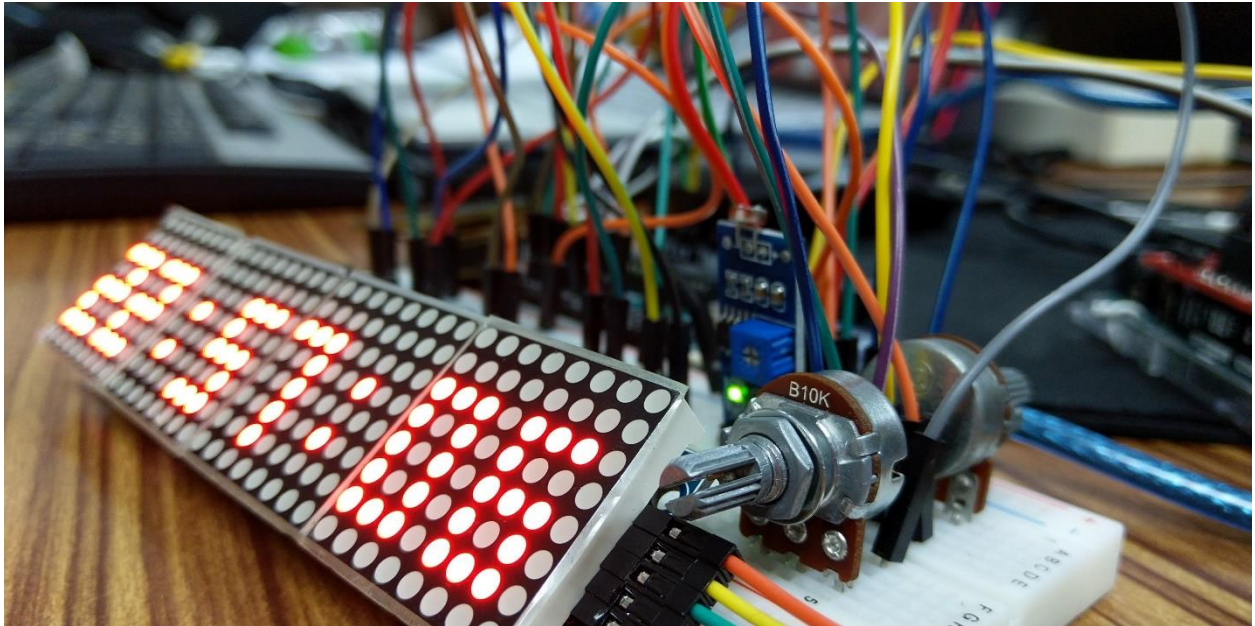
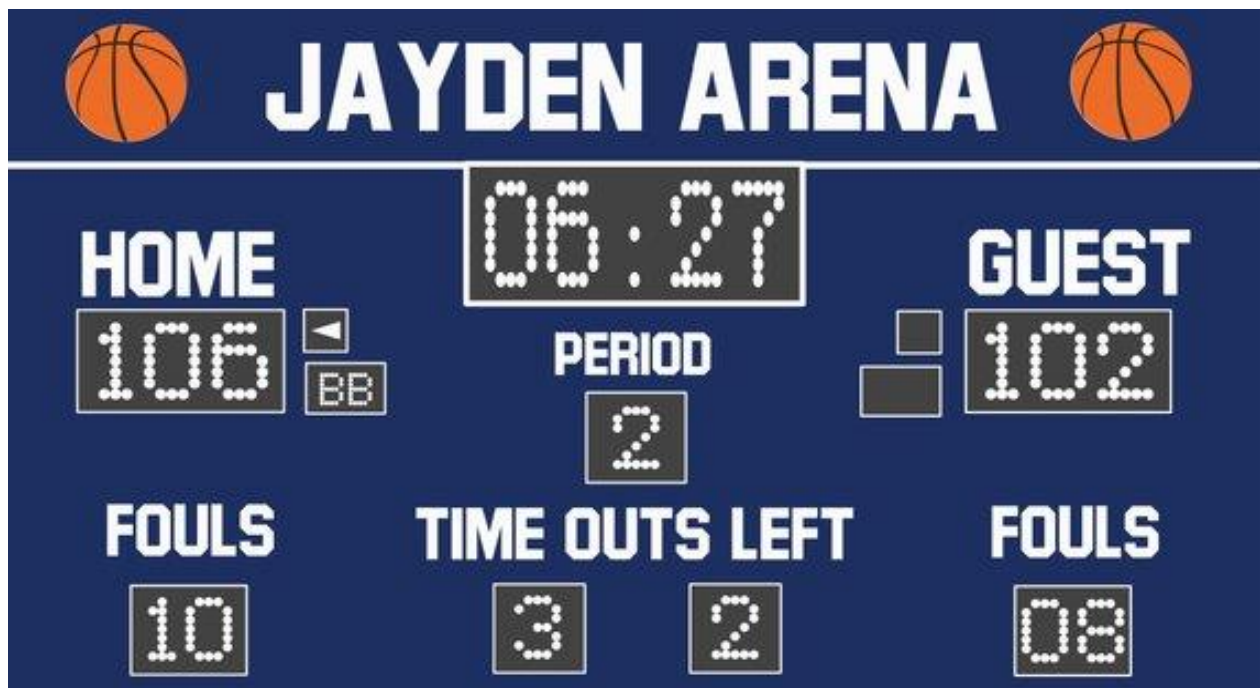


Project # 1 : mini clock



1.แนวคิดการออกแบบ (Conceptual Design)

การออกแบบได้แนวคิดมาจากนาฬิกาจับเวลาของการแข่งขันกีฬาบาสเกตบอล ซึ่งนาฬิกาที่ใช้ในกีฬาบาสเกตบอลนั้น จะมีฟังก์ชันที่หลากหลาย เช่น การนับรอบ ควอเตอร์ การนับการฟาวล์ เมื่อมีการเล่นผิดพลาด การนับเวลา และการถอยหลัง เมื่อมีการส่งต่อบอล โดยในกติกาบาสต้องไม่อยู่กับที่ หรือต่อบอลไว้ไม่เกินคนละ 5 วินาที ซึ่งสำหรับงานชิ้นนี้ จะยกตัวอย่างมาบางส่วนเช่น เวลาที่ใช้นับปกติ การจับเวลา การนับถอยหลัง และการตั้งนาฬิกาปลุก

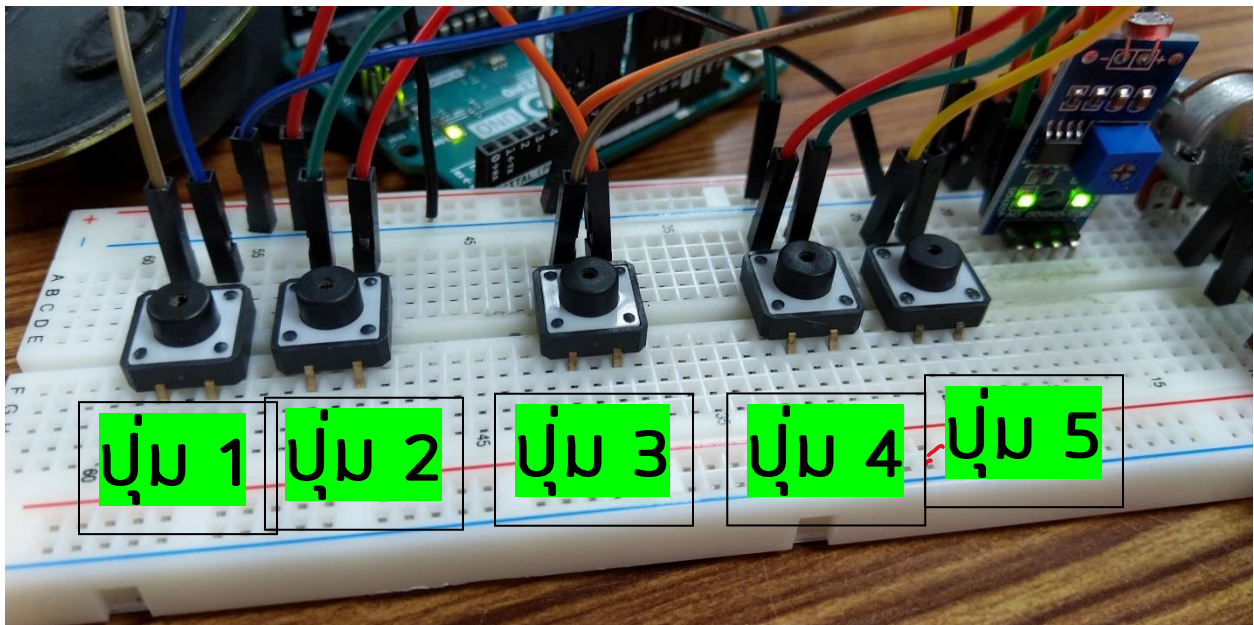


ซึ่งในงานชิ้นนี้จะมีการใช้อุปกรณ์อย่างอื่นเพื่อมาใช้ควบคุมด้วยอาทิ

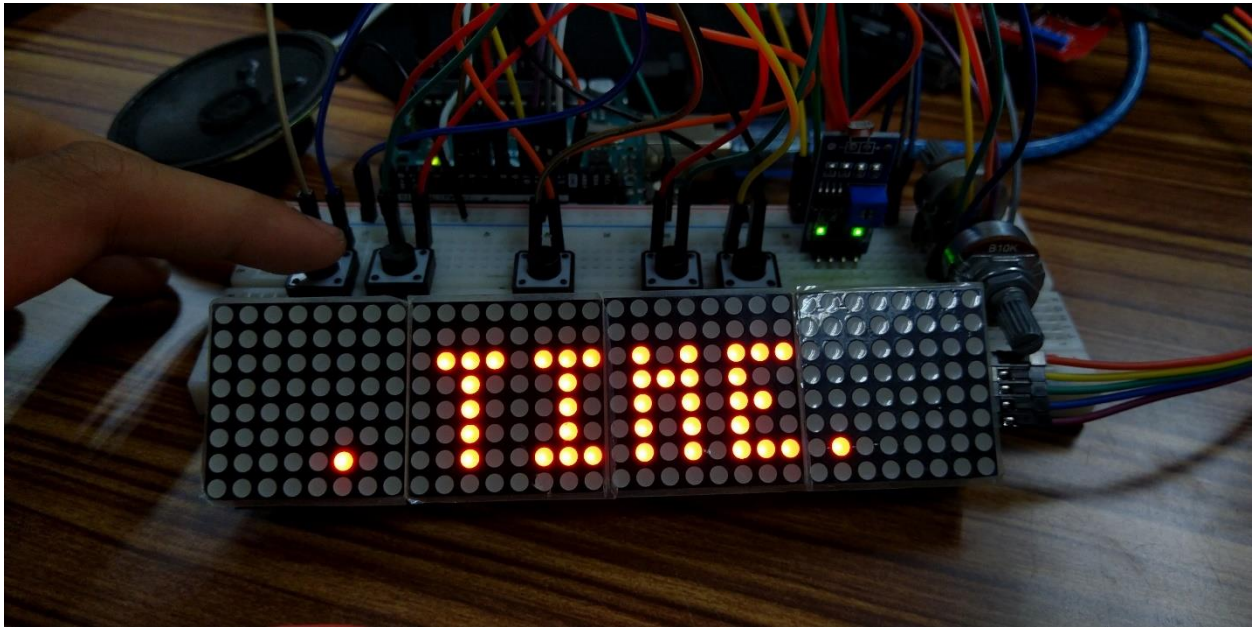
- ตัวต้านทานแบบปรับค่าได้ จำนวน 2 ตัว
- ปุ่ม TACT SWITCH จำนวน 5 ปุ่ม
- เซนเซอร์วัดค่าแสง LDR (ต่อ ANALOG)
- Speaker 1 ตัว

2.การใช้งานโดยย่อ

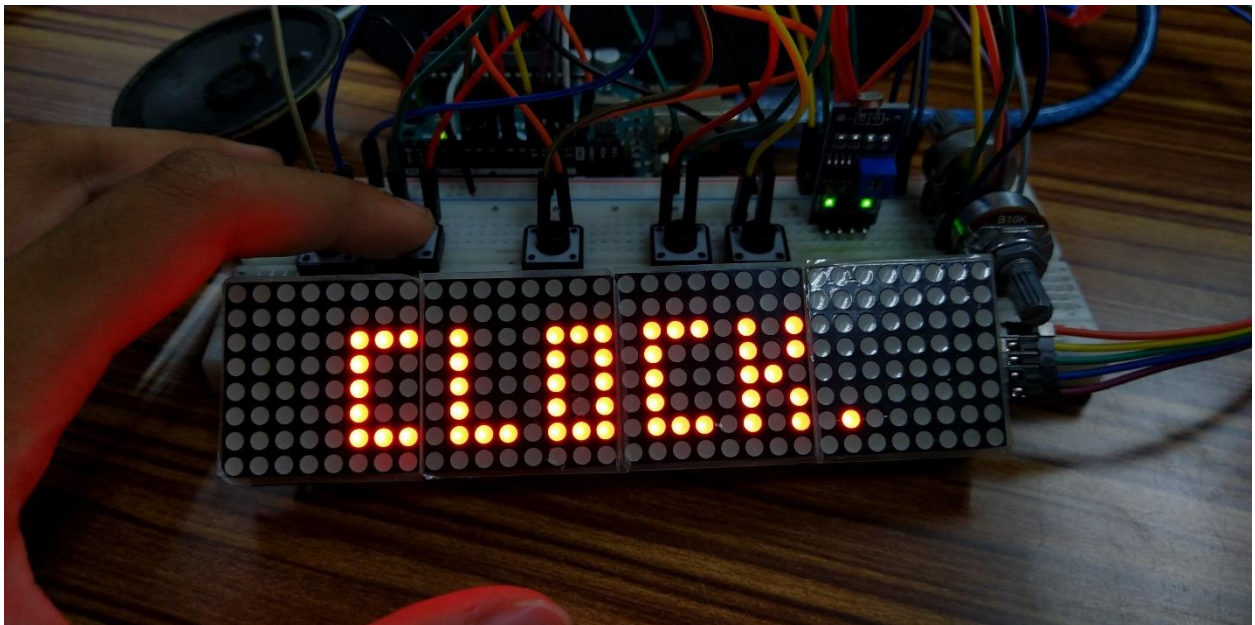
การกดปุ่ม TACT SWITCH



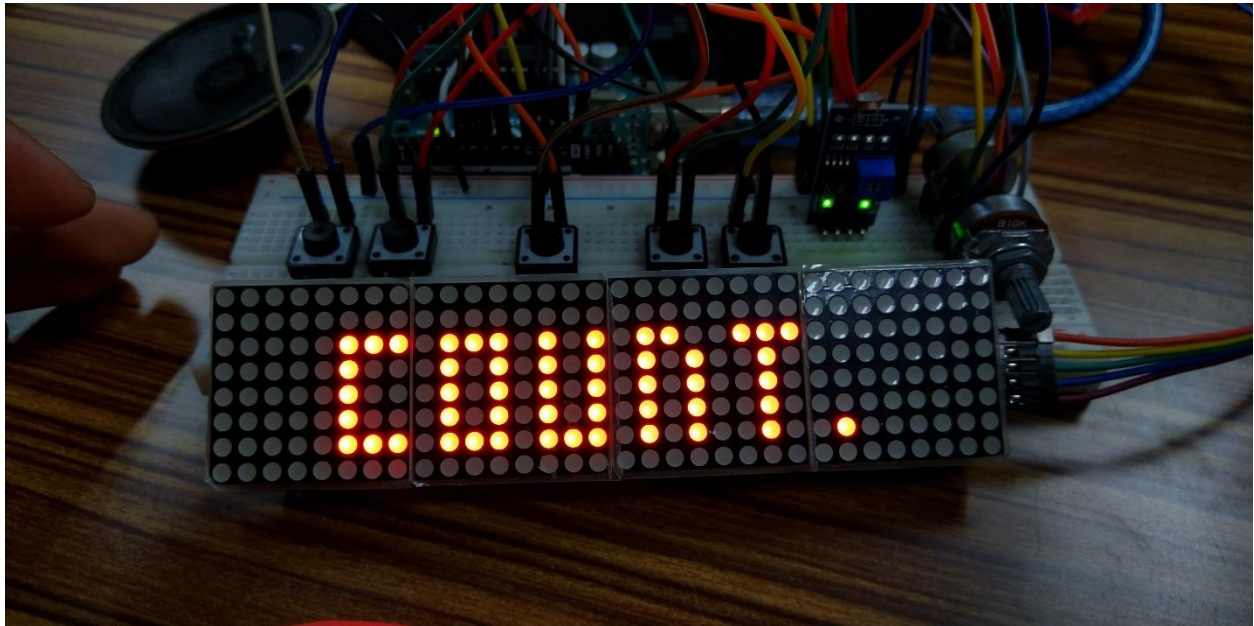
- ปุ่ม 1 จะเป็นการไปหน้าเวลาหลัก



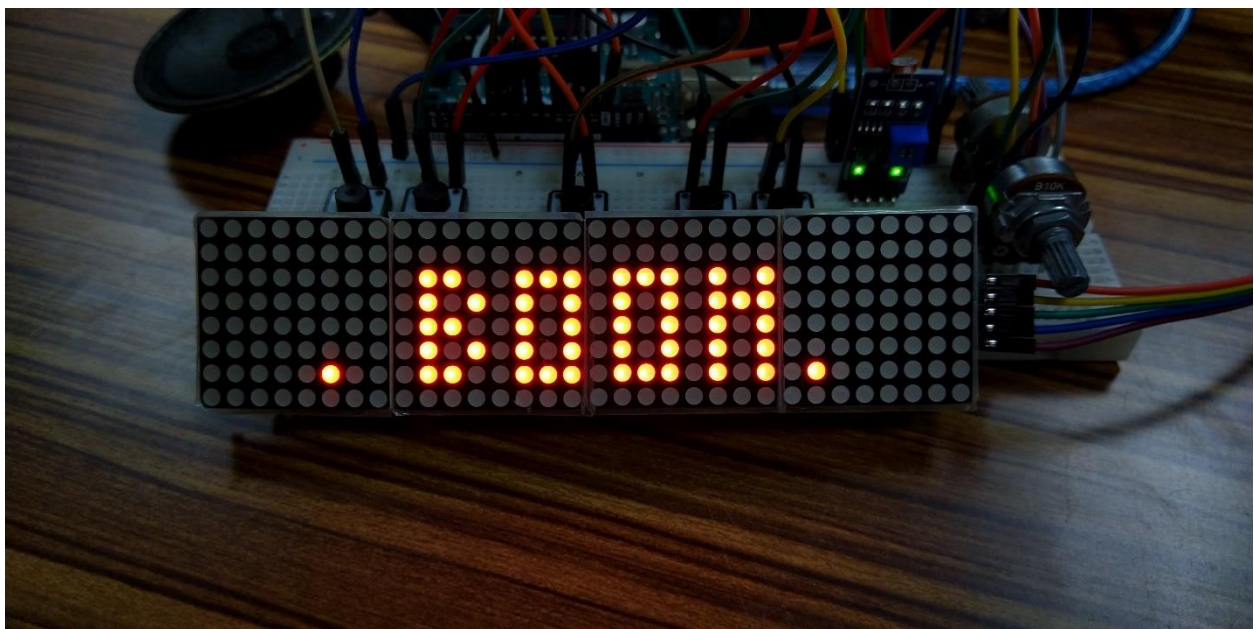
- ปุ่ม 2 จะเป็นการไปหน้าการตั้งปลุก



- ปุ่ม 3 จะเป็นการไปหน้าการจับเวลา

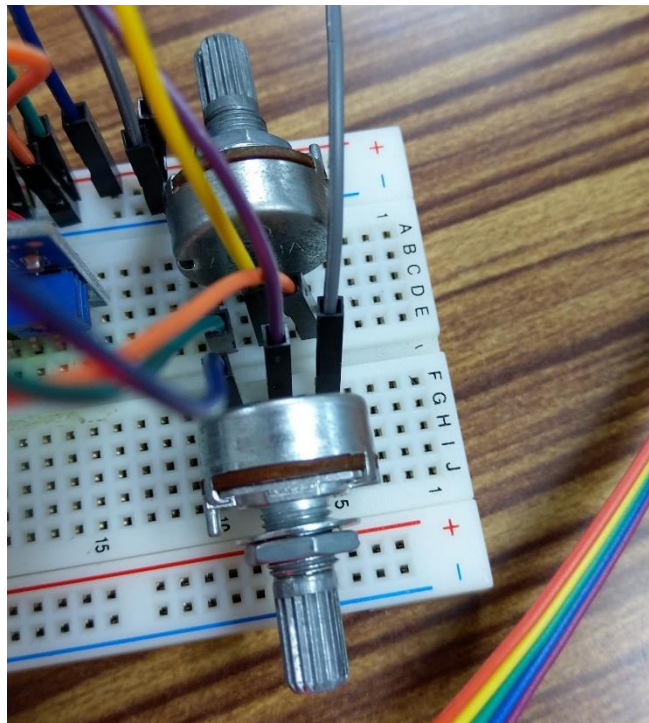


- ปุ่ม 4 จะเป็นการไปหน้าการนับถอยหลัง



- ปุ่ม 5 จะเป็นปุ่มกำหนดค่าเวลา โดยมีเงื่อนไขว่า ต้องกดไปที่หน้ารูปแบบเวลาที่ต้องการกำหนดก่อน แล้วจึงค่อยกดปุ่มที่ 5 ซึ่งสามารถปรับค่าได้โดยการหมุนตัวต้านค่าแบบปรับค่าได้ทั้ง 2 ตัวขวามือ เมื่อตั้งเวลาเสร็จจึงกดปุ่ม 5 ซ้ำอีกทีนึง การใช้ตัวต้านทานแบบปรับค่าได้

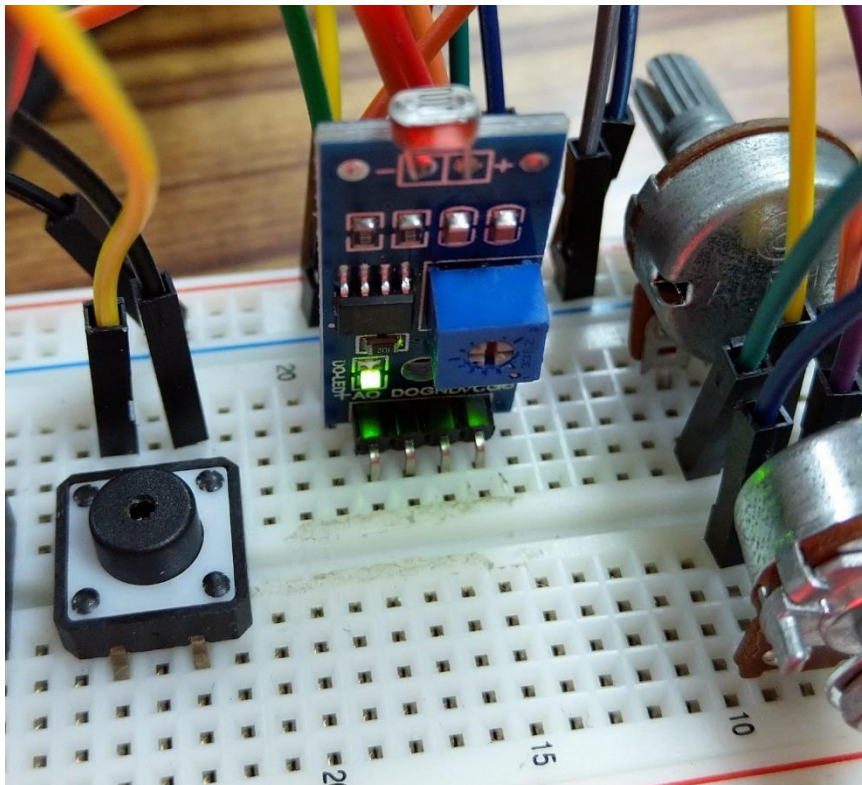
- ตัวคั่นบน ใช้รับ ชั่วโมง
- ตัวคั่นล่าง ใช้รับ นาที



โดย 2 ปุ่มนี้จะใช้ได้ก็ต่อเมื่อกดปุ่ม 5 ในหน้าเวลานั้นๆ แล้วเท่านั้น

การใช้เซนเซอร์วัดค่าแสง LDR

- ใช้ได้เมื่อนาฬิกาปลุกดัง แล้วให้เอามือไปปิดตรงเซนเซอร์ เสียงจะดับลง (หมายเหตุ จำลองสภาพการนอนของเจ้าของผลงาน ที่นอนกลางวันส่วนใหญ่ และ กลางคืนทำงาน ดังนั้นเมื่อฟ้าเริ่มมืด และได้ตั้งนาฬิกาปลุกไว้ จะไม่เกิดเสียงขึ้นนั่นเอง แต่ ถ้าเข้านอน ตั้งนาฬิกาปลุกไว้ ถ้านาฬิกาปลุกก็ต้องตื่นมาปิดเซนเซอร์ เพื่อไม่ให้รบกวนห้อง ข้างๆ)



3.โปรแกรมและการอธิบายโปรแกรมโดยย่อ (อธิบายในระดับฟังก์ชัน)

- ประกาศใช้ ไลบรารีและประกาศตัวแปร ที่ใช้ในการเก็บค่าเวลาของโหมดเวลาต่าง เช่น sec แทน วินาที mi แทน นาที hour แทน ชั่วโมง เป็นต้น

```
1 #include "LedControl.h"
2 #include "FontLEDClock.h"
3
4 LedControl lc = LedControl(11, 13, 12, 4 );
5 int timer1_counter;
6 uint8_t intensity = 1;
7
8 int page = 3;
9
10 bool stateSet = false;
11 bool stateUp = false;
12 bool stateDown = false;
13 bool stateSoundBeep = false;
14
15 int sec = 50, mi = 10, hour = 23; // เวลาหลัก
16 int sec1 = 00, mi1 = 11, hour1 = 23; // นาฬิกาปลุก
17 int sec2 = 0, mi2 = 0, hour2 = 0; // ตัวจับเวลา
18 int sec3 = 0, mi3 = 0, hour3 = 0; // พับถอยหลัง
19
20
```

- ตั้งค่า PinMode ต่างๆ และกำหนดการ Interrupts + ค่าของ sec

```
21 void setup()
22 {
23     pinMode(A5, INPUT);
24     pinMode(A1, INPUT);
25     pinMode(A0, INPUT);
26
27     pinMode(4, OUTPUT);
28
29     pinMode(5, INPUT_PULLUP);
30     pinMode(6, INPUT_PULLUP);
31     pinMode(7, INPUT_PULLUP);
32     pinMode(8, INPUT_PULLUP);
33     pinMode(10, INPUT_PULLUP);
34
35     Serial.begin(9600);
36
37     // initialize timer1
38     noInterrupts(); // disable all interrupts
39     TCCR1A = 0;
40     TCCR1B = 0;
41     timer1_counter = 3036;
42     TCNT1 = timer1_counter; TCCR1B |= (1 << CS12); TIMSK1 |= (1 << TOIE1);
43     interrupts();
44
45     int devices = lc.getDeviceCount(); // find no of LED modules
46     //we have to init all devices in a loop
47     for (int address = 0; address < devices; address++) { // set up each device
48         lc.shutdown(address, false);
49         lc.setIntensity(address, 8);
50         lc.clearDisplay(address);
51     }
52 }
```

- ประกาศใช้ฟังก์ชัน Time1 และกำหนดเงื่อนไขของ Time2 ที่ต้องใช้ตัวแปร sec1,mi1,hour1 ถ้าเวลาหลัก(ทั้งวินาที,นาท้,ชั่วโมง)เท่ากับเวลาของนาฬิกาปลุกที่ได้ตั้งค่าไว้โดยการกดปุ่ม 2 และกดปุ่ม 5 ตามลำดับ ถ้าเป็นจริงให้ส่งเสียง ต่อมา กำหนดเงื่อนไข ถ้า stateUp มีการกดให้จับเวลาไปข้างหน้า เช่นเดียวกัน ถ้า stateDown มีการกดให้นับเวลาลดลง ซึ่งทั้ง 2 ฟังก์ชันนี้ จะถูกใช้ในการกดปุ่ม 4 และ ปุ่ม 3 ตามลำดับ

```

54 ISR(TIMER1_OVF_vect) // interrupt service routine
55 {
56     TCNT1 = timer1_counter; // preload timer
57     //Serial.println("Interrupt Time :" + (String)(++sec1));
58     ++sec;
59     TIME1(); // ตัวนับเวลาปกติ
60     // TIME2(); // ตัวตั้งนาฬิกาปลุก
61     if (sec == sec1 && mi == mi1 && hour == hour1)
62     {
63         stateSoundBeep = true;
64         Serial.println("On");
65     }
66     if (stateUp == true) // จับเวลาไปข้างหน้า
67     {
68         sec2++;
69         TIME3();
70     }
71     if (stateDown == true) // นับถอยหลัง
72     {
73         sec3--;
74         TIME4();
75     }
76 }
77 }
78
79 String Time;
80 int p = 0;

```

- ในส่วนนี้แรกเริ่มเป็นการกำหนดตัวแปรที่นำมาเก็บค่าของ เซนเซอร์วัดค่าแสง LDR ที่กำหนดไว้ว่า ถ้าอ่านค่าจากตัวต้านทานได้ 1000 ค่าต่ำสุดที่จะนำไปใช้ของ LDR จะเป็น 0 ซึ่งค่าที่ LDR อ่านได้จะแปรผกผันกับค่าจากตัวต้านทานแบบปรับค่าได้
- กำหนดเงื่อนไขหาก stateSoundBeep เป็นจริง ให้ส่งเสียง
- ถ้า ค่าที่อ่านมาจากการ map ค่า ของตัวแปร Mlight น้อยกว่า 5 ก็จะสั่งให้ stateSoundBeep เป็นเท็จ และให้หยุดเล่นเสียงทันที

```

81 void loop()
82 {
83     float light = analogRead(A1);
84     int Mlight = map(light, 1000, 0, 1, 8);
85     // for (int i = 0; i < 4; i++)
86     // {
87     //     lc.setIntensity(i, Mlight);
88     // }
89     Serial.println(Mlight);
90
91     if (stateSoundBeep == true) // สั่งให้เสียงดัง
92     {
93         noTone(4);
94         music();
95         digitalWrite(4, HIGH);
96         Serial.println("ONNNNNNNNNNNNNN");
97     }
98     //สั่งให้เสียงเงียบ
99     if (Mlight < 5)
100     {
101         noTone(4);
102         stateSoundBeep = false;
103         Serial.println("OFFFFFFFFF");
104     }
105

```


- ในส่วนนี้การอธิบายจะอยู่ในรูป และในการกดปุ่มแต่ละปุ่มจะมีฟังก์ชันขึ้นมา

```
107 if (digitalRead(5) == LOW) // ตรวจสอบการกดปุ่มแรก ถ้าใช้เปลี่ยนเป็น หน้าที 0
108 {
109     clear_display();
110     printsting(" .TIME. ");
111     delay(500);
112     clear_display();
113     page = 0;
114     stateSet = false;
115 }
116 if (digitalRead(6) == LOW) // ตรวจสอบการกดปุ่มแรก ถ้าใช้เปลี่ยนเป็น หน้าที 1
117 {
118     clear_display();
119     printsting(" CLOCK. ");
120     delay(500);
121     clear_display();
122     page = 1;
123     stateSet = false;
124 }
125 if (digitalRead(7) == LOW) // ตรวจสอบการกดปุ่มแรก ถ้าใช้เปลี่ยนเป็น หน้าที 2
126 {
127     clear_display();
128     printsting(" COUNT.");
129     delay(500);
130     clear_display();
131     page = 2;
132     stateSet = false;
133 }
134 if (digitalRead(8) == LOW) // ตรวจสอบการกดปุ่มแรก ถ้าใช้เปลี่ยนเป็น หน้าที 3
135 {
136     clear_display();
137     printsting(" .BOOM. ");
138     delay(500);
139     clear_display();
140     page = 3;
141     stateSet = false;
142 }
143 if (digitalRead(10) == LOW) // ปุ่มสุดท้าย
144 {
145     mil++;
146     if (stateSoundBeep == true) //ถ้ากดปุ่มสุดท้าย ถ้าเสียงดังอยู่ให้ปิด
147     {
148         stateSoundBeep = false;
149         Serial.println("OFF");
150     }
151
152     if (page == 2) //ถ้าอยู่ในหน้า page 2 ถ้า state เป็น ture จะ set เป็น false
153     {
154         clear_display();
155         printsting("1 2 3 4"); delay(1000);
156         printsting(" GO "); delay(1000); music1(); noTone(4);
157         clear_display();
158         stateUp = !stateUp;
159     }
```

- ในส่วนนี้การอธิบายจะอยู่ในรูป

```
160 else if (page == 3) //ถ้าอยู่ในหน้า page 3 ถ้า stateSet เป็น true หรือตั้งเวลาอยู่ เวลาจะไม่นับถอยหลังจนกว่าจะกดปุ่มซ้ำอีกที
161 {
162     stateSet = !stateSet;
163     if (stateSet == true) // TRUE
164     {
165         stateDown = false;
166     }
167     else // False
168     {
169         clear_display();
170         printstring("4 3 2 1"); delay(1000);
171         printstring(" GO "); delay(1000); music1();noTone(4);
172         clear_display();
173         stateDown = true;
174     }
175 }
176 else
177 {
178     stateSet = !stateSet;
179     music1(); noTone(4);
180 }
181 // settime();
182 clear_display(); //ทุกครั้งที่เกิดปุ่ม จะเคลียร์หน้าจอ
183 }
184 // Serial.println(map(analogRead(A0),0,1000,0,23));
185 // Serial.println(map(analogRead(A5), 0, 1000, 0, 23));
```

```
186 switch (page) //เป็นลำดับบอกที่อยู่หน้าที่เท่าไร
187 {
188     case 0: // แสดงเวลา ของเวลาหลัก
189         Time = addZero(hour) + ":" + addZero(mi) + ":" + addZero(sec) ;
190         p = 0;
191         while (Time[p])
192         {
193             print_tiny_char((p * 4), 2, Time[p]); p++;
194
195             // settime
196             if (stateSet == true) //สามารถตั้งเวลาได้
197             {
198                 hour = (map(analogRead(A0), 0, 1000, 0, 23));
199                 mi = (map(analogRead(A5), 0, 1000, 0, 59));
200             }
201         }
202         break;
```

```
204 case 1: // แสดงเวลา ของนาฬิกาปลุก
205     Time = addZero(hour1) + ":" + addZero(mi1) + ":" + addZero(sec1) ;
206     p = 0;
207     while (Time[p]) {
208         print_tiny_char((p * 4), 2, Time[p]); p++;
209     }
210     if (stateSet == true) //สามารถตั้งเวลาได้
211     {
212         hour1 = (map(analogRead(A0), 0, 1000, 0, 23));
213         mi1 = (map(analogRead(A5), 0, 1000, 0, 59));
214     }
215     break;
216
217 case 2: // แสดงเวลา ของการจับเวลา ที่ไม่สามารถตั้งเวลาได้
218     Time = addZero(hour2) + ":" + addZero(mi2) + ":" + addZero(sec2) ;
219     p = 0;
220     while (Time[p])
221     {
222         print_tiny_char((p * 4), 2, Time[p]); p++;
223     }
224     break;
225
226 case 3: //แสดงเวลา ของการนับถอยหลัง
227     Time = addZero(hour3) + ":" + addZero(mi3) + ":" + addZero(sec3) ;
228     p = 0;
229     while (Time[p]) {
230         print_tiny_char((p * 4), 2, Time[p]); p++;
231     }
232     if (stateSet == true) //สามารถตั้งเวลาได้
233     {
234         hour3 = (map(analogRead(A0), 0, 1000, 0, 23));
235         mi3 = (map(analogRead(A5), 0, 1000, 0, 59));
236     }
237     break;
```


- ส่วนต่อมาจะเป็นฟังก์ชันต่างๆ

```
244 ///////////////////////////////////////////////////////////////////
245 void plot (uint8_t x, uint8_t y, uint8_t value) {
246
247     //select which matrix depending on the x coord
248     uint8_t address;
249     if (x >= 0 && x <= 7)
250     {
251         address = 3;
252     }
253     if (x >= 8 && x <= 15)
254     {
255         address = 2;
256         x = x - 8;
257     }
258     if (x >= 16 && x <= 23)
259     {
260         address = 1;
261         x = x - 16;
262     }
263     if (x >= 24 && x <= 31)
264     {
265         address = 0;
266         x = x - 24;
267     }
268
269     if (value == 1) {
270         lc.setLed(address, y, x, true);
271     } else {
272         lc.setLed(address, y, x, false);
273     }
274 }
```

ฟังก์ชัน plot() จะเป็นการแสดงตัวอักษรผ่านทางหน้าจอ โดยใช้ตัวแปร address
ชี้ address แต่ละ address เป็นการบอกว่าอยู่ในเมทริกซ์ตัวไหน

- ตรวจจับ TIME1 , TIME2, TIME3 และ TIME4

```
276 void TIME1() ////////////////////////////////////// ปกติ
277 {
278     if (sec > 59)
279     {
280         sec = 0;
281         mi++;
282     }
283     if (mi > 59)
284     {
285         mi = 0;
286         hour++;
287     }
288     if (hour > 23)
289     {
290         hour = 0;
291     }
292 }
293
294 void TIME2() ////////////////////////////////////// นาฬิกาปลุก
295 {
296     if (sec1 < 0)
297     {
298         sec1 = 59;
299         mil--;
300     }
301     if (mil < 0)
302     {
303         mil = 59;
304         hour1--;
305     }
306     if (hour1 < 0)
307     {
308         hour1 = 23;
309     }
```

```

326 void TIME3() //////////////////////////////////////// จับเวลา
327 {
328     if (sec2 > 59)
329     {
330         sec2 = 0;
331         mi2++;
332     }
333     if (mi2 > 59)
334     {
335         mi2 = 0;
336         hour2++;
337     }
338     if (hour2 > 59)
339     {
340         hour2 = 0;
341     }
342 }
343 void TIME4() //////////////////////////////////////// นับถอยหลัง
344 {
345     if (sec3 < 0)
346     {
347         sec3 = 59;
348         mi3--;
349     }
350     if (mi3 < 0)
351     {
352         mi3 = 59;
353         hour3--;
354     }
355     if (hour3 < 0)
356     {
357         hour3 = 23;
358     }

```

ในตัวฟังก์ชัน TIME ทั้ง 1,2,3,4 นี้ จะใช้ ตัวแปร sec, mi, hour คนละตัวแปรกันแต่การทำงานเหมือนกันโดยเป็นเงื่อนไขของเวลา

- ฟังก์ชัน เสียง

```
360 void music()  
361 {  
362     for (int h = 0; h < 2; h++)  
363     {  
364         tone(4, 587);  
365         delay(100);  
366         tone(4, 587);  
367         delay(100);  
368         tone(4, 659);  
369         delay(100);  
370         tone(4, 659);  
371         delay(100);  
372         tone(4, 349);  
373         delay(100);  
374         tone(4, 349);  
375         delay(100);  
376         tone(4, 440);  
377         delay(100);  
378         tone(4, 494);  
379         delay(100);  
380         tone(4, 494);  
381         delay(100);  
382         tone(4, 587);  
383         delay(100);  
384         tone(4, 587);  
385         delay(100);  
386         tone(4, 494);  
387         delay(100);  
388     }  
389 }  
390 void music1()  
391 {  
392     for (int i = 0; i < 3; i++)  
393     {  
394         tone(4, 587);  
395         delay(100);  
396         tone(4, 587);  
397         delay(100);  
398     }  
399 }
```

- ฟังก์ชัน การพิมพ์ข้อความบน แมทริกซ์

```
400 void print_tiny_char(uint8_t x, uint8_t y, char c) {
401     uint8_t dots;
402     if (c >= 'A' && c <= 'Z' || (c >= 'a' && c <= 'z') ) {
403         c &= 0x1F; // A-Z maps to 1-26
404     }
405     else if (c >= '0' && c <= '9')
406     {
407         c = (c - '0') + 32;
408     }
409     else if (c == ' ') {
410         c = 0; // space
411     }
412     else if (c == '.') {
413         c = 27; // full stop
414     }
415     else if (c == ':') {
416         c = 28; // colon
417     }
418     else if (c == '\') {
419         c = 29; // single quote mark
420     }
421     else if (c == '!') {
422         c = 30; // single quote mark
423     }
424     else if (c == '?') {
425         c = 31; // single quote mark
426     }
427
428     for (uint8_t col = 0; col < 3; col++) {
429         dots = pgm_read_byte_near(&mytinyfont[c][col]);
430         for (uint8_t row = 0; row < 5; row++) {
431             if (dots & (16 >> row))
432                 plot(x + col, y + row, 1);
433             else
434                 plot(x + col, y + row, 0);
435         }
436     }
437 }
```

- ฟังก์ชันที่ใช้แสดงตัวอักษร

```
438 void printsting(String input) //ใส่ได้ 8 ตัวพอดี
439 {
440     int positionX = 1, positionY = 2;
441     for (int i = 0; i < input.length(); i++)
442     {
443         print_tiny_char(positionX, positionY, input[i]);
444         positionX += 4;
445     }
446 }
447
448 void clear_display()
449 {
450     for (uint8_t address = 0; address < 4; address++)
451     {
452         lc.clearDisplay(address);
453     }
454 }
455
456 String addZero(int s)
457 {
458     String str = "";
459     if (s / 10 == 0)
460     {
461         str = "0" + (String)s;
462     } else str = (String)s;
463     return str;
464 }
```


4. Source Code

```
#include "LedControl.h"
```

```
#include "FontLEDClock.h"
```

```
LedControl lc = LedControl(11, 13, 12, 4 );
```

```
int timer1_counter;
```

```
uint8_t intensity = 1;
```

```
int page = 3;
```

```
bool stateSet = false;
```

```
bool stateUp = false;
```

```
bool stateDown = false;
```

```
bool stateSoundBeep = false;
```

```
int sec = 50, mi = 10, hour = 23; // เวลาหลัก
```

```
int sec1 = 00, mi1 = 11, hour1 = 23; // นาฬิกาปลุก
```

```
int sec2 = 0, mi2 = 0, hour2 = 0; // ตัวจับเวลา
```

```
int sec3 = 0, mi3 = 0, hour3 = 0; // นับถอยหลัง
```

```
void setup()

{

    pinMode(A5, INPUT);

    pinMode(A1, INPUT);

    pinMode(A0, INPUT);


    pinMode(4, OUTPUT);


    pinMode(5, INPUT_PULLUP);

    pinMode(6, INPUT_PULLUP);

    pinMode(7, INPUT_PULLUP);

    pinMode(8, INPUT_PULLUP);

    pinMode(10, INPUT_PULLUP);


    Serial.begin(9600);


    // initialize timer1

    noInterrupts(); // disable all interrupts

    TCCR1A = 0;
```

```
TCCR1B = 0;

timer1_counter = 3036;

TCNT1 = timer1_counter; TCCR1B |= (1 << CS12); TIMSK1 |= (1 << TOIE1);

interrupts();
```

```
int devices = lc.getDeviceCount(); // find no of LED modules
```

```
//we have to init all devices in a loop
```

```
for (int address = 0; address < devices; address++) { // set up each device
```

```
    lc.shutdown(address, false);
```

```
    lc.setIntensity(address, 8);
```

```
    lc.clearDisplay(address);
```

```
}
```

```
}
```

```
ISR(TIMER1_OVF_vect) // interrupt service routine
```

```
{
```

```
    TCNT1 = timer1_counter; // preload timer
```

```
    //Serial.println("Interrupt Time :" + (String)(++sec1));
```

```
    ++sec;
```

```
    TIME1(); // ตัวนับเวลาปกติ
```

```

// TIME2(); // ตัวตั้งนาฬิกาปลุก

if (sec == sec1 && mi == mi1 && hour == hour1)

{

    stateSoundBeep = true;

    Serial.println("On");

}

if (stateUp == true) // จับเวลาไปข้างหน้า

{

    sec2++;

    TIME3();

}

if (stateDown == true) // นับถอยหลัง

{

    sec3--;

    TIME4();

}

}

String Time;

```



```

int p = 0;

void loop()

{

    float light = analogRead(A1);

    int Mlight = map(light, 1000, 0, 1, 8);

    // for (int i = 0; i < 4; i++)

    // {

    //     lc.setIntensity(i, Mlight);

    // }

    Serial.println(Mlight);


    if (stateSoundBeep == true) // สั่งให้เสียงดัง

    {

        noTone(4);

        music();

        digitalWrite(4, HIGH);

        Serial.println("ONNNNNNNNNNNNNNNN");

    }

    //สั่งให้เสียงเงียบ

    if (Mlight < 5)

```

```
{  
  
  noTone(4);  
  
  stateSoundBeep = false;  
  
  Serial.println("OFFFFFFFFF");  
  
}
```

```
delay(200);  
  
if (digitalRead(5) == LOW) // ตรวจสอบการกดปุ่มแรก ถ้าใช่เปลี่ยนเป็น หน้าที 0  
{  
  
  clear_display();  
  
  printsting(" .TIME. ");  
  
  delay(500);  
  
  clear_display();  
  
  page = 0;  
  
  stateSet = false;  
  
}  
  
if (digitalRead(6) == LOW) // ตรวจสอบการกดปุ่มแรก ถ้าใช่เปลี่ยนเป็น หน้าที 1  
{  
  
  clear_display();  
  
  printsting(" CLOCK. ");
```

```

delay(500);

clear_display();

page = 1;

stateSet = false;

}

if (digitalRead(7) == LOW) // ตรวจสอบการกดปุ่มแรก ถ้าใช่เปลี่ยนเป็น หน้าที 2
{

clear_display();

printsting(" COUNT.");

delay(500);

clear_display();

page = 2;

stateSet = false;

}

if (digitalRead(8) == LOW) // ตรวจสอบการกดปุ่มแรก ถ้าใช่เปลี่ยนเป็น หน้าที 3
{

clear_display();

printsting(" .BOOM. ");

delay(500);

clear_display();

```

```

page = 3;

stateSet = false;

}

if (digitalRead(10) == LOW) // ปุ่มสุดท้าย

{

    mi1++;

    if (stateSoundBeep == true) //ถ้ากดปุ่มสุดท้าย ถ้าเสียงดังอยู่ให้ปิด

    {

        stateSoundBeep = false;

        Serial.println("OFF");

    }


if (page == 2) //ถ้าอยู่ในหน้า page 2 ถ้า state เป็น true จะ set เป็น false

{

    clear_display();

    printsting("1 2 3 4"); delay(1000);

    printsting("  GO  "); delay(1000); music1(); noTone(4);

    clear_display();

    stateUp = !stateUp;

}

```


else if (page == 3) //ถ้าอยู่ในหน้า page 3 ถ้า stateSet เป็น true หรือตั้งเวลาอยู่
เวลาจะไม่นับถอยหลังจนกว่าจะกดปุ่มซ้ำอีกทีหนึ่ง

```
{  
  
    stateSet = !stateSet;  
  
    if (stateSet == true) // TRUE  
  
    {  
  
        stateDown = false;  
  
    }  
  
    else // False  
  
    {  
  
        clear_display();  
  
        printsting("4 3 2 1"); delay(1000);  
  
        printsting(" GO "); delay(1000); music1();noTone(4);  
  
        clear_display();  
  
        stateDown = true;  
  
    }  
  
}  
  
else  
  
{  
  
    stateSet = !stateSet;
```

```

    music1(); noTone(4);

}

//  settime();

clear_display(); //ทุกครั้งทีกดปุ่ม จะเคลียร์หน้าจอ

}

// Serial.println(map(analogRead(A0),0,1000,0,23));

// Serial.println(map(analogRead(A5), 0, 1000, 0, 23));

switch (page) //เป็นตัวบอกว่าอยู่หน้าที่เท่าไร

{

    case 0: // แสดงเวลา ของเวลาหลัก

        Time = addZero(hour) + ":" + addZero(mi) + ":" + addZero(sec) ;

        p = 0;

        while (Time[p])

        {

            print_tiny_char((p * 4), 2, Time[p]); p++;

        }

        // settime

        if (stateSet == true) //สามารถตั้งเวลาได้

        {

            hour = (map(analogRead(A0), 0, 1000, 0, 23));

```

```

        mi = (map(analogRead(A5), 0, 1000, 0, 59));

    }

}

break;

```

case 1: // แสดงเวลา ของนาฬิกาปลุก

```

    Time = addZero(hour1) + ":" + addZero(mi1) + ":" + addZero(sec1) ;

    p = 0;

    while (Time[p]) {

        print_tiny_char((p * 4), 2, Time[p]); p++;

    }

    if (stateSet == true) //สามารถตั้งเวลาได้

    {

        hour1 = (map(analogRead(A0), 0, 1000, 0, 23));

        mi1 = (map(analogRead(A5), 0, 1000, 0, 59));

    }

    break;

```

case 2: // แสดงเวลา ของการจับเวลา ไม่สามารถตั้งเวลาได้

```

    Time = addZero(hour2) + ":" + addZero(mi2) + ":" + addZero(sec2) ;

```

```

p = 0;

while (Time[p])

{

    print_tiny_char((p * 4), 2, Time[p]); p++;

}

break;

```

case 3: //แสดงเวลา ของการนับถอยหลัง

```

Time = addZero(hour3) + ":" + addZero(mi3) + ":" + addZero(sec3) ;

p = 0;

while (Time[p]) {

    print_tiny_char((p * 4), 2, Time[p]); p++;

}

if (stateSet == true) //สามารถตั้งเวลาได้

{

    hour3 = (map(analogRead(A0), 0, 1000, 0, 23));

    mi3 = (map(analogRead(A5), 0, 1000, 0, 59));

}

break;

}

```

```
// print_tiny_char(31, 2, '.');  
  
// Serial.println();  
  
}
```

```
//////////////////////////////////// Function //////////////////////////////////////
```

```
void plot (uint8_t x, uint8_t y, uint8_t value) {
```

```
    //select which matrix depending on the x coord
```

```
    uint8_t address;
```

```
    if (x >= 0 && x <= 7)
```

```
    {
```

```
        address = 3;
```

```
    }
```

```
    if (x >= 8 && x <= 15)
```

```
    {
```

```
        address = 2;
```

```
        x = x - 8;
```

```
    }
```

```
    if (x >= 16 && x <= 23)
```



```

{
    sec = 0;

    mi++;
}

if (mi > 59)

{
    mi = 0;

    hour++;
}

if (hour > 23)

{
    hour = 0;
}
}

```

void TIME2() ////////////////////////////////////// นาฬิกาปลุก

```

{

if (sec1 < 0)

{

    sec1 = 59;

```

```
    mi1--;  
  
}  
  
if (mi1 < 0)  
{  
  
    mi1 = 59;  
  
    hour1--;  
  
}  
  
if (hour1 < 0)  
{  
  
    hour1 = 23;  
  
}  
  
////////////////////////////////////  
  
if (sec1 >= 60)  
{  
  
    sec1 = 0;  
  
    mi1++;  
  
}  
  
if (mi1 >= 60)  
{  
  
    mi1 = 0;
```

```

    hour1++;

}

if (hour1 >= 24)

{

    hour1 = 0;

}

}

void TIME3() ////////////////////////////////////// นาฬิกาปลุก

{

    if (sec2 > 59)

    {

        sec2 = 0;

        mi2++;

    }

    if (mi2 > 59)

    {

        mi2 = 0;

        hour2++;

    }

    if (hour2 > 59)

```

```

{
    hour2 = 0;
}
}

void TIME4() //////////////////////////////////// นับถหลัง
{
    if (sec3 < 0)
    {
        sec3 = 59;
        mi3--;
    }

    if (mi3 < 0)
    {
        mi3 = 59;
        hour3--;
    }

    if (hour3 < 0)
    {
        hour3 = 23;
    }
}

```



```
}
```

```
void music()
```

```
{
```

```
    for (int h = 0; h < 2; h++)
```

```
    {
```

```
        tone(4, 587);
```

```
        delay(100);
```

```
        tone(4, 587);
```

```
        delay(100);
```

```
        tone(4, 659);
```

```
        delay(100);
```

```
        tone(4, 659);
```

```
        delay(100);
```

```
        tone(4, 349);
```

```
        delay(100);
```

```
        tone(4, 349);
```

```
        delay(100);
```

```
        tone(4, 440);
```

```
        delay(100);
```

```
        tone(4, 494);
```

```
    delay(100);

    tone(4, 494);

    delay(100);

    tone(4, 587);

    delay(100);

    tone(4, 587);

    delay(100);

    tone(4, 494);

    delay(100);

}

}

void music1()

{

    for (int i = 0; i < 3; i++)

    {

        tone(4, 587);

        delay(100);

        tone(4, 587);

        delay(100);

    }

}
```

```

}

void print_tiny_char(uint8_t x, uint8_t y, char c) {

    uint8_t dots;

    if (c >= 'A' && c <= 'Z' || (c >= 'a' && c <= 'z') ) {

        c &= 0x1F; // A-Z maps to 1-26

    }

    else if (c >= '0' && c <= '9')

    {

        c = (c - '0') + 32;

    }

    else if (c == ' ') {

        c = 0; // space

    }

    else if (c == '.') {

        c = 27; // full stop

    }

    else if (c == ':') {

        c = 28; // colon

    }

    else if (c == "\"") {

```

```

    c = 29; // single quote mark
}

else if (c == '!') {

    c = 30; // single quote mark
}

else if (c == '?') {

    c = 31; // single quote mark
}


for (uint8_t col = 0; col < 3; col++) {

    dots = pgm_read_byte_near(&mytinyfont[c][col]);

    for (uint8_t row = 0; row < 5; row++) {

        if (dots & (16 >> row))

            plot(x + col, y + row, 1);

        else

            plot(x + col, y + row, 0);

    }

}

}

void printsting(String input) //ใส่ได้ 8 ตัวพอดี

```

```
{  
  
    int positionX = 1, positionY = 2;  
  
    for (int i = 0; i < input.length(); i++)  
  
    {  
  
        print_tiny_char(positionX, positionY, input[i]);  
  
        positionX += 4;  
  
    }  
  
}
```

```
void clear_display()  
  
{  
  
    for (uint8_t address = 0; address < 4; address++)  
  
    {  
  
        lc.clearDisplay(address);  
  
    }  
  
}
```

```
String addZero(int s)  
  
{  
  
    String str = "";
```

```
if (s / 10 == 0)

{

    str = "0" + (String)s;

} else str = (String)s;

return str;

}
```