**การทดสอบการทำงานของเครื่องแสดงผลอักษรเบรลล์**

**1. การทดสอบทางไฟฟ้าของสวิตช์ปุ่มกด**

1.1 สวิตช์ปุ่มกดอักษรเบรลล์ (Braille Key)

สวิตช์ปุ่มกดอักษรเบรลล์นั้นมีทั้งหมด 8 ปุ่ม แต่ละปุ่มนั้นจะต่ออนุกรมกับตัวต้านทานและต่อเข้ากับขาอินพุตของไมโครคอนโทรลเลอร์ ในขณะที่ไมโครคอนโทรลเลอร์ทำงานนั้น ค่าสถานะของสวิตช์ปุ่มกดอักษรเบรลล์ทุก ๆ ปุ่มจะมีค่าเป็น 1 เมื่อทำการกดสวิตช์ ค่าสถานะของสวิตช์นั้นจะเปลี่ยนเป็น 0 การทดสอบนี้ได้ทำการอ่านค่าสถานะของสวิตช์ในขณะที่ไมโครคอนโทรลเลอร์ทำงาน ผลการทดสอบได้ค่าดังนี้

|  |  |  |
| --- | --- | --- |
| ตำแหน่งของปุ่มกดอักษรเบรลล์ | สถานะขณะกดปุ่ม | สถานะขณะปล่อยปุ่ม |
| B1 | 0 | 1 |
| B2 | 0 | 1 |
| B3 | 0 | 1 |
| B4 | 0 | 1 |
| B5 | 0 | 1 |
| B6 | 0 | 1 |
| B7 | 0 | 1 |
| B8 | 0 | 1 |

จากตารางผลการทดสอบสรุปได้ว่าสวิตช์ปุ่มกดอักษรเบรลล์นั้นสามารถใช้งานได้ทุกปุ่ม นอกจากนั้น ไมโครคอนโทรลเลอร์สามารถอ่านค่าจากสวิตช์ปุ่มกดอักษรเบรลล์ได้ครบทุกปุ่ม และการกดปุ่มอักษรเบรลล์นั้นจะทำให้สถานะที่อ่านได้จากไมโครคอนโทรลเลอร์เปลี่ยนจาก 1 ไปเป็น 0 ทั้ง 8 ปุ่ม

1.2 สวิตช์ปุ่มกดคำสั่ง (Command Key)

สวิตช์ปุ่มกดคำสั่งนั้นมีทั้งหมด 6 ปุ่ม แต่ละปุ่มนั้นจะต่ออนุกรมกับตัวต้านทานและต่อเข้ากับขาอินพุตของไมโครคอนโทรลเลอร์ ในขณะที่ไมโครคอนโทรลเลอร์ทำงานนั้น ค่าสถานะของสวิตช์ปุ่มกดคำสั่งทุก ๆ ปุ่มจะมีค่าเป็น 1 เมื่อทำการกดสวิตช์ ค่าสถานะของสวิตช์นั้นจะเปลี่ยนเป็น 0 การทดสอบนี้ได้ทำการอ่านค่าสถานะของสวิตช์ในขณะที่ไมโครคอนโทรลเลอร์ทำงาน ผลการทดสอบได้ค่าดังนี้

|  |  |  |
| --- | --- | --- |
| ตำแหน่งของปุ่มกดคำสั่ง | สถานะขณะกดปุ่ม | สถานะขณะปล่อยปุ่ม |
| C1 | 0 | 1 |
| C2 | 0 | 1 |
| C3 | 0 | 1 |
| C4 | 0 | 1 |
| C5 | 0 | 1 |
| C6 | 0 | 1 |

จากตารางผลการทดสอบสรุปได้ว่าสวิตช์ปุ่มกดคำสั่งนั้นสามารถใช้งานได้ทุกปุ่ม นอกจากนั้น ไมโครคอนโทรลเลอร์สามารถอ่านค่าจากสวิตช์ปุ่มกดลูกศรได้ครบทุกปุ่ม และการกดปุ่มคำสั่งนั้นจะทำให้สถานะที่อ่านได้จากไมโครคอนโทรลเลอร์เปลี่ยนจาก 1 ไปเป็น 0 ทั้ง 6 ปุ่ม

1.3 สวิตช์ปุ่มกดลูกศร (Thumb Key)

สวิตช์ปุ่มกดลูกศรนั้นมีทั้งหมด 4 ปุ่ม แต่ละปุ่มนั้นจะต่ออนุกรมกับตัวต้านทานและต่อเข้ากับขาอินพุตของไมโครคอนโทรลเลอร์ ในขณะที่ไมโครคอนโทรลเลอร์ทำงานนั้น ค่าสถานะของสวิตช์ปุ่มกดลูกศรทุก ๆ ปุ่มจะมีค่าเป็น 1 เมื่อทำการกดสวิตช์ ค่าสถานะของสวิตช์นั้นจะเปลี่ยนเป็น 0 การทดสอบนี้ได้ทำการอ่านค่าสถานะของสวิตช์ในขณะที่ไมโครคอนโทรลเลอร์ทำงาน ผลการทดสอบได้ค่าดังนี้

|  |  |  |
| --- | --- | --- |
| ตำแหน่งของปุ่มกดลูกศร | สถานะขณะกดปุ่ม | สถานะขณะปล่อยปุ่ม |
| T1 | 0 | 1 |
| T2 | 0 | 1 |
| T3 | 0 | 1 |
| T4 | 0 | 1 |

จากตารางผลการทดสอบสรุปได้ว่าสวิตช์ปุ่มกดลูกศรนั้นสามารถใช้งานได้ทุกปุ่ม นอกจากนั้น ไมโครคอนโทรลเลอร์สามารถอ่านค่าจากสวิตช์ปุ่มกดลูกศรได้ครบทุกปุ่ม และการกดปุ่มลูกศรนั้นจะทำให้สถานะที่อ่านได้จากไมโครคอนโทรลเลอร์เปลี่ยนจาก 1 ไปเป็น 0 ทั้ง 6 ปุ่ม

1.4 สวิตช์ปุ่มกดเว้นวรรค (Spacebar Key)

สวิตช์ปุ่มกดเว้นวรรคนั้นมีทั้งหมด 2 ปุ่ม ทั้งสองปุ่มนั้นทำหน้าที่เดียวกันและจะต่ออนุกรมกับตัวต้านทานและต่อเข้ากับขาอินพุตของไมโครคอนโทรลเลอร์ ในขณะที่ไมโครคอนโทรลเลอร์ทำงานนั้น ค่าสถานะของสวิตช์ปุ่มกดเว้นวรรคจะมีค่าเป็น 1 เมื่อทำการกดสวิตช์ ค่าสถานะของสวิตช์นั้นจะเปลี่ยนเป็น 0 การทดสอบนี้ได้ทำการอ่านค่าสถานะของสวิตช์ในขณะที่ไมโครคอนโทรลเลอร์ทำงาน ผลการทดสอบได้ค่าดังนี้

|  |  |  |
| --- | --- | --- |
| ตำแหน่งของปุ่มกดเว้นวรรค | สถานะขณะกดปุ่ม | สถานะขณะปล่อยปุ่ม |
| SP1 | 0 | 1 |
| SP2 | 0 | 1 |

จากตารางผลการทดสอบสรุปได้ว่าสวิตช์ปุ่มกดเว้นวรรคนั้นสามารถใช้งานได้ นอกจากนั้น ไมโครคอนโทรลเลอร์สามารถอ่านค่าจากสวิตช์ปุ่มกดเว้นวรรคที่เปลี่ยนสถานะจาก 1 ไปเป็น 0 ได้

**2. การทดสอบทางซอฟต์แวร์ของสวิตช์ปุ่มกด**

ไมโครคอนโทรลเลอร์นั้นจะอ่านค่าสถานะของสวิตช์ทั้งหมดบนเครื่องแสดงผลอักษรเบรลล์มาทำการประมวลผล และทำการเข้ารหัสเพื่อทำการส่งต่อไปยังอุปกรณ์ที่ทำหน้าที่สื่อสารกับคอมพิวเตอร์ต่อไป การทดสอบทางซอฟต์แวร์นั้นดำเนินการโดยการใช้อุปกรณ์แปลงสัญญาณ USB-to-TTL ในการทำการดักจับข้อมูลที่ไมโครคอนโทรลเลอร์ส่งไปยังอุปกรณ์ที่ทำหน้าที่สื่อสารกับคอมพิวเตอร์ออกมาในรูปแบบของข้อมูลฐาน 16 จำนวน 5 ไบต์ ได้แก่ ไบต์ของเบรลล์ (Braille Byte) ไบต์ของเว้นวรรค (Spacebar Byte) ไบต์ของคำสั่ง (Command Byte) ไบต์ของลูกศร (Thumb Byte) และไบต์ของเคอร์เซอร์ (Cursor Byte) ซึ่งอุปกรณ์ที่ทำหน้าที่ติดต่อสื่อสารกับคอมพิวเตอร์นั้นจะนำค่าที่ได้นี้ไปทำการสั่งพิมพ์ตัวอักษรบนหน้าจอคอมพิวเตอร์ สำหรับผลการทดสอบนั้นได้ผลดังนี้

2.1 สวิตช์ปุ่มกดอักษรเบรลล์ (Braille Key)

สวิตช์ปุ่มกดอักษรเบรลล์นั้นทำหน้าที่ในการพิมพ์ข้อความ ตัวเลข หรือตัวอักษรต่าง ๆ ไปยังคอมพิวเตอร์ผ่านการกดปุ่มทั้ง 8 ปุ่มเพื่อสร้างเป็นรูปแบบที่แตกต่างกันได้สูงสุดถึง 255 รูปแบบ ซึ่งรูปแบบเหล่านั้นจะถูกนำไปประมวลผลและส่งต่อไปยังคอมพิวเตอร์ต่อไป การทดสอบนี้จะทำการกดปุ่มอักษรเบรลล์และสังเกตผลที่ขึ้นบนหน้าจอคอมพิวเตอร์ ซึ่งได้ผลการทดลองดังนี้

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ตำแหน่งของ  ปุ่มกดอักษรเบรลล์ | ไบต์ 0 | ไบต์  1 | ไบต์ 2 | ไบต์ 3 | ไบต์ 4 | อักษรภาษาไทย | อักษรภาษาอังกฤษ |
| 1 | 0x01 | 0 | 0 | 0 | 0 | อะ | a |
| 2 | 0x02 | 0 | 0 | 0 | 0 | ๆ | 1 |
| 3 | 0x04 | 0 | 0 | 0 | 0 | อ็ | ‘ |
| 4 | 0x08 | 0 | 0 | 0 | 0 |  | ` |
| 5 | 0x10 | 0 | 0 | 0 | 0 |  | “ |
| 6 | 0x20 | 0 | 0 | 0 | 0 |  | , |
| 7 | 0x40 | 0 | 0 | 0 | 0 |  | ; |
| 8 | 0x80 | 0 | 0 | 0 | 0 |  |  |
| 2, 3, 5, 6, 7 | 0x76 | 0 | 0 | 0 | 0 | ๗ |  |
| 1, 5, 6, 8 | 0xB1 | 0 | 0 | 0 | 0 | ใ |  |

ผลการทดสอบสรุปได้ว่าสวิตช์ปุ่มกดอักษรเบรลล์ทั้ง 8 ปุ่มสามารถพิมพ์ข้อความทั้งภาษาไทยและอังกฤษได้อย่างไม่มีปัญหา นั่นหมายถึงการรับข้อมูลของอุปกรณ์สื่อสารกับคอมพิวเตอร์ และการส่งข้อมูลของไมโครคอนโทรลเลอร์นั้นสามารถทำงานได้ปกตินั่นเอง

2.2 สวิตช์ปุ่มกดคำสั่ง (Command Key)

สวิตช์ปุ่มกดคำสั่งใช้สำหรับเรียกใช้คำสั่งลัดหรือคำสั่งพิเศษต่าง ๆ บนคอมพิวเตอร์ การทดสอบนี้จะทำการกดปุ่มคำสั่งในรูปแบบต่าง ๆ และสังเกตคำสั่งที่คอมพิวเตอร์เรียกใช้ ซึ่งมีผลการทดลองดังนี้

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ตำแหน่งของ  ปุ่มกดคำสั่ง | ไบต์ 0 | ไบต์  1 | ไบต์ 2 | ไบต์ 3 | ไบต์ 4 | รูปแบบการกระทำ |
| 1 | 0 | 0 | 0x01 | 0 | 0 | สั่งกดปุ่ม ENTER |
| 2 | 0 | 0 | 0x02 | 0 | 0 | สั่งกดปุ่ม BACKSPACE |
| 3 | 0 | 0 | 0x04 | 0 | 0 | สั่งกดปุ่ม TAB |
| 4 | 0 | 0 | 0x08 | 0 | 0 | สั่งกดปุ่ม INSERT |
| 5 | 0 | 0 | 0x10 | 0 | 0 |  |
| 6 | 0 | 0 | 0x20 | 0 | 0 |  |
| 2, 3 | 0 | 0 | 0x06 | 0 | 0 | สั่งกดปุ่มเปลี่ยนภาษา |
| 2, 4, 6 | 0 | 0 | 0x2A | 0 | 0 | สั่งกดปุ่ม F10 |

ผลการทดสอบสรุปได้ว่าสวิตช์ปุ่มกดคำสั่งทั้ง 6 ปุ่มสามารถกดเป็นรูปแบบเพื่อเรียกใช้คำสั่งต่าง ๆ บนคอมพิวเตอร์ได้

2.3 สวิตช์ปุ่มกดลูกศร (Thumb Key)

สวิตช์ปุ่มกดลูกศรนั้นใช้สำหรับเลื่อนตำแหน่งของคาเรท (Caret) บนคอมพิวเตอร์ การทดสอบนี้จะทำการกดปุ่มลูกศรทั้ง 4 ปุ่ม และสังเกตผลบนหน้าจอคอมพิวเตอร์ ซึ่งมีผลการทดลองดังนี้

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ตำแหน่งของ  ปุ่มกดลูกศร | ไบต์ 0 | ไบต์  1 | ไบต์ 2 | ไบต์ 3 | ไบต์ 4 | รูปแบบการกระทำ |
| 1 | 0 | 0 | 0 | 0x01 | 0 | กดปุ่มลูกศรขึ้น / เลื่อนขึ้น |
| 2 | 0 | 0 | 0 | 0x02 | 0 | กดปุ่มลูกศรซ้าย / เลื่อนซ้าย |
| 3 | 0 | 0 | 0 | 0x04 | 0 | กดปุ่มลูกศรขวา / เลื่อนขวา |
| 4 | 0 | 0 | 0 | 0x08 | 0 | กดปุ่มลูกศรลง / เลื่อนลง |

ผลการทดสอบสรุปได้ว่าสวิตช์ปุ่มกดลูกศรทั้ง 4 ปุ่มสามารถสั่งให้ตำแหน่งของคาเรทเปลี่ยนไปได้ตามที่ออกแบบไว้

2.4 สวิตช์ปุ่มกดเว้นวรรค (Spacebar Key)

สวิตช์ปุ่มกดเว้นวรรคนั้นทำหน้าที่สองอย่างหลัก ๆ ได้แก่ทำหน้าที่พิมพ์เว้นวรรคในงานเอกสารต่าง ๆ บนคอมพิวเตอร์ และยังสามารถกดปุ่มเว้นวรรคร่วมกับปุ่มกดอักษรเบรลล์เพื่อสร้างรูปแบบใหม่สำหรับเรียกใช้คำสั่งพิเศษต่าง ๆ เพิ่มเติมได้อีกด้วย ซึ่งการทดลองนี้ได้ทำการทดสอบกดปุ่มเว้นวรรคร่วมกับปุ่มกดอักษรเบรลล์ทั้ง 8 ปุ่ม เพื่อทดสอบการเรียกใช้คำสั่งลัดต่าง ๆ ซึ่งได้ผลการลดลองดังนี้

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ตำแหน่งของ  ปุ่มกดอักษรเบรลล์ | ไบต์ 0 | ไบต์  1 | ไบต์ 2 | ไบต์ 3 | ไบต์ 4 | รูปแบบการกระทำ |
| Space | 0 | 0x01 | 0 | 0 | 0 | พิมพ์เว้นวรรค |
| Space + 1 | 0x01 | 0x01 | 0 | 0 | 0 | CTRL + A / เลือกทั้งหมด |
| Space + 1, 2 | 0x03 | 0x01 | 0 | 0 | 0 | CTRL + B / ตัวหนา |
| Space + 1, 3, 5, 6 | 0x35 | 0x01 | 0 | 0 | 0 | CTRL + Z / Undo |
| Space + 2, 5 | 0x12 | 0x01 | 0 | 0 | 0 | CTRL + N / สร้างไฟล์ใหม่ |
| Space + 2, 3, 5 | 0x16 | 0x01 | 0 | 0 | 0 | เรียก Task Manager |

ผลการทดสอบสรุปได้ว่าสวิตช์ปุ่มกดเว้นวรรคสามารถกดเพื่อพิมพ์ตัวอักษรเว้นวรรค และสามารถกดร่วมกับปุ่มกดอักษรเบรลล์เพื่อสร้างรูปแบบคำสั่งต่าง ๆ ได้

**3. การทดสอบทางซอฟต์แวร์สำหรับการแสดงผลและการขับหน้าจอแสดงผล**

การทดสอบหน้าจอแสดงผลนั้นสามารถดำเนินการได้โดยการทดสอบส่งข้อมูลจากไมโครคอนโทรลเลอร์ไปยังโมดูล HV509 ที่ต่อ Cascade กันอยู่จำนวน 5 โมดูล (สำหรับชับหน้าจอแสดงผลจำนวน 40 เซลล์) แล้วทำการบันทึกรูปแบบการแสดงผลในแต่ละรอบการส่งข้อมูล ซึ่งมีรายละเอียดดังนี้

ผู้ทำการทดลองได้เตรียมข้อมูลที่ใช้ในการส่งจากไมโครคอนโทรลเลอร์คือประโยคคำว่า “Hello” ซึ่งทำการทดสอบโดยการส่งข้อมูลไปยังโมดูล HV509 ทีละตัวอักษร ดังนี้

1. ทำการควบคุมสถานะของขา Latch ให้เป็น 0 ด้วยไมโครคอนโทรลเลอร์ เพื่อทำการเริ่มส่งข้อมูล
2. ทำการส่งข้อมูลตัวอักษร “H” ไปยังโมดูล HV509 ได้ผลลัพธ์การแสดงผลบนหน้าจอดังนี้ (ตำแหน่ง 1 อยู่ทางซ้ายมือสุดของหน้าจอแสดงผล)

**ตำแหน่ง 1 2 3 4 ... 37 38 39 40**

- - - - - - - H

1. ทำการส่งข้อมูลตัวอักษร “E” ไปยังโมดูล HV509 ได้ผลลัพธ์การแสดงผลบนหน้าจอดังนี้ (ตำแหน่ง 1 อยู่ทางซ้ายมือสุดของหน้าจอแสดงผล)

**ตำแหน่ง 1 2 3 4 ... 37 38 39 40**

- - - - - - H E

1. ทำการส่งข้อมูลตัวอักษร “L” ไปยังโมดูล HV509 ได้ผลลัพธ์การแสดงผลบนหน้าจอดังนี้ (ตำแหน่ง 1 อยู่ทางซ้ายมือสุดของหน้าจอแสดงผล)

**ตำแหน่ง 1 2 3 4 ... 37 38 39 40**

- - - - - H E L

1. ทำการส่งข้อมูลตัวอักษร “L” ไปยังโมดูล HV509 ได้ผลลัพธ์การแสดงผลบนหน้าจอดังนี้ (ตำแหน่ง 1 อยู่ทางซ้ายมือสุดของหน้าจอแสดงผล)

**ตำแหน่ง 1 2 3 4 ... 37 38 39 40**

- - - - H E L L

1. ทำการส่งข้อมูลตัวอักษร “O” ไปยังโมดูล HV509 ได้ผลลัพธ์การแสดงผลบนหน้าจอดังนี้ (ตำแหน่ง 1 อยู่ทางซ้ายมือสุดของหน้าจอแสดงผล)

**ตำแหน่ง 1 2 3 4 ... 37 38 39 40**

- - - - E L L O

1. ทำการส่งข้อมูลตัวอักษรเว้นวรรคจำนวน 34 ตัว ไปยังโมดูล HV509 ได้ผลลัพธ์การแสดงผลบนหน้าจอดังนี้ (ตำแหน่ง 1 อยู่ทางซ้ายมือสุดของหน้าจอแสดงผล)

**ตำแหน่ง 1 2 3 4 ... 37 38 39 40**

H E L L - - - -

ผลการทดลองนั้นเป็นไปตามที่ได้ออกแบบไว้ ข้อมูลที่ส่งไปยังโมดูล HV509 นั้นจะค่อย ๆ เลื่อนจากทางขวามาซ้ายครั้งละ 1 ตัวอักษร โดยที่จะต้องทำการส่งข้อมูลให้ครบทั้ง 40 ตัวเพื่อที่จะทำให้ข้อมูลนั้นแสดงผลเต็มหน้าจอ

**4. การปรับแก้ตำแหน่งการแสดงผลของหน้าจอแสดงผลอักษรเบรลล์**

เนื่องจากการออกแบบแผ่นวงจรของเบรลล์เซลล์นั้นมีข้อจำกัดทางด้านพื้นที่ ทำให้การออกแบบการแสดงผลของเบรลล์เซลล์นั้นไม่สามารถทำได้ถูกต้องตามที่ออกแบบเอาไว้ ซึ่งการแก้ไขนั้นสามารถทำได้โดยการปรับแก้ค่าข้อมูลไบนารี่ (Binary Modification) ของข้อมูลที่ได้รับมาจากหน้าจอคอมพิวเตอร์ผ่านทางซอฟต์แวร์อ่านหน้าจอ ยกตัวอย่างข้อมูลเช่น

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ตัวอักษรบนหน้าจอคอมพิวเตอร์ | ค่าไบต์ของข้อมูล | การแสดงผลที่คาดหวัง | การแสดงผลจริง | ค่าไบต์ของการแสดงผลจริง |
| ก | 0x1B | ⠛ | ⢇ | 0x78 |
| A | 0x41 | ⡁ | ⣝ | 0xDD |
| X | 0x6D | ⡭ | ⡔ | 0x54 |

การแสดงผลจากข้อมูลในตารางด้านบนนั้นคลาดเคลื่อนผิดจากที่ออกแบบไว้เนื่องจากข้อจำกัดที่กล่าวไปข้างต้น ทำให้จำเป็นต้องมีการวิเคราะห์หารูปแบบการคลาดเคลื่อนของการแสดงผล รวมถึงการใช้ซอฟต์แวร์ในการแก้ไขปัญหานี้ เพื่อที่จะทำให้สามารถแสดงผลตามรูปแบบที่ได้ออกแบบไว้โดยใช้ข้อมูลชุดเดิมและไม่จำเป็นต้องแก้ไขการออกแบบแผ่นวงจรใด ๆ นั่นเอง

4.1 รูปแบบความคลาดเคลื่อนของการแสดงผล

1) การแสดงผลนั้นมีการกลับสถานะ

ข้อมูลที่ส่งมาจากซอฟต์แวร์อ่านหน้าจอจะให้ค่าไบต์มาหนึ่งค่าต่อหนึ่งตัวอักษร ซึ่งค่าไบต์นั้นจะนำไปสกัดเอาค่าไบนารี่ออกมาใช้สั่งสถานะของก้านเพียโซอิเล็กทริก ปกติแล้วถ้าค่าไบนารี่เป็น 1 การแสดงผลของดอทนั้น ๆ ก็ควรที่จะแสดง (งอขึ้น) แต่การแสดงผลของหน้าจอแสดงผลนี้จะกลับกัน กล่าวคือเมื่อสถานะเป็น 0 ดอทนั้นก็จะแสดง ซึ่งขัดกับรูปแบบข้อมูลที่ได้รับมา

การแก้ไขปัญหานี้สามารถทำได้โดยการแปลงค่าไบต์ที่ได้รับมาให้เป็น Compliment หรือนำค่า 255 ไปลบออกจากไบต์ข้อมูลเดิม ก็จะทำให้การแสดงผลนั้นกลับสถานะเป็นอย่างเดิม

2) ลำดับการเรียงของการแสดงผลในเบรลล์เซลล์ผิดพลาด

ปัญหานี้เกิดจากข้อจำกัดของการออกแบบแผ่นวงจร ทำให้ไม่สามารถกำหนดรูปแบบที่เหมาะสมให้กับการแสดงผลได้ด้วยฮาร์ดแวร์ ปัญหานี้แก้ไขได้โดยการจัดเรียงลำดับของไบนารี่ในไบต์ข้อมูลนั้น ๆ ใหม่ ให้ตรงตามรูปแบบที่ต้องการที่จะให้แสดงผล

จากการทดลองการแสดงผลโดยสั่งให้แสดงผลทีละก้าน มีรายละเอียดดังนี้

**สีเขียว:** จุดที่หน้าจอแสดงผลควรที่จะแสดงผล

**สีแดง:**  จุดของหน้าจอแสดงผลที่ยังไม่ได้ทำการจัดเรียงจุดใหม่

**สีน้ำเงิน:** จุดที่แสดงผลถูกต้องตามรูปแบบอยู่แล้ว

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| รูปแบบการสั่งการแสดงผล | จุด 1 | จุด 2 | จุด 3 | จุด 4 | จุด 5 | จุด 6 | จุด 7 | จุด 8 |
| สั่งให้แสดงจุดเบรลล์ที่ 1 | **UP** |  |  |  |  | **UP** |  |  |
| สั่งให้แสดงจุดเบรลล์ที่ 2 |  | **UP** |  |  |  |  | **UP** |  |
| สั่งให้แสดงจุดเบรลล์ที่ 3 |  |  | **UP** |  |  |  |  | **UP** |
| สั่งให้แสดงจุดเบรลล์ที่ 4 |  |  |  | **UP** |  |  |  |  |
| สั่งให้แสดงจุดเบรลล์ที่ 5 |  |  |  |  | **UP** |  |  |  |
| สั่งให้แสดงจุดเบรลล์ที่ 6 | **UP** |  |  |  |  | **UP** |  |  |
| สั่งให้แสดงจุดเบรลล์ที่ 7 |  | **UP** |  |  |  |  | **UP** |  |
| สั่งให้แสดงจุดเบรลล์ที่ 8 |  |  | **UP** |  |  |  |  | **UP** |

เมื่อทำการสั่งให้จุดที่ 1 แสดงผล จะทำให้จุดที่ 6 นั้นแสดงผลขึ้นมาแทน หรือเมื่อสั่งให้จุดที่ 7 แสดงผล จุดที่ 2 ก็จะขึ้นมาแทนนั่นเอง ดังนั้นจากผลการทดลองจึงทำให้ได้ข้อสรุปการจัดเรียงรูปแบบการแสดงผลของไบต์ข้อมูลใหม่ ดังนี้

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ตำแหน่งของบิตเดิม | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| ตำแหน่งของบิตใหม่ | 2 | 1 | 0 | 4 | 3 | 7 | 6 | 5 |

**5. การทดสอบอุปกรณ์อื่น ๆ บนเครื่องแสดงผลอักษรเบรลล์**

5.1 การทดสอบบัซเซอร์

ทำการทดสอบอุปกรณ์บัซเซอร์โดยการควบคุมด้วยไมโครคอนโทรลเลอร์ การทดสอบมีรายละเอียดดังนี้

|  |  |  |
| --- | --- | --- |
| สถานะของการควบคุม | แรงดัน (V) | สถานะของบัซเซอร์ |
| 0 | 0.0 | ดัง / ทำงาน |
| 1 | 5.0 | เงียบ / ไม่ทำงาน |

เมื่อไมโครคอนโทรลเลอร์ทำการจ่ายสถานะ 0 ให้กับอุปกรณ์บัซเซอร์ พบว่าบัซเซอร์สามารถทำงานได้ปกติ เมื่อไมโครคอนโทรลเลอร์ควบคุมสถานะให้เป็น 1 บัซเซอร์ก็จะหยุดทำงาน