TE0322

RF Microcontroller & Interfacing Labs

Laboratory Manual

DEPARTMENT OF TELECOMMUNICATION ENGINEERING

SRM UNIVERSITY

S.R.M. NAGAR, KATTANKULATHUR – 603 203.

DEPARTMENT OF TELECOMMUNICATION ENGINEERING

TE0322-RF Microcontroller & Interfacing Lab

(2010-2011)

Revision No : 2 (odd Sem)

Date :

PREPARED BY,

S. Vijayananth

Authorized by

HOD/TCE

SRM UNIVERSITY

Department of Telecommunication Engineering

TE0322-RF Microcontroller & Interfacing Lab

CONTENTS

STUDY OF 8051

A brief history of the 8051:

In 1981, Intel Corporation introduced an 8-bit micro – controller called the 8051. This micro controller had 128bytes of RAM, 4k bytes of on-clip ROM, tow timers, one serial port and four ports all on a single chip. At that time it was also referred to as a system on a chip. The 8051 is an 8-bit processor, meaning that the CPU can work on only 8-bit of date at a time. Data larger than 8-bit has to be taken into 8-bit pieces to be processed by the CPU. The 8051 has a total of four I/O ports, each 8-bit wide.

FEATURES	QUANTITY
ROM	4k bytes
RAM	128 bytes
TIMER	2
I/O Pins	32
SERIAL PORT	1
Interrupt service	6

8-bit with data type, any data layer larger then 8-bits must be broken into 8-bit frame before it is processed on address [pointing to the data to be fetched.

The most widely used registers at the 8051 are A, B, R_0, R_1, R_2, R_3, R_4, R_5, R_6,R_7, R_8, DPTR (data pointer) and PC (Program Counter). All of the above system registers and 8-digit except DPTR and the program counter. The accumulator A, is used for an arithmetic and logic instructions. To, understand the use of these registers, we will use them in various simple instructions.

PIN DESTRUCTION OF THE 8051:

Although, 8051 family members come in different packages such as DIP (Dual imines Package). QFP (Quad flat package) and LLC, they all have 40 pins that are dedicated for various functions such that I/O, $\overline{RD}, \overline{WR}$, address, data and interrupts. It must be noted that some companies provide a 20 pi version for less demanding application.

Examination figure, we note that of the 40pins, a total of 32 pins are set aside for the four ports. P_0, P_1, P_2 and P_3 where each port takes 8 pins. The rest of the pins

are designated as CLC, GND, XTAL1, XTAL2, RST, $\overline{EA}$, P $\overline{SEN}$, of these 8 pins. Six of them are used by all members of the 8051 and 8031 Family.

FUNCTION OF EACH PIN:

1. VCC – pin 40 provides supply voltage to chip. The voltage source is +5V.
2. GND –pin 20 is the ground.
3. $XTAL_1$ and $XTAL_2$ – pin 18 and pin 19. The 8051 has an on – chip oscillator but requires a external clock to run it. Most often a quartz crystal oscillator is connected to inputs $XTAL_1$ and $XTAL_2$.
4. RST – pin 9 is the RESET pin. It is an input and is active high upon applying a high pulse to this pin, the micro – controller will reset and terminate all activity. This is often referred to a power on reset.
5. $\overline{BA}$,- The 8051 family members such as the 8751, 89C51 or 35000, for the 8031. The $\overline{EA}$, pin must be connected to 9Nd to indicate that the code is stored externally. $\overline{EA}$, Which stands for “external access” is pin 31.
6. P $\overline{SEN}$ - This is an output pin P $\overline{SEN}$ stands for “program store enable”.
7. ALE-Ale (address later enable) is our o/p pin and is active high. When connecting an 8031 to external memory, port 0 provides both address and data.

EXTERNAL INTERRUPTS INTO AND INT 1:

There are only two external hardware interrupts in the 8051. INTO and INT1. They are located on pins p3.2 and p3.3 of port 3, respectively. The interrupt vector table locations 0003H and 0013H are set aside for INTO and INT1.

PROGRAMMING 8051 TIMERS:

a. Timer 0 registers:

The 16-bit register of timer 0 is accused as low byte and high byte. The low byte register called TL0 and the high byte register is as TH0

b. Timer 1 registers:

Timer 1 is also 16 bits and its 16bit register in split into 2 bytes referred to as TL1 and TH1, these register are accessible in the same way as the register of timer 0.

ARITHMETIC OPERATION

AIM:

To write the addition, subtraction, multiplication and division program of the input using 89c51 controller.

ALGORITHM:

- Start the program and save a project.
- Insert different values of a & b.
- With the operation required, the program is writer.
- File is saved with ‘file name’. asm.
- The file is debug and then run.
- Output of variables are shown
- End

Pre Lab Questions:

1. Indicate the size (8- or 16-bit) of each of the following registers.

   PC = A= B=
   R0= R1= R2= R7=

2. For Question 1, indicate the largest value (in decimal) that each register can contain.

   PC = A= B=
   R0= R1= R2= R7=

Program

```
mov a,#05h ;immediate
mov b,#10
mul ab          ;reg
mov r1,#00 ;cy
mov a,05     ;direct
mov r0,09
add a,r0   ;reg
jnc l1
inc r1 ;if cy
l1:     mov a,#40h
clr c
subb a,@r0 ;indirect
jnc l2
cpl a
inc a
l2:
sjmp $
end
```

Post Lab Questions:

1. For Question 1, indicate the largest value (in hex) that each register can contain.

   PC = A= B=
   R0= R1= R2= R7=

2. Who generates each of the following files and what is the use of each.

   .asm
   .lst
   .obj
   .abs
   .hex

Result:

Thus the above programs are executed and run with the help of 89C51microcontroller.

LOGICAL OPERATION

AIM:

To write the logical operation like ANL, ORI and XRL for the input using 89c51 controller.

ALGORITHM:

- Start the program and save a project.
- Insert different value for registers.
- With the operation required program is written.
- File is saved with “file name ”.asm
- The file is delay and run/
- Outputs are shown.
- End

Pre Lab Questions:

1. Find the value in A, the accumulator, after the following code.

```
MOV A,#45H
RR  A
RR  A
RR  A
A =         in hex
```

2. Find the value in A, the accumulator, after the following code.

```
MOV A,#45H
RL  A
RL  A
RL  A
A =         in hex
```

3. In the absence of the "SWAP A" instruction, how would you perform the operation?

PROGRAM:

```
;mul msd with 0ah and add lsd with result
mov r0,#57h
mov a,r0
anl a,#0f0h
swap a
mov b,#0ah
mul ab
mov r1,a
mov a,r0
anl a,#0fh
add a,r1
mov r2,a
end
```

Post Lab Questions:

1. Can the SWAP instruction work on any register?

2. Find the value in A after the following code.

```
CLR  A
XRL  A, #0FFH
A =          in hex
```

3. Find the value in A after the following code.

```
CLR  A
CPL  A
XRL  A, #0FFH
```

Result:

Thus the above programs are executed and run with the help of 89C51microcontroller.

SQUARE WAVE

AIM:

To generate square wave functions.

ALGORITHM:

- Start the window.
- Load TMOD
- Load FF2H into THO and TLO
- Delay sub route using timer is called.
- In the ‘DELAY’ sub-routes, time 0 is started by the SETB TRO, instructions.
- Timer 0 is stopped by the instruction ‘CLR TRO’, the delay sub routine ends and the process is repeated.

Pre Lab Questions:

1. What is the maximum frequency that can be generated using Mode 1 if the crystal frequency is 11.0592 MHz? Show your calculation.

2. What is the maximum frequency that can be generated using Mode 2 if the crystal frequency is 11.0592 MHz? Show your calculation.

3. What is the lowest frequency that can be generated using Mode 1 if the crystal frequency is 11.0592 MHz? Show your calculation.

Program

```
l1:
cpl p1.0
acall delay
sjmp l1
delay:
mov r1,#0ffh
l3:
mov r0,#0xff
l2:
djnz r0,l2
djnz r1,l3
ret
end
```

Post Lab Questions:

1. What is the lowest frequency that can be generated using Mode 1 if the crystal frequency is 11.0592 MHz? Show your calculation.

2. In mode 1, when is TFx set to high? In mode 2, when is TFx set to high?

Result:

This program is executed without any error.

SQUARE WAVE

AIM:

To generate square wave functions.

ALGORITHM:

- Start the window.
- Load TMOD
- Load FF2H into THO and TLO
- Delay sub route without timer is called.
- In the ‘DELAY’ sub-routes, time 0 is started by the SETB TRO, instructions.
- Timer 0 is stopped by the instruction ‘CLR TRO’, the delay sub routine ends and the process is repeated.

PRE – LAB :

1. What is function of dnjz.

2. what is the difference between movx and mov

3. what is the advantage of Timer 1 and Timer 0

Program:

```
L1 :    MOV R1,@0FFH;
        CPL P1.5;
        ACALL DELAY;
        SJMP L1;

DELAY :
        MOV R3,#03H;
AGAIN:
        DJNZ R3,AGAIN;
        DJNZ R1,AGAIN;
        RET;
```

Result:

This program is executed without any error.

ASCENDING AND DESCENDING ORDER

AIM:

To generate the ascending and descending order of the given input.

ALGORITHM:

- Start the window.
- Open a new project.
- Create the program for ascending or descending order.
- Compare the different not using ‘SUBB’ Command
- Delay an run the program.
- End

Pre Lab Questions:

1. Upon reset, all ports of the 8051 are configured as _____ (output, input).

2. Which ports of the 8051 have internal pull-up resistors?

3. What is the difference between RET function and END?

Program

ASCENDING ORDER

```
MOV R0,#09H
MOV R1,#07H;
MOV R2,#01H;
MOV A,R0;
SUBB A,R1;
JNC L1;
JC L2;
L1 :
        MOV A,1;
        MOV B,0;
        MOV 1,B;
        MOV 0,A;
SJMP L3;

L2 :    SJMP L3;
L3:     MOV A, R1;
        SUBB A,R2;
        JNC L4;
        JC L5;
L4:     MOV A,1;
        MOV B,2;
        MOV 1,B;
        MOV 2,A;
        SJMP L6;
L5:     SJMP L6;
L6:     MOV A,1;
        SUBB A,0;
        JNC L7;
        JC L8;
L7:     RET;
L8:     MOV A,1;
        MOV B,0;
        MOV 1,B;
        MOV 0,A;
END;
```

DESCENDING ORDER

```
MOV R0,#03H
MOV R1,#05H;
MOV R2,#07H;
MOV A,R0;
SUBB A,R1;
JC L1;
JNC L2;
L1 :
        MOV A,1;
        MOV B,0;
        MOV 1,B;
        MOV 0,A;
        SJMP L3;

L2 :    SJMP L3;
L3:     MOV A, R1;
        SUBB A,R2;
        JC L4;
        JNC L5;
L4:     MOV A,1;
        MOV B,2;
        MOV 1,B;
        MOV 2,A;
        SJMP L6;
L5:     SJMP L6;
L6:     MOV A,1;
        SUBB A,0;
        JNC L7;
        JC L8;
L7:     RET;
L8:     MOV A,1;
        MOV B,0;
        MOV 1,B;
        MOV 0,A;
END;
```

Post Lab Questions:

1. In the 8051, explain why we must write "1" to a port in order for it to be used for input.

2. Explain why we need to buffer the switches used as input in order to avoid damaging the 8051 port.

Result:

The given inputs are set in ascending & descending order in the output.

Pre Lab Questions:

1. Explain the difference between the ADD and ADDC instructions.

2. Show how to perform the subtraction: 29H - 21H.

3. True or False. "DA A" must be used for adding BCD data only.

BCD to ASCII & ASCII to BCD

AIM:

To convert the binary number format into ASCII codes and vise versa.

ALGORITHM:

- Move a value to the register 23n.
- Move R0 to value to accumulator.
- Perform the ‘ANL’ operation.
- Add the value to ‘A’ and then move the value.
- ‘SWAP’ the values.
- The binary value of 30H is added.
- Value is moved to new register
- End

Program

```
MOV R0,#023H;

MOV A,R0;
```

```
ANL A,#0FH;
ADD A,#030H;
MOV R2,A;
MOV A,R0;
ANL A, #0F0H
SWAP A;
ADD A,#030H;
MOV R3,A;
END;
```

Result:

Both the conversion is done and thus the program is executed successfully.

Post Lab Questions:

1. Can we use the "DA A" instruction to convert data such as 9CH into BCD without first performing an ADD instruction? Explain your answer.

2. Show a simple program to add 2345H and 56F8H.

3. Show a simple program to subtract 2345H from 56F8H.

Pre Lab Questions:

1. Find the value of the CY flag after the execution of the following code.

   (a)
   ```
   MOV   A,#85H
   ADD   A,#92H
   ```
   (b)
   ```
   MOV   A,#15H
   ```

```
        ADD  A,#72H
(c)     MOV          A,#0F5H
        ADD  A,#52H
(d)     MOV A,#0FF
        INC A
```

2. Upon reset, what is the value in the SP register?

3. Upon pushing data onto the stack, the SP register is _____ (decremented, incremented).

STAIRCASE PROGRAMS

AIM:

To form a stair case program.

ALGORITHM:

- Start the window.
- enable the port.
- Load TMOD.
- Load OFFH into THO and OF2H into TLO.
- Delay sub route, timer is started by the ‘SET TRO’, instructions.
- Timer 0 is stopped by the instructions ‘CLR TRO’, the delay sub route ends and the process is repeated.
- Use the INC for increment and DEC for decrement.
- Use DJNZ for the decrement and JUMP if no zero bits

Program

```
MOV P0,#00H
MO TMOD, #01H
L1 :
        MOV TLO,
        MOV THO,
        INC PO
        ACALL DELAY;
        MOV TLO,#032H
```

```
        MOV THO,# 0F5H
        INC PO
        ACALL DELAY;
        MOV TLO, #0CBH
        MOV THO,# 0F8H
        INC PO
        ACALL DELAY;
        MOV TLO, #065H
        MOV THO,# 0FCH
        INC PO;
        ACALL DELAY;
        MOV PO,#00H;
        SJMP L1;

DELAY : SETB TR0
L2 :    JNB TF0, L2
        CLR TR0;
        CLR TF0;
        RET;
        END;
```

Result:

The program is executed and result is verified.

Post Lab Questions:

1. Upon popping data from the stack, the SP register is _____ (decremented, incremented).

2. Can you change the value of the SP register? If yes, explain why you would want to do that.

3. 0The stack uses the same area of RAM as bank _____.

Pre Lab Questions:

1. Name all of the interrupts in the 8051 and their vector table addresses.

2. In timer mode 1, indicate when TF0 causes the interrupt.

3. In timer mode 2, indicate when TF0 causes the interrupt.

COMPLEMENT PROGRAM

AIM:

To create a compliment program.

ALGORITHM:

- Enable the register R and set any value.
- Move ro, to the accumulator.
- Use CPI for the compliment accumulator.
- Add 00011010b to accumulator.
- End

Program:

MOV R0, #1111 1110B

MOV A,R0;

CPL A;

```
ADD A,#00000001B;
END;
```

Result:

The program is executed successfully.

Post Lab Questions:

1. On reset, INT0 (and INT1) are _____ (edge, level) triggered.

2. On reset, which interrupt has the highest priority?

3. True or False. There is only a single interrupt for the serial data transfer.