ARM

Course PROJECT

Group 1

Under the guidance of **Sahana P.** ma’am.

Members:

1.Guruprasad Kamat

2. Fardin Khan

3. Shravan Hoskoti

4. Rochan Bachari

**Office security system.**

The Buzzer goes off signifying the incorrect password and an intrusion.

The DC motor rotates signifying the gate opening.

The password is incorrect.

The password is correct

Enter the password

External interrupt to let a person in though the password is incorrect.

(Emergency Case)

Methodology.

* The password is taken through an external peripheral (keyboard) using UART.
* Once the password is entered, it is received in terms of ascii values which are compared to check the password.
* If the password happens to be correct, the LCD function is called which displays “HELLO” and the UART display(monitor) displays “You are welcome” Then the stepper motor function is called which rotates for 5 seconds indicating the opening of gate.
* If the password happens to be wrong, the UART display (monitor) displays “No entry” as a result the buzzer function is called which buzzer goes off indicating incorrect password and intrusion.
* In case of an emergency, when the entered password is incorrect but the person needs to be let in, external interrupt comes into play and the LCD and stepper motor functions are called to let the person in.

CODE:

Main function:

#include<lpc21xx.h>

#include "buz.h"

#include "stepp.h"

#include "checkk.h"

#include "lcdd.h"

#include "seriall.h"

#include "extint.h"

#include "timer.h"

unsigned char mg;

int main()

{

int x=0,i;

serial();

PINSEL1=0x00000001;

EXTMODE=0x00000001;

EXTPOLAR=0X00000000;

VICVectAddr0=(unsigned long)extint0\_ISR;

VICVectCntl0=0x2014;

VICIntEnable=0x00004000;

for(i=0;i<3;i++)

{

while(!(U0LSR & 0x01));

mg=U0RBR;

U0THR=mg;

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

if(U0THR==98)

{

x=U0THR;

delay();

delay();

delay();

delay();

delay();

delay();

delay();

}

else if(U0THR==118)

{

x=x+U0THR;

delay();

delay();

delay();

delay();

delay();

delay();

delay();

}

else if(U0THR==97)

{

x=x+U0THR;

check(x);

delay();

delay();

delay();

delay();

delay();

delay();

delay();

}

else if(U0THR==99 || U0THR<=100 || U0THR>=101 || U0THR>=102 || U0THR>=103 || U0THR>=104 || U0THR>=105 || U0THR>=106 || U0THR>=107 || U0THR>=108 || U0THR>=109 || U0THR>=110 || U0THR>=111 || U0THR>=112 || U0THR>=113 || U0THR>=114 || U0THR>=115 || U0THR>=116 || U0THR>=117 || U0THR>=119 || U0THR>=120 || U0THR>=121 || U0THR>=122)

{

check1();

delay();

delay();

delay();

delay();

delay();

delay();

delay();

}

}

}

Buzzer function:

#include<lpc21xx.h>

void delay(void);

void buzzer()

{

PINSEL0=0x00000000;

IODIR0=0x00000200; ///po.9 made output

IOCLR0=0x00000200;

while(1)

{

IOSET0=0x00000200;

delay();

delay();

delay();

IOCLR0=0x00000200;

}

}

Stepper motor function:

#include<lpc21xx.h>

void delay(void);

void InitTimer0(void);

void step(void)

{

int x=3;

PINSEL2=0x0010000;

IO0DIR=0xf0000000;

IO1DIR=0x00000000;

//x=IOPIN1;

//x=x & 0x00f00000;

PINSEL0 = 0X00000000 ; // Configure P0.16-p0.23 as GPIO

IODIR0=0X00FFF000;

InitTimer0(); // initialise timer0 -

T0TCR = 0x01; // start timer

while(x!=0)

{ //if(x==0x00100000)

//{

while(!( T0IR==0x01)); // wait for overflow

{ T0IR=0x01;

IO0SET=0x10000000;

delay();

IO0CLR=0x10000000;

delay();

IO0SET=0x20000000;

delay();

IO0CLR=0x20000000;

delay();

IO0SET=0x40000000;

delay();

IO0CLR=0x40000000;

delay();

IO0SET=0x80000000;

delay();

IO0CLR=0x80000000;

delay();

//}

//else

//{

delay(); delay(); delay(); delay(); delay();

IO0SET=0x80000000;

delay();

IO0CLR=0x80000000;

delay();

IO0SET=0x40000000;

delay();

IO0CLR=0x40000000;

delay();

IO0SET=0x20000000;

delay();

IO0CLR=0x20000000;

delay();

IO0SET=0x10000000;

delay();

IO0CLR=0x10000000;

delay();

//}

}x=x-1;

}

}

UART display function:

#include<lpc21xx.h>

void delay(void);

void buzzer(void);

void lcd(void);

void step(void);

void check1()

{

unsigned char msg1[]={" No Entry!"};

unsigned int i;

for(i=0;i<12;i++)

{

while(!(U0LSR & 0x20));

U0THR=msg1[i];

}

buzzer(); //calling buzzer

}

void check(int x)

{

unsigned char msg[]={" You Are Welcome!"};

unsigned int i;

if(x==313)

{

for(i=0;i<17;i++)

{

while(!(U0LSR & 0x20));

U0THR=msg[i];

}

delay(); delay(); delay();

step(); //calling stepper

delay(); delay();// calling lcd

lcd();// calling lcd

}

}

LCD Function:

#include<lpc21xx.h>

void delay(void);

void data(unsigned int);

void command(unsigned int);

void lcd(void)

{

unsigned char message[]= {"hello"};

unsigned int c[]={0x30,0x30,0x20,0x20,0x28,0x01,0x06,0x0e,0x89};

unsigned int i,j;

PINSEL0=0X00000005;

IO0DIR=0x000000fc;

for(i=0;i<9;i++)

{

command(c[i]);

delay();

}

while(1)

{

command(0x89);

delay();

for(j=0;j<5;j++)

{

data(message[j]);

delay();

}

delay();

command(0x10);

delay();

}

}

void command(unsigned int x)

{

unsigned int y;

y=x;

y=y & 0x0f;

IO0CLR=0x000000fc;

IO0CLR=0x00000004;

IO0SET=y;

IO0SET=0x00000008;

delay();

delay();

delay();

delay();

IO0CLR=0x00000008;

y=x;

y=y & 0x0f;

y=y<<4;

IO0CLR=0x00000004;

IO0SET=y;

IO0SET=0x00000008;

delay();

delay();

delay();

delay();

IO0CLR=0x00000008;

}

void data(unsigned int a)

{

unsigned int b;

b=a;

b=b & 0x0f;

IO0CLR=0x000000fc;

IO0SET=0x00000004;

IO0SET=b;

IO0SET=0x00000008;

delay();

delay();

delay(); delay(); delay(); delay();

b=a;

b=b& 0x0f;

b=b<<4;

IO0CLR=0x000000fc;

IO0SET=0x00000004;

IO0SET=b;

IO0SET=0x00000008;

delay(); delay(); delay(); delay(); delay(); delay(); delay();

IO0CLR=0x00000008;

}

Serial function:

#include<lpc21xx.h>

void delay()

{

unsigned int i;

for(i=0;i<10000;i++);

}

void serial()

{

PINSEL0=0X00000005;

U0LCR= 0X83;

U0DLL=0X61;

U0LCR=0X03;

}

External Interrupt function:

#include<lpc21xx.h>

#include<stdio.h>

void delay(void);

void lcd(void);

void step(void);

void extint0\_ISR(void);

void extint0\_ISR(void)

{

PINSEL1=0x00000000;

step();

lcd();

delay();

delay();

EXTINT=0x00000001;

VICVectAddr=0;

}

Timer function:

#include <LPC21xx.H>

#define DESIRED\_COUNT 1000 // for 1sec

#define PRESCALER 11999

void InitTimer0(void);

void InitTimer0(void)

{

T0PR=PRESCALER;

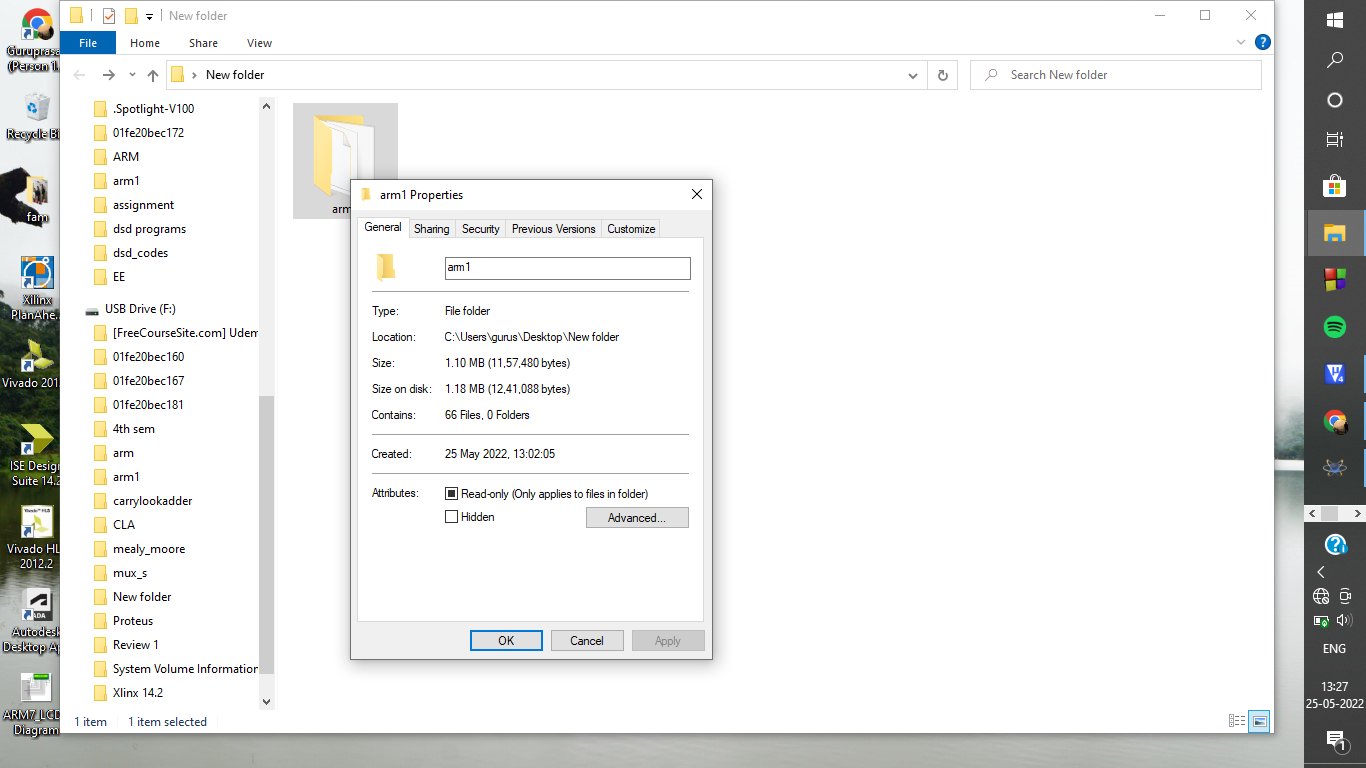
T0MR0=DESIRED\_COUNT; //interrupt every 1 sec for interval = 1000

T0MCR=3; //interrupt and reset when counter=match

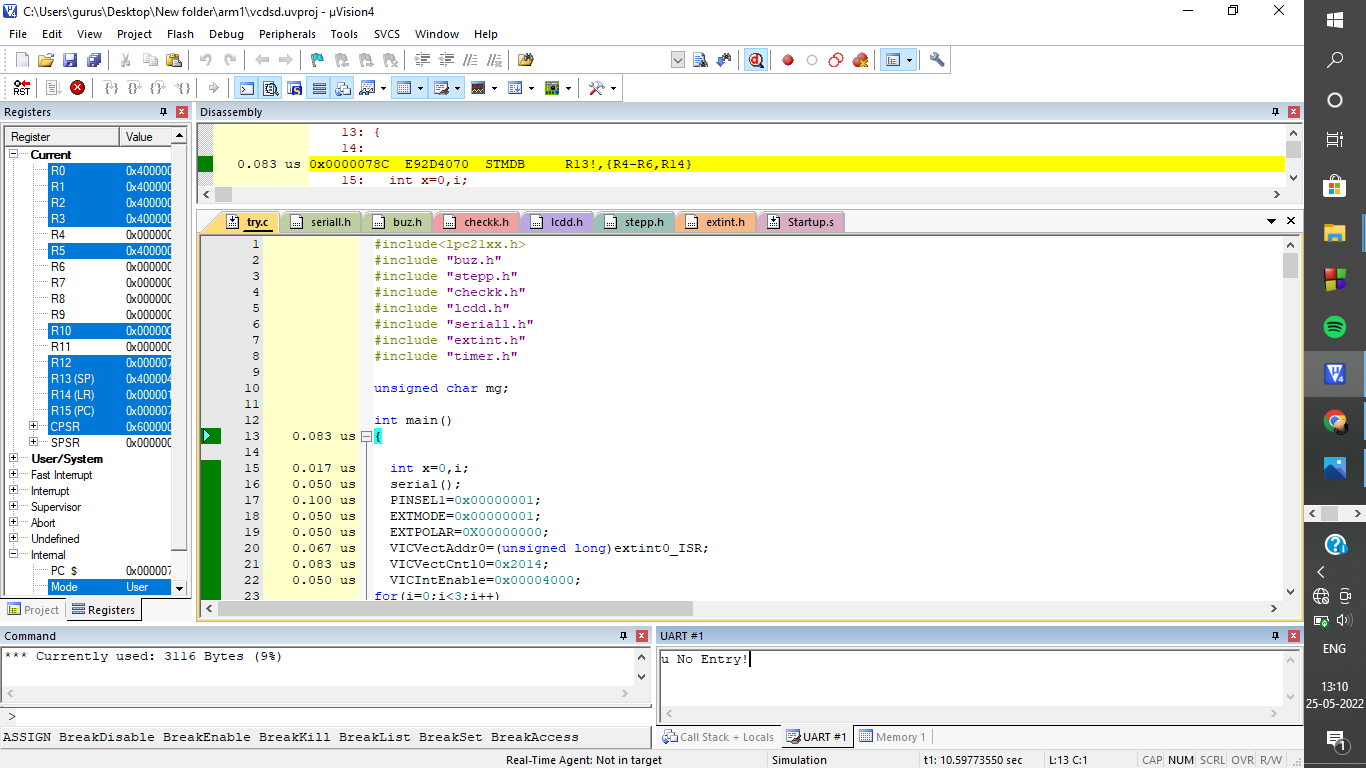
T0TCR=2; // reset timer

}

Memory:



Time:



Simulation:

