#pragma config FOSC = HS // Oscillator Selection bits (HS oscillator)

#pragma config WDTE = OFF // Watchdog Timer Enable bit (WDT disabled)

#pragma config PWRTE = OFF // Power-up Timer Enable bit (PWRT disabled)

#pragma config BOREN = ON // Brown-out Reset Enable bit (BOR enabled)

#pragma config LVP = OFF // Low-Voltage (Single-Supply) In-Circuit Serial Programming Enable bit (RB3/PGM pin has PGM function; low-voltage programming enabled)

#pragma config CPD = OFF // Data EEPROM Memory Code Protection bit (Data EEPROM code protection off)

#pragma config WRT = OFF // Flash Program Memory Write Enable bits (Write protection off; all program memory may be written to by EECON control)

#pragma config CP = OFF // Flash Program Memory Code Protection bit (Code protection off)

#define Input RC7

#define Input1 RC6

#include <xc.h>

#include <stdio.h>

#include <string.h>

#include "supporing\_cfile\lcd.h"

#include "supporing\_cfile\eusart1.h"

/\*

Hardware related definition

\*/

#define \_XTAL\_FREQ 200000000 //Crystal Frequency, used in delay

/\*

Other Specific definition

\*/

void system\_init(void); // This will initialize the system.

void main(void) {

TRISC=0xC0; //RD7 as input

unsigned int a;

TRISD = 0x00;

unsigned char count;

unsigned char RF\_ID[13];

system\_init();

lcd\_com(0x80);

lcd\_puts("Circuit Digest");

while (1){

for (count=0; count<12; count++){

RF\_ID[count] = 0;

RF\_ID[count]=EUSART1\_Read();

}

lcd\_com(0xC0); // Set the cursor for second line begining

lcd\_puts("ID: ");

lcd\_puts(RF\_ID);

int balance=50;

if()

}

int result =0;

result = strcmp("2700197A3F7B", RF\_ID);

if(result == 0)

{

if(balance>0)

{

lcd.clr();

lcd\_com(0xC0); // Set the cursor for second line begining

lcd\_puts("CARd Ok GAte ");

lcd\_puts("Opening");

delay\_ms(1000);

balance=50-10;

lcd\_puts("balance is");

lcd\_puts(balance);

}

}

else{

lcd.clr();

lcd\_com(0xC0); // Set the cursor for second line begining

lcd\_puts("CARd not ok");

lcd\_puts("Close");

}

}

/\*

This Function is for system initializations.

\*/

void system\_init(void){

TRISB = 0x00; //PORT B set as output pin

lcd\_init(); // This will initialize the lcd

EUSART1\_Initialize(); // This will initialize the Eusart

}