**LED MATRIX (8X32) PROJECT**

GAME : STAR LASER

**กลุ่ม : ขายตรงแบบ 300 %A picture containing floor, table, indoor, computer

Description automatically generated**

**LED DOT MATRIX PROJECT’S DETAILED**

**1.เกมส์นี้จะเป็นการแข่งขันระหว่างผู้เล่นกับ BOT**

**2.ผู้เล่นสามารถ เคลื่อนย้ายยาน โดยการโยกปุ่มอนาล๊อค**

**3.ผู้เล่นสามารถ ยิงกระสุน โดยการกดปุ่ม B(ปุ่มขวา)**

**4.เกมส์นี้จะมีผลลัพธิ์ออกมา 2 รูปแบบ คือ ชนะ หรือ แพ้**

**PICTURE AND DISCUSSION**

**A picture containing floor, table, indoor, computer

Description automatically generated**

**เริ่มต้น ก่อนที่จะเข้าไปสู่ตัวเกม เราจะเจอกับ Interface**

**“Press BT” ซึ่งหมายความว่าให้เรากดปุ่มใดปุ่มหนึ่ง**

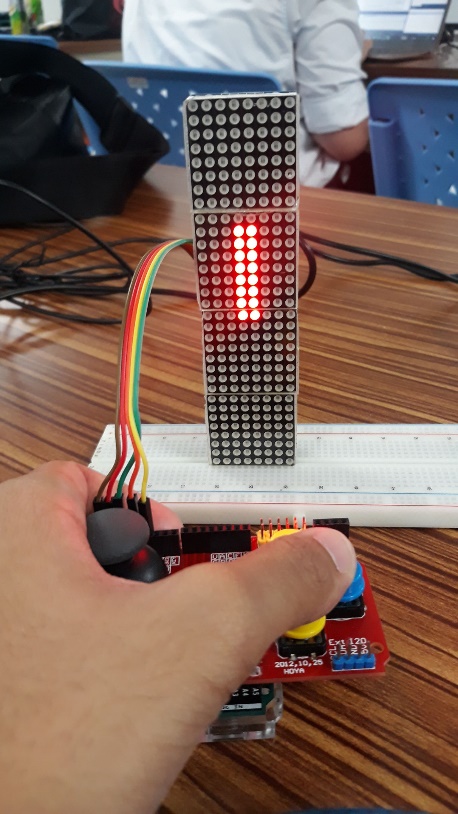
**(A , B , C หรือ D) เพื่อเล่นเกม**

**A picture containing indoor, table, person

Description automatically generated**

**A picture containing electronics

Description automatically generated A picture containing electronics

Description automatically generated **

**หลังจากที่เรากดปุ่มเพื่อเข้าเกมแล้ว หน้าจอก็จะแสดง Interface ที่เป็นตัวเลข 3 , 2 และ 1 เพื่อให้ทราบว่า**

**เกมที่เราจะเล่น กำลังจะเริ่มใน 3 วินาที**

**3**

**2**

**1**

**…**

**GAME START!**

**A picture containing indoor, table, computer

Description automatically generated**

**PART OF GAME**

**1.ยานผู้เล่น - ยานผู้เล่นจะอยู่ข้างล่างสุดของตัวจอ**

**ผู้เล่นสามารถขยับยานได้ 4 ทิศ (ขึ้น,ลง,ซ้าย,ขวา)**

**โดยกดโยกปุ่มอนาล๊อค (Analog)**

**2.ยานศัตรู - ยานศัตรูจะอยู่ด้านบนสุดของตัวจอ**

**ยานของศัตรูจะเคลื่อนที่อัตโนมัติแบบสุ่ม**

**3.พลังชีวิต – พลังชีวิตจะอยู่ด้านซ้ายและขวาของตัวจอ**

**ด้านซ้ายของจอ – พลังชีวิตของผู้เล่นจำนวน 16 ชีวิต**

**ด้านขวาของจอ – พลังชีวิตของศัตรูจำนวน 16 ชีวิต**

**หมายเหตุ : พลังชีวิตจะลดลงทีละ 1 ต่อกระสุน 1 นัด**

**4.ผลลัพธ์ของเกม – ชนะ หรือ แพ้**

**ถ้าพลังชีวิตของศัตรูหมดก่อน ผู้เล่นก็จะ ชนะ**

**ถ้าพลังชีวิตของผู้เล่นหมดก่อน ผู้เล่นก็จะ แพ้**

**A picture containing indoor, table, sitting, floor

Description automatically generated A picture containing table

Description automatically generated**

**MAIN PROCESS OF PROJECT**

**ก่อนที่จะเริ่ม coding นั้น เราจะใช้ lib LedControl.h เข้ามาช่วย เพื่อให้ประหยัดพื้นที่ในการเขียน code และเรียกใช้ฟังก์ชั่นอื่นๆได้อีกด้วย เช่น**

**lc.setLed(); lc.setRow(); lc.setCol(); เป็นต้น**

**A close up of a piece of paper

Description automatically generated**

**ในรูปเราจะกำหนดให้ Din = 11 / CLK = 13 / CS = 10**

**และเราจำควบคุมโมดูลทั้งหมด 4 ตัวด้วยกัน**

**หมายเหตุ :**

**8x8 LED MATRIX = 1 module**

**8x32 LED MATRIX = 4 modules**

**A screenshot of a social media post

Description automatically generated**

**ฟังก์ชั่นการทำงานหลักของเราก็จะเป็นดังรูปนี้**

**รายละเอียดการทำงานของแต่ละฟังก์ชั่นจะอยู่ด้านล่าง**

**A screenshot of a social media post

Description automatically generated**

**Input\_system();**

**คือฟังก์ชั่นที่บ่งบอกว่าเราจะกำหนดค่าปุ่มไว้ที่เท่าไหร่ ใช้งานอย่างไร ให้ไปทิศทางไหน และฟังก์ชั่นนี้ก็จะใช้ map(); กับค่า Analog หรือปุ่มโยกเพื่อให้การเคลื่อนที่ง่ายและตรงกับการเคลื่อนย้ายพิกัดใน LED MATRIX 8x32**

**เช่น เราต้องการให้ยานเราเคลื่อนที่ไปทางซ้าย เราก็จะเปลี่ยนค่าจากช่วง 0-1023 ให้เป็น -1(ซ้าย) และ 1(ขวา) เพื่อที่จะได้เข้าใจง่ายมากขึ้น พอเรากำหนดค่าให้ปุ่มต่างๆแล้ว เราก็จะมาเข้าสู่การทำงานของเกมกันเลยครับ // ส่วนของ input**

**ต่อมาเราก็จะใช้ Switch case เนื่องจากตัวเกมของเราแบ่งได้เป็น 3 ช่วงหลักๆ คือ**

**1.ช่วงแรก**

**A screenshot of a cell phone

Description automatically generated**

**เราจะเข้า case แรกของตัวโปรแกรมนั่นก็คือ main state ซึ่งเรากำหนดไว้ตั้งแต่แรกแล้วว่า main state = 0 A picture containing object

Description automatically generated**

**ซึ่งมาถึงเราก็จะเจอฟังก์ชั่น 1.1 Setup\_interface(); ก่อน**

**A close up of text on a white background

Description automatically generated**

**ซึ่งฟังก์ชั่นนี้จะแสดง Interface PRESS BT. ตอนแรกของตัวโปรแกรมนั่นเอง ซึ่งจะมีการใช้ฟังก์ชั่น plot(,); มาเกี่ยวข้องกับฟังกืชั่นนี้อีกด้วย**

**A close up of text on a white background

Description automatically generated**

**1.2 clear\_display(); คือฟังก์ชั่นที่สั่งให้เคลีย address ใน module ตัวนั้นๆ ซึ่งเราจะสั่งให้เคลีย 4 module ไปก่อน**

**A close up of text on a white surface

Description automatically generated**

**1.3.1 plot(int x,int y); คือฟังก์ชั่นที่กำหนดพิกัดในการแสดงตำแหน่งของไฟในตัว LED MATRIX**

**1.3.2 delete\_plot(int x,int y); คือฟังก์ชั่นที่กำหนดพิกัดในการลบตำแหน่งของไฟในตัว LED MATRIX**

**\*\*\*เหตุผลที่สร้างฟังก์ชั่นนึ้ขึ้นมา\*\*\***

* **เนื่องจากการจ่ายไฟของ LED MATRIX 8x32 นั้น จะจ่ายจากช่อง 0-7 ในแกน x และ ช่อง 0-7 ในแกน y ของแต่ละโมดูล**

**เพื่อไม่ให้สับสนในการกำหนดพิกัด จึงสร้างฟังก์ชั่นที่**

**สามารถป้อนค่า 1-8 ได้ในแกน x และ plot จุดไฟ**

**สามารถป้อนค่า 1-32 ได้ในแกน y และ plot จุดไฟ**

* **และปกติ LED MATRIX จะนับตั้งแต่ล่างขึ้นข้างบนและนับไม่ติดกันในแกน y คือ 0-7 (4รอบ) เราจึงเลยสร้างฟังก์ชั่นนี้ขึ้นมาพร้อมกับให้เริ่มนับจากข้างบนซ้ายเป็นพิกัด 1,1 จนถึง ด้านล่างขวา 8,32 เพื่อง่ายต่อการเขียนโปรแกรม**

**// ฟังก์ชั่นที่กล่าวมาจากข้างต้นเป็นฟังก์ชั่นที่ถูกเรียกใช้ภายในโปรแกรมอยู่ตลอดทั้ง 3 ส่วน**

**เนื่องจากวงจรเราเป็นการต่อแบบ INTERNAL\_PULLUP**

**เราจึงกำหนดให้ว่า ถ้ายังไม่กดปุ่ม (1) ให้แสดง INTERFACE (PRESS BT) ไปเรื่อยๆ จนกว่าจะมีการกดปุ่ม (0)**

**หลังจากกดปุ่มแล้ว ก็จะมี INTERFACE ตัวเลขขึ้นมาให้แสดงว่าอีก 3 วินาทีเกมจะเริ่มขึ้น 1.4 Pre\_interface();**

**A close up of text on a white background

Description automatically generated**

เป็นการใส่ INTERFACE เพิ่มเติมก่อนเริ่มเกม เพื่อนที่จะแสดงว่า เกมกำลังจะเริ่มใน 3…2…1🡪 GAME START!

**หลังจากนี้จะเข้าสู่ตัวเกมของโปรแกรมทันที**

**1.5 Set\_newgame(); คือ ฟังก์ชั่นที่เป็นการเริ่ม Set เกมส์ใหม่ของโปรแกรมซึ่งจะทำงานหลังจาก Pre\_interface();**

**A screenshot of a cell phone

Description automatically generated A screenshot of a social media post

Description automatically generated**

กำหนดฟังก์ชั่นนึ้ขึ้นมาเพื่อแสดงยานผู้เล่น,ยานศัตรู,พลังชีวิตของผู้เล่นและศัตรู/ฟังก์ชั่นสถานะกระสุน

กำหนดฟังก์ชั่นให้วาดและลบตัวยานทั้งสองฝั่งเพื่อเคลื่อนที่แบบไม่ทิ้งรอยเก่า

**A screenshot of a cell phone

Description automatically generated**

กำหนดฟังก์ชั่นพลังชีวิตทั้งสองฝั่ง (มีค่า16ชีวิต) แล้วฝั่งซ้ายเป็นของผู้เล่น ฝั่งขวาเป็นของศัตรู

**หลังจากนั้นเรากำหนดให้ main\_sate = 1 เพื่อที่จะทำให้เราเข้าสู่ case 1 ได้ (ช่วงที่สอง) // จบช่วงแรก**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**2.ช่วงที่สอง คือ ช่วงของตัวเกมหลัก โดยเราจะเจอฟังก์ชั่น Control\_ship(); ก่อน**

**A screenshot of a cell phone

Description automatically generated**

**2.1 Control\_ship();**

**A screenshot of a cell phone

Description automatically generated**

เราจะกำหนดให้ปุ่มของเราเคลื่อนที่โดยการโยกปุ่ม และโยกได้ 4 ทิศ และจำกัดขอบเขตให้กับยานด้วย

**2.2 Control\_enemy();**

**A screenshot of a cell phone

Description automatically generated**

เราจะกำหนดให้ยานของศัตรูเคลื่อนที่อัตโนมัติแบบสุ่ม และจำกัดขอบเขตให้กับยานด้วย

**2.3 Check\_bullet();**

**A screenshot of a cell phone

Description automatically generated**

เราจะทำการเช็คกระสุนว่ากระสุนจะเคลื่อนที่ในรูปแบบไหน แล้วเราจะกำหนดขนาดกระสุนด้วย (ถ้าปุ่มขวากด ปืนถึงจะยิงกระสุน)

ถ้ากระสุนเรายิงไม่โดนอะไรเลย ศัตรูก็จะเลือดไม่ลดนั่นเอง

**2.4 Check\_enemybullet();**

กระสุนฝ่ายศัตรูจะถูกยิงโดยอัตโนมัติเพราะเราใช้ฟังก์ชั่น random(); ค่ากระสุนนั่นเองและกำจัดจำนวนกระสุนให้กับยานศัตรูด้วย

ถ้าศัตรูยิงเราไม่โดน เลือดของยานเราก็จะไม่ลดนั่นเอง

**A screenshot of a cell phone

Description automatically generated**

**2.5 Check\_collision();**

**A screenshot of a cell phone

Description automatically generated**

ฟังก์ชั่นนี้จะเป็นการเช็คตำแหน่งของกระสุนที่ชนกัน ก่อนอื่นเราต้องกำหนดให้ระยะกระสุนในแต่ละฝั่งตันที่ y=30 ไม่ฉะนั้นกระสุนจะเลยออกนอกเจอไปเรื่อยๆ

ถ้ากระสุนของผู้เล่นและกระสุนของศัตรูมาชนกัน จะทำให้กระสุน ณ ตำแหน่งที่ชนกันไฟดับ กล่าวคือเป็นการชนแล้วหายนั่นเอง

ทั้งสองอย่างนี้ เมื่อเป็นจริงอันใดอันหนึ่งให้ set สถานะของกระสุนกลับไปพร้อมยิงอีกครั้ง

**หลังจากฟังก์ชั่นที่กล่าวไปข้างต้น จะเป็นเงื่อนไขพลังชีวิตของผู้เล่นและศัตรูนั่นเอง**

**A screenshot of a cell phone

Description automatically generated**

เงื่อนไขนี้กล่าวคือ ถ้า HP หรือเลือดฝั่งใดฝั่งหนึ่งเป็น 24 (ก็คือเลือดหมดนั่นเอง) จะให้ล้างจอแล้วไปเข้า main\_state = 2 เลย เพื่อเข้า case 2 หรือ ช่วงที่3ของตัวโปรแกรมครับ

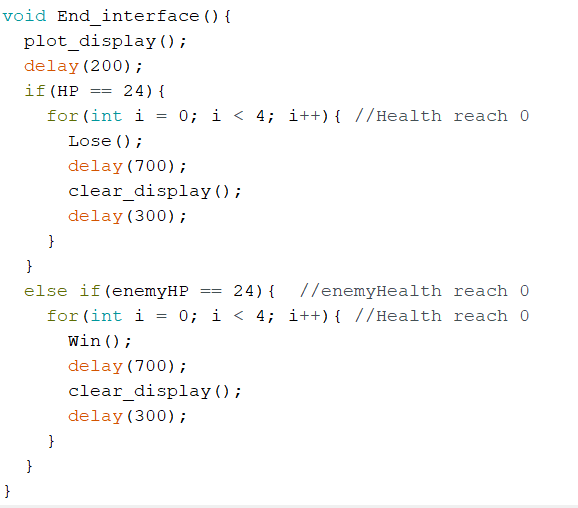
**// จบช่วงที่สอง**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**3.ช่วงที่สาม(สุดท้าย) คือ ช่วงที่แสดง Interface ว่าเราชนะหรือแพ้(Win or Loss)**

**A screenshot of a cell phone

Description automatically generated**

****

**3.1 End\_interface(); - ฟังก์ชั่นนี้จะแสดง**

**3.1.1 win(); 3.1.2 lose():**

**A close up of a newspaper

Description automatically generatedA screenshot of text

Description automatically generatedEND OF PROJECT**

ฟังก์ชั่นช่วงสุดท้ายจะเป็นการแสดง interface ว่าเราชนะหรือแพ้ โดย

ถ้าเลือดผู้เล่นหมด (HP = 24 หมายถึง ถ้าลบไฟจนถึง y = 24) จอภาพก็จะแสดงคำว่า LOSS

ถ้าเลือดศัตรูหมด (enemyHP = 24 หมายถึง ถ้าลบไฟจนถึง y = 24 ) จอภาพก้จะแสดงคำว่า WIN

เมื่อแสดงค่า WIN || LOSS เสร็จแล้ว ก็จะส่งค่า main\_state = 0 เพื่อกลับไปหน้าหลักของเกม