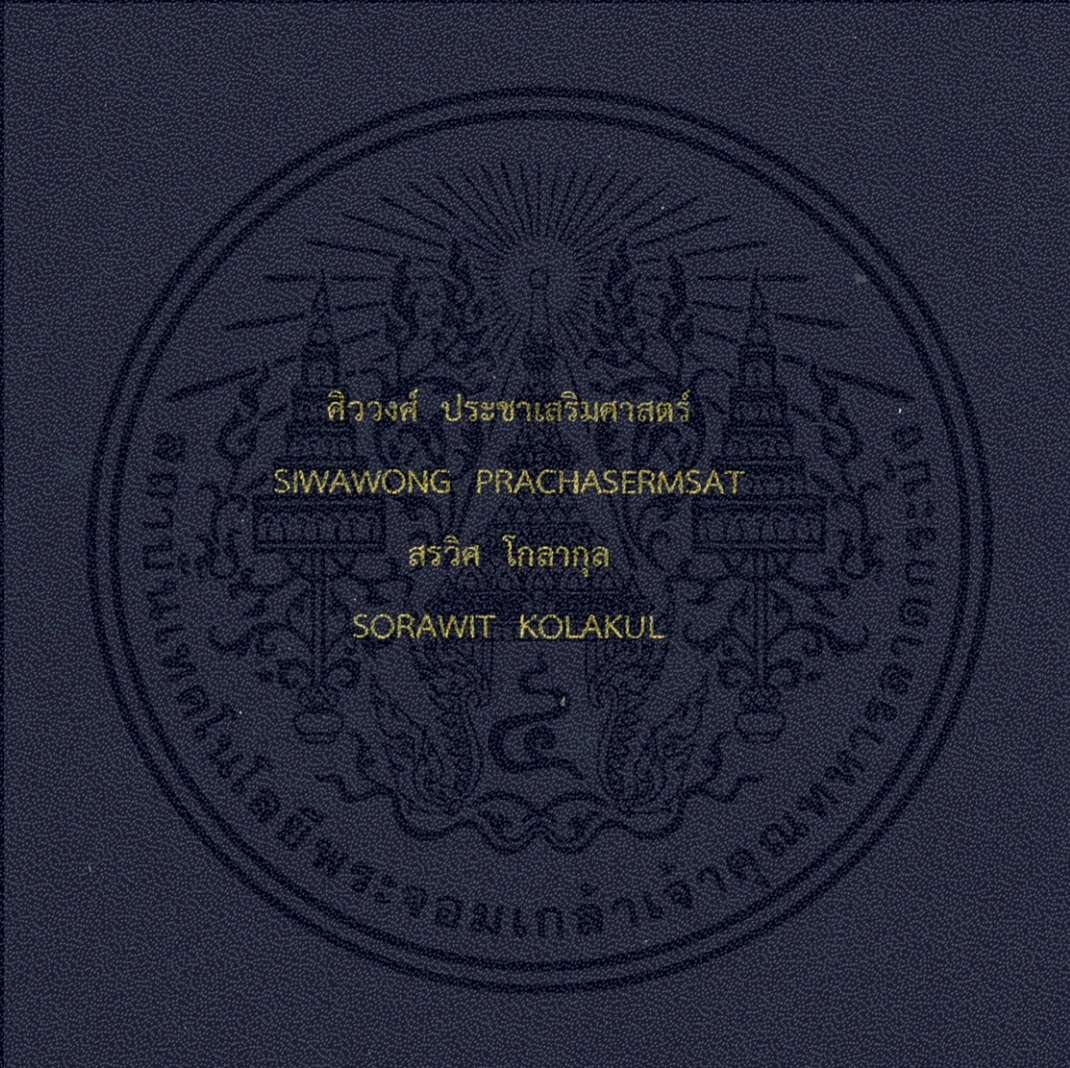


แอนดรอยด์ควบคุมแอลอีดีเมทริกซ์ผ่านบลูทูธ  
ANDRIOD CONTROL LED MATRIX VIA BLUETOOTH



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2556

แอนดรอยด์ควบคุมแอลอีดีเมทริกส์ผ่านบลูทูธ

ANDRIOD CONTROL LED MATRIX VIA BLUETOOTH

โดย

ศิววงศ์ ประชาเสริมศาสตร์

สรวิศ โกลากุล

อาจารย์ที่ปรึกษา

อาจารย์ เฉลิมพันธ์ หวังวิวัฒนา

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2556

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2556

สาขาวิชา วิศวกรรมอิเล็กทรอนิกส์

คณะ วิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง แอนดรอยด์ควบคุมแอลอีดีเมทริกซ์ผ่านบลูทูธ

ANDRIOD CONTROL LED MATRIX VIA BLUETOOTH

ผู้จัดทำ 1. นาย ศิววงศ์ ประชาเสริมศาสตร์ รหัสประจำตัว 53011581

2. นาย สรวิศ โกลากุล รหัสประจำตัว 53011642

ปริญญานิพนธ์นี้ผ่านการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว



(อาจารย์ เลลิมพันธ์ หวังวิวัฒนา)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอนดรอยด์ควบคุมแอลอีดีเมทริกส์ผ่านบลูทูธ

นาย ศิววงศ์ ประชาเสริมศาสตร์ รหัสนักศึกษา 53011581

นาย สรวิต โกลากุล รหัสนักศึกษา 53011642

อาจารย์เฉลิมพันธ์ หวังวิวัฒนา อาจารย์ที่ปรึกษา

ปีการศึกษา 2556

### บทคัดย่อ

ในปฏิญานิพนธ์ฉบับนี้ ได้สร้างและออกแบบระบบแสดงผลผ่านแอลอีดีเมทริกส์โดยการส่งข้อมูลจากแอนดรอยด์โดยผ่านทางบลูทูธ จะประกอบไปด้วย 3 ส่วนหลักดังนี้

1. ส่วนไมโครคอนโทรล 2. ส่วนแอนดรอยด์ และ 3. วงจรขับไฟและแอลอีดีเมทริกส์ ซึ่งส่วนไมโครคอนโทรลและส่วนแอนดรอยด์ได้เชื่อมต่อกันผ่านทางบลูทูธ การทำงานของระบบการแสดงผลผ่านทางแอลอีดีเมทริกส์โดยใช้แอปพลิเคชันแอนดรอยด์ ในการส่งข้อมูล แอปพลิเคชันแอนดรอยด์จะเป็นส่วนคำสั่ง ไมโครคอนโทรลเป็นส่วนประมวลผล และแอลอีดี เมทริกส์ เป็นส่วนแสดงผล บทความนี้เป็นการนำเสนอการประยุกต์ใช้เทคโนโลยีบลูทูธในการควบคุมอุปกรณ์ไฟฟ้าและสร้างระบบ ต้นแบบเพื่อแสดงการทำงานของระบบควบคุมอุปกรณ์ไฟฟ้าผ่านการสื่อสารไร้สายบลูทูธเพื่อให้เกิดความสะดวกสบาย และเพิ่มประสิทธิภาพในการใช้งานอุปกรณ์ไฟฟ้า การทำงานของระบบควบคุมอุปกรณ์ไฟฟ้าผ่านแอปพลิเคชันแอนดรอยด์ สัญญาณควบคุมจะถูกส่งผ่านบลูทูธจากโทรศัพท์เคลื่อนที่ ไปยังไมโครคอนโทรลเลอร์และไมโครคอนโทรลเลอร์ทำการ ประมวลผลเพื่อสั่งการแอลอีดีเมทริกส์ เพื่อแสดงออกมาเป็นข้อความที่กำหนด ด้วยสัญญาณการควบคุม ไมโครคอนโทรลเลอร์ระบบการทำงานนั้น พัฒนาจากโปรแกรมต่างๆเช่น ภาษาซีภาษาไพธอน เป็นต้น และจัดทำเป็นระบบต้นแบบแสดงการทำงานของระบบ การทำงานของระบบควบคุมเป็นไปตามโปรแกรมที่พัฒนาขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ANDRIOD CONTROL LED MATRIX VIA BLUETOOTH

Mr.Siwawong Prachasermsat 53011581

Mr.Sorawit Kolakul 53011642

Mr.Chalernpun Wangwiwathana Adviser

Academic Year 2013

### Abstract

This article presents an application of Bluetooth technology to control LED matrix for Display massage. The model is constructed to indicate the controlling system of the controlling electrical devices via wireless Bluetooth. The purposes of this study are a convenience and an efficient of operating electrical devices. The controlling signals are sent via Bluetooth from application android to a microcontroller. The signals processed by the microcontroller operate the LED matrix attached to the microcontroller. The system is programmed by C and Python languages. The model shows that the system is operated following the set programs.

## กิตติกรรมประกาศ

ขอขอบคุณอาจารย์เฉลิมพันธ์ หวังวิวัฒนา อาจารย์ที่ปรึกษาสำหรับข้อมูลการทำงานในการทำงานครั้งนี้ หากไม่ได้ความเมตตาจากอาจารย์ และพ่อแม่ในการทำงานจะไม่ประสบความสำเร็จเลย และขอขอบคุณความช่วยเหลือจากเพื่อนๆ ในการช่วยเหลือและให้คำปรึกษาในการทำงาน



ศิววงศ์ ประชาเสริมศาสตร์

สรวิศ โกลากุล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

เรื่อง	หน้า
บทคัดย่อ.....	I,II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV,V
สารบัญรูป.....	VI,VII,VIII
สารบัญตาราง.....	VIII
บทที่ 1 บทนำ.....	1
1.1 ที่มาและความสำคัญ.....	1
1.2 จุดประสงค์ของงาน.....	1
1.2.1 เพื่อพัฒนาระบบที่มีการประยุกต์ใช้เทคโนโลยีบลูทูธ และไมโครคอนโทรลเลอร์ในชีวิตประจำวัน.....	1
1.2.2 เพื่อทำการสร้างแผ่นป้ายแสดงตัวอักษร ตามคำสั่งที่ส่งมาจากแอนดรอยด์.....	1
1.3 ขอบเขตของงานที่ทำ.....	1
1.4 ประโยชน์และผลลัพธ์ที่คาดว่าจะได้รับ.....	1
1.4.1 ความรู้ทางด้านระบบมือถือแอนดรอยด์.....	1
1.4.2 การใช้ภาษาจาวา(JAVA) และภาษา C.....	1
1.4.3 การใช้บอร์ด Arduino Uno R3.....	1
1.4.4 วงจรขับและวงจรแอลอีดีเมทริกส์.....	1
บทที่ 2 หลักการหรือทฤษฎีที่เกี่ยวข้องกับการทำงาน.....	2
2.1 แอนดรอยด์(Android).....	2
2.1.1 คุณสมบัติและความสามารถของแอนดรอยด์.....	2
2.1.2 เวอร์ชันของระบบปฏิบัติการแอนดรอยด์.....	2
2.1.3 โครงสร้างของแอนดรอยด์.....	2
2.1.4 ข้อเด่นของแอนดรอยด์.....	2
2.2 Bluetooth.....	10
2.2.1 ระยะเวลาทำการ.....	10
2.2.2 รุ่น.....	10
2.2.3 ข้อดี (Strength).....	11
2.2.4 ข้อเสีย (weakness).....	11
2.2.5 บอร์ด HC-05.....	12
2.2.5.1 คุณสมบัติต่างๆ.....	12
2.2.5.2 รายละเอียดการทำงานต่างๆ.....	12
2.3 บอร์ดอาร์ดูโน(Arduino).....	14

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

เรื่อง	หน้า
23.1 Chip และ IC ภายใน Arduino Board ที่สำคัญ.....	14
23.2 บอร์ด Arduino Uno R3.....	17
24 8x8 LED Matrix.....	19
25 รูปแบบในการส่งข้อมูล (transmission mode).....	22
25.1 การส่งข้อมูลแบบอนุกรม (serial transmission).....	22
26 ชิฟต์รีจิสเตอร์(Shift Register).....	25
26.1 ตัวเลื่อนข้อมูล.....	25
26.2 ตัวเลื่อนข้อมูลแบบเข้าขนานและ ออกขนาน.....	26
26.3 ตัวเลื่อนข้อมูลแบบเข้าขนานออกอนุกรม.....	27
26.4 ตัวเลื่อนข้อมูลแบบส่งเข้าแบบอนุกรม และส่งออกแบบขนาน.....	30
26.5 ตัวเลื่อนข้อมูลชนิดรับเข้าแบบอนุกรมและส่งออกแบบอนุกรม.....	31
26.6 เทคนิคการขยาย Port Output ด้วย Shift Register.....	32
26.7 ไอซี MAX7219.....	34
27 Clock pulse.....	39
28 ภาพรวมของ SPI.....	39
28.1 พื้นฐานการทำงานของ การสื่อสารแบบ SPI.....	39
28.2 ข้อดีของการสื่อสารแบบ SPI.....	40
28.3 ข้อเสียของการสื่อสารแบบ SPI.....	40
บทที่ 3 รายละเอียดในการออกแบบ และการสร้าง.....	42
31 บล็อกไดอแกรม.....	42
32 ส่วนของภาคส่ง.....	43
33 ส่วนของภาครับ.....	44
34 ส่วนของภาคคอนโทรลเลอร์.....	45
35 ส่วนของภาคแสดงผล.....	51
บทที่ 4 การทดลองและผลการทดลอง.....	53
41 การทดลอง.....	53
42 สรุปผลการทดลอง.....	54
บทที่ 5 บทสรุปและวิจารณ์.....	55
51 สรุปผลการออกแบบวงจร.....	55
52 สรุปผลการทดลอง.....	55
53 วิจารณ์ผลการทดลอง.....	55
54 ประโยชน์ที่ได้รับ.....	55

## สารบัญรูป

เรื่อง	หน้า
บทที่ 2 บทสรุปและวิจารณ์.....	2
รูปที่ 2.1 สถาปัตยกรรมแอนดรอยด์.....	4
รูปที่ 2.2 องค์ประกอบการทำงานของแอปพลิเคชัน.....	8
รูปที่ 2.3 โมดูลบลูทูธ HC-05.....	12
รูปที่ 2.4 ซีพียู ATMEGA328P ขนาด 28 ขา.....	14
รูปที่ 2.5 เปรียบเทียบการจัดการหน่วยความจำของสถาปัตยกรรม แบบ Von-Neumann และ Harvard.....	15
รูปที่ 2.6 แสดงขาของบอร์ดอาร์ดูโน.....	16
รูปที่ 2.7 บอร์ด Arduino Uno R3.....	17
รูปที่ 2.8 แสดงหน้าตาโปรแกรมพัฒนาบอร์ดอาร์ดูโน.....	18
รูปที่ 2.9 แสดงตัวอย่างอุปกรณ์ 8x8 แอลอีดีเมทริกซ์ ที่มีขนาดแตกต่างกัน (ด้านหน้า).....	19
รูปที่ 2.10 แสดงตัวอย่างอุปกรณ์ (ด้านหลัง) และตำแหน่งของขาสัญญาณ..... (มีแถวละ 8 ขา แบ่งเป็นแถวบนและแถวล่าง)	19
รูปที่ 2.11 แสดงผังวงจรของ 8x8 LED Matrix..... แบบ Row Anode, Column Cathode	20
รูปที่ 2.12 แสดงผังวงจรของ 8x8 LED Matrix..... แบบ Row Cathode, Column Anode	20
รูปที่ 2.13 แสดงการส่งข้อมูลแบบขนาน.....	22
รูปที่ 2.14 แสดงการรับส่งข้อมูลแบบอนุกรม.....	23
รูปที่ 2.15 การส่งข้อมูลแบบอะซิงโครนัส.....	23
รูปที่ 2.16 รูปแสดงการส่งข้อมูลแบบซิงโครนัส.....	24
รูป 2.17 โครงสร้างและตารางเอาท์พุทของ 7474 ซึ่งเป็น FLIP-FLOP ชนิด D..... $nQ = \bar{Q} = Q \text{ complement}$ , $nQ0 = \bar{Q}0 = Q0 \text{ complement}$	25
รูป 2.18 แสดงการดัดแปลง JK FLIPFLOP ให้มีคุณสมบัติแบบเดียวกับ D FLIPFLOP.....	26
รูป 2.19 ตัวรับข้อมูลแบบ 2 บิต.....	27
รูป 2.20 แสดงความสัมพันธ์ระหว่าง DATA CLOCK และ DATA OUTPUT.....	27
รูป 2.21 ตัวรับข้อมูลแบบ 8 บิต.....	27
รูป 2.22 ตัวเลื่อนข้อมูลชนิดรับส่งข้อมูลเข้าแบบขนาน..... และส่งข้อมูลแบบอนุกรมโดยใช้ฟลิปฟลอป ชนิด J - K	28

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


## สารบัญรูป

เรื่อง	หน้า
รูป 223 ตัวรับข้อมูลแบบขนาน 4 บิต และส่งออกอนุกรม.....	29
รูป 224 แสดงการเลื่อนของข้อมูล 1001 ออกไปทางขวา Q0..... โดยบิตที่มีค่าต่ำสุด (LSB) จะเลื่อนออกไปก่อน	29
รูป 225 โครงสร้างของตัวเลื่อนข้อมูลแบบเข้าอนุกรมและออกแบบขนาน..... ใช้ฟลิปฟล็อปชนิด J-K	30
รูป 226 การนำ D FLIPFLOP มาจัดวงจรเป็นรับเข้าแบบอนุกรม..... และส่งออกแบบขนาน	31
รูป 227 หลังจากมีสัญญาณ CLOCK เข้ามาครบ 4 ลูก..... DATA ที่ส่งเข้ามาในจังหวะที่สัมพันธ์กับสัญญาณ CLOCK จะปรากฏออกมาที่ ขา D3 D2 D1 และ D0	31
รูป 228 แสดงการหน่วงเวลาของข้อมูลโดยการส่งเข้าแบบ..... อนุกรมและส่งออกแบบอนุกรม	32
รูป 229 แสดงการหน่วงเวลาของข้อมูลโดยใช้ D FLIPFLOP..... จำนวน 2 ภาค เพื่อสร้างเป็นตัวรับแบบอนุกรม และส่งออกแบบอนุกรม	32
รูปที่ 230 การส่งข้อมูลแบบซิงโครนัส.....	33
รูปที่ 231 IC 74HC595.....	33
รูปที่ 232 ไอซี MAX7219.....	34
รูปที่ 233 ขาของไอซี MAX7219.....	35
รูปที่ 234 ฟังก์ชันไดอะแกรม(Functional Diagram).....	35
รูปที่ 235 ทามมิงไดอะแกรม(Timing Diagram).....	36
รูปที่ 237 แสดงตำแหน่งจุดเชื่อมต่อสัญญาณของโมดูล MAX7219.....	37
รูปที่ 238 แสดง Clock pulse.....	39
รูปที่ 239 SPI bus: single master and single.....	39
รูปที่ 240 แสดงลักษณะของสัญญาณนาฬิกา.....	41
บทที่ 3 รายละเอียดในการออกแบบ และการสร้าง.....	42
รูปที่ 3.1 บล็อกไดอะแกรมของระบบ.....	42
รูปที่ 3.2 แอพพลิเคชั่นในการป้อนคำสั่ง.....	43
รูปที่ 3.3 การเชื่อมต่อ HC-05 กับ แอนดรอยด์.....	44
รูปที่ 3.4 แสดงขาต่างๆของบอร์ด Arduino Uno R3.....	45
รูปที่ 3.5 ส่วนของภาคแสดงผลแอลอีดีเมทริกซ์(LED MATRIX)..... และวงจรชิพรีจิสเตอร์(SHIFT REGISTER) MAX7219	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

เรื่อง	หน้า
บทที่ 4 การทดลองและผลการทดลอง.....	53
รูปที่ 4.1 ไฟกระพริบที่บลูทูธ.....	53
รูปที่ 4.2 หน้าจอโปรแกรม BTInterface Free Trial.....	53
รูปที่ 4.3 การแสดงผลของ Led Matrix Display.....	54



เรื่อง	หน้า
บทที่ 2 บทสรุปและวิจารณ์.....	2
ตารางที่ 2.1 ตารางเวอร์ชันของแอนดรอยด์.....	3
ตารางที่ 2.2 ข้อมูลโทรศัพท์รุ่นโน้ต 8.....	9
ตารางที่ 2.3 แสดงตำแหน่งของบิตในรีจิสเตอร์สำหรับคำสั่ง..... และข้อมูลขนาด 16 บิต	36
ตารางที่ 2.4 แสดงเลขที่อยู่ของรีจิสเตอร์ภายในของไอซี MAX7219.....	37
ตารางที่ 2.5 สายสัญญาณต่างๆที่ใช้ในการส่งแบบ SPI.....	40

# บทที่ 1

## บทนำ

### 1.1 ที่มาและความสำคัญ

เนื่องจากในปัจจุบันความสะดวกสบายในชีวิตเป็นสิ่งทีคนต้องการ และการสื่อสารในระบบไร้สายมีการใช้งานกันอย่างแพร่หลายทั้งทาง อินเทอร์เน็ต(Internet) อินฟราเรด(Infrared) และ บลูทูธ (บลูทูธ) เป็นต้น และในปัจจุบันมือถือมีการใช้งานอย่างแพร่หลายในหลายระบบและมีฟังก์ชัน (Function)ต่างๆ ที่สามารถทำงานได้อย่างมีประสิทธิภาพและสามารถทำการเชื่อมต่อกับ ไมโครคอนโทรเลอร์(Microcontroller) ซึ่งทำให้เราสามารถที่จะสั่งการไมโครคอนโทรเลอร์ได้ ซึ่งผู้จัดทำเล็งเห็นถึงความสำคัญของเทคโนโลยีในปัจจุบัน ที่จะช่วยให้ชีวิตสะดวกสบายและเป็นการพัฒนาความรู้ทางด้านเทคโนโลยีในระบบใหม่ ซึ่งเป็นการสั่งการโดยระบบโทรศัพท์แบบแอนดรอยด์ (Android) ด้วยการสื่อสารไร้สายแบบบลูทูธ

### 1.2 จุดประสงค์ของงาน

1.2.1 เพื่อพัฒนาระบบที่มีการประยุกต์ใช้เทคโนโลยีบลูทูธและไมโครคอนโทรเลอร์ในชีวิตประจำวัน

1.2.2 เพื่อทำการสร้างแผ่นป้ายแสดงตัวอักษรตามคำสั่งที่ส่งมาจากแอนดรอยด์

### 1.3 ขอบเขตของงานที่ทำ

การทำงานจะเชื่อมต่อการทำงานระหว่างบอร์ดไมโครคอนโทรเลอร์กับวงจรแอลอีดีเมทริกส์ โดยการทำงานจะสั่งการจากมือถือในระบบแอนดรอยด์ผ่านระบบไร้สายบลูทูธ

### 1.4 ประโยชน์และผลลัพธ์ที่คาดว่าจะได้รับ

1.4.1 ความรู้ทางด้านระบบมือถือแอนดรอยด์

1.4.2 การใช้ภาษาจาวา(JAVA) และภาษา C

1.4.3 การใช้บอร์ดอาร์ดูโน ยูโน อาร์ 3 (Arduino Uno R3)

1.4.4 วงจรขับและวงจรแอลอีดีเมทริกส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

# หลักการหรือทฤษฎีที่เกี่ยวข้องกับงานที่ทำ

### 2.1 แอนดรอยด์(Android)

แอนดรอยด์ (Android) เป็นระบบปฏิบัติการสำหรับอุปกรณ์พกพา เช่น โทรศัพท์มือถือ แท็บเล็ตคอมพิวเตอร์ เน็ตบุ๊ก ทำงานบนลินุกซ์ เคอร์เนล เริ่มพัฒนาโดยบริษัทแอนดรอยด์ จากนั้นบริษัทแอนดรอยด์ถูกซื้อโดยกูเกิล และนำแอนดรอยด์ไปพัฒนาต่อ

ภายหลังถูกพัฒนาในนามของ Open Handset Alliance ทางกูเกิลได้เปิดให้นักพัฒนาสามารถแก้ไขโค้ดต่างๆด้วยภาษาจาวา และควบคุมอุปกรณ์ผ่านทางชุด Java libraries ที่กูเกิลพัฒนาขึ้น แอนดรอยด์ได้เป็นที่รู้จักต่อสาธารณชนเมื่อวันที่ 5 พฤศจิกายน พ.ศ. 2550 โดยทางกูเกิลได้ประกาศก่อตั้ง Open Handset Alliance กลุ่มบริษัทฮาร์ดแวร์, ซอฟต์แวร์ ที่ร่วมมือกันเพื่อพัฒนามาตรฐานเปิด สำหรับอุปกรณ์มือถือ ลิขสิทธิ์ของโค้ดแอนดรอยด์นี้จะใช้ในลักษณะของซอฟต์แวร์เสรี โทรศัพท์เครื่องแรกที่สามารถใช้งานระบบปฏิบัติการแอนดรอยด์ได้คือ เอชทีซีดรีม ออกจำหน่ายเมื่อ 22 ตุลาคม 2551

เวอร์ชันล่าสุดของแอนดรอยด์คือ 4.2 (JellyBean) ความสามารถใหม่ของ แอนดรอยด์ 4.2 ที่เพิ่มขึ้นมาคือ Photo Sphere ที่สามารถถ่ายรูปได้ 360 องศา และ Keyboard Gestures ที่สามารถลากนิ้วแทนการสัมผัสตัวอักษรได้

#### 2.1.1 คุณสมบัติและความสามารถของแอนดรอยด์

2.1.1.1 การเชื่อมต่อเทคโนโลยีของแอนดรอยด์นั้นมีความสามารถในการเชื่อมต่อที่ประกอบด้วย GSM/EDGE, CDMA, บลูทูธ, Wi-Fi, NFC และ WiMAX

2.1.1.2 Messaging สนับสนุนการทำงานของ SMS, MMS

2.1.1.3 ระบบฐานข้อมูล แอนดรอยด์นั้นมี SQLite ในการเก็บข้อมูล (ข้อมูล)

2.1.1.4 เว็บเบราว์เซอร์ แอนดรอยด์นั้นจะมี Google Chrome ถูกติดตั้งมาให้พร้อมใช้งานทันที

2.1.1.5 มีเดีย (Media) แอนดรอยด์นั้นสนับสนุนไฟล์วีดีโอ เสียง และรูปภาพในรูปแบบของ MPEG4, H.264, MP3, AAC, JPG และ PNG

2.1.1.6 สตรีมมิง(Streaming) แอนดรอยด์สนับสนุน RTP/RTSP และ HTMLProgressive Down โหลด

2.1.1.7 มัลติทัช (Multi-touch) แอนดรอยด์นั้นรองรับการสั่ง ที่หน้าจอได้มากกว่าการสัมผัส 1 จุด

2.1.1.8 การสนับสนุนฮาร์ดแวร์อื่นๆ ในแอนดรอยด์นั้นสามารถรองรับการทำงานเช่น กล้องถ่ายรูป ,GPS, เซ็นสิท, การวัดอัตราความเร็วและเทอร์โมมิเตอร์ เป็นต้น

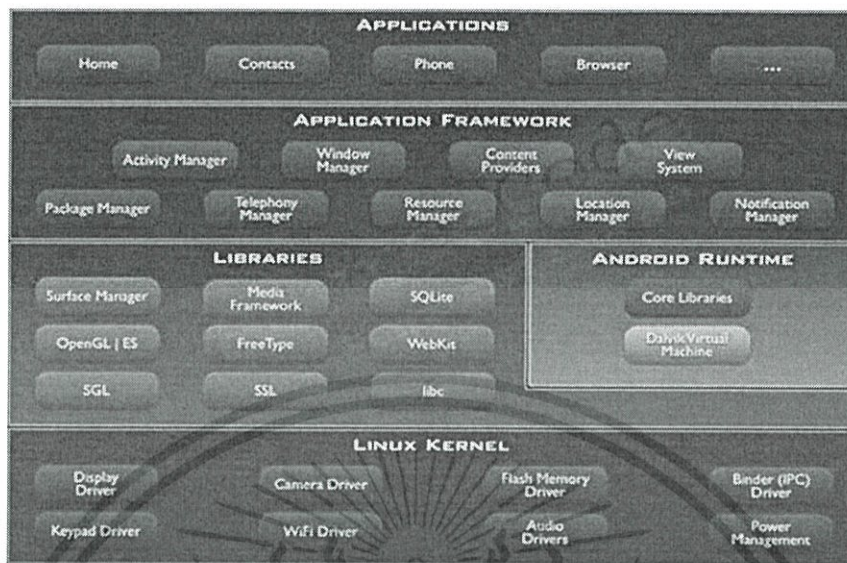
2.1.2 เวอร์ชันของระบบปฏิบัติการแอนดรอยด์ จะใช้รหัสชื่อเวอร์ชันเป็นชื่อขนมหวาน โดยจะเรียงตามลำดับ

เวอร์ชัน	ชื่อเวอร์ชัน	เปิดตัว
1.0	Astro	28 กันยายน 2551
1.1	Bender	9 กุมภาพันธ์ 2552
1.5	Cupcake (คัพเค้ก)	30 เมษายน 2552
1.6	Donut (โดนัท)	15 สิงหาคม 2552
2.0/2.1	Eclair (เอแคลร์)	26 ตุลาคม 2552 (2.0) 12 มกราคม 2553 (2.1)
2.2	Froyo (โฟรเซนโยเกิร์ต)	20 พฤษภาคม 2553
2.3	Gingerbread (ขนมปังขิง)	6 ธันวาคม 2553
3.0/3.1	Honeycomb (รังผึ้ง)	22 กุมภาพันธ์ 2554
4.0	Ice Cream Sandwich (ไอศกรีมแซนด์วิช)	19 ตุลาคม 2554
4.1	Jelly Bean (เจลลี่บีน)	28 มิถุนายน 2555

ตารางที่ 2.1 ตารางเวอร์ชันของแอนดรอยด์

### 2.1.3 โครงสร้างของแอนดรอยด์

การทำความเข้าใจโครงสร้างของระบบปฏิบัติการแอนดรอยด์ ถือว่าเป็นสิ่งสำคัญเพราะถ้า นักพัฒนาโปรแกรม สามารถมองภาพโดยรวมของระบบได้ทั้งหมด จะให้สามารถเข้าใจถึง กระบวนการทำงานได้ดียิ่งขึ้น และสามารถนำไปช่วยในการออกแบบโปรแกรมที่ต้องการพัฒนา เพื่อให้เกิดประสิทธิภาพในการทำงานของแอนดรอยด์



รูปที่ 2.1 สถาปัตยกรรมแอนดรอยด์

จากโครงสร้างของระบบปฏิบัติการแอนดรอยด์ จะสังเกตได้ว่า มีการแบ่งออกมาเป็นส่วนๆ ที่มีความเกี่ยวเนื่องกัน โดยส่วนบนสุดจะเป็นส่วนที่ผู้ใช้งานทำการติดต่อโดยตรงซึ่งก็คือส่วนของ (Applications) จากนั้นก็จะลำดับลงมาเป็นองค์ประกอบอื่นๆตามลำดับ และสุดท้ายจะเป็นส่วนที่ติดต่อกับอุปกรณ์โดยผ่านทางลินุกซ์ เคอร์เนล(Linux Kernel) โครงสร้างของแอนดรอยด์ พอที่จะอธิบายเป็นส่วนๆได้ดังนี้

2.1.3.1 แอปพลิเคชัน(Applications) ส่วนแอปพลิเคชัน หรือส่วนของโปรแกรมที่มีมากับระบบปฏิบัติการหรือเป็นกลุ่มของโปรแกรมที่ผู้ใช้งานได้ทำการติดตั้งไว้ โดยผู้ใช้งานสามารถเรียกใช้โปรแกรมต่างๆได้โดยตรง ซึ่งการทำงานของแต่ละโปรแกรมจะเป็นไปตามที่ผู้พัฒนาโปรแกรมได้ออกแบบและเขียนโค้ดโปรแกรมเอาไว้

2.1.3.2 แอปพลิเคชัน เฟรมเวิร์ค(Application Framework) เป็นส่วนที่มีการพัฒนาขึ้นเพื่อให้ นักพัฒนาสามารถพัฒนาโปรแกรมได้สะดวก และมีประสิทธิภาพมากยิ่งขึ้น โดยนักพัฒนาไม่จำเป็นต้องพัฒนาในส่วนที่มีความยุ่งยากมากๆ เพียงแค่ทำการศึกษาถึงวิธีการเรียกใช้งานแอปพลิเคชัน เฟรมเวิร์คในส่วนที่ต้องการใช้งาน แล้วนำมาใช้งาน ซึ่งมีหลายกลุ่มด้วยกัน ตัวอย่างเช่น

2.1.3.3 แอคทิวิตี้ แมนเนจเจอร์(Activities Manager) เป็นกลุ่มของชุดคำสั่งที่จัดการเกี่ยวกับวงจรการทำงานของหน้าต่างโปรแกรม(แอคทิวิตี้)

2.1.3.4 คอนเทนท์ โพรไวเดอร์(Content Providers) เป็นกลุ่มของชุดคำสั่ง ที่ใช้ในการเข้าถึงข้อมูลของโปรแกรมอื่น และสามารถแบ่งปันข้อมูลให้โปรแกรมอื่นเข้าถึงได้

2.1.3.5 วิวซิสเต็ม(View System) เป็นกลุ่มของชุดคำสั่งที่เกี่ยวกับการจัดการโครงสร้างของหน้าจอที่แสดงผลในส่วนที่ติดต่อกับผู้ใช้งาน (User Interface)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2.1.3.6 เทเลโฟนี แมนเนจเจอร์(Telephony Manager) เป็นกลุ่มของชุดคำสั่งที่ใช้ในการเข้าถึงข้อมูลด้านโทรศัพท์ เช่นหมายเลขโทรศัพท์ เป็นต้น
- 2.1.3.7 รีซอส แมนเนจเจอร์(Resource Manager) เป็นกลุ่มของชุดคำสั่งในการเข้าถึงข้อมูลที่เป็นข้อความ, รูปภาพ
- 2.1.3.8 โลเคชัน แมนเนจเจอร์(Location Manager) เป็นกลุ่มของชุดคำสั่งที่เกี่ยวข้องกับตำแหน่งทางภูมิศาสตร์ ที่ระบบปฏิบัติการได้รับค่าจากอุปกรณ์
- 2.1.3.9 โนติฟิเคชัน แมนเนจเจอร์(Notification Manager) เป็นกลุ่มของชุดคำสั่งที่จะถูกเรียกใช้เมื่อโปรแกรม ต้องการแสดงผลให้กับผู้ใช้งาน ผ่านทางแถบสถานะ(Status Bar) ของหน้าจอ
- 2.1.3.10 ไลบรารี(Libraries) เป็นส่วนของชุดคำสั่งที่พัฒนาด้วย C/C++ โดยแบ่งชุดคำสั่งออกเป็นกลุ่มตามวัตถุประสงค์ของการทำงาน เช่น เซอร์เฟซ แมนเนจเจอร์(Surface Manage) จัดการเกี่ยวกับการแสดงผล, มีเดีย เฟรมเวิร์ค(Media Framework) จัดการเกี่ยวกับการการแสดงผลและเสียง, Open GL | ES และ SGL จัดการเกี่ยวกับภาพ 3มิติ และ 2มิติ, SQLite จัดการเกี่ยวกับระบบฐานข้อมูล เป็นต้น
- 2.1.3.11 แอนดรอยด์ รันไทม์(Android Runtime) จะมี ดาร์วิค เวอร์ทัว แมชชีน(Darvik Virtual Machine) ที่ถูกออกแบบมา เพื่อให้ทำงานบนอุปกรณ์ที่มี หน่วยความจำ(Memory), หน่วยประมวลผลกลาง(CPU) และพลังงาน(Battery)ที่จำกัด ซึ่งการทำงานของดาร์วิค เวอร์ทัว แมชชีน จะทำการแปลงไฟล์ที่ต้องการทำงาน ไปเป็นไฟล์ .DEX ก่อนการทำงาน เหตุผลก็เพื่อให้มีประสิทธิภาพเพิ่มขึ้นเมื่อใช้งานกับ หน่วยประมวลผลกลางที่มีความเร็วไม่มาก ส่วนต่อมาก็คือคอร์ ลิบรารี(Core Libraries) ที่เป็นส่วนรวบรวมคำสั่งและชุดคำสั่งสำคัญ โดยถูกเขียนด้วยภาษาจาวา (Java Language)
- 2.1.3.12 ลินุกซ์ เคอร์เนล(Linux Kernel) เป็นส่วนที่ทำหน้าที่หัวใจสำคัญ ในจัดการกับบริการหลักของระบบปฏิบัติการ เช่น เรื่องหน่วยความจำ พลังงาน ติดต่อกับอุปกรณ์ต่างๆ ความปลอดภัย เครือข่าย โดยแอนดรอยด์ได้นำเอาส่วนนี้มาจากระบบปฏิบัติการลินุกซ์ รุ่น 2.6 (Linux 2.6. Kernel) ซึ่งได้มีการออกแบบมาเป็นอย่างดี

#### 2.1.4 ข้อเด่นของแอนดรอยด์

เนื่องจากระบบปฏิบัติการแอนดรอยด์มีการเจริญเติบโตอย่างรวดเร็ว และมีส่วนแบ่งตลาดของอุปกรณ์ด้านนี้ ขึ้นทุกขณะ ทำให้กลุ่มผู้ใช้งาน และกลุ่มนักพัฒนาโปรแกรม ให้ความสำคัญกับระบบปฏิบัติการแอนดรอยด์เพิ่มมากขึ้น

เมื่อมองในด้านของกลุ่มผลิตภัณฑ์ บริษัทที่มีการพัฒนาผลิตภัณฑ์รุ่นใหม่ ได้มีการนำเอา ระบบปฏิบัติการแอนดรอยด์ไปใช้ในสินค้าของตนเอง พร้อมทั้งยังมีการปรับแต่งให้ระบบปฏิบัติการมีความสามารถ การจัดวาง โปรแกรม และลูกเล่นใหม่ๆ ที่แตกต่างจากคู่แข่งในท้องตลาด โดยเฉพาะอย่างยิ่ง กลุ่มสินค้าที่เป็น มือถือรุ่นใหม่ (SmartPhone) และอุปกรณ์จอสัมผัส (Touch Screen) โดย

มีคุณลักษณะแตกต่างกันไป เช่นขนาดหน้าจอ ระบบโทรศัพท์ ความเร็วของหน่วยประมวลผล ปริมาณหน่วยความจำ แม้กระทั่งอุปกรณ์ตรวจจับต่างๆ(Sensor)

หากมองในด้านของการพัฒนาโปรแกรม ทางบริษัท กูเกิ้ล ได้มีการพัฒนาแอปพลิเคชันเฟรมเวิร์ค(Application Framework) ไว้สำหรับนักพัฒนาใช้งาน ได้อย่างสะดวก และไม่เกิดปัญหาเมื่อนำชุดโปรแกรมที่พัฒนาขึ้นมา ไปใช้กับอุปกรณ์ที่มีคุณลักษณะต่างกัน เช่นขนาดจออุปกรณ์ ไม่เท่ากัน ก็ยังสามารถใช้งานโปรแกรมได้เหมือนกัน เป็นต้น

ส่วนประกอบของแอปพลิเคชัน คือ หลักที่ใช้ในการสร้างแอนดรอยด์แอปพลิเคชัน(Android Application) โดยส่วนประกอบของแอปพลิเคชัน แบ่งออกเป็น 4 ประเภท ได้แก่ แอคทิวิตี้ (Activity),เซอร์วิส( Service),คอนเทนต์ โพรไวเดอร์(Content Provider), และบรอดแคส รีซีฟเวอร์ (Broadcast Receiver) ซึ่งแต่ละประเภทของส่วนประกอบของแอปพลิเคชัน นี้มีเป้าหมายในการใช้งานที่แตกต่างกัน มีรูปแบบการกระตุ้นให้เกิดการทำงานที่แตกต่างกัน (กล่าวคือบางส่วนประกอบของแอปพลิเคชัน ถูกกระตุ้นให้เกิดการทำงานโดยซิสเต็ม และบางส่วนประกอบของแอปพลิเคชัน ถูกกระตุ้นให้เกิดการทำงานโดยส่วนประกอบของแอปพลิเคชัน อื่น ๆ ) รวมถึงมีวงจรชีวิตที่แตกต่างกันด้วย (กล่าวคือแต่ละส่วนประกอบของแอปพลิเคชัน จะมีรูปแบบที่แตกต่างกันว่าส่วนประกอบของแอปพลิเคชัน นั้น ๆ จะถูกสร้าง (Create) เมื่อใด หรือถูกทำลาย (Destroy) เมื่อใด )

ส่วนประกอบของ แอปพลิเคชัน (Application Component) สามารถแบ่งออกได้เป็น 4 ประเภทดังนี้

#### 2.1.4.1 แอคทิวิตี้(Activity)

คือ ส่วนประกอบของแอปพลิเคชันที่ใช้ในการควบคุมการสร้างยูสเซอร์ อินเตอร์เฟซ(User Interface) เช่น การแสดงผลหน้าจอรายการอีเมล, การแสดงผลหน้าจอแบบฟอร์มการส่งอีเมล เป็นต้น รวมถึงควบคุมการมีปฏิสัมพันธ์ระหว่างผู้ใช้กับยูสเซอร์ อินเตอร์เฟซ ด้วย เช่น เมื่อผู้ใช้เลือกรายการอีเมลก็จะทำการตอบสนองผู้ใช้โดยการแสดงข้อมูลรายการอีเมลที่เลือก เป็นต้น สำหรับการสร้างแอคทิวิตี้ นั้น ทำได้โดยการสร้างระดับ และให้สืบทอดจากระดับ แอคทิวิตี้ หรือสืบทอดจากระดับใด ๆ ก็ตามที่ได้รับการสืบทอดมาจากระดับแอคทิวิตี้ โดยแอคทิวิตี้หนึ่งๆจะควบคุมการแสดงผลยูสเซอร์ อินเตอร์เฟซ หนึ่ง ๆ เท่านั้น และนั่นแสดงให้เห็นว่าแอปพลิเคชันหนึ่ง ๆ จะประกอบด้วยแอคทิวิตี้ จำนวนมากที่ทำงานร่วมกันอยู่ อย่างไรก็ตามถึงแม้ว่าแอคทิวิตี้ จะทำงานร่วมกัน แต่แอคทิวิตี้ เหล่านี้ยังคงเป็นอิสระจากกัน

#### 2.1.4.2 เซอร์วิส(Service)

คือ ส่วนประกอบของแอปพลิเคชัน ที่ไม่มียูสเซอร์ อินเตอร์เฟซ และจะทำการประมวลผลในพื้นหลัง กล่าวคือเป็นการประมวลผลที่ดำเนินไปพร้อมกับที่ผู้ใช้สามารถไปใช้งานแอปพลิเคชัน อื่น ๆ ได้ หรือกล่าวอีกมุมหนึ่ง การประมวลผลในพื้นหลัง คือการประมวลผลที่สามารถทำงานขนานกันกับการทำงานอื่น ๆ ของผู้ใช้ ทั้งนี้ก็เพื่อทำให้เกิดการทำงานใด ๆ โดยที่ผู้ใช้ไม่จำเป็นต้องอยู่ในหน้าจอ นั้น ๆ ได้ ซึ่งอาจเป็นเพราะการทำงานนั้นต้องใช้ระยะเวลานาน เช่น การใช้เซอร์วิส เปิดเพลง เพื่อให้ผู้ใช้สามารถไปใช้แอปพลิเคชันอื่น ๆ ได้ แต่เพลงยังคงเล่นอยู่ หรือ การใช้เซอร์วิส ดาวน์โหลดข้อมูล

ใด ๆ ที่มีขนาดใหญ่ เพื่อให้ผู้ใช้สามารถไปใช้แอปพลิเคชันอื่น ๆ ได้ แต่การดาวน์โหลดยังคงดำเนินอยู่เป็นต้น สำหรับการสร้างเซอร์วิส นั้นทำได้โดยการสร้าง ระดับ และให้สืบทอดจากระดับเซอร์วิส หรือสืบทอดจากระดับใด ๆ ก็ตามที่ได้รับสืบทอดมาจากระดับเซอร์วิส

#### 2.1.4.3 คอนเทนท์ โพรไวเดอร์(Content Provider)

คือ ส่วนประกอบของแอปพลิเคชันที่ทำหน้าที่ในการควบคุมข้อมูลใด ๆ ของแอปพลิเคชัน ที่ต้องการแชร์ ให้แอปพลิเคชัน อื่น ๆ สามารถนำข้อมูลนั้น ๆ ไปใช้งานได้ หรือกล่าวในทางกลับกันก็คือแอปพลิเคชัน ใด ๆ สามารถนำข้อมูล (รวมถึงแก้ไขข้อมูลได้ ถ้าคอนเทนท์ โพรไวเดอร์ อนุญาต) ของแอปพลิเคชัน อื่น ๆ มาใช้งานได้ โดยกระทำผ่านคอนเทนท์ โพรไวเดอร์ เช่น ซิสเทม ได้จัดเตรียมคอนเทนท์ โพรไวเดอร์ ที่เป็นข้อมูลรายชื่อผู้ติดต่อ (Contact) ไว้ เพื่อให้แอปพลิเคชัน ที่ต้องการใช้ข้อมูลรายชื่อผู้ติดต่อนี้ สามารถนำข้อมูลไปใช้หรือแก้ไขข้อมูลได้ เป็นต้น สำหรับการสร้างคอนเทนท์ โพรไวเดอร์ นั้น ทำได้โดยการสร้าง ระดับ และให้สืบทอดจากระดับคอนเทนท์ โพรไวเดอร์ หรือสืบทอดจาก ระดับ ใด ๆ ก็ตามที่ได้รับสืบทอดมาจาก ระดับคอนเทนท์ โพรไวเดอร์

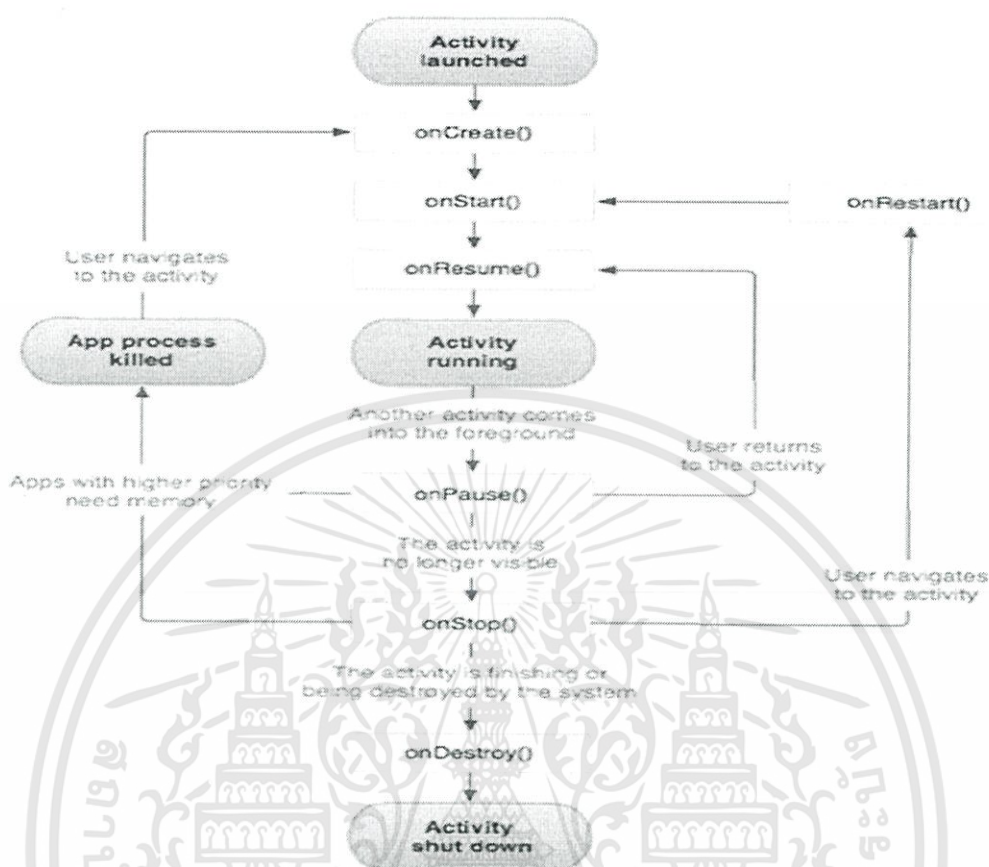
#### 2.1.4.4 บรอดแคส รีซีฟเวอร์(Broadcast Receiver)

คือ ส่วนประกอบของแอปพลิเคชันที่ไม่มียูสเซอร์อินเตอร์เฟซ โดยจะทำหน้าที่รับรู้สิ่งที่เกิดขึ้นของซิสเทม และนำมาบอกให้ผู้ใช้ได้รับรู้ เช่น เมื่อแบตเตอรี่ต่ำ, เมื่อหน้าจอถูกเก็บภาพ, เมื่อมีการพิกหน้าจอ เป็นต้น ทั้งนี้แอปพลิเคชัน ใด ๆ สามารถนำบรอดแคส รีซีฟเวอร์ มาใช้ประโยชน์ได้ เช่น เมื่อแอปพลิเคชัน ได้ ดาวน์โหลด ข้อมูลเสร็จเรียบร้อยแล้ว เป็นต้น ซึ่งโดยส่วนมากแล้วการตอบสนองของบรอดแคส รีซีฟเวอร์ จะกระทำผ่านโนติฟิเคชัน เพื่อแจ้งสิ่งที่เกิดขึ้นให้ผู้ใช้ได้รับรู้ สำหรับการสร้างบรอดแคส รีซีฟเวอร์ นั้น ทำได้โดยการสร้าง ระดับ และให้สืบทอดจาก ระดับ บรอดแคส รีซีฟเวอร์ หรือสืบทอดจาก ระดับ ใด ๆ ก็ตามที่ได้รับสืบทอดมาจาก ระดับบรอดแคส รีซีฟเวอร์

#### แอกทิวิตี ไลฟ์ ไซเคิล(Activity Life Cycle)

แอกทิวิตี มีหน้าที่ในการสร้าง และควบคุมการทำงานของหน้าจอ เพื่อโต้ตอบกับผู้ใช้งาน ซึ่งจะมีได้แค่แอกทิวิตี เดียวเท่านั้นที่โต้ตอบกับผู้ใช้งานในขณะใดขณะหนึ่ง แอกทิวิตี แต่ละตัวจะมีวงจรชีวิตเป็นของตนเอง โดยแบ่งเป็นสถานะดังนี้

- Running เป็นสถานะที่แอกทิวิตี กำลังได้ทำงานในขณะนั้น
- Pause เป็นสถานะที่แอกทิวิตี ปรากฏอยู่บนหน้าจอ แต่ไม่ได้ทำงานในขณะนั้น
- Stop เป็นสถานะที่แอกทิวิตี ไม่ปรากฏบนหน้าจอ
- Kill เป็นสถานะที่แอกทิวิตี ถูกทำลายไปแล้วเมื่อแอกทิวิตีอยู่ในสถานะ พัก หรือหยุด แอกทิวิตีสามารถถูกทำลายได้โดยระบบ



รูปที่ 2.2 องค์ประกอบการทำงานของแอปพลิเคชัน

onCreate(Bundle) – จะถูกเรียกเมื่อแอกทिवิตี มีการสร้างตัวเองครั้งแรก เป็นวิธีสำคัญในสร้าง หน้าจอการทำงานต่างๆ หรือกำหนดสถานะต่างๆ ของตัวควบคุมที่จะใช้งาน

onRestart() – จะถูกเรียกหลังจากที่แอกทिवิตี กลับมาจากสถานะหยุด เพื่อเริ่มทำงานใหม่อีกครั้ง

onStart() – จะถูกเรียกเมื่อแอกทिवิตี เริ่มแสดงให้ผู้ใช้งานเห็น

onResume() – จะถูกเรียกเมื่อแอกทिवิตี สามารถที่จะโต้ตอบกับผู้ใช้งานได้

onPause() – จะถูกเรียกเมื่อมีแอกทिवิตี อื่นกำลังทำงาน เป็นวิธีสำคัญในการเก็บข้อมูลที่สำคัญ หรือ ทำการปล่อยหน่วยความจำเพื่อให้แอกทिवิตี อื่นได้ใช้งาน

onStop() – จะถูกเรียกเมื่อแอกทिवิตี ไม่ปรากฏบนหน้าจอ แล้วจะเข้าสู่สถานะ Stop

onDestroy() – จะถูกเรียกเมื่อแอกทिवิตี กำลังจะถูกทำลาย

และยังมีวิธีที่สำคัญอีก 2 ตัว คือ

onSaveInstanceState(Bundle) – จะถูกเรียกก่อน onPause() เพื่อเก็บสถานะต่างๆ ของแอกทिवิตี ไว้ใช้เมื่อแอกทिवิตี นี้ได้กลับมาทำงานอีกครั้ง

onRestoreInstanceState(Bundle) – จะถูกเรียกก่อน onResume() เพื่อนำสถานะต่างๆ ของแอกทिवิตี ที่เก็บไว้มาใช้งาน

NOTE 8 SPEC		
ทั่วไป	2G	GSM 850/900/1800/1900
	3G	HSDPA 850/900/1900/2100
	4G	LTE
	SIM	Micro-SIM
ความจำ	Card slot	MicroSD, up to 64 GB
	Internal	16 GB storage, 2 GB RAM
ข้อมูล	GPRS	
	EDGE	ระดับ 12
	Speed	HSDPA, 21 Mbps; HSUPA, 5.76 Mbps
	WLAN	Wi-Fi 802.11 a/b/g/n, dual-band, Wi-Fi Direct, Wi-Fi hotspot, DLNA
	Bluetooth	Yes, v4.0 with A2DP
	Infrared port	Yes
	USB	Yes, MicroUSB v2.0, USB Host
FEATURES	OS	Android OS, v4.1.2 (Jelly Bean)
	Chipset	Exynos 4412
	CPU	Quad-core 1.6 GHz Cortex-A9
	GPU	Mali-400MP
	Sensors	Accelerometer, gyro, proximity, compass
	Messaging	SMS(threaded view), MMS, Email, Push Email, IM
	Browser	HTML5
	Radio	No
	GPS	Yes, with A-GPS support and GLONASS
	JAVA	Yes, via Java MIDP emulator
BATTERY	Non-removable Li-Ion 4600 mAh battery	
	Stand-by	
	Talk time	Up to 8 h (multimedia)
	Music play	Up to 120 h
TESTS	Loudspeaker	<u>Voice 66dB / Noise 64dB / Ring 74dB</u>
	Audio quality	<u>Noise -85.2dB / Crosstalk -83.3dB</u>

ตารางที่ 2.2 ข้อมูลโทรศัพท์รุ่นโน้ต 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2 บลูทูธ(Bluetooth)

จะใช้สัญญาณวิทยุความถี่สูง 2.4 กิกะเฮิรตซ์ (GHz.) แต่จะแยกย่อยออกไป ตามแต่ละประเทศ อย่างในแถบยุโรปและอเมริกา จะใช้ช่วง 2.400 ถึง 2.4835 กิกะเฮิรตซ์. แบ่งออกเป็น 79 ช่องสัญญาณ และจะใช้ช่องสัญญาณที่แบ่งนี้ เพื่อส่งข้อมูลสลับช่องไปมา 1,600 ครั้งต่อ 1 วินาที ส่วนที่ญี่ปุ่นจะใช้ความถี่ 2.402 ถึง 2.480 กิกะเฮิรตซ์ แบ่งออกเป็น 23 ช่อง ระยะทำการของบลูทูธ จะอยู่ที่ 5-10 เมตร โดยมีระบบป้องกันโดยใช้การบ่อนรหัสก่อนการเชื่อมต่อ และ ป้องกันการดักสัญญาณระหว่างสื่อสาร โดยระบบจะสลับช่องสัญญาณไปมา จะมีความสามารถในการเลือกเปลี่ยนความถี่ที่ใช้ในการติดต่อเองอัตโนมัติ โดยที่ไม่จำเป็นต้องเรียงตามหมายเลขช่อง ทำให้การดักฟังหรือลักลอบขโมยข้อมูลทำได้ยากขึ้น

โดยหลักของบลูทูธจะถูกออกแบบมาเพื่อใช้กับอุปกรณ์ที่มีขนาดเล็ก เนื่องจากใช้การขนส่งข้อมูลในจำนวนที่ไม่มาก อย่างเช่น ไฟล์ภาพ, เสียง, แอปพลิเคชันต่างๆ และสามารถเคลื่อนย้ายได้ง่าย ขอให้อยู่ในระยะที่กำหนดไว้เท่านั้น (ประมาณ 5-10 เมตร) นอกจากนี้ยังใช้พลังงานต่ำ กินไฟน้อย และสามารถใช้งานได้นาน โดยไม่ต้องนำไปชาร์จไฟบ่อยๆ ด้วย

ส่วนความสามารถการส่งถ่ายข้อมูลของบลูทูธ จะอยู่ที่ 1 เมกะบิตต่อวินาที (1 Mbps) และคงจะไม่มีปัญหาอะไรมากกับขนาดของไฟล์ที่ใช้กันบนโทรศัพท์มือถือ หรือ การใช้งานแบบทั่วไป ซึ่งถือว่าเหลือเฟือมาก แต่ถ้าเป็นข้อมูลที่มีขนาดใหญ่ล่ะก็ คงจะช้าเกินไป และถ้าถูกนำไปเปรียบกับ ไวร์เลส แลน(Wireless LAN)แล้ว ความสามารถของบลูทูธ คงจะห่างชั้นกันเยอะ ซึ่งในส่วนของไวร์เลส แลน ก็ยังมีระยะการรับ-ส่งที่ไกลกว่า แต่ขอได้เปรียบของบลูทูธ จะอยู่ที่ขนาดเล็กกว่า การติดตั้งทำได้ง่ายกว่า และที่สำคัญ การใช้พลังงานก็น้อยกว่ามาก อยู่ที่ 0.1 วัตต์ หากเทียบกับคลื่นมือถือแล้ว ยังห่างกันอยู่หลายเท่าเหมือนกัน

### 2.2.1 ระยะทำการ

ความสามารถในการส่งข้อมูลของบลูทูธนั้นขึ้นกับแต่ละระดับ ที่ใช้ ซึ่งมี 3 ระดับ ดังนี้

ระดับที่ 1 กำลังส่ง 100 มิลลิวัตต์ ระยะประมาณ 100 เมตร

ระดับที่ 2 กำลังส่ง 2.5 มิลลิวัตต์ ระยะประมาณ 10 เมตร

ระดับที่ 3 กำลังส่ง 1 มิลลิวัตต์ ระยะประมาณ 1 เมตร

### 2.2.2 รุ่น

ปัจจุบันข้อกำหนดบลูทูธ ออกมาแล้ว 5 รุ่น

บลูทูธ 1.0, บลูทูธ 1.1,บลูทูธ 1.2 z, บลูทูธ 2.0,บลูทูธ 2.0 อีดีอาร์

## บลูทูธ 2.1 อีดีอาร์, บลูทูธ 3.0

### เป้าหมายของ บลูทูธ

เทคโนโลยีบลูทูธพัฒนาขึ้นมาโดยมีเป้าหมายคือ

2.2.2.1 พัฒนาให้มีราคาต่ำที่สามารถให้คนทั่วไปใช้ได้

2.2.2.2 ทำให้บลูทูธมีขนาดเล็กที่สุดเพื่อให้ใช้งานได้ง่าย

2.2.2.3 ให้บลูทูธใช้พลังงานในการทำงานน้อยเพื่อให้สามารถติดต่อกันได้โดยไร้ข้อจำกัด

2.2.2.4 พัฒนาให้บลูทูธมีความทนทานในการใช้งานและสามารถส่งทั้งข้อมูลและเสียงได้อย่างมีประสิทธิภาพ

2.2.2.5 เป็นมาตรฐานเปิดคือให้ผู้ที่สนใจสามารถนำไปพัฒนาต่อได้ทำให้เทคโนโลยีพัฒนาได้อย่างรวดเร็ว (บลูทูธ 2001)

### 2.2.3 ข้อดี (Strength)

2.2.3.1 เพิ่มความสะดวกสบายในการใช้อุปกรณ์อิเล็กทรอนิกส์ต่างๆ

2.2.3.2 สามารถโอนถ่ายข้อมูลต่างๆได้ง่ายและรวดเร็วขึ้น

2.2.3.3 ประหยัดค่าใช้จ่ายในการส่งข้อมูลหรือรูปภาพต่างๆ

2.2.3.4 การใช้บลูทูธจะช่วยประหยัดเวลาในการตั้งอุปกรณ์ต่างๆ และสามารถเคลื่อนย้ายอุปกรณ์ได้ง่าย

2.2.3.5 ลดความกังวลในการใช้โทรศัพท์ เนื่องจากถ้าผู้ใช้เลือกใช้ ชุดหูฟัง แบบมีสายต่อก็คงต้องคอยกังวลว่าสายจะไปเกี่ยวอะไรหรือไม่ แต่ถ้าเลือกใช้ชุด ชุดหูฟังแบบบลูทูธผู้ใช้ก็หมดความกังวลได้เลยว่าสายชุดหูฟัง จะไปเกี่ยวอะไรหรือไม่เวลาใช้โทรศัพท์

2.2.3.6 เมื่อเปรียบเทียบกับการใช้อินฟราเรดแล้ว การใช้ บลูทูธ มีข้อดีกว่า เนื่องจากการรับส่งข้อมูลแบบอินฟราเรดต้องใช้แสงเป็นสื่อในการติดต่อและผู้ส่งกับผู้รับจะต้องอยู่ในตำแหน่งที่ตรงกันห้ามมีสิ่งกีดขวางระหว่างผู้ส่งกับผู้รับ บลูทูธ ใช้สัญญาณวิทยุในการติดต่อสื่อสารผู้รับและผู้ส่งสามารถอยู่จุดไหนก็ได้ภายใต้รัศมีไม่เกิน 10 เมตร

### 2.2.4 ข้อเสีย (weakness)

2.2.4.1 ความง่ายตายในการโอนถ่ายข้อมูลอาจทำให้เกิดอาชญากรรมเพิ่มมากขึ้นได้ถ้าบุคคลเหล่านั้นนำบลูทูธไปใช้งานในแบบที่ไม่เหมาะสม

2.2.4.2 ถ้ามีการเปิดบลูทูธทิ้งไว้นานอาจมีกลุ่มบุคคลที่ไม่ประสงค์ดีปล่อยตัวไวรัสมาที่อุปกรณ์อิเล็กทรอนิกส์ของเราได้ ซึ่งก่อให้เกิดความเสียหายต่อทรัพย์สินและข้อมูลต่างๆได้

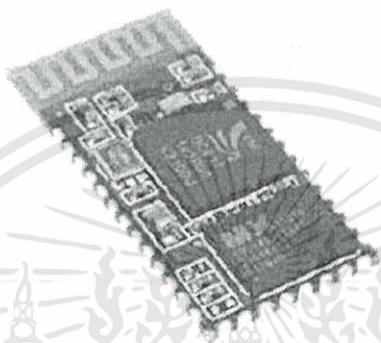
2.2.4.3 การใช้ชุดหูฟังแบบบลูทูธและโทรศัพท์มือถืออย่างเพละติเพลินและความสะดวกสบาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อาจทำให้ผู้ใช้ขาดความระมัดระวังได้

2.2.4.4 การส่งข้อมูลทาง บลูทูธ อาจทำให้เกิดการดักฟังหรือการลักลอบขโมยข้อมูลต่างๆได้ ถึงแม้ว่าจะทำได้ยากก็ตาม

## 2.2.5 บอร์ดเอชซี 05 (HC-05)



รูปที่ 2.3 โมดูลบลูทูธ HC-05

บลูทูธโมดูลเอชซี 05 เป็นโมดูลไร้สายที่ใช้สื่อสารกันด้วยบลูทูธ SPP (Serial Port Protocol) โดยพอร์ทอนุกรม(Serial port) เป็น บลูทูธ V2.0+อีดีอาร์ 3 เมกะบิตต่อวินาที โมดูลเลชัน (Modulation)พร้อมกับความถี่ใช้งาน 2.4 กิกะเฮิรซ์ สามารถเชื่อมต่อกับอุปกรณ์บลูทูธอื่นๆได้ รวมทั้งเชื่อมต่อด้วย Serial Interface ระหว่างคอมพิวเตอร์ด้วยวงจรรีเสต232(RS232) เพื่อเข้าสู่คำสั่ง(AT Command) ในการปรับแต่งค่าต่างๆของโมดูล

โมดูลรุ่นนี้สามารถตั้งค่าให้การทำงานได้ทั้ง มาสเตอร์ และ สเลฟ ซึ่งรายละเอียดจะอยู่ใน ส่วนของคำสั่ง AT Command

### 2.2.5.1 คุณสมบัติต่างๆ

2.2.5.1.1 ความตบสนองสัญญาณ -80 เดซิเบลเมตร

2.2.5.1.2 กำลังส่งคลื่นวิทยุเพิ่มได้ถึง +4 เดซิเบลเมตร

2.2.5.1.3 แรงดัน 1.8-3.6 โวลต์

2.2.5.1.4 การควบคุม PIO

2.2.5.1.5 พร้อมกับสามารถปรับแต่งความเร็วบิต(baud rate) ได้

2.2.5.1.6 เสออากาศในตัว

### 2.2.5.2 รายละเอียดการทำงานต่างๆ

2.2.5.2.1 ความเร็วบิตปกติ 38400, บิตข้อมูล 8, บิตหยุด 1

2.2.5.2.2 รองรับความเร็วบิต 9600,19200,38400,57600,115200,230400,460800

## 2.3 บอร์ดอาร์ดูโน(Arduino)

บอร์ดอาร์ดูโนเป็นรูปแบบของอินพุตและเอาต์พุต บอร์ดอย่างง่าย ๆ ที่มี อินพุตและเอาต์พุต ขึ้นพื้นฐานที่พอเพียงกับการใช้งาน และ การเรียนรู้ โดยตัวบอร์ดจะมาพร้อมกับชุดคำสั่งที่ใช้ควบคุม พอร์ต(port) อินพุตและเอาต์พุต ไม่ว่าจะเป็น พอร์ตดิจิทัล(port digital), พอร์ตอนาล็อก(port analog), PWM และพอร์ตอนุกรม ซึ่งอาร์ดูโนนั้นเป็นเครื่องมือที่จะทำให้คอมพิวเตอร์สามารถรับ สัญญาณจากภายนอกและส่งสัญญาณไป ควบคุมอุปกรณ์ภายนอกได้อย่างมีประสิทธิภาพมากกว่าใช้ เครื่องพีซีตั้งโต๊ะ ตัวบอร์ดออกแบบจากไมโครคอมพิวเตอร์ชิพเดียว, และมีโปรแกรมพัฒนาสำหรับ เขียนโปรแกรมให้บอร์ดทำงานอาร์ดูโน สามารถประยุกต์ทำเครื่องใช้อัจฉริยะ รับสัญญาณจากสวิทช์ หรือ เซนเซอร์, และควบคุม หลอดไฟ, มอเตอร์, หรืออุปกรณ์อื่นๆ โปรแกรม อาร์ดูโน เป็นได้ทั้งแบบ ทำงานอิสระ หรือทำงานติดต่อกับโปรแกรมที่ทำงานบนเครื่องพีซี ตัวบอร์ดสามารถประกอบขึ้นใช้เอง หรือจะซื้อสำเร็จที่มีขาย ส่วนโปรแกรมพัฒนาอาร์ดูโน สามารถดาวน์โหลดได้ฟรี

อาร์ดูโน เป็นภาษาอิตาลี ซึ่งใช้เป็นชื่อของโครงการพัฒนาไมโครคอนโทรลเลอร์ตระกูล เอวี อาร์(AVR) แบบเปิดเผยที่ได้รับการปรับปรุงมาจากโครงการพัฒนาของ เอวีอาร์ อีกโครงการหนึ่งที่ชื่อว่า “Wiring” แต่เนื่องจากโครงการของ “Wiring” เลือกใช้เอวีอาร์เบอร์ ATmega128 ซึ่งเป็น ไมโครคอนโทรลเลอร์ที่มีจำนวนของหน่วยความจำ และอินพุตและเอาต์พุต ค่อนข้างมากและที่สำคัญ ATmega128 เป็นชิพที่มีตัวถังแบบเอสเอ็มดี(SMD) จึงทำให้เป็นอุปสรรคสำหรับผู้เริ่มต้นในการสร้าง บอร์ดและต่อวงจรขึ้นมาใช้งานกันเอง และบอร์ดจะมีขนาดค่อนข้างใหญ่ ซึ่งอาจดูว่าเกินความจำเป็น สำหรับผู้เริ่มต้น จึงไม่ค่อยได้รับความนิยมเท่าที่ควร แต่หลังจากที่ทีมงานอาร์ดูโนนำโค้ดที่มีของ “Wiring” มาพัฒนาปรับปรุงใหม่โดยให้สามารถใช้งานกับไมโครคอนโทรลเลอร์ขนาดเล็ก เช่น Mega8, Mega168, Mega328 ได้ จึงทำให้ระบบวงจรของบอร์ดมีขนาดเล็กลงกว่า “Wiring” มาก และยังใช้อุปกรณ์น้อยชิ้น ทำให้ง่ายต่อการต่อวงจรใช้งานกันเอง และยังประหยัดต้นทุนในการสร้าง บอร์ดไปได้มาก ด้วยเหตุนี้เองที่ทำให้ “อาร์ดูโน” ได้รับความนิยมจากผู้ใช้งานทั่วโลกเป็นอย่างมากใน ระยะเวลาอันรวดเร็ว

อาร์ดูโน มีจุดเด่นในเรื่องของความง่ายต่อการเรียนรู้และใช้งาน เนื่องจากมีการออกแบบคำสั่งต่างๆขึ้นมาสนับสนุนการใช้งาน ด้วยรูปแบบที่ง่ายไม่ซับซ้อน ในตลาดไมโครคอนโทรลเลอร์มี ตัวเลือกมากมาย เช่น Parallax Basic Stamp, Netmedia's BX-24, Pidgets, MIT's Handyboard, และอีกหลายเจ้าที่มีคุณสมบัติใกล้เคียงกัน คือทำโปรแกรมให้ใช้งานง่าย และเน้นการ โปรแกรมไมโครคอนโทรลเลอร์เป็นหลัก อาร์ดูโน ก็เช่นเดียวกันแต่มีข้อแตกต่างที่เห็นได้ชัดคือ

- มีราคาไม่แพง เนื่องจากมีโค้ดที่เปิดเผย และวงจรแจกฟรี สามารถต่อวงจรขึ้นมาใช้งานได้เอง
- ทำงานได้หลายแพลตฟอร์ม
- โปรแกรมพัฒนาอาร์ดูโน ทำงานได้ทั้งบนวินโดวส์ Macintosh OSX และ บนลินุกซ์ ในขณะที่บอร์ด อื่นทำงานได้เฉพาะบนวินโดวส์
- ใช้งานง่าย มีโปรแกรมพัฒนาที่ไม่ซับซ้อน มีโปรแกรมพัฒนาอาร์ดูโน ใช้งานง่ายสำหรับมือใหม่ และ มีความสามารถครบความต้องการของนักพัฒนามืออาชีพ
- เปิดเผยแพร่โค้ด และ นำไปพัฒนาต่อยอดได้ โปรแกรมอาร์ดูโนตีพิมพ์แบบเปิดเผยซอร์สโค้ด และสามารถเพิ่มเติมความสามารถผ่าน C++ library, ถ้าต้องการศึกษาให้ลึกซึ่งสามารถเข้าไปเล่น เอวี อาร์ C ซึ่งเป็นต้นแบบของอาร์ดูโน, และสามารถเพิ่มเติม เอวีอาร์ - C โค้ดได้โดยตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เปิดเผยวงจรและนำไปพัฒนาขยาย ฮาร์ดแวร์(Hardware) ได้ อาร์ดูโนใช้ไมโครคอนโทรลเลอร์ของ Atmel วงจรของบอร์ดตีพิมพ์แบบเปิดเผยวงจรภายใต้การอนุญาตสามารถนำไปดัดแปลงต่อขยาย และเพิ่มประสิทธิภาพ เพื่อศึกษาการทำงานของได้ฟรี

อาร์ดูโน เป็นบอร์ดไมโครคอนโทรลเลอร์โดยใช้ เอวีอาร์ ขนาดเล็กเป็นตัวประมวลผลและสั่งงาน เหมาะสำหรับนำไปใช้ในการศึกษาเรียนรู้ระบบไมโครคอนโทรลเลอร์ และนำไปประยุกต์ใช้งานเกี่ยวกับการควบคุมอุปกรณ์อินพุตและเอาต์พุตต่างๆได้มากมาย ทั้งในแบบที่เป็นการทำงานตัวเดียวอิสระ หรือเชื่อมต่อสั่งงานร่วมกับอุปกรณ์อื่นๆ เช่น คอมพิวเตอร์ พีซี ทั้งนี้ก็เนื่องมาจากว่า อาร์ดูโน สนับสนุนการเชื่อมต่อกับอุปกรณ์อินพุตและเอาต์พุตต่างๆได้มากมาย ทั้งแบบดิจิทัล และอะนาล็อก เช่น การรับค่าจากสวิตช์ หรืออุปกรณ์ตรวจจับ(Sensor) แบบต่างๆ รวมไปถึง การควบคุมอุปกรณ์เอาต์พุตต่างๆ ตั้งแต่ ไตโอดเปล่งแสง, หลอดไฟ, มอเตอร์, รีเลย์ ฯลฯ โดยระบบฮาร์ดแวร์ของอาร์ดูโน สามารถสร้างและประกอบขึ้นใช้งานได้เอง ในกรณีที่ผู้ใช้พอมีความรู้ด้านอิเล็กทรอนิกส์อยู่บ้าง หรือสามารถซื้อแผงวงจรถวายรูปที่มีการผลิตออกจำหน่ายกันในราคาที่ไม่แพง อีกทั้งยังเผยแพร่โค้ดเปิดเผย และตัวอย่างต่างๆ ให้ผู้ใช้งานไปใช้งาน หรือพัฒนาดัดแปลงต่อยอดได้โดยไม่เสียค่าใช้จ่ายใดๆ

### 2.3.1 ชิปและไอซี(IC) ภายในอาร์ดูโนที่สำคัญ

ATmega328P-PU

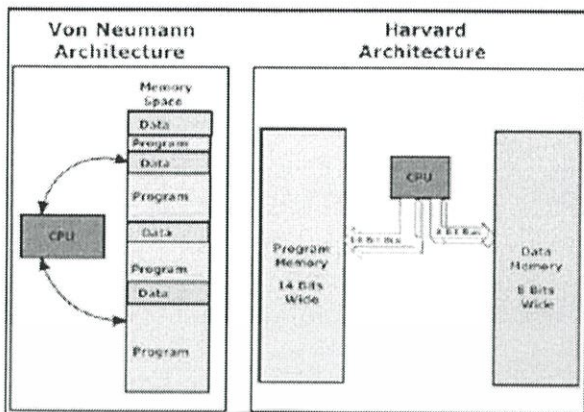
• ATmega328P-PU



รูปที่ 2.4 ชิพยี่ห้อ ATMEGA328P ขนาด 28 ขา

ตัวบอร์ด อาร์ดูโน ที่ใช้ในโปรเจกต์นี้จะกล่าวถึงสถาปัตยกรรมของ เอวีอาร์ ขนาด 8 บิต โดยในสถาปัตยกรรม เอวีอาร์ ซึ่งออกแบบโดย ATMEL เมื่อปี 1996 เป็นชิพยี่ห้อแบบอาร์ไอเอสซี(Reduced Instruction Set Computer) มีสถาปัตยกรรมการต่อหน่วยความจำแบบฮาร์วาร์ด(Harvard) ซึ่งแยกหน่วยความจำโปรแกรม และหน่วยความจำ ข้อมูลออกจากกันโดยเด็ดขาด ดังแสดงในรูปที่ 2.2 โดยใช้หน่วยความจำแบบแฟลช(Flash) สำหรับเป็นหน่วยความจำโปรแกรม และใช้หน่วยความจำแบบแอสแรม (SRAM) สำหรับหน่วยความจำข้อมูล และ นอกจากนี้ยังมีหน่วยความจำ แบบอีพีพีพร้อม (EEPROM) ซึ่งสามารถเก็บข้อมูลเอาไว้ได้โดยไม่จำเป็นต้องมีไฟเลี้ยงอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 เปรียบเทียบการจัดการหน่วยความจำของสถาปัตยกรรมแบบ Von-Neumann และ Harvard

จะเห็นว่าโปรเซสเซอร์ที่ใช้สถาปัตยกรรมแบบฮาร์วาร์ด จะแยกหน่วยความจำสำหรับเก็บข้อมูลออกจากโปรแกรมอย่างชัดเจน สถาปัตยกรรมเอวีอาร์ และ เอ็มซีเอส51(MCS-51) จะใช้รูปแบบนี้ในการจัดการหน่วยความจำ ส่วนสถาปัตยกรรมแบบ Von-neumann การตัดสินใจว่าจะเก็บโปรแกรมหรือข้อมูลจะแบ่งเก็บอย่างไรจะทำได้โดยอิสระ โดยขึ้นอยู่กับโปรแกรมเมอร์ หรือ อาจจะเป็นระบบปฏิบัติการเป็นผู้ดำเนินการให้ลักษณะเด่นของสถาปัตยกรรมเอวีอาร์คือ คาสั่งส่วนใหญ่สามารถทำงานได้เสร็จภายใน 1 รอบพลิกฟลอป ตัวซีพียูเอวีอาร์ขนาด 8 บิตจะแบ่งออกเป็นประเภทการใช้งานได้ 5 กลุ่ม ได้แก่

- ไทนี่เอวีอาร์(tiny avr) เป็นซีพียูในรุ่นเล็ก ซึ่งต้องการความเล็กกะทัดรัดของวงจร โดยเหมาะกับระบบควบคุมขนาดเล็กๆ ที่ต้องการหน่วยความจำและวงจรสนับสนุนไม่มากนัก ซีพียูในรุ่นนี้จะมีราคาถูกกว่ากลุ่มอื่น
- เมกะเอวีอาร์(mega avr) จะมีชื่ออีกอย่างว่า ATmega โดยมีวงจรสนับสนุนภายในเพิ่มเติมตลอดจนเพิ่มขนาดของหน่วยความจำให้ใช้งานมากกว่าตระกูลไทนี่ เหมาะกับงานควบคุมทั่วไป
- เอ็กซ์เมกะ(XMEGA) เพิ่มความละเอียดของวงจรดิจิทัลและอะนาล็อก จากปกติมีความละเอียด 10 บิตในรุ่นเล็กกว่าเป็น 12 บิต และวงจรควบคุมดีเอ็มเอ(DMA controller) ซึ่งช่วยลดภาระของซีพียูในการควบคุมการรับส่งข้อมูลระหว่างอุปกรณ์อินพุตและเอาต์พุต กับหน่วยความจำ
- FPSLIC (เอวีอาร์ core with FPGA) สำหรับงานที่ต้องการควบคุมที่ต้องการความยืดหยุ่นในขั้นตอนการออกแบบและพัฒนา โดยผู้ออกแบบสามารถออกแบบวงจรในระดับฮาร์ดแวร์เพิ่มเติมด้วยภาษาบรรยายฮาร์ดแวร์ (HDL: Hardware Description Language) เช่น ภาษา VHDL หรือภาษา Verilog และให้วงจรที่ออกแบบทำงานร่วมกับซีพียูเอวีอาร์
- คุณสมบัติแอปพลิเคชันเอวีอาร์ เป็นซีพียูที่ออกแบบมาโดยเพิ่มวงจรควบคุมเฉพาะด้านเข้าไปซึ่งไม่พบในซีพียูกลุ่มอื่นๆ เช่นวงจรควบคุมยูเอสบี(USB controller) หรือวงจรแคนบัส(CAN bus) เป็นต้น

(PCINT14/RESET) PC6	1	28	PC5 (ADC5/SCL/PCINT13)
(PCINT16/RXD) PD0	2	27	PC4 (ADC4/SDA/PCINT12)
(PCINT17/TXD) PD1	3	26	PC3 (ADC3/PCINT11)
(PCINT18/INT0) PD2	4	25	PC2 (ADC2/PCINT10)
(PCINT19/OC2B/INT1) PD3	5	24	PC1 (ADC1/PCINT9)
(PCINT20/XCK/T0) PD4	6	23	PC0 (ADC0/PCINT8)
VCC	7	22	GND
GND	8	21	AREF
(PCINT6/XTAL1/TOSC1) PB6	9	20	AVCC
(PCINT7/XTAL2/TOSC2) PB7	10	19	PB5 (SCK/PCINT5)
(PCINT21/OC1B/T1) PD5	11	18	PB4 (MISO/PCINT4)
(PCINT22/OC0A/AIN0) PD6	12	17	PB3 (MOSI/OC2A/PCINT3)
(PCINT23/AINT1) PD7	13	16	PB2 (SS/OC1B/PCINT2)
(PCINT0/CLKIO/CP1) PB0	14	15	PB1 (OC1A/PCINT1)

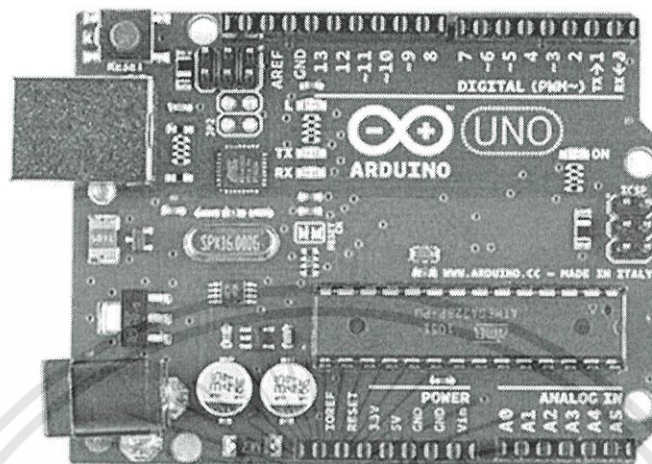
รูปที่ 2.6 แสดงขาของบอร์ดอาร์ดูโน

ซีพียู เอวีอาร์ มีให้เลือกใช้งานหลายเบอร์ แต่ละเบอร์จะมีขนาด ราคา ความสามารถ และขนาดหน่วยความจำตลอดจนถึงวงจรสนับสนุนภายในที่แตกต่างกันออกไป ในโครงการนี้จะเลือกใช้ซีพียูรุ่น ATmega328P ซึ่งมีคุณสมบัติดังนี้

- 2.3.1.1 หน่วยความจำโปรแกรมแบบแฟลช ขนาด 32 กิโลไบต์
- 2.3.1.2 หน่วยความจำข้อมูลแบบแอสรอม ขนาด 2 กิโลไบต์
- 2.3.1.3 หน่วยความจำข้อมูลแบบอีพีพีรอม ขนาด 1 กิโลไบต์
- 2.3.1.4 สนับสนุนการเชื่อมต่อแบบ I2C bus
- 2.3.1.5 พอร์ตอินพุตเอาต์พุตจำนวน 23 บิต
- 2.3.1.6 วงจรสื่อสารอนุกรม
- 2.3.1.7 วงจรนับ/จับเวลาขนาด 8 บิต จำนวน 2 ตัว และขนาด 16 บิตจำนวน 1 ตัว
- 2.3.1.8 สนับสนุนช่องสัญญาณสำหรับสร้างควมกว้างพัลส์(Pulse Width Modulation) จำนวน 6 ช่อง
- 2.3.1.9 วงจรแปลงอนาลอกเป็นดิจิตอลขนาด 10 บิตในตัวจำนวน 8 ช่อง
- 2.3.1.10 ทำงานได้ตั้งแต่ย่านแรงดัน 1.8-5.5 โวลต์
- 2.3.1.11 ความถี่ใช้งานสูงสุด 20 เมกะเฮิรซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.3.2 บอร์ดอาร์ดูโน ยูโน อาร์3(Arduino Uno R3)



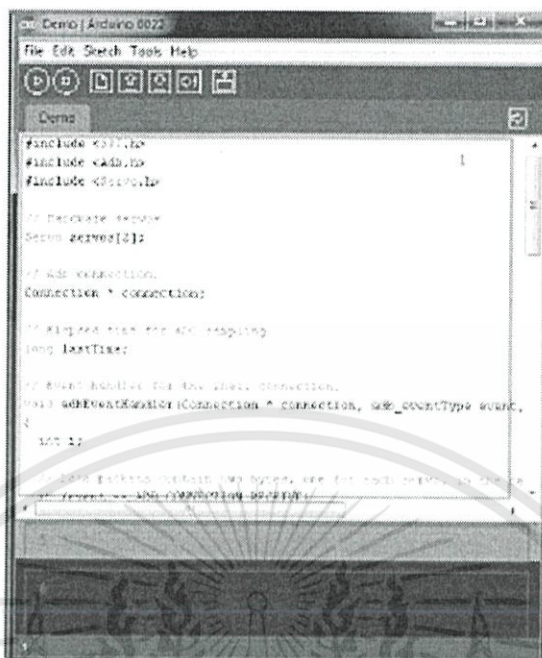
รูปที่ 2.7 อาร์ดูโน ยูโน อาร์3

### 2.3.2.1 คุณสมบัติ

- 2.3.2.1.1 ATmega328 เป็นไมโครคอนโทรเลอร์
- 2.3.2.1.2 ไฟเลี้ยง 7 -12 โวลต์
- 2.3.2.1.3 14 ขาดิจิตอลอินพุตและเอาต์พุต (6 PWM เอาต์พุต)
- 2.3.2.1.4 6 ขาอะนาล็อก
- 2.3.2.1.5 32 กิโล ความจำแฟลช (Flash memory)
- 2.3.2.1.6 16 เมกะเฮิรตซ์ ความเร็วพลิกฟลอป

### 2.3.2.2 ซอฟต์แวร์(Software) อาร์ดูโนเอ็นไวรอนเมนต์ (Arduino Environment)

โปรแกรมอาร์ดูโนเอ็นไวรอนเมนต์ เป็นซอฟต์แวร์ที่พัฒนาขึ้นมาเพื่อการเขียนโปรแกรมลงบนบอร์ด อาร์ดูโน ซึ่งจะสามารถหาการดาวน์โหลดได้ที่ <http://อาร์ดูโน.cc/en/Main/Software> โดยจะสามารถรองรับได้ทั้งระบบปฏิบัติการวินโดว, ลินุกส์ และแมก และภายในเว็บไซต์ก็จะมีตัวอย่างให้ทดลองเล่นมากมาย



รูปที่ 2.8 แสดงหน้าต่างโปรแกรมพัฒนาบอร์ดอาร์ดูโน

หน้าต่างของโปรแกรมจะมีส่วนประกอบดังนี้

หมายเลข 1 : ช่องแสดงโค้ด

หมายเลข 2 : จอแสดงผล

หมายเลข 3 : ปุ่มอัปโหลด

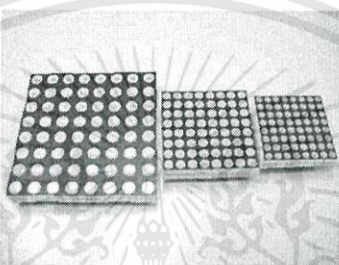
หมายเลข 4 : ปุ่มคอมไพเลอร์ (Compiler)

หมายเลข 5 : ช่องแสดงสถานะโปรแกรม

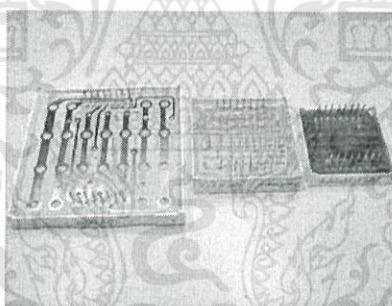
หมายเลข 6 : ช่องแสดงความผิดพลาด (Error)

## 2.4 8x8 แอลอีดีเมทริกซ์

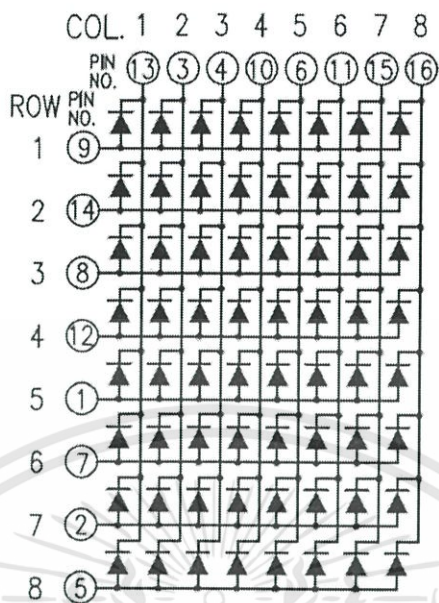
อุปกรณ์ไดโอดเปล่งแสงสีเดียวแบบอาร์เรย์ ขนาด 8x8 ที่ได้เลือกนำมาเป็นตัวอย่าง (โมเดล TOM-1588BMY-B ของบริษัท Taiwan Oasis เปิดดูไฟล์ [ข้อมูลsheet](#)) มีขาในแนวนอน (Rows) 8 ขา สำหรับต่อกับขั้วแอโนด และมีขาในแนวตั้งหรือคอลัมน์ (Columns) อีก 8 ขา สำหรับต่อกับขั้วแคโทด (Row Anode, Column Cathode) ขาในแนวนอน จะนำไปต่อผ่านตัวต้านทานอีก 8 ตัว เช่น ขนาด 220 โอห์ม (ใช้เป็น current-limiting resistors)



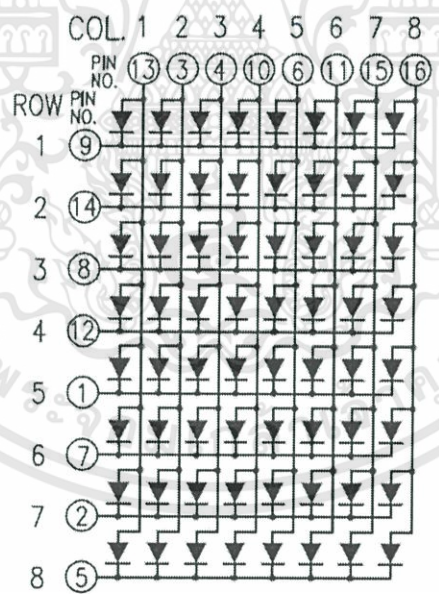
รูปที่ 2.9 แสดงตัวอย่างอุปกรณ์ 8x8 แอลอีดีเมทริกซ์ที่มีขนาดแตกต่างกัน (ด้านหน้า)



รูปที่ 2.10 แสดงตัวอย่างอุปกรณ์ (ด้านหลัง) และตำแหน่งของขาสัญญาณ (มีแถวละ 8 ขา แบ่งเป็นแถวบนและแถวล่าง)



รูปที่ 2.11 แสดงผังวงจรของ 8x8 แอลอีดีเมทริกซ์ แบบแถวแอนนอด, คอลัมแคโทด



รูปที่ 2.12 แสดงผังวงจรของ 8x8 แอลอีดีเมทริกซ์ แบบคอลัมแอนนอด, แถวแคโทด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าลองป้อนแรงดัน +5 โวลต์ ให้ขาทั้งหมดในแนวนอน (ขา R1..R8) และต่อขาทั้งหมดในแนวตั้ง (C1..C8) ไปยังกราวด์ จะทำให้แอลอีดี ทั้ง 64 ดวง อยู่ในสถานะ "ติด" (ON) แต่ถ้าลองเปลี่ยนให้ขาใดขาหนึ่งในแนวตั้งต่อกับ +5 โวลต์ จะทำให้แอลอีดี 8 ดวง ในคอลัมน์นั้น "ไม่ติด" (OFF)

ขาแต่ละขาในแนวตั้ง จะทำหน้าที่เป็นตัวรับหรือผ่านกระแส (Current Sink) ให้ไหลไปยังกราวด์ของระบบได้ และปริมาณกระแสสูงสุดที่ไหลผ่านขาแนวตั้งแต่ละขา จะเกิดขึ้นเมื่อแอลอีดีในแถวแนวตั้ง "ติด" พร้อมทั้ง 8 ดวง สมมุติว่า แรงดันไฟ (Forward Voltage) ของ แอลอีดี เท่ากับ 2 โวลต์ โดยประมาณ จะมีกระแสไหลผ่าน แอลอีดี หนึ่งตัว (และผ่านตัวต้านทาน) เท่ากับ  $(5V-2V)/220 = 13.6$  มิลลิแอมป์ (mA) ดังนั้นถ้า แอลอีดี ทั้ง 8 ดวงในแถวแนวตั้งเดียวกัน อยู่ในสถานะ "ON" พร้อมกัน จะมีกระแสไหล 108.8 มิลลิแอมป์ ต่อแถวแนวตั้ง

การควบคุมการเปิด-ปิด แอลอีดี ด้วยไอซี ULN2803A และ 74HC595

เนื่องจากกระแสที่ไหลลง กราวด์ ผ่านขาในแนวตั้งแต่ละขา อาจมีปริมาณสูงเกิน 100 มิลลิแอมป์ ดังนั้นจึงต้องเลือกใช้ไอซีที่ทำหน้าที่เป็นสวิตช์เปิด-ปิดการไหลกระแสให้เหมาะสม ในกรทดลองนี้ ได้เลือกใช้ไอซี ULN2803A (Darlington Array IC) ซึ่งมีขาเอาต์พุตจำนวน 8 ขา และสามารถแต่ละขาสามารถผ่านกระแสลง กราวด์ ได้หลายร้อยมิลลิแอมป์ (เปิดดูไฟล์ [ข้อมูล sheet](#))

ขาในแนวนอน (8 ขา) และแนวตั้ง (8 ขา) รวมทั้งหมด 16 ขา จะต้องนำไปต่อกับขาดิจิทัลของไมโครคอนโทรลเลอร์ เช่น บอร์ด อาร์ดูโน แต่สามารถลดจำนวนการใช้ขาสัญญาณได้ โดยใช้ไอซี 74HC595 (Shift Register) ซึ่งมีขาเอาต์พุตขนาด 8 บิต และใช้ทั้งหมด 2 ตัว (เปิดดูไฟล์ [ข้อมูล sheet](#))

การแสดงผลอาร์เรย์ขนาด  $8 \times 8$  นั้น จะทำทีละแถวแนวตั้ง จนครบทุกแถวแนวตั้ง แล้ววนกลับเริ่มใหม่ ถ้าทำซ้ำๆ โดยเว้นระยะเวลาให้มากในการแสดงผลแต่ละแถวแนวตั้ง จะสังเกตเห็นว่าในแต่ละช่วงเวลาจะมีเพียงแถวแนวตั้งเดียวที่ทำงาน (Active) ในขณะที่แถวอื่นจะไม่ทำงาน แต่ถ้าเว้นระยะให้น้อยลง เช่น ต่ำกว่า 1 มิลลิวินาที จะมองเห็นเสมือนว่า ทุกแถวแนวตั้งทำงานพร้อมกัน (ตามหลักการที่เรียกว่า แอลอีดี Time-Multiplexing)

## 2.5 รูปแบบในการส่งข้อมูล (transmission mode)

การส่งข้อมูลในระบบเครือข่าย สามารถทำได้ 2 ลักษณะ คือ การส่งแบบขนาน และการส่งแบบอนุกรม

การส่งแบบขนาน (parallel transmission) คือการส่งข้อมูลพร้อมกันทีละหลาย ๆ บิตในหนึ่งรอบสัญญาณนาฬิกา โดยการส่งจะรวมบิต 0 และ 1 หลาย ๆ บิตเข้าเป็นกลุ่มจำนวน  $n$  บิต ผู้ส่งส่งครั้งละ  $n$  บิต ผู้รับจะรับครั้งละ  $n$  บิตเช่นกัน ซึ่งจะคล้ายกับเวลาที่เรารู้สึกว่าคุณจะพูดเป็นคำ ๆ ไม่พูดทีละตัวอักษร

กลไกการส่งข้อมูลแบบขนานใช้หลักการง่าย ๆ เมื่อส่งครั้งละ  $n$  บิต ต้องใช้สาย  $n$  เส้น แต่ละบิตมีสายของตนเอง ในการส่งแต่ละครั้งทุกเส้นต้องใช้สัญญาณนาฬิกาอันเดียวกัน ทำให้สามารถส่งออกไปยังอุปกรณ์อื่นพร้อมกันได้



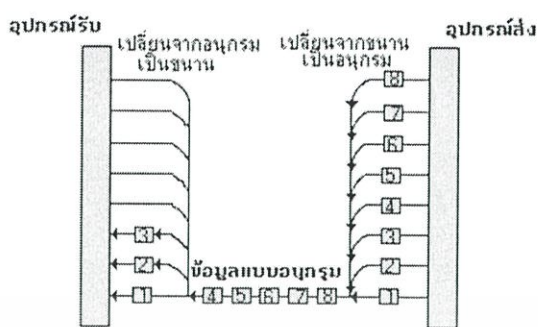
รูปที่ 2.13 แสดงการส่งข้อมูลแบบขนาน

โดยให้  $n=8$  โดยทั่วไปแล้วปลายของสายทั้ง 2 ข้างจะถูกต่อกับคอนเน็กเตอร์ด้านละ 1 ตัว ข้อดีของการส่งข้อมูลแบบขนานคือ ความเร็ว เพราะส่งข้อมูลได้ครั้งละ  $n$  บิต ดังนั้น ความเร็วจึงเป็น  $n$  เท่าของการส่งแบบอนุกรม แต่ข้อเสียที่สำคัญคือ ค่าใช้จ่าย ทั้งนี้เพราะต้องใช้สายจำนวน  $n$  เส้น

ตัวอย่างการส่งข้อมูลแบบขนาน เช่น การส่งข้อมูลภายในระบบบัสของเครื่องคอมพิวเตอร์ หรือการส่งข้อมูลจากเครื่องคอมพิวเตอร์ไปยังเครื่องพิมพ์ (printer) เป็นต้น

### 2.5.1 การส่งข้อมูลแบบอนุกรม (serial transmission)

จะใช้วิธีการส่งทีละ 1 บิตในหนึ่งรอบสัญญาณนาฬิกา ทำให้ดูเหมือนว่าบิตต่าง ๆ เรียงต่อเนื่องกันไปจากอุปกรณ์หนึ่งไปยังอีกอุปกรณ์หนึ่ง ดังรูป

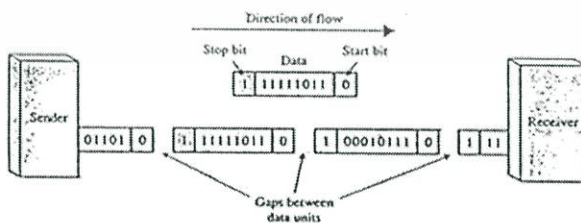


รูปที่ 2.14 แสดงการรับส่งข้อมูลแบบอนุกรม

ข้อดีของการส่งข้อมูลแบบอนุกรม คือการใช้ช่องทางการสื่อสารเพียง 1 ช่อง ทำให้ลดค่าใช้จ่ายลง แต่ข้อเสียคือ ความเร็วของการส่งที่ต่ำ ตัวอย่างของการส่งข้อมูลแบบอนุกรม เช่น โมเด็มจะใช้การส่งแบบอนุกรมเนื่องจากในสัญญาณโทรศัพท์มีสายสัญญาณเส้นเดียว และอีกเส้นหนึ่งเป็นสายดิน

2.5.1.1 การส่งข้อมูลแบบอนุกรม แบ่งได้เป็น 2 แบบ ดังนี้

2.5.1.2 การส่งข้อมูลแบบอะซิงโครนัส (asynchronous transmission) เป็นการส่งข้อมูลและผู้รับและผู้ส่ง "ไม่ต้องใช้สัญญาณนาฬิกาเดียวกัน แต่ข้อมูลที่รับต้องถูกแปลตามรูปแบบที่ได้ตกลงกันไว้ก่อน เนื่องจากไม่ต้องใช้สัญญาณนาฬิกาเดียวกันทำให้ผู้รับไม่สามารถคาดการณ์ได้ว่าเมื่อใดจะมีข้อมูลส่งมาให้ ดังนั้นผู้ส่งจึงจำเป็นต้องแจ้งผู้รับให้ทราบว่าจะมีการส่งข้อมูลมาให้โดยการเพิ่มบิตพิเศษเข้ามาอีกหนึ่งบิต เอาไว้ก่อนหน้าบิตข้อมูล เรียกว่า บิตเริ่ม (start bit) โดยทั่วไปมักใช้บิต 0 และเพื่อให้ผู้รับทราบจุดสิ้นสุดของข้อมูลจึงต้องมีการเพิ่มบิตพิเศษอีกหนึ่งบิตเรียกว่าบิตจบ (stop bit) มักใช้บิต 1 นอกจากนี้แล้วการส่งข้อมูลแต่ละกลุ่มต้องมีช่องว่างระหว่างกลุ่ม โดยช่องว่างระหว่างไบต์อาจใช้วิธีปล่อยให้ช่องสัญญาณว่าง หรืออาจใช้กลุ่มของบิตพิเศษที่มีบิตจบก็ได้ รูปต่อไปนี้แสดงการส่งข้อมูลแบบอะซิงโครนัส ให้บิตเริ่มเป็นบิต 0 บิตจบเป็นบิต 1 และให้ช่องว่างแทนไม่มีการส่งข้อมูล (สายว่าง)



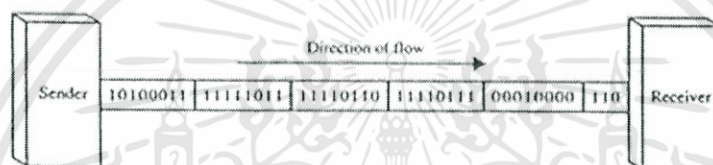
(ที่มา: Fourouzan, 1998 : 124 )

รูปที่ 2.15 การส่งข้อมูลแบบอะซิงโครนัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อดีของการส่งข้อมูลแบบอะซิงโครนัส มี 2 ประการ คือ ค่าใช้จ่ายถูกและมีประสิทธิภาพ การส่งข้อมูลแบบนี้จะนำไปใช้ในการสื่อสารที่ต้องการใช้ความเร็วไม่สูงนัก ตัวอย่างเช่น การติดต่อระหว่างเครื่องคอมพิวเตอร์กับเครื่องปลายทาง (terminal) ที่โดยธรรมชาติแล้วเป็นการสื่อสารแบบอะซิงโครนัส เพราะผู้ใช้จะพิมพ์ทีละ 1 ตัวอักษรจากเครื่องปลายทางไปยังเครื่องคอมพิวเตอร์จึงไม่ต้องใช้ความเร็วสูงในการติดต่อสื่อสาร

2.5.1.3 การส่งข้อมูลแบบซิงโครนัส (synchronous transmission) เป็นการส่งบิต 0 และ 1 ที่ต่อเนื่องกันไปโดยไม่มีการแบ่งแยก ผู้รับต้องแยกบิตเหล่านี้ออกมาเป็นไบต์ หรือเป็นตัวอักษรเอง



(ที่มา : Fourouzan, 1998 : 125)

รูปที่ 2.16 รูปแสดงการส่งข้อมูลแบบซิงโครนัส

จากภาพแสดงการส่งข้อมูลแบบซิงโครนัส ผู้ส่งทำการส่งบิตติดต่อกันยาว ๆ ถ้าผู้ส่งต้องการแบ่งช่วงกลุ่มข้อมูลก็ส่งกลุ่มบิต 0 หรือ 1 เพื่อแสดงสถานะว่าง เมื่อแต่บิตมาถึงผู้รับ ผู้รับจะนับจำนวนบิตแล้วจับกลุ่มของบิตให้เป็นไบต์ที่มี 8 บิต

การส่งข้อมูลแบบซิงโครนัสมีประสิทธิภาพสูงกว่าแบบอะซิงโครนัสมาก และทำให้มีการใช้ความสามารถของสายสื่อสารได้เกือบทั้งหมด ข้อดีของการส่งข้อมูลแบบซิงโครนัส คือความเร็วในการส่งข้อมูล ทั้งนี้เพราะไม่มีบิตพิเศษหรือช่องว่างที่ไม่ได้ถูกนำไปใช้เมื่อถึงผู้รับ จึงทำให้ความเร็วของการส่งข้อมูลแบบซิงโครนัสเร็วกว่าแบบอะซิงโครนัส ด้วยเหตุนี้จึงมีการนำไปใช้งานที่ต้องการความเร็วสูง เช่น การส่งข้อมูลระหว่างเครื่องคอมพิวเตอร์

## 2.6 ชิฟรีจิสเตอร์(Shift Register)

2.6.1 ตัวเลื่อนข้อมูลจัดเป็นการนำฟลิปฟลอป(flip flop) ไปใช้งานได้อย่างหนึ่ง การเลื่อนข้อมูลจะมีลักษณะแตกต่างกันดังนี้

2.6.1.1 เข้าขนาน ออกขนาน (Parallel In Parallel ออก หรือ PIPO)

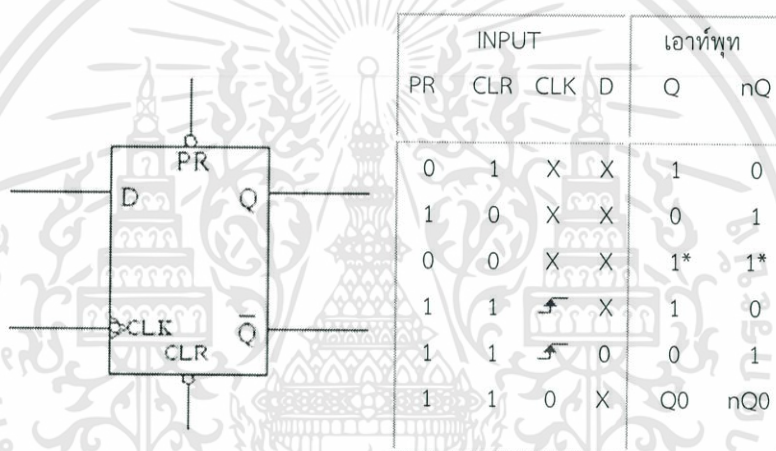
2.6.1.2 เข้าขนาน ออกอนุกรม (Parallel In Serial ออก หรือ PISO)

2.6.1.3 การเข้าอนุกรม ออกขนาน (Serial In Parallel ออก หรือ SIPO)

2.6.1.4 เข้าอนุกรม ออกอนุกรม (Serial In Serial ออก หรือ SISO)

2.6.1.5 การรับส่งข้อมูลในระบบคอมพิวเตอร์

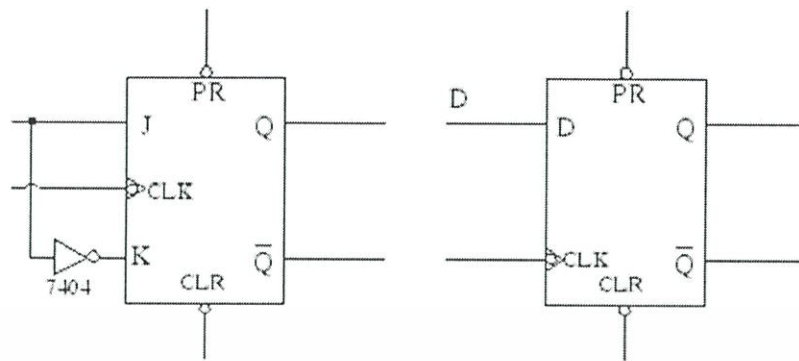
ตัวเลื่อนข้อมูลดังกล่าวแล้วข้างต้นใช้โครงสร้างจาก ฟลิปฟลอป ซึ่งแบ่งออกเป็น R-S ฟลิปฟลอป , D ฟลิปฟลอป หรือ J-K ฟลิปฟลอป ก็ได้



1\* ไม่ stable

รูป 2.17 โครงสร้างและตารางเอาต์พุตของ 7474 ซึ่งเป็นฟลิปฟลอป ชนิด D  
 $nQ = \bar{Q} = Q$  complement ,  $nQ0 = \bar{Q0} = Q0$  complement

โดยโครงสร้างและลอจิกของ JK ฟลิปฟลอป สามารถจะนำมาดัดแปลงให้มีคุณสมบัติเทียบเท่ากับ D ฟลิปฟลอป ได้ดังแสดงในรูป



7476 JK-ฟลิปฟลอป

7473 D-ฟลิปฟลอป

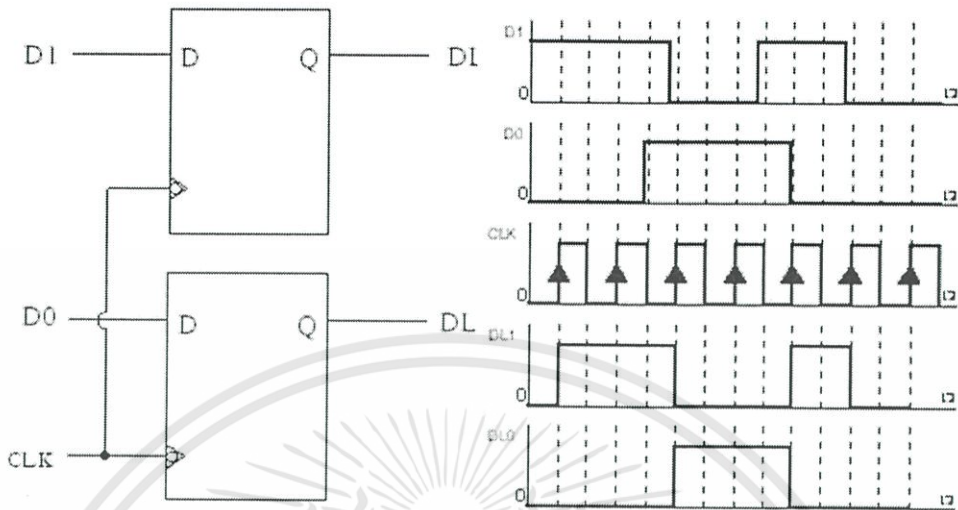
รูป 2.18 แสดงการดัดแปลง JK ฟลิปฟลอป ให้มีคุณสมบัติแบบเดียวกับ D ฟลิปฟลอป

### 2.6.2 ตัวเลื่อนข้อมูลแบบเข้าขนานและ ออกขนาน

ตัวเลื่อนข้อมูลในลักษณะนี้ จะนำมาใช้เป็นตัวจับบันทึก หรือคงค่า (latch) ข้อมูล เพื่อให้เห็นภาพของการใช้งานชัดเจน ขอให้ดูตัวอย่างดังต่อไปนี้

#### 2.6.2.1 ตัวบันทึกข้อมูลแบบ 2 บิต

จากคุณสมบัติดังตารางในรูป 1 เรานำฟลิปฟลอปดังกล่าวแล้วมาสร้างเป็นตัวรับข้อมูลแบบ 2 บิต จะได้ดังรูป



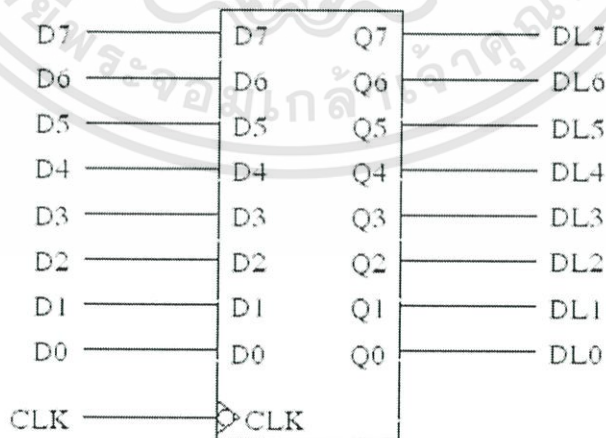
รูป 2.19 ตัวรับข้อมูลแบบ 2 บิต

รูป 2.20 แสดงความสัมพันธ์ระหว่างข้อมูลฟลิปฟล็อป และข้อมูลเอาต์พุต

ในรูป 5 เป็นความสัมพันธ์ระหว่างสัญญาณเอาต์พุต กับ สัญญาณ ข้อมูล ที่สัญญาณ คล็อก แสดงจังหวัดเวลาต่าง ๆ กัน โดยการเปลี่ยนแปลงจะเกิดขึ้นที่ขอบขาขึ้นของสัญญาณพัลส์ จากรูป 4 เป็นตัวอย่างของตัวบันทึกข้อมูลแบบ 2 บิต หากจะให้จำนวนบิตมากขึ้น ก็เพียงแค่เพิ่มจำนวนฟลิปฟล็อป แล้วใช้สัญญาณ คล็อก ร่วมกัน

2.6.2.2 ตัวบันทึกข้อมูล แบบ 8 บิต

จากหลักการในรูป 4 สามารถดัดแปลงให้รับข้อมูลแบบ 8 บิตได้โดยใช้ D-ฟลิปฟล็อป จำนวน 8 ตัวมาต่อขนานกันแล้ว ใช้สัญญาณ คล็อกร่วมกัน จะได้ตัวรับข้อมูลแบบ 8 บิต



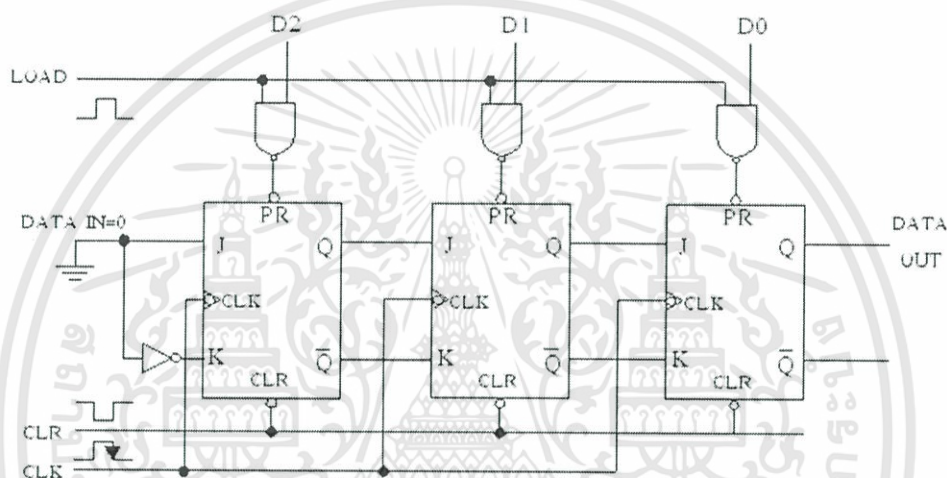
รูป 2.21 ตัวรับข้อมูลแบบ 8 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แม้จะใช้ฟลิปฟล็อปจำนวนหลายตัวต่อเป็นตัวบันทึกข้อมูลแบบ 8 บิตก็ตาม แต่ก็มีไอซีชนิด 8 บิต แลทช์(LATCH) สำเร็จรูปให้ใช้หลายตัว เช่น 74LS373 หรือ 74LS374 ซึ่งใช้เป็นตัวบันทึกข้อมูลสำหรับไอซีไมโครโปรเซสเซอร์

### 2.6.3 ตัวเลื่อนข้อมูลแบบเข้าขนานออกอนุกรม

การส่งข้อมูลให้กับตัวเลื่อนชนิดนี้ ทำได้โดยการส่งข้อมูลให้ตัวเลื่อนแบบขนาน และรับข้อมูลจากตัวเลื่อนแบบอนุกรม



รูป 2.22 ตัวเลื่อนข้อมูลชนิดรับส่งข้อมูลเข้าแบบขนาน และส่งข้อมูลแบบอนุกรม

โดยใช้ฟลิปฟล็อป ชนิด J - K

จากรูป 7 ข้อมูลที่ต้องการจะ โหลด เข้ามาแบบขนานจะต้องป้อนเข้ามารอไว้ที่ D2 D1 D0 เมื่อได้สัญญาณ โหลด เป็น "1" จะได้รับการ โหลด เป็น "0" จะยังคงสภาพ "0" อยู่เหมือนเดิม

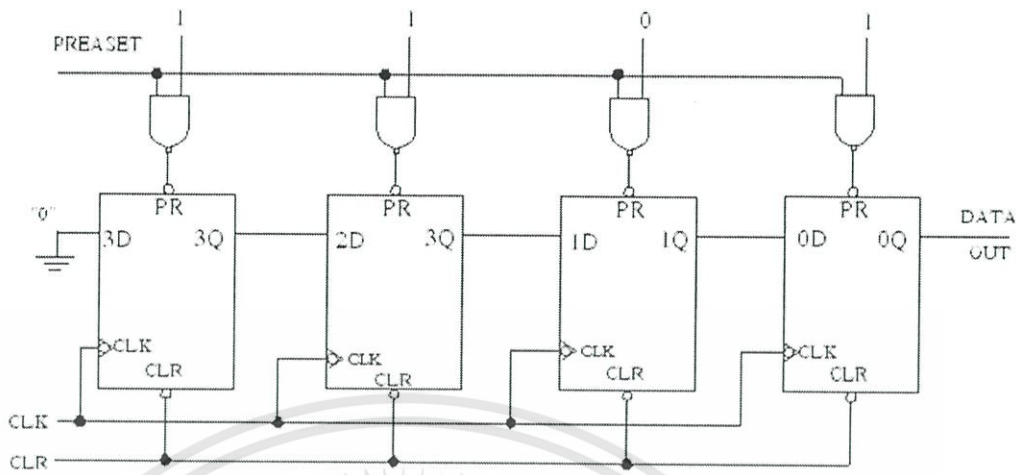
ลำดับขั้นตอนในการใช้เข้าขนาน ออกอนุกรม ตามโครงสร้างนี้จะต้องทำตามลำดับขั้นดังนี้

2.6.3.1 เคลียร์(CLEAR) วงจร โดยส่งสัญญาณเคลียร์ เป็นลอจิก "0" ช่วงขณะ (ปกติ ขานี้จะต้องเป็น "1" เพื่อให้ JK ฟลิปฟล็อป ทำงาน)

2.6.3.2 โหลด ค่าที่ต้องการ

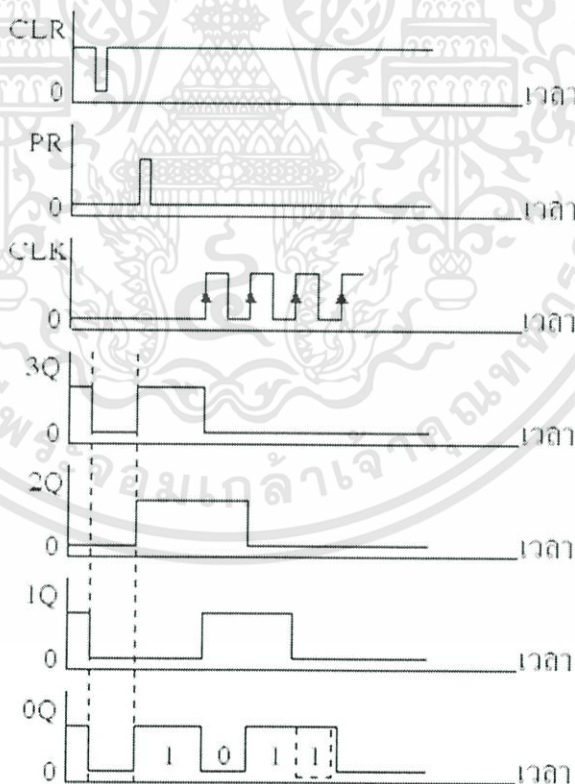
2.6.3.3 ชิฟท์(SHift) ข้อมูลตามกระบวนการ

ต่อไปนี้เป็นตัวอย่างการใช้ตัวเลื่อนข้อมูลแบบรับส่งข้อมูลแบบขนานแล้วส่งออกแบบอนุกรม โดยใช้ฟลิปฟล็อป ชนิด D



รูป 2.23 ตัวรับข้อมูลแบบขนาน 4 บิต และส่งออกอนุกรม

เมื่อให้สัญญาณเคลียร์ เอาท์พุทของทุกตัวมาเป็นลอจิก "0" เมื่อ ให้สัญญาณเริ่ม ตัวที่มีข้อมูลเข้ามาเป็น "1" จะมีเอาท์พุท Q เป็นลอจิก 1 ด้วย ให้ดูรูปประกอบ



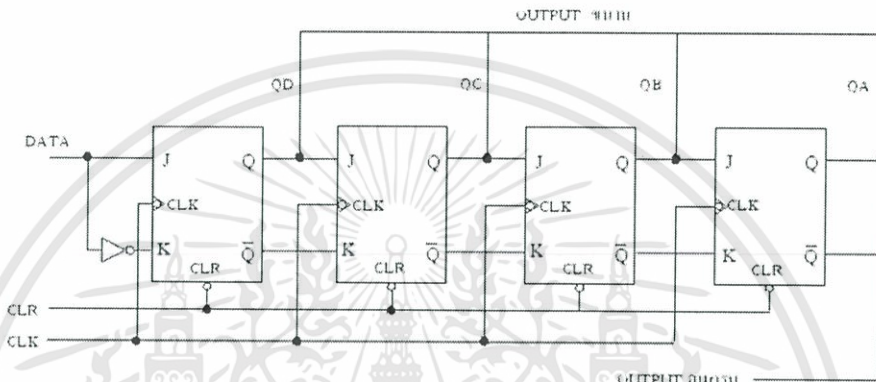
รูป 2.24 แสดงการเลื่อนของข้อมูล 1001 ออกไปทางขวา Q0 โดยบิตที่มีค่าต่ำสุด (LSB) จะเลื่อนออกไปก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การส่งข้อมูลในลักษณะนี้ตัวข้อมูลจะเลื่อนออกไปทางสายสัญญาณเพียงเส้นเดียว โดยบิตที่มีค่าต่ำสุด (Least Significant Bit หรือ LSB) จะเลื่อนออกไปหลังสุด จำนวนบิตของข้อมูลจะกำหนดจำนวนทศของ ฟลิปฟลอป และจำนวนสัญญาณนาฬิกาที่ต้องใช้ ในการเลื่อนข้อมูล

#### 2.6.4 ตัวเลื่อนข้อมูลแบบส่งเข้าแบบอนุกรม และส่งออกแบบขนาน

มีโครงสร้างดังแสดงในรูป



รูป 2.25 โครงสร้างของตัวเลื่อนข้อมูลแบบเข้าอนุกรมและออกแบบขนาน

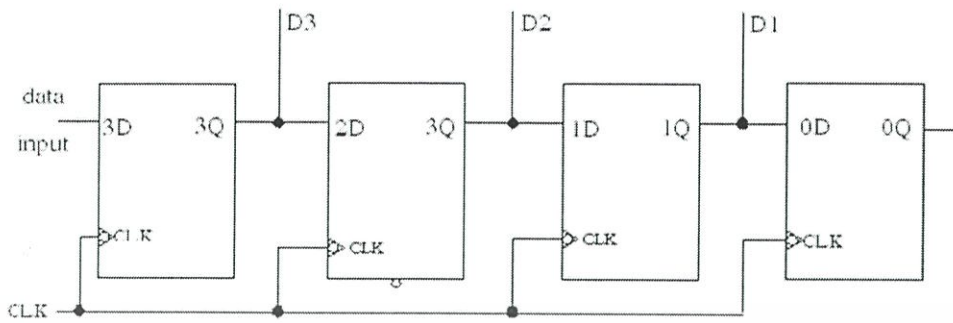
ใช้ฟลิปฟลอปชนิด J-K

2.6.4.1 ตัวเลื่อนข้อมูลแบบเข้าอนุกรม ออกขนานสามารถนำเอาท์พุทมาใช้ได้ใน 2 ลักษณะ คือ

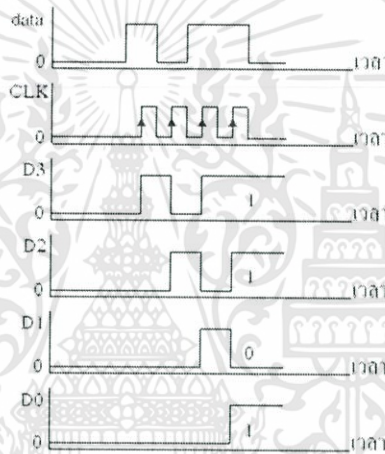
2.6.4.2 นำเอาท์พุทขนานมาใช้ คือ เอาท์พุท QA, QB, QC และ QD

2.6.4.3 นำเอาท์พุทอนุกรมมาใช้ คือ จาก QA มาใช้ การนำเอาท์พุทขนานมาใช้ จะต้องอาศัยการเลื่อน 4 จังหวะ จึงจะได้ข้อมูลมา ครบถ้วน ส่วนข้อมูลอนุกรมจะได้ครบก็ต่อเมื่อต้องอาศัยการเลื่อน 8 จังหวะ ปกติตัวเลื่อนแบบอนุกรมเข้า อนุกรมออก จะนำมาใช้เมื่อต้องการหน่วยเวลาสัญญาณเท่านั้น

ต่อไปนี้เป็นตัวอย่างการใช้ฟลิปฟลอปชนิด D ในการเลื่อนข้อมูลแบบรับเข้าอนุกรม และส่งออกแบบขนาน



รูป 2.26 การนำ D ฟลิปฟลอป มาจัดวงจรเป็นรับเข้าแบบอนุกรม และส่งออกแบบขนาน

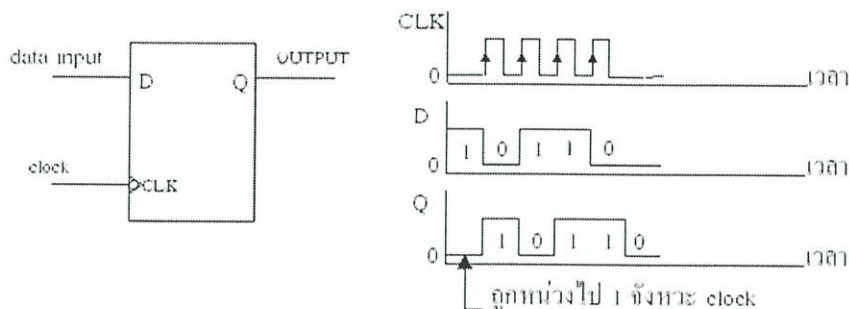


รูป 2.27 หลังจากมีสัญญาณคล็อก เข้ามาครบ 4 ลูก ข้อมูล ที่ส่งเข้ามาในจังหวะที่สัมพันธ์ กับสัญญาณคล็อก จะปรากฏออกมาที่ ขา D3 D2 D1 และ D0

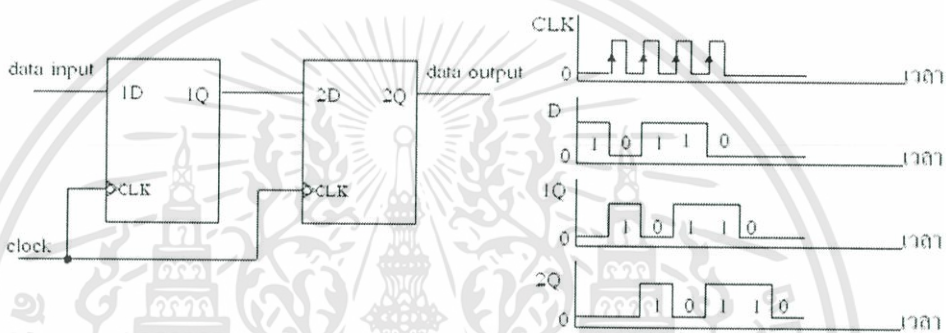
การรับข้อมูลแบบอนุกรมในลักษณะนี้สัญญาณ คล็อก ของฝ่ายรับจะต้องมีคาบเวลาเท่ากับ สัญญาณ คล็อก ของฝ่ายหลัง (แต่ไม่จำเป็นต้องเป็นสัญญาณเดียวกัน) ข้อมูลที่รับได้จึงจะเป็นข้อมูลที่ ถูกต้อง

### 2.6.5 ตัวเลื่อนข้อมูลชนิดรับเข้าแบบอนุกรมและส่งออกแบบอนุกรม

ในการส่งข้อมูลแบบอนุกรมและตัวรับแบบอนุกรมนี้ไม่ใช้ในการรับส่งข้อมูลทั่วไป แต่ใช้เพื่อ หน่วงเวลาของข้อมูล หากต้องการ หน่วงเวลาของข้อมูลไป 1 ช่วงสัญญาณนาฬิกาก็ได้ ฟลิปฟลอป ชนิด D จำนวน 1 ภาค และเพิ่มขึ้นทุกจำนวน 1 ภาค ต่อการหน่วง ไป 1 จังหวะของสัญญาณนาฬิกา อย่างไรก็ตามจำนวนบิตของข้อมูลที่ส่งต่อหน่วยเวลาจะต้องเท่ากับความถี่ของสัญญาณนาฬิกาที่ให้กับ ฟลิปฟลอปชนิดD



รูป 2.28 แสดงการหน่วงเวลาของข้อมูลโดยการส่งเข้าแบบ อนุกรมและส่งออกแบบอนุกรม



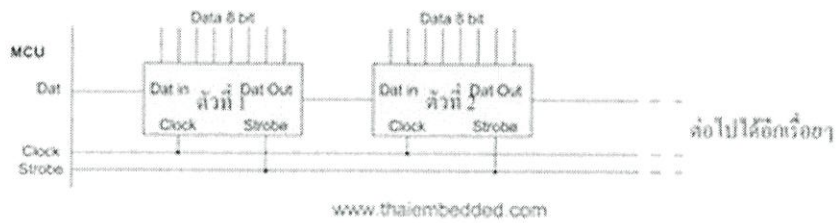
รูป 2.29 แสดงการหน่วงเวลาของข้อมูลโดยใช้ D ฟลิปฟลอป จำนวน 2 ภาค เพื่อสร้างเป็นตัวรับแบบอนุกรม และส่งออกแบบอนุกรม

ในการรับส่งข้อมูลแบบขนานนี้จะต้องตกลงกันทั้งฝ่ายรับและฝ่ายส่งว่าจะใช้อัตราเร็วในการรับส่งเท่าใดซึ่งจะต้อง ตกลงกันสองฝ่ายข้อมูลที่ได้รับจึงจะถูกต้อง

- แบบที่ 1. ให้ ข้อมูล=FF หลังจากออก (ข้อมูล) ในแต่ละครั้ง ให้เลื่อนซ้าย 1 บิต
- แบบที่ 2. ให้ ข้อมูล=FF หลังจากออก (ข้อมูล) ในแต่ละครั้ง ให้เลื่อนขวา 1 บิต
- แบบที่ 3. ให้ ข้อมูล=1 หลังจากออก (ข้อมูล) ในแต่ละครั้ง ให้เลื่อนขวา 1 บิต
- แบบที่ 4. ให้ ข้อมูล=1 หลังจากออก (ข้อมูล) ในแต่ละครั้ง ให้เลื่อนซ้าย 1 บิต

### 2.6.6 เทคนิคการขยายเอาต์พุตด้วยชิพทรีจิสเตอร์

สำหรับ ไมโครคอนโทรลเลอร์ เบอร์เล็กๆ ที่มีขา อินพุตและเอาต์พุต ไม่มากนัก เช่น 89C2051 ถ้าเราต้องการขยายพอร์ทแบบเอาต์พุต ให้ได้เยอะๆ เช่นต้องการจะต่อ 7 เซกเมนต์ (7 Segment) ซัก 30 ตัว ถึงจะใช้วิธีสแกนดิสเพล(Scan Display) ก็เยอะ ลองคำนวณดู ใช้ 8 บิต สำหรับ 7 เซกเมนต์ และในแต่ละตัวก็ต้อง มีขาคอมมอน(Common) อีก สรุปว่าคงไม่พอครับ เทคนิคหนึ่งที่น่าสนใจก็คือใช้ชิพทรีจิสเตอร์ ในการขยายพอร์ท โดยเราจะใช้ขา ข้อมูล เพียง 3 ขา เท่านั้น คล็อก, ข้อมูล, สต็อก(Stoke) แต่ละผู้ผลิตอาจจะตั้งชื่อขาสัญญาณแตกต่างกันไปบ้างแต่หลักการเดียวกัน โดยข้อมูลจะส่งไปพร้อมกับคล็อก หรือที่เรียกการส่งข้อมูลแบบนี้ว่าซิงโคนัส



รูปที่ 2.30 การส่งข้อมูลแบบซิงโครนัส

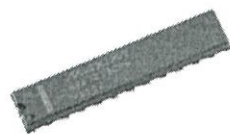
การต่อสาย 74HC595 แบบอนุกรมหลายตัว ไอซีที่นิยมนำมาใช้จะเป็นเบอร์ที่ลงท้ายด้วย 595 เช่น 74HC595, 74LS595, TPIC6B595(Open Collector) การต่อใช้งานขา ข้อมูลจาก ไมโครคอนโทรเลอร์ จะต่อเข้ากับ ข้อมูลเข้าของตัวที่ 1 และ ข้อมูลออก ของตัวที่ 1 จะต่อเข้ากับ ข้อมูลเข้าของตัวที่ 2 เป็นอย่างนี้ไปเรื่อยๆจนถึงตัวสุดท้าย ข้อมูลออก ของตัวสุดท้ายจะไม่ถูกต่อ



รูปที่ 2.31 IC 74HC595

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.6.7 ไอซีMAX7219



รูปที่ 2.32 ไอซีMAX7219

ไอซี MAX7219 สามารถนำมาใช้ควบคุมการทำงานของ 7 เซกเมนต์ มากกว่าหนึ่งหลัก หรือ แอลอีดีเมทริกซ์ แบบแคโทด ขนาด8x8 โดยใช้ขาเพียง3ขาคือขาฟลิปฟลอป(CLK), ขา โหลด(โหลด) และขาข้อมูลเข้า(DIN) และสามารถเชื่อมต่อกับไมโครคอนโทรลเลอร์ผ่านบัส เอสพีไอ(Serial Peripheral Interface)

ข้อมูลเชิงเทคนิคของไอซี MAX7219 ที่สำคัญมีดังนี้

### 2.6.7.1 คุณสมบัติ

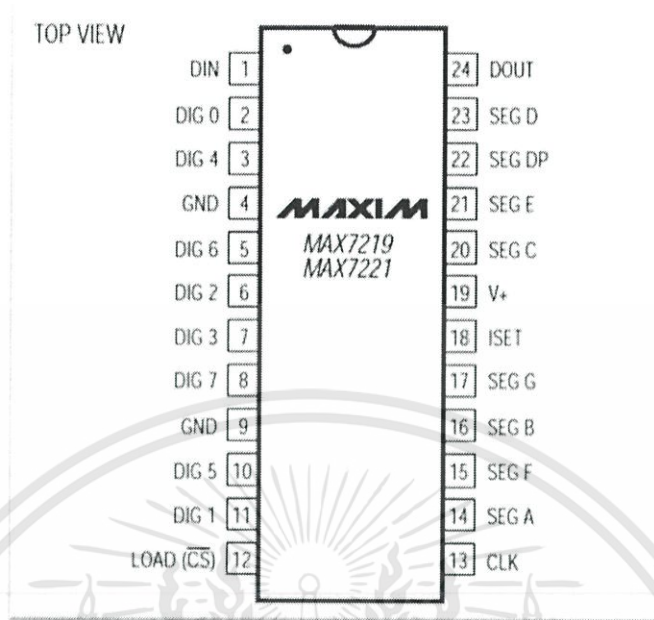
2.6.7.1.1 ใช้แรงดันไฟเลี้ยงได้ในช่วง 4 โวลต์ ถึง 5.5 โวลต์

2.6.7.1.2 ขา โหลด, คล็อก(CLK), ขาเข้า(DIN) และขาออก เป็นขาสำหรับการเลื่อนบิตข้อมูลตาม จังหวะสัญญาณฟลิปฟลอป(คล็อก) เข้าที่ขาเข้า และมีข้อมูลออกที่ขา ขาออก

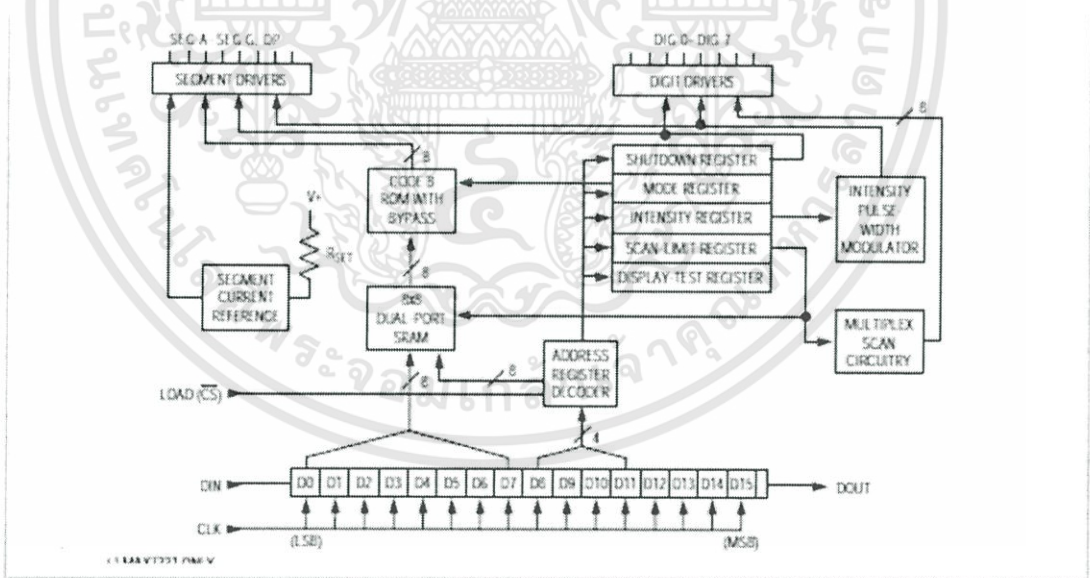
2.6.7.1.3 ขา DIG0, ..., DIG7 สามารถรับกระแสได้สูง(Current Sink) และนำไปใช้ต่อกับขาแคโทด ร่วม เพื่อเลือกDIGIT (เลือกให้หลักใดแสดงผล หรือ "active")

2.6.7.1.4 ขา SEG\_A, ..., SEG\_G, SEG\_DP เป็นขาสำหรับกำหนดสถานะติด-ดับของ แอลอีดี Matrix Display ที่ตรงกับแต่ละSegment

2.6.7.1.5 ขา ISET ให้ต่อตัวต้านทาน (เช่น 10k โอห์ม) ไปยัง  $V_{CC}$  ใช้กำหนดกระแสสูงสุดของแต่ละ Segment

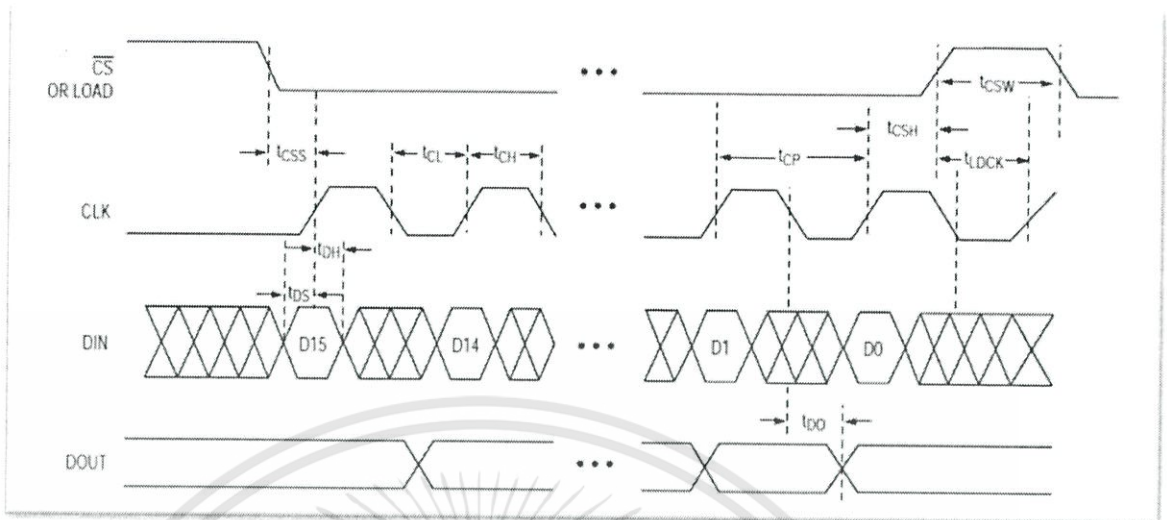


รูปที่ 2.33 ขาของไอซีMAX7219



รูปที่ 2.34 ฟังก์ชันไดอะแกรม(Functional Diagram)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.35 ทามมิ่งไดอาแกรม(Timing Diagram)

การส่งข้อมูลเข้าไปในไอซีMAX7219 เพื่อเขียนลงในรีจิสเตอร์แต่ละตัว จะใช้วิธีเลื่อนบิตตามจังหวะของสัญญาณฟลิปฟลอป โดยส่งครวละ 16บิต และส่งบิตD15เข้าไปก่อนตามลำดับ ซึ่งรูปแบบการส่งข้อมูลแบบนี้สอดคล้องกับการทำงานของเอสพีไอ ในโหมด 0

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
X	X	X	X	ADDRESS				MSB		ข้อมูล				LSB	

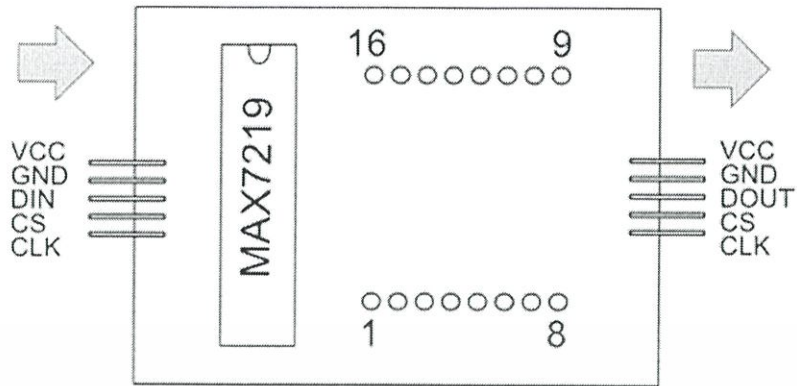
ตารางที่ 2.2 แสดงตำแหน่งของบิตในรีจิสเตอร์สำหรับคำสั่งและข้อมูลขนาด16บิต

REGISTER	ADDRESS					HEX CODE
	D15-D12	D11	D10	D9	D8	
No Op	X	0	0	0	0	X0
Digit 0	X	0	0	0	1	X1
Digit 1	X	0	0	1	0	X2
Digit 2	X	0	0	1	1	X3
Digit 3	X	0	1	0	0	X4
Digit 4	X	0	1	0	1	X5
Digit 5	X	0	1	1	0	X6
Digit 6	X	0	1	1	1	X7
Digit 7	X	1	0	0	0	X8
Decode Mode	X	1	0	0	1	X9
Intensity	X	1	0	1	0	XA
Scan Limit	X	1	0	1	1	XB
Shutdown	X	1	1	0	0	XC
Display Test	X	1	1	1	1	XF

ตารางที่ 2.3 แสดงเลขที่อยู่ของรีจิสเตอร์ภายในของไอซี MAX7219

#### 2.6.7.2 ขั้นตอนการใช้งานสำหรับแผง แอลอีดี 8x8 นั้นก็ทำได้ง่าย โดยกำหนดค่าสำหรับรีจิสเตอร์ดังนี้

- 2.6.7.2.1 รีจิสเตอร์ "Decode Mode" (0x9): 0xFF ซึ่งหมายถึง Decode for digits 7-0
  - 2.6.7.2.2 รีจิสเตอร์ "Shutdown" (0xC) : 0x01 ซึ่งหมายถึง Normal operation
  - 2.6.7.2.3 รีจิสเตอร์ "Intensity" (0xA) : 0x00 .. 0x0F ซึ่งหมายถึง ระดับความสว่างของ แอลอีดี ซึ่งสามารถเลือกได้ 16 ระดับ 0x00 = ต่ำสุด และ 0x0F = สูงสุด
  - 2.6.7.2.4 รีจิสเตอร์ "Scan Limit" (0xB) : 0x07 ซึ่งหมายถึง ใช้ขา DIG0 ถึง DIG7 ในการ "สแกน" (scan) โดยอัตโนมัติ
  - 2.6.7.2.5 รีจิสเตอร์ "Digit 0" .. "Digit 7" : ให้กำหนดค่าสำหรับสถานะติด-ดับของแอลอีดี จำนวน 8 ดวง ของหลักที่ 1 ถึง 8
- ไอซี MAX7219 จะทำขั้นตอน "scan" โดยอัตโนมัติ โดยเลือกแถวแนวนอนทีละแถวตามลำดับ และส่งข้อมูลที่อยู่ในรีจิสเตอร์ ออกที่ขา SEG\_A,... , SEG\_G, SEG\_DP สำหรับแถวแนวนอนที่ได้เลือกในแต่ละช่วงเวลา



รูปที่ 2.37 แสดงตำแหน่งจุดเชื่อมต่อสัญญาณของโมดูล MAX7219

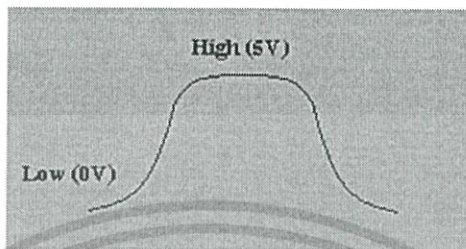
ขา 1-8 แถวล่าง จากซ้ายไปขวา: 1=DIG4, 2=DIG6, 3=D6, 4=D5, 5=DIG7, 6=D3, 7=DIG5, 8=DIG2

ขา 9-16 แถวบน จากขวาไปซ้าย: 9=DIG0, 10=D4, 11=D2, 12=DIG3, 13=D7, 14=DIG1, 15=D1, 16=D0



## 2.7 คล็อก pulse

คล็อก pulse (CK) เป็นสัญญาณ Logic ที่ทำให้ ฟลิปฟลอป แต่ละตัวมีการเปลี่ยนแปลงสถานะของ Logic ทาง เอาท์พุท ซึ่งมีรูปคลื่นและส่วนประกอบดังรูปที่ 2.42

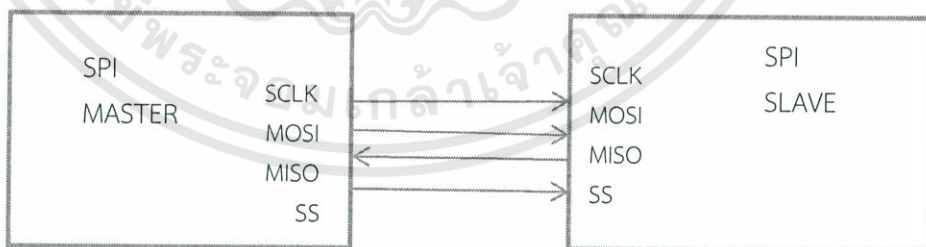


รูปที่ 2.38 แสดง พัลส์คล็อก

## 2.8 ภาพรวมของเอสพีไอ(SPI)

เอสพีไอ (Serial Peripheral Interface) คือ การสื่อสารอนุกรมแบบซิงโครนัสเป็นรูปแบบที่ใช้วิธีส่ง ข้อมูล โดยใช้สัญญาณ คล็อก มาเป็นตัวกำหนดจังหวะในการรับส่งข้อมูล

เอสพีไอทำงานในรูปแบบที่ให้อุปกรณ์ตัวหนึ่งทำหน้าที่เป็นมาสเตอร์ ในขณะที่อีกตัวหนึ่งทำหน้าที่เป็น สเลฟ และส่งข้อมูลในโหมด ฟูลดูเพล็กซ์(Full-duplex) ซึ่งหมายความว่า สัญญาณสามารถส่งหากันระหว่าง มาสเตอร์ และ สเลฟ ได้อย่างต่อเนื่อง ในการสื่อสารแบบเอสพีไอ นี้ ไม่มีมาตรฐานกำหนดตายตัวว่าข้อมูลที่ส่งหากันต้องอยู่ในรูปแบบหรือฟอร์แมตแบบไหนเป็นการคิดโปรโตคอลการสื่อสารกันเองซึ่งรูปแบบการสื่อสาร แสดงดังรูป



รูปที่ 2.39 SPI bus: single master and single

### 2.8.1 พื้นฐานการทำงานของการสื่อสารแบบเอสพีไอ

เอสพีไอต้องการสายสัญญาณ สี่เส้น บางครั้งเราเรียกว่าบัสอนุกรม "four wire" เส้นสัญญาณทั้งสี่เส้น แสดงดังตารางที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เส้น	ชื่อ	การทำงาน
SCLK	Serial คล็อก	สัญญาณนาฬิกาที่มาสเตอร์เป็นผู้ส่ง
MOSI	มาสเตอร์เอาต์พุต	เอาต์พุตจากมาสเตอร์ที่ส่งให้สเลฟ
MISO	มาสเตอร์อินพุต	เอาต์พุตจากสเลฟที่ส่งให้มาสเตอร์ (ชิป AD7730BNZ)
SS	Slave Select	เป็นสัญญาณที่มาสเตอร์ใช้เป็นตัวเลือกว่าจะติดต่อกับสเลฟตัวใด

#### ตารางที่ 2.4 สายสัญญาณต่างๆที่ใช้ในการส่งแบบเอสพีไอ

ตัวที่เป็นมาสเตอร์ ทำหน้าที่เป็นตัวควบคุมการสื่อสารทั้งหมด โดยควบคุมการสื่อสารตามสัญญาณนาฬิกา ตัวมาสเตอร์จะเป็นตัวที่ตัดสินใจ รับ หรือ ส่งข้อมูล ภายในการสื่อสาร ส่วนตัวที่เป็นสเลฟจะทำหน้าที่รับคำสั่งต่างๆจากมาสเตอร์ สายสัญญาณเส้น SS หรือ Slave select ใช้ในกรณี ที่เรามีตัว สเลฟ มากกว่า 1 ตัว โดยการทำให้เส้น SS มีระดับสัญญาณเป็นต่ำ เมื่อต้องการติดต่อกับ สเลฟตัวใด จากรูป หากต้องการติดต่อสื่อสารกับ สเลฟ ตัวใด ก็เพียงทำให้สัญญาณ SS ของ สเลฟนั้น มีระดับสัญญาณเป็นต่ำ

โดยการติดต่อสื่อสารแบบ เอสพีไอนี้จะมีข้อดี-ข้อเสีย ดังต่อไปนี้

#### 2.8.2 ข้อดีของการสื่อสารแบบเอสพีไอ

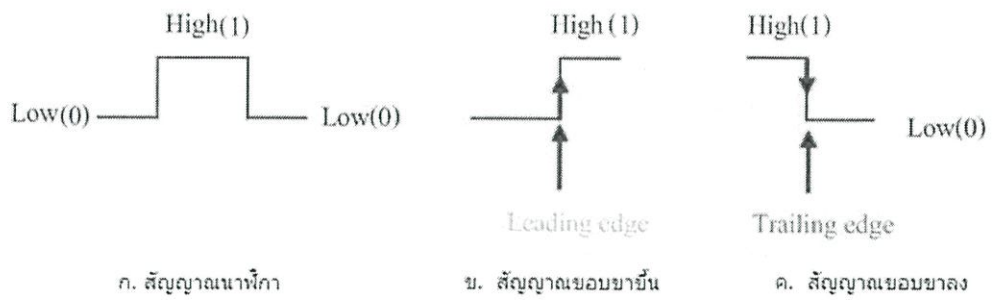
- 2.8.2.1 สามารถสื่อสารแบบฟูลดูเพล็กซ์รับ และ ส่งได้ในเวลาเดียวกัน
- 2.8.2.2 เป็นการรับส่งที่มีคุณภาพและส่งได้ที่ความเร็วสูง มีโอกาสที่ข้อมูลจะสูญหายระหว่างการส่งน้อย
- 2.8.2.3 โปรโตคอลที่ใช้มีความยืดหยุ่น ไม่มีการกำหนดจำนวนบิตที่ใช้

#### 2.8.3 ข้อเสียของการสื่อสารแบบเอสพีไอ

- 2.8.3.1 ต้องใช้สายสัญญาณมาก เพราะจะต้องส่ง คล็อก ไปด้วย
- 2.8.3.2 การใช้ชิปซีเลคท์(chip-select) ต้องมีการแชร์บัส
- 2.8.3.3 ไม่มีการเช็คโพล์คอนโทรล( flow control)

สัญญาณนาฬิกา (คล็อก Pulse : CK) เป็นสัญญาณที่เป็นตัวกระตุ้น (Trigger) ให้ฟลิปฟล็อปมีการทำงานเปลี่ยนสถานะทางด้านเอาต์พุต ส่วนมากแล้วสัญญาณนาฬิกามักเป็นสัญญาณรูปสี่เหลี่ยม (Square Wave) มีลักษณะและส่วนประกอบของสัญญาณดังรูปที่ 44

- ก. สัญญาณนาฬิกา
- ข. สัญญาณขอบขาขึ้น
- ค. สัญญาณขอบขาลง



รูปที่ 2.40 แสดงลักษณะของสัญญาณนาฬิกา

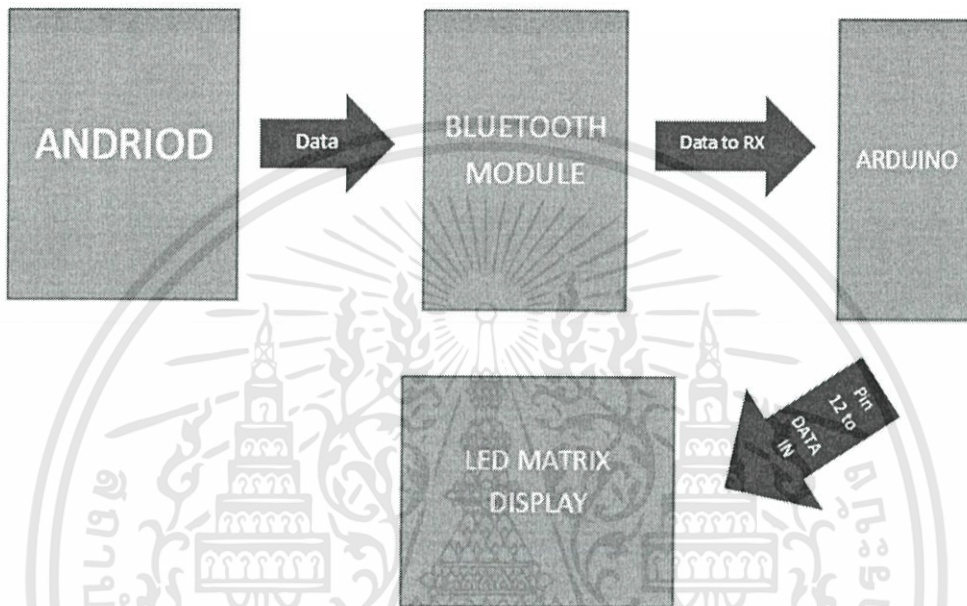
แสดงสัญญาณนาฬิกาเต็มลูก ซึ่งเป็นสัญญาณรูปสี่เหลี่ยม แต่การใช้สัญญาณนาฬิกากระตุ้นให้ฟลิปฟล็อปทำงานเปลี่ยนสถานะนั้นสามารถใช้ได้ใน 2 กรณีคือ

1. สัญญาณนาฬิกาขอบขาขึ้น (Leading Edge or Positive Edge) เป็นสัญญาณกระตุ้นให้ฟลิปฟล็อปทำงานตรงตำแหน่งขอบขาขึ้นของสัญญาณ คือเปลี่ยนจากลอจิก "0" ไปเป็นลอจิก "1" ลักษณะสัญญาณนาฬิกาขอบขาขึ้น แสดงดังรูปที่ ข.
2. สัญญาณนาฬิกาขอบขาลง (Trailing Edge or Negative Edge) เป็นสัญญาณกระตุ้นให้ฟลิปฟล็อปทำงานตรงตำแหน่งขอบขาลงของสัญญาณ คือเปลี่ยนจากลอจิก "1" ไปเป็นลอจิก "0" ลักษณะสัญญาณนาฬิกาขอบขาลง แสดงดังรูปที่ ค.

### บทที่ 3

## รายละเอียดในการออกแบบ และการสร้าง

### 3.1 บล็อกไดอแกรม(Block Diagram)



รูปที่ 3.1 บล็อกไดอแกรมของระบบ

#### การทำงาน

แอนดรอยด์จะทำการส่งข้อมูลไปยังบลูทูธโมดูลโดยผ่านการเชื่อมต่อแบบบลูทูธจากนั้นบลูทูธโมดูลจะทำการส่งข้อมูลไปยังอาดูโน่ผ่านทางขา RX โดยจะส่งข้อมูลไปให้อาดูโน่ประมวลผล จากนั้นอาดูโน่จะส่งข้อมูลไปยังแอลอีดีเมทริกส์โดยส่งผ่านทางขา 12 ของอาดูโน่ไปยังขาเข้าข้อมูลของแอลอีดีเมทริกส์ดีสเพลย์ ทำให้ไฟติดเป็นตัวอักษรที่กำหนด

## 3.2 ส่วนของภาคส่ง

3.1.1 มือถือแอนดรอยด์ รุ่น Samsung Note8 ใช้ บลูทูธ ในการเชื่อมต่อกับ บลูทูธโมดูล โดยใช้แอปพลิเคชันในการควบคุม

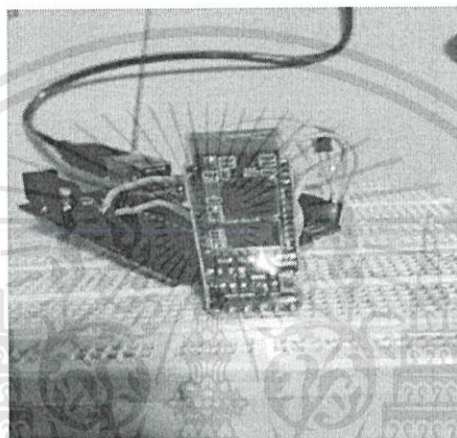


รูปที่ 3.2 แอปพลิเคชันในการป้อนคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3 ส่วนของภาครับ

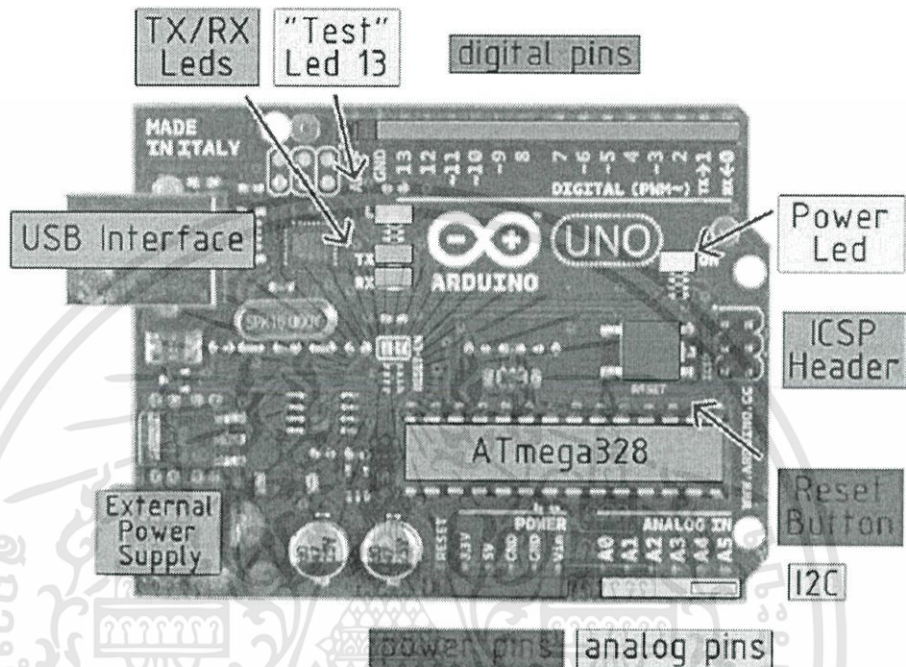
- 3.1.1 บลูทูธ Module รุ่น HC-05 ใช้ในการรับข้อมูลจากมือถือ Android เป็น slave mode
- 3.1.2 VCC: ขาไฟเลี้ยง
- 3.1.3 กราวด์: Ground
- 3.1.4 TXD: เป็นขาเอาต์พุตของโมดูล โดยจะไปเชื่อมกับขา RX ของไมโครคอนโทรลเลอร์
- 3.1.5 RXD: เป็นขาอินพุตของโมดูล โดยจะไปเชื่อมต่อกับขา TX ของไมโครคอนโทรลเลอร์



รูปที่ 3.3 การเชื่อมต่อบลูทูธโมดูล กับแอนดรอยด์

### 3.4 ส่วนของภาคคอนโทรลเลอร์

จะใช้บอร์ดอาร์ดูโน ในการควบคุมโดยใช้ไมโครคอนโทรลเลอร์ ATmega328 ในการควบคุม โดยการเขียนโปรแกรมผ่านโปรแกรมอาร์ดูโน ทำให้ แอลอีดีเมทริกซ์ ที่ต่อไว้กับเอาต์พุตติด



รูปที่ 3.4 แสดงขาต่างๆของบอร์ด อาร์ดูโน Uno R3

โค้ดทำงานอาดูโน

โค้ดของ อาร์ดูโน Uno R3 ตัวอักษร

```
{0x7E,0x81,0x81,0x81,0x7E,0x0}, //0
```

```
{0x4,0x82,0xFF,0x80,0x0,0x0}, //1
```

```
{0xE2,0x91,0x91,0x91,0x8E,0x0}, //2
```

```
{0x42,0x89,0x89,0x89,0x76,0x0}, //3
```

```
{0x1F,0x10,0x10,0xFF,0x10,0x0}, //4
```

```
{0x8F,0x89,0x89,0x89,0x71,0x0}, //5
```

```
{0x7E,0x89,0x89,0x89,0x71,0x0}, //6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

{0x1,0x1,0xF9,0x5,0x3,0x0},//7  
 {0x76,0x89,0x89,0x89,0x76,0x0},//8  
 {0x8E,0x91,0x91,0x91,0x7E,0x0},//9  
 {0x0,0x0,0x0,0x0,0x0,0x0},// blank space  
 {0x0,0x0,0x90,0x0,0x0,0x0}, //:  
 {0x0,0x10,0x10,0x10,0x10,0x0},// -  
 {0x0,0x0,0x80,0x0,0x0,0x0},// .  
 {0xFC,0x9,0x11,0x21,0xFC,0x0},//Ñ  
 {0xFE,0x11,0x11,0x11,0xFE,0x0},//A  
 {0xFF,0x89,0x89,0x89,0x76,0x0},//B  
 {0x7E,0x81,0x81,0x81,0x42,0x0},//C  
 {0xFF,0x81,0x81,0x81,0x7E,0x0},//D  
 {0xFF,0x89,0x89,0x89,0x81,0x0},//E  
 {0xFF,0x9,0x9,0x9,0x1,0x0},//F  
 {0x7E,0x81,0x81,0x91,0x72,0x0},//G  
 {0xFF,0x8,0x8,0x8,0xFF,0x0},//H  
 {0x0,0x81,0xFF,0x81,0x0,0x0},//I  
 {0x60,0x80,0x80,0x80,0x7F,0x0},//J  
 {0xFF,0x18,0x24,0x42,0x81,0x0},//K  
 {0xFF,0x80,0x80,0x80,0x80,0x0},//L  
 {0xFF,0x2,0x4,0x2,0xFF,0x0},//M  
 {0xFF,0x2,0x4,0x8,0xFF,0x0},//N  
 {0x7E,0x81,0x81,0x81,0x7E,0x0},//O

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{0xFF,0x11,0x11,0x11,0xE,0x0},//P
{0x7E,0x81,0x81,0xA1,0x7E,0x80},//Q
{0xFF,0x11,0x31,0x51,0x8E,0x0},//R
{0x46,0x89,0x89,0x89,0x72,0x0},//S
{0x1,0x1,0xFF,0x1,0x1,0x0},//T
{0x7F,0x80,0x80,0x80,0x7F,0x0},//U
{0x3F,0x40,0x80,0x40,0x3F,0x0},//V
{0x7F,0x80,0x60,0x80,0x7F,0x0},//W
{0xE3,0x14,0x8,0x14,0xE3,0x0},//X
{0x3,0x4,0xF8,0x4,0x3,0x0},//Y
{0xE1,0x91,0x89,0x85,0x83,0x0},//Z

```

โค้ดที่ใช้งานในกาทดลอง

```

#include "LEDControlMS.h"

pin 12 is connected to the dataIn
pin 11 is connected to the CLK
pin 10 is connected to load

We have only a single MAX72XX.

#define NBR_MTX 4

LEDControl lc=LEDControl(12,11,10, NBR_MTX);

String scrollString;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

int stringLength;

char ch0, ch1, ch2, ch3;

int nextCharIndex=0;

void setup() {

  /*

  The MAX72XX is in power-saving mode on startup,
  we have to do a wakeup call

  */

  Serial.begin (9600);
  Serial.println("Setup");
  Serial.println(scrollString);
  Serial.println(stringLength);
  for (int i=0; i< NBR_MTX; i++){
    lc.shutdown(i,false);
    /* Set the brightness to a low value */
    lc.setIntensity(i,1);
    /* and clear the display */
    lc.clearDisplay(i);
  }

  delay(100);

  lc.clearAll();

  ch0= scrollString[0];

  ch1= scrollString[1];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ch2= scrollString[2];

ch3= scrollString[3];

nextCharIndex=4;

}

void loop()

{

  if(Serial.available() > 0)

  { scrollString = Serial.readStringUntil('\n');

    stringLength = Serial.parseInt();

    stringLength = scrollString.length();

  }

  Serial.println(scrollString);

  Serial.println(stringLength);

  delay(500);

  lc.displayChar(0, lc.getCharArrayPosition(ch0));

  lc.displayChar(1, lc.getCharArrayPosition(ch1));

  lc.displayChar(2, lc.getCharArrayPosition(ch2));

  lc.displayChar(3, lc.getCharArrayPosition(ch3));

  ch0=ch1;

  ch1=ch2;

  ch2=ch3;

  ch3=scrollString[nextCharIndex++];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if (nextCharIndex >= stringLength) nextCharIndex = 0;

delay(600); \

lc.clearAll();

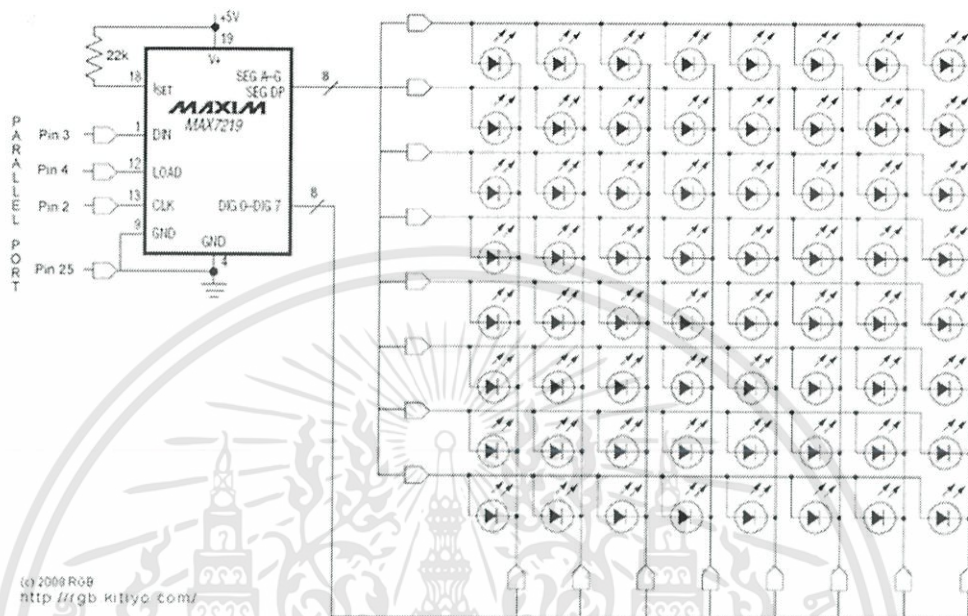
delay(25);

}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5 ส่วนของภาคแสดงผล



รูปที่ 3.5 ส่วนของภาคแสดงผลแอลอีดีเมทริกซ์(แอลอีดี MATRIX) และวงจรชิพรีจิสเตอร์(SHIFT REGISTER) MAX7219

โค้ดการเซต low แล้ว คอลัมน์

```
void แอลอีดีControl::setRow(int addr, int row, byte value) {
```

```
    int offset;
```

```
    if(addr<0 || addr>=maxDevices)
```

```
        return;
```

```
    if(row<0 || row>7)
```

```
        return;
```

```
    offset=addr*8;
```

```
    status[offset+row]=value;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

spiTransfer(addr, row+1,status[offset+row]);
}

void LEDControl::setColumn(int addr, int col, byte value) {

byte val;

if(addr<0 || addr>=maxDevices)

return;

if(col<0 || col>7)

return;

for(int row=0;row<8;row++) {

val=value >> (7-row);

val=val & 0x01;

setLED(addr,row,col,val);

}

}

```

เราจะทำการเปิดแถว ค้างไว้ และเปิด คอลัมน์ ทีละแถว ทำให้ไฟติดด้วยการส่งสัญญาณไปยังขาที่กำหนด จากนั้นจึงเปิดแถวต่อไปค้างไว้ ทำอย่างนี้ไปเรื่อยๆจนครบ โดยเทคนิคนี้เรียกว่า Persistence of vision โดยมีชิพรีจิสเตอร์เป็นตัวช่วยในการทำงานโดยการส่งข้อมูลแบบเอสพีไอ

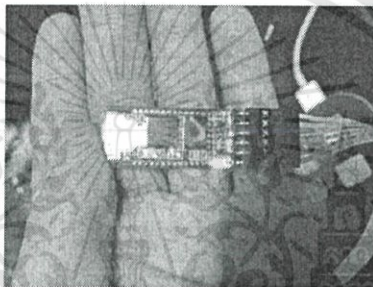
## บทที่ 4

### การทดลองและผลการทดลอง

#### 4.1 การทดลอง

##### 4.1.1 ทำการเชื่อมต่อกับบลูทูธโมดูล

เมื่อทำการต่อบลูทูธเสร็จจะเห็นว่าไฟที่บลูทูธจะกระพริบเป็นจังหวะดังรูปที่ 4.1



รูปที่ 4.1 ไฟกระพริบที่บลูทูธ

##### 4.1.2 พิมพ์ตัวอักษรเป็นภาษาอังกฤษที่ต้องการแสดงผล แล้วกดส่ง

จากโปรแกรมที่ใช้จะเห็นหน้าจอเป็นดังรูปที่ 4.2



รูปที่ 4.2 หน้าจอโปรแกรม BTInterface Free Trial

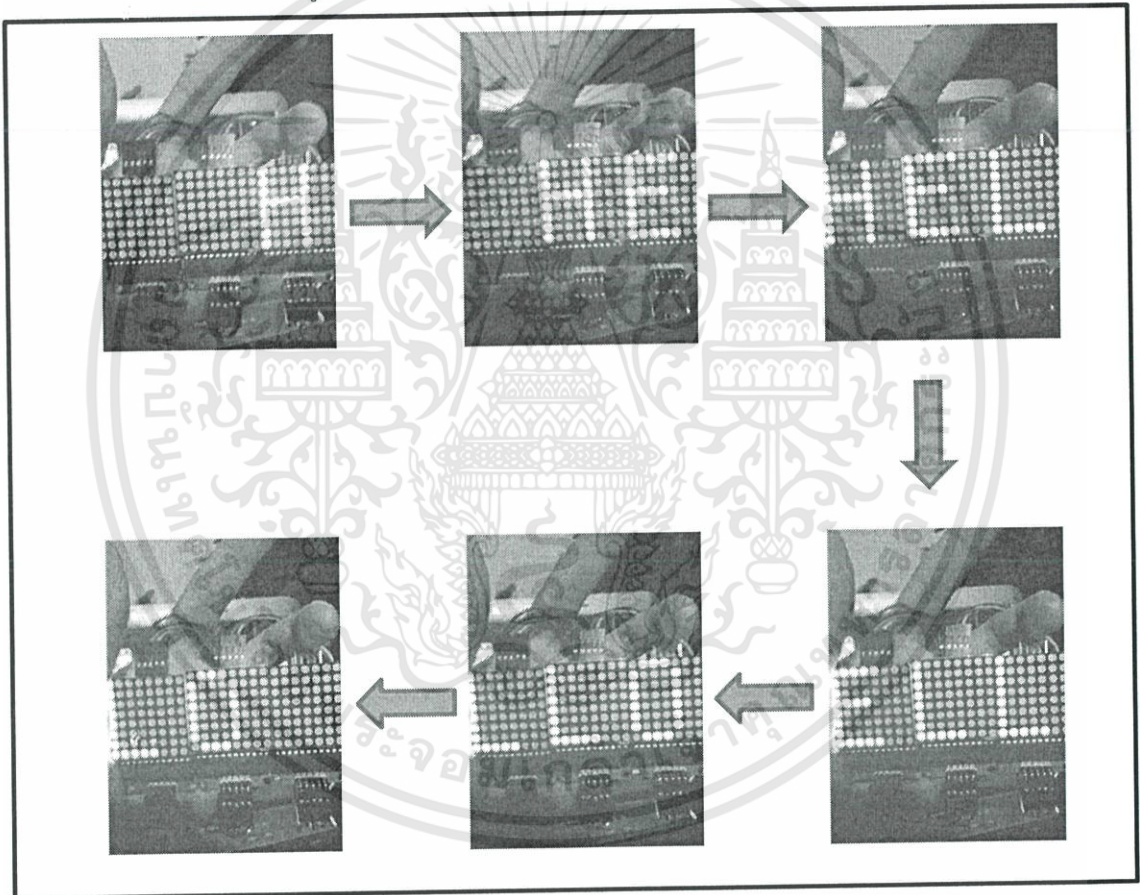
##### 4.1.3 แอลอีดีเมทริกซ์ แสดงผลตามตัวอักษรที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.2 สรุปผลการทดลอง

เมื่อพิมพ์คำว่า HELLO บนโปรแกรมดังรูปที่ 4.3 แล้วกดส่ง จะเห็นว่าตัวอักษรที่พิมพ์เข้าไป จะวิ่งจากซ้ายไปขวาดังรูปที่ 4.4

รูปที่ 4.3 หน้าจอโปรแกรม BTInterface Free Trial



รูปที่ 4.3 การแสดงผลของ แอลอีดี Matrix Display

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### บทสรุปและวิจารณ์

#### 5.1 สรุปผลการออกแบบวงจร

จากการออกแบบวงจรพบว่าเป็นไปตามที่คาดหวังไว้ โดยการทำงานจะทำการเชื่อมต่อแอนตรอยด์กับบลูทูธโมดูลเพื่อทำการส่งข้อมูลไปยังอาดูโน่ เพื่อที่จะประมวลผลและนำข้อมูลนำออกมาแสดงผ่านแอลอีดีเมทริกส์

#### 5.2 สรุปผลการทดลอง

เป็นไปตามที่คาดหวัง แอลอีดีเมทริกส์สามารถที่จะแสดงคำว่า HELLO ออกมาได้โดยการส่งผ่านบลูทูธโมดูล โดยสั่งจากแอนตรอยด์

#### 5.3 วิจารณ์ผลการทดลอง

ผลการทดลองมีความคลาดเคลื่อนตรงการแสดงผลแอลอีดีเมทริกส์ที่มีหน่วยงานทำงานและต้องพิมพ์ เว้นวรรค 2 ครั้งเพื่อแสดงข้อมูลที่ถูกต้องก่อนแสดงผล

#### 5.4 ประโยชน์ที่ได้รับ

จากการทำงานประโยชน์ที่ได้รับคือความเข้าใจในการทำงานของอาดูโน่และแอลอีดีเมทริกส์ และเทคนิคการขยายพอร์ตการทำงานของไมโครคอนโทรเลอร์ซึ่งมีจำนวนเอาพุทที่น้อย ซึ่งทำให้เข้าใจการทำงานของระบบเมทริกส์

## บรรณานุกรม

ส่วนของ Bluetooth และ HC-05

<http://th.wikipedia.org/wiki/%E0%B8%9A%E0%B8%A5%E0%B8%B9%E0%B8%97%E0%B8%B9%E0%B8%98>

<http://pirun.kps.ku.ac.th/~b4928024/Bluetooth4.html>

[www.synes.co.th/nicupload/20130121103905.pdf](http://www.synes.co.th/nicupload/20130121103905.pdf)

ส่วนของ Android

[http://www.techmoblog.com/android\\_phone\\_guide/](http://www.techmoblog.com/android_phone_guide/)

[http://th.wikipedia.org/wiki/%E0%B9%81%E0%B8%AD%E0%B8%99%E0%B8%94%E0%B8%A3%E0%B8%AD%E0%B8%A2%E0%B8%94%E0%B9%8C\\_\(%E0%B8%A3%E0%B8%B0%E0%B8%9A%E0%B8%9A%E0%B8%9B%E0%B8%8F%E0%B8%B4%E0%B8%9A%E0%B8%B1%E0%B8%95%E0%B8%B4%E0%B8%81%E0%B8%B2%E0%B8%A3\)](http://th.wikipedia.org/wiki/%E0%B9%81%E0%B8%AD%E0%B8%99%E0%B8%94%E0%B8%A3%E0%B8%AD%E0%B8%A2%E0%B8%94%E0%B9%8C_(%E0%B8%A3%E0%B8%B0%E0%B8%9A%E0%B8%9A%E0%B8%9B%E0%B8%8F%E0%B8%B4%E0%B8%9A%E0%B8%B1%E0%B8%95%E0%B8%B4%E0%B8%81%E0%B8%B2%E0%B8%A3))

ส่วนแอปพลิเคชัน

Googleplay –Andriod Bluetooth Teminal

คู่มือพัฒนาแอปพลิเคชัน Android อย่างมืออาชีพ (ดร.จักรชัย โสอินทร์ ,พงษ์ศธร จันทรียอย ,ณัฐณี ษา วีระมงคลเลิศ)

ส่วนของ Arduino

[www.ee.buu.ac.th/~acitl/project/2011/Android\\_I\\_O.../lesson%202.pdf](http://www.ee.buu.ac.th/~acitl/project/2011/Android_I_O.../lesson%202.pdf)

<http://www.arduino.cc/>

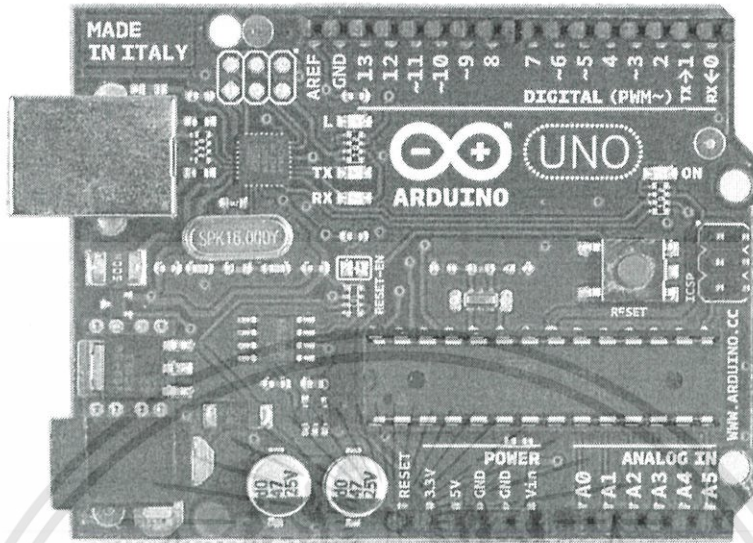
เริ่มต้นการเขียนโปรแกรมด้วยภาษา C++ (ยุทธนา สีสลาวัฒน์กุล)

ภาคผนวก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Arduino UNO



## Product Overview

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the [index of Arduino boards](#).

## Index

Technical Specifications

Page 2

How to use Arduino  
Programming Enviroment, Basic Tutorials

Page 6

Terms & Conditions

Page 7

Enviromental Policies  
half sqm of green via Impatto Zero®

Page 7



radiospares

RADIONICS



เราเป็นเอกฉันท์กันไว้สำหรับบริการเพื่อการศึกษาเท่านั้น เมื่อสั่งซื้อให้หน้าจั่วชื่อโรงเรียนหรือหน่วยงาน  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Technical Specification

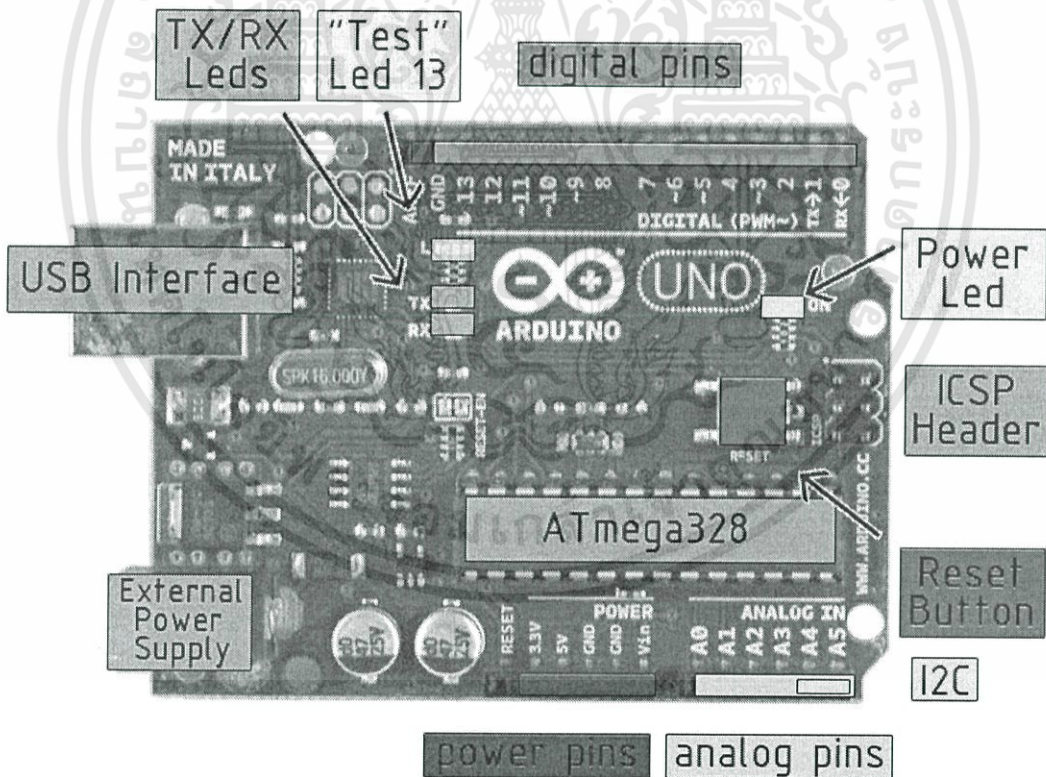


EAGLE files: [arduino-duemilanove-uno-design.zip](#) Schematic: [arduino-uno-schematic.pdf](#)

## Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB of which 0.5 KB used by bootloader
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

## the board



**radiospares**

**RADIONICS**

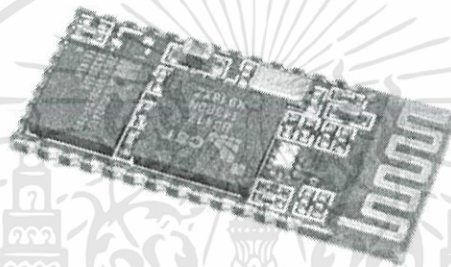


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# HC-05

## -Bluetooth to Serial Port Module

### Overview



HC-05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup.

Serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Bluecore 04-External single chip Bluetooth system with CMOS technology and with AFH(Adaptive Frequency Hopping Feature). It has the footprint as small as 12.7mmx27mm. Hope it will simplify your overall design/development cycle.

### Specifications

#### Hardware features

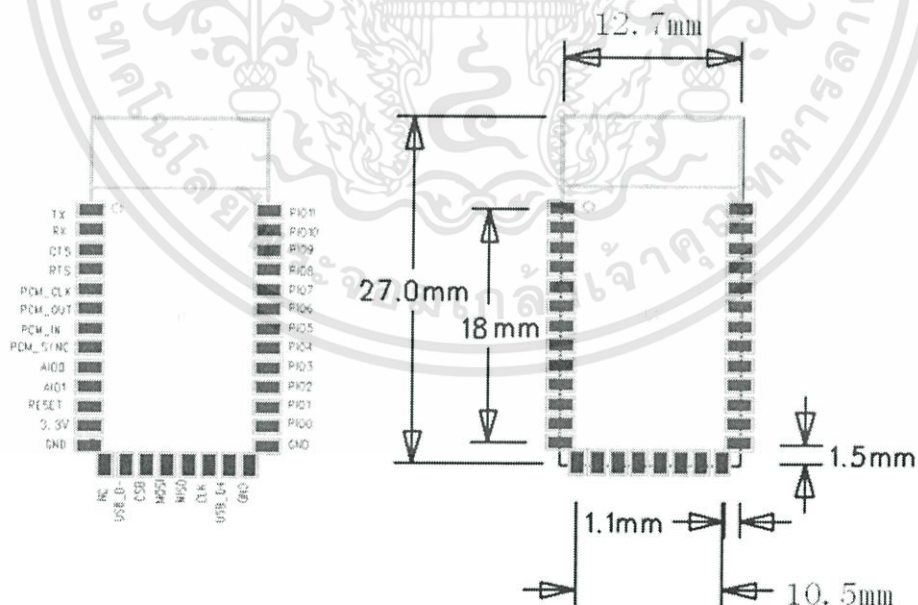
- Typical -80dBm sensitivity
- Up to +4dBm RF transmit power
- Low Power 1.8V Operation ,1.8 to 3.6V I/O
- PIO control
- UART interface with programmable baud rate
- With integrated antenna
- With edge connector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Software features

- Default Baud rate: 38400, Data bits:8, Stop bit:1,Parity:No parity, Data control: has. Supported baud rate: 9600,19200,38400,57600,115200,230400,460800.
- Given a rising pulse in PIO0, device will be disconnected.
- Status instruction port PIO1: low-disconnected, high-connected;
- PIO10 and PIO11 can be connected to red and blue led separately. When master and slave are paired, red and blue led blinks 1time/2s in interval, while disconnected only blue led blinks 2times/s.
- Auto-connect to the last device on power as default.
- Permit pairing device to connect as default.
- Auto-pairing PINCODE:"0000" as default
- Auto-reconnect in 30 min when disconnected as a result of beyond the range of connection.

## Hardware



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



# Serially Interfaced, 8-Digit LED Display Drivers

MAX7219/MAX7221

## General Description

The MAX7219/MAX7221 are compact, serial input/output common-cathode display drivers that interface microprocessors ( $\mu$ Ps) to 7-segment numeric LED displays of up to 8 digits, bar-graph displays, or 64 individual LEDs. Included on-chip are a BCD code-B decoder, multiplex scan circuitry, segment and digit drivers, and an 8x8 static RAM that stores each digit. Only one external resistor is required to set the segment current for all LEDs. The MAX7221 is compatible with SPI™, QSPI™, and Microwire™, and has slew-rate-limited segment drivers to reduce EMI.

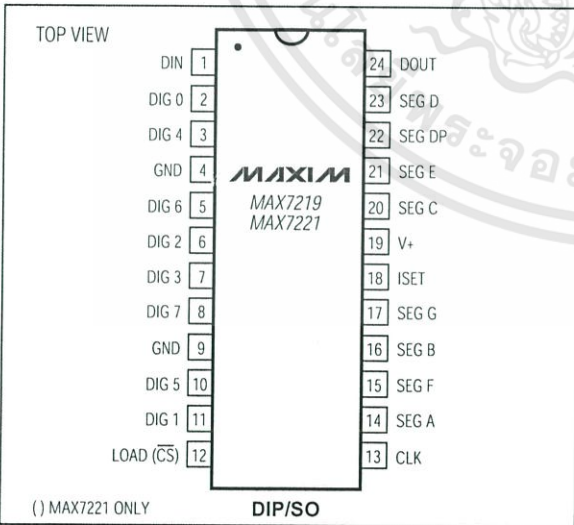
A convenient 3-wire serial interface connects to all common  $\mu$ Ps. Individual digits may be addressed and updated without rewriting the entire display. The MAX7219/MAX7221 also allow the user to select code-B decoding or no-decode for each digit.

The devices include a 150 $\mu$ A low-power shutdown mode, analog and digital brightness control, a scan-limit register that allows the user to display from 1 to 8 digits, and a test mode that forces all LEDs on.

## Applications

- Bar-Graph Displays
- 7-Segment Displays
- Industrial Controllers
- Panel Meters
- LED Matrix Displays

## Pin Configuration



## Features

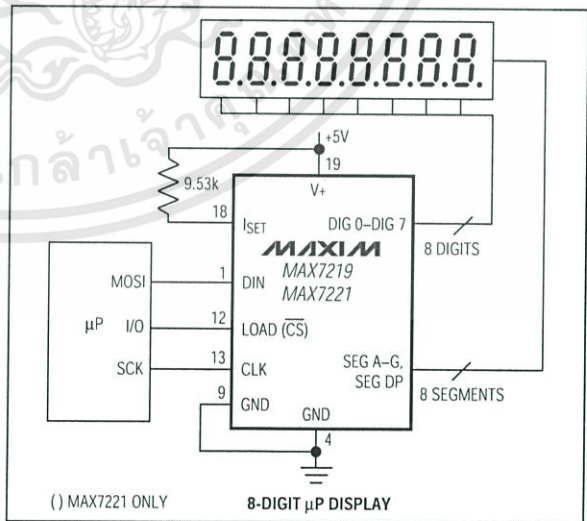
- ◆ 10MHz Serial Interface
- ◆ Individual LED Segment Control
- ◆ Decode/No-Decode Digit Selection
- ◆ 150 $\mu$ A Low-Power Shutdown (Data Retained)
- ◆ Digital and Analog Brightness Control
- ◆ Display Blanked on Power-Up
- ◆ Drive Common-Cathode LED Display
- ◆ Slew-Rate Limited Segment Drivers for Lower EMI (MAX7221)
- ◆ SPI, QSPI, Microwire Serial Interface (MAX7221)
- ◆ 24-Pin DIP and SO Packages

## Ordering Information

PART	TEMP. RANGE	PIN-PACKAGE
MAX7219CNG	0°C to +70°C	24 Narrow Plastic DIP
MAX7219CWG	0°C to +70°C	24 Wide SO
MAX7219C/D	0°C to +70°C	Dice*
MAX7219ENG	-40°C to +85°C	24 Narrow Plastic DIP
MAX7219EWG	-40°C to +85°C	24 Wide SO
MAX7219ERG	-40°C to +85°C	24 Narrow CERDIP

Ordering Information continued at end of data sheet.  
\*Dice are specified at  $T_A = +25^\circ\text{C}$ .

## Typical Application Circuit



SPI and QSPI are trademarks of Motorola Inc. Microwire is a trademark of National Semiconductor Corp.



For free samples & the latest literature: <http://www.maxim-ic.com>, or phone 1-800-998-8800.  
For small orders, phone 408-737-7600 ext. 3468.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Serially Interfaced, 8-Digit LED Display Drivers

## ABSOLUTE MAXIMUM RATINGS

Voltage (with respect to GND)

V+	-0.3V to 6V
DIN, CLK, LOAD, CS	-0.3V to 6V
All Other Pins	-0.3V to (V+ + 0.3V)

Current

DIG0-DIG7 Sink Current	500mA
SEGA-G, DP Source Current	100mA

Continuous Power Dissipation (T<sub>A</sub> = +85°C)

Narrow Plastic DIP	0.87W
Wide SO	0.76W
Narrow CERDIP	1.1W

Operating Temperature Ranges

MAX7219C_G/MAX7221C_G	0°C to +70°C
MAX7219E_G/MAX7221E_G	-40°C to +85°C
Storage Temperature Range	-65°C to +160°C
Lead Temperature (soldering, 10sec)	+300°C

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

## ELECTRICAL CHARACTERISTICS

(V+ = 5V ±10%, R<sub>SET</sub> = 9.53kΩ ±1%, T<sub>A</sub> = T<sub>MIN</sub> to T<sub>MAX</sub>, unless otherwise noted.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Operating Supply Voltage	V+		4.0		5.5	V
Shutdown Supply Current	I+	All digital inputs at V+ or GND, T <sub>A</sub> = +25°C			150	μA
Operating Supply Current	I+	R <sub>SET</sub> = open circuit			8	mA
		All segments and decimal point on, I <sub>SEG-</sub> = -40mA		330		
Display Scan Rate	f <sub>OSC</sub>	8 digits scanned	500	800	1300	Hz
Digit Drive Sink Current	I <sub>DIGIT</sub>	V+ = 5V, V <sub>OUT</sub> = 0.65V	320			mA
Segment Drive Source Current	I <sub>SEG</sub>	T <sub>A</sub> = +25°C, V+ = 5V, V <sub>OUT</sub> = (V+ - 1V)	-30	-40	-45	mA
Segment Current Slew Rate (MAX7221 only)	ΔI <sub>SEG</sub> /Δt	T <sub>A</sub> = +25°C, V+ = 5V, V <sub>OUT</sub> = (V+ - 1V)	10	20	50	mA/μs
Segment Drive Current Matching	ΔI <sub>SEG</sub>			3.0		%
Digit Drive Leakage (MAX7221 only)	I <sub>DIGIT</sub>	Digit off, V <sub>DIGIT</sub> = V+			-10	μA
Segment Drive Leakage (MAX7221 only)	I <sub>SEG</sub>	Segment off, V <sub>SEG</sub> = 0V			1	μA
Digit Drive Source Current (MAX7219 only)	I <sub>DIGIT</sub>	Digit off, V <sub>DIGIT</sub> = (V+ - 0.3V)	-2			mA
Segment Drive Sink Current (MAX7219 only)	I <sub>SEG</sub>	Segment off, V <sub>SEG</sub> = 0.3V	5			mA