ECS502U: Lab11

Question1

1. **Record the signal of one of pins P1.0, P1.1, P1.2, or P1.3 (and continue doing so for the next tasks). Which lines of the above code are responsible for these?**

The lines responsible for displaying and varying the signal read on the oscilloscope whilst channel one is connected to P1.0 (also for P1.1, P1.2 and P1.3) are **line 15** (MOV P1, #81H) making **a high signal** to be displayed at the beginning of the program**, line 23** (MOV P1, #0F0H) making **a low signal** to be displayed and **line 37** (MOV P1, #0FFH) making **a high signal** which are displayed in the oscilloscope.

2. **Use the other channel of the oscilloscope to record the signal of P3.3. Which lines of the code are responsible for these?**

The lines responsible for recording the signal of P3.3 and displaying it to the oscilloscope are **line 9** (SETB RDpin) making P3.3 to have **a high signal**, **line 38** (CLR RDpin) making P3.3 to have **a low signal** and **line 41** (SETB RDpin) making P3.3 to have a **high signal** .

3. **Press some buttons on the keypad, such that an effect on the pin of port P1 being measure becomes observable. What is the time between changing RDpin's value and the result becoming available on one of port P1's pins? (Different between Channel 1 and 2)**

The time that it takes between changing RDpin's value and the result becoming available on one of the pins of port P1 can be between 20 ns and 22 ns. Depending on the perception of the user.

Question2

1. **You may simulate the program using virtual hardware, or go straight to the development board (connecting the LED/4-button board). Address the following tasks:**

ORG 8000H

```
start:
      JB P3.2, b2
      MOV A, #'Q'
      SJMP out
b2:
      JB P3.3, b3
      MOV A, #'M'
      SJMP out
```

b3:
```
      JB P3.4, b4
      MOV A, #'U'
      SJMP out
```
b4:
```
      JB P3.5, start
      MOV A, #'L'
```

out:
```
      LCALL send
      SJMP start
```

send:
```
      CLR TI;CLEARING THE TEXT FLAG
      MOV SBUF,A; put char in STBUF (scon.1 =TI)
```

```
txloop: JNB TI,txloop (wait until char is sent)
        RET (return to the label called)
```

(a) Describe the expected behaviour of the program, assuming that the procedure \send" transmits the contents of the accumulator via the serial port.

The program checks if either one of the pins (from 2 to 5) of port P3 are set low "0", jumping to labels b2,b3, until it reaches label b4 in which case if it has not detected a low signal in the last pin, it will jump back to start. When a low signal is read, it will store a value ("Q", "M", "U" or "L") into the accumulator A. Lastly, the value stored in accumulator is transmitted to the serial port and printed out to the screen according to the buttons that were pressed and set.

b) Which higher-level language (think of C or Java) statement is modelled by the code from lines 3 to 19?

Good statements to model the assembly code in a C or Java programming could be using SWITCH statements or IF/ELSE statements.

(c) Implement the procedure \send". It should take the contents of the accumulator and transmit it via the serial port. You can safely assume that the serial port has been configured already.
send:
```
      CLR TI;CLEARING THE TEXT FLAG
      MOV SBUF,A; put char in STBUF (scon.1 =TI)
```

```
txloop: JNB TI,txloop (wait until char is sent)
        RET (return to the label called)
```

The code works first at clearing the transmit interrupt flag (CLR TI). Then, it loads the value of the accumulator to the serial buffer register (MOV SBUF, A). Lastly, a loop is created in order to wait for the 8051 finishes transfer and return to start.