Date: 29/08/2025

4. PROGRAMS ON SERIAL COMMUNICATIONS

AIM: Write a program for serial transmission and serial reception with a baud rate of 9600bps.

TOOLS REQUIRED: PC, Keil µvision5

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 :

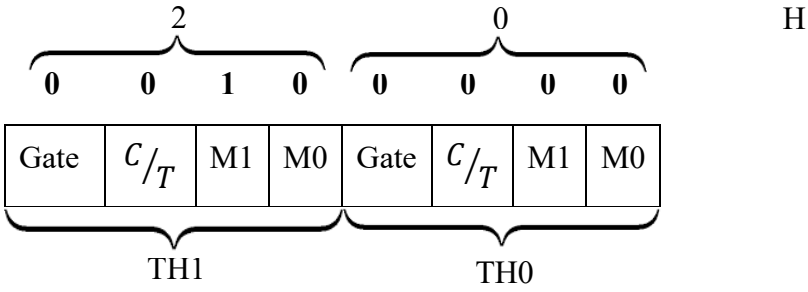
6. Right-click on Source group1 in project bar-> Add existing files to source group1-> choose the experiment number folder path and select all files in the folder -> select .asm code file ->click on add-> click on close.
7. Write the assembly language program in .asm code file and save it.

Executing the Code File:

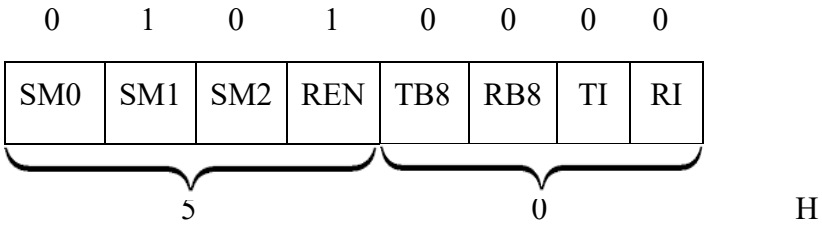
8. Right-click on .asm code file->Click on Build target to check the errors (i.e 0-Errors,0-Warning)
9. Go to debug->Click on Start/Stop Debug Session -> click on ok for dialog box message "running code size limit 2K" -> and Click on RUN in debug label
10. Observe the output in Register windows, Memory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

CALCULATION:

TMOD



$TH1 = \frac{-28800}{9600} = -3$



THEORY:

- SCON is an 8-bit register used to program the start bit, stop bit and data bits of data farming, among other things

SM0	SM1	SM2	REN	TB8	RB8	TI	RI

SM0	SCON.7	Serial port mode specifier
SM1	SCON.6	Serial port mode specifier
SM2	SCON.5	Used for multiprocessor communication
REN	SCON.4	Set/cleared by software to enable/disable reception
TB8	SCON.3	Not widely used
RB8	SCON.2	Not widely used
TI	SCON.1	Transmit interrupt flag. Set by HW at the begin of the stop bit mode 1. And cleared by SW
RI	SCON.0	Receive interrupt flag. Set by HW at the begin of the stop bit mode 1. And cleared by SW

Note: Make SM2, TB8, and RB8 =0

- SMO, SM1
  - They determine the farming of data by specifying the number of bits per character, and the start and stop bits

SM0	SM1	
0	0	Serial Mode 0
0	1	**Serial Mode 1, 8-bit data, 1 stop bit, 1 start bit**
1	0	Serial Mode 2
1	1	Serial Mode 3

Only mode 1 is of interest to us

- SM2
  - This enables the multiprocessing capability of the 8051
- REN (receive enable)
  - It is a bit-addressable register
    - When it is high, it allows 8051 to receive data on RxD pin
    - If low, the receiver is disable
- TI (transmit interrupt)
  - When 8051 finishes the transfer of 8-bit character
    - It raises TI flag to indicate that it is ready to transfer another byte
    - TI bit is raised at the beginning og the stop bit
- RI (receive interrupt)
  - When 8051 receives data serially via RxD, it gets rid of the start and stop bits and places the byte in SBUF register
    - It raises the RI flag bit to indicate that a byte has been received and should be picked up before it is lost
    - RI is raised halfway

Baud Rate	TH1 (Decimal)	TH1 (Hex)
9600	-3	FD
4800	-6	FA
2400	-12	F4
1200	-24	E8

- through the stop bit

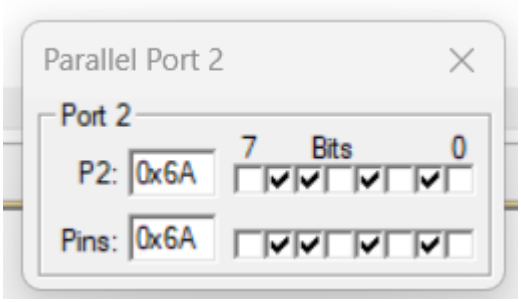
OUTPUT:

view → serial windows → #UART 1

Peripherals → I/O Ports → Port 2

Letter: J

ASCII: 106



PROGRAM:

ADDRESS	OPCODES	LABELS	MNEMONICS	OPERANDS
0000			ORG	00H
0000	75A0FF		MOV	P2,#0FFH
0003	758920		MOV	TMOD,#20H
0006	758DFD		MOV	TH1,#-3
0009	759850		MOV	SCON,#50H
000C	D28E		SETB	TR1
000E	90002D		MOV	DPTR,#MYDATA
0011	E4	H1	CLR	A
0012	93		MOVC	A,@A+DPTR
0013	600D		JZ	RECV
0015	111A		ACALL	SEND
0017	A3		INC	DPTR
0018	80F7		SJMP	H1
001A	F599	SEND	MOV	SBUF,A
001C	3099FD	H2	JNB	TI,H2
001F	C299		CLR	TI
0021	22		RET	
0022		RECV		
0022	3098FD	H3	JNB	RI,H3
0025	C298		CLR	RI
0027	E599		MOV	A,SBUF
0029	F5A0		MOV	P2,A
002B	80F5		SJMP	RECV
002D	57454C43	MYDATA	db	"Hello Microcontrollers",0

RESULT: The given Assembly Language Program on serial communication is successfully created and verified.