

Assignment 4

กลุ่ม ขอบตาดำ ไปก่อนละกัน

สมาชิก 1. นายธนทัต จงกิตติสกุล 64010312

2. นางสาวธัญชนก จรุงพัฒนานนท์ 64010259

แนวคิดและวิธีการเล่นเกมแบบสั้นๆ

- เป็นเกมตีปิงปองที่มีไม้ 2 ไม้ โดยทั้ง 2 ไม่ใช้การบังคับพร้อมกัน ผ่านสวิตช์ซ้ายขวา ถ้าตีลูกปิงปองไม่โดน จะถือว่าเกม over และเริ่มเกมใหม่

อุปกรณ์ที่ใช้ในโครงงาน

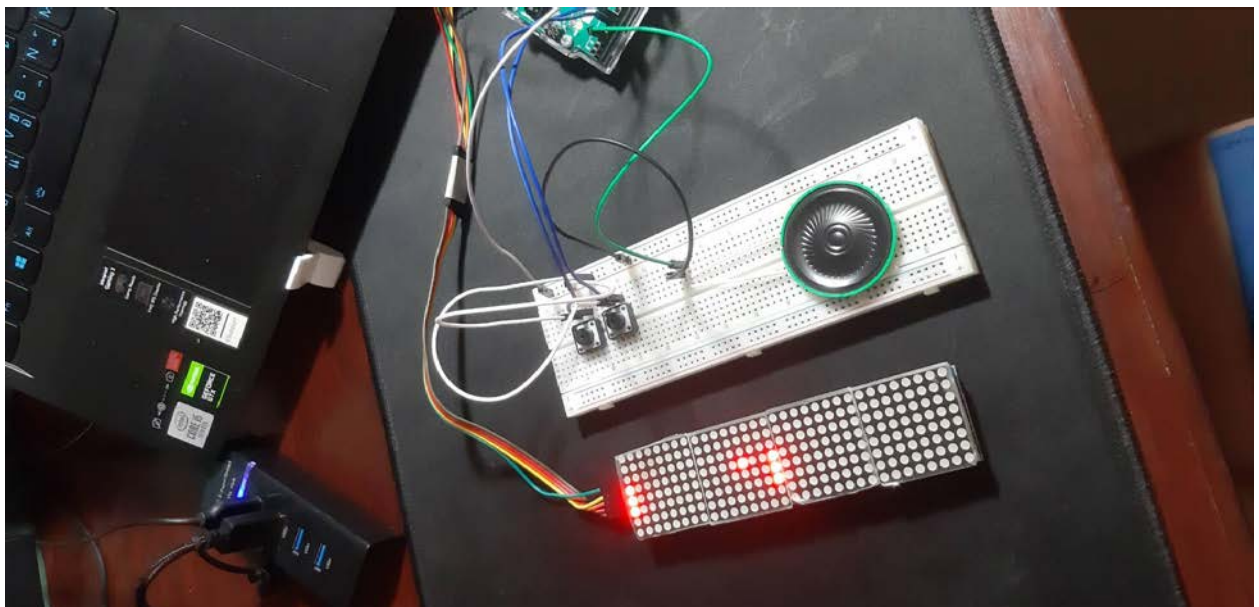
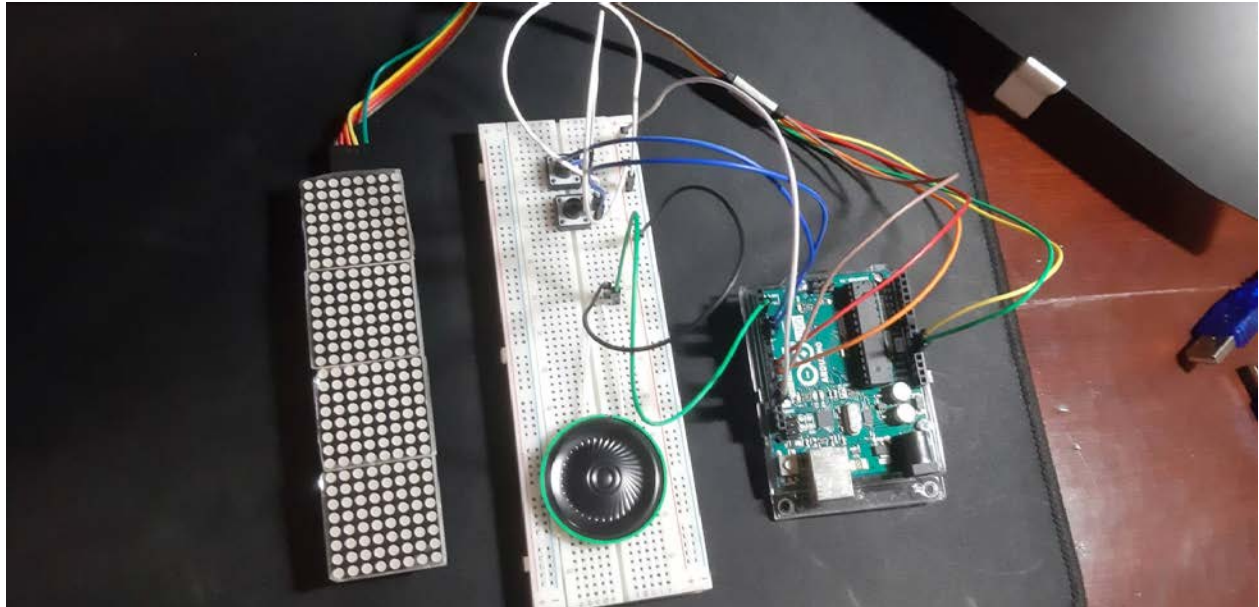
- Dot Matrix
- สายไฟ
- สวิตช์
- ลำโพง 1 อัน
- บอร์ด Arduino UNO
- Breadboard

โครงสร้างของโปรแกรม

- ลูกปิงปองจะแต่งตั้งได้โดยการทำงานของเวกเตอร์ อย่างที่เราได้สร้างใน struct vector2i ซึ่งใน struct จะประกอบด้วยค่าของแกน x และแกน y โดยตำแหน่งในการตั้งของลูกปิงปองนั้น จะเกิดจากฟังก์ชัน plus vector ที่เอาค่าของตำแหน่งและทิศทางของลูกปิงปองมาบวกกัน ทั้งในแกน x และแกน y ทำให้ลูกปิงปองสามารถแต่งตั้งได้ โดยเมื่อลูก

ป้องกันโดนกับไม้ป้องกัน จะมีการสุมค่าทิศทางของลูกป้องกัน ทำให้เกมดูไม่น่าเบื่อและ
มีความท้าทายมากยิ่งขึ้น

รูปถ่ายชิ้นงาน



Source Code

```
#include "LedControl.h"

#define BL 5

#define BR 6

#define speaker 3

#define NOTE_B0 100

#define NOTE_B1 200

#define DO      261.6

#define RE      293.7

#define ME      329.6

#define FA(x) Beep(349.2, x);

#define SO(x) Beep(392.0, x);

#define LA(x) Beep(440.0, x);

#define TE(x) Beep(493.9, x);

#define DDO(x) Beep(523.3, x);

LedControl lc = LedControl(11, 13, 10, 4);

unsigned long delaytime = 100;

struct Vector2i
```

```
{  
  
    int x, y;  
  
};
```

```
Vector2i plusVector(Vector2i a, Vector2i b)
```

```
{  
  
    Vector2i temp;  
  
    temp.x = a.x + b.x;  
  
    temp.y = a.y + b.y;  
  
    return temp;  
  
}
```

```
Vector2i pongPos;
```

```
Vector2i pongDir;
```

```
Vector2i barTopPos;
```

```
Vector2i barBotPos;
```

```
byte G[] =
```

```
{
```

```
B00000000,  
  
B11111111,  
  
B10000001,  
  
B10010001,  
  
B10010001,  
  
B10010001,  
  
B11110001,  
  
B00000000  
};  
  
byte A[] =  
  
{  
  
B00000000,  
  
B11111111,  
  
B00010001,  
  
B00010001,  
  
B00010001,  
  
B00010001,  
  
B11111111,  
  
B00000000  
};
```

```
byte M[] =
```

 $\{$

B00000000,

B11111111,

B00000010,

B00000100,

B00000100,

B00000010,

B11111111,

B00000000

}

```
byte E[] =
```

{

B00000000,

B11111111,

B10011001,

B10011001,

B10011001,

B10011001,

B10011001,

```
B00000000

};

byte O[] =

{

    B00000000,

    B11111111,

    B10000001,

    B10000001,

    B10000001,

    B10000001,

    B11111111 ,

    B00000000

};

byte V[] =

{

    B00000000,

    B00001111,

    B00110000,

    B11000000,

    B11000000,
```

```

    B00110000,

    B00001111 ,

    B00000000

};

byte R[] =

{

    B00000000,

    B11111111,

    B00010001,

    B00010001,

    B00110001,

    B01010001,

    B10001110,

    B00000000

};

void setSprite(byte *sprite)

{

    for (int r = 0; r < 8; r++)

    {

        lc.setRow(0, r, sprite[r]);

```



```

    }

}

int i = 0;

int numOfLoop = 0;

int noteDuration = 1000 / 8;

int pauseBetweenNotes = noteDuration * 0.2;

void gameoversound()

{

    for (numOfLoop = 0; numOfLoop < 4; numOfLoop++)

    {

        for (i = 25; i < 30; i++)

        {

            tone(speaker, 20 * i, noteDuration);

            delay(pauseBetweenNotes);

        }

        for (i = 30; i >= 25; i--)

        {

            tone(speaker, 20 * i, noteDuration);

            delay(pauseBetweenNotes);

        }

    }

}

```

```
    }  
}  
  
delay(500);  
  
tone(speaker, 500, 200);  
  
delay(200);  
  
tone(speaker, 1200, 200);  
  
delay(200);  
  
tone(speaker, 300, 200);  
  
delay(200);  
  
tone(speaker, 1000, 200);  
  
delay(200);  
  
tone(speaker, 400, 200);  
  
delay(200);  
  
tone(speaker, 1100, 200);  
  
delay(200);  
}
```

```
void setup()  
{  
  
    randomSeed(analogRead(A0));
```

```
int devices = lc.getDeviceCount();

for (int address = 0; address < devices; address++)

{

    lc.shutdown(address, false);

    lc.setIntensity(address, 8);

    lc.clearDisplay(address);

}

pinMode(BL, INPUT_PULLUP);

pinMode(BR, INPUT_PULLUP);


pongPos = { 3, 7 };

pongDir = { 0, 1 };


barTopPos = { 3, 0 };

barBotPos = { 3, 15 };

}


void loop()

{

    int left = !digitalRead(BL);
```

```
int right = !digitalRead(BR);

barTopPos.x -= left;

barTopPos.x += right;

barBotPos.x -= left;

barBotPos.x += right;

if (barTopPos.x < 1)

{

    barTopPos.x = 1;

    barBotPos.x = 1;

}

if (barTopPos.x > 6)

{

    barTopPos.x = 6;

    barBotPos.x = 6;

}


if ((barTopPos.x - 1 <= pongPos.x && pongPos.x <= barTopPos.x + 1) &&
(pongPos.y == 1 || pongPos.y == 14))

{

    tone(speaker, DO);

    delay(150);
```

```
noTone(speaker);

pongDir.y = -pongDir.y;

pongDir.x = random(-1, 1);

}

if (pongPos.x == 0 || pongPos.x == 7)

{

    pongDir.x = -pongDir.x;

}

pongPos = plusVector(pongPos, pongDir);

if (pongPos.y < 0 || pongPos.y > 15 )

{

    //tone(speaker,NOTE_B1);

    //delay(800 );

    //noTone(speaker);

    gameoversound();

    setSprite(G);

    lc.clearDisplay(1);

    delay(300);

    setSprite(A);

    lc.clearDisplay(1);
```

```
delay(300);

setSprite(M);

lc.clearDisplay(1);

delay(300);

setSprite(E);

lc.clearDisplay(1);

delay(300);

setSprite(O);

lc.clearDisplay(1);

delay(300);

setSprite(V);

lc.clearDisplay(1);

delay(300);

setSprite(E);

lc.clearDisplay(1);

delay(300);

setSprite(R);

lc.clearDisplay(1);

delay(300);

delay(2000);
```

```

pongPos = { 3, 7 };

pongDir = { 0, 1 };


barTopPos = { 3, 0 };

barBotPos = { 3, 15 };

}


lc.clearDisplay(0);

lc.clearDisplay(1);

lc.setLed(!((pongPos.y / 8), pongPos.x, pongPos.y % 8, true);

lc.setLed(!((barTopPos.y / 8), barTopPos.x, barTopPos.y % 8, true);

lc.setLed(!((barTopPos.y / 8), barTopPos.x + 1, barTopPos.y % 8, true);

lc.setLed(!((barTopPos.y / 8), barTopPos.x - 1, barTopPos.y % 8, true);

lc.setLed(!((barBotPos.y / 8), barBotPos.x, barBotPos.y % 8, true);

lc.setLed(!((barBotPos.y / 8), barBotPos.x + 1, barBotPos.y % 8, true);

lc.setLed(!((barBotPos.y / 8), barBotPos.x - 1, barBotPos.y % 8, true);

delay(delaytime);

}

// lc.setLed(y / 8, x, y % 8, true);

```