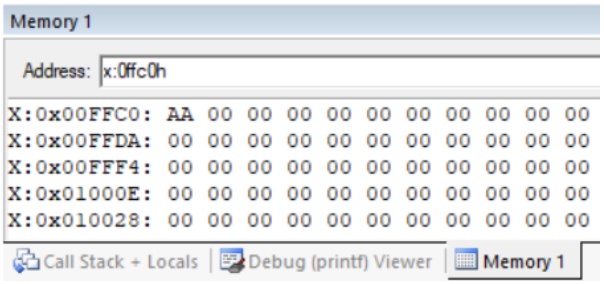
- Since XTAL = 11.0592 MHz,
- 12 Machine Cycles = 1 Clock Cycle
- Single clock period = 11.0592MHz/12 = 921.6Khz
  = 1/921.6KHz = 1.085us
- Required delay = number of clocks X single clock period
  2sec = number of clocks x 1.085us
  Number of clocks = 18,43,317
- Mode 2 is 8 bit timer, Maximum clocks 256
- Maximum time delay from 00H state to FFH state is
  = 256 X 1.085us = 277.76us
- 2 sec = loop count X 277.76us
  Loop count = 7200 = 1C20H
- Therefore, we have Initial values in TH = #00H, TMOD = #20H

OUTPUTS:

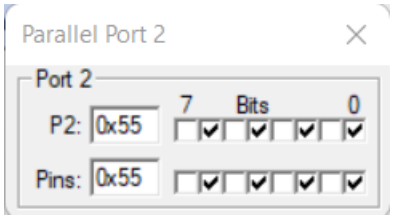
a)

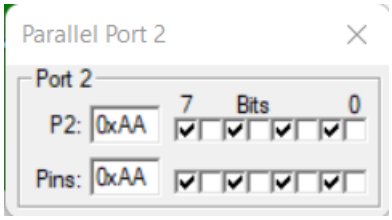
View -> Memory windows -> memory 1





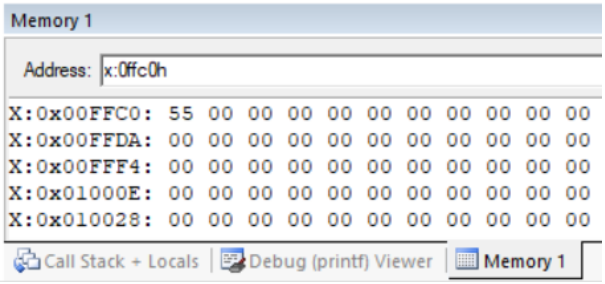
Peripherals -> I/O ports -> Port 2

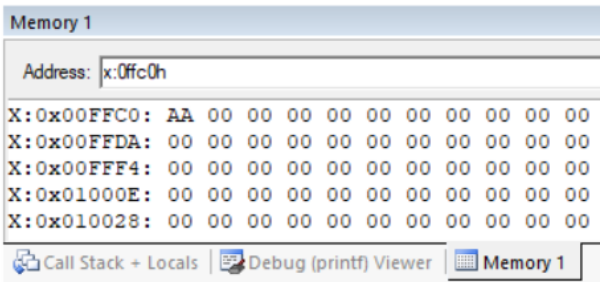




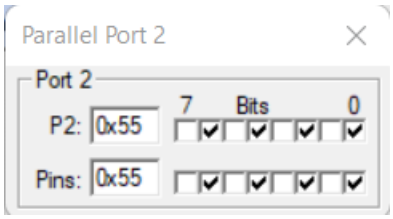
b)

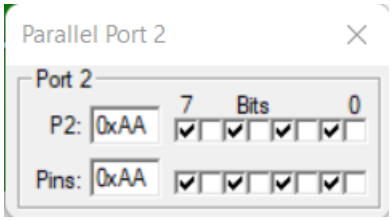
View -> Memory windows -> memory 1





Peripherals -> I/O ports -> Port 2





PROGRAMS:

a)

ADDRESS	OPCODES	LABELS	MNEMONICS	OPERANDS
0000			ORG	00H
0000	758901		MOV	TMOD,#01H
0003	7455		MOV	A,#55H
0005	90FFC0		MOV	DPTR,#0FFC0H
0008	1110	L2	ACALL	DELAY
000A	F0		MOVX	@DPTR,A
000B	F5A0		MOV	P2,A
000D	F4		CPL	A
000E	80F8		SJMP	L2
0010	780E	DELAY	MOV	R0,#0EH
0012	758A00	L1	MOV	TL0,#00H
0015	758C00		MOV	TH0,#00H
0018	D28C		SETB	TR0
001A	308DFD	HERE	JNB	TF0, HERE
001D	C28D		CLR	TF0
001F	C28C		CLR	TR0
0021	D8EF		DJNZ	R0,L1
0023	22		RET	
			END	

b)

ADDRESS	OPCODES	LABELS	MNEMONICS	OPERANDS
0000			ORG	00H
0000	758920		MOV	TMOD,#20H
0003	7455		MOV	A,#AAH
0005	90FFC0		MOV	DPTR, #0FFC0H
0008	1110	LOOP	ACALL	DELAY
000A	F4		CPL	A
000B	F0		MOVX	@DPTR,A
000C	F5A0		MOV	P2,A
000E	80F8		SJMP	LOOP
0010	791C	DELAY	MOV	R1,#1CH
0012	7A20	BACK2	MOV	R2,#20H
0014	758D00	BACK1	MOV	TH1,#00H
0017	D28E		SETB	TR1
0019	308FFD	HERE	JNB	TF1,HERE
001C	C28F		CLR	TF1
001E	C28E		CLR	TR1
0020	DAF2		DJNZ	R2,BACK1
0022	D9EE		DJNZ	R1,BACK2
0024	22		RET	
			END	

RESULT:

a) Assembly language program (ALP) using **Timer 0 in Mode 1** for a 1-second delay, to complement the data at **Port 2** and external memory location 0xFFC0, successfully executed.

b) Assembly language program (ALP) using **Timer 1 in Mode 2** for a 2-second delay, to complement the data at **Port 2** and external memory location 0xFFC0, successfully executed.