**Embedded Systems Lab**

**Assignment 6**

**Name** – Neelangshu Roy

**Reg No** – 20214060

**Group** – C1

TO BE DONE IN KEIL :

Q1. Write a C program in Keil for the 8051 to display a pattern of AA and 55h on port P1 with the delay of 100 milliseconds.

Code –

#include<reg51.h>

void createDelay(int t)

{

int i,j;

t\*=10;

for(i=0;i<t;i++)

for(j=0;j<1275;j++);

}

void main(void)

{

int turn=0;

P1=0x00;

while(1)

{

if(!turn)P1=0xaa;

else P1=0x55;

createDelay(100);

turn^=1;

}

}

Q2. Write an 8051 C program in Keil to get a byte of data from P0. If it is less than 100, send it to P1; otherwise, send it to P2.

Code –

#include<reg51.h>

void main(void)

{

unsigned int n;

P0=0x00;

while(1)

{

n=P0;

if(n<100)

{

P1=n;P2=0x00;

}

else

{

P2=n;P1=0x00;

}

}

}

Q3. Write an 8051 C program in Keil to convert 11111101 (FD hex) to decimal and display the digits on P0, P1 and P2.

Code –

#include<reg51.h>

void main(void)

{

unsigned int n;

P3=0x00;

while(1)

{

n=P3; // input is taken from P3, so 0xFD is fed to port P3

P2=n%10; // contains least significant digit

n/=10;

P1=n%10;

n/=10;

P0=n; // contains most significant digit

}

}

TO BE DONE IN EdSim51 SIMULATOR :

Q1. Write a Program to check whether a number is palindrome or not. If palindrome store FFh in accumulator.

Code –

mov a, 40h ; number is stored at 40h

mov b, #10

div ab

mov r0, b ; unit's column

mov b, #10

div ab

mov r1, b ; ten's column

mov b, #10

div ab

mov r2, b ; hundred's column

mov a, #00

clr psw.7 ; clearing carry before the

subb a, r2 ; subb is VERY IMPORTANT !!

jc notthis

clr psw.7

subb a, r1

jc notthis

jmp ispalin ; single digit number

notthis: mov a, #00

clr psw.7

subb a, r2

jnc tilltens

mov a, r2

clr psw.7

subb a, r0

jz ispalin

jmp isnotpalin

tilltens: mov a, r1

clr psw.7

subb a, r0

jz ispalin

isnotpalin: mov a, #00

jmp isover

ispalin: mov a, #0ffh

isover: nop

end

Q2. Write an assembly language program to compute prime factors of a number.

Code –

; number is in 30h

; start storing prime divisors from 40h

mov r0, #40h

mov a, 30h

clr psw.7

subb a, #01

jc finish

mov a, 30h

clr psw.7

subb a, #01

jz finish

mov a, 30h

subb a, #04

jc itsover

mov r2, #02

loop1: mov a, r2

clr psw.7

subb a, 30h

jz itsover

mov a, r2

mov b, a

mov a, 30h

div ab

mov a, b

clr psw.7

subb a, #01

jnc gonext

mov a, r2

mov r7, a

call checkifprime

mov a, r6

clr psw.7

subb a, #01

jc go\_on

mov a, r2

mov @r0, a

inc r0

go\_on: nop

gonext: inc r2

jmp loop1

jmp itsover

checkifprime: nop ;number is in r7

mov a, r7

clr psw.7

mov r6, #00

subb a, #01

jz return

inc r6

clr psw.7

subb a, #01

jz return

mov r5, #02

loop: mov a, r5

clr psw.7

subb a, r7

jz return

mov a, r7

mov b, r5

div ab

mov a, b

clr psw.7

subb a, #01

jc isnotprime

inc r5

jmp loop

isnotprime: dec r6

return: ret

itsover: mov a, #40h

clr psw.7

subb a, r0

jnz finish

mov @r0, 30h

inc r0

finish: mov @r0, #0ffh

end ; after the end, FFh is shown

Q3. Write an assembly language program to print binary pattern on the Port 1.

Code –

mov p0, #80h ; enable display

anl p3, #0e7h ; set it to 4th 7-segment display

mov r0, #0c0h ; for 0

mov r1, #0f9h ; for 1

mov a, #0

loop: clr psw.7

subb a, #01

jz second

mov p1, r0

mov a, #1

jmp next ; alternate between 0 and 1

second: mov p1, r1

next: jmp loop

; 0 and 1 are displayed(in red) alternatively on the 4th 7-segment display on the simulator