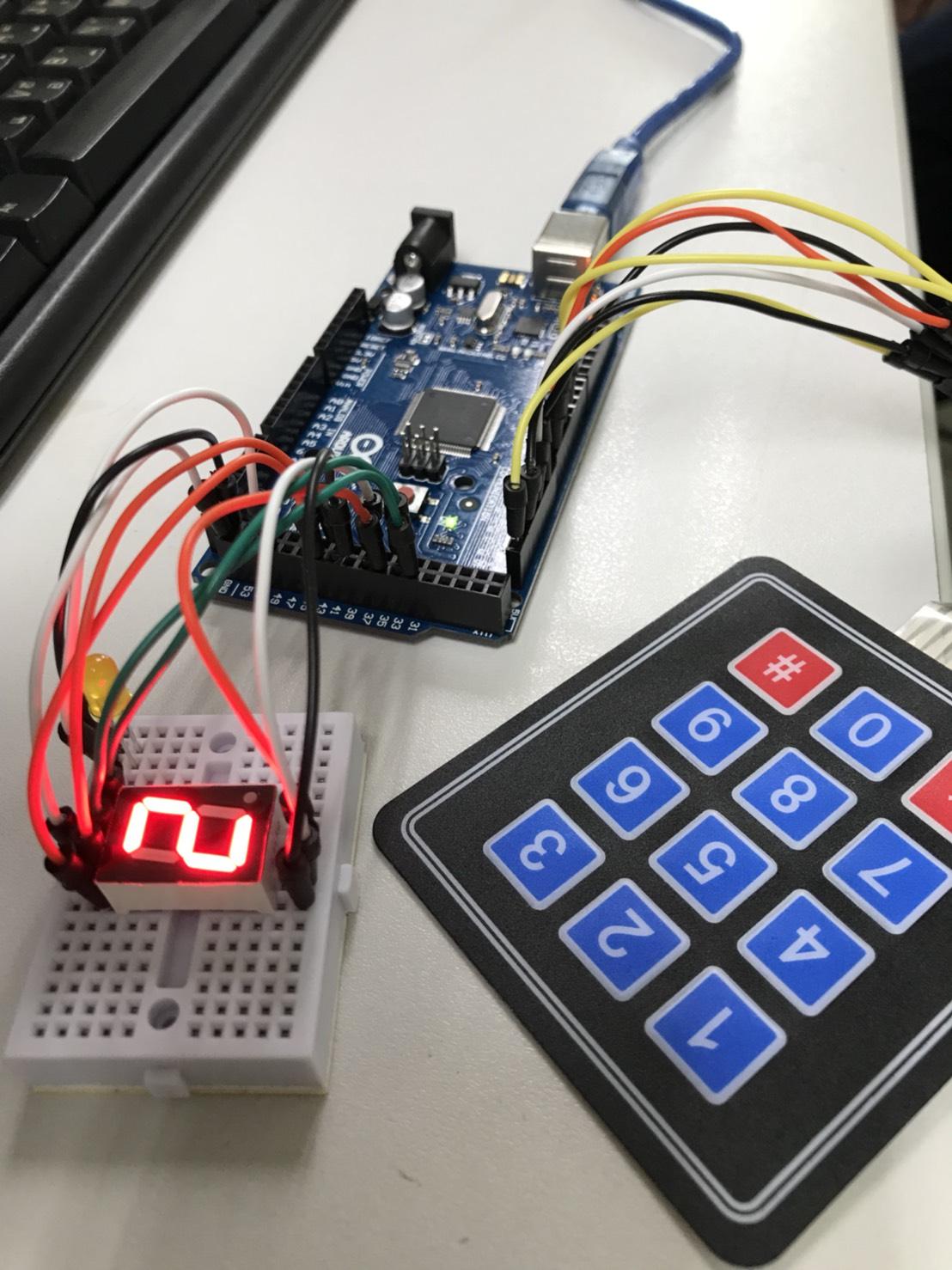
****

**Project Report : Memory Game & Addition Game**

Sarun Nuntaviriyakul 6031851521

Rhach Roongnirandon 6031838421

**Objective**

To create a game that use the knowledge of arduino that is easy to play and will help the player improve in memory and calculation.

**Description**

A memory game that will display let the user choose from Memory game or Addition game by pressing \* or # . Memory game will show a number on the 7 segment display and the user need to remember the number and press it . Addition game will let the player add up the number by adding the number display on the 7 segment display and press enter by the # . If the player get it wrong the display will display E n d and the LED will light up. The game will then start again.

**Specifications**

**Controller** Arduino Mega 2560 r3

**LED**  1

**Additional material** Keypad 4x3 Keypad

7 Segment display

**Dimensions of the box**

Width x Height x Depth

8 cm x 5.5 cm x 17.5 cm

**Hardware list**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Quantity** | **Unit Price**  **(THB)** | **Total price**  **(THB)** |
| Arduino Mega 2560 r3 | 1 | 440.00 | 440.00 |
| LED | 1 | 2.00 | 2.00 |
| Jumper Cable | 17 | 1.00 | 17.00 |
| 4x3 keypad | 1 | 35.00 | 35.00 |
| 7 Segment display | 1 | 10.00 | 10.00 |
|  |  | **Total** | 504.00 |

**How the code work**

Memory Game

The program will random a number between 0-9 and put it in an array . And detect the input from the user and check it if it is equal to the value in the array . By checking the number the user press until the number of the random number in the array.

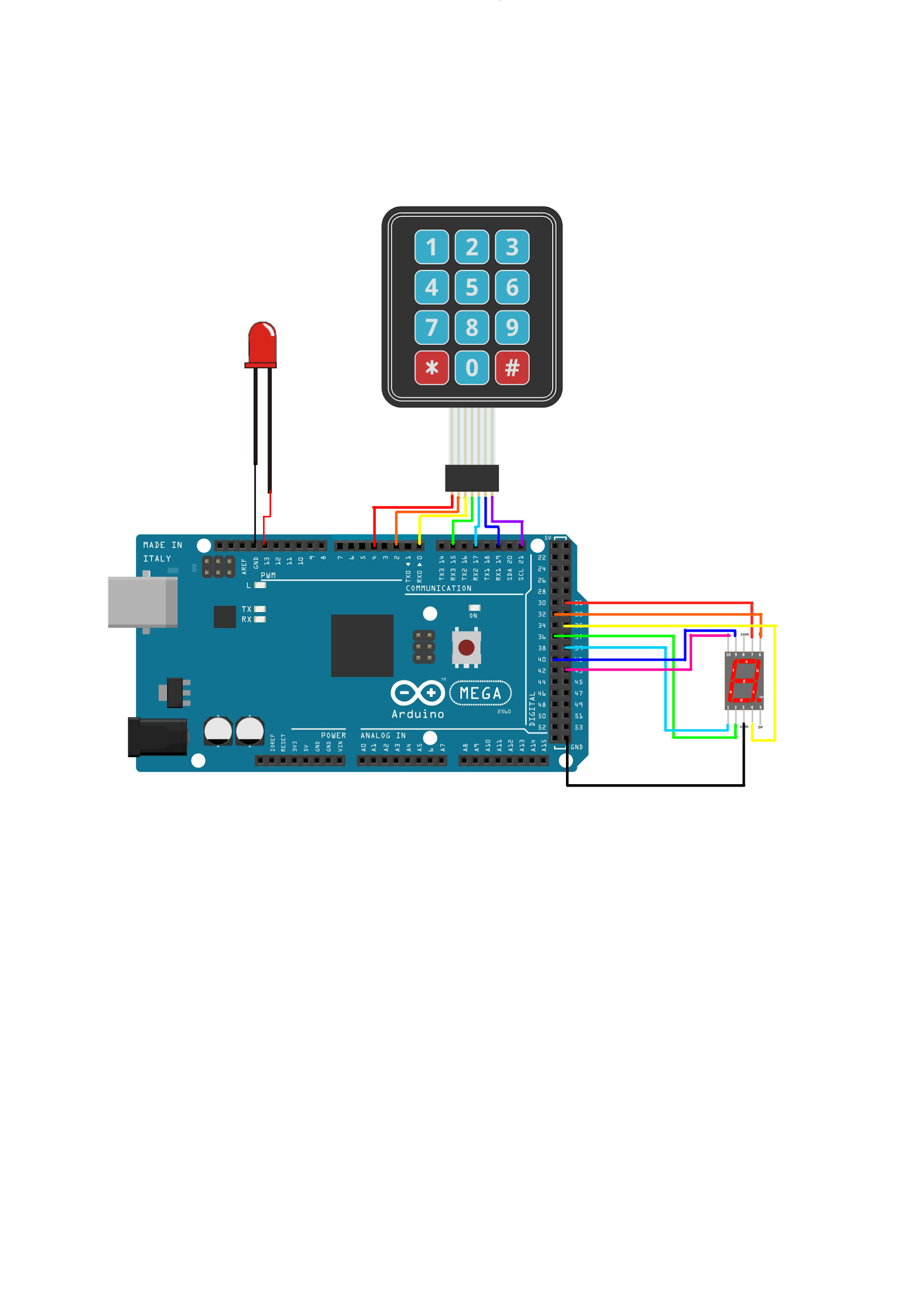
Addition Game

The program will random a number between 0-9 and will display it in a 7 segment display .

The player will have to add the number up. And confirm the number by pressing #

The sum of all random number will add up and check it with the input number .

**Schematic Diagram**

****

**The code**

**Include the library of the Keypad and 7 segment display**

[#include](https://www.facebook.com/hashtag/include?source=feed_text) <Keypad.h>

#include <sevenSegmentDisplay.h>

**Define the variable for the 7 segment display**

[#define](https://www.facebook.com/hashtag/define?source=feed_text) segA 31

#define segB 32

#define segC 35

#define segD 36

#define segE 39

#define segF 40

#define segG 43

**Define a reset method**

void(\* resetFunc) (void) = 0; //declare reset function @ address 0

sevenSegmentDisplay ssd(COMMON\_CATHODE, segA,segB,segC,segD,segE,segF,segG,A0);

**Define the variable for the Keypad**

#define numA 21

#define numB 19

#define numC 17

#define numD 15

#define numE 0

#define numF 2

#define numG 4

const byte ROWS = 4;

const byte COLS = 3;

char keys[ROWS][COLS] = {

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

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

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

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

};

byte rowPins[ROWS] = {numG,numF,numE,numD};

byte colPins[COLS] = {numC, numB, numA};

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

int ran[100];

int endGame=0;

int count=0;

int a=0;

int k=0;

String num;

int add=0;

long ans=0;

char key;

char plus;

int check=0;

**Define the variable for the LED**

#define led 13

**Run the game in the setup**

void setup() {

pinMode(led,OUTPUT);

randomSeed(analogRead(0));

Serial.begin(9600);

Serial.print("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

Serial.write("\t\t\tWelcome to Memory Game\n");

Serial.print("\t Press \* for Memory Game \t Press # for Addition Game\n");

Serial.print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n");

while (1)

{

**Get the input from the user for Memory game or Addition game**

char key = keypad.getKey();

while(!key)

{

key = keypad.getKey();

}

**If the user select Memory Game**

if (key=='\*')

{

Serial.print("\t\t\t\tMemory Game\n");

while(1)

{

**Random a number and put it in an array**

ran[k]=random(0,10);

delay(200);

**Show the number**

showNum(ran[k]);

**Check if the user is wrong**

if(checkEnd(k)!=0)

{

**Display an ending message on the 7 segment display**

ledOn(1);

ssd.set('E');

delay(1200);

ssd.set('n');

delay(1200);

ssd.set('d');

delay(1200);

ssd.off();

break;

}

**Blink the LED**

ledOn(0);

endGame++;

K++;

**Print the score**

Serial.print("Score : ");

Serial.println(endGame);

}

Serial.print("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

Serial.println("\t\t\t\tGAME OVER!\n");

Serial.print("Your Score is ");

Serial.print(endGame);

delay(1000);

**Reset the game**

resetFunc(); //call reset

break;

}

**The user select Addition Game**

else if (key=='#')

{

Serial.print("\t\t\t\tAddition Game\n");

while(1)

{

**Random a number**

ran[k]=random(0,10);

delay(200);

showNum(ran[k]);

**Add the result**

add=add+ran[k];

ans=0;

while(1)

{

plus=keypad.getKey();

switch(plus)

{

case '0' ... '9':

**Get the input from the user**

ans=ans\*10 + (plus-'0');

break;

}

if(plus=='#')

{

break;

}

}

**Check if the answer is wrong**

if(add!=ans)

{

break;

}

endGame++;

k++;

ledOn(0);

Serial.print("Score : ");

Serial.println(endGame);

}

ledOn(1);

Serial.print("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

Serial.println("\t\t\t\tGAME OVER!\n");

Serial.print("Your Score is ");

Serial.print(endGame);

ssd.set('E');

delay(1200);

ssd.set('n');

delay(1200);

ssd.set('d');

delay(1200);

ssd.off();

delay(1000);

resetFunc(); //call reset

break;

}

}

}

void ledOn(int i){

digitalWrite(led,HIGH);

**Set the LED to blink**

if(i==0)

{

delay(100);

digitalWrite(led,LOW);

}

}

**Check if the user answer is correct**

int checkEnd(int k)

{

int a=0;

while(a<=k){

char key = keypad.getKey();

while(!key)

{

key = keypad.getKey();

}

**Change from ASCII to number**

key=key-'0';

**Check every number in the array**

if(ran[a]!=key)

{

return -1;

}

a++;

delay(100);

}

return 0;

}

**Show the number on the 7 segment display**

void showNum(int a){

switch(a){

case 1:{ ssd.set(1);

break; }

case 2:{ ssd.set(2);

break; }

case 3:{ssd.set(3);

break; }

case 4:{ssd.set(4);

break; }

case 5:{ssd.set(5);

break; }

case 6:{ssd.set(6);

break; }

case 7:{ ssd.set(7);

break; }

case 8:{ ssd.set(8);

break; }

case 9:{ssd.set(9);

break; }

case 0:{ ssd.set(0);

break; }

}

}

void loop() {

}

Video Demonstration