

## 240-319 Embedded System Developer Module

## LAB 17 : EEPROM ปีการศึกษา 2567

## อุปกรณ์ที่ใช้ในการทดลอง

- บอร์ด Arduino UNO พร้อมบอร์ด CoE PSU Arduino Shield
- มอดูล PCF-8574
- คีย์แพดแบบเมมเบรน 4x4
- มอดูล DS-1307
- มอดูล LCD ชนิด I2C

## 17.1 คลังโปรแกรม my\_EEPROM.h

ให้นักศึกษาเปิดไฟล์ my\_EEPROM.h ประกอบด้วยฟังก์ชันสำเร็จรูปสำหรับการติดต่ออีอีพรีมในตัวประมวลผล ATmega328P พร้อมพิจารณาการทำงานของฟังก์ชันต่าง ๆ ดังนี้

void EEPROM_Erase_and_Write1B(uint16_t addr, uint8_t data);	//ลบและเขียนข้อมูลในตำแหน่งที่ระบุ
void EEPROM_Erase_only(uint16_t addr);	//ลบข้อมูลในตำแหน่งที่ระบุ
void EEPROM_Write_to_Empty_location(uint16_t addr, uint8_t data);	//ลบข้อมูลในตำแหน่งที่ระบุ โดยตำแหน่ง //นั้นจะต้องเป็นตำแหน่งที่ไม่มีข้อมูลเก็บ //อยู่มาก่อน
unsigned char EEPROM_read1byte(uint16_t addr) ;	//อ่านข้อมูลในตำแหน่งที่ระบุ
void display_all_data_in_EEPROM(void);	//แสดงข้อมูลในหน่วยความจำอีอีพรีมทุก //ตำแหน่งผ่าน Serial Monitor
void update_if_data_changed(uint16_t addr, uint8_t data);	//เขียนข้อมูลในตำแหน่งที่ระบุหากพบว่า //ข้อมูลที่ต้องการเขียนมีค่าแตกต่างกันกับ //ข้อมูลที่บันทึกไว้เดิม

ตัวประมวลผล ATmega328P มีหน่วยความจำอีอีพรีมให้ใช้งาน 1024 ตำแหน่ง โดยตำแหน่งใดที่ยังไม่มีการเขียนข้อมูลจะมีค่า 0xFF เก็บอยู่ ซึ่งหมายถึง ไม่มีข้อมูลเก็บอยู่ภายใน ในการใช้ฟังก์ชัน EEPROM\_Erase\_only จะส่งผลให้ข้อมูลตำแหน่งที่ถูกลบเปลี่ยนค่าเป็น 0xFF

## 17.2 Checkpoint 1

พิจารณาโปรแกรม "Example1" ในรูปที่ 17.1 จากนั้นต้องจตามรูปที่ 17.2 จากนั้นให้สร้างโปรเจกต์ใหม่บน Arduino IDE พร้อมคัดลอกโค้ดในโปรแกรม Example1 ไปใส่ พร้อมทั้งดาวน์โหลดไฟล์ my\_EEPROM.h จาก LMS และจัดเก็บไฟล์นี้ไว้ในโฟลเดอร์เดียวกันกับโปรแกรม Example1 ทดลองคอมไพล์และสังเกตผลลัพธ์การทำงานบนโปรแกรม Serial Monitor

จากนั้นให้นำโปรแกรม Example1 มาดัดแปลงโดยให้โปรแกรมสำหรับ Checkpoint1 มีความสามารถดังนี้

- เมื่อเริ่มทำงาน โปรแกรมจะค้นหาว่าหน่วยความจำอีอีพรีมมีข้อมูลเก็บอยู่ (ข้อมูลที่มีใช้ค่า 0xFF) ในตำแหน่งใดบ้าง หากพบว่ามีข้อมูลอยู่ภายในอีอีพรีมมากกว่า 1 ตำแหน่ง ให้ลบค่าในตำแหน่งที่มีข้อมูลอยู่ตำแหน่งนั้น ๆ ออกให้หมด
- หากเริ่มทำงานแล้วโปรแกรมพบว่าไม่มีข้อมูลใดเลยเก็บอยู่ในอีอีพรีม ให้เขียนค่า 0xFF ลงไปในอีอีพรีมตำแหน่ง 1021-1023 โดยให้ค่า 0xFF, 0xCE, 0x0F ถูกเก็บอยู่ในหน่วยความจำตำแหน่ง 1021, 1022, 1023 ตามลำดับ
- หากโปรแกรมเริ่มทำงานใหม่ (หลังจากการรีเซ็ตตัวประมวลผล) แล้ว ให้ทำการแสดข้อมูลในอีอีพรีม หากพบว่ามีข้อมูลอยู่ 0xFF, 0xCE, 0x0F อยู่ติดกันในอีอีพรีมสามตำแหน่ง ก็ให้ลบข้อมูลทั้งสามนั้นแล้วเขียนข้อมูล 0xFF ลงในหน่วยความจำ 3 ตำแหน่งที่ต่ำกว่าลงมาดังแสดงในรูปที่ 17.1A

```

#include "my_EEPROM.h"

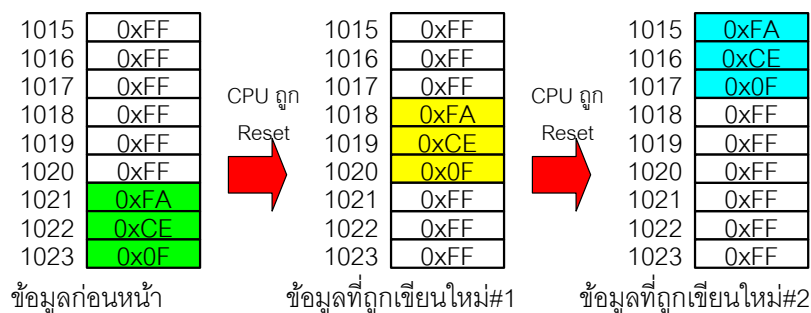
uint16_t scan_EEPROM(void)
{
    int i, data_location, number_data_found;
    uint8_t d;
    number_data_found = 0;
    Serial.println("Performing EEPROM scanning...");
    for(i=0;i<1024;i++)
    {
        d = EEPROM_readlbyte(i);
        if (d != 0xFF)
        {
            data_location = i;
            number_data_found++;
            Serial.print(d);
            Serial.print(" is found in EEPROM location[");
            Serial.print(i);
            Serial.println("]");
        }
    }
    if (number_data_found > 1)
        Serial.println("More than one data are found in EEPROM");
    else if(number_data_found ==0)
        Serial.println("no data found in EEPROM");
}

void setup()
{
    Serial.begin(38400);
    display_all_data_in_EEPROM();
    scan_EEPROM();
}

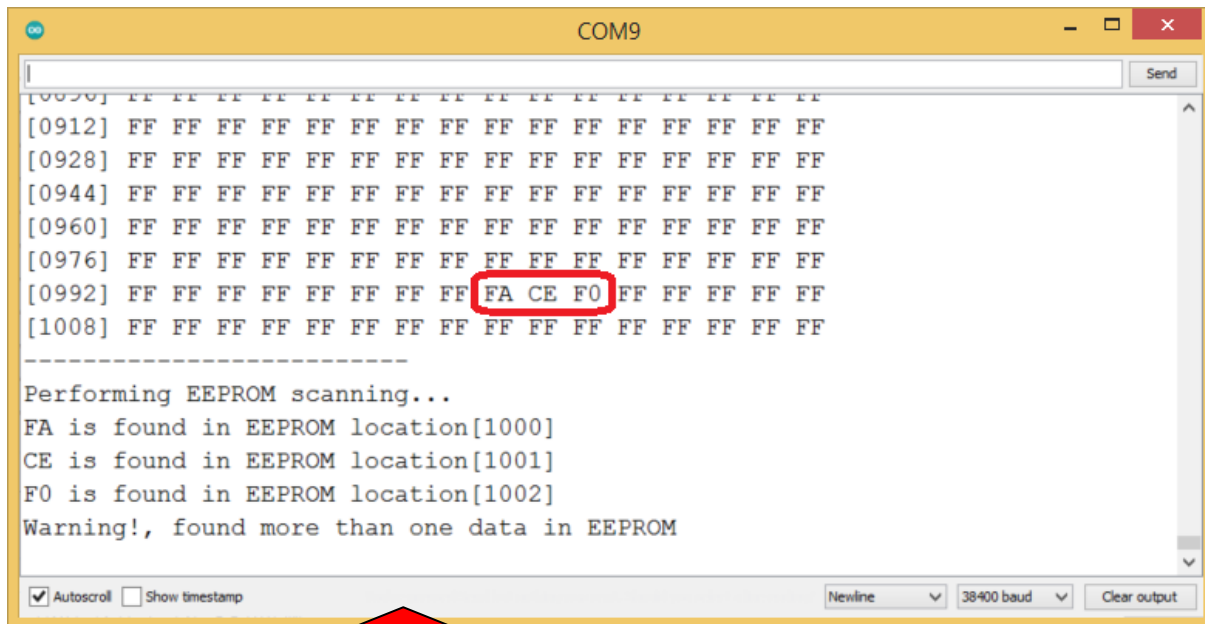
void loop() {
}

```

รูปที่ 17.1 โค้ดโปรแกรม Example1



รูปที่ 17.1A ตัวอย่างข้อมูลการเขียนข้อมูลในอีพีรอมในแต่ละครั้งหลักซีพียูถูกรีเซ็ต



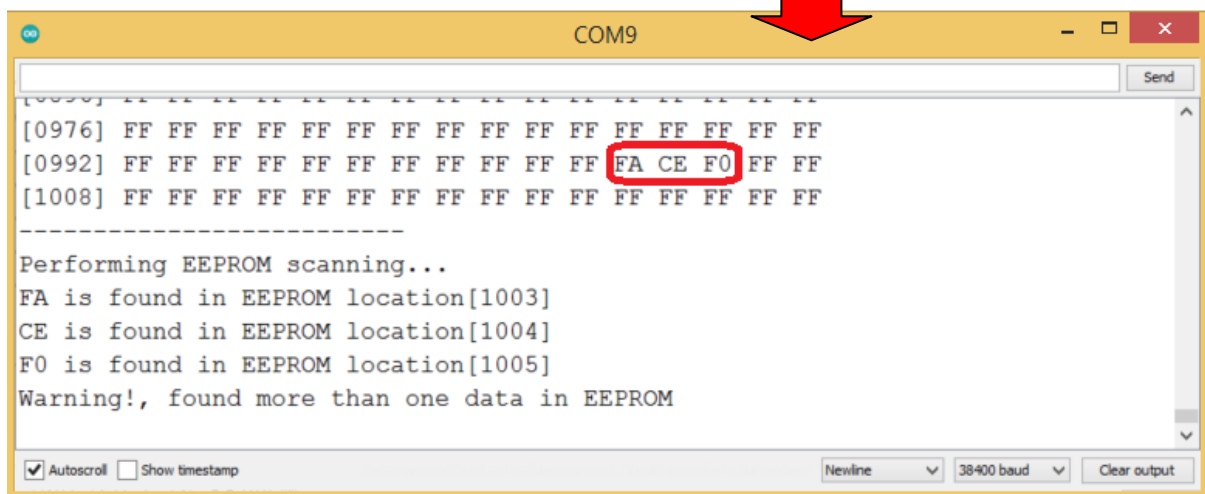
```

[0912] FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
[0928] FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
[0944] FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
[0960] FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
[0976] FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
[0992] FF FF FF FF FF FF FF FF FA CE F0 FF FF FF FF FF
[1008] FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

-----
Performing EEPROM scanning...
FA is found in EEPROM location[1000]
CE is found in EEPROM location[1001]
F0 is found in EEPROM location[1002]
Warning!, found more than one data in EEPROM
  
```

ก่อนถูกรีเซ็ต

หลังจากถูกรีเซ็ต



```

[0976] FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
[0992] FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
[1008] FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

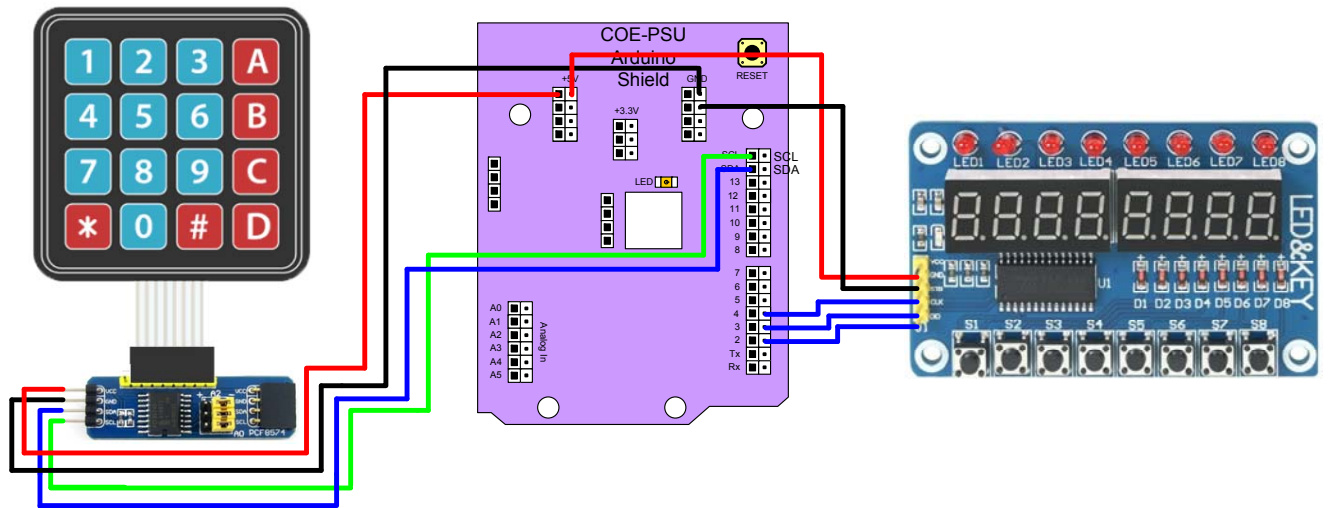
-----
Performing EEPROM scanning...
FA is found in EEPROM location[1003]
CE is found in EEPROM location[1004]
F0 is found in EEPROM location[1005]
Warning!, found more than one data in EEPROM
  
```

รูปที่ 17.1B ตัวอย่างหน้าจอ Serial Monitor แสดงข้อมูลที่ถูกรวบรวมและเขียนลงในอีอีพรีมในแต่ละครั้งหลังรีเซ็ต

กำหนดให้โปรแกรมมีการเขียนค่า 0xFACE0F ลงในอีอีพรีมเพียง 1 ครั้ง หลังจากที่ได้รับกระแสไฟจ่ายให้ทำงาน นั่นหมายความว่า ทุกครั้งที่มีการกดสวิตช์รีเซ็ต หรือ ทุกครั้งที่มีการเริ่มจ่ายไฟให้กับวงจร จะมีการเขียนค่าทั้งสามไบต์นี้ลงในอีอีพรีมตำแหน่งที่ลดลงจากตำแหน่งที่เจอข้อมูลเดิมสามตำแหน่ง ดังตัวอย่างในรูปที่ 17.1B

#### ข้อมูลสำหรับ TA ในการตรวจ Checkpoint 1

ในการตรวจ checkpoint นี้ ให้นักศึกษาแสดงให้เห็นว่า ทุกครั้งที่มีการกดรีเซ็ตแล้ว ค่า 0xFACE0F จะถูกเขียนลงในหน่วยความจำอีอีพรีมที่ลดลงจากตำแหน่งเดิมที่เขียนก่อนหน้านี้ 3 ตำแหน่ง ดังตัวอย่างในรูปที่ 17.1B

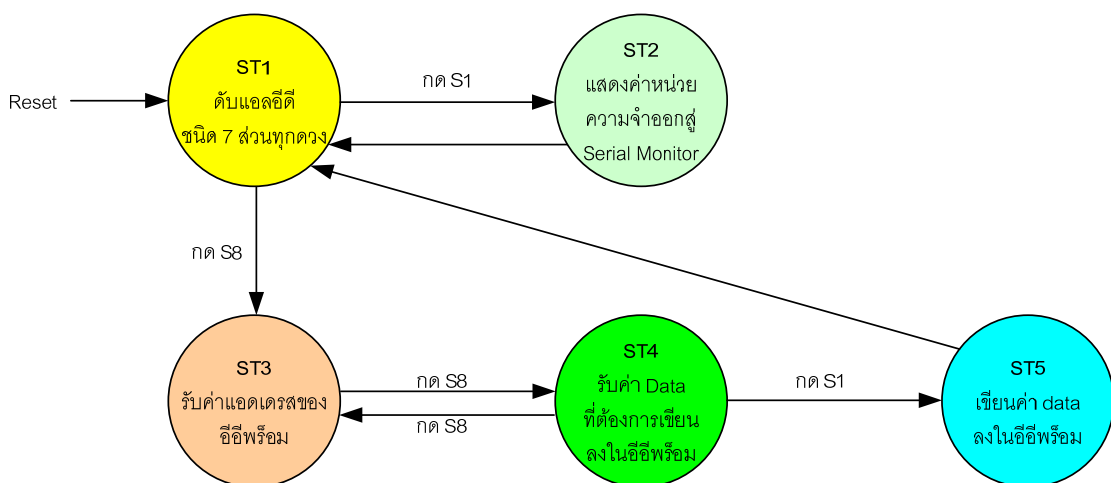


รูปที่ 17.2 การต่อวงจรสำหรับการทดลองที่ 17

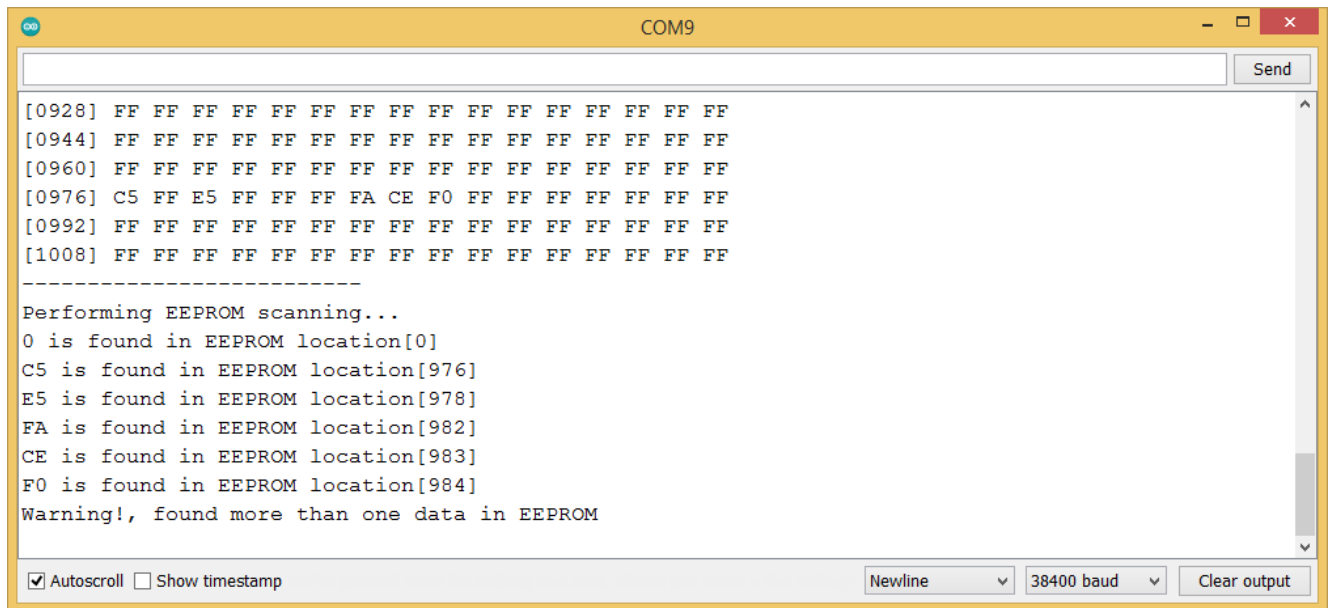
### 17.3 Checkpoint 2

ต่อวงจรตามรูปที่ 17.2 จากนั้นให้เขียนโปรแกรมเพื่ออ่านค่าจากคีย์แป้นเพื่อป้อนค่าให้กับหน่วยความจำอีอีพรีม โดยกำหนดให้มีลำดับการทำงานดังนี้

- เมื่อเริ่มเปิดเครื่อง ให้ระบบอยู่ในสถานะ ST1 ซึ่งส่งผลให้แอลอีดีชนิด 7 ส่วนทุกดวงบนบอร์ด TM1638 ดับทั้งหมด
- กดปุ่มสวิทช์ S8 บนบอร์ด TM1638 เพื่อเริ่มต้นการรับค่าตำแหน่งหน่วยความจำที่ต้องการติดต่อด้วย และให้เลขที่แสดงผลบนแอลอีดีชนิด 7 ส่วนบนบอร์ด TM1638 จำนวน 4 หลักทางซ้ายมือกระพริบติดดับเป็นจังหวะ และทำการรับค่าตำแหน่งหน่วยความจำจากคีย์แป้นในรูปแบบ เลขฐานสิบไม่เกิน 4 หลัก เพื่อระบุหมายเลขตำแหน่งหน่วยความจำอีอีพรีมที่ต้องการติดต่อ และให้กดปุ่มสวิทช์ S8 บนบอร์ด TM1638 เพื่อเสร็จสิ้นขั้นตอนการป้อนหมายเลขแอดเดรสของอีอีพรีม
- เมื่อเสร็จสิ้นขั้นตอนการรับค่าแอดเดรสแล้ว ให้กระพริบตัวเลขบน TM1638 ที่สองตำแหน่งขวาสุดแทนเพื่อรอรับค่าที่ต้องการเขียนให้กับอีอีพรีมตำแหน่งนั้น ๆ โดยกำหนดให้ป้อนเป็นเลขฐานสิบหก และถือว่าปุ่ม \* คือค่า E และถือว่าปุ่ม # คือ ค่า F เมื่อเปลี่ยนค่าที่ต้องการเขียนเสร็จ ให้กดปุ่มสวิทช์ S1 บน TM1638 เพื่อเขียนค่าที่ต้องการลงในอีอีพรีม
- เมื่อเขียนค่าที่ต้องการเสร็จ ให้ดับแอลอีดีทุกดวง และกลับมาอยู่สถานะ ST1 ดังรูปที่ 17.3
- ในขณะที่อยู่ในสถานะ ST1 หากมีการกดสวิทช์ S1 บนบอร์ด TM1638 ให้แสดงค่าจากอีอีพรีมทุกตำแหน่งออกทาง Serial Monitor ดังรูปที่ 17.4



รูปที่ 17.3 สถานะการทำงานของวงจรใน checkpoint ที่ 2



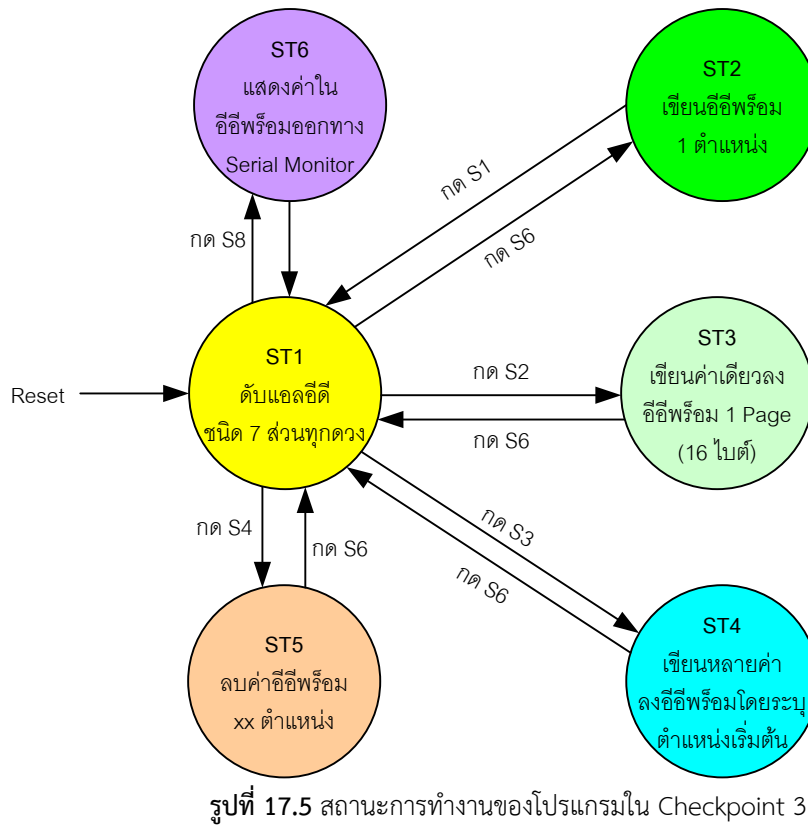
รูปที่ 17.4 ตัวอย่างการแสดงผลข้อมูลในหน่วยความจำอีอีพรอมใน checkpoint ที่ 2

### ข้อมูลสำหรับ TA ในการตรวจ Checkpoint 2

ให้ TA สุ่มค่าตำแหน่งหน่วยความจำที่ต้องการเขียนอย่างน้อย 2 ตำแหน่งที่ไม่ติดกัน และค่าที่เขียนลงในหน่วยความจำทั้งสองตำแหน่งต้องไม่ใช่ค่า 0xFF หลังจากนั้น ให้ปลดไฟเลี้ยงของบอร์ด Arduino ออกแล้วจ่ายไฟเลี้ยงอีกครั้ง จากนั้นกดสวิตช์ S1 เพื่อดูว่าค่าที่เพิ่งเขียนไปยังอยู่ใน EEPROM หรือไม่

เกณฑ์การให้คะแนน Checkpoint 2 สำหรับ TA	คะแนน (เต็ม 10)
โปรแกรมสามารถเขียนข้อมูลตำแหน่งที่ TA ต้องการลงใน EEPROM ตำแหน่งที่ TA ระบุ โดยการใช้การป้อนข้อมูลจากคีย์แพดและเมื่อถอดไฟเลี้ยงออก นักศึกษาสามารถแสดงให้เห็นว่าข้อมูลตำแหน่งนั้น ๆ ยังอยู่ใน EEPROM ด้วยการแสดงผลออกทาง Serial Monitor	7.5
ในการการรับข้อมูลแอดเดรส เป็นเลขฐานสิบ มีการกระพริบติดดับของค่าแอดเดรสเป็นจังหวะครั้งละ 1 วินาที	0.5
ในการการรับข้อมูลที่จะเขียน เป็นเลขฐานสิบหก มีการกระพริบติดดับของค่าข้อมูลเป็นจังหวะครั้งละ 1 วินาที	0.5
ในการรับค่าแอดเดรส หากมีการกดปุ่ม A-D หรือปุ่ม * หรือปุ่ม # ระบบจะต้องไม่ตอบสนอง เนื่องจากไม่ใช่เลขฐานสิบที่ถูกต้อง	0.5
การรับค่าแอดเดรสของหน่วยความจำหรือค่าข้อมูลที่จะเขียน ให้รับที่หลักหน่วยก่อน แล้วค่อยมารับที่หลักสิบ จากนั้นค่อยมารับที่หลักร้อยและหลักพันตามลำดับ (ดูในคลิปตัวอย่างใน LMS) แบบเดียวกับการป้อนข้อมูลให้เครื่องคิดเลข	0.5
เมื่อรับข้อมูลที่จะเขียนเสร็จแล้ว หากมีการกดปุ่ม S8 จะสามารถสลับไปมาระหว่างสถานะ ST3 และสถานะ ST4 ได้ไปเรื่อย ๆ และยังไม่มีการเขียนค่าลง EEPROM จนกว่าจะมีการกดปุ่ม S1	0.5

หมายเหตุ Link Video สาธิตการทำงานของ Checkpoint 2 <https://youtu.be/MLbuXcdfVlc>



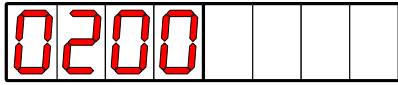
### 17.5 Checkpoint 3 (Bonus)

ใน Checkpoint 3 นี้ ใครทำไม่เสร็จก็ไม่เป็นไร แต่หากทำเสร็จจะมีคะแนนเสริมให้ โดยให้ใช้วงจรที่ต่อในรูปที่ 17.2 เหมือนเดิม แต่ให้เขียนโปรแกรมเพื่อจัดการข้อมูลในอีพีร้อม โดยมีสถานะการทำงานอยู่ 6 สถานะ ได้แก่

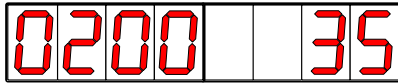
- สถานะ ST1 เป็นสถานะเริ่มต้นเมื่อเริ่มเปิดเครื่อง โดยแอลอีดีชนิด 7 ส่วนทุกดวงบนบอร์ด TM1638 ดับทั้งหมด
- สถานะ ST2 เป็นสถานะที่รับการเขียนค่าในหน่วยความจำตำแหน่งที่ระบุเพียง 1 ตำแหน่ง (เหมือนกับ Checkpoint 2)
- สถานะ ST3 เป็นสถานะที่รับค่าหนึ่งค่า มาเขียนหน่วยความจำครั้งละ 1 page ซึ่งประกอบด้วย 16 ตำแหน่งหน่วยความจำ โดยกำหนดให้รับค่าแอดเดรสเริ่มต้น และค่าที่ต้องการเขียน และเมื่อกดปุ่ม S8 จะมีการเขียนค่าลงไปหน่วยความจำทั้ง page (จำนวน 16 ไบต์)
- สถานะ ST4 เป็นสถานะที่มีการเขียนค่าลงหน่วยความจำหลายค่า โดยรับค่าตำแหน่งเริ่มต้นที่ต้องการเขียนก่อน จากนั้นหากมีการกด S8 ก็จะเริ่มรับค่าแรกที่เขียนในหน่วยความจำตำแหน่งนั้นผ่านคีย์เพ็คครั้งละ 2 หลัก และจะมีการนำค่าที่รับได้เขียนในตำแหน่งที่ระบุทันที หากมีการกดปุ่มบนคีย์แพ็คใหม่ ก็จะถือว่าต้องการเขียนค่าใหม่ในตำแหน่งถัดไป และให้เป็นเช่นนี้ไปเรื่อย ๆ จนกว่าจะกดปุ่ม S6 เพื่อออกจากสถานะ ST4 ดังตัวอย่างในรูปที่ 17.6
- สถานะ ST5 เป็นการลบค่าจำนวน xx ตำแหน่ง ออกจากหน่วยความจำอีพีร้อม โดยกำหนดให้เลขสี่หลักด้านซ้ายมือของแอลอีดีเป็นค่าตำแหน่งเริ่มต้นของหน่วยความจำ (รับเข้าเป็นเลขฐานสิบ) ที่ถูกลบ และเลขสองหลักด้านซ้ายมือแทนจำนวนหน่วยความจำที่ถูกลบซึ่งเป็นเลขฐานสิบ (รับค่าได้ตั้งแต่ 00-99) ดังรูปที่ 17.7
- กำหนดให้รับค่าแอดเดรสของหน่วยความจำด้วยเลขฐานสิบ และค่าที่เขียนลงในหน่วยความจำให้ใช้เป็นเลขฐานสิบหก



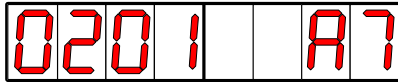
เริ่มต้นแอลอีดีดับทุกหลัก



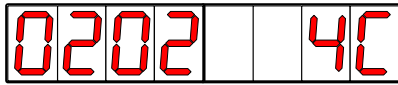
ขั้นที่ 1 รับค่าแอดเดรสจากคีย์แพด (ในที่นี้เป็นการระบุตำแหน่ง  $200_{10}$ )  
เมื่อเสร็จให้กด F8 เพื่อเปลี่ยนไปรับค่าข้อมูล



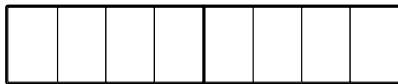
ขั้นที่ 2 หากต้องการบันทึกค่า 35 ลงในหน่วยความจำตำแหน่ง 200  
ให้ทำการกดปุ่ม 3 ก่อน ตามด้วยปุ่ม 5  
เมื่อกดปุ่ม 5 เสร็จระบบก็จะบันทึกค่า 35 ลงในหน่วยความจำตำแหน่ง 200



ขั้นที่ 3 หากต้องการบันทึกค่า A7 ลงในหน่วยความจำตำแหน่ง 201  
ให้ทำการกดปุ่ม A ก่อน จากนั้นระบบจะเปลี่ยนตำแหน่งเป็น 201 ให้อัตโนมัติ ตามด้วยการกดปุ่ม 7 โดยผู้ใช้  
เมื่อกดปุ่ม 7 เสร็จระบบก็จะบันทึกค่า A7 ลงในหน่วยความจำตำแหน่ง 201

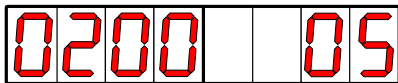


ขั้นที่ 4 หากต้องการบันทึกค่า 4C ลงในหน่วยความจำตำแหน่ง 202  
ให้ทำการกดปุ่ม 4 ก่อน จากนั้นระบบจะเปลี่ยนตำแหน่งเป็น 202 ให้อัตโนมัติ ตามด้วยการกดปุ่ม C โดยผู้ใช้  
เมื่อกดปุ่ม C เสร็จระบบก็จะบันทึกค่า 4C ลงในหน่วยความจำตำแหน่ง 202

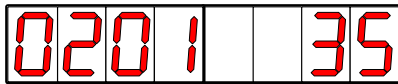


ขั้นที่ 5 หากต้องการหยุดการเขียนข้อมูลเพียงเท่านั้น ก็ให้กดปุ่ม S6 บนบอร์ด TM1638  
ระบบจะดับแอลอีดีทุกดวง

รูปที่ 17.6 ตัวอย่างการทำงานของสถานะ ST4 ใน Checkpoint 3



หากต้องการลบค่าในอีซีพีรอมตั้งแต่ตำแหน่ง 200-204 ก็ให้พิมพ์ตำแหน่ง 200 ทางซ้ายมือ และป้อนเลข 0  
ตามด้วย 5 เพื่อบอกว่าลบ 5 ตำแหน่ง



หากต้องการลบค่าในอีซีพีรอมตั้งแต่ตำแหน่ง 201-235 ก็ให้พิมพ์ตำแหน่ง 200 ทางซ้ายมือ และป้อนเลข 3  
ตามด้วย 5 เพื่อบอกว่าลบจำนวน 35 ตำแหน่ง

รูปที่ 17.7 ตัวอย่างการทำงานของสถานะ ST5 ใน Checkpoint 3

-----THE END-----