KARNATAK LAW SOCIETY’S

GOGTE INSTITUTE OF TECHNOLOGY

UDYAMBAG, BELAGAVI-590008

(An Autonomous Institution under Visvesvaraya Technological University, Belagavi)

**(APPROVED BY AICTE, NEW DELHI)**

Department of Electronics and Communication Engineering



*Course Project Report on*

**Cipher Machine**

**Using LPC2148**

*Submitted in partial fulfillment of the requirement for the award of the degree of*

**Bachelor of Engineering**

**in**

**Advanced processors**

*Submitted by*

Abhay kagwad

Hrushikesh Kitwadkar

Kaustubh Halyal

**Guide**

Prof. Jyoti BR

**2019 – 2020**

**DECLARATION BY THE STUDENT**

We, ***Abhay Kagwad, Hrushikesh Kitwadkar, Kaustubh Halyal***, hereby declare that the course project report entitled **Enigma Cipher Machine** submitted by us to KLS Gogte Institute of Technology, Belagavi, in partial fulfillment of the Degree of **Bachelor of Engineering** in **Electronics and Communication Engineering** is a record of the project carried out in **Advanced processors**. This report is for the academic purpose.

We further declare that the course project report has not been submitted and will not be submitted, either in part or full, to any other institution and University for the award of any diploma or degree.

Name of the students: Abhay Kagwad(2GI17EC002)

Hrushikesh Kitwadkar(2GI17EC039)

Kaustubh Halyal(2GI17EC043)

Signature :

Place: K L S GOGTE INSTITUTE OF TECHNOLOGY, BELAGAVI

Date:

**Enigma Cipher Machine Implementation and Display on LCD**

**ABSTRACT:**

The project is realizing Encryption and Decryption algorithm with UART interfacing and displaying original message on the LCD. We need a input key to encrypt the plain text and same key is used to decrypt the cipher text in order to get original plain text .

**OBJECTIVE:**

Verifying encryption and Decryption algorithm with UART interfacing and display the message on the LCD.

**ABOUT:**

The project includes UART along with the LCD interfacing of LPC2148. The UART is 16 byte FIFO which receives and transmits the characters serially. The UART0 has transmitter and receiver pins as **P0.0** and **P0.1** respectively. These pins are set their respective operations using **PINSEL0** register with a value of **5**. Each time one character is received and transmitted by UART . The plain text and Key is received from the user to the UART meanwhile the same characters is displayed on the LCD.

The UART0 receives the character through U0RBR and transmits the character through U0THR register .LCD is used to display a text in 2\*16 LCD module using 4 data lines only. Some delay is occurring when a single command or data is executed .

The key with which we encrypt the text ,the same key should be used to decrypt the cipher text otherwise it display **“ Invalid key”** on LCD. The UART and LCD will work parallelaly in this process .

**CODE PATTERN:**

The code is very simple, the plain text and keys are received and transmitted through UART and is made with through a Header file created as **uartfunctions.h** and the characters is displayed on the LCD through a Header file created as **lcdfunctions.h.**

Some delay is included as there are some delay occurs to execute a command or data instruction. The key is compared and then respective operation is done.

**BLOCK DIAGRAM**

Encryption Algorithm

Plain Plain

Decryption Algorithm

Text Text

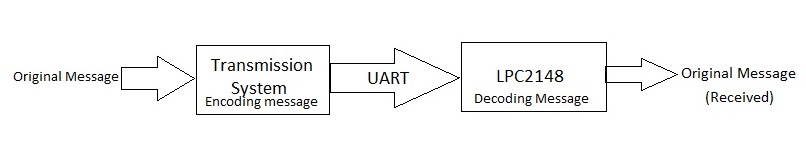
Cipher text over a protected

channel

**Private Key Private Key**

The figure represents a simple encryption and decryption model of a plain text. The plain text is given as input to an encryption algorithm which encrypts the text along with key. Cipher text obtained will be transmitted over a channel. It will be then given as input to decryption algorithm which decrypts the cipher text giving back plain text.

**Circuit Diagram:**



The original message (to be transmitted) is fed to the transmission system to encode the signal using Key. The transmission system used here is the desktop Computer.

The key is added which each character of message in transmitter. The encoded message is then transmitted through UART. The receiver used is LPC2148.The received message is then decoded using the key used earlier in transmission system.

The decoded message is then displayed on LCD screen.

**CODE:**

#include<LPC21XX.h>

#include "uartfunctions.h"

#include "lcdfunctions.h"

int main()

{

char plaintxt[10],ciphertxt[10],dplaintxt[10];

int i;

int count=0;

char key[2]=" ";

uartinit();

lcd\_init();

clearDisplay();

uartwrite("\nEnter message :");

delay(50000);

for(i=0;i<10;i++)

{

plaintxt[i]=uartread();

uartwritech(plaintxt[i]);

}

uartwrite("\nEnter the key :");

key[0]=uartread();

uartwritech(key[0]);

uartwrite("\nCipher text is :");

for(i=0;i<10;i++)

{

ciphertxt[i]=plaintxt[i]+key[0];

}

uartwrite(ciphertxt);

uartwrite("\n");

//DECRYPTION

uartwrite("\nDecryption");

delay(500000);

uartwrite("\nEnter the key value :");

key[1]=uartread();

uartwritech(key[1]);

delay(50000);

if(!(key[0]==key[1])&&count<3)

{

uartwrite("\nEntered key is incorrect\n");

count++;

}

if(count==3)

{

uartwrite("\nMaximum attempts to enter key are reached\n\n");

return(0);

}

key[0]=' ';

for(i=0;i<10;i++)

{

dplaintxt[i]=ciphertxt[i]-key[1];

}

uartwrite("\nOriginal message:");

uartwrite(dplaintxt);

uartwrite("\n\n\n");

LCD("Original msg",dplaintxt);

delay(50000000);

return(0);

}

**Header Files.**

**UART**

#include<LPC21xx.h>

void uartinit()

{

PINSEL0=0x5;

U0LCR=0x83;

U0DLL=97;

U0DLM=00;

U0LCR=0x03;

}

void uartwrite(char ch[30])

{

int i;

for(i=0;i<30;i++)

{

while((U0LSR & 0x20)==0);

if(ch[i]=='\0')

break;

U0THR=ch[i];

}

}

void uartwritech(char c)

{

while((U0LSR & 0x20)==0);

U0THR=c;

}

char uartread()

{

while((U0LSR & 0x1)==0);

return(U0RBR);

}

**LCD Header**

#include<LPC21xx.h>

#include<string.h>

void lcd\_init(void);

void writeCommand(void);

void clearDisplay(void);

void delay(unsigned int);

void writeData(void);

void lcdData(void);

void lcdCommand(void);

void LCD(char disp[30],char disp1[30]);

void LCDhead(char l[]);

unsigned char temp1;

unsigned long int temp,r=0;

char \*ptr;

char disp2[20];

void LCDhead(char l[20])

{

strcpy(disp2,l);

return;

}

void LCD(char disp[30],char disp1[30])

{

IO0DIR=0x000000FC;

IO0PIN=0x00000000;

lcd\_init();

delay(3200);

clearDisplay();

delay(500);

temp1=0x80;

lcdCommand();

delay(645);

ptr=disp;

while(\*ptr!='\0')

{

temp1=\*ptr;

delay(3200);

lcdData();

ptr++;

delay(3200);

}

temp1=0xC0;

lcdCommand();

delay(800);

ptr=disp1;

while(\*ptr!='\0')

{

temp1=\*ptr;

delay(3200);

lcdData();

ptr++;

delay(3200);

}

return;

}

void lcd\_init(void){

temp=0x30;

writeCommand();

delay(3200);

temp=0x30;

writeCommand();

delay(3200);

temp=0x30;

writeCommand();

delay(3200);

temp=0x20;

writeCommand();

delay(3200);

temp=0x28;

lcdCommand();

delay(3200);

//load a command for display on cursor and blinking off

temp1=0x0C;

lcdCommand();

delay(800);

//command for cursor increment after data dump

temp1=0x06;

lcdCommand();

delay(800);

temp1=0x80;

lcdCommand();

delay(800);

}

void lcdCommand(void){

temp=temp1&0xF0;

writeCommand();

temp=temp1&0x0F;

temp=temp<<4;

writeCommand();

delay(500);

}

void writeCommand(void){

IO0CLR=0x000000FC;

IO0SET=temp;

IO0CLR=0x00000004;

IO0SET=0x00000008;

delay(10);

IO0CLR=0x00000008;

}

void writeData(void){

IO0CLR=0x000000FC;

IO0SET=temp;

IO0SET=0x000000004;

IO0SET=0x000000008;

delay(10);

IO0CLR=0x000000008;

}

//data o/p routine that o/p high nibble first first then low nibble next

void lcdData(){

temp=temp1&0xF0;

temp=temp;

writeData();

temp=temp1&0x0F;

temp=temp<<4;

writeData();

delay(100);

}

void clearDisplay(void){

temp1=0x01;

lcdCommand();

delay(500);

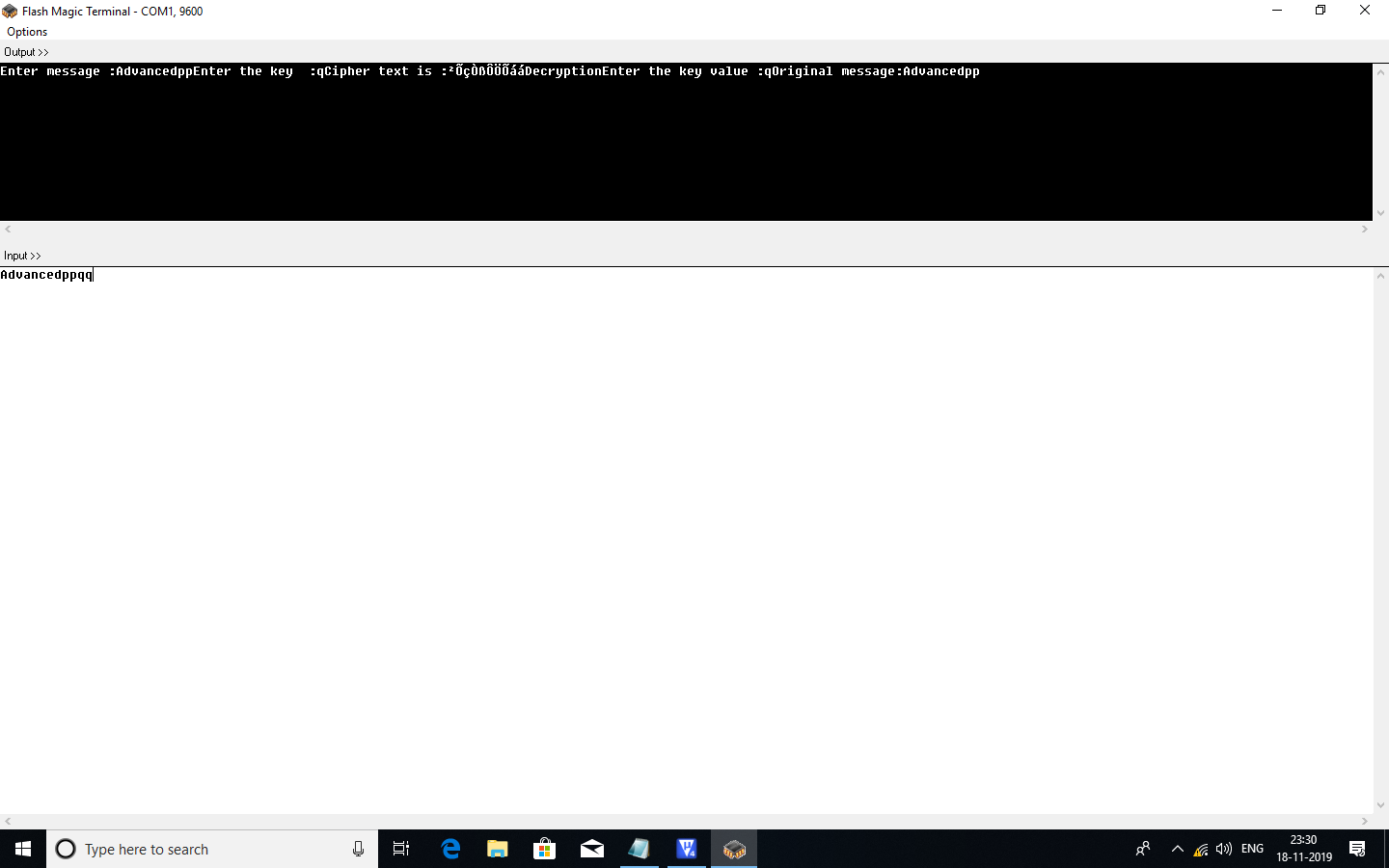
}

void delay(unsigned int r1){

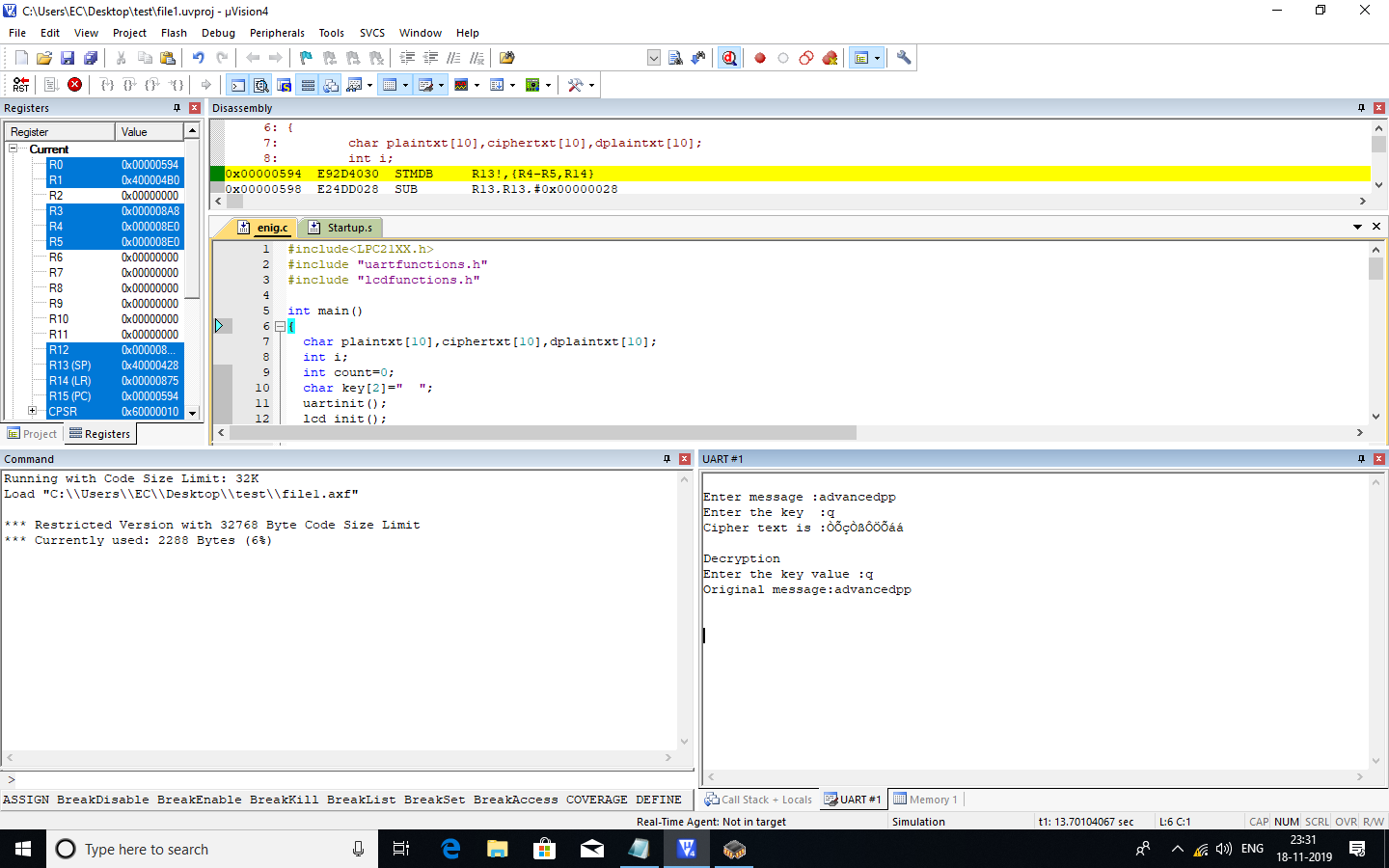
for(r=0; r<r1; r++);

}

**OUTPUT:** a)UART Output:



b)Debug Window:



**APPLICATION:**

This project can be implemented for

1. Defense Communication.
2. Data Security.
3. Encrypting Sensitive Data (Whatsapp).
4. To prevent cyber Threats.

**CONCLUSION:**

The UART0 is used to transmit and receive the characters and this further can be successfully used to interface the GPIO’s depending on the character received and transmitted. The LCD and UART will work paralelly in this process.