REPORT

**RFID BASED SMART LOCK**

**Nidhi Mehta (121029)**

**Priya Shah (121037)**

**Project Idea and Application**

As the name suggests, the project is based on RFID- radio frequency identification. RFID based smart lock consists of RFID Reader, RFID Tag, LCD display, 4X4 keypad, plunger (solenoid lock) and microcontroller unit. RFID can be interfaced to microcontroller through USART. Data is transferred from RFID cards to reader and from there to microcontroller. The main aim of the project was to drive the lock by RFID. The door will unlock only when RFID tag matches the roll number of authorized student.

RFID based Smart lock can be used in many companies and institutions. It permits only specific people having authorization to allow through the door. This can minimize the chaos and also give security from unauthorized people. It also manages attendance for a particular place. It can also change the authorization of people according to time. For example, in an institute, on Monday morning a particular batch will attend the lab and it will be unauthorized for everyone else.

**Introduction**

Every student will have their own unique RFID tags. A particular lab will give entry to number of specified RFIDs only. Students will access the lab door with their RFID tag. Only a list of pre- specified students will be authorized to enter. The lab door will have a RFID reader to respond to the RFID tags. If the tags match with the specified list; the door lock will open. It will also take the attendance of students entering the lab by comparing tag with the list.

RFID reader will read RFID tag. If the tags match to the programmed list of tags then the microcontroller will drive the lock to open.

**Detailed Description**

RFID based smart lock consists of RFID Reader, RFID Tag, LCD display, 4X4 keypad, plunger (solenoid lock) and microcontroller unit. RFID can be interfaced to microcontroller through USART.

The user is given two options:

1. To enter the room having valid RFID card
2. To add RFID card if not already in system.

If a person wants to enter the room, a valid RFID card must be placed at the reader to validate. If and only if the RFID number is pre-written in the EEPROM, the reader will authorize the RFID tag. If the RFID card is not authorized then the door will not open. If one wants to add his/her RFID card then they need authorization of a superior. The superior must have a RFID master tag. He will be able to add new RFID tags and their corresponding roll numbers by keypad. All this data will be saved in EEPROM. When a student will place the RFID card on the RFID reader, the card number will be compared to all the roll numbers in the EEPROM with their corresponding RFID.

The transmitter pin of the reader is connected to receiver pin of Atmega32. The microcontroller drives the input to check the tag. The communication between reader and microcontroller is done by USART. The baud rate is 9600.

We used 12V relay to drive the plunger. But as the current would go up to 5mA, we decided to switch to LED. We used 4bit data line to interface the 16X2 LCD. So the data in LCD was interfaced nibble by nibble. We also used 4X4 keypad using 8bit data line.

The data is stored in EEPROM through microcontroller. We can read or write data in EEPROM.

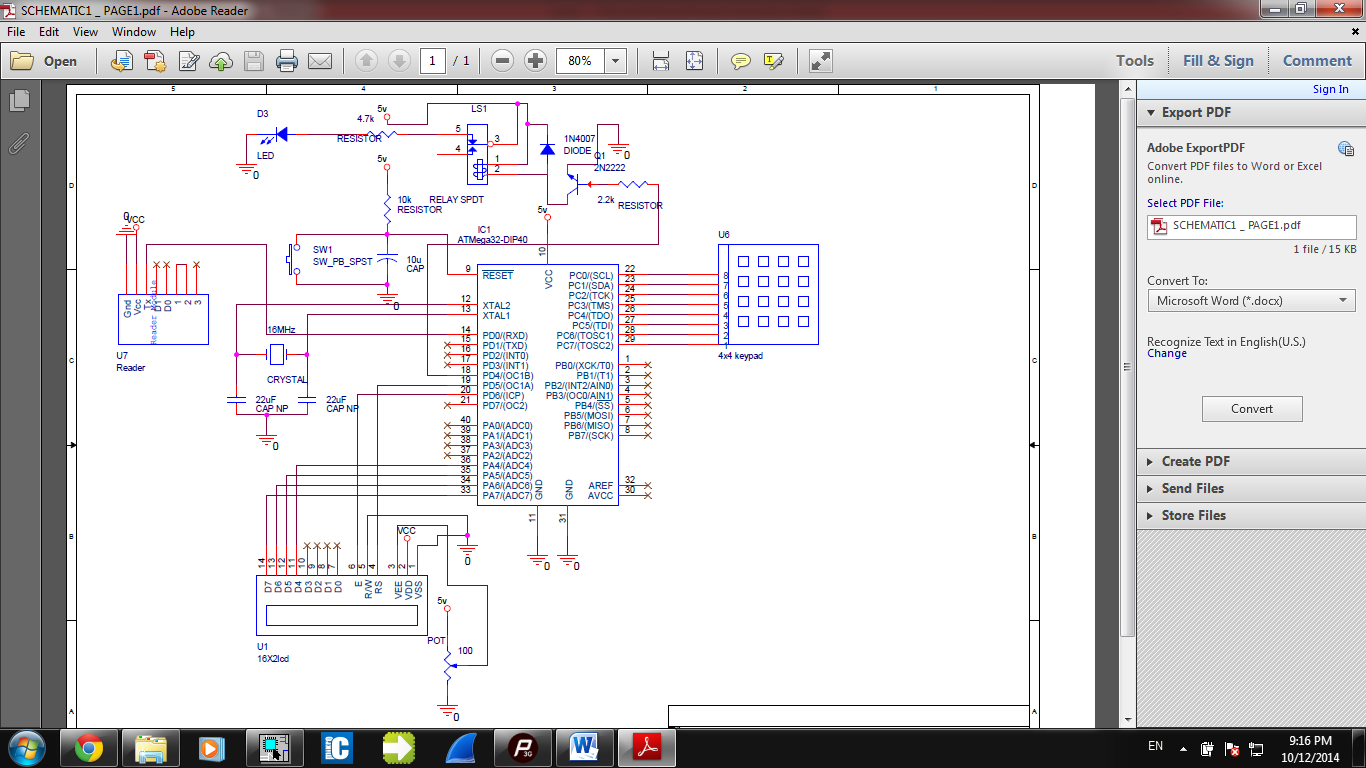
Later we designed a power supply circuit of 5V to drive the whole system. We decided to use a 5V relay as now we had to turn ON LED instead of plunger.

**Keypad:** In Keypad one can enter numbers 0-9 and there are following other functionalities:

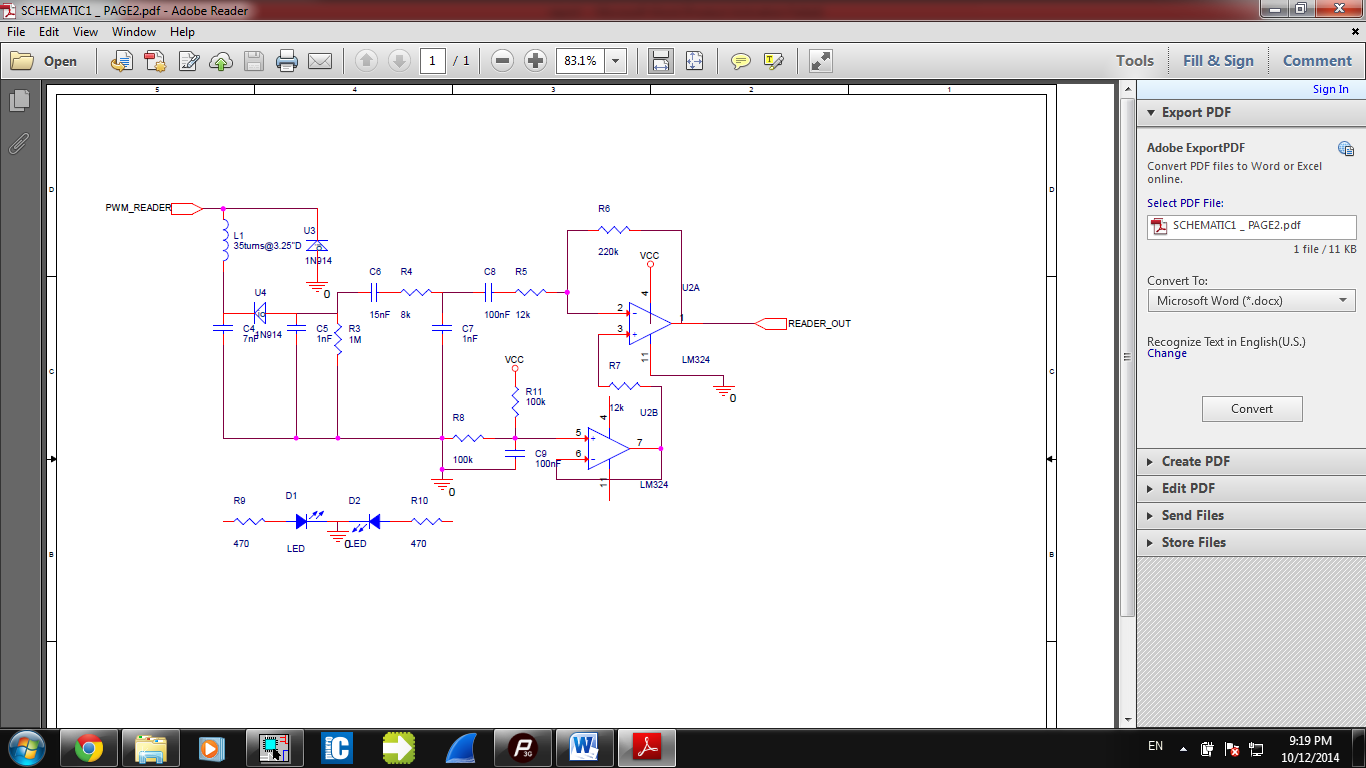
1.Backspace

2.Reset

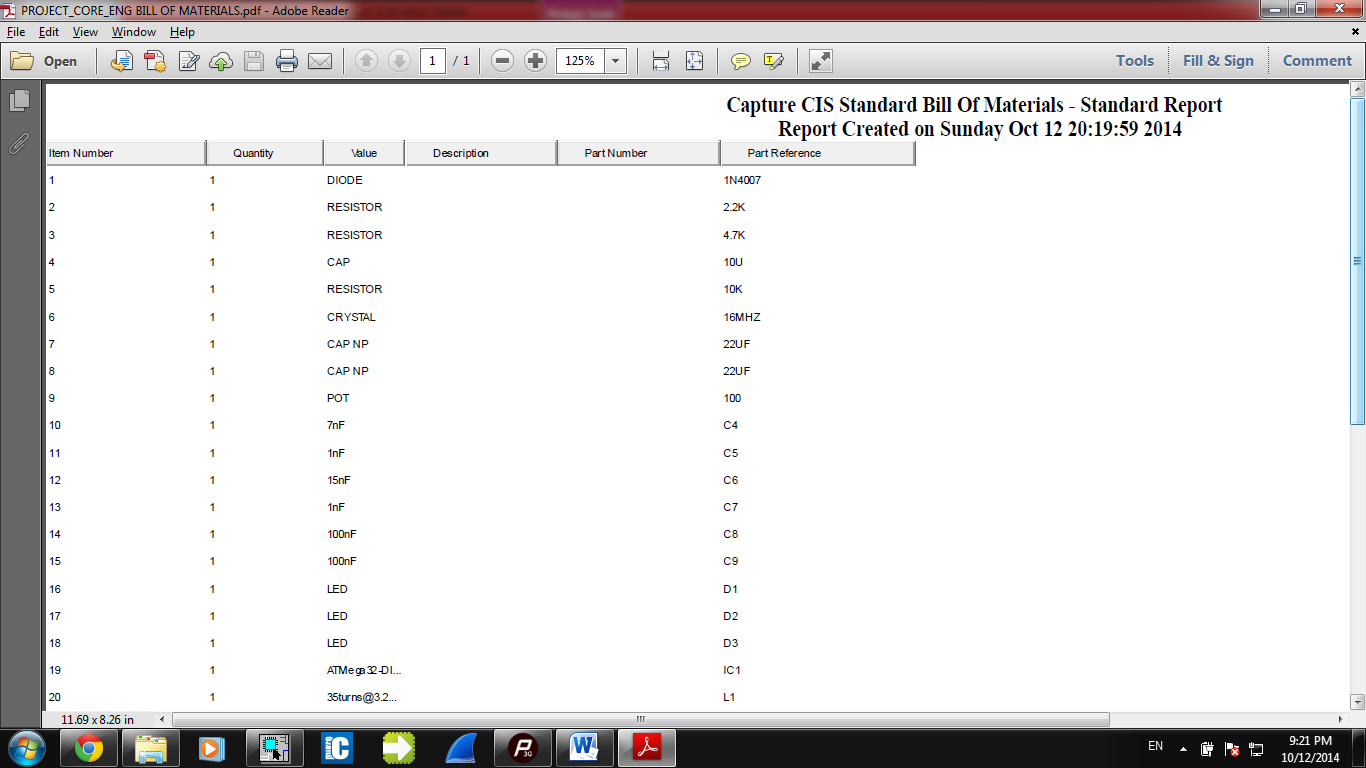
3.Enter.

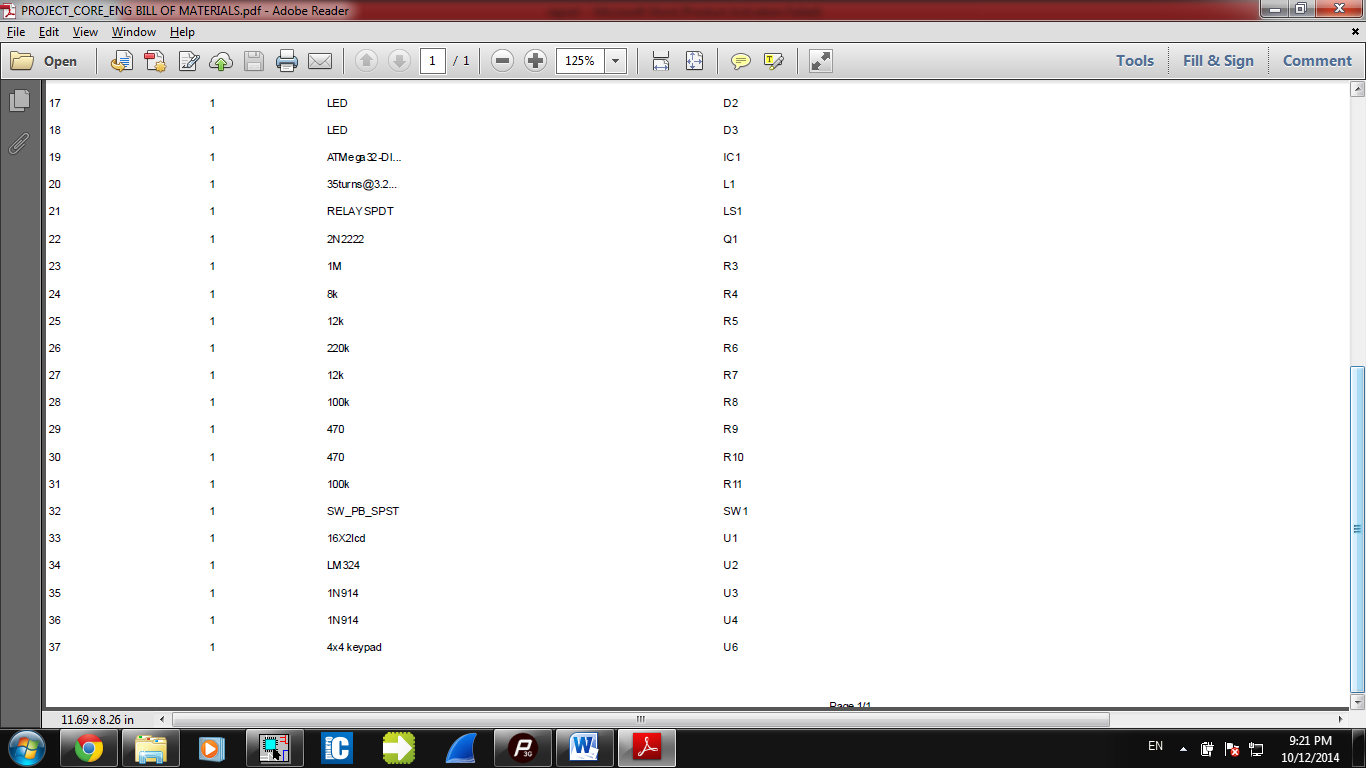
**Schematic Diagram**

[**SCHEMATIC OF MICROCONTROLLER AND ITS PERIPHERALS**](SCHEMATIC1%20_%20PAGE1.pdf)



[**SCHEMATIC OF READER**](SCHEMATIC1%20_%20PAGE2.pdf)

**Bill of Materials** 



[**BILL OF MATERIALS**](PROJECT_CORE_ENG%20BILL%20OF%20MATERIALS.pdf)

**Flowchart of Firmware**

START

1. ENTER ROOM

2. ADD CARDS

3

IS CHOICE 2?

IS CHOICE 1?

NO

NO

YES

YES

INVALID

PLACE CARD

4

PLACE MASTER

READER RECEIVES RFID NUMBER FROM RFID TAG

2

1

4

3

2

1

DOES RFID NUMBER MATCH RFIDs STORED IN EEPROM?

DOES RFID NUMBER MATCH MASTER RFID STORED IN ARRAY?

NO

NO

YES

YES

6

PLACE RFID TAG

VALID

CORRESPONDING ROLL NUMBER IN EEPROM

STORE RFID IN EEPROM

OPENS DOOR LOCK

ENTER ROLL NUMBER

3

ROLL NUMBER IS ENTERED THROUGH KEYPAD AND STORED IN EEPROM

5

5

TO ADD MORE CARDS

PRESS \*

NO

IS KEY \*?

3

YES

6

**C Code**

//LCD Connections

#define LCD\_RS PORTD.B5 // RS

#define LCD\_EN PORTD.B6 //Enable

#define LCD\_D4 PORTA.B4 //Data Bit 4

#define LCD\_D5 PORTA.B5 //Data Bit 5

#define LCD\_D6 PORTA.B6 //Data Bit 6

#define LCD\_D7 PORTA.B7 //Data Bit 7

unsigned char RFID\_No[12],ch2,ch,rfid[13];

unsigned char masterrfid[12]={'2','9','0','0','9','1','0','1','D','E','6','7'};

int flag=0,j=0,k,rolladd,rfidadd=0;

unsigned char rollno\_key[7];

unsigned int i=0;

unsigned char keypad[4][4] ={'A','3','2','1',

'B','6','5','4',

'R','9','8','7',

' ','#','0','\*'};

int keypadfunc()

{

unsigned char ch1;

unsigned char collumn,row;

DDRC=0xF0;

PORTC=0x0F;

do

{

PORTC =0x0F;

collumn =(PINC&0x0F) ;

}

while(collumn!=0x0F);

// Once the key is pressed

do

{

do

{

PORTC =0x0F;

delay\_ms(20);

collumn=(PINC&0x0F);

}

while(collumn==0x0F);

// Debounce check

PORTC =0x0F;

delay\_ms(20);

collumn=(PINC&0x0F);

}

while(collumn==0x0F);

while(1)

{

// ROW and COLLUMN CHECK

// Check if key in first row is pressed

PORTC=0xEF;

delay\_us(1);

collumn=(PINC&0x0F);

if(collumn!=0x0F)

{

row=0;

break;

}

// Check if key in second row is pressed

PORTC=0XDF;

delay\_us(1);

collumn=(PINC&0x0F);

if(collumn!=0x0F)

{

row=1;

break;

}

// Check if key in third row is pressed

PORTC=0xBF;

delay\_us(1);

collumn=(PINC&0x0F);

if(collumn!=0x0F)

{

row=2;

break;

}

//// Key in fourth row is pressed

PORTC=0X7F;

delay\_us(1);

collumn=(PINC&0x0F);

row=3;

break;

}

// COLLUMN CHECK

//Once collumn is decide,

if(collumn==0x0E)

{

//PORTD=(keypad[row][0]);

ch1=keypad[row][0];

}

else if(collumn==0x0D)

{

//PORTD=(keypad[row][1]);

ch1=keypad[row][1];

}

else if(collumn==0x0B)

{

//PortD=(keypad[row][2]);

ch1=keypad[row][2];

}

else

{

//PortD=(keypad[row][3]);

ch1=keypad[row][3];

}

return ch1;

}

void LCD\_data(unsigned char Data)

{

PORTA=Data&0xF0; // Send Higher nibble (D7-D4)

LCD\_RS=1; // Register Select =1 (for data select register)

LCD\_EN=1; //Enable=1 for H to L pulse

delay\_us(5);

LCD\_EN=0;

PORTA=((Data<<4)&0xF0); // Send Lower nibble (D3-D0)

LCD\_EN=1; //Enable=1 for H to L pulse

delay\_us(5);

LCD\_EN=0;

delay\_us(100);

}

//LCD Print

void LCD\_Print(char \* str)

{

unsigned char i=0;

// Till NULL charecter is reached, take each character

while(\*(str+i)!=0)

{

LCD\_data(\*(str+i)); // Data sent to LCD data register

i++;

delay\_ms(10);

}

}

//LCD Command

void lcdcommand(unsigned char command)

{

PORTA=command&0xF0; // Send Higher nibble (D7-D4)

LCD\_RS=0; // Register Select =0 (for Command register)

LCD\_EN=1; //Enable=1 for H to L pulse

delay\_us(5);

LCD\_EN=0;

delay\_us(100);

PORTA=((command<<4)&0xF0); // Send Lower nibble (D3-D0)

LCD\_EN=1; //Enable=1 for H to L pulse

delay\_us(5);

LCD\_EN=0;

delay\_us(40);

}

// Cursor Posotion

void Cursor\_Position(unsigned short int x,unsigned short int y)

{

unsigned char firstcharadd[] ={0x80,0xC0}; // First line address 0X80

//Second line address 0XC0

lcdcommand((firstcharadd[x-1]+y-1));

}

void clear()

{

lcdcommand(0x01);

delay\_ms(2);

}

//LCD Iniatialize

void LCD\_Initialize()

{

LCD\_EN=0;

lcdCommand(0x33); // Initialize LCD for 4 bit mode

lcdCommand(0x32); // Initialize LCD for 4 bit mode

lcdCommand(0x28); // Initialize LCD for 5X7 matrix mode

lcdCommand(0x0E); //Display on,cursor blinking

clear();

lcdCommand(0x06); //Shift cursor to right

lcdCommand(0x80);

}

int checkValidInput() //Checks whether the RFID number is detected properly By xoring even n odd bits respectively and comparing result with last two bits repectively

//if they match then it returns 1 else returns 0.

{

unsigned short int i = 0;

unsigned char RFID\_No1[13];

for(i=0;i<13;i++)

RFID\_No1[i]=RFID\_No1[i];

for(; i<13; i++)

RFID\_No1[i] = RFID\_No1[i]>64 ? RFID\_No1[i]-55 : RFID\_No1[i]-48;

if( ( ( (RFID\_No1[0] ^ RFID\_No1[2]) ^ RFID\_No1[4] ) ^ RFID\_No1[6] ) ^ RFID\_No1[8] == RFID\_No1[10]

&& ( ( (RFID\_No1[1] ^ RFID\_No1[3]) ^ RFID\_No1[5] ) ^ RFID\_No1[7] ) ^ RFID\_No1[9] == RFID\_No1[11])

{

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

RFID\_No1[i] = RFID\_No1[i]>9 ? RFID\_No1[i]+55 : RFID\_No1[i]+48;

return 1;

}

return 0;

}

void eep\_write(unsigned int uiAddress, unsigned char ucData)

{

/\* Wait for completion of previous write \*/

while(EECR & (1<<EEWE));

EEARH=0x00;

/\* Set up address and data registers \*/

EEARL = uiAddress;

EEDR = ucData;

/\* Write logical one to EEMWE \*/

EECR |= (1<<EEMWE);

/\* Start eeprom write by setting EEWE \*/

EECR |= (1<<EEWE);

}

unsigned char eep\_read(unsigned int uiAddress)

{

/\* Wait for completion of previous write \*/

while(EECR & (1<<EEWE))

;

EEARH=0x00;

/\* Set up address register \*/

EEARL = uiAddress; --

/\* Start eeprom read by writing EERE \*/

EECR |= (1<<EERE);

/\* Return data from data register \*/

return EEDR;

}

void enterID()

{

E:flag=0;

LCD\_Initialize(); //Initialize

LCD\_Print("Place Mastercard");

Delay\_ms(50);

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

{

while(UCSRA.B7==0); // Wait till Data is received

RFID\_No[i]=UDR;

}

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

rfid[i]=RFID\_No[i];

rfid[12]='\0';

Cursor\_Position(2,1);

LCD\_Print(rfid);

delay\_ms(2000);

if(checkValidInput())

{

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

{

if(rfid[i]==masterrfid[i]) //Compares value with the master rfid already stored in array.

flag++; //increments flag every time match occurs

}

delay\_ms(500);

if(flag==12) //if match is equal to 12 as their are 12 bits then MasterRFID is correct.

{

clear();

LCD\_print("Correct");

delay\_ms(2000);

}

else

{

clear();

LCD\_print("Wrong");

delay\_ms(2000);

goto E;

}

}

else

{

clear();

LCD\_Print("Place again:");

delay\_ms(500);

goto D;

}

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

rfid[i]='\0';

do{

LCD\_Initialize(); //Initialize

LCD\_Print("Place ID:");

delay\_ms(50);

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

{

while(UCSRA.B7==0); // Wait till Data is received

RFID\_No[i]=UDR;

}

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

{

eep\_write(j,RFID\_No[i]);

j++;

}

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

rfid[i]=RFID\_No[i];

k=j;

Cursor\_Position(2,1);

LCD\_Print(rfid);

delay\_ms(2000);

clear(); //Initialize

LCD\_Print("Enter rollno:");

A: i=1;

Cursor\_Position(2,1);

while(ch2!=' ')

{

ch2=keypadfunc();

if(ch2=='B') //--------------------------------------------------------------->backspace

{

ch2=' ';

Cursor\_position(2,i-1);

if(i<=1)

{

Cursor\_position(2,1);

i=2;

}

LCD\_data(ch2);

i--;

ch2=0;

Cursor\_position(2,i);

rollno\_key[i-1]=0;

k--;

continue;

}

if(ch2=='R') //---------------------------------------------------------------------->reset

{

goto A;

}

rollno\_key[i-1]=ch2;

i++;

LCD\_data(ch2);

if(ch2==' ')

continue;

eep\_write(k,ch2);

k++;

}

delay\_ms(2000);

ch2=0;

j=k;

LCD\_Initialize(); //Initialize

LCD\_Print("Add more cards" );

Cursor\_Position(2,1);

LCD\_Print("Press \*");

ch=keypadfunc();

}while(ch=='\*');

}

void usart\_initialize()

{

UCSRB=0x18; // Rx Enable

UCSRC=0x86; // Data Size : 8-bit, Stop Bit:1,No parity

UBRRL=0x33; // X= (Fosc/(16(Desired Baud Rate)))-1

// =(8\*10^6/(16 \*9600))-1

// =52.08-1

// =51 (Dec)

//Here, URSEl=0, so Fosc is divided by 16 if it was 1 Fosc would

//Have been diveded by 8

}

void main()

{

unsigned char choice='0';

DDRA=0xF0; // For D3-D0

DDRD.B5=1; //For RS

DDRD.B6=1; //For Enable

DDRD.B3=1;

DDRD.B0=0;

PORTD.B3=0;

rollno\_key[6]='\0';

LCD\_Initialize(); //Initialize

usart\_initialize();

while(1)

{

X: LCD\_Initialize(); //Initialize

LCD\_Print("1.Enter room");

Cursor\_Position(2,1);

LCD\_Print("2.Add cards");

choice=keypadfunc();

if(choice!='0')

{

clear();

LCD\_Data(choice);

delay\_ms(500);

}

if(choice=='1')

{

Clear();

LCD\_Print("Place card");

rfidadd=0;

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

{

while(UCSRA.B7==0); // Wait till Data is received

RFID\_No[i]=UDR;

}

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

rfid[i]=RFID\_No[i];

rfid[12]='\0';

if(checkValidInput())

{

flag=0;

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

{

if(rfid[i]==masterrfid[i])

flag++;

}

if(flag==12)

{

Cursor\_Position(2,1);

LCD\_Print("Valid");

delay\_ms(1000);

PORTD.B3=1;

delay\_ms(5000);

PORTD.B3=0;

goto X;

}

for(j=0;j<6;j++)

{

LCD\_Initialize();

flag=0;

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

{

if(rfid[i]==eep\_read(rfidadd))//reads each rfid and compares with detected rfid

flag++;

rfidadd++; //counter to increment address in EEPROM where rfids are stored.

}

rolladd=rfidadd;

if(flag==12)

break;

for(i=0;i<6;i++)

{

rolladd++; //counter to increment address in EEPROM where rollnos are stored.

}

rfidadd=rolladd;

}

if(flag!=12)

{

Cursor\_Position(2,1);

LCD\_Print("Invalid");

delay\_ms(2000);

goto X;

}

clear();

LCD\_Print("Valid");

delay\_ms(1000);

PORTD.B3=1;

Cursor\_Position(2,1);

for(i=0;i<6;i++)

{

LCD\_data(eep\_read(rolladd));

rolladd++;

}

delay\_ms(5000);

PORTD.B3=0;

}

else

{

LCD\_Print("Place again");

delay\_ms(2000);

goto Y;

}

}

if(choice=='2')

{

clear();

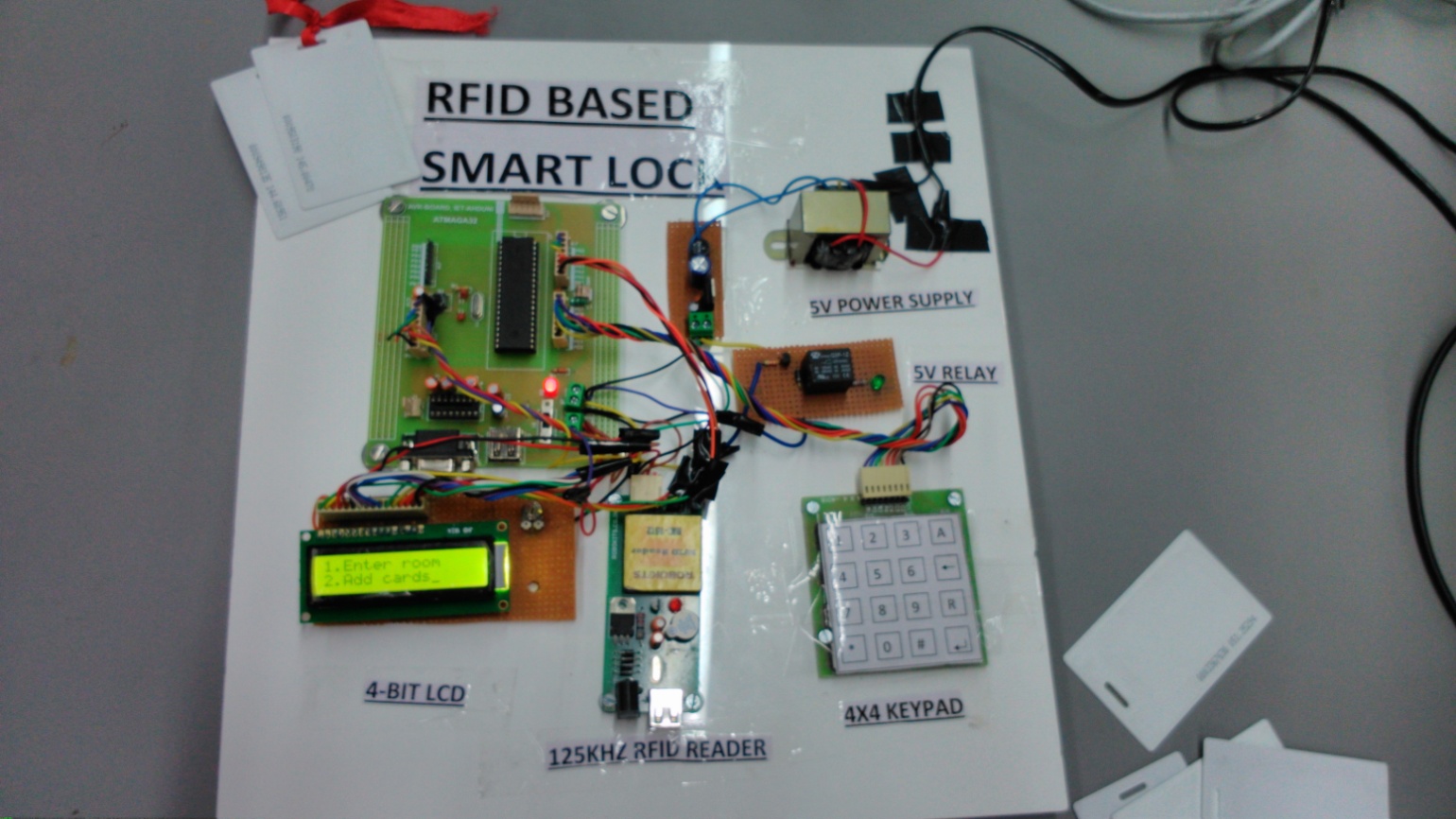
enterID(); //Function to add more cards

}

}

}

**Photos/Videos**



[VIDEOS](MOV_0020.mp4)

[PHOTOS](photos.zip)

**Conclusion & Learning**

* We tried building an RFID reader. We managed to construct the design on PCB, and see the deflection of the output when RFID tag was placed on Antenna. But we could not capture the RFID number.
* We learned to make Antenna for RFID reader.
* We learned the techniques of soldering.
* Importance of time management.
* We understood the importance of firmware as well as hardware and software.
* We learned that when there is a fault in the system, we should not keep on amending the software. There might be a possibility of hardware failure.
* We understood the importance of testing the hardware and software, independently as well as composed, in different scenarios.