VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

SCHOOL OF COMPUTER SCIENCE ENGINEERING

WINTER SEMESTER 2022-2023

LAB ASSIGNMENT - 5

Slot: L11 – L12

Class: VL2022230504038

Programme Name & Branch: B. Tech CSE

Course code & Title: BECE204P – Microprocessors and Microcontrollers Lab

Faculty Name: Venu Allapakam

EXPERIMENT – 5 : Interrupt Programming

Aim: To write an ALP in which the interrupt service routine P1.2 is complemented continuously

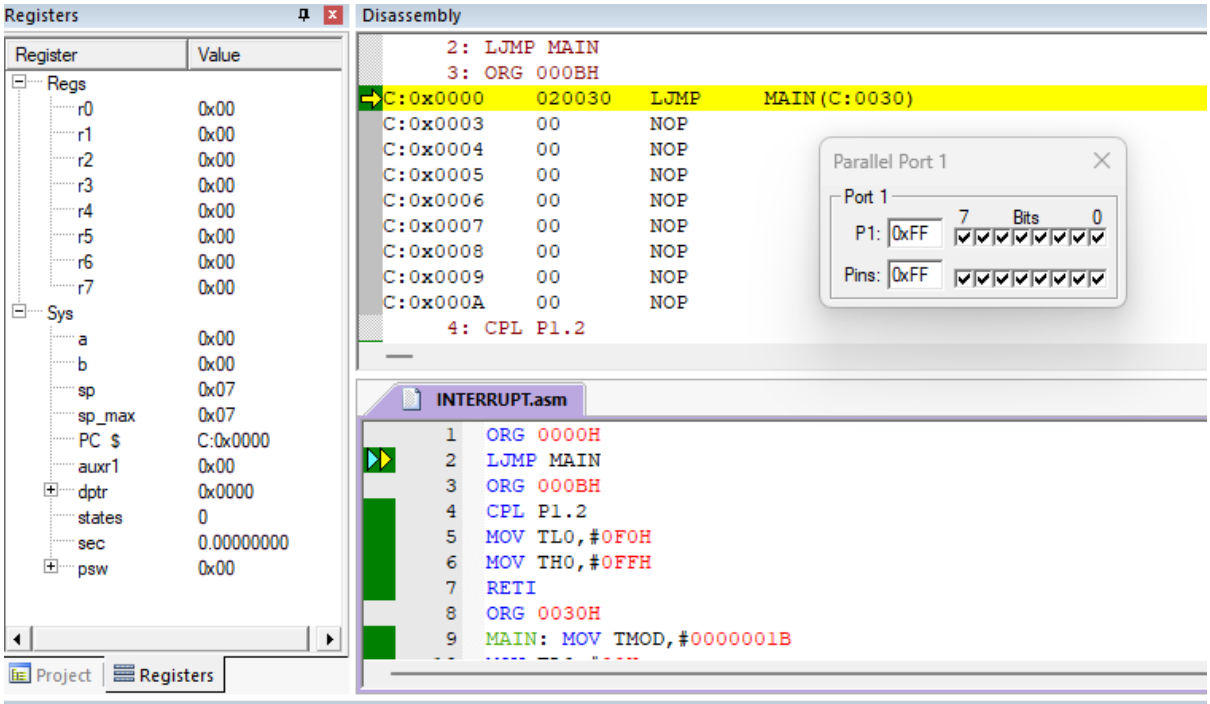
Software Requirement: Keil Software

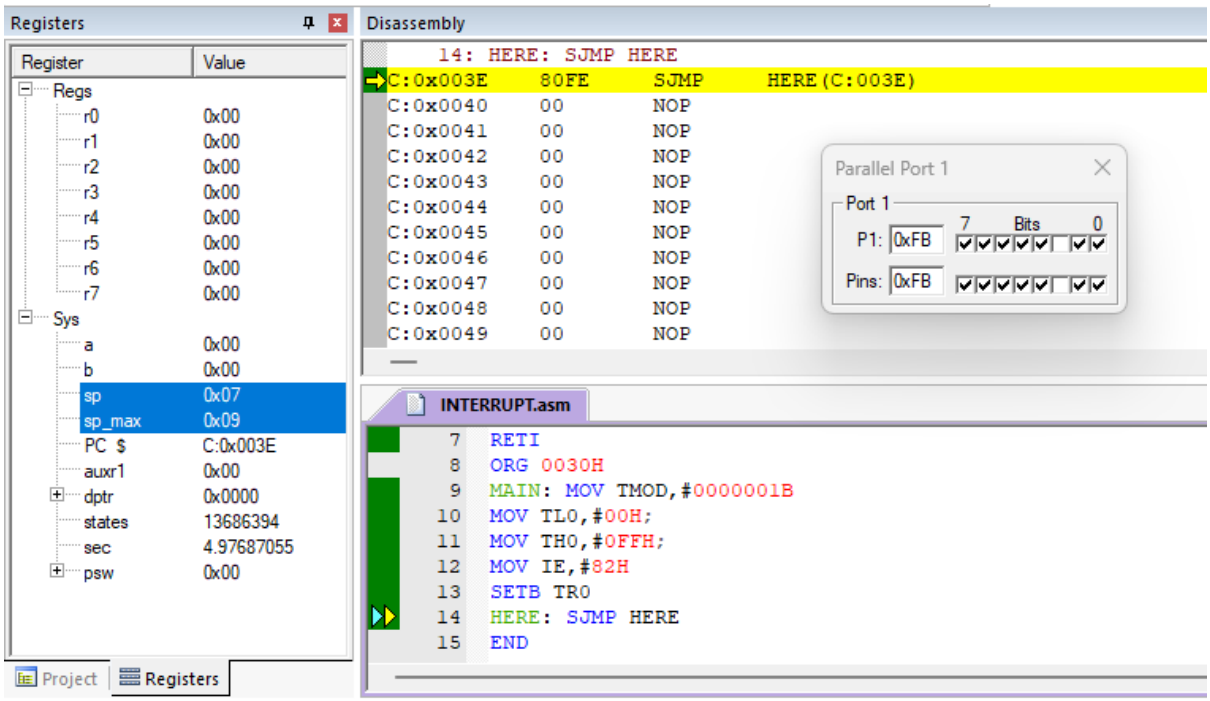
Program:

```
ORG 0000H
LJMP MAIN
ORG 000BH
CPL P1.2
MOV TL0,#0F0H
MOV TH0,#0FFH
RETI
ORG 0030H
MAIN: MOV TMOD,#0000001B
MOV TL0,#00H;
MOV TH0,#0FFH;
MOV IE,#82H
SETB TR0
HERE: SJMP HERE
END
```

Output:

Before Execution & After Execution:





Result:

Hence the program with interrupt service routine that continuously complements the bit P1.2 has been executed successfully.

Program 2:

Aim: To write an 8051 program to get data from P0 and send it to P1 continuously while an interrupt will make timer 0 toggle the P2.1 bit every 100 microseconds

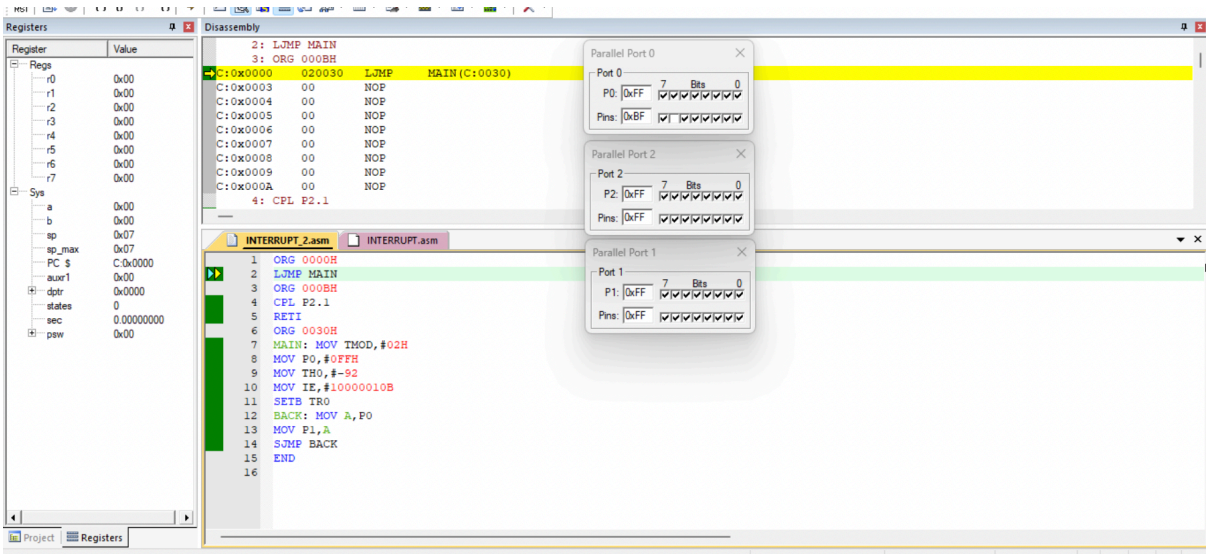
Software Requirement: Keil Software

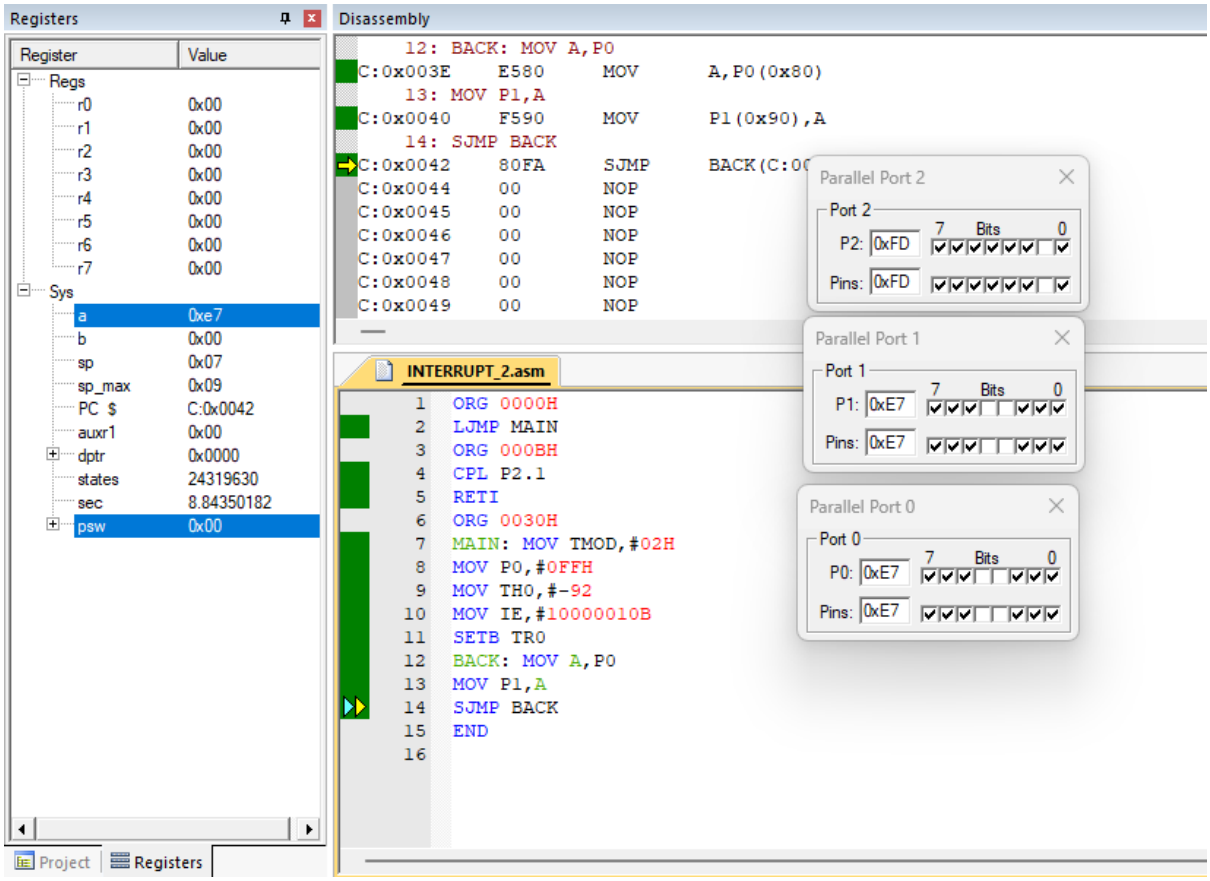
Program:

```
ORG 0000H
LJMP MAIN
ORG 000BH
CPL P2.1
RETI
ORG 0030H
MAIN: MOV TMOD,#02H
MOV P0,#0FFH
MOV TH0,#-92
MOV IE,#10000010B
SETB TR0
BACK: MOV A,P0
MOV P1,A
SJMP BACK
END
```

Output:

Before Execution &After Execution:





Result:

Hence, the data from P0 was sent to P1 continuously while the interrupt is making timer 0 toggle the P2.1 bit every 100 microseconds.

Program 3:

Aim: To write an ALP to get data from a single bit of P1.2 and send it to P1.7 continuously while a serial service interrupt routine receives data from PC and display it on P2.

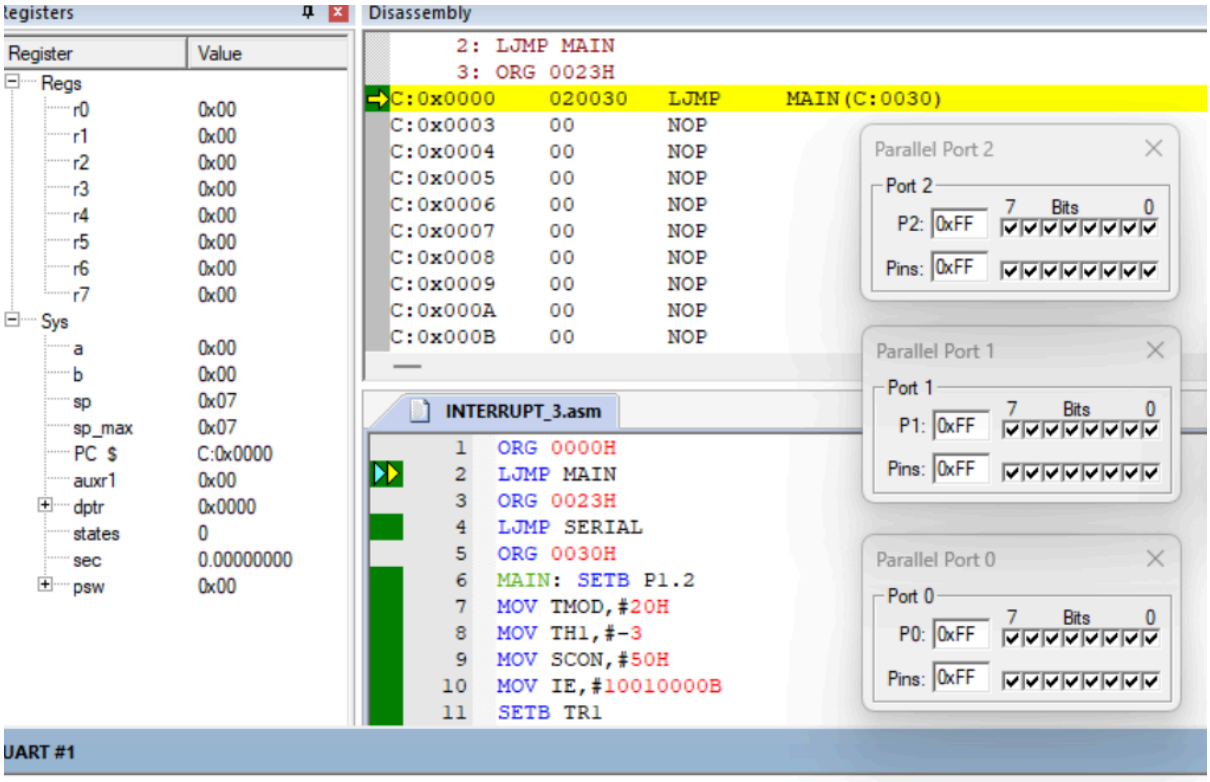
Software Requirement: Keil software

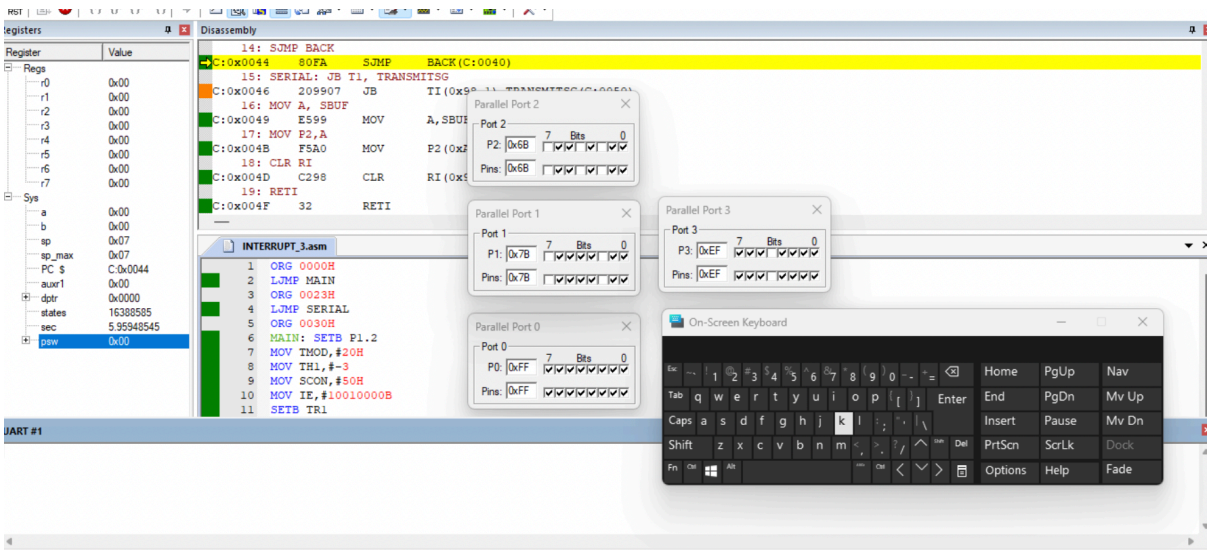
Program:

```
ORG 0000H
LJMP MAIN
ORG 0023H
LJMP SERIAL
ORG 0030H
MAIN: SETB P1.2
MOV TMOD,#20H
MOV TH1,#-3
MOV SCON,#50H
MOV IE,#10010000B
SETB TR1
BACK: MOV C,P1.2
MOV P1.7,C
SJMP BACK
SERIAL: JB T1, TRANSMITSG
MOV A, SBUF
MOV P2,A
CLR RI
RETI
TRANSMITSG: CLR TI
RETI
END
```

Output:

Before Execution &After Execution:





Result:

Hence, the data Is being continuously sent from P1.2 to P1.7 and during the interrupt, the data is sent from PC to P2.