

EMBEDDED SYSTEMS LAB

ASSIGNMENT-5

Q1) Write a C Program for the 8051 to print Fibonacci Series . Take Input from port P1 and display output on port P4 with delay of 100 milliseconds(Create Manual delay using Loop)
Ans)

```
#include<reg51.h>

void delay(){
    int i,j,k;
    for(i=0;i<1000;i++){
        for(j=0;j<100;j++){
            for(k=0;k<100;k++)
                {}
        }
    }
    return;
}
```

```
int main(){
    int i,x,res,y,z;
    x=P0;
    y=0x00;
    z=0x01;
    P3=y;
    delay();
    P3=z;
    delay();
```

```

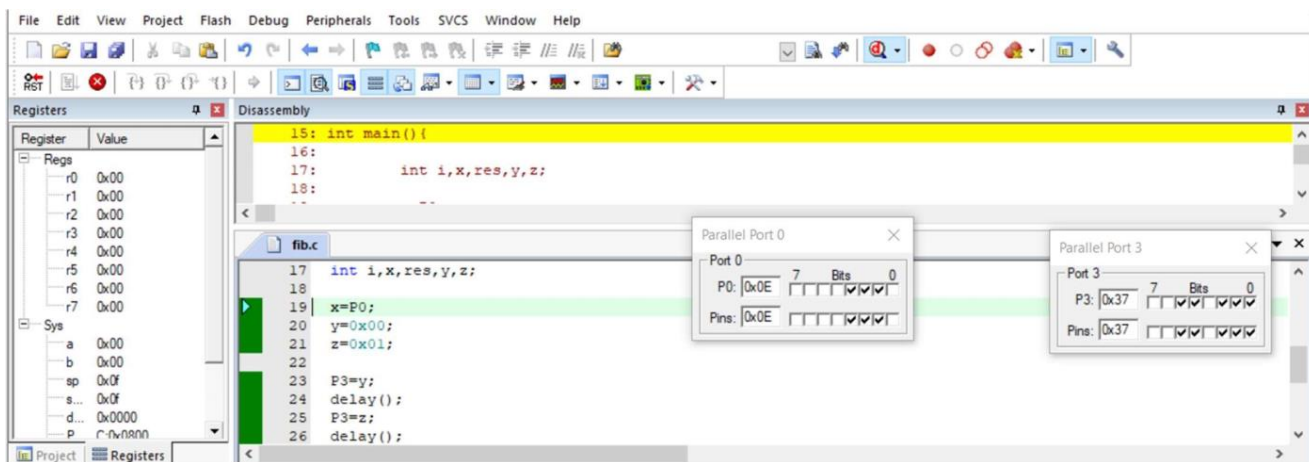
    for(i=0x02;i<=x;i++){
P3=y+z;
delay();

y=z;
z=P3;

    }

    return 0;
}

```



Q2) Write a program in C for 8051 to find the square root of a number and show the output on port.

Ans) #include<reg51.h>

```

int main(){
    int x=P0;

    int i;

    for(i=0x01;i<=P0/2;i++){
        if(i*i == x){
            P1=i;

            break;

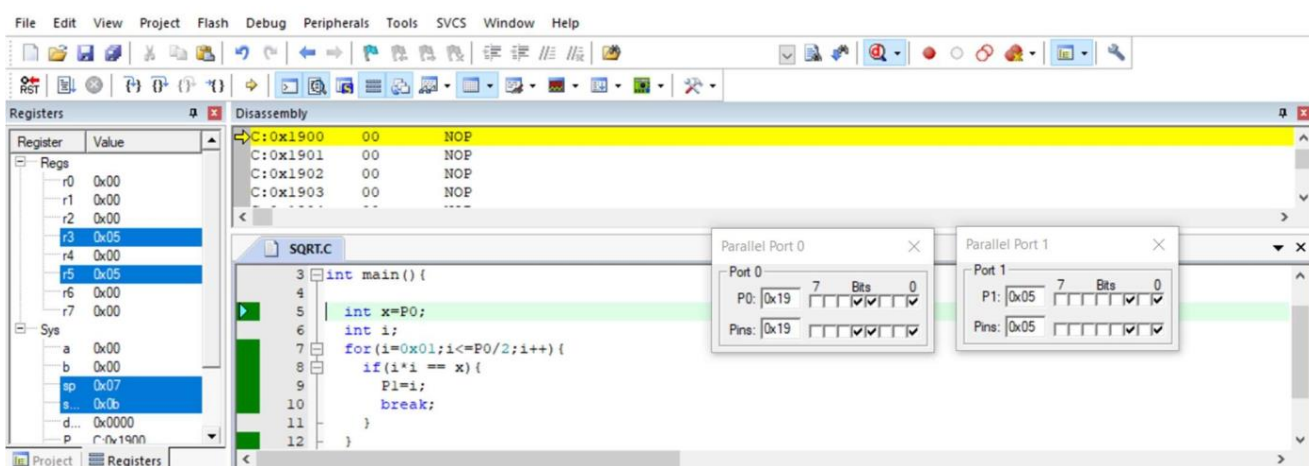
        }

    }

    return 0;
}

```

}



Q3) Write a C program for the 8051 to sort a list of n numbers. Sorted list should be displayed Port P0, P1 and P2 & P3 periodically with the delay of 150 milliseconds. (Hint 1st number on P0, 2nd number on P1, 3rd number on P2, 4th number on P3, 5th number P1 and so on) Ans)

```
#include<reg51.h>
```

```
void delay(){
```

```
int i,j,k;
```

```
for(i=0;i<150;i++){
```

```
for(j=0;j<100;j++){
```

```
for(k=0;k<100;k++){ }
```

```
}
```

```
}
```

```
}
```

```
void insertionSort(int arr[], int n)
```

```
{
```

```
int i, key, j;
```

```
for (i = 1; i < n; i++) {
```

```
key = arr[i];
```

```
j = i - 1;
```

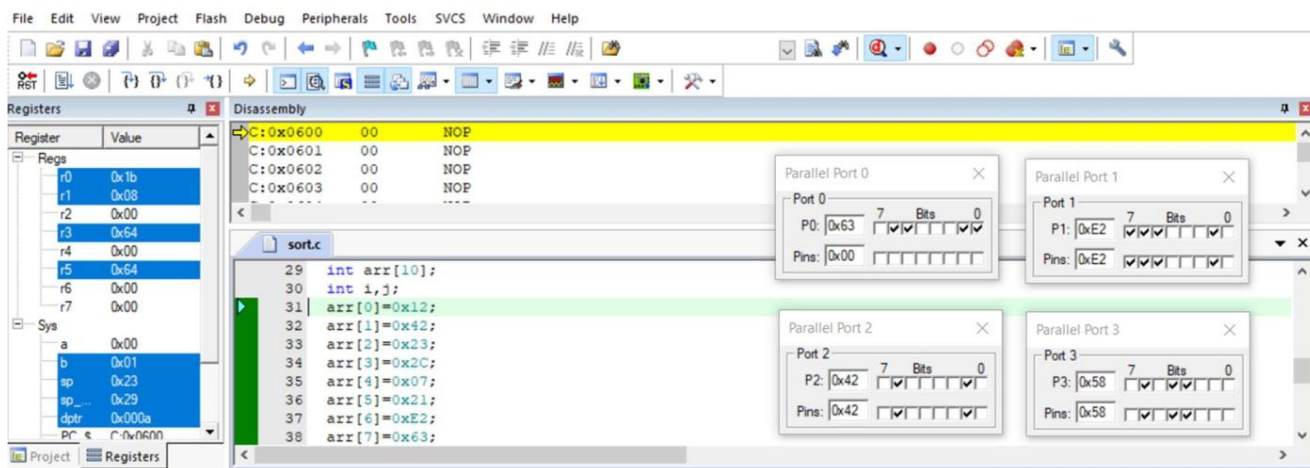
```
while (j >= 0 && arr[j] > key) {
```

```
    arr[j + 1] = arr[j];  
    j = j - 1;  
}  
    arr[j + 1] = key;  
}  
}
```

```
int main(){  
  
    int arr[10];  
    int i,j;  
    arr[0]=0x12;  
    arr[1]=0x42;  
    arr[2]=0x23;  
    arr[3]=0x2C;  
    arr[4]=0x07;  
    arr[5]=0x21;  
    arr[6]=0xE2;  
    arr[7]=0x63;  
    arr[8]=0x40;  
    arr[9]=0x58;  
    insertionSort(arr,10);  
    for(i=0x00;i<0x0A;i++){  
        if(i%4==0x00) P0=arr[i];  
        else if(i%4==0x01) P1=arr[i];  
        else if(i%4==0x02) P2=arr[i];  
        else P3=arr[i];  
        delay();  
    }  
}
```

return 0;

}



□ To be Done using EdSim51 simulator in 8051

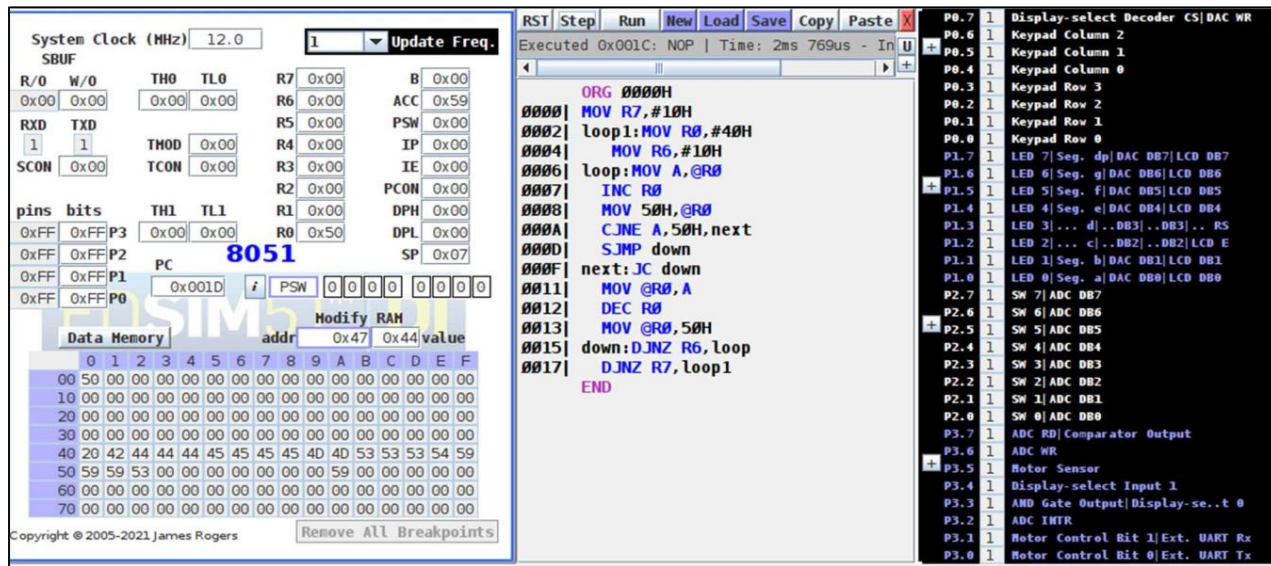
Q.1 Write an assembly language program to sort the string “embedded systems” in ascending order.

Ans:

;ALP to sort string "EMBEDDED SYSTEMS"

;The numbers in the location starting ;from
40H are the ascii values of ;characters in the
string.

```
ORG 0000H MOV
R7,#10H loop1:MOV
R0,#40H  MOV
R6,#10H
loop:MOV A,@R0
  INC R0
  MOV 50H,@R0
  CJNE A,50H,next
  SJMP down next:JC
down
  MOV @R0,A
  DEC R0  MOV
  @R0,50H
down:DJNZ R6,loop
  DJNZ R7,loop1
END
```



Q.2 You are required to count the number of times “d” occurred in the string of “embedded systems” and display it at a memory location using indirect addressing mode.

Ans:

```
;ALP to count how many times 'D'
;occurs in "EMBEDDED SYSTEMS"
```

```
MOV R0,#40H; array starting address
MOV R1,#10H ; array size
MOV R2,#00H; counter variable
MOV R3,#44H
LOOP: MOV A,@R0;
INC R0
CLR C
SUBB A,R3
JNZ SKIP
INC R2
SKIP: DJNZ R1,LOOP
```

EdSim51DI - Version 2.1.32 | Q1.asm

System Clock (MHz) 12.0

1 Update Freq.

R7 0x00 B 0x00

R6 0x00 ACC 0x0F

R5 0x00 PSW 0x40

R4 0x00 IP 0x00

R3 0x44 IE 0x00

R2 0x03 PCON 0x00

R1 0x02 DPH 0x00

R0 0x4E DPL 0x00

SP 0x07

TH0 0x00 TL0 0x00

THOD 0x00 TCON 0x00

TH1 0x00 TL1 0x00

PC 0x0008

PSW 0 1 0 0 0 0 0 0

8051

Modify RAM

addr 0x47 0x44 value

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	4E	02	03	44	00	00	00	00	00	00	00	00	00	00	00	00
10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
40	20	42	44	44	44	45	45	45	45	40	40	53	53	53	54	59
50	59	59	53	00	00	00	00	00	00	00	59	00	00	00	00	00
60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Copyright © 2005-2021 James Rogers

Remove All Breakpoints

RST Step Run New Load Save Copy Paste

Time: 119us - Instructions: 91

U

+

+

;

ALP to count how many times 'D'

occurs in "EMBEDDED SYSTEMS"

0000 MOV R0,#40H; array starting address

0002 MOV R1,#10H ; array size

0004 MOV R2,#00H; counter variable

0006 MOV R3,#44H

0008 LOOP: MOV A,@R0;

0009 INC R0

000A CLR C

000B SUBB A,R3

000C JNZ SKIP

000E INC R2

000F SKIP: DJNZ R1,LOOP

P0.7 1 Display-select Decoder CS|DAC WR

P0.6 1 Keypad Column 2

P0.5 1 Keypad Column 1

P0.4 1 Keypad Column 0

P0.3 1 Keypad Row 3

P0.2 1 Keypad Row 2

P0.1 1 Keypad Row 1

P0.0 1 Keypad Row 0

P1.7 1 LED 7|Seg. dp|DAC DB7|LCD DB7

P1.6 1 LED 6|Seg. g|DAC DB6|LCD DB6

P1.5 1 LED 5|Seg. f|DAC DB5|LCD DB5

P1.4 1 LED 4|Seg. e|DAC DB4|LCD DB4

P1.3 1 LED 3|... d|..DB3|..DB3|.. RS

P1.2 1 LED 2|... c|..DB2|..DB2|LCD E

P1.1 1 LED 1|Seg. b|DAC DB1|LCD DB1

P1.0 1 LED 0|Seg. a|DAC DB0|LCD DB0

P2.7 1 SW 7|ADC DB7

P2.6 1 SW 6|ADC DB6

P2.5 1 SW 5|ADC DB5

P2.4 1 SW 4|ADC DB4

P2.3 1 SW 3|ADC DB3

P2.2 1 SW 2|ADC DB2

P2.1 1 SW 1|ADC DB1

P2.0 1 SW 0|ADC DB0

P3.7 1 ADC RD|Comparator Output

P3.6 1 ADC WR

P3.5 1 Motor Sensor

P3.4 1 Display-select Input 1

P3.3 1 AND Gate Output|Display-se..t 0

P3.2 1 ADC INTR

P3.1 1 Motor Control Bit 1|Ext. UART Rx

P3.0 1 Motor Control Bit 0|Ext. UART Tx

DI i LD

1 2 3

AND Gate Disabled

U No Parity 8-bit UART @ 4800 Baud