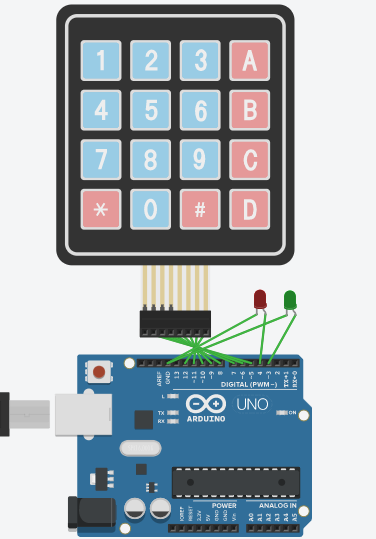
IIOT Assignment-7

(Dheeraj Tiwari)

1. Perform an experiment to Interface Hex keypad as to implement Password security system?

Ans :



#include<Keypad.h>

const byte rows=4;

const byte cols=4;

String password="1A2B3C";

String temp="";

int led1=3;

int led2=4;

char hexkeys[rows][cols]={

{'1','2','3','A'},

{'4','5','6','B'},

{'7','8','9','C'},

{'\*','0','#','D'},

};

byte rowPins[rows]={5,6,7,8};

byte colPins[cols]={9,10,11,12};

Keypad

customKeypad=Keypad(makeKeymap(hexkeys),rowPins,colPins,rows,cols);

void setup(){

Serial.begin(9600);

pinMode(led1,OUTPUT);

pinMode(led2,OUTPUT);}

void loop(){

char key=customKeypad.getKey();

if(key){

Serial.println(key);

if(key=='0'|| key=='1' || key=='2' || key=='3'|| key=='4'||

key=='5'||

key=='6' || key=='7' || key=='8'|| key=='9' ||

key=='A'||key=='B' || key=='C'|| key=='D')

{

temp=temp+key;

}

if(key=='\*')

{

if(temp==password){

digitalWrite(led1,HIGH);

Serial.println("right password");}

else{

digitalWrite(led2,HIGH);

Serial.println("wrong password");

}}

if(key=='#')

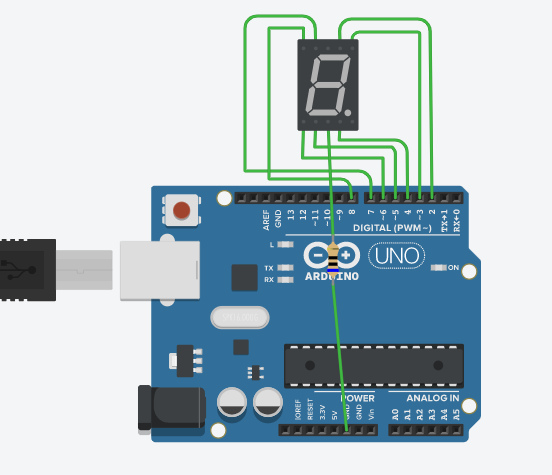
{temp="";

digitalWrite(led1,LOW);

}}}

1. Perform an experiment to Interface 7- segment display with Arduino Board, display 0-9 digits?

Ans:



int num\_array[10][7] = { { 1,1,1,1,1,1,0 }, // 0

{ 0,1,1,0,0,0,0 }, // 1

{ 1,1,0,1,1,0,1 }, // 2

{ 1,1,1,1,0,0,1 }, // 3

{ 0,1,1,0,0,1,1 }, // 4

{ 1,0,1,1,0,1,1 }, // 5

{ 1,0,1,1,1,1,1 }, // 6

{ 1,1,1,0,0,0,0 }, // 7

{ 1,1,1,1,1,1,1 }, // 8

{ 1,1,1,0,0,1,1 }}; //9

void setup()

{

for(int i=2;i<9;i++){

pinMode(i, OUTPUT); } }

void loop()

{

for (int counter = -1; counter < 9;counter++)

{

delay(1000);

Num\_Write(counter+1);

}delay(3000);}

void Num\_Write(int number)

{ int pin= 2;

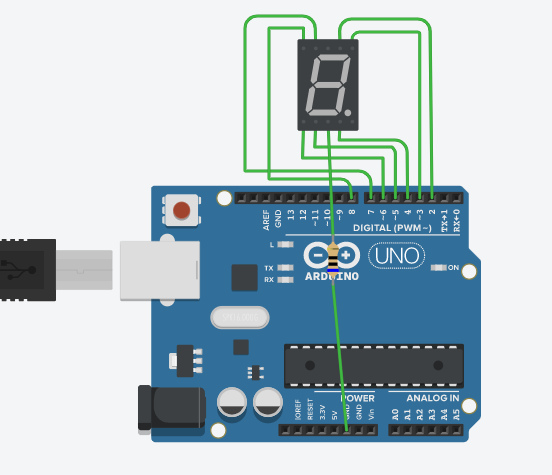
for (int j=0; j < 7; j++) {

digitalWrite(pin, num\_array[number][j]);

pin++;

}}

1. Que-Assign: Perform an experiment to Interface 7- segment display with Arduino Board, display 9-0 digits?

Ans : 

int num\_array[10][7] = { { 1,1,1,1,1,1,0 }, // 0

{ 0,1,1,0,0,0,0 }, // 1

{ 1,1,0,1,1,0,1 }, // 2

{ 1,1,1,1,0,0,1 }, // 3

{ 0,1,1,0,0,1,1 }, // 4

{ 1,0,1,1,0,1,1 }, // 5

{ 1,0,1,1,1,1,1 }, // 6

{ 1,1,1,0,0,0,0 }, // 7

{ 1,1,1,1,1,1,1 }, // 8

{ 1,1,1,0,0,1,1 }}; // 9

void setup()

{

// set pin modes

for(int i=2;i<9;i++){

pinMode(i, OUTPUT); }

}

void loop()

{

//counter loop

for (int counter = 10; counter > 0; --counter)

{

delay(1000);

Num\_Write(counter-1);

}

delay(3000);

}

// this functions writes values to the sev seg pins

void Num\_Write(int number)

{

int pin= 2;

for (int j=0; j < 7; j++) {

digitalWrite(pin, num\_array[number][j]);

pin++;

}}

1. Explain the working of 7-Segment LED display?

Ans: The 7-segment display, also written as “seven segment display”, consists of seven LEDs (hence its name) arranged in a rectangular fashion as shown. Each of the seven LEDs is called a segment because when illuminated the segment forms part of a numerical digit (both Decimal and Hex) to be displayed. An additional 8th LED is sometimes used within the same package thus allowing the indication of a decimal point, (DP) when two or more 7-segment displays are connected together to display numbers greater than ten.

Each one of the seven LEDs in the display is given a positional segment with one of its connection pins being brought straight out of the rectangular plastic package. These individually LED pins are labelled from a through to g representing each individual LED. The other LED pins are connected together and wired to form a common pin.

So by forward biasing the appropriate pins of the LED segments in a particular order, some segments will be light and others will be dark allowing the desired character pattern of the number to be generated on the display. This then allows us to display each of the ten decimal digits 0 through to 9 on the same 7-segment display.

1. Perform an experiment to interface hex keypad with Arduino such that when user press
   1. 1-single LED will glow
   2. 2-Two LED’S will glow
   3. \*-Both LED’s gets off

Ans :

#include<Keypad.h>

const byte rows=4;

const byte cols=4;

int led1=3;

int led2=4;

char hexkeys[rows][cols]={

{'1','2','3','A'},

{'4','5','6','B'},

{'7','8','9','C'},

{'\*','0','#','D'},

};

byte rowPins[rows]={5,6,7,8};

byte colPins[cols]={9,10,11,12};

Keypad

customKeypad=Keypad(makeKeymap(hexkeys),rowPins,colPins,rows,cols);

void setup(){

Serial.begin(9600);

}

void loop(){

char customKey=customKeypad.getKey();

if(customKey=='1'){

digitalWrite(led1,HIGH);

}else if(customKey=='2'){

digitalWrite(led2,HIGH);

}else if(customKey=='\*'){

digitalWrite(led1,LOW);

digitalWrite(led2,LOW);

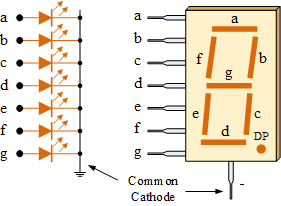
}else{};}

1. Explain the difference between common cathode and common anode?

Ans: The displays common pin is generally used to identify which type of 7-segment display it is. As each LED has two connecting pins, one called the “Anode” and the other called the “Cathode”, there are therefore two types of LED 7-segment display called: **Common Cathode** (CC) and **Common Anode** (CA).

1. **The Common Cathode (CC)** – In the common cathode display, all the cathode connections of the LED segments are joined together to logic “0” or ground. The individual segments are illuminated by application of a “HIGH”, or logic “1” signal via a current limiting resistor to forward bias the individual Anode terminals (a-g).

### Common Cathode 7-segment Display:



2. **The Common Anode (CA)** – In the common anode display, all the anode connections of the LED segments are joined together to logic “1”. The individual segments are illuminated by applying a ground, logic “0” or “LOW” signal via a suitable current limiting resistor to the Cathode of the particular segment (a-g).

### Common Anode 7-segment Display:

