4

พื้นฐานของ Arduino

4.1 ความรู้เบื้องต้นเกี่ยวกับ Arduino

Arduino ถูกคิดค้นขึ้นที่ Ivrea Interaction Design Institute มันถูกออกแบบมาสำหรับ การสร้างต้นแบบอย่างรวดเร็ว โดยบุ่งเป๋าไปที่นักอดิเรกโดยไม่ต้องมีพื้นฐานการเขียนโปรแก รมใดๆ ในไม่ช้า แพลตฟอร์มที่ใช้งานง่ายดึงดูดผู้ชมที่ครอบคลุมชุมชนในวงกว้าง และเริ่ม เปลี่ยนแปลงเพื่อปรับแนวโน้มล่าสุดในตลาด ตั้งแต่บอร์ด 8 บิตไปจนถึงผลิตภัณฑ์ IoT อุปกรณ์สวมใส่ และสภาพแวดล้อมแบบผังตัว บอร์ด Arduino เป็นโอเพ่นซอร์สอย่างสมบูรณ์ และสามารถใช้สำหรับการพัฒนาแอปพลิเคชันที่มีความต้องการเฉพาะ ซอฟต์แวร์ Arduino นั้นใช้งานง่ายและง่ายต่อการเริ่มต้นด้วยสภาพแวดล้อมที่ยืดหยุ่นสำหรับผู้ใช้ขั้นสูง สามารถใช้ งานได้บนแพลตฟอร์ม Mac, Linux และ Window Arduino สามารถเรียนรู้สิ่งใหม่ได้

ข้อดีของ Arduino:

ค่าใช้จ่าย:บอร์ด Arduino มีราคาถูกกว่าเมื่อเทียบกับไมโครอื่น ๆ บอร์ดควบคุม

แพลตฟอร์ม:ซอฟต์แวร์ Arduino (IDE) เข้ากันได้กับ ระบบปฏิบัติการ เช่น Macintosh OSX, Windows และ Linux

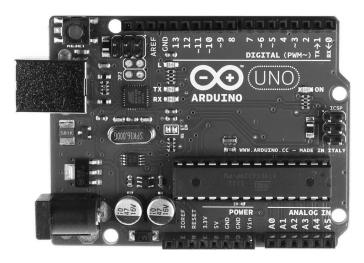
เป็นมิตรกับผู้ใช้:ซอฟต์แวร์ Arduino (IDE) ใช้งานง่ายและง่ายต่อการ เริ่มต้นและมีความยืดหยุ่นสำหรับโปรแกรมเมอร์ที่มีทักษะ

โอเพ่นซอร์ส:Arduino เป็นซอฟต์แวร์โอเพ่นซอร์สที่สามารถ โปรแกรมด้วยภาษา C, C++ หรือ AVR-C ผู้ใช้จึงสามารถออกแบบโมดูลได้ หลากหลาย

4.1.1 Arduino Uno

Arduino/Genuino Uno มีไมโครคอนโทรลเลอร์ ATmega328 ออนบอร์ด มีพอร์ตอินพุต แบบอะนาล็อก 6 พอร์ต (A0–A5) และพอร์ต I/O ดิจิตอล 14 พอร์ต โดย 6 พอร์ตเป็นพิน PWM แต่ละพินสามารถทำงานด้วยแรงดันไฟฟ้า 0–5 V ทำงานที่ความถี 16 MHzรูปที่4.1 แสดงบอร์ด Arduino Uno (ตาราง 4.1).





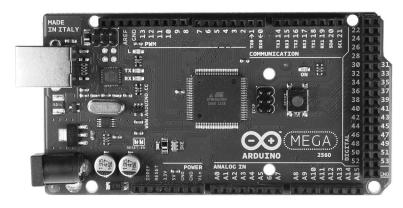
รูปที4.1 บอร์ด Arduino Uno

ตาราง 4.1 คำอธิบายพินของ Arduino UNO

เข็มหมุด	คำอธิบาย
วิน	เป็นแรงดันไฟภายนอกที่จ่ายให้กับบอร์ด 3.3
3.3 V	V บนบอร์ด
+ 5 V	แรงดันไฟขาออก +5 V
GND	กราวด์
ไอโอเรฟ	คือการเลือกแหล่งพลังงานที่เหมาะสมโดยให้แรงดันไฟ อ้างอิง
ซีเรียล	สามารถส่งและรับข้อมูลแบบอนุกรมด้วย 0 (Rx) 1 (Tx) ทริกเกอร์
การขัดจังหวะภายนอก	การขัดจังหวะด้วยค่าต่ำ (พิน 2 และ 3)
PWM	8 บิต หก PWM (3, 5, 6, 9, 10, 11)
SPI	รองรับการสือสาร SPI [10 (SS), 11 (MOSI), 12 (MISO) และ 13 (SCK)]
นำ	ขับเคลือนด้วย LED ในตัว
TWI	การสือสาร TWI [A4 (SDA) และ A5 (SCL)] แรงดัน
AREF	อ้างอิงกับอินพุตแบบอะนาล็อก
รีเซ็ต	ใช้สำหรับรีเซ็ตไมโครคอนโทรลเลอร์ออนบอร์ด

4.1.2 Arduino Mega

Arduino Mega มีไมโครคอนโทรลเลอร์ ATmega2560 ออนบอร์ด มีอินพุตแบบอะนาล็อก 16 ช่อง, I/O ดิจิตอล 54 ช่อง, การเชื่อมต่อ USB, 4 UART, ช่องเสียบไฟ และปุ่มรีเซ็ต ทำงานบนความถี่ 16 MHzรูปที่ 4.2แสดงบอร์ด Arduino Mega (ตาราง 4.2).



รูป 4.2 บอร์ด Arduino Mega

ตาราง 4.2 คำอธิบายพิน

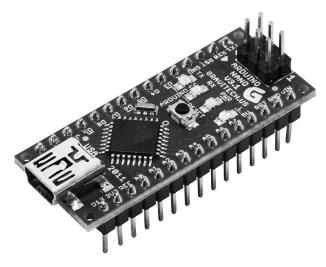
เข็มหมุด	คำอธิบาย
- วิน	แรงดันไฟฟ้าภายนอกไปยังบอร์ด Arduino
+ 5 V	เอาต์พุต 5 V . ที่มีการควบคุม
3.3 V	ออนบอร์ด 3.3 V แหล่งจ่า
GND	ยกราวด์
ไอโอเรฟ	คือการเลือกแหล่งพลังงานที่เหมาะสมโดยให้แรงดันไฟ อ้างอิง
Serial0	สามารถส่งและรับข้อมูลอนุกรมที่มี 0 (Rx) และ 1 (Tx) สามารถส่งและรับ
Serial1	ข้อมูลอนุกรมที่มี 19 (Rx) และ 18 (Tx) สามารถส่งและรับข้อมูลอนุกรมที่มี
Serial2	14 (Rx) และ 16 (Tx))
การขัดจังหวะภายนอก	มันทริกเกอร์การขัดจังหวะภายนอกที่ค่าต่ำด้วย 2 (ขัดจังหวะ 0) 3 (อินเตอร์รัปต์ 1), 18 (อินเตอร์รัปต์ 5), 19 (อินเตอร์รัปต์ 4) และ 20 (อินเตอร์รัปต์ 2)
PWM	8 บิต PWM (พิน: 2–13 และ 44–46)
SPI	รองรับการสือสาร SPI [10 (SS), 11 (MOSI), 12 (MISO), และ 13 (SCK)]
นำ	LED ขับเคลือนที่ขา 13
TWI	รองรับการสือสาร TWI [พิน: 20 (SDA), 21 (SCL)] เป็นแรงดัน
AREF	อ้างอิงสำหรับอินพุตแบบอะนาล็อก
รีเซ็ต	ใช้สำหรับรีเซ็ตไมโครคอนโทรลเลอร์บนบอร์ด

4.1.3 Arduino นาโน

Arduino/Genuino Nano มีไมโครคอนโทรลเลอร์ ATmega328 ออนบอร์ด มีอินพุตแบ บอะนาล็อก 8 ช่อง, พอร์ต I/O ดิจิตอล 14 พอร์ต และ PWM 6 ช่อง มีหน่วยความจำแฟลช ออนบอร์ด 32 KB, EEPROM 1 KB, SRAM 2 KB และทำงานที่ความถี่ 16 MHzรูปที่4.3 แสดง Arduino Nano (ตาราง 4.3และ4.4).

32

Internet of Things กับ Raspberry Pi และ Arduino



รูป 4.3 บอร์ด Arduino นาโน

ตาราง 4.3 คำอธิบายพินของ Arduino NANO

เข็มหมุด	คำอธิบาย		
	แรงดันไฟภายนอกเข้าบอร์ด เอาต์พุตเป็น		
+ 5 V	+5 V		
3.3 V	การจ่ายไฟ 3.3 V บนกราวด์		
GND			
ไอโอเรฟ	ช่วยในการเลือกแหล่งพลังงานที่เหมาะสมโดยการจัดหา a อ้างอิงแรงดันไฟฟ้า		
ซีเรียล	สามารถส่งและรับข้อมูลอนุกรมด้วย 0 (Rx) และ 1 (Tx) ทริกเกอร์การ		
การขัดจังหวะภายนอก	ขัดจังหวะด้วยค่าต่ำ (พิน 2 และ 3)		
PWM	8 บิต PWM (3, 5, 6, 9, 10, 11)		
SPI	รองรับการสื่อสาร SPI กับ [10 (SS), 11 (MOSI), 12 (MISO) และ 13 (SCK)]		
นำ	LED ขับเคลือนที่ขา 13		
I2C	รองรับการเชื่อมต่อสองสาย [A4 (SDA) และ A5 (SCL)] เป็นแรงดัน		
AREF	อ้างอิงสำหรับอินพุตแบบอะนาล็อก		
รีเซ็ต	ใช้สำหรับรีเซ็ตไมโครคอนโทรลเลอร์บนบอร์ด		

ตาราง 4.4 ตารางเปรียบเทียบสำหรับบอร์ด Arduino บางตัว

ชื่อ	โปรเซสเซอร์	ซีพียู ความเร็ว	ปฏิบัติการ/ _{ป๋อนข้อมูล} แรงดันไฟฟ้า	ดิจิทัลไอโอ/ PWM	อนาล็อก เข้า/ออก	UART	แฟลช [kB]
LilyPad	ATmega168V ATmega328P	8 MHz	2.4–5.5 โวลต์/ 2.4–5.5 V	14/6	6/0	— 1	16
เมก้า 2560	ATmega2560	16 MHz	5 โวลต์/4-12 โวลต์	54/15	16/0	4	256
ไมโคร	ATmega32U4	16 MHz	5 โวลต์/4-12 โวลต์	20/4	12/0	1	32
อูโนะ	ATmega328P	16 MHz	5 โวลต์/4–12 โวลต์	14/6	6/0	1	32
เลโอนาร์โด	ATmega32U4	16 MHz	5 โวลต์/4-12 โวลต์	20/4	12/0	1	32
ยุน	ATmega32U4 AR9331 Linux	16 MHz 400 MHz	5 V	20/4	12/0	1	32
อีเธอร์เน็ต	ATmega328P	16 MHz	5 โวลต์/4-12 โวลต์	14/4	6/0	— 3	32
เจมม่า	ATtiny85	8 MHz	3.3 โวลต์/ 4-16 วี	3/2	1/0	— 8	3
MKRZero	SAMD21 Cortex-M0+ ตำ 32 บิต พลัง ARM MCU	48 MHz	3.3 V	22/12	4 (ADC 8/10/ 12 ບົຕ)/1 (DAC 10 ບົຕ)	1	256

4.2 Arduino IDE

Arduino integrated development environment (IDE) เป็นซอฟต์แวร์โอเพ่นซอร์ส และทำให้ง่ายต่อการเขียนโค้ดและอัปโหลดไปยังมอร์ด

4.2.1 ขั้นตอนในการติดตั้ง Arduino IDE

ขันตอนที่ 1: ติดตั้ง Arduino IDE และเปิดหน้าต่าง

ในการเริ่มต้น ให้ติดตั้ง Arduino IDEรูปที่4.4แสดงหน้าต่างของ Arduino IDE

ข้นตอนที่ 2: เลือกเวอร์ชันของบอร์ด Arduino

Arduino มีหลายเวอร์ชัน เช่น UNO, MEGA, NANO เป็นต้น ก่อนเริ่ม-ในโครงการค้นหารุ่นที่เหมาะสมโดยเลือกพารามิเตอร์ตามความต้องการ บอร์ดทั่วไป สำหรับผู้เริ่มต้นคือ Arduino UNO เลือกบอร์ดและพอร์ตอนุกรม

Internet of Things กับ Raspberry Pi และ Arduino

ฝน ปอ. อ นั้น

```
**Partial Marker Lass Fried Easten Note Help **

**Total Control Help **

**Void setup() {

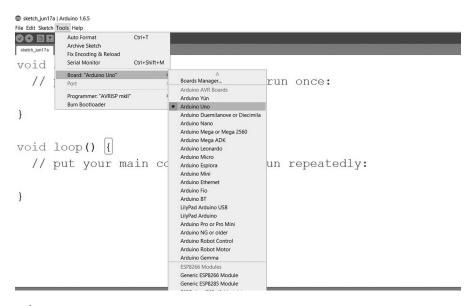
// put your setup code here, to run once:

}

**Void loop() {

// put your main code here, to run repeatedly:
}
```

รูป 4.4 หน้าต่าง Arduino IDE



รูปที4.5 การเลือกบอร์ด Arduino

ใน Arduino IDE ในการเลือกบอร์ด Arduino ให้คลิกที่ "เครื่องมือ" จากนั้นคลิกที่ "บอร์ด"รูปที่4.5แสดงการเลือก "Arduino Uno"

ข้นตอนที่ 3: เขียนและคอมไพล์โปรแกรม

เขียนโปรแกรมในหน้าต่าง Arduino IDE จากนั้น "เรียกใช้" โปร-กรัม.รูปที่4.6แสดงหน้าต่างเพื่อคอมไพล์โปรแกรม

```
© sterch_jun17a | Archino 1.65
File Edit Sterch Tools Help

✓ ○ □ □ □ veriv

stetch_jun17a

Void setup() {
    // put your setup code here, to run once:
}

void loop() {
    // put your main code here, to run repeatedly:
}
```

รูป 4.6 รวบรวมโปรแกรม

ข้นตอนที่ 4: เชื่อมต่อ Arduino กับ PC

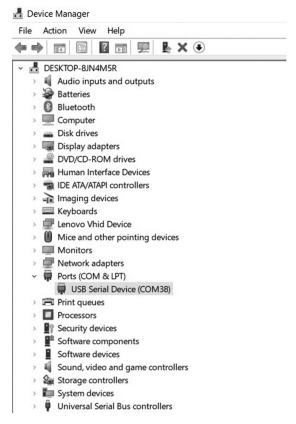
เชือมต่อ Arduino กับพอร์ต USB ของพีซีด้วยสาย USB ทั้งหมด บอร์ด Arduino มีที่อยู่พอร์ตอนุกรมที่แตกต่างกัน (COM2, COM4 ฯลฯ) ดังนั้น จึงจำเป็นต้องกำหนดค่าพอร์ตใหม่สำหรับ Arduino แต่ละตัวและเลือกใน IDE ใน การตรวจสอบพอร์ตทีเชือมต่อ Arduino ให้คลิกขวาที "PC" จากนั้นเลือก "manager"; หน้าต่างจะเปิดขึ้น จากนั้นดับเบิลคลิกที่ "ตัวจัดการอุปกรณ์" หน้าต่างตามที่แสดงในรูปที่4.7จะเปิด คลิกที่พอร์ต COM และ LPT และพบพอร์ตที่ อุปกรณ์เชือมต่ออยู่

ตอนนี้คลิกที่หัว "เครื่องมือ" ที่หน้าต่าง Arduino IDE ไปที่ พอร์ตและเลือกหมายเลขพอร์ตเดียวกัน ซึ่งพบได้ในโปรแกรมจัดการอุปกรณ์ (เลือก COM1 หรือ COM2 เป็นต้น)รูปที่ 4.8แสดง "COM38" เป็นพอร์ตอนุกรมของ บอร์ด

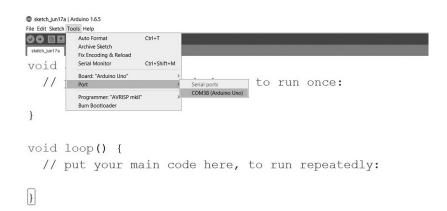
ข้นตอนที่ 5: อัปโหลดโปรแกรมไปยังบอร์ด Arduino

อัปโหลดโปรแกรมไปยังบอร์ด Arduinoรูปที่4.9แสดงวิธีการ ลงโปรแกรม.

Internet of Things กับ Raspberry Pi และ Arduino



รูปที่ 4.7 หน้าต่างตรวจสอบพอร์ต Arduino



รูป 4.8 พอร์ตอนุกรมของบอร์ด

ฝน ปอ. อ นั้น

เบเบ , เบ เบเบ , เบ เบ , เบอินเทอร์เน็ตของสรรพสิ่ง (IoT)

```
Percention of the feet of the
```

รูป 4.9 หน้าต่างสำหรับอัพโหลดโปรแกรม

4.3 คำสั่งพื้นฐานสำหรับ Arduino

- 1.**pinMode(x, เอาต์พุต);**//กำหนดหมายเลขพิน x เป็นพินเอาต์พุต โดยที่ x คือจำ นวนพินดิจิทัล
- 2.**digitalWrite(x, สูง);**//เปิดหมายเลขพิน x เป็น HIGH หรือ ON โดยที่ x คือ หมายเลขพินดิจิทัล
- 3.**pinMode(x, อินพุต);**//กำหนดหมายเลขพิน x เป็นพินอินพุต โดยที่ x คือจำนว นพินดิจิทัล
- 4.**digitalRead (พินดิจิตอล);**//อ่านพินดิจิตอลเช่น 13 หรือ 12 หรือ 11 เป็นต้น
- 5.**analogRead (ขาอะนาล็อก);**//อ่านพินอะนาล็อกเช่น A0 หรือ A1 หรือ A2 เป็นต้น

4.4 คำสั่ง LCD

- 1.**lcd.begin(16, 2);**//เริ่มต้น LCD 16*2 หรือ 20*4
- 2.lcd.print("ราชา");//พิมพ์สตริง "RAJESH" บน LCD
- 3.**lcd.setCursor(x, y);**//ตั้งค่าเคอร์เซอร์ของ LCD ในตำแหน่งที่ต้องการ โดยที่ x คือจำนวน COLUMN และ y
- 4.lcd.print(LPU);//พิมพ์ LPU เป็นจำนวนเต็มบน LCD
- 5.จอแอลซีดีที่ชัดเจน();//ล้างเนื้อหาของ LCD

Internet of Things กับ Raspberry Pi และ Arduino

4.5 คำสั่งการสื่อสารแบบอนุกรม

- 1.**Serial.begin(รับส่งข้อมูล)**;//เริ่มต้นการสือสารแบบอนุกรมเพื่อกำหนดอัตราบอด เป็น 600/1200/2400/4800/9600
- 2.**Serial.print("ราเจช");**//สตริงคงที่สำหรับการพิมพ์แบบอนุกรมพร้อมกำหนด อัตราบอดบน Tx line
- 3.**Serial.println("ราเจช");**//สตริงคงที่สำหรับการพิมพ์แบบอนุกรมพร้อมกำหนด อัตราบอดและป้อนคำสั่งบน Tx line
- 4.**Serial.print ("LPU");**//สตริง int พิมพ์แบบอนุกรมพร้อมกำหนดอัตราบอดบน Tx line
- 5.**Serial.print ("LPU");**//สตริง int พิมพ์แบบอนุกรมพร้อมกำหนดอัตราบอดและ ป้อนคำสั่งบน Tx line
- 6.**อนุกรม.เขียน (BYTE);**//serial โอนหนึ่งไบต์บน Tx line
- 7.Serial.read();//อ่านหนึ่งไบต์ซีเรียลจาก Rx line

4.6 เล่นกับ LED และ Arduino

ไดโอดเปล่งแสง (LED) เป็นอุปกรณ์ที่สามารถใช้เป็นตัวบ่งชื่ได้ LED มีข้ว 2 ข้ว คือ แอโนดและ แคโทด ไฟ LED มีให้เลือกหลายสีรูปที่ 4.10แสดง LED

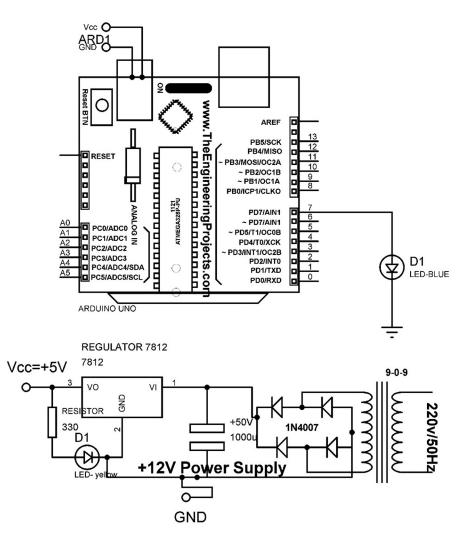


รูป 4.10 ไดโอดเปล่งแสง

สามารถใช้สีต่างๆ เพื่อแสดงเงื่อนไขต่างๆ ได้ สีของ LED เกิดจากการเปล่งแสงในบริเวณ เฉพาะของสเปกตรัมแสงที่มองเห็นได้จากสารประกอบต่างๆ

เพื่อให้เข้าใจการทำงานของ LED ให้เชื่อมต่อข้วบวกของ LED กับขา 4 ของ Arduino และ ข้วลบกับกราวด์ อัปโหลดภาพร่างที่อธิบายไว้ในหัวข้อ 4.4.1 ไปยัง Arduino และสังเกตการ กะพริบของ LED

รูปที่ 4.11แสดงแผนภาพวงจรของ Arduino ที่เชื่อมต่อกับ LED



ร**ูป 4.11** แผนภาพวงจรเพื่อเชื่อมต่อ LED กับ Arduino

40

Internet of Things กับ Raspberry Pi และ Arduino

ฝน ปอ. อ นั้น

4.6.1 ร่าง

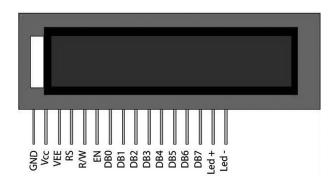
```
int LED_CONTROL=4;
การตังค่าเป็นโมฆะ ()
{
    โหมดพิน (LED_CONTROL, OUTPUT); // เริ่มต้นพิน 4 เป็นพินเอาต์พุต
}
วงเป็นโมฆะ ()
{
    digitalWrite (LED_CONTROL, สูง); // ทำให้พิน 4 ดีเลย์สูง
    (1000); // 1,000 มิลลิวินาทีล่าซ้า
    digitalWrite (LED_CONTROL, ต่ำ); // ทำให้พิน 4 ดีเลย์สูง
    (1000); // 1,000 มิลลิวินาทีล่าซ้า
}
```

4.7 เล่นกับ LCD ด้วย Arduino

้จอแสดงผลคริสตัลเหลว (LCD) เป็นโมดูลแสดงผลที่ใช้กันทั่วไป A 16×2 จอ LCD ใช้เป็น อุปกรณ์แสดงผลในวงจร โมดูลนี้เป็นที่ต้องการมากกว่าเจ็ดส่วน เนื่องจากไม่มีข้อจำกัดในการ แสดงอักขระพิเศษ หรือแม้แต่กำหนดเอง และประหยัด

A 16×2 LCD สามารถแสดงอักขระได้ 16 ตัวต่อแถว และมี 2 แถว ใน LCD นื้ 5×เมทริกซ์ 4 พิกเซลแสดงอักขระ มีการลงทะเบียนสองรายการคือการลงทะเบียนข้อมูลและการลง ทะเบียนคำสั่งรูปที่ 4.12แสดง 16×2 จอแอลซีดี

A 20×4 LCD มี 4 แถวและสามารถแสดงได้ 20 ตัวอักษรต่อแถว A 5×ใช้เมทริกซ์ 4 พิกเซลเพื่อแสดงอักขระ คำอธิบายพินเหมือนกับ LCD (16×2).รูปที่ 4.13แสดง 20×4 จอ แอลซีดี (ตาราง4.5).



รูป **4.12** จอแสดงผลคริสตัลเหลว (16×2).

ฝน ปອ. ອ นั้น



รูป **4.13** จอแสดงผลคริสตัลเหลว (20×4).

ตาราง 4.5 LCD Pin Description

เข็มหนุด	คำอธิบาย			
พิน (1) กราวด์	กราวด์ (0 V)			
w̄u (2) V cc	แหล่งจ่ายไฟ (5 V)			
w̄u (3) Vee	ตัวต้านทานปรับค่าได้ใช้เพื่อปรับความคมชัด			
ปักหมุด (4) ลงทะเบียน เลือก	เมื่อต่ำก็จะเลือกคำสั่งรีจิสเตอร์และถ้าสูงก็จะเลือก การลงทะเบียนข้อมูล			
ปักหมุด (5) อ่าน/เขียน	สูงเพื่ออ่านรีจิสเตอร์และต่ำเพื่อเขียนบนรีจิสเตอร์ ส่งข้อมูลไปยังสาย			
ปักหมุด (6) เปิดใช้งาน	ข้อมูลเมื่อได้รับพัลส์สูงไปต่ำ			
w̄u (7) DB0				
w̄u (8) DB1				
w̄u (9) DB2	สายข้อมูล 8 บิต			
w̄u (10) DB3				
หมุด (11) DB4				
w̄u (12) DB5				
w̄u (13) DB6				
w̄u (14) DB7				
w̄u (15) LED+	แบ็คไลท์ Vcc (5 V) แบ็คไลท์			
ขา (16) LED-	กราวด์ (0 V)			

การเชื่อมต่อ LCD

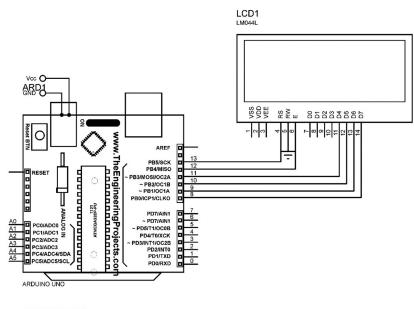
เชื่อมต่อส่วนประกอบดังต่อไปนั้:

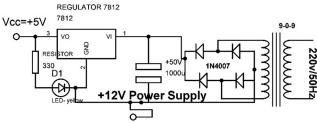
- ขาดิจิตอล Arduino (13) ถึง ขา RS (4) ของ LCD
- Arduino ดิจิตอลพิน (GND) ถึง RW พิน (5) ของ LCD
- ขาดิจิตอล Arduino (12) ถึง E ขา (6) ของ LCD

Internet of Things กับ Raspberry Pi และ Arduino

- Arduino ดิจิตอลพิน (11) ถึง D4 พิน (11) ของ LCD
- ขาดิจิตอล Arduino (10) ถึง D5 ขา (12) ของ LCD
- ขาดิจิตอล Arduino (9) ถึง D6 ขา (13) ของ LCD
- ขาดิจิตอล Arduino (8) ถึงขา D7 (14) ของ LCD

รูปที่ 4.14แสดงแผนภาพวงจรของ Arduino ที่เชื่อมต่อกับ LCD





รูป 4.14 แผนภาพวงจรสำหรับอ่าน LCD

4.7.1 ร่าง

```
# sɔu <LiquidCrystal.h> LiquidCrystal
LCD (13, 12, 11, 10, 9, 8);
การตั้งค่าเป็นโมฆะ ()
 lcd.begin(20, 4); // เริ่มต้น LCD lcd.print ("ยินดี
 ต้อนรับ"); // พิมพ์สตริงบน LCD ล่าช้า (2000); // หน่วง
 เวลา 2000mS
 lcd.clear();
}
วงเป็นโมฆะ ()
 lcd.setCursor(0, 1); // ตั้งค่าเคอร์เซอร์ของ LCD
 lcd.print("ECE Department"); // พิมพ์สตริงบน LCD ล่าซ้า
 (2000); // หน่วงเวลา 2000mS
 lcd.setCursor(0, 2); // ตั้งค่าเคอร์เซอร์ของ LCD
 lcd.print("Rajesh Singh"); // พิมพ์สตริงบน LCD ล่าซ้า
 (2000); // หน่วงเวลา 2000mS
}
```