21BDS0340

Abhinav Dinesh Srivatsa

Microprocessors and Microcontrollers Lab

Task – IV

**Question 1**

Aim:

Write an 8051-assembly language program to transfer data serially at baud rate 4800 with 8-bit data (Your Reg. Number and Name) and 1 stop bit. Observe the transmitted data in the serial window of the simulator.

Tools Required:

8051 microcontroller

Keil microcontroller software

Program

|  |  |  |  |
| --- | --- | --- | --- |
| Memory Locations | Label | Mnemonics | Comments |
|  |  | ORG 0200H |  |
|  |  | DB "Abhinav Dinesh Srivatsa (21BDS0340)" | Moving my name and registration number to space 200H |
|  |  | ORG 0000H |  |
| 0000H |  | MOV DPTR, #0200H | Move 200H to DPTR |
| 0003H |  | MOV TMOD, #20H | Timer 1 mode 2 |
| 0006H |  | MOV TH1, #-6 | Set baud rate = 4800 |
| 0009H |  | MOV SCON, #50H |  |
| 000CH |  | SETB TR1 | Start timer 1 |
| 000EH | AGAIN: | MOVC A, @A+DPTR | Move data from location 200H to A |
| 000FH |  | MOV SBUF, A | Move data from A to SBUF |
| 0011H |  | CLR A | Clear A |
| 0012H |  | INC DPTR | Increment DPTR |
| 0013H | HERE: | JNB TI, HERE | Stay here till all bits transmitted |
| 0016H |  | CLR TI | Clear transmission interrupt flag |
| 0018H |  | SJMP AGAIN | Jump to AGAIN to restart |
|  |  | END |  |

Output:

Graphical user interface, text, application

Description automatically generated

Result:

This program displays my name and registration number through serial communication transmission

**Question 2**

Aim:

Write an 8051-assembly language program to receive bytes of data serially and put them in P2. Set the baud rate at 9600, 8-bit data, and 1 stop bit.

Tools Required:

8051 microcontroller

Keil microcontroller software

Program

|  |  |  |  |
| --- | --- | --- | --- |
| Memory Locations | Label | Mnemonics | Comments |
|  |  | ORG 0000H |  |
| 0000H |  | MOV TMOD, #20H | Timer 1 mode 2 |
| 0003H |  | MOV TH1, #-3 | Set baud rate 9600 |
| 0006H |  | MOV SCON, #50H |  |
| 0009H |  | SETB TR1 | Start timer 1 |
| 000BH | HERE: | JNB RI, HERE | Stay here till reception complete |
| 000EH |  | MOV A, SBUF | Move SBUF to A |
| 0010H |  | MOV P2, A | Move A to P2 |
| 0012H |  | CLR RI | Clear reception interrupt |
| 0014H |  | SJMP HERE | Jump to HERE to start again |
|  |  | END |  |

**Question 3**

Aim:

Assume that a switch is connected to pin P1.0. Write a program to monitor the switch and perform the following:

1. If SW = 0 send the message “VIT” to the Serial #0 port
2. If SW = 1 send the message “University” to the Serial #1 port.

Tools Required:

8051 microcontroller

Keil microcontroller software

Program

|  |  |  |  |
| --- | --- | --- | --- |
| Memory Locations | Label | Mnemonics | Comments |
|  |  | ORG 0000H |  |
|  |  | SW1 EQU P1.0 | Assign switch 1 to P1.0 |
| 0000H | MAIN: | MOV TMOD, #20H | Timer 1 mode 2 |
| 0003H |  | MOV TH1, #-3 | Set baud rate to 9600 |
| 0006H |  | MOV SCON, #50H |  |
| 0009H |  | SETB TR1 | Start timer 1 |
| 000BH |  | SETB SW1 | Set bit switch 1 as input |
| 000DH | S1: | JB SW1, NEXT | Jump to NEXT if switch 1 is high |
| 0010H |  | MOV DPTR, #STR1 | Move STR1’s start index to DPTR |
| 0013H | FN: | CLR A | Clear A |
| 0014H |  | MOVC A, @A+DPTR | Move byte from DPTR to A |
| 0015H |  | JZ S1 | Jump to S1 if A = 0 |
| 0017H |  | ACALL SENDCOM | Call SENDCOM to transmit data |
| 0019H |  | INC DPTR | Increment DPTR |
| 001AH |  | SJMP FN | Jump to FN when completed transmission |
| 001CH | NEXT: | MOV DPTR, #STR2 | Move STR2’s start index to DPTR |
| 001FH | LN: | CLR A | Clear A |
| 0020H |  | MOVC A, @A+DPTR | Move byte from DPTR to A |
| 0021H |  | JZ S1 | Jump to S1 if A = 0 |
| 0023H |  | ACALL SENDCOM | Call SENDCOM to transmit data |
| 0025H |  | INC DPTR | Increment DPTR |
| 0026H |  | SJMP LN | Jump to LN when completed transmission |
| 0028H | SENDCOM: | MOV SBUF, A | Move data from A to SBUF |
| 002AH | HERE: | JNB TI, HERE | Stay here till all bits transmitted |
| 002DH |  | CLR TI | Clear transmission interrupt flag |
| 002FH |  | RET | Return to origin of call |
|  | STR1: | DB "VIT" |  |
|  | STR2: | DB "University" |  |
|  |  | END |  |

Output:

When P1.0 is high:

Graphical user interface, application

Description automatically generated

When P1.0 is low:

Graphical user interface, application

Description automatically generated

Result:

This program sends the string ‘VITUniversity’ to the serial port when the switch connected to P1.0 is low and sends the string ‘University’ when the switch P1.0 is high