

ระบบแสดงสถานะการใช้ทองของบุคคล

INDIVIDUAL STATUS SYSTEM

ธณภณ เวียงนาชัย

THANAPOL WIANGNACHAI

ปริญญาโทนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมสารสนเทศ
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2557

ระบบแสดงสถานะการใช้ห้องของบุคคล

INDIVIDUAL STATUS SYSTEM

ธณภณ เวียงนาชัย

THANAPOL WIANGNACHAI

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2557

INDIVIDUAL STATUS SYSTEM

THANAPOL WIANGNACHAI

THIS THESIS IS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INFORMATION ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2014

คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองปริญญาานิพนธ์

หัวข้อปริญญาานิพนธ์	ระบบแสดงสถานะการใช้ห้องของบุคคล
Thesis Title	INDIVIDUAL STATUS SYSTEM
ชื่อนักศึกษา	นายธณภณ เวียงนาชัย
ระดับปริญญา	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมสารสนเทศ
ปริญญาานิพนธ์ปีการศึกษา	2557

(.....)

ผศ.กฤตากร กล่อมการ
อาจารย์ที่ปรึกษาปริญญาานิพนธ์

(.....)

ผศ.ไพศาล สิทธิโยภาสกุล
อาจารย์ที่ปรึกษาปริญญาานิพนธ์ร่วม

หัวข้อปริญญานิพนธ์	ระบบแสดงสถานะการใช้ห้องของบุคคล	
Thesis Title	INDIVIDUAL STATUS SYSTEM	
ชื่อนักศึกษา	นายธณภณ เวียงนาชัย	รหัสนักศึกษา 54010550
ระดับปริญญา	วิศวกรรมศาสตรบัณฑิต	
สาขาวิชา	วิศวกรรมสารสนเทศ	
ปีการศึกษา	2557	
อาจารย์ที่ปรึกษาปริญญานิพนธ์	อาจารย์สรพงษ์ วชิรรัตนพรกุล	
อาจารย์ที่ปรึกษาปริญญานิพนธ์ร่วม	ผศ.ไพศาล สิทธิโยภาสกุล	

บทคัดย่อ

โครงการนี้แสดงถึงการสร้างระบบตรวจสอบสถานะการใช้ห้องของบุคคลและทำการแสดงผลผ่านทางจอแสดงผล อุปกรณ์ที่สร้างขึ้นมีส่วนประกอบสำคัญคือไมโครคอนโทรลเลอร์ตัวส่งสัญญาณไร้สายและตัวอ่านสัญญาณวิทยุหลักการทำงานง่ายๆคือเมื่อบุคคลเข้ามาอยู่ภายในห้องก็นำตัวเก็บสัญญาณวิทยุแต่ไปยังตัวรับสัญญาณวิทยุหลังจากนั้นตัวรับสัญญาณวิทยุจะทำการอ่านค่าที่ได้แล้วส่งไปยังไมโครคอนโทรลเลอร์จากนั้นทำการประมวลผลและส่งสัญญาณไร้สายไปยังฝั่งตัวรับ เมื่อรับข้อมูลมาแล้วจะส่งข้อมูลที่ไปประมวลผลที่ไมโครคอนโทรลเลอร์จากนั้นก็ส่งข้อมูลที่ไปแสดงผลที่จอแสดงผล

Thesis Title	INDIVIDUAL STATUS SYSTEM
Student	Mr.Thanapol wiangnachai Student ID. 54010550
Degree	Bachelor of Engineering
Program	Information Engineering
Academic Year	2014
Thesis Advisor	Mr.Sorapong Wachirattanakul
Thesis Co-Advisor	Asst.Prof. Paisan Sithiyopasakul

ABSTRACT

This project represents the creation of monitoring the use of the party room, then displayed via the monitor. The device is built with integral microcontroller and a wireless transmitter radio read a simple working principle is that when a person enters the room was taken to collect radio signals to a receiver after a radio receiver. Read the radio will be then sent to the microcontroller then processes and transmits a wireless signal to the receiver. When receiving data, then sends the data to be processed in the microcontroller then sends data to the display monitor.

กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้สำเร็จได้อย่างดีด้วยความอนุเคราะห์ช่วยเหลือจาก อาจารย์สรพงษ์ วชิรรัตนพรกุล และ ผศ.ไพศาล สิทธิโยภาสกุล ที่ช่วยให้คำชี้แนะและแนวคิดในการสร้างผลงานขึ้นมา
ขอบคุณพ่อแม่ที่เลี้ยงดูและมอบกำลังใจให้กับพวกเราจนทำให้พวกเราสามารถมีวันนี้ได้
ขอบคุณพี่ๆ มหาบัณฑิตที่คอยช่วยเหลือดูแล ให้คำแนะนำ และให้กำลังใจตลอดมา
ขอบคุณเพื่อนๆ ที่คอยชี้แนะให้แนวคิดให้กำลังใจ และอยู่ด้วยกันมาตลอดขอบคุณในมิตรภาพ
ที่มอบให้

ขอบคุณผู้มีพระคุณทุกๆ คนที่อาจจะไม่ได้กล่าวถึง ณ ที่แห่งนี้
สิ่งที่มีประโยชน์สิ่งที่เกิดคุณค่าอันเกิดจากปริญญาานิพนธ์นี้คณะผู้จัดทำขอมอบให้แก่ผู้มี
พระคุณทุกท่าน

ธณภณ เวียงนาชัย

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	V
สารบัญตาราง.....	VI
สารบัญรูป.....	VII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญ.....	1
1.2 จุดประสงค์.....	1
1.3 ขอบเขตของโครงการ.....	1
1.4 ผลที่คาดว่าจะได้รับ.....	1
1.5 อุปกรณ์ที่ต้องใช้.....	1
1.5.1 ฮาร์ดแวร์.....	1
1.5.2 ซอฟต์แวร์.....	2
บทที่ 2 ทฤษฎีพื้นฐานที่ใช้.....	3
2.1 ทฤษฎีเกี่ยวกับไมโครคอนโทรลเลอร์ Arduino Nano.....	3
2.1.1 ที่มา Arduino.....	3
2.1.2 จุดเด่นที่ทำให้บอร์ด Arduino เป็นที่นิยม.....	7
2.1.3 รูปแบบการเขียนโปรแกรมบน Arduino.....	8
2.2 ทฤษฎีเกี่ยวกับ RFID (Radio frequency identification).....	11

สารบัญ (ต่อ)

	หน้า
2.3 ทฤษฎีเกี่ยวกับ Monitor (AMOLED, LCD,TFT, LED).....	14
2.3.1 AMOLED.....	16
2.3.2 LCD.....	17
2.3.3 LED.....	20
2.3.4 TFT (thin-film transistor).....	20
2.4 ทฤษฎีเกี่ยวกับ NRF24L01 Module โมดูลสื่อสารไร้สาย.....	21
บทที่ 3 การวิเคราะห์ และออกแบบระบบ.....	23
3.1 Block Diagram.....	23
3.2 การออกแบบฮาร์ดแวร์ (Hardware).....	23
3.3 การต่ออุปกรณ์โมดูลต่างๆเข้าด้วยกัน.....	24
3.4 การออกแบบโปรแกรม.....	26
บทที่ 4 การวิเคราะห์ และออกแบบระบบ.....	29
4.1 การทดลองวงในส่วนของภาครับและภาคส่ง.....	29
4.2 การแสดงผลของชิ้นงาน.....	32
บทที่ 5 การวิเคราะห์ และออกแบบระบบ.....	34
5.1 บทสรุปโครงการ.....	34
5.2 ปัญหาที่พบในระหว่างดำเนินงาน.....	34
5.3 แนวทางแก้ไขและพัฒนาต่อ.....	34
บรรณานุกรม.....	35
ภาคผนวก.....	36

สารบัญตาราง

	หน้า
ตารางที่ 2.1 แสดงวัสดุและช่วงความถี่.....	12
ตารางที่ 2.1 แสดงวัสดุและช่วงความถี่ (ต่อ).....	13
ตารางที่ 2.1 แสดงวัสดุและช่วงความถี่ (ต่อ).....	13
ตารางที่ 4.1 ตารางแสดงระยะทางการส่งสัญญาณของตัวส่งสัญญาณไร้สาย.....	30

สารบัญรูป

	หน้า
รูปที่ 2.1 เครื่องมือวัดและแสดงอุณหภูมิ.....	4
รูปที่ 2.2 หุ่นยนต์.....	5
รูปที่ 2.3 Drone	5
รูปที่ 2.4 หน้าจอแสดงผล.....	6
รูปที่ 2.5 บอร์ด Arduino ต่อกับ LED.....	7
รูปที่ 2.6 บอร์ด Arduino ต่อกับ Xbee	7
รูปที่ 2.7 การเขียนโปรแกรมบน Arduino	8
รูปที่ 2.8 เลือกหมายเลข Com พอร์ต ของบอร์ด	8
รูปที่ 2.9 กดปุ่ม Verify เพื่อตรวจสอบความถูกต้อง.....	9
รูปที่ 2.10 Upload โค้ดโปรแกรม.....	9
รูปที่ 2.11 บอร์ด Arduino UNO R3.....	10
รูปที่ 2.12 หน้าจอ smart phone.....	14
รูปที่ 2.13 แผนภาพแสดงจอ plasma ที่ขยายให้ดูรายละเอียดภายใน (เซลล์เดียว).....	15
รูปที่ 2.14 แสดงเทคโนโลยี OLED ที่ใช้โพลีเมอร์อินทรีย์เรืองแสงในการผลิตแสงสี.....	16
รูปที่ 2.15 LCD: Twisted Nematic (TN หรือ TN-Film).....	17
รูปที่ 2.16 LCD: In-Plane Switching (IPS).....	18
รูปที่ 2.17 LCD: Vertically Aligned (VA)	19
รูปที่ 2.18 การจัดเรียง subpixel แบบทั่วไป.....	20
รูปที่ 2.19 การจัดเรียง subpixel แบบ PenTile	21
รูปที่ 2.20 NRF24L01 Module โมดูลสื่อสารไร้สาย.....	22
รูปที่ 2.20 interface size.....	22
รูปที่ 3.1 block diagram.....	23
รูปที่ 3.2 ตัวอย่างการส่งสัญญาณของโมดูล อาร์เอฟไอดี.....	24
รูปที่ 3.3 วงจรตัวส่งที่ออกแบบไว้.....	24
รูปที่ 3.4 วงจรตัวรับที่ออกแบบไว้ 2.....	25
รูปที่ 3.5 และเมื่อกดเข้าไปจะปรากฏโค้ด (Code) ภาษา C ดังรูปที่ 3.3.....	26
รูปที่ 3.6 โค้ดภาษา C ภายในโปรแกรมตัวอย่าง.....	27
รูปที่ 3.7 การลงโปรแกรมใส่ Arduino UNO	27
รูปที่ 3.8 การเปิดหน้าต่างเพื่อดูค่าอนาล็อกที่ส่งมาจากเซนเซอร์.....	28

สารบัญรูป(ต่อ)

	หน้า
รูปที่ 4.1 วงจรในส่วนของภาคส่ง.....	29
รูปที่ 4.2 ชิ้นส่วนภาคส่ง.....	29
รูปที่ 4.3 วงจรในส่วนของภาครับ.....	30
รูปที่ 4.4 ชิ้นส่วนของภาครับ.....	30
รูปที่ 4.5 ชิ้นงานทั้งสองส่วน.....	32
รูปที่ 4.6 แสดงผลเมื่อได้รับค่าแท้ก.....	32
รูปที่ 4.7 แสดงผลเมื่อได้รับค่าแท้ก.....	33
รูปที่ 4.8 แสดงผลเมื่อได้รับค่าแท้ก.....	33
รูปที่ 4.9 แสดงผลเมื่อค่าแท้กไม่มีข้อมูล.....	33
รูปที่ 4.10 แสดงสถานะ Login.....	34
รูปที่ 4.11 แสดงสถานะ Logout.....	34

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

ในสังคมปัจจุบันเวลาถือเป็นสิ่งที่สำคัญและเวลาที่มีความต้องการจะเข้าติดต่อกับบุคคล อาจจะไม่สามารถติดต่อสื่อสารกับบุคคลที่ต้องการจะมาพบได้ ทำให้ไม่สามารถรู้ได้ว่าบุคคลที่ต้องการจะพบอยู่ ณ ที่ทำงานหรือไม่ ทำให้ผู้จัดตระหนักถึงปัญหาและได้เล็งเห็นถึงความสำคัญตรงจุดนี้เพื่อสร้าง ระบบตรวจสอบสถานะบุคคล เพื่อตรวจสอบและแสดงสถานะของบุคคลและแสดงผลผ่านทางหน้าจอ

โมดูลอาร์เอฟไอดี (RFID) จะรับค่ามาจากตัวแท็ก (Tag) จากนั้นจะส่งค่าที่ได้รับไปประมวลที่ ไมโครคอนโทรลเลอร์ (Microcontroller) แล้วนำค่าที่ได้ส่งต่อไปยังตัวส่งสัญญาณไร้สาย (Wireless) ไปยังภาคตัว เมื่อได้รับค่าแล้วจะส่งไปยังไมโครคอนโทรลเลอร์ เพื่อประมวลผลจากนั้นจะส่งผลที่ได้ไป ยังจอแสดงผล (Display)

1.2 จุดประสงค์

ในการทำโครงการนี้มีจุดประสงค์ในการทำงานโดยสามารถระบุได้ดังนี้

1. เพื่อออกแบบอุปกรณ์ในใช้แสดงการเข้าออกสถานที่หรือห้อง อีกทั้งยังแสดงถึงสถานะว่าเจ้าของสถานที่หรือห้องนั้น กำลังอยู่ในห้องหรือสถานที่แห่งนั้นหรือไม่
2. เพื่อความสะดวกรวดเร็วในการติดต่อกับบุคคล

1.3 ขอบเขตของโครงการ

1. สามารถแสดงสถานะการใช้ห้องหรือสถานที่ผ่านหน้าจอได้
2. สามารถส่งสัญญาณจากอาร์เอฟไอดี จากฝั่งตัวรับไปยังฝั่งจอแสดงผลได้

1.4 ผลที่คาดว่าจะได้รับ

1. ระบบที่พัฒนาขึ้นสามารถแสดงผลผ่านจอแสดงผลได้
2. ระบบที่พัฒนาขึ้นสามารถช่วยลดการสูญเสียเวลาในการติดต่อกับบุคคลได้
3. มีความรู้ความเข้าใจในระบบการส่งสัญญาณวิทยุมากขึ้น
4. มีความรู้ความเข้าใจในการเขียนโปรแกรมมากขึ้น

1.5 อุปกรณ์ที่ต้องใช้

1.5.1 ฮาร์ดแวร์(Hardware)

- | | |
|--|-----------------|
| - เครื่องคอมพิวเตอร์สำหรับเขียนโปรแกรม | จำนวน 1 เครื่อง |
| - โมดูลจอแสดงผล | จำนวน 1 ตัว |
| - ไมโครคอนโทรลเลอร์ | จำนวน 2 ตัว |

- โมดูลส่งสัญญาณวิทยุ จำนวน 2 ตัว
- โมดูลอาร์เอฟไอดี จำนวน 1 ตัว

1.5.2 ซอฟต์แวร์ (Software)

- Arduino IDE เป็นโปรแกรมสำหรับเขียนชุดคำสั่ง ลงในบอร์ด Arduino Nano เพื่อใช้งาน
- Tera Term เป็นโปรแกรมสำหรับเอาไว้เชื่อมต่อกับพอร์ตสื่อสาร

บทที่ 2

ทฤษฎีพื้นฐานที่ใช้

2.1 Arduino

2.1.1 Arduino

เป็นระบบที่ใช้ในการพัฒนาอุปกรณ์อิเล็กทรอนิกส์ต้นแบบ ซึ่งออกแบบให้ใช้งานง่ายทั้งฮาร์ดแวร์ และซอฟต์แวร์ Arduino มีผู้ริเริ่มเป็นชาวอิตาลีคนหนึ่ง ดังนั้นจึงอ่านออกเสียงไปในทางอิตาลีว่า อาดูโยโน่ หรือ บางคนก็อ่านว่า อาดูโน่ หรือ อาดูโยอีโน่ เริ่มต้นในปี 2005 ผู้ริเริ่มของ Arduino ชื่อว่า Massimo Banzi และ David Cuartielles ซึ่งอาศัยอยู่ในเมือง Ivrea ทางตะวันตกเฉียงเหนือของประเทศอิตาลี ทั้งสองคนตั้งใจสร้างอุปกรณ์ประเภทไมโครคอนโทรลเลอร์ราคาถูกที่นักเรียนนักศึกษาสามารถเข้าถึง และซื้อหามาเป็นเจ้าของได้ โรงงานเล็กๆ ในเมืองที่วานนี้ก็ถูกใช้เป็นที่ผลิตบอร์ด Arduino เวอร์ชันแรก โดยใช้ชื่อโครงการของพวกเขาว่า Arduin of Ivrea

นอกจากจะตั้งใจให้ราคาของอุปกรณ์นั้นถูกเมื่อเทียบกับไมโครคอนโทรลเลอร์ตระกูลอื่นๆ ในท้องตลาด และ Arduino สามารถพัฒนาโดยโปรแกรมที่ "แจกฟรี" ภายใต้เงื่อนไขในการใช้งานลักษณะเปิดใช้งานฟรี ดังนั้นจึงเลือกใช้การพัฒนาบนพื้นฐานของระบบ Wiring

ไมโครคอนโทรลเลอร์เปรียบเสมือนกับสมองของมนุษย์ คือมีหน้าที่คิด คำนวณทางคณิตศาสตร์ คำนวณทางลอจิก สั่งการ มีส่วนความจำ เพื่อใช้เป็นข้อมูลในการคำนวณ หรือประมวลผลต่างๆ "แต่จะไม่สามารถทำงานได้เอง" โดยไม่มีมือ เท้า แขน ขา หรือ ตา หู จมูก ซึ่งเปรียบได้กับ อุปกรณ์ส่วนควบ (Accessories) อื่น เช่น เซนเซอร์ มอเตอร์ ระบบสื่อสารผ่านอินเทอร์เน็ต ระบบแสดงผลผ่านจอภาพ เป็นต้น ดังนั้นโดยสรุปคือ ไมโครคอนโทรลเลอร์จะทำหน้าที่ในการคิดคำนวณ รับค่าจากระบบวัดผลภายนอก เข้ามาประมวลผล เพื่อสั่งการตอบสนองออกไปที่อุปกรณ์ต่อเชื่อมอื่นๆ ตัวมันเองจริงๆ จะทำอะไรไม่ได้มากไปกว่าการคิด

ไมโครคอนโทรลเลอร์ในปัจจุบันก็มีอยู่หลายยี่ห้อ เช่น PIC ของบริษัทไมโครชิพ Z80 MCS-51 ARM-Cortex AVR และ อื่นๆ อีกมากมาย Arduino ก็เป็นไมโครคอนโทรลเลอร์แบบหนึ่งที่มีเอกลักษณ์เฉพาะตัวที่ต่างจากยี่ห้ออื่นๆ คือ การเป็นการใช้งานแบบฟรี ซึ่งทำให้ได้เปรียบเรื่องราคาและจำนวนผู้ใช้งานทั่วโลก

Arduino เวอร์ชันแรกปรากฏต่อสายตาชาวโลกในเดือนกันยายน ปี 2006 เรียกชื่อว่า Arduino Mini และก็เล็กราคาถูก ราคาขายในเมืองไทยก็ตกอยู่ไม่กี่ร้อยบาท

ถึงปัจจุบัน Arduino มีบอร์ดหลายแบบให้เลือกใช้งานตามความถนัดและความเหมาะสมมากกว่า 20 รุ่น แต่ละรุ่นก็มีขนาด ความจุ ความเร็ว จำนวนขาพอร์ตอินพุต เอาท์พุต แตกต่างกันไป มีตั้งแต่ราคาหลักสองสามร้อยบาท ไปจนกระทั่งพันกว่าบาท นอกจากนี้ยังมีอุปกรณ์เชื่อมต่อ (Shield) ให้อีกสารพัด ราคาที่เป็นไปตามการใช้งาน คือ สมเหตุสมผล ทำให้เป็นที่นิยมมากขึ้นเรื่อยๆ

สาเหตุที่ราคาถูกก็สามารถสรุปได้ดังนี้

- ระบบเป็นแบบ เปิดใช้ฟรี ไม่มีลิขสิทธิ์ในการนำไปใช้งานต่อเชิงพาณิชย์ แจก ไฟล์ที่ใช้ในการ สร้างต้นแบบให้ฟรีๆ ทำให้ประเทศผู้ผลิตอย่างจีนสามารถนำไปผลิตได้ในราคาถูกสบายๆ โดยไม่ต้อง กังวลเรื่องค่าลิขสิทธิ์

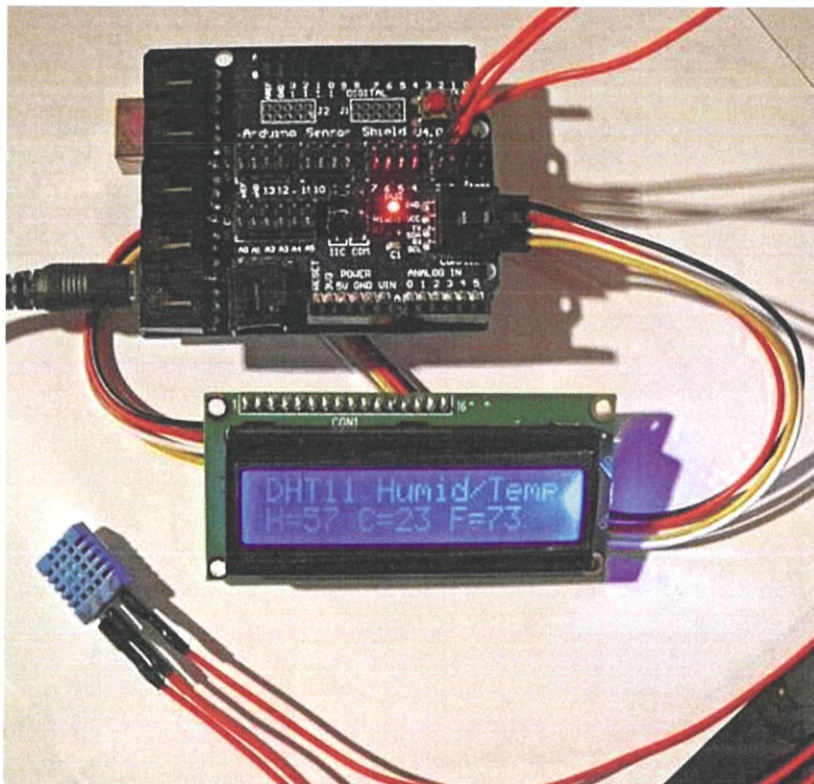
- ซอฟต์แวร์ หรือ Arduino IDE ที่ใช้ในการพัฒนายังแจกให้ฟรี ดาร์ลโหลดกันได้อย่างถูกกฎหมาย เอาไปใช้งานต่อ สร้างผลิตภัณฑ์แล้วขายต่อ ก็ไม่ต้องเสียเงินให้แบบไมโครคอนโทรลเลอร์

เจ้าอื่นๆ

- มีซอฟต์แวร์ (แบบฟรีๆ) ที่สร้างโดยคนใจดี เข้ามาร่วมอุดมการณ์ (เช่น Fritzing และ Processing) กันอีก ทำให้เกิดเป็นชุมชนขนาดใหญ่ที่มีเครื่องมือเครื่องมือเข้ามาให้ใช้กันฟรีๆ เยอะมาก ทั้ง Blog และ website สารพัด

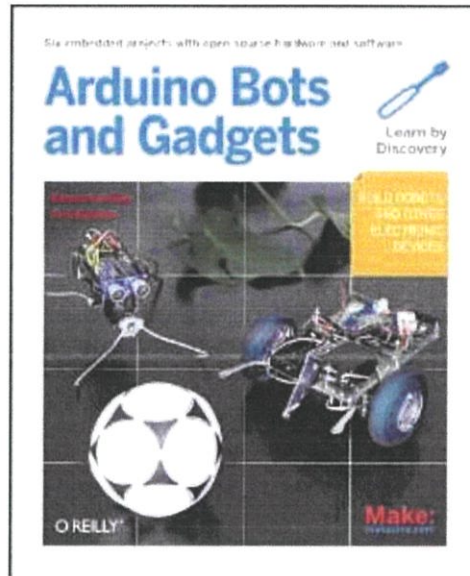
- อย่างที่ว่าครับ ชุมชนคนใช้ Arduino ในต่างประเทศมีอยู่มากมาย หาได้จาก Website ดังๆ เช่น Arduino.cc , Makezine.com, instructables.com เว็บพวกนี้แจกแบบร่างและไฟล์ติดตั้ง (Sketch) ให้ฟรีๆ เอาไปสร้างโดยหาวีสดุเอารอบๆตัว

ยกตัวอย่างอุปกรณ์ Arduino แบบต่างๆดังนี้



รูปที่ 2.1 เครื่องมือวัดและแสดงอุณหภูมิ

- หุ่นยนต์



รูปที่ 2.2 หุ่นยนต์

- QuadRotor



รูปที่ 2.3 Drone

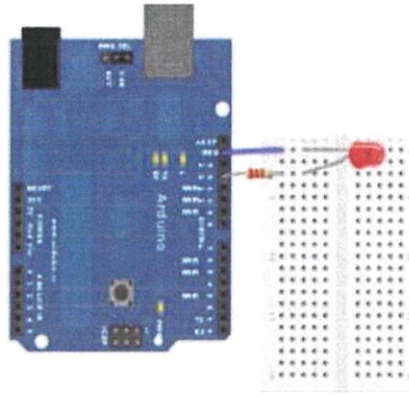
ใช้งานกับระบบปฏิบัติการแอนดรอยด์ เช่น มือถือ



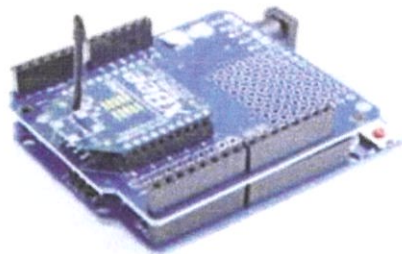
รูปที่ 2.4 หน้าจอแสดงผล

สรุปคือเป็นบอร์ดไมโครคอนโทรลเลอร์ตระกูล AVR ที่มีการพัฒนาแบบเปิดฟรี คือมีการเปิดเผยข้อมูลทั้งด้าน Hardware และ Software ตัว บอร์ด Arduino ถูกออกแบบมาให้ใช้งานได้ง่าย ดังนั้นจึงเหมาะสำหรับผู้เริ่มต้นศึกษา ทั้งนี้ผู้ใช้งานยังสามารถดัดแปลง เพิ่มเติม พัฒนาต่อยอดทั้งตัวบอร์ด หรือโปรแกรมต่อได้อีกด้วย

ความง่ายของบอร์ด Arduino ในการต่ออุปกรณ์เสริมต่างๆ คือผู้ใช้งานสามารถต่อวงจรอิเล็กทรอนิกส์จากภายนอกแล้วเชื่อมต่อเข้ามาที่ขา I/O ของบอร์ด (ดูตัวอย่างรูปที่ 1) หรือเพื่อความสะดวกสามารถเลือกต่อกับบอร์ดเสริม (Arduino Shield) ประเภทต่างๆ (ดูตัวอย่างรูปที่ 2) เช่น ArduinoXbee Shield, Arduino Music Shield, Arduino Relay Shield, Arduino Wireless Shield, Arduino GPRS Shield เป็นต้น มาเสียบกับบอร์ดบนบอร์ด Arduino แล้วเขียนโปรแกรมพัฒนาต่อได้เลย



รูปที่ 2.5 บอร์ด Arduino ต่อกับ LED

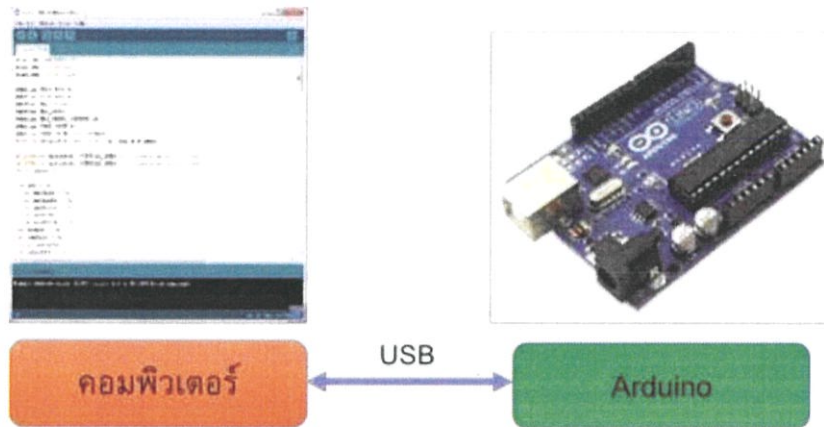


รูปที่ 2.6 บอร์ด Arduino ต่อกับ Xbee

2.1.2 จุดเด่นที่ทำให้บอร์ด Arduino เป็นที่นิยม

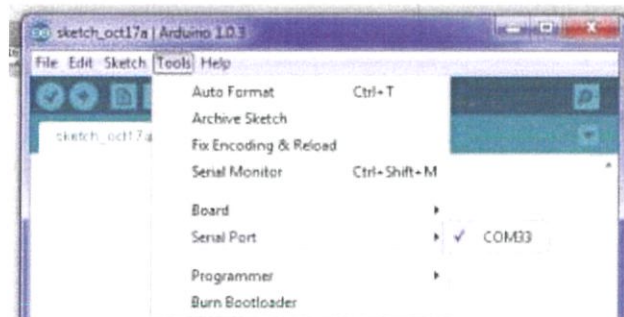
- ง่ายต่อการพัฒนา มีรูปแบบคำสั่งพื้นฐาน ไม่ซับซ้อนเหมาะสำหรับผู้เริ่มต้น
- มี Arduino Community กลุ่มคนที่ร่วมกันพัฒนาที่แข็งแกร่ง
- ผู้ใช้สามารถนำบอร์ดไปต่อยอดใช้งานได้หลายด้าน
- ราคาไม่แพงจนเกินไป
- Cross Platform สามารถพัฒนาโปรแกรมบน OS ใดก็ได้

2.1.3 รูปแบบการเขียนโปรแกรมบน Arduino



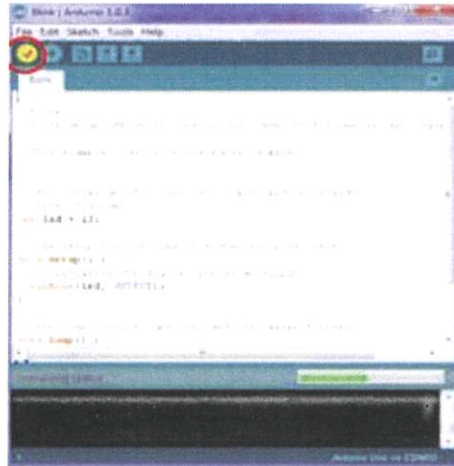
รูปที่ 2.7 การเขียนโปรแกรมบน Arduino

1. เขียนโปรแกรมบนคอมพิวเตอร์ ผ่านทางโปรแกรม Arduino IDE ซึ่งสามารถดาวน์โหลดได้จาก Arduino.cc/en/main/software
2. หลังจากที่เราเขียนโค้ดโปรแกรมเรียบร้อยแล้ว ให้ผู้ใช้งานเลือกรุ่นบอร์ด Arduino ที่ใช้และหมายเลข com พอร์ต

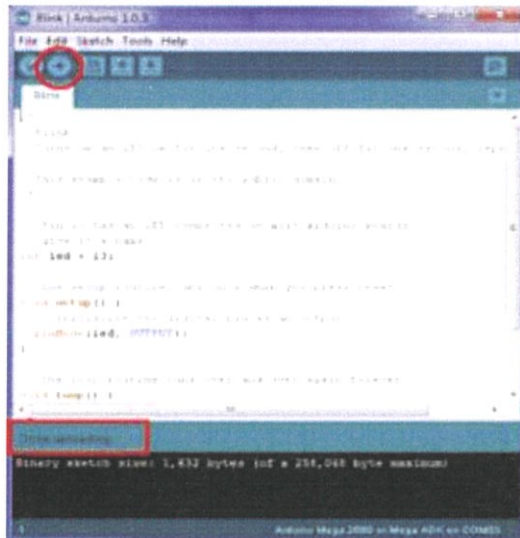


รูปที่ 2.8 เลือกหมายเลข Com พอร์ต ของบอร์ด

3. กดปุ่ม Verify เพื่อตรวจสอบความถูกต้องและ Compile โค้ดโปรแกรม จากนั้น กดปุ่ม Upload โค้ด โปรแกรมไปยังบอร์ด Arduino ผ่านทางสาย USB เมื่ออัปโหลดเรียบร้อยแล้ว จะแสดงข้อความแถบข้างล่าง “Done uploading” และบอร์ดจะเริ่มทำงานตามที่เขียนโปรแกรมไว้ได้ทันที

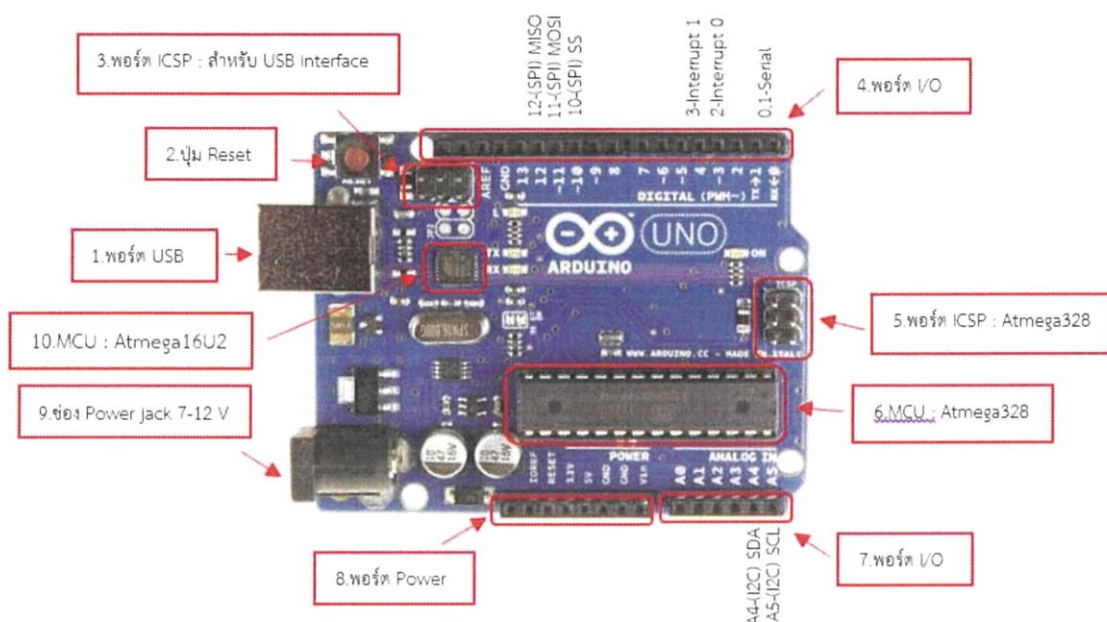


รูปที่ 2.9 กดปุ่ม Verify เพื่อตรวจสอบความถูกต้อง



รูปที่ 2.10 Upload โค้ดโปรแกรม

4. Layout & Pin out Arduino Board (Model: Arduino UNO R3)



รูปที่ 2.11 บอร์ด Arduino UNO R3

1. USBPort: ใช้สำหรับต่อกับ Computer เพื่ออัปโหลดโปรแกรมเข้า MCU และจ่ายไฟให้กับบอร์ด
2. Reset Button: เป็นปุ่ม Reset ใช้กดเมื่อต้องการให้ MCU เริ่มการทำงานใหม่
3. ICSP Port ของ Atmega16U2 เป็นพอร์ตที่ใช้โปรแกรม Visual Com port บน Atmega16U2
4. I/OPort: Digital I/O ตั้งแต่ขา D0 ถึง D13 นอกจากนี้ บาง Pin จะทำหน้าที่อื่นๆ เพิ่มเติมด้วย เช่น Pin0,1 เป็นขา Tx,Rx Serial, Pin3,5,6,9,10 และ 11 เป็นขา PWM
5. ICSP Port: Atmega328 เป็นพอร์ตที่ใช้โปรแกรม Bootloader
6. MCU: Atmega328 เป็น MCU ที่ใช้บนบอร์ด Arduino
7. I/OPort: นอกจากจะเป็น Digital I/O แล้ว เปลี่ยนเป็น ช่องรับสัญญาณอนาล็อก ตั้งแต่ขา A0-A5
8. Power Port: ไฟเลี้ยงของบอร์ดเมื่อต้องการจ่ายไฟให้กับวงจรภายนอก ประกอบด้วยขาไฟเลี้ยง +3.3 V, +5V, GND, Vin
9. Power Jack: รับไฟจาก Adapter โดยที่แรงดันอยู่ระหว่าง 7-12 V
10. MCU ของ Atmega16U2 เป็น MCU ที่ทำหน้าที่เป็น USB to Serial โดย Atmega328 จดติดต่อกับ Computer ผ่าน Atmega16U2

2.2 ทฤษฎีเกี่ยวกับ อาร์เอฟไอดี (RFID - Radio frequency identification)

เทคโนโลยีอาร์เอฟไอดี คือ เทคโนโลยีหนึ่งที่ใช้ในการระบุสิ่งต่างๆ โดยอาศัยคลื่นวิทยุ ซึ่งต่างจากเทคโนโลยีอื่นๆ เช่น บาร์โค้ดที่อาศัยคลื่นแสง หรือการสแกนลายนิ้วมือ เป็นต้น ในส่วนนี้จะอธิบายให้เข้าใจถึงหลักการของเทคโนโลยีอาร์เอฟไอดี และแนวคิดต่างๆ ที่เกี่ยวข้องเกี่ยวกับเทคโนโลยีนี้

ในปัจจุบันนี้เทคโนโลยีบ่งชี้อัตโนมัตินี้ (Auto-ID: automatic identification) ได้เข้ามามีบทบาทสำคัญในการดำเนินชีวิตประจำวัน ซึ่งเห็นได้จากการนำมาประยุกต์ใช้งานในหลายๆ ด้าน เช่น โลจิสติกส์ ระบบคลังสินค้า ร้านค้าปลีก และสายการผลิตในโรงงาน เป็นต้น โดยมีวัตถุประสงค์เพื่อใช้แสดงตัวตนของมนุษย์ สัตว์ สินค้า และวัตถุดิบในกระบวนการผลิต โดยทั่วไปเทคโนโลยีบ่งชี้อัตโนมัตินี้แบบต่างๆ ได้ถูกคิดค้นขึ้นมาเพื่ออำนวยความสะดวกในการทำธุรกิจ โดยเฉพาะอย่างยิ่งในการบันทึกข้อมูลแบบอัตโนมัติอย่างรวดเร็ว แทนที่จะต้องใช้การนับหรือจดบันทึกด้วยมนุษย์ ซึ่งอาจเกิดข้อผิดพลาดได้ง่าย






คลื่นวิทยุกับเทคโนโลยีอาร์เอฟไอดี

จากที่กล่าวในขั้นต้นว่า เทคโนโลยีอาร์เอฟไอดี อาศัยคลื่นวิทยุในการทำงาน ดังนั้นเมื่อพูดถึงเทคโนโลยีอาร์เอฟไอดี สิ่งหนึ่งที่ขาดไม่ได้ที่จะต้องกล่าวถึง คือคลื่นวิทยุ

คลื่นวิทยุ (Radio frequency) เป็นคลื่น Electromagnetic ประเภทหนึ่ง ที่มีความยาวคลื่นระหว่าง 0.1 cm. ถึง 1,000 km. หรืออยู่ในช่วงความถี่ระหว่าง 30 Hz และ 300 GHz เมื่อเป็นคลื่นวิทยุจะเห็นได้ว่า วัสดุที่นำใช้กับคลื่นวิทยุย่อมมีผลต่อการใช้งานอย่างหลีกเลี่ยงไม่ได้ ในวัสดุประเภทที่ที่คลื่นวิทยุสามารถผ่านได้สะดวกโดยไม่มีการสูญเสียพลังงานใด วัสดุเหล่านี้เรียกว่า RF-lucent หรือ RF-friendly หากนำเทคโนโลยีอาร์เอฟไอดี มาใช้กับวัสดุเหล่านี้จะไม่มีผลเสียต่อการใช้งาน อย่างไรก็ตามยังมีวัสดุบางประเภทที่เป็นอุปสรรคในการนำเทคโนโลยีอาร์เอฟไอดี มาใช้งาน วัสดุประเภทแรกเรียกว่า RF-opaque วัสดุประเภทนี้จะหักเหคลื่นวิทยุ หรือทำให้คลื่นวิทยุกระจัดกระจายออกไป ส่วนวัสดุอีกประเภทเรียกว่า RF-absorbent คลื่นวิทยุสามารถที่จะผ่านวัสดุประเภทนี้ได้ แต่อย่างไรก็ตามคลื่นที่ผ่านมานั้นจะถูกดูดซับไว้หมด หรือต้องสูญเสียพลังงานมากในการที่จะทะลุผ่านได้

ถึงแม้ว่า วัสดุแต่ละประเภทจะมีผลต่อคลื่นวิทยุ อย่างไรก็ตามวัสดุประเภทหนึ่งจะมีผลคลื่นวิทยุแต่ละช่วงความถี่ที่แตกต่างกัน กล่าวคือ วัสดุนั้นอาจจะมีลักษณะเป็น RF-lucent ในคลื่นความถี่หนึ่ง ในขณะที่วัสดุเดียวกันนี้อาจจะเป็น RF-opaqueหรือRF-absorbent ในคลื่นความถี่ในช่วงอื่นก็ได้ ดังตัวอย่างต่อไปนี้

ตารางที่ 2.1 แสดงวัสดุและช่วงความถี่

วัสดุ	LF	HF	UHF	Microwave
ผ้า 	RF-lucent	RF-lucent	RF-lucent	RF-lucent
ไม้แห้ง 	RF-lucent	RF-lucent	RF-lucent	RF-absorbent
Graphite 	RF-lucent	RF-lucent	RF-opaque	RF-opaque
โลหะ 	RF-lucent	RF-lucent	RF-opaque	RF-opaque
น้ำมันเครื่อง 	RF-lucent	RF-lucent	RF-lucent	RF-lucent

ตารางที่ 2.1 แสดงวัสดุและช่วงความถี่ (ต่อ)

วัสดุ	LF	HF	UHF	Microwave
กระดาษ 	RF-lucent	RF-lucent	RF-lucent	RF-lucent
พลาสติก 	RF-lucent	RF-lucent	RF-lucent	RF-lucent
แชมพู 	RF-lucent	RF-lucent	RF-absorbent	RF-absorbent
น้ำ 	RF-lucent	RF-lucent	RF absorbent	RF-absorbent
ไม้ซัน 	RF-lucent	RF-lucent	RF-absorbent	RF-absorbent

จากที่กล่าวในขั้นต้นว่า เทคโนโลยี อาร์เอฟไอดี ใช้คลื่นวิทยุ ในการทำงาน และคลื่นวิทยุที่ใช้ในเทคโนโลยีอาร์เอฟไอดีจะอยู่ในช่วงความถี่ระหว่าง 30 Hz และ 300 GHz

จากช่วงความถี่ดังกล่าวทำให้สามารถแบ่งคลื่นวิทยุได้เป็น 4 ประเภท โดยจะกล่าวถึงรายละเอียดในบทถัดไปครับ

อย่างไรก็ตาม ปัจจุบันเทคโนโลยีอาร์เอฟไอดี ได้มีการพัฒนาขึ้นไปมาก โดยเฉพาะอย่างยิ่งเทคโนโลยี UHF RFID Tag สำหรับคลื่นความถี่ UHF ได้มีการพัฒนาให้สามารถทำงานบนพื้นผิวโลหะหรือความชื้นได้มากขึ้น

2.3 หน้าจอ Monitor (AMOLED, LCD, TFT, LED)

ผู้ผลิตอุปกรณ์เทคโนโลยีไม่ว่าจะเป็นแท็บเล็ต โทรศัพท์ ทีวี หรือจอคอมพิวเตอร์ มักพยายามโน้มน้าวให้คนทั่วไปไปทราบว่าอุปกรณ์ของพวกเขามีการแสดงผลที่ดีกว่าของเจ้าอื่น โดยพยายามสร้างชื่อเรียกหรือคำย่อที่ฟังดูมีสไตล์และบ่งบอกถึงความยิ่งใหญ่ของเทคโนโลยี คุณผู้อ่านคงเคยได้ยินมาบ้าง เช่น Super AMOLED, PenTile, LED, IPS, SUPER-IPS คำเหล่านี้บางคำเป็นชื่อที่ใช้เรียกเทคโนโลยีที่ถูกต้อง แต่บางคำก็เป็นเพียงแค่ชื่อเรียกในการตลาด

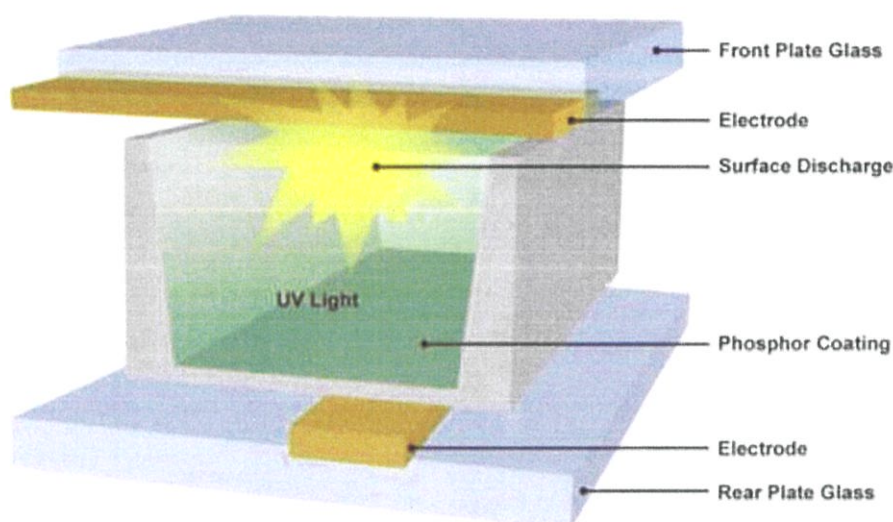


รูปที่ 2.12 หน้าจอ smart phone

ในความเป็นจริงแล้ว คุณอาจจะพบหน้าจอแสดงผลแบบดิจิตอลในชีวิตประจำวันเพียง 3 ประเภทเท่านั้น คือ LCD, OLED และ plasma Plasma

ในปัจจุบันเราจะพบจอแสดงผลแบบ plasma เฉพาะในทีวีที่มีความละเอียดสูงและมีขนาดใหญ่เท่านั้น (HDTVs) ซึ่งจอ plasma มีความคมชัดที่ยอดเยี่ยม จอ plasma ไม่เปลี่ยนสีหรือเปลี่ยนความสว่างแม้จะดูจากด้านข้าง จึงเหมาะสำหรับสถานที่สาธารณะเช่นในสนามกีฬาหรือโดมกว้างที่มีคนดูจำนวนมาก ซึ่งในตลาดจะจำหน่ายจอ Plasma HDTV ราคาค่อนข้างสูงกว่าจอ LCD ในขนาดเท่ากัน ส่วนต่างของราคามาจากการที่จอ plasma ใช้เวลาสั้นมากในการตอบสนอง (หมายถึงการเปลี่ยนสีของพิกเซลอย่างรวดเร็ว) ซึ่งจะทำให้จอมีอายุการใช้งานที่สั้นลงกว่าจอ plasma

มีข้อเสียบางอย่างที่เสียเปรียบจอบแบบอื่น นั่นคือมันไม่สามารถนำไปใช้กับอุปกรณ์ที่มีขนาดเล็กได้ เนื่องจากมันหนาและหนัก แถมยังกินไฟเกินกว่าจะนำไปใส่ในสมาร์ทโฟนหรือแท็บเล็ต นอกจากนี้ยังเป็นเรื่องยากที่จะบีบเอาจุดพิกเซลจำนวนมากใส่ลงไป ในจอ plasma ที่มีขนาดเล็กทำให้อุปกรณ์นั้นจะพบกับปัญหาความละเอียดต่ำในที่สุด ซึ่งในมิตินี้ของเทคโนโลยี plasma จะมีช่องว่างระหว่างจุดพิกเซลซึ่งปกติแล้วผู้ใช้งานจะมองไม่เห็นช่องว่างนี้เมื่อนั่งห่างออกไป 10 ฟุตจากหน้าจอทีวีขนาดใหญ่ แต่ถ้าเป็นบนเครื่องแล็ปท็อปหรือ พีซี มันสามารถตรวจพบได้อย่างแน่นอน



รูปที่ 2.13 แผนภาพแสดง จอ plasma ที่ขยายให้ดูรายละเอียดภายใน (เซลล์เดียว)

วิธีการทำงานหน้าจอแสดงผล ประกอบขึ้นจากแผ่นแก้วสองชุดวางชิดกัน ช่องว่างนี้จะถูกแบ่งออกเป็นเซลล์แสงกว้าง 100-200 μ มีชั้นผนังกันไว้ โดยใช้ขั้วไฟฟ้าในแนวระจกคอยควบคุมตำแหน่งของเซลล์เหล่านั้น แต่ละเซลล์จะบรรจุก๊าซที่ผสมระหว่างก๊าซซีนอนและก๊าซเฉื่อยอื่นๆ กลไกการทำงานของจอภาพพลาสมา จะมีการเรืองแสงขึ้นเองเหมือนการทำงานของหลอดฟลูออเรสเซนต์ กล่าวคือ ก๊าซในเซลล์เหล่านี้เมื่อถูกกระตุ้นด้วยแรงดันไฟฟ้าจะเกิดการไอออนไนซ์ขึ้นทำให้ก๊าซแตกประจุและปล่อยแสงอุลตราไวโอเล็ตออกมา สารเรืองแสงจะดูดซับอุลตราไวโอเล็ตและสร้างสีที่มองเห็นได้ด้วยตา ทำให้เรามองเห็นเป็นภาพได้จอบแสดงผลแบบดิจิตอลที่ใช้อุปกรณ์สมาร์ทโฟน OLED (Organic Light-Emitting Diode) ไดโอดเปล่งแสงชีวภาพเป็นเทคโนโลยีที่ถูกนำมาใช้ในการแสดงผลแบบดิจิตอล ผู้ใช้มักจะมี OLED ที่มีชื่อเรียกในการตลาดว่า AMOLED ซึ่ง "AM" มาจากคำว่า Active Matrix

จอบแสดงผลแบบ OLED นั้นแบ่งตามชนิดได้ 2 ประเภทคือ

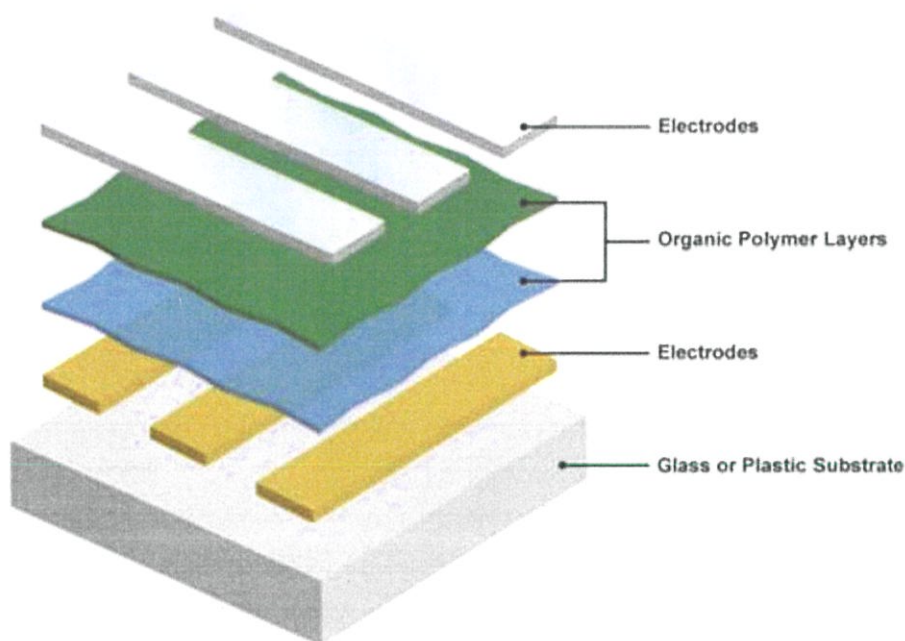
- Passive matrix OLED Displays (PMOLED) มักพบในจอบ PDA แบบเก่าและเครื่องเล่น MP3 ขนาดเล็ก จอบมือถือใหม่ๆ มักจะใช้อีกแบบหนึ่งมากกว่า

- Active matrix OLED Displays (AMOLED) ถูกนำไปใช้ในสมาร์ทโฟนและแท็บเล็ตในปัจจุบันหลายๆ รุ่น ภายในมีลักษณะเป็นแผ่นฟิล์มบาง มีวงจรในตัวเอง สามารถควบคุมการเกิดภาพได้เองภายในชั้นฟิล์ม

2.3.1 อะโมเลด (AMOLED)

ข้อดีคือ มีความคมชัดมาก ให้สีสันสดใส แสดงผลสีดำได้คมชัด มีเวลาตอบสนองที่รวดเร็ว มีองศาในการมองที่กว้าง และประหยัดพลังงานโดยใช้พลังงานน้อยกว่าแบบแรกและยังสามารถขยายให้มีขนาดใหญ่ได้ จึงถูกนำไปใช้ทำจอวิดีโอ คอมพิวเตอร์ ที่วีขนาดใหญ่ หรือป้ายโฆษณาขนาดใหญ่ด้วย

ข้อเสียคือ ค่าใช้จ่ายมีราคาแพงกว่าจอ LCD หรือ plasma และยังมีปัญหาเกี่ยวกับสีฟ้าอยู่บ่อยครั้ง ซึ่งเมื่อใช้งานไปนานๆ ไดโอดสีน้ำเงินจะมีประสิทธิภาพน้อยลง แต่ก็ได้รับการแก้ไขให้ดีขึ้นเรื่อยๆ ใน AMOLED รุ่นล่าสุดในปัจจุบัน ความสมดุลของสีในการใช้งานระยะยาวยังคงเป็นสิ่งที่ท้าทายผู้ผลิต และที่ควรระวังก็คือจอแบบ AMOLED จะกินไฟมากกว่า LCD เมื่อต้องแสดงผลสีขาว (ซึ่งหน้าเว็บปกติหรือโปรแกรมประยุกต์ส่วนใหญ่มักจะมีสีขาว จึงไม่น่าแปลกใจที่แบตเตอรี่จะหมดไว) Super AMOLED และ Super AMOLED Plus เป็นชื่อในการตลาดสำหรับการพัฒนามาตรฐานเทคโนโลยี OLED ซึ่งอาจมีการเปลี่ยนแปลงรูปแบบที่แตกต่างกันของสีแดง สีฟ้า สีเขียว หรือ subpixels หรือ การเพิ่มความไวต่อการสัมผัส แต่อย่างไรก็ตามยังคงอยู่บนพื้นฐานของ OLED อยู่ดี



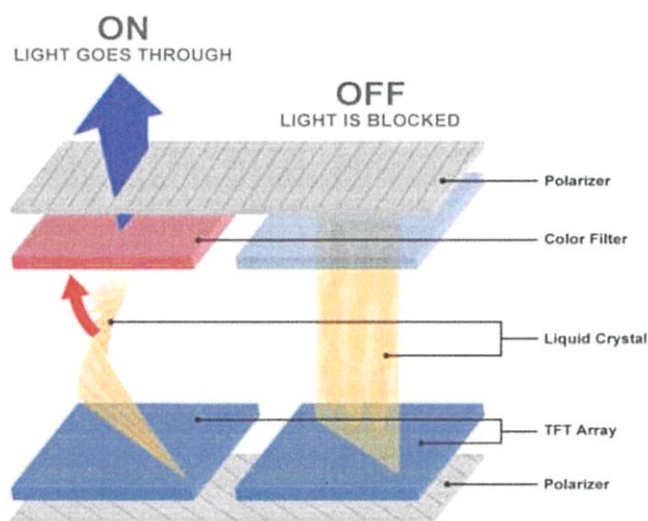
รูปที่ 2.14 แสดงเทคโนโลยี OLED ที่ใช้โพลิเมอร์อินทรีย์เรืองแสงในการผลิตแสงสี

2.3.2 LCD

หน้าจอแบบ LCD มีหลายรูปแบบออกเป็นดังนี้

2.3.2.1 แอลซีดี LCD (Liquid Crystal Displays) หรือจอแบบแอลซีดี มักถูกใช้ทำเป็นจอของ HDTV หลายๆ รุ่น รวมถึงจอภาพของเครื่องคอมพิวเตอร์ โน้ตบุ๊ก แท็บเล็ตและมือถือด้วย ซึ่งเทคโนโลยีนี้ได้รับการปรับปรุงอย่างมากมาในช่วงหลายๆ ปีนี้ ขอกกล่าวถึงรูปแบบของ LCD ที่สำคัญๆ 3 แบบ ได้แก่ LCD twisted nematic (TN), LCD In-Plane Switching (IPS) และ LCD patterned vertical alignment (VA)

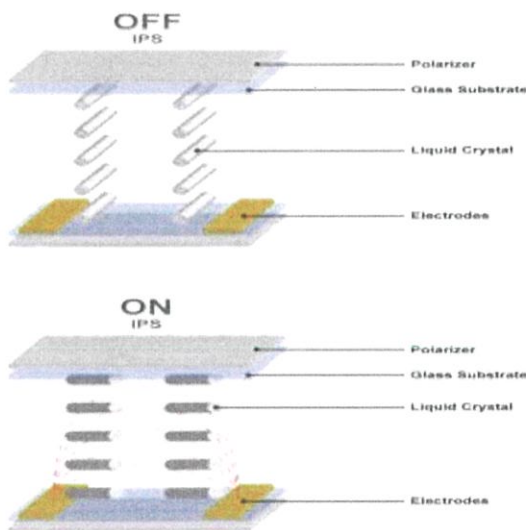
2.3.2.2 แอลซีดี ทวิสต์เตด เนมาติก TN (Twisted Nematic) เป็นจอแสดงผลที่มีราคาไม่แพงและง่ายต่อการผลิต มีการตอบสนองที่ค่อนข้างรวดเร็ว แต่ก็ยังไม่ใกล้เคียงกับ OLED หรือ plasma ซึ่งเวลาตอบสนองที่รวดเร็วช่วยลดการเบลอเมื่อวัตถุเคลื่อนที่อย่างรวดเร็วในจอ และมีอัตราเฟรมที่สูงซึ่งเป็นสิ่งจำเป็นสำหรับการแสดงผลรูปแบบ 3D นำเสียดายที่ TN ให้คุณภาพต่ำที่สุดเมื่อมองโดยรวม เพราะมีช่วงของสีที่แคบ โดยแต่ละ subpixel จะมีสีแดง สีเขียว สีน้ำเงินเพียง 64 เฉดสี เท่านั้นเอง หากไม่สามารถแสดงผลสีที่ได้รับมาก็จะมีกระบวนการ "dithering" หรือการดึงจุดพิกเซลที่อยู่ติดกันให้ไล่สีใกล้เคียงกับข้อมูลมากที่สุด TN มีองศาในการมองไม่ค่อยดี หากมองจากด้านข้างสีอาจจะเลื่อนไปบ้าง หากดูจากด้านล่างมักจะมืด หากดูจากด้านบนมันจะเห็นกลางๆ ซึ่งสามารถไปพิสูจน์ได้จากเครื่องเล่นที่จอหลายๆ รุ่น สาเหตุมาจากเทคโนโลยี LCD TN นั้นมีการบิดของแสง (polarized) ผ่านตัวกรองหลายชั้น ทำให้แสง (สี) ที่ได้ออกมามีมุมมองที่เห็นชัดเจนเพียงแค่มุมเดียว (มองตรงๆ)



รูปที่ 2.15 LCD Twisted Nematic (TN หรือ TN-Film)

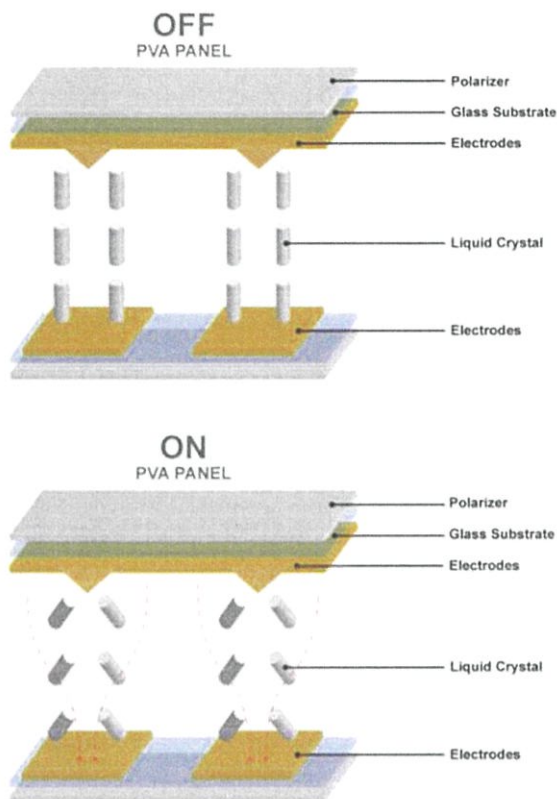
2.3.2.3 แอลซีดี อิน-เพลส สวิตซ์ซิ่ง IPS (In-Plane Switching) Hitachi พัฒนาเทคโนโลยี IPS มาตั้งแต่ปี 1996 เพื่อต่อสู้กับปัญหาเครื่องซักผ้าที่มองไม่ค่อยเห็นแผงควบคุม ซึ่งตอนนั้นเป็นจอ LCD TN ที่ยังคงพบอยู่ในปัจจุบัน และโดนต่อว่าเรื่องมุมมองที่

จำกัดเกินไป ตั้งแต่นั้นมาก็มีการปรับแต่งและปรับปรุงวิธีการใหม่ขึ้นเช่นซูเปอร์ IPS (S-IPS) แอดวานซ์ ซูเปอร์ IPS (AS-IPS) และ IPS Pro วันนี้จอแบบ LCD IPS ได้พัฒนาไปหลายสาย พันธุ์แต่ยังคงใช้ชื่อเรียกกันว่า IPS ซึ่งปัจจุบัน LG กลายเป็นผู้ผลิตจอ IPS ที่ใช้บนอุปกรณ์ส่วนใหญ่ที่โฆษณาว่ามีหน้าจอสัมผัสแบบ IPS หน้าจอแบบ IPS มีข้อดีที่สำคัญหลายๆ อย่างที่เอามาจากหน้าจอแบบ TN เช่นผลิตได้ไม่ยากและมีขนาดแพงกว่าจอแบบ TN เล็กน้อยเท่านั้น ซึ่ง subpixel สีแดง สีเขียว สีฟ้า สามารถแสดงสีได้ชนิดละ 8 บิต (256 ระดับ) และเพิ่มเติมความสว่างให้ชัดเจนมากขึ้น ทำให้ไม่ต้องไปใช้เทคนิคการไล่สีให้ชัดใจด้วยช่วงสีที่กว้างเพียงพอ ทำให้บรรดานักถ่ายภาพ ศิลปิน โปรดปรานจอแบบนี้อย่างมาก และเครื่อง PC ขึ้นนำจะมีจอแบบ IPS สำหรับนักถ่ายภาพ นักออกแบบ และยังถูกนำไปใช้งานการพิมพ์และงานอื่นๆ มากมาย ด้วยองศาในการมองที่กว้างมาก ผู้ใช้จะเห็นสีที่เหมาะสมและคมชัดแม้ว่ามองจากมุมไหนก็ตาม ข้อดีนี้ทำให้จอ IPS นำไปใช้ในสมาร์ทโฟนและแท็บเล็ตอย่างเช่น iPhone 4, iPhone 4S, iPad 2, new iPad, Amazon Kindle Fire และ Asus Eee Pad Transformer Prime เหล่านี้เป็นเพียงแค่บางส่วนเท่านั้นที่ใช้งานจอ IPS แต่ก็เชื่อว่าจะมีข้อดีเพียงอย่างเดียว โดยทั่วไปแล้วจอ IPS จะมีสีไม่สดใสเท่าจอแบบ TN (หากมองตรงๆ) และจอ IPS ก็ยังทำอัตรา refresh ได้ไม่บ่อยนักเมื่อเทียบกับแบบ TN ซึ่งคุณสมบัติดังกล่าวเป็นสิ่งจำเป็นมากที่สุดสำหรับการแสดงผลในแบบ 3D ทำให้กราฟิกที่เคลื่อนไหวอย่างรวดเร็วในบางครั้งอาจจะมึนตึบเห็นหลุดออกมาให้เห็นบ้าง แต่จอแบบ IPS ก็ได้ถูกพัฒนาอย่างต่อเนื่องเพื่อลดปัญหาดังกล่าว แต่นั่นก็ทำให้มีราคาแพงขึ้นตามเช่นกัน



รูปที่ 2.16 LCD In-Plane Switching (IPS)

2.3.2.4 แอลซีดี เวอร์ติคอลลี อลิแกนด VA (LCD Vertically Aligned) เป็นจอชนิดหนึ่งที่อยู่ระหว่าง IPS (ช้าแต่คุณภาพสูง) และ TN (สว่าง ความเร็วสูง แต่มีคุณภาพต่ำ) จอแบบ VA สามารถแสดงความสว่างได้ 8 บิตต่อ subpixel แต่ไม่มีความกว้างของสีเยอะเท่าแบบ IPS พุดง่ายๆ ก็คืออยู่กึ่งกลางระหว่างจอ LCD สองแบบแรก จอแบบ LCD VA ถูกแบ่งย่อยอีกเป็นหลายชนิด แต่โดยภาพรวมคือให้ภาพสีดำที่ดำมืดสนิทและมีอัตราส่วนความคมชัดที่ดีมาก



รูปที่ 2.17 LCD: Vertically Aligned (VA)

เชื่อว่าคนจำนวนมากจะสับสนกับตัวย่อ และชื่อต่างๆ เมื่ออ่านรายละเอียดเกี่ยวกับการแสดงผลแบบดิจิตอล ศัพท์เทคนิคจำนวนมากถูกตั้งขึ้นเพื่อใช้ในการตลาด ด้านล่างนี้เป็นคำอธิบายสั้นๆ ให้เข้าใจตรงกันมากขึ้น

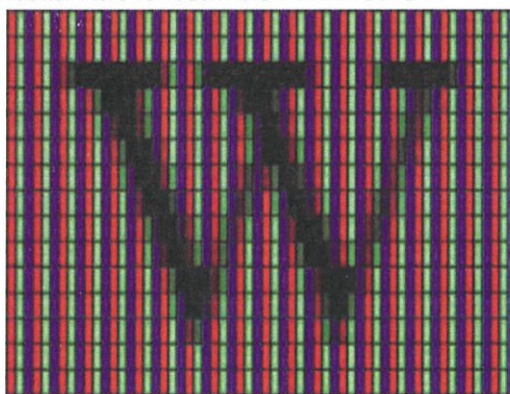
2.3.3 LED

2.3.3.1 แอลอีดี LED (light-emitting diode) คือชนิดของไฟที่อยู่เบื้องหลังของจอ LCD เป็น 1 ใน 2 ประเภทที่สำคัญ ไฟประเภทแรกคือ CCFL (cold cathode fluorescent lamp) ใช้เทคโนโลยีเช่นเดียวกับหลอดไฟนีออนภายในบ้าน บางและแบน ไฟอีกประเภทคือ LED (light-emitting diode) ซึ่งประเภทนี้ทำให้จอทีวีมีช่วงสีที่กว้างขึ้น ยืดอายุการใช้งาน และใช้พลังงานต่ำ ที่วิบารุ่นมี LEDs เฉพาะแนวขอบ (ถูกเรียกทางการตลาดว่า edge-lit LED) แต่ไม่ค่อยได้รับความนิยม 2.3.4 ทินฟิล์มทรานซิสเตอร์ TFT (thin-film transistor)

ทรานซิสเตอร์ที่มีฟิล์มบางเป็นสารตั้งต้น คล้ายแก้ว ถูกเคลือบด้วยแผ่นฟิล์มบางๆของโลหะซิลิคอนหรือพลาสติก มักถูกนำไปใช้คู่กับหน้าจอ LCD ของสมาร์ตโฟน เพื่อขับเซลล์ของ LCD ออกมา โดยถูกออกแบบให้มีแผ่นฟิล์มขนาดใหญ่ภายในมีทรานซิสเตอร์ขนาดเล็กและตัวเก็บประจุ ปัจจุบันจอแบบ AMOLED และ จอ LCD ส่วนใหญ่จะมี TFT เป็นส่วนประกอบ

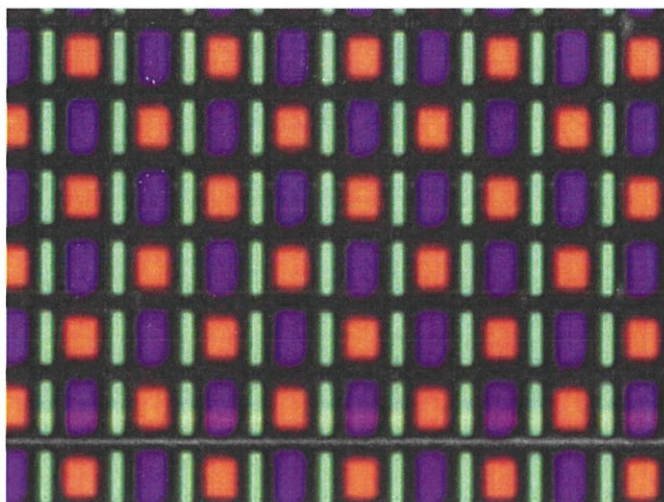
Active matrix : เป็นระบบการควบคุม subpixel (จุดพิกเซลย่อย) ของทรานซิสเตอร์แต่ละชุดและตัวเก็บประจุ (เช่นใน TFT) จะช่วยให้ควบคุมแรงดันได้แม่นยำมากขึ้นและมีการเปลี่ยนแปลงกระแสไฟเร็วกว่าเทคโนโลยีแบบ passive-matrix (เกือบทั้งหมดของจอแสดงผลดิจิทัลในปัจจุบันเป็นแบบ active matrix)

Passive matrix เทคโนโลยีช่วยควบคุมแรงดันไฟฟ้าของแต่ละ subpixels ด้วยรูปแบบตารางที่เรียบง่ายของวัสดุที่เป็นสื่อตัวนำไฟฟ้า ไม่ค่อยพบเห็นเทคโนโลยีนี้ในจอ LCD ปัจจุบันแล้วเนื่องจากราคาของ TFT ได้ถูกลงมากและมีคุณภาพดีขึ้น ซึ่ง Passive matrix มีความแม่นยำและการตอบสนองของพิกเซลน้อยกว่าแบบ Active matrix การจัดเรียง subpixel แบบมาตรฐาน ประกอบไปด้วยจุดพิกเซลสีแดง สีเขียว และสีน้ำเงิน บนจอแสดงผลดิจิทัลทั้งหมด จะประกอบไปด้วยจุดพิกเซลหลายๆ จุดเรียงกันอยู่บนหน้าจอ และภายในจุดพิกเซลจะมี subpixels ซึ่งโดยปกติจะประกอบด้วยสีแดง สีเขียว และสีฟ้า จากรูปแสดงการเปลี่ยนแปลงความสว่างของ subpixel ทั้งสามสีในการผลิตเฉดสีต่างๆ ออกมา ที่เราไม่เห็นเพราะว่า subpixels มีขนาดเล็กเกินไปที่จะมองด้วยตาเปล่า เราสามารถมองเห็นการผสมสีของ subpixel ออกมาเป็นสีผสมสีหนึ่ง ซึ่งหน้าจอแสดงผลบางอย่างอาจจะมี subpixel ที่สี ซึ่งใช้แสดงสีขาวหรือสีเหลือง แต่ก็พบได้ยาก



รูปที่ 2.18 การจัดเรียง subpixel แบบทั่วไป

การจัดวาง subpixel แบบ PenTile ถูกเปิดเผยขึ้นครั้งแรกผ่านสมาร์ทโฟน Google Nexus One ซึ่งใช้จอแสดงผลแบบ AMOLED และมีการเรียงตัวของ subpixel สีแดง เขียว น้ำเงิน ที่ต่างจากปกติที่มักจะยาวเท่าๆ กันและเรียงต่อกันเป็นสี่เหลี่ยมผืนผ้า แต่รูปแบบการเรียง subpixel แบบ PenTile จะมี subpixel ที่มีขนาดต่างกัน และมีจำนวนน้อยกว่าแบบปกติ โดยทำงานร่วมกับตัวควบคุมการแสดงผลแบบพิเศษ ซึ่งเป้าหมายก็คือการผลิตจุดพิกเซลจำนวนมากด้วยจำนวน subpixel ที่น้อยลง



รูปที่ 2.19 การจัดเรียง subpixel แบบ PenTile

การกำหนดความละเอียดจอแสดงผลแบบ Pentile นำไปสู่การโต้เถียงอย่างรุนแรง บ้างก็ว่าเป็นเทคโนโลยีที่อาศัย subpixels ของจุดข้างเคียงมาช่วยแสดงผลให้จุดพิกเซลนั้น แสดงสีได้ชัดขึ้น อย่างไรก็ตามในตอนนี้ PenTile ได้กลายเป็นเครื่องหมายการค้าของ Samsung ไปเรียบร้อยแล้ว

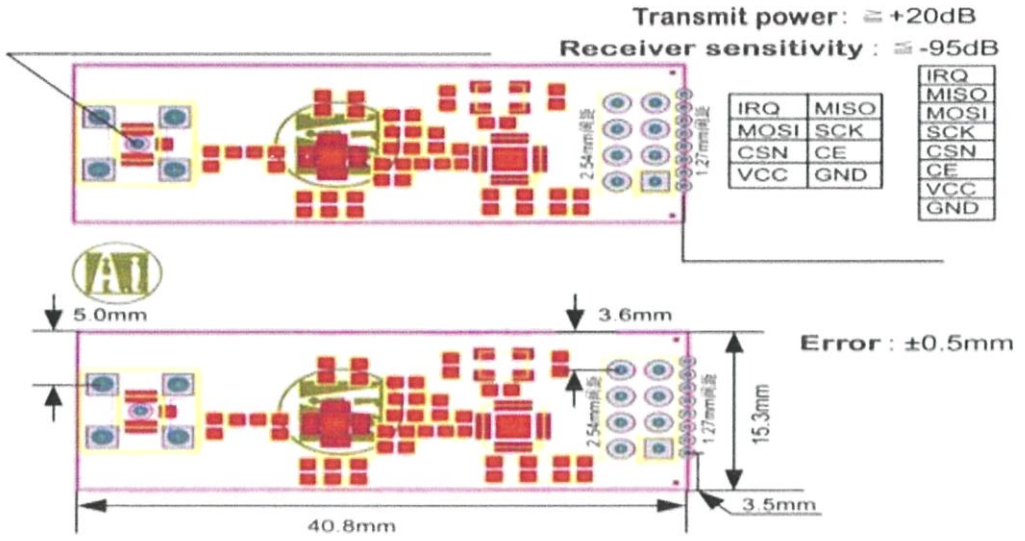
2.4 NRF24L01 Module โมดูลสื่อสารไร้สาย แบบ infrared

โมดูล NRF24L01 เป็นโมดูลสื่อสารไร้สาย ที่สามารถเขียนโปรแกรมให้เป็นที่ ทั้งตัวรับและตัวส่ง สามารถใช้กับ Arduino ได้หลาย ๆ ตัวพร้อมกัน มีความเร็ว 2.4G จึงสื่อสารได้รวดเร็วและไม่ต้องการเสาอากาศที่ยาว มีขนาดเล็กสะดวกในการต่อใช้งาน สามารถประยุกต์ใช้งานได้หลายอย่างเช่น ใช้เป็นอุปกรณ์ส่งข้อมูลของเซนเซอร์อัตโนมัติสำหรับควบคุม อุณหภูมิ ความชื้น การแจ้งเตือนต่าง ๆ ควบคุมและติดตามหุ่นยนต์ Robot Control and Monitoring ได้ในระยะ 15-500 เมตร โมดูลนี้ใช้ชิพ nRF24L01+ m ทำงานด้วยความเร็วสูง High-speed SPI interface ใช้พลังงานต่ำ รองรับการทำงานร่วมกับ Arduino และมีเสาอากาศมาให้ในตัว



รูปที่ 2.20 NRF24L01 Module โมดูลสื่อสารไร้สาย แบบ infrared

Interface Size

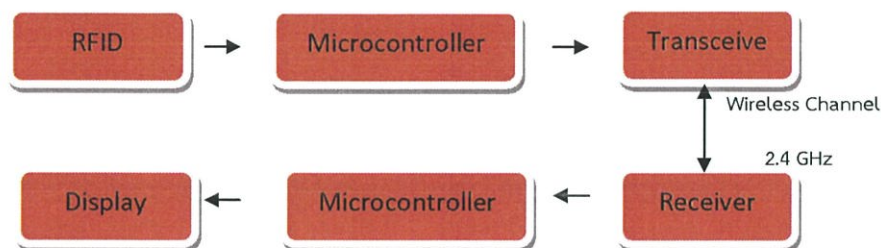


รูปที่ 2.21 interface size

บทที่ 3

การวิเคราะห์และออกแบบระบบ

3.1 Block Diagram



รูปที่ 3.1 block diagram

การออกแบบในส่วนของฮาร์ดแวร์ ในส่วนของภาคส่งจะใช้โมดูล อาร์เอฟไอดี เชื่อมต่อกับ ไมโครคอนโทรลเลอร์ ArduinoUno (ATmega328) ได้โดยตรง และใช้โมดูลส่งสัญญาณไร้สาย ในการส่งข้อมูลไปยังภาครับ ในส่วนของภาครับ จะรับสัญญาณผ่านทาง โมดูลสื่อสารไร้สาย จากนั้นจะส่งข้อมูลต่อไปยังไมโครคอนโทรลเลอร์ Arduino Uno (Atmega328) จากนั้นจะแสดงผลผ่านทาง TFT (Thin-film transistor)

3.2 การออกแบบฮาร์ดแวร์(Hardware)

3.2.1 คุณสมบัติและการทำงานของArduinoUNO

Arduino UNO ใช้ ATmega328 รั้นที่ความถี่16 MHz หน่วยความจำแฟลช 32 KB แรม 2 KB บอร์ดใช้ไฟเลี้ยง 7 ถึง 12 V มีระดับแรงดันไฟฟ้าในการทำงานและขาสัญญาณอยู่ที่ 5 V (TTL) มี Digital Input / Output 14 ขา (เป็น PWM ได้ 6 ขา) มี Analog Input 6 ขา Serial UART 1 ชุด I2C 1 ชุด SPI 1 ชุดเขียนโปรแกรมบนซอฟต์แวร์ Arduino IDE และโปรแกรมผ่านพอร์ต USB

ในส่วนของการทำงานฝั่งส่ง Arduino UNO จะรับค่ามาจาก อาร์เอฟไอดี หลังจากนั้นจะประมวลผลได้ เมื่อได้ข้อมูลแล้วจะส่งต่อไปยัง โมดูลสื่อสารไร้สาย ในส่วนของการทำงานฝั่งรับ Arduino UNO จะรับค่ามาจาก NRF24L01 แล้วประมวลผลเพื่อส่งต่อไปยังจอแสดงผลต่อไป

3.2.2 คุณสมบัติและการทำงานของโมดูล RFID (Radio-frequency identification)

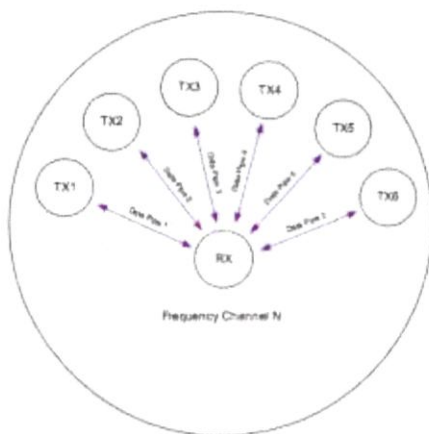
สามารถนำไปใช้แสดงตัวตนของมนุษย์ สัตว์ สินค้า และวัตถุ癖ในกระบวนการผลิต

ส่วนในที่นี่นำมาใช้ในการระบุตัวตนของบุคคล โดย อาร์เอฟไอดี Tag แต่ละ อันจะมีค่าแตกต่างกัน เมื่อ อาร์เอฟไอดี รับค่ามาจาก อาร์เอฟไอดี Tag ก็ส่งค่าที่ได้รับจาก อาร์เอฟไอดี Tag ไปยังไมโครคอนโทรลเลอร์ Arduino UNO ต่อไป

3.2.3 คุณสมบัติและการทำงานของโมดูลสื่อสารไร้สาย (NRF24L01)

โมดูลสื่อสารไร้สาย สามารถตั้งให้เป็นได้ทั้งตัวรับและตัวส่ง มีราคาถูก สามารถส่งสัญญาณได้ในระยะไกลและมีการส่งสัญญาณที่ดี มีเสาอากาศ

ส่วนในที่นี่นำมาใช้เป็นทั้งตัวรับและตัวส่งสัญญาณ โดยใช้โมดูลสื่อสารไร้สาย 2 ตัว โดยฝั่งส่งจะรับข้อมูลจากไมโครคอนโทรลเลอร์ จากนั้นจะส่งไปยังตัวรับ เมื่อตัวรับได้รับข้อมูลมาแล้วจะส่งข้อมูลที่ได้รับไปยังไมโครคอนโทรลเลอร์ เพื่อประมวลผลต่อไป



รูปที่ 3.2 ตัวอย่างการส่งสัญญาณของโมดูล อาร์เอฟไอดี

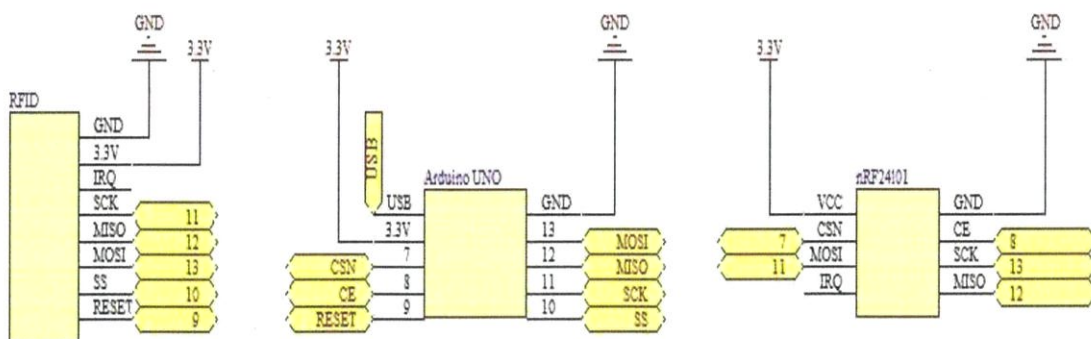
3.2.4 คุณสมบัติและการทำงานของโมดูลจอแสดงผล (Thin-film transistor)

โมดูลจอแสดงผลสามารถแสดงผลที่ความละเอียด 800x480 และแสดงผลเป็นสีได้

ส่วนในที่นี่นำมาใช้ในการแสดงผลและค่าที่ได้รับมาจากไมโครคอนโทรลเลอร์

3.3 การต่ออุปกรณ์โมดูลต่างๆเข้าด้วยกัน

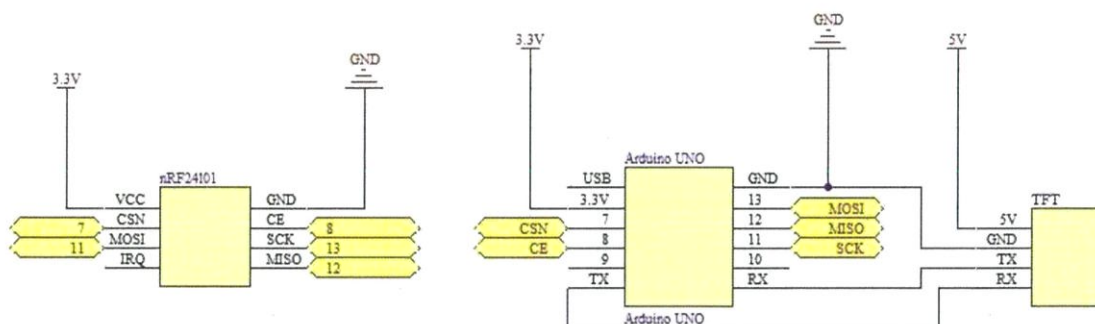
การต่อวงจรในฝั่งตัวส่งนั้นจะใช้แหล่งพลังงานจากแหล่งจ่ายพลังงาน 6.0 V 1000 mAh เป็นแหล่งพลังงานให้กับวงจร และมีลายวงจรดังรูปที่ 3.2



รูปที่ 3.3 วงจรตัวส่งที่ออกแบบไว้

อธิบายวงจรตัวส่ง

1. ต่อขา GND ของ Arduino ไปที่ GND ของโมดูล อาร์เอฟไอดี และ GND ของ โมดูลสื่อสารไร้สาย
2. ต่อขา 3.3 V จาก Arduino UNO ไปที่ VCC ของโมดูล RFID และ VCC ของ โมดูลสื่อสารไร้สาย
3. ต่อขา 7 จาก Arduino UNO ไปที่ขา CSN ของ โมดูลสื่อสารไร้สาย
4. ต่อขา 8 จาก Arduino UNO ไปที่ขา CE ของ โมดูลสื่อสารไร้สาย
5. ต่อขา 9 จาก Arduino UNO ไปที่ขา RESET ของโมดูล อาร์เอฟไอดี
6. ต่อขา 10 จาก Arduino UNO ไปที่ขา SS ของโมดูล อาร์เอฟไอดี
7. ต่อขา 11 จาก Arduino UNO ไปที่ขา SCK ของโมดูล อาร์เอฟไอดี และขา MOSI ของ nRF24L01
8. ต่อขา 12 จาก Arduino UNO ไปที่ขา MISO ของโมดูล, อาร์เอฟไอดี และ โมดูลสื่อสารไร้สาย
9. ต่อขา 13 จาก Arduino UNO ไปที่ขา MOSI ของโมดูล อาร์เอฟไอดี และขา SCK ของ โมดูลสื่อสารไร้สาย



รูปที่ 3.4 วงจรตัวรับที่ออกแบบไว้ 2

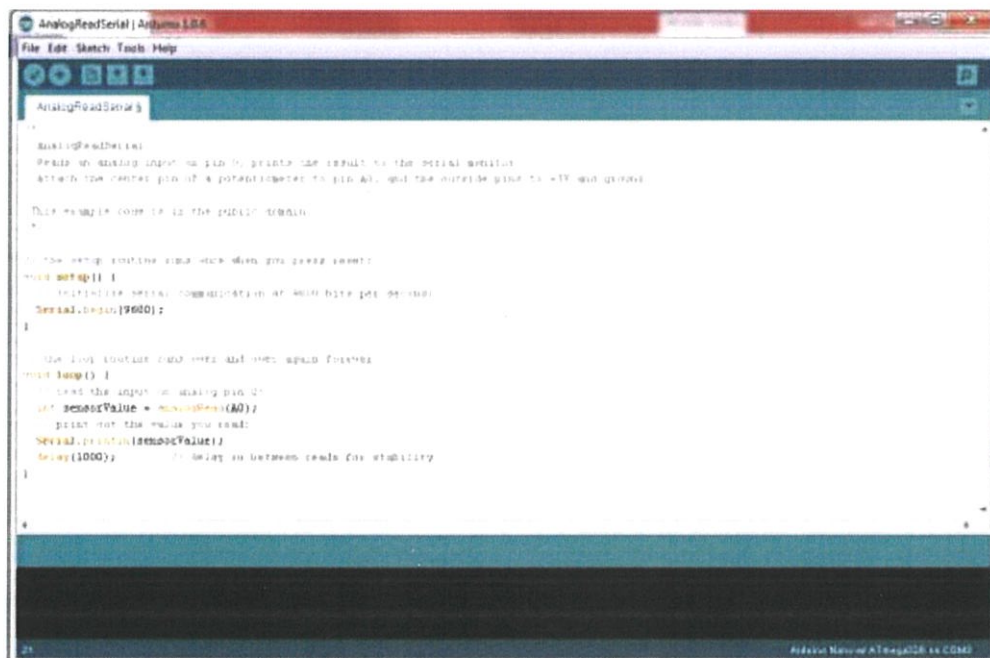
อธิบายวงจรตัวรับ

1. ต่อขา GND ของ Arduino ไปที่ขา GND ของ โมดูลสื่อสารไร้สาย และ GND ของ TFT
2. ต่อขา 3.3 V จาก Arduino UNO ไปที่ VCC ของ โมดูลสื่อสารไร้สาย และ VCC ของ TFT

3. ต่อขา 7 จาก Arduino UNO ไปที่ขา CSN ของ โมดูลสื่อสารไร้สาย
4. ต่อขา 8 จาก Arduino UNO ไปที่ขา CE ของ โมดูลสื่อสารไร้สาย
5. ต่อขา 11 จาก Arduino UNO ไปที่ขา SCK ของ โมดูลสื่อสารไร้สาย
6. ต่อขา 12 จาก Arduino UNO ไปที่ขา MISO ของ โมดูลสื่อสารไร้สาย
7. ต่อขา 13 จาก Arduino UNO ไปที่ขา MOSI ของ โมดูลสื่อสารไร้สาย
8. ต่อขา Tx จาก Arduino UNO ไปที่ขา Tx ของ TFT
9. ต่อขา Rx จาก Arduino UNO ไปที่ขา Rx ของ TFT

3.4 การออกแบบโปรแกรม (Program)

ใช้โปรแกรมตัวอย่างในโปรแกรม Arduino IDE ซึ่งเป็นโปรแกรมที่ใช้สำหรับอ่านค่าชุดข้อมูลอนาล็อก(AnalogReadSerial)สามารถนำโค้ดมาจาก File>Examples>Basics>AnalogReadSerial



```

AnalogReadSerial

//
// AnalogReadSerial.ino
// Reads an analog input on pin 0, prints the result to the serial monitor
// Attach the center pin of a potentiometer to pin A0, and the outside pins to +5V and ground
//
// This example code is in the public domain.
//

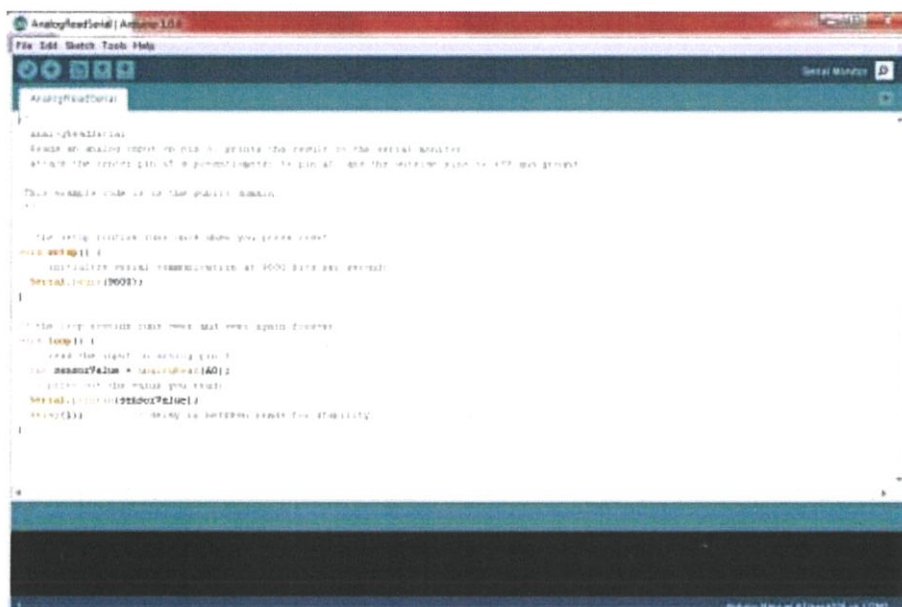
// Setup: initialize serial communication at 9600 bits per second:
void setup() {
  Serial.begin(9600);
}

// The loop routine runs over and over again forever:
void loop() {
  // Read the input on analog pin 0:
  int sensorValue = analogRead(A0);
  // Print out the value you read:
  Serial.println(sensorValue);
  delay(1000); // delay in between reads for stability
}

```

รูปที่ 3.5 และเมื่อกดเข้าไปจะปรากฏโค้ด (Code) ภาษา C ดังรูปที่ 3.3

หลังจากนั้นกด Serial Monitor ดังแสดงในรูปที่ 3.8 เพื่อทำการดูค่าอนาล็อกที่เซนเซอร์ส่งมา จะขึ้นเป็นหน้าต่าง



รูปที่ 3.8 การเปิดหน้าต่างเพื่อดูค่าอนาล็อกที่ส่งมาจากเซนเซอร์

บทที่ 4

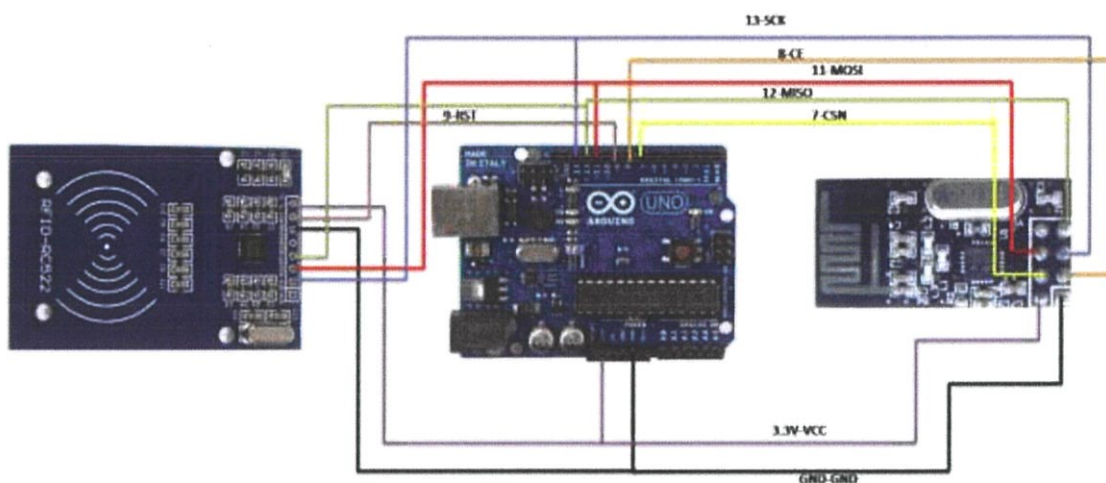
ผลการทดลอง

ในหัวข้อนี้จะกล่าวถึงรายละเอียดผลการทำงานของชิ้นงานและผลการทำงานของโปรแกรม และผลการทดลองโดยจะแบ่งเป็น 2 ส่วนคือ

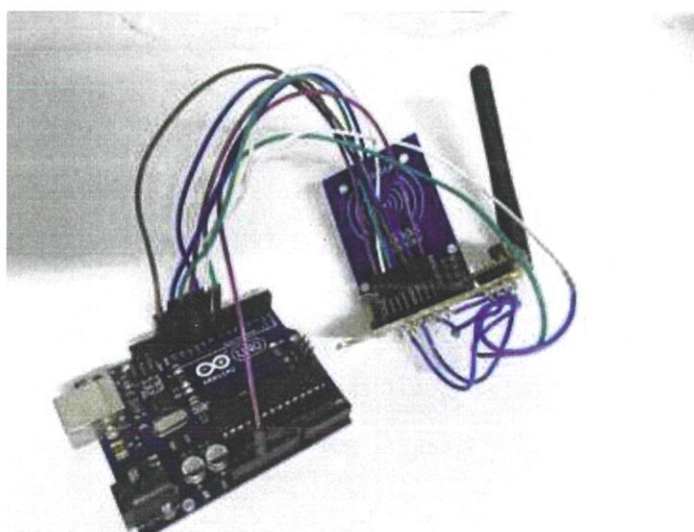
1. การทดลองวงจรในส่วนของภาครับและภาคส่ง
2. การแสดงผลของชิ้นงาน

4.1 การทดลองวงจรในส่วนของภาครับและภาคส่ง

ตัวอย่างการต่อวงจรในส่วนของภาคส่ง แสดงดังรูปที่ 4.1 และ รูปที่ 4.2







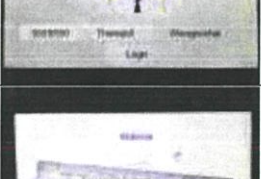



รูปที่ 4.1 วงจรในส่วนของภาคส่ง

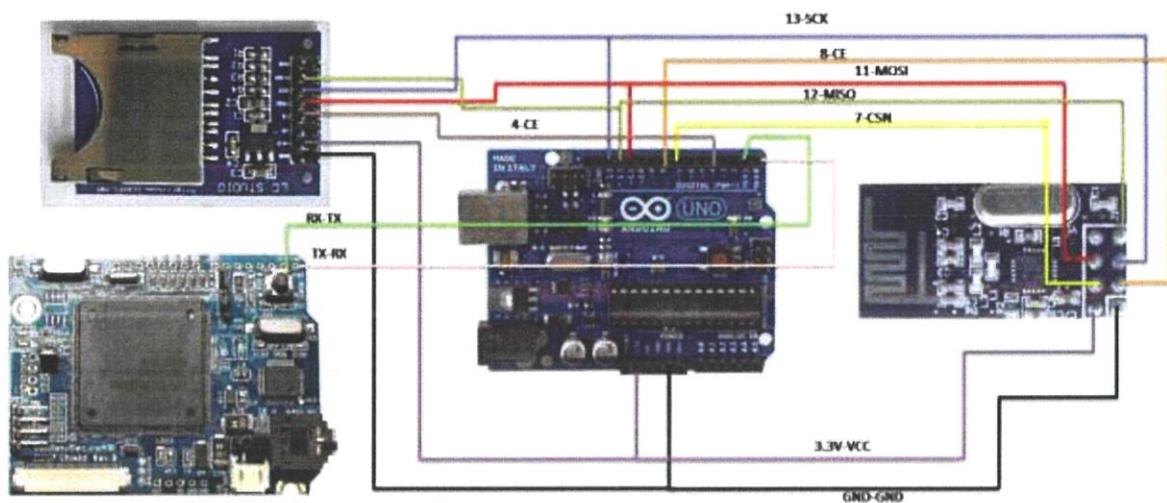


รูปที่ 4.2 ชิ้นส่วนภาคส่ง

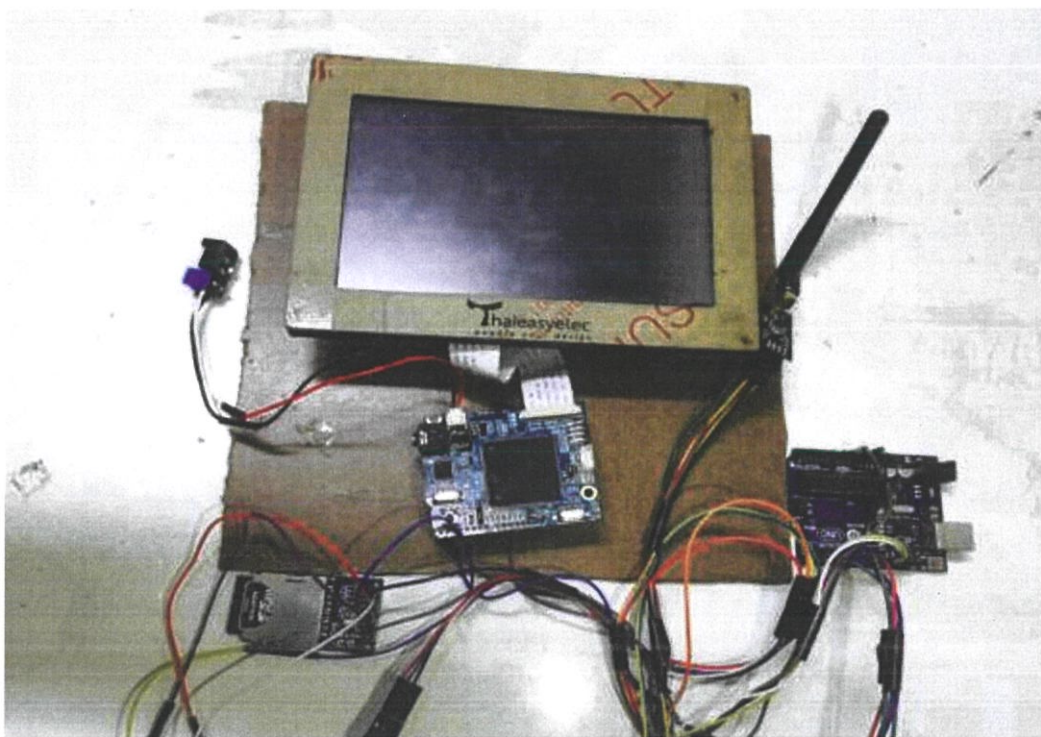
ตารางที่ 4.1 ตารางแสดงระยะทางการส่งสัญญาณของตัวส่งสัญญาณไร้สาย

ระยะทาง (m)	ค่าความหน่วงเวลา (s)	การแสดงผล
20	0.6	
25	0.8	
30	0.8	
35	1	
40	1	
45	1.3	
50	1.5	
55	-	

ตัวอย่างการต่อวงจรในส่วนของภาครับ แสดงดังรูปที่ 4.3 และรูปที่ 4.4

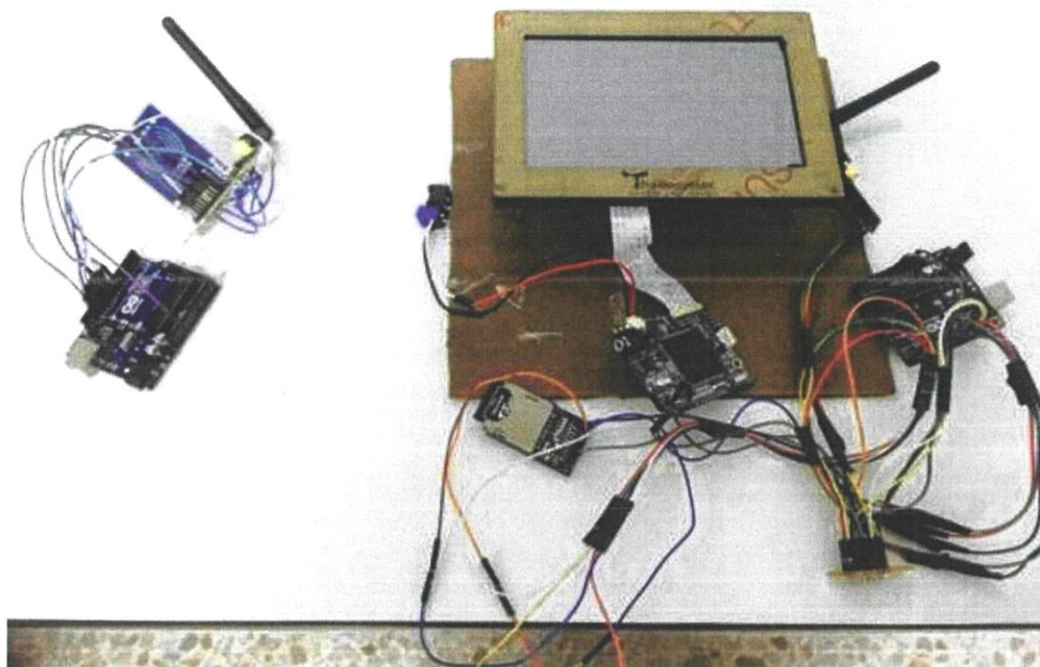


รูปที่ 4.3 วงจรในส่วนของภาครับ



รูปที่ 4.4 ชิ้นส่วนของภาครับ

เมื่อนำชิ้นส่วนทั้งสองมารวมจะได้ดังรูปที่ 4.5

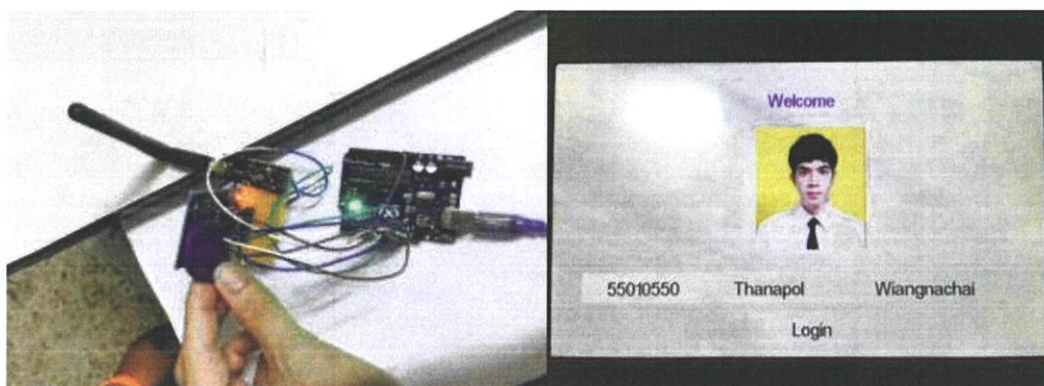


รูปที่ 4.5 ชิ้นงานทั้งสองส่วน

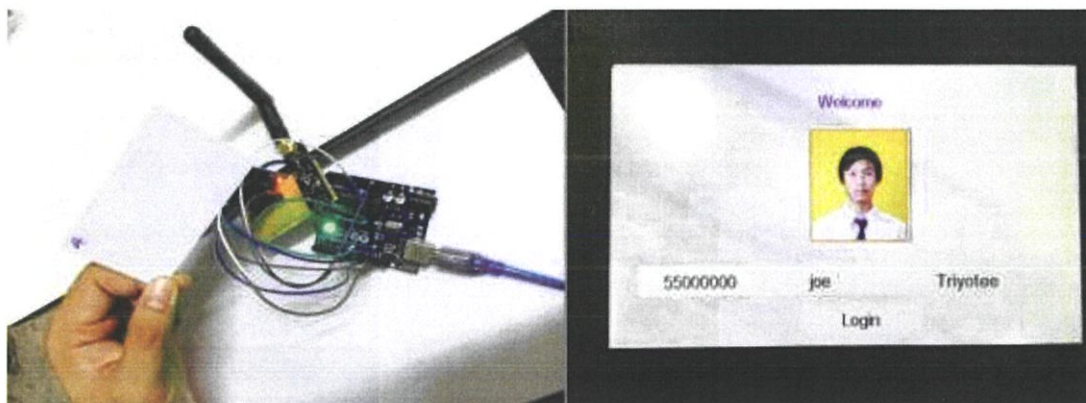
4.2 การแสดงผลของชิ้นงาน

ในส่วนของการแสดงผลนั้น จะแสดงผลก็ต่อเมื่อ ฝั่งรับได้รับข้อมูลจากตัวแท็กแล้วจะทำการประมวลผลผ่านไมโครคอนโทรลเลอร์ และทำการส่งสัญญาณไปยังภาครับเพื่อแสดงผลผ่านจอแสดงผล

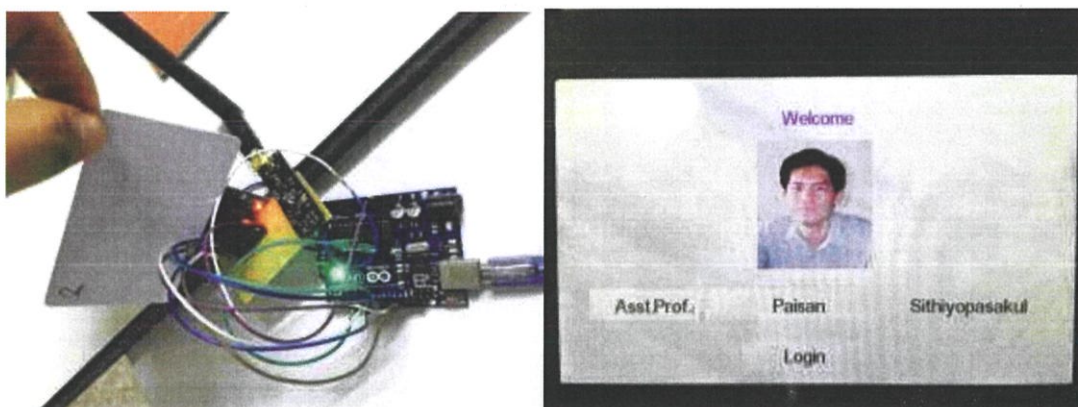
เมื่อได้รับข้อมูลจากตัวแท็กจะแสดงผลดังรูปที่ 4.6 รูปที่ 4.7 และรูปที่ 4.8



รูปที่ 4.6 แสดงผลเมื่อได้รับค่าแท็ก

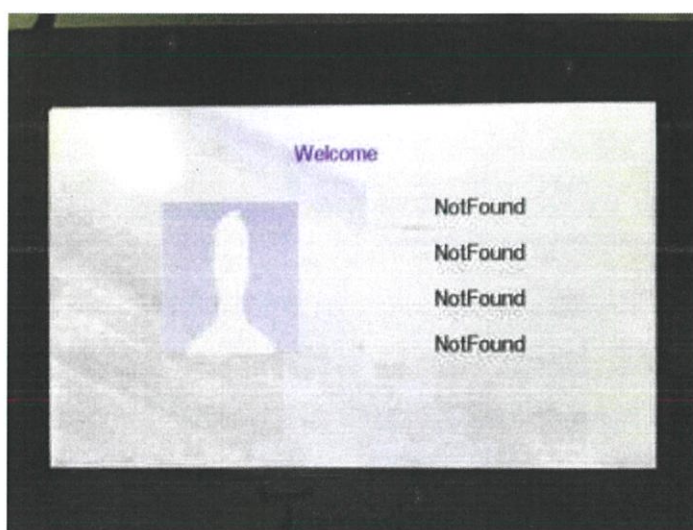


รูปที่ 4.7 แสดงผลเมื่อได้รับค่าแท้ก



รูปที่ 4.8 แสดงผลเมื่อได้รับค่าแท้ก

เมื่อเรานำแท็กที่ไม่มีข้อมูลมาตรวจสอบจะแสดงผลดังรูปที่ 4.9



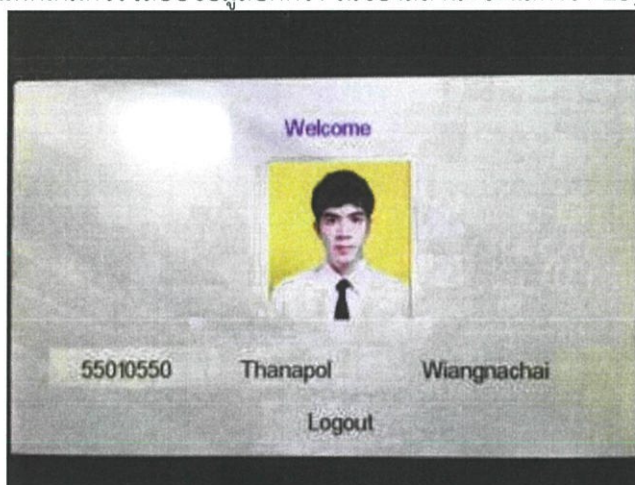
รูปที่ 4.9 แสดงผลเมื่อค่าแท้กไม่มีข้อมูล

เมื่อเราทำการตรวจสอบตัวแท็กในรอบแรกนั้น จะมีสถานะแสดงขึ้นที่ช่องด้านล่างว่า Login ดังรูปที่ 4.10



รูปที่ 4.10 แสดงสถานะ Login

เมื่อเรานำแท็กเดิมตรวจสอบข้อมูลอีกครั้ง ในช่องสถานะจะแสดงว่า Logout ดังรูปที่ 4.11



รูปที่ 4.11 แสดงสถานะ Logout

บทที่ 5

สรุปและวิจารณ์ผลการทดลอง

5.1 บทสรุปโครงการ

โครงการนี้นำเสนอในเรื่องของการ ออกแบบ และคิดค้น อุปกรณ์เพื่อใช้ในการ แสดงสถานะของ บุคคลขณะกำลังใช้ห้องหรืออาคาร โดยใช้หลักการของการตรวจความถี่ของคลื่นวิทยุ เมื่อนำป้ายความถี่ วิทยุมาตรวจจับกับตัวตรวจจับความถี่วิทยุแล้ว จะนำข้อมูลที่ตรวจจับได้ไปประมวลผลและทำการส่ง สัญญาณข้อมูลดังกล่าวไปยังภาครับและแสดงผลผ่านจอแสดงผล โดยในการทดลองการตรวจจับคลื่นวิทยุ และการส่งสัญญาณผ่านตัวส่งสัญญาณไร้สายนั้น สามารถรับค่าที่ได้มาแล้วนำไปประมวลผลแสดงว่าป้าย คลื่นความถี่วิทยุนี้เป็นของใครได้ แต่ยังมีข้อจำกัดในด้านระบบฐานข้อมูล ที่ยังไม่สามารถเพิ่มผู้ใช้และ กรอกข้อมูลได้โดยตรง

5.2 ปัญหาที่พบในระหว่างดำเนินงาน

1. ระยะเวลาในการทำชิ้นงานน้อย
2. สายไฟในการเชื่อมต่ออุปกรณ์ชำรุดง่าย
3. การส่งสัญญาณไร้สายไม่สามารถส่งผ่านกำแพงหนามากๆและหลายๆชั้นได้
4. จอแสดงผล แสดงสีของภาพได้ไม่คมชัดเนื่องจากสายไฟในการต่อชำรุด
5. ใช้อุปกรณ์หลายชิ้นในการเชื่อมต่อ
6. โค้ดของโปรแกรมมีปัญหาในการส่งงานจอภาพบ่อย

5.3 แนวทางแก้ไข และพัฒนาต่อ

1. สร้างระบบฐานข้อมูลทำให้สามารถเพิ่มหรือลบผู้ใช้ได้โดยตรง
2. เพิ่มระบบเสียงขณะตรวจจับป้ายความถี่วิทยุ
3. แสดงข้อมูลขณะมีผู้ใช้ห้องหรืออาคารให้ละเอียดขึ้น
4. เพิ่มโมดูลเวลาเพื่อบันทึกลงในตารางการเข้าออกห้อง

บรรณานุกรม

- [1]ArduinoAll (2557). Arduino Nano 3.0.[Online]. Available: <http://tinyurl.com/mbyteev>
- [2] Hangzhou weiku information & Technology Co.,Ltd. (2557). AVR ATmega328P. [Online].Available: http://datasheet.seekic.com/pdfimage/ATM/Atmel_ATMEGA168-20AU3.jpg
- [3] Seekic. (2557). MCU: AVR ATmega328P. [Online]. Available: <http://www.seekic.com/icdata/ATmega328P.html>
- [4] Seekic. (2557).Alcohol Gas Sensor: LM393 (MQ-3).[Online]. Available: <http://www.seekic.com/icdata/ATmega328P.html>
- [5] ArduinoAll.(2557). LM393. [Online]. Available: <http://tinyurl.com/kdgywjl>
- [6] Arduitrronics. (2558). HC-05. [Online]. Available: <http://www.arduitronics.com/product/55/bluetooth-serial-module-hc-05-master-slave-mode>
- [7] Edoardo De Marchi. (2558). HC-05Bluetooth. [Online]. Available: <https://developer.mbed.org/users/edodm85/notebook/HC-05-bluetooth/>
- [8] Datasheetdir. (2557). หลักการทำงานของเซนเซอร์. [Online]. Available: <http://www.datasheetdir.com/LM393AM+Comparators>
- [9] Robowiki. (2558).Robocode\Eclipse[Online]. Available: http://robowiki.net/wiki/Robocode/Eclipse/Create_a_Project
- [10] Gravitech.(2557). Arduino Startup kit บทที่ 3: เริ่มเขียนโปรแกรม[Online]. Available: <https://www.gravitechthai.com/guru2.php?p=260>
- [11] EC electronic. (2558). Tera Term[Online]. Available: http://www.ec.in.th/index.php?route=cms/article&article_id=7
- [12] Anuchit Chalothorn. (2558).มาใช้ Genymotion Android Emulator กัน. [Online]. Available: <http://goo.gl/BO0NV1>
- [13] Electronicaembajadores.(2257).ส่วนประกอบ MQ-3[Online]. Available:<http://www.electronicaembajadores.com/Productos/Detalle/1/SSGAMQ3/sensor-de-alcohol---mq-3>

ภาคผนวก ก.

คู่มือการใช้งานอุปกรณ์

วิธีการเพิ่มข้อมูลของผู้ใช้สำหรับตัวแท็ก

ให้นำ SD card จากฝั่งรับมาเสียบใส่คอมพิวเตอร์ จากนั้นเปิดไฟล์ Excel ขึ้นมา โดยที่ช่องแรกให้ใส่ตัวแท็กที่ต้องการ ช่องที่สองใส่ค่านำหน้า ช่องที่สามใส่ชื่อ ช่องที่สี่ใส่นามสกุล ช่องที่ห้า ใส่ไอทีรูปภาพ ช่องที่หกใส่ %

วิธีการลงโปรแกรม Sun7

ดาวน์โหลดโปรแกรมจาก

<http://www.thaieasyelec.com/downloads/SUN7/SUN7%20Studio%20Setup%20V1.36.exe> หลังจากนั้นเปิดตัวลงโปรแกรมเลือกที่อยู่ไฟล์จากนั้นก็ติดตั้ง และกดสำเร็จ

วิธีการเพิ่มรูปภาพผ่านโปรแกรม Sun7

เปิดโปรแกรมขึ้นมาจากนั้น คลิกที่ Image แล้วเลือกรูปภาพที่ต้องการ เลือกตำแหน่งภาพที่ต้องการแสดงลงบนจอ จากนั้นกดที่ Tools แล้วกด Generate

ภาคผนวก ข.

Source Code

Source Code ฝั่งส่ง

```
#include <SPI.h>
#include <RFID.h>
#include <nRF24L01p.h>

#define SS_PIN 10
#define RST_PIN 9

RFID rfid(SS_PIN, RST_PIN);
nRF24L01p transmitter(7,8);//CSN,CE

// Setup variables:
int serNum0;
int serNum1;
int serNum2;
int serNum3;
int serNum4;
int a;
int b;
void setup()
{
  Serial.begin(9600);
  SPI.begin();
  rfid.init();
  SPI.setBitOrder(MSBFIRST);
  transmitter.channel(90); // ตั้งช่องความถี่ให้ตรงกัน
  transmitter.TXaddress("ALL"); // ตั้งชื่อตำแหน่งให้ตรงกัน ชื่อตั้งได้สูงสุด 5 ตัวอักษร
  transmitter.init();
}

String message;
```

```

void loop()
{

    if (rfid.isCard()) {
        if (rfid.readCardSerial()) {
            // if (rfid.serNum[0] != serNum0 && rfid.serNum[1] != serNum1 &&
rfid.serNum[2] != serNum2 && rfid.serNum[3] != serNum3 && rfid.serNum[4] !=
serNum4 )
            // {
                if(rfid.serNum[0])
                {

                    Serial.println(" ");
                    Serial.println("Card found");
                    serNum0 = rfid.serNum[0];
                    serNum1 = rfid.serNum[1];
                    serNum2 = rfid.serNum[2];
                    serNum3 = rfid.serNum[3];
                    serNum4 = rfid.serNum[4];

                    if(serNum0 == 97 && serNum1 == 217)
                    {
                        transmitter.txPL("1"); // ค่าที่ต้องการส่ง
                        transmitter.send(FAST); // สั่งให้ส่งออกไป
                        Serial.println("1");
                        delay(500);
                    }
                    else if(serNum0 == 149 && serNum1 == 214)
                    {
                        transmitter.txPL("2"); // ค่าที่ต้องการส่ง
                        transmitter.send(FAST); // สั่งให้ส่งออกไป
                        Serial.println("2");
                        delay(500);
                    }
                }
            }
        }
    }
}

```

```
}  
    else if(serNum0 == 158 && serNum1 == 36)  
{  
    transmitter.txPL("3"); // ค่าที่ต้องการส่ง 158, 36  
    transmitter.send(FAST); // สั่งให้ส่งออกไป  
    Serial.println("3");  
    delay(500);  
}  
    else if(serNum0 == 46 && serNum1 == 49)  
{  
    transmitter.txPL("4"); // ค่าที่ต้องการส่ง  
    transmitter.send(FAST); // สั่งให้ส่งออกไป  
    Serial.println("4");  
    delay(500);  
}  
    else if(serNum0 == 30 && serNum1 == 215)  
{  
    transmitter.txPL("5"); // ค่าที่ต้องการส่ง  
    transmitter.send(FAST); // สั่งให้ส่งออกไป  
    Serial.println("5");  
    delay(500);  
}  
    else if(serNum0 == 94 && serNum1 == 68)  
{  
    transmitter.txPL("6"); // ค่าที่ต้องการส่ง  
    transmitter.send(FAST); // สั่งให้ส่งออกไป  
    Serial.println("6");  
    delay(500);  
}  
    else if(serNum0 == 142 && serNum1 == 181)  
{  
    transmitter.txPL("7"); // ค่าที่ต้องการส่ง
```

```
transmitter.send(FAST); // สั่งให้ส่งออกไป
Serial.println("7");
delay(500);
}

    else if (serNum0 == 254 && serNum1 == 111)
    {
        transmitter.txPL("8"); // ค่าที่ต้องการส่ง
        transmitter.send(FAST); // สั่งให้ส่งออกไป
        Serial.println("8");
        delay(500);
    }

    else if (serNum0 == 126 && serNum1 == 68)
    {
        transmitter.txPL("9"); // ค่าที่ต้องการส่ง
        transmitter.send(FAST); // สั่งให้ส่งออกไป
        Serial.println("9");
        delay(500);
    }

    else if (serNum0 == 110 && serNum1 == 112)
    {
        transmitter.txPL("10"); // ค่าที่ต้องการส่ง
        transmitter.send(FAST); // สั่งให้ส่งออกไป
        Serial.println("10");
        delay(500);
    }

    else if (serNum0 == 254 && serNum1 == 196)
    {
        transmitter.txPL("11"); // ค่าที่ต้องการส่ง
        transmitter.send(FAST); // สั่งให้ส่งออกไป
        Serial.println("11");
        delay(500);
    }

    else if (serNum0 == 62 && serNum1 == 79)
```

```
{
transmitter.txPL("12"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // สั่งให้ส่งออกไป
Serial.println("12");
delay(500);
}

    else if(serNum0 == 94 && serNum1 == 121)
{
transmitter.txPL("13"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // สั่งให้ส่งออกไป
Serial.println("13");
delay(500);
}

    else if(serNum0 == 110 && serNum1 == 224)
{
transmitter.txPL("14"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // สั่งให้ส่งออกไป
Serial.println("14");
delay(500);
}

else if(serNum0 == 174 && serNum1 == 119)
{
transmitter.txPL("15"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // สั่งให้ส่งออกไป
Serial.println("15");
delay(500);
}

else if(serNum0 == 110 && serNum1 == 120)
{
transmitter.txPL("16"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // สั่งให้ส่งออกไป
Serial.println("16");
delay(500);
}
```

```
}
else if(serNum0 == 126 && serNum1 == 162)
{
transmitter.txPL("17"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // สั่งให้ส่งออกไป
Serial.println("17");
delay(500);
}
else if(serNum0 == 62 && serNum1 == 37)
{
transmitter.txPL("18"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // สั่งให้ส่งออกไป
Serial.println("18");
delay(500);
}
else if(serNum0 == 174 && serNum1 == 121)
{
transmitter.txPL("19"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // สั่งให้ส่งออกไป
Serial.println("19");
delay(500);
}
else if(serNum0 == 142 && serNum1 == 73)
{
transmitter.txPL("20"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // สั่งให้ส่งออกไป
Serial.println("20");
delay(500);
}
else if(serNum0 == 14 && serNum1 == 49)
{
transmitter.txPL("21"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // สั่งให้ส่งออกไป
```

```
Serial.println("21");
delay(500);
}
else if(serNum0 == 46 && serNum1 == 162)
{
transmitter.txPL("22"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // สั่งให้ส่งออกไป
Serial.println("22");
delay(500);
}
```

//

//

```
else
{
transmitter.txPL("Not Found"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // สั่งให้ส่งออกไป
delay(500);
}
```

```
Serial.println("Cardnumber:");
Serial.print("Dec: ");
Serial.print(rfid.serNum[0],DEC);
Serial.print(", ");
Serial.print(rfid.serNum[1],DEC);
Serial.print(", ");
Serial.print(rfid.serNum[2],DEC);
```

```
Serial.print(", ");
Serial.print(rfid.serNum[3],DEC);
Serial.print(", ");
Serial.print(rfid.serNum[4],DEC);
Serial.println(" ");
```

```
} else {
```

```
if(serNum0 == 97 && serNum1 == 217)
{
    transmitter.txPL("1"); // ค่าที่ต้องการส่ง
    transmitter.send(FAST); // สั่งให้ส่งออกไป
    Serial.println("1");
    delay(500);
}
else if(serNum0 == 149 && serNum1 == 214)
{
    transmitter.txPL("2"); // ค่าที่ต้องการส่ง
    transmitter.send(FAST); // สั่งให้ส่งออกไป
    Serial.println("2");
    delay(500);
}
    else if(serNum0 == 158 && serNum1 == 36)
{
    transmitter.txPL("3"); // ค่าที่ต้องการส่ง 158, 36
    transmitter.send(FAST); // สั่งให้ส่งออกไป
    Serial.println("3");
    delay(500);
}
    else if(serNum0 == 46 && serNum1 == 49)
{
    transmitter.txPL("4"); // ค่าที่ต้องการส่ง
```

```
transmitter.send(FAST); // สั่งให้ส่งออกไป
Serial.println("4");
delay(500);

}

    else if(serNum0 == 30 && serNum1 == 215)
    {
transmitter.txPL("5"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // สั่งให้ส่งออกไป
Serial.println("5");
delay(500);
}

    else if(serNum0 == 94 && serNum1 == 68)
    {
transmitter.txPL("6"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // สั่งให้ส่งออกไป
Serial.println("6");
delay(500);
}

    else if(serNum0 == 142 && serNum1 == 181)
    {
transmitter.txPL("7"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // สั่งให้ส่งออกไป
Serial.println("7");
delay(500);
}

    else if(serNum0 == 254 && serNum1 == 111)
    {
transmitter.txPL("8"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // สั่งให้ส่งออกไป
Serial.println("8");
delay(500);
}
}
```

```
        else if(serNum0 == 126 && serNum1 == 68)
        {
            transmitter.txPL("9"); // ค่าที่ต้องการส่ง
            transmitter.send(FAST); // สั่งให้ส่งออกไป
            Serial.println("9");
            delay(500);
        }

        else if(serNum0 == 110 && serNum1 == 112)
        {
            transmitter.txPL("10"); // ค่าที่ต้องการส่ง
            transmitter.send(FAST); // สั่งให้ส่งออกไป
            Serial.println("10");
            delay(500);
        }

        else if(serNum0 == 254 && serNum1 == 196)
        {
            transmitter.txPL("11"); // ค่าที่ต้องการส่ง
            transmitter.send(FAST); // สั่งให้ส่งออกไป
            Serial.println("11");
            delay(500);
        }

        else if(serNum0 == 62 && serNum1 == 79)
        {
            transmitter.txPL("12"); // ค่าที่ต้องการส่ง
            transmitter.send(FAST); // สั่งให้ส่งออกไป
            Serial.println("12");
            delay(500);
        }

        else if(serNum0 == 94 && serNum1 == 121)
        {
            transmitter.txPL("13"); // ค่าที่ต้องการส่ง
            transmitter.send(FAST); // สั่งให้ส่งออกไป
            Serial.println("13");
```

```
delay(500);
}
    else if(serNum0 == 110 && serNum1 == 224)
    {
        transmitter.txPL("14"); // ค่าที่ต้องการส่ง
        transmitter.send(FAST); // สั่งให้ส่งออกไป
        Serial.println("14");
        delay(500);
    }
    else if(serNum0 == 174 && serNum1 == 119)
    {
        transmitter.txPL("15"); // ค่าที่ต้องการส่ง
        transmitter.send(FAST); // สั่งให้ส่งออกไป
        Serial.println("15");
        delay(500);
    }
    else if(serNum0 == 110 && serNum1 == 120)
    {
        transmitter.txPL("16"); // ค่าที่ต้องการส่ง
        transmitter.send(FAST); // สั่งให้ส่งออกไป
        Serial.println("16");
        delay(500);
    }
    else if(serNum0 == 126 && serNum1 == 162)
    {
        transmitter.txPL("17"); // ค่าที่ต้องการส่ง
        transmitter.send(FAST); // สั่งให้ส่งออกไป
        Serial.println("17");
        delay(500);
    }
    else if(serNum0 == 62 && serNum1 == 37)
    {
        transmitter.txPL("18"); // ค่าที่ต้องการส่ง
```

```
transmitter.send(FAST); // ส่งให้ส่งออกไป
Serial.println("18");
delay(500);
}
else if(serNum0 == 174 && serNum1 == 121)
{
transmitter.txPL("19"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // ส่งให้ส่งออกไป
Serial.println("19");
delay(500);
}
else if(serNum0 == 142 && serNum1 == 73)
{
transmitter.txPL("20"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // ส่งให้ส่งออกไป
Serial.println("20");
delay(500);
}
else if(serNum0 == 14 && serNum1 == 49)
{
transmitter.txPL("21"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // ส่งให้ส่งออกไป
Serial.println("21");
delay(500);
}
else if(serNum0 == 46 && serNum1 == 162)
{
transmitter.txPL("22"); // ค่าที่ต้องการส่ง
transmitter.send(FAST); // ส่งให้ส่งออกไป
Serial.println("22");
delay(500);
}
```

////////////////////////////////////

```
////////////////////////////////////
```

```
    else
    {
    transmitter.txPL("Not Found"); // ค่าที่ต้องการส่ง
    transmitter.send(FAST); // สั่งให้ส่งออกไป
    delay(500);
    }

    Serial.println("Cardnumber2:");
    Serial.print("Dec: ");
    Serial.print(rfid.serNum[0],DEC);
    Serial.print(", ");
    Serial.print(rfid.serNum[1],DEC);
    Serial.print(", ");
    Serial.print(rfid.serNum[2],DEC);
    Serial.print(", ");
    Serial.print(rfid.serNum[3],DEC);
    Serial.print(", ");
    Serial.print(rfid.serNum[4],DEC);
    Serial.println(" ");
    }
}

rfid.halt();
}
```

Source Code ฝั่งรับ

```
#include <SoftwareSerial.h>
#include <SPI.h>
#include <nRF24L01p.h>
#include <stdio.h>
#include <stdarg.h>
#include <SD.h>

nRF24L01p receiver(7,8);//CSN,CE
SoftwareSerial Serial_RFID(2, 3); // RX, TX RFID

String message = "";

String inputString = "";      // a string to hold incoming data
String inputRFID = "";       // a string to hold incoming data
boolean stringComplete = false; // whether the string is complete

uint8_t count_words;
uint8_t count_add;
uint8_t count_sel_tab;

String words[4] = "";        // save string

boolean flag_save = false;   // flag save string
boolean flag_setup = false;  // flag setup GUI

File myFile;

String check = "";
char buffa = NULL;
char buff2[4][20];
```

```
int login[60]={NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,
              NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,
              NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,
              NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,
              NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,
              NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL};
```

```
uint8_t id = 0, namee = 1, sname = 2, image = 3;
```

```
uint8_t i = 0,j = 0;
```

```
uint8_t z=0;
```

```
void setup()
```

```
{
```

```
    inputString.reserve(128);
```

```
    words[0].reserve(32);
```

```
    words[1].reserve(32);
```

```
    count_words = 0;
```

```
    count_add = 0;
```

```
    count_sel_tab = 0;
```

```
    SPI.begin();
```

```
    SPI.setBitOrder(MSBFIRST);
```

```
    receiver.channel(90); // ตั้งช่องความถี่ให้ตรงกัน
```

```
    receiver.RXaddress("ALL"); // ตั้งชื่อตำแหน่งให้ตรงกัน ชื่อตั้งได้สูงสุด 5 ตัวอักษร
```

```
    receiver.init();
```

```
    Serial.begin(9600);
```

```
    while (!Serial) {
```

```
        ;
```

```
    }
```

```

Serial.print(F("Initializing SD card..."));

if (!SD.begin(4)) {
  Serial.println(F("initialization failed!"));
  return;
}
Serial.println(F("initialization done.));

//selectname('1');

}

void loop()
{
  clearbuff2();
  if(receiver.available()){
    //Serial.println(F("asdadsdas"));
    receiver.read(); // สั่งให้เริ่มอ่าน
    receiver.rxPL(message); // สั่งให้อ่านเก็บไว้ที่ตัวแปร
    Serial.println(message);

    if(message)
    {
      if(selectname(message))
      {
        Serial.println(F("GUIGotoScr SCS2"));
        delay(350);
        Serial.print(F("GUIChangelmg pp1 "));
        Serial.println(buff2[image]);

        Serial.println(F("GUIClrTxt txt_id"));
      }
    }
  }
}

```

```
Serial.println(F("GUIClrTxt txt_name"));
Serial.println(F("GUIClrTxt txt_sname"));
Serial.println(F("GUIClrTxt txt_status"));
```

```
Serial.print(F("GUIAddTxt txt_id "));
Serial.println(buff2[id]);
Serial.print(F("GUIAddTxt txt_name "));
Serial.println(buff2[namee]);
Serial.print(F("GUIAddTxt txt_sname "));
Serial.println(buff2[sname]);
```

```
z=checklogin(message);
if(z==0)
{
    Serial.println(F("GUIAddTxt txt_status Logout"));
    delay(2500);
    Serial.println(F("GUIGotoScr SCS0"));
}
else if(z==1)
{
    Serial.println(F("GUIAddTxt txt_status Login"));
    delay(2500);
    Serial.println(F("GUIGotoScr SCS0"));
}
else
{
    Serial.println(F("GUIAddTxt txt_status CannotLogin"));
    delay(2500);
    Serial.println(F("GUIGotoScr SCS0"));
}
}
else
```

```

{
  Serial.println(F("GUIGotoScr SCS3"));
  delay(200);
  Serial.println(F("GUIChangelmg pp1 3"));
  Serial.println(F("GUIClrTxt txt_id"));
  Serial.println(F("GUIClrTxt txt_name"));
  Serial.println(F("GUIClrTxt txt_sname"));
  Serial.println(F("GUIClrTxt txt_status"));

  Serial.println(F("GUIAddTxt txt_id NotFound"));
  Serial.println(F("GUIAddTxt txt_name NotFound"));
  Serial.println(F("GUIAddTxt txt_sname NotFound"));
  Serial.println(F("GUIAddTxt txt_status NotFound"));

  delay(2500);
  Serial.println(F("GUIGotoScr SCS0"));

}

}

message="";
}

////////////////////////////////////
if (stringComplete) {
// split string
for(int i=0; i<inputString.length(); i++) {
  if(inputString.charAt(i) != ' ')
    words[count_words] += inputString.charAt(i);
  else
    count_words++;
}
}

```

```

    }
    // check command <SETUPGUI>
    if (words[0].equals(F("SETUPGUI"))) {

        Serial.println(F("GUIConfigTxt SCS0 txt_id 1"));
        Serial.println(F("GUIConfigTxt SCS0 txt_name 1"));
        Serial.println(F("GUIConfigTxt SCS0 txt_sname 1"));
        Serial.println(F("GUIConfigTxt SCS0 txt_status 1"));
        //Serial.println("GUIConfigTab SCS0 tab_rfid 1");
        Serial.println(F("GUIDisableImg pp1"));
        //Serial.println("GUIDisableImg pp2");

    }

    count_words = 0;
    words[0] = "";
    words[1] = "";
    inputString = "";
    stringComplete = false;
}

}

void serialEvent() {
    while (Serial.available()) {
        char inChar = (char)Serial.read();
        // check and save string to inputString
        if(inChar == '>') {
            flag_save = false;
            stringComplete = true;
        }
        else if (inChar == '<') {

```

```
    flag_save = true;
}

if(flag_save == true) {
    if (inChar != '<') {
        inputString += inChar;
    }
}
}
```

```
bool selectname(String num)
{
    bool flagup = false;
    myFile = SD.open(F("testex2.csv"));
    if(myFile)
    {
        Serial.println(F("Start"));

        while(myFile.available())
        {
            while((buffa=myFile.read())!=';')
            {
                check += buffa; //read row key
            }
            while((buffa=myFile.read())!='%') //check end row
            {
                if(check.toInt() == num.toInt())
                {
                    while(buffa!=';')
                    {
                        buff2[i][j] = buffa;
                        buffa = myFile.read();
                    }
                }
            }
        }
    }
}
```

```

        j++;
    }
    i++;
    j=0;
    flagup = true;
}
}
i=0;
myFile.read(); //newline char
myFile.read(); //newline char
}
myFile.close();
return flagup;
}

else
{
    Serial.println(F("error opening test.txt"));
    return false;
}
}

```

```

uint8_t checklogin(String num)
{
    int a=0;
    for(a=0;a<100;a++)
    {
        if(num.toInt()==login[a])
        {
            login[a]=NULL;
            return 0;
        }
    }
}

```

```
for(a=0;a<100;a++)
{
    if(login[a]==NULL)
    {
        login[a]=num.toInt();
        return 1;
    }
}

return 2;
}

void clearbuff2()
{
    for(int r=0; r<4; r++)
    {
        for(int t=0; t<20; t++)
        {
            buff2[r][t]=NULL;
        }
    }
}
```