**ใบงานการทดลอง LED 7 ส่วน**

**๑ วัตถุประสงค์เชิงพฤติกรรม**

**๑.๑ วัตถุประสงค์เชิงพฤติกรรมทางด้าน hardware**

๑) เพื่อให้นักศึกษาสามารถอธิบายการทำงานของ LED 7 ส่วนได้

๒) เพื่อให้นักศึกษา วาดวงจรเชื่อมต่อ LED 7 ส่วน เข้ากับไมโครคอนโทรลเลอร์ได้

๓) เพื่อให้นักศึกษาประกอบวงจรเชื่อมต่อ LED 7 ส่วน เข้ากับไมโครคอนโทรลเลอร์ได้

๔) เพื่อให้นักศึกษาสามารถทดสอบการทำงานของ LED 7 ส่วน ได้

**๑.๒ วัตถุประสงค์เชิงพฤติกรรมทางด้าน software**

๑) เพื่อให้นักศึกษาสามารถอธิบายหลักการเขียนโปรแกรมควบคุม LED 7 ส่วนได้

๒) เพื่อให้นักศึกษาสามารถเขียนโปรแกรมควบคุม LED 7 ส่วนได้

๓) เพื่อให้นักศึกษาสามารถเขียนโปรแกรมควบคุม LED 7 ส่วนหลายหลัก ให้ปรากฏเป็นตัวเลขตามต้องการได้

**๒. เนื้อหา**

**๒.๑ คุณสมบัติและการทำงานของ LED 7 ส่วน**

ตัวแสดงผลแบบ LED 7 ส่วน เป็นอุปกรณ์ที่ใช้สำหรับการแสดงตัวเลขและสัญลักษณ์บางอย่าง ภายในประกอบด้วย LED จำนวน 7 ตัว (หรือมากกว่า) การทำงานมีลักษณะตรงไปตรงมา คือควบคุมให้ LED ติดหรือดับเพื่อแสดงข้อมูลต่อผู้ใช้ มีลักษณะดังรูปที่ ๑



**รูปที่ ๑** ลักษณะของ LED 7 ส่วน

เพื่อความประหยัด จะมีการต่อขาด้านหนึ่งของตัวแสดงผลแบบ LED 7 ส่วนไว้ร่วมกัน โดยสามารถทำได้ 2 แบบ คือการนำขา A หรือ K ของ LED 7 ส่วนมาต่อร่วมกัน เรียกว่า แอโนดร่วม (Common Anode) และ แคโทดร่วม (Common Cathode) ตามลำดับ ดังรูปที่ ๒



(ก)



(ข)

**รูปที่ ๒** การต่อตัวแสดงผล LED 7 ส่วนแบบต่างๆ (ก) แบบแอโนดร่วม (ข) แบบแคโทดร่วม

**๒.๒ การควบคุม LED 7 ส่วนให้ปรากฏเป็นเลขและอักษรต่างๆ**

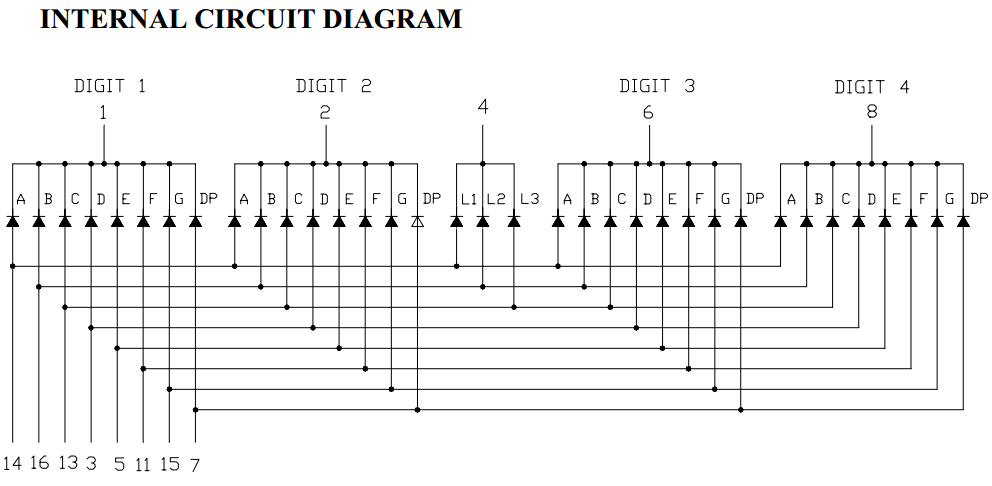
**คำถาม** จงเติมค่าลอจิก 0 หรือ 1 เพื่อให้ LED สว่างเป็นตัวเลขหรือตัวอักษรด้านซ้ายมือ (7 segment เป็นชนิด Common Cathode)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Digit | Segments | | | | | | |
| A | b | C | D | e | f | g |
|  | **1** | **1** | **1** | **1** | **1** | **1** | **0** |
|  | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
|  | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
|  | 1 | 1 | 1 | 1 | 0 | 0 | 1 |
|  | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
|  | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
|  | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
|  | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
|  | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
|  | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
|  | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
|  | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
|  | 1 | 0 | 0 | 1 | 1 | 1 | 0 |
|  | 0 | 1 | 1 | 1 | 1 | 0 | 1 |
|  | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
|  | 1 | 0 | 0 | 0 | 1 | 1 | 1 |

**๒.๓ การควบคุม LED 7 ส่วนชนิด 4 หลัก**



**รูปที่ ๓** ตำแหน่ง Segments และ Digit ต่างๆ

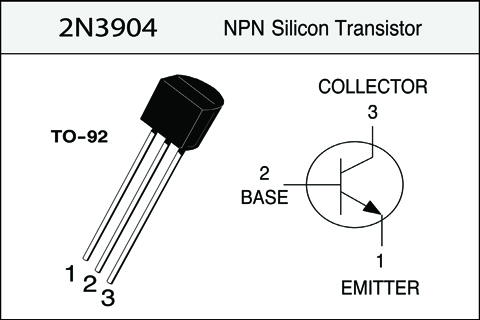


**รูปที่ ๔** วงจรภายในและตำแหน่งการต่อขาใช้งานของ LED 7 ส่วน เบอร์ LTC-4727JR

**๓. ส่วน hardware ของการทดลอง**



**รูปที่ ๕** วงจรพื้นฐานสำหรับการเชื่อมต่อ LED 7 ส่วน 4 หลัก



**รูปที่ ๖** การจัดวางขาของทรานซิสเตอร์ 2N3904

**๔. ส่วน software ของการทดลอง**

๔.๑ เขียนโปรแกรมควบคุมให้ LED 7 ส่วนติดสว่างเป็นตัวเลขต่างๆ ครั้งละ 1 หลัก

**ในฟังก์ชัน setup**

* กำหนดให้ขา D2 – D13 เป็นขาสัญญาณดิจิตอล แบบ เอาต์พุต
* ส่งค่าออกทางขา D2 - D9 เพื่อควบคุมการติด-ดับของตัวเลข 7 ส่วน ให้เป็นเลขต่างๆ
* ส่งค่า 1 ออกทางขา D10 – D13 เพื่อเลือกหลักที่จะแสดงผล

๔.๒ เขียนโปรแกรมควบคุมให้ LED 7 ส่วนติดสว่างเป็นตัวเลขต่างๆ ครบทุกหลัก

**ในฟังก์ชัน setup**

* กำหนดให้ขา D2 – D13 เป็นขาสัญญาณดิจิตอล แบบ เอาต์พุต

**ในฟังก์ชัน loop**

* ส่งค่าออกทางขา D2 - D9 เพื่อควบคุมการติด-ดับของตัวเลข 7 ส่วน ในหลักหน่วย ให้เป็นตัวเลข 8
* ส่งค่า 1 ออกทางขา D10 เพื่อให้หลักหน่วยสว่าง
* รอเวลา 25 มิลลิวินาที (หรือปรับได้ตามความเหมาะสม)
* ส่งค่าออกทางขา D2 - D9 เพื่อควบคุมการติด-ดับของตัวเลข 7 ส่วน หลักสิบ ให้เป็นตัวเลข 8
* ส่งค่า 1 ออกทางขา D11 เพื่อให้หลักสิบสว่าง
* รอเวลา 25 มิลลิวินาที (หรือปรับได้ตามความเหมาะสม)
* ส่งค่าออกทางขา D2 - D9 เพื่อควบคุมการติด-ดับของตัวเลข 7 ส่วน หลักร้อย ให้เป็นตัวเลข 8
* ส่งค่า 1 ออกทางขา D12 เพื่อให้หลักร้อยสว่าง
* รอเวลา 25 มิลลิวินาที (หรือปรับได้ตามความเหมาะสม)
* ส่งค่าออกทางขา D2 - D9 เพื่อควบคุมการติด-ดับของตัวเลข 7 ส่วน หลักพัน ให้เป็นตัวเลข 8
* ส่งค่า 1 ออกทางขา D11 เพื่อให้หลักพันสว่าง

๔.๓ เขียนโปรแกรมควบคุมให้ LED 7 ส่วนติดสว่างเป็นตัวเลขต่างๆ ครบทุกหลัก เป็นตัวเลขต่างๆ ตามต้องการ

* เป็นตัวเลขคงที่ หรือ เปลี่ยนตามเวลา เช่น นาฬิกา

**โปรแกรมควบคุมให้ LED 7 เลข 0-9**

**int led\_array[16][7] = {**

**{ 1,1,1,1,1,1,0 }, // = 0**

**{ 0,1,1,0,0,0,0 }, // = 1**

**{ 1,1,0,1,1,0,1 }, // = 2**

**{ 1,1,1,1,0,0,1 }, // = 3**

**{ 0,1,1,0,0,1,1 }, // = 4**

**{ 1,0,1,1,0,1,1 }, // = 5**

**{ 1,0,1,1,1,1,1 }, // = 6**

**{ 1,1,1,0,0,0,0 }, // = 7**

**{ 1,1,1,1,1,1,1 }, // = 8**

**{ 1,1,1,0,0,1,1 }, // = 9**

**{ 1,1,1,0,1,1,1 }, // = A**

**{ 0,0,1,1,1,1,1 }, // = b**

**{ 0,0,0,1,1,0,1 }, // = c**

**{ 0,1,1,1,1,0,1 }, // = d**

**{ 1,0,0,1,1,1,1 }, // = E**

**{ 1,0,0,0,1,1,1 } // = F**

**};**

**int digit1 = 13;**

**int digit2 = 12;**

**int digit3 = 11;**

**void setup() {**

**pinMode(2, OUTPUT);**

**pinMode(3, OUTPUT);**

**pinMode(4, OUTPUT);**

**pinMode(5, OUTPUT);**

**pinMode(6, OUTPUT);**

**pinMode(7, OUTPUT);**

**pinMode(8, OUTPUT);**

**pinMode(9, OUTPUT);**

**pinMode(digit1, OUTPUT);**

**pinMode(digit2, OUTPUT);**

**pinMode(digit3, OUTPUT);**

**}**

**void loop() {**

**for (int counter = 0; counter < 16; ++counter)**

**{**

**digitalWrite(digit2, 1);**

**delay(1000);**

**led\_Write(counter);**

**for (int i = counter; i > 16; i= i + 1){**

**digitalWrite(digit1, 1);**

**led\_Write(i);**

**}**

**}**

**}**

**void led\_Write(int number){**

**int pin= 2;**

**for (int j = 0; j < 16; j++) {**

**digitalWrite(pin, led\_array[number][j]);**

**pin++;**

**}**

**}**