

240-319 Embedded System Developer Module

LAB 15 : Sleep modes of ATmega328P

อุปกรณ์ที่ใช้ในการทดลอง

- บอร์ด Arduino UNO พร้อมบอร์ด CoE PSU Arduino Shield
- บอร์ด CoE สวิตช์กดติดปล่อยดับและบัลลูนเซอร์
- บอร์ด CoE แสดงผลแอลอีดี 8 ดวง
- มอดูลวัดระดับแรงดันและกระแสแบบยูเอสบี
- มอดูล TM-1638

15.1 คลังโปรแกรม <avr/sleep.h> และคลังโปรแกรม my_wdt.h

ศึกษาการทำงานของฟังก์ชันต่าง ๆ ในคลังโปรแกรม <avr/sleep.h>

```
//---https://www.nongnu.org/avr-libc/user-manual/group__avr__sleep.html
void sleep_enable(void);           //เปิดการทำงานของตัวประมวลผลให้สามารถเข้าสู่ภาวะหลับได้
void sleep_disable(void);          //ปิดทางป้องกันมิให้ตัวประมวลผลสามารถเข้าสู่ภาวะหลับ
void sleep_cpu(void);              //สั่งให้ตัวประมวลผลเข้าสู่ภาวะหลับ
void sleep_bod_disable(void) ;     //ปิดการทำงานของ BOD (ใช้ก่อนจะเข้าสู่ภาวะหลับ)
void set_sleep_mode(mode);         //ตั้งโหมดการทำงานของภาวะหลับ ซึ่งเลือกได้ดังนี้
//      SLEEP_MODE_PWR_SAVE
//      SLEEP_MODE_PWR_DOWN
//      SLEEP_MODE_IDLE
//      SLEEP_MODE_EXT_STANDBY
//      ตัวอย่าง set_sleep_mode(SLEEP_MODE_PWR_SAVE);
```

จากนั้นให้ศึกษาการทำงานของฟังก์ชันต่าง ๆ ในคลังโปรแกรม my_wdt.h

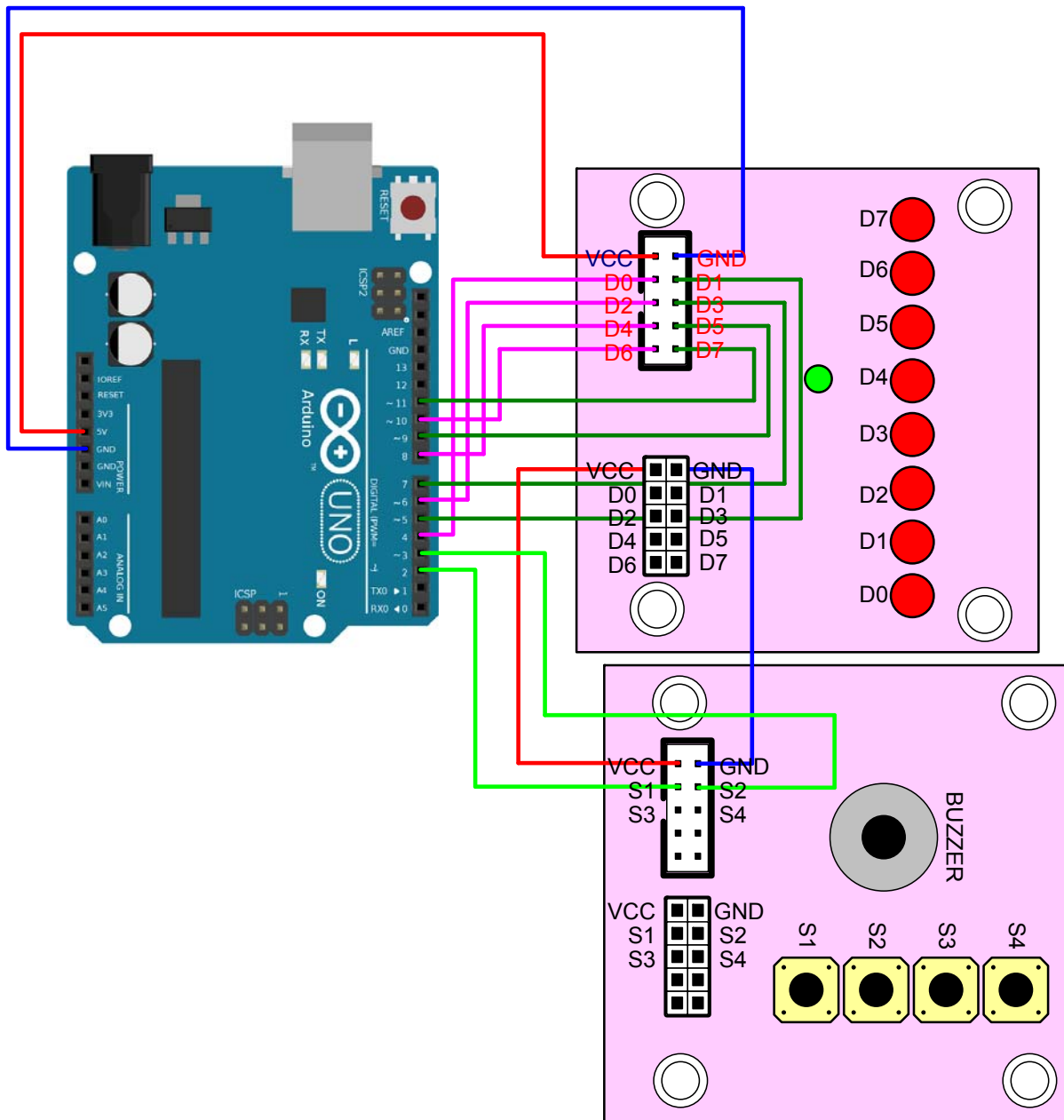
```
void WDT_interrupt_enable(uint8_t timeout_v); //เปิดการทำงานของ WDT ให้ทำงานในโหมดการขัดจังหวะซีพียู
void WDT__enable(uint8_t timeout_v);          //เปิดการทำงานของ WDT ให้ทำงานในโหมดการรีเซ็ตซีพียู
void WDT__disable(void);                      //ปิดการทำงานของ WDT
```

15.2 การทำ Checkpoint และการคิดคะแนน

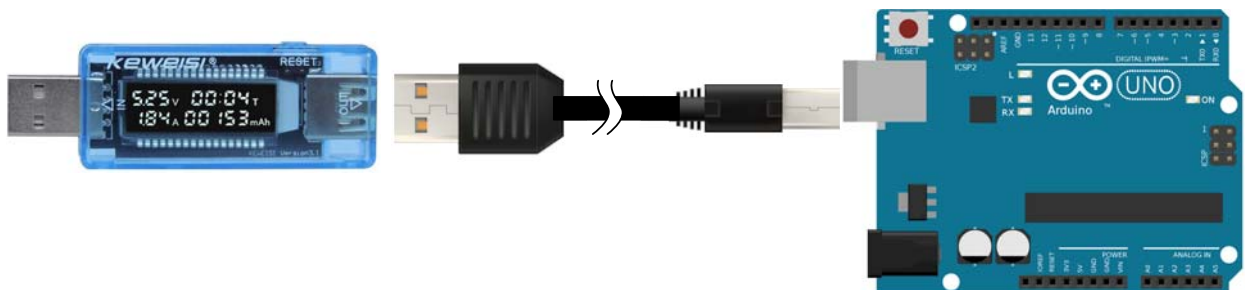
Checkpoint 1- Checkpoint 3 ให้ทำทุกคน ส่วน Checkpoint 4 ใครจะทำหรือไม่ก็ได้ แต่หากทำจะมีคะแนนพิเศษให้ การทดลองนี้แม้จะไม่ได้มีการกำหนดให้ส่งรายงาน แต่แนะนำให้นักศึกษาควรบันทึกผลการทดลองและวิเคราะห์ผลการทดลองอย่างละเอียดเพื่อสรุปความรู้ที่ได้จากการทดลองเพิ่มเติม เพื่อใช้เป็นความรู้ในการทำข้อสอบในโอกาสต่อไป

15.3 Checkpoint 1

ต่อวงจรตามรูปที่ 15.1 จากนั้นเขียนโปรแกรมเพื่ออ่านค่าจากจากสวิตช์กดติดปล่อยดับจำนวนสองตัว โดยทุกครั้งที่มีการกด S1 จะทำให้ค่าที่แสดงผลบนบอร์ด LED ทั้งแปดดวงเพิ่มค่าขึ้นครั้งละ 1 ค่า แต่หากมีการกด S2 จะทำให้ค่าที่แสดงผลบนแอลอีดีมีค่าลดลงครั้งละ 1 ค่า



รูปที่ 15.1 การต่อวงจรสำหรับ Checkpoint 1-Checkpoint 2



รูปที่ 15.2 การเชื่อมต่อมอดูลวัดกระแสเพื่อวัดปริมาณกระแสไฟฟ้าเข้าสู่บอร์ด Arduino UNO

ใน Checkpoint นี้แบ่งออกเป็น 2 ส่วนย่อย ให้วัดการกินกระแสของวงจรในแต่ละ Checkpoint และบันทึกผลการทดลองใส่ในรายงานด้วย

- Checkpoint 1.1 กำหนดให้เขียนโปรแกรมโดยให้โปรแกรมหลักเป็นการวนลูปเปล่า และให้ใช้วิธีการติดต่อกับ S1 และ S2 ด้วยกลไกการขัดจังหวะ ซึ่งหมายถึงการเพิ่มหรือลดค่าของการแสดงผลเอาต์พุตจะถูกดำเนินการในโปรแกรมบริการการขัดจังหวะ (Interrupt Service Routine: ISR) เท่านั้น

- Checkpoint 1.2 กำหนดให้ตัวประมวลผลเข้าสู่ภาวะหลับตลอดเวลา และจะตื่นขึ้นมาจากภาวะหลับอันเนื่องมาจากการขัดจังหวะของสวิตช์ S1 หรือสวิตช์ S2 และหลังจากเพิ่ม/ลดค่าของเอาต์พุตที่แสดงผลแล้วให้ตัวประมวลผลเข้าสู่ภาวะหลับอีกครั้ง

ให้ต่อมอดูลวัดกระแสคั่นระหว่างพอร์ตยูเอสบีซีของเครื่องคอมพิวเตอร์กับบอร์ด Arduino UNO ดังรูปที่ 15.2 ในการอ่านค่ากระแสของวงจรใน Checkpoint 1.1 และ Checkpoint 1.2 ให้อ่านค่ากระแสในขณะที่เอาต์พุตมีแอลอีดีสว่างเพียงหนึ่งดวงเท่านั้น และให้แจ้งค่ากระแสของแต่ละ Checkpoint ให้ผู้ตรวจทราบด้วย

15.4 Checkpoint 2

ให้ใช้วงจรเดียวกับ Checkpoint 1 แต่ต่อวงจรตามรูปที่ 15.1 จากนั้นเขียนโปรแกรมเพื่ออ่านค่าจากจากสวิตช์กดติดปล่อยดับจำนวนสองตัว โดยทุกครั้งที่มีการกด S1 จะทำให้มีการแสดงไฟวิ่งติดครั้งละสองดวงโดยวิ่งจากขอบนอกเข้ามาสู่ตรงกลาง (ดูวิดีโอตัวอย่างใน LMS) แต่หากมีการกด S2 จะทำให้มีการแสดงไฟวิ่งจากตรงกลางครั้งละสองดวงและวิ่งออกไปสู่ขอบนอก (ดูวิดีโอตัวอย่างใน LMS)

ใน Checkpoint นี้แบ่งออกเป็น 2 ส่วนย่อย ให้วัดการกินกระแสของวงจรในแต่ละ Checkpoint และบันทึกผลการทดลองใส่ในรายงานด้วย

- Checkpoint 2.1 กำหนดให้เขียนโปรแกรมโดยใช้วิธีการติดต่อกับสวิตช์ทั้งสองตัวแบบโพลลิง ซึ่งหมายถึงให้ซีพียูวนลูปรอตรวจสอบการกดสวิตช์ทั้งสองตัวตลอดเวลา โดยไม่มีการเข้าสู่ภาวะหลับ

- Checkpoint 2.2 กำหนดให้ตัวประมวลผลเข้าสู่ภาวะหลับตลอดเวลา และจะตื่นขึ้นมาจากภาวะหลับอันเนื่องมาจากการขัดจังหวะของสวิตช์ S1 หรือสวิตช์ S2 และหลังจากแสดงผลไฟวิ่งตามข้อกำหนดแล้วให้ซีพียูเข้าสู่ภาวะหลับอีกครั้ง

ในการอ่านค่ากระแสของวงจรใน Checkpoint 2.1 และ Checkpoint 2.2 ให้อ่านค่ากระแสในขณะที่ไฟวิ่งทุกดวงดับแล้วเท่านั้น และให้แจ้งค่ากระแสที่อ่านได้ของแต่ละ Checkpoint ให้ผู้ตรวจทราบด้วย

15.5 Checkpoint 3

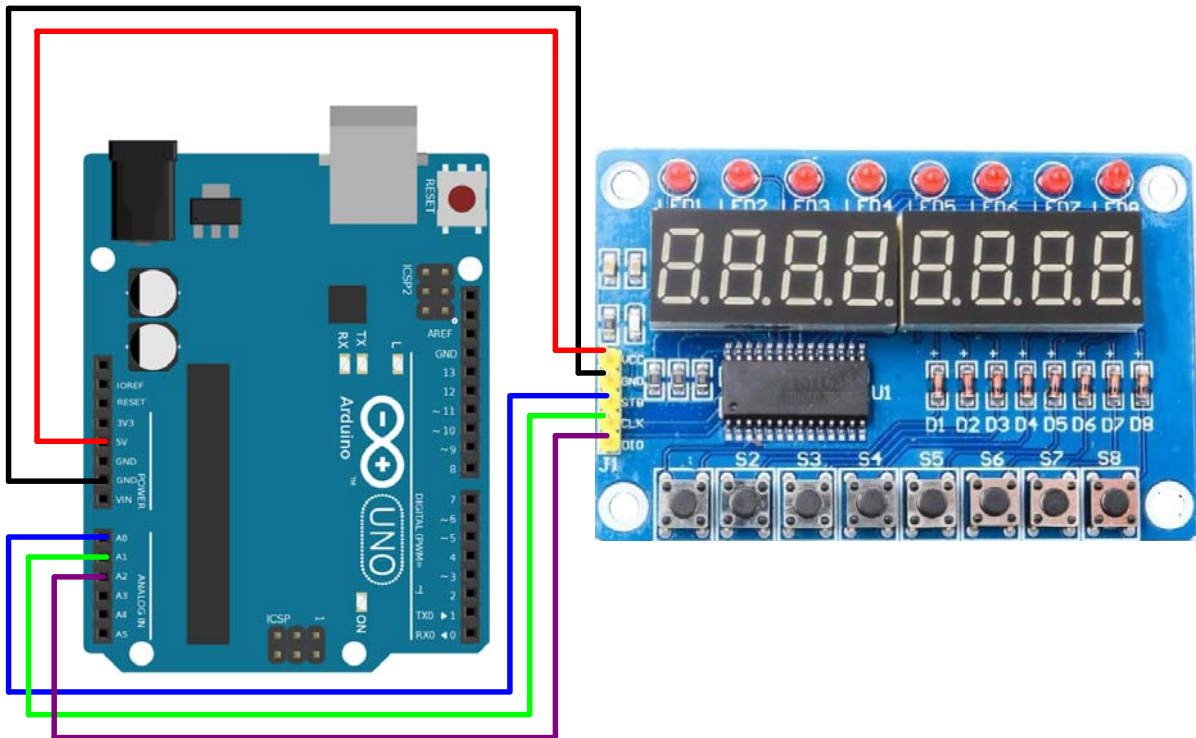
ต่อวงจรตามรูปที่ 15.3 จากนั้นเขียนโปรแกรมเพื่อสร้างวงจรสำหรับนับถอยหลัง 12 วินาที โดยให้แสดงผลค่าเวลานับถอยหลังบนแอลอีดีชนิด 7 ส่วนจำนวนสี่ตัว กำหนดให้สองตัวแรกแสดงค่าเวลาวินาทีในหลักหน่วยและหลักสิบ และแอลอีดีสองตัวหลังแสดงค่าเวลาในหลัก 1/100 วินาที (แสดงผลได้ตั้งแต่ 00-99) โดยให้การแสดงผลค่าเวลาเปลี่ยนครั้งละ 1/100 วินาที (ดูคลิปตัวอย่างประกอบใน LMS) กำหนดให้วงจรมีคุณสมบัติการทำงานดังต่อไปนี้

- เมื่อเริ่มจ่ายไฟให้กับบอร์ด ให้วงจรแสดงผลดังรูปที่ 15.4 (a) จากนั้นหากมีการกดสวิตช์ S8 บนบอร์ด TM1638 จะทำให้วงจรเริ่มนับถอยหลังจนกระทั่งถึงค่าศูนย์ หลังจากนั้นให้กระพริบเลข 00-00 จำนวน สองครั้ง ดังรูปที่ 15.4 (b) (ดูคลิปตัวอย่างประกอบใน LMS) แล้วจึงดับการแสดงผลของแอลอีดี 7 เซกเมนต์ทุกตัวและจากนั้นสั่งให้ตัวประมวลผลเข้าสู่ภาวะหลับ

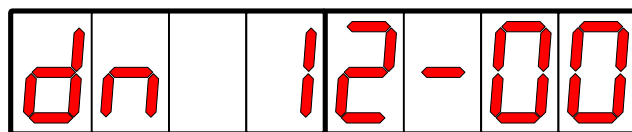
- เมื่อเริ่มจ่ายไฟให้กับบอร์ด ให้วงจรแสดงผลดังรูปที่ 15.4 (a) แต่หากไม่มีการกดสวิตช์เป็นเวลา 5 วินาที ให้ดับการแสดงผลของแอลอีดีชนิด 7 ส่วนทุกตัวแล้วให้วงจรเข้าสู่ภาวะหลับ

ข้อมูลสำหรับ TA

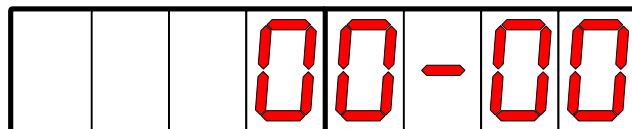
ในการส่งงานกับ TA กำหนดให้ผู้ส่งงานทำการวัดกระแสของวงจรขณะที่วงจรทำงานปกติ และเทียบกับตอนที่ตัวประมวลผลเข้าสู่ภาวะหลับด้วยว่ามีการกินกระแสต่างกันอย่างไร



รูปที่ 15.3 การต่อวงจรสำหรับ Checkpoint 3 - Checkpoint 4



(a)



(b)

รูปที่ 15.4 ตัวอย่างการติดของแอลอีดีชนิด 7 ส่วนของ Checkpoint 3

15.6 Checkpoint 4

ให้สร้างวงจรนับซึ่งมีการทำงานสองโหมด ได้แก่ โหมดถอยหลัง 12 วินาที และ โหมดนับเดินหน้า 12 วินาที โดยกำหนดให้วงจรมีคุณสมบัติการทำงานดังต่อไปนี้

- เมื่อเริ่มจ่ายไฟให้กับบอร์ด ให้วงจรแสดงผลดังรูปที่ 15.5 (a) แต่หากไม่มีการกดสวิตช์ใด ๆ เป็นเวลา 5 วินาที ให้ดับการแสดงผลของแอลอีดีชนิด 7 ส่วนทุกตัวแล้วให้วงจรเข้าสู่ภาวะหลับ

- เมื่อเริ่มจ่ายไฟให้กับบอร์ด ให้วงจรแสดงผลดังรูปที่ 15.5 (a) จากนั้นหากมีการกดสวิตช์ S8 บนบอร์ด TM1638 จะทำให้วงจรเริ่มนับถอยหลังจนกระทั่งถึงค่าศูนย์ หลังจากนั้นให้กระพริบเลข 00-00 จำนวน สองครั้ง ดังรูปที่ 15.5 (b) (ดูคลิปตัวอย่างประกอบใน LMS) แล้วจึงดับการแสดงผลของแอลอีดี 7 เซกเมนต์ทุกตัวและจากนั้นสั่งให้ตัวประมวลผลเข้าสู่ภาวะหลับ

- เมื่อเริ่มจ่ายไฟให้กับบอร์ด ให้วงจรแสดงผลดังรูปที่ 15.4 (a) จากนั้นหากมีการกดสวิตช์ S1 บนบอร์ด TM1638 ให้วงจรเปลี่ยนมาสู่โหมดการนับไปหน้า โดยแสดงผลตามรูปที่ 15.4 (c)

- หากวงจรอยู่ในโหมดการนับไปหน้าและมีการแสดงผลตามรูปที่ 15.4 (c) แล้ว แต่หลังจากนั้นไม่มีการกดสวิตช์ใดเลยเป็นเวลา 5 วินาที ให้ดับการแสดงผลของแอลอีดีชนิด 7 ส่วนทุกตัวแล้วให้วงจรเข้าสู่ภาวะหลับ

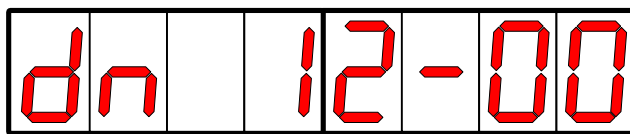
- หากวงจรอยู่ในโหมดการนับไปหน้าและมีการแสดงผลตามรูปที่ 15.4 (c) แล้ว แต่หลังจากนั้นมีการกดสวิตช์ S8 บนบอร์ด TM1638 ให้วงจรเริ่มนับไปหน้าจนกระทั่งถึงค่าสูงสุด หลังจากนั้นให้กระพริบค่าสุดท้าย 2 ครั้งแล้วดับการแสดงผล หลังจากนั้นหากตรวจพบว่าไม่มีการกดสวิตช์ใดเลยเป็นเวลา 5 วินาที ให้สั่งการให้ตัวประมวลผลเข้าสู่ภาวะหลับ

- ก่อนเข้าสู่ภาวะหลับให้เปิดการทำงานของวงจร Watchdog timer และตั้งให้มีการรับการขัดจังหวะของ Watchdog ทุก ๆ 4 วินาที และภายในรูทีนบริการการขัดจังหวะของ Watchdog ให้มีการอ่านค่าคีย์ของบอร์ด TM1638 เพื่อให้ตัวประมวลผลสามารถตื่นจากภาวะหลับและสามารถกลับมาสู่ภาวะการทำงานปกติได้หากมีการกดปุ่มใด ๆ บนบอร์ด TM1638

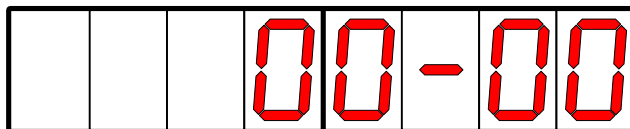
- เมื่อตัวประมวลผลมีการตื่นจากภาวะหลับ ให้วงจรนับกลับมาทำงานยังโหมดการทำงานล่าสุดก่อนหน้านั้น

ข้อมูลสำหรับ TA

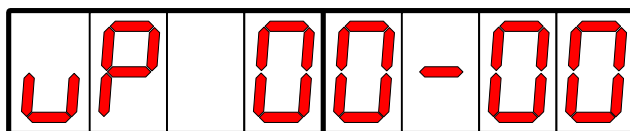
ในการส่งงานกับ TA กำหนดให้ผู้ส่งงานทำการวัดกระแสของวงจรขณะที่วงจรทำงานปกติ และเทียบกับตอนที่ตัวประมวลผลเข้าสู่ภาวะหลับด้วยว่ามีการกินกระแสต่างกันอย่างไร



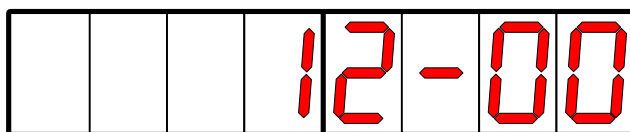
(a)



(b)



(c)



(d)

รูปที่ 15.5 ตัวอย่างการติดของแอลอีดีชนิด 7 ส่วนของ Checkpoint 4

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