

Activity 01: Programming environment

นักเรียนจะได้เรียนรู้การเขียนโปรแกรมโดยใช้ภาษาซีบน Arduino UNO board กิจกรรมชุดนี้จะเป็นการเตรียมความพร้อมของนักเรียนเพื่อให้เข้าใจการใช้งาน Arduino integrated development environment (Arduino IDE) เช่น การแปลภาษา (Compile) การประมวลผล (Execute) เป็นต้น

วัตถุประสงค์

1. เพื่อศึกษาการทำงานของสภาพแวดล้อมในการพัฒนาโปรแกรม
2. เพื่อเรียนรู้การใช้งาน Arduino IDE
3. เพื่อเรียนรู้การแปลภาษาและการประมวลผล

อุปกรณ์

1. Arduino UNO board (Fig.1)
2. Arduino integrated development environment (IDE) (<http://arduino.cc/en/Main/Software>)
3. USB type A – type B 1 set (to power the UNO)

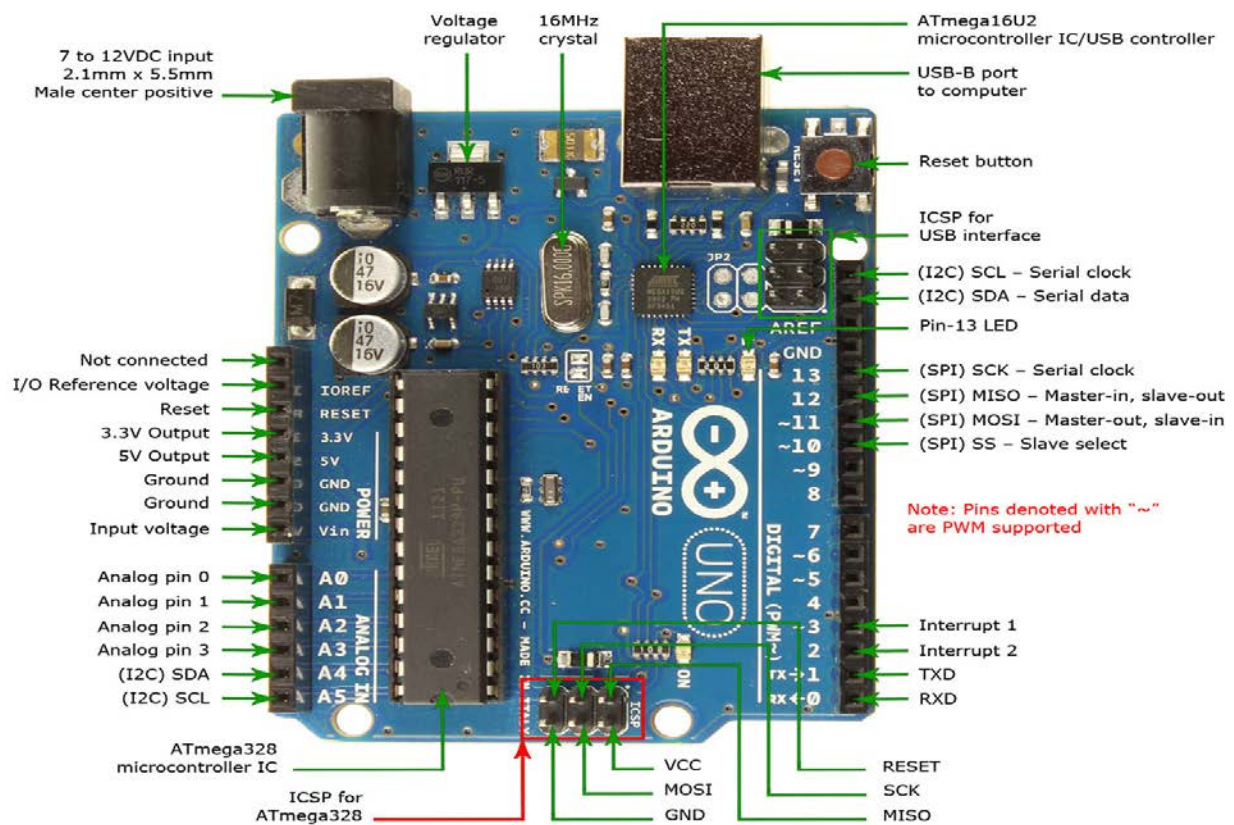


Figure 1. Basic board: the Arduino UNO

คำสั่ง

1. การติดตั้ง Arduino IDE บนเครื่องคอมพิวเตอร์ของคุณ คุณต้องไปที่ยูอาร์แอล <https://www.arduino.cc/en/Main/Software>
2. ดาวน์โหลด IDE จากเว็บไซต์แล้วติดตั้งลงบนเครื่องคอมพิวเตอร์ของคุณ Fig 1. แสดงหน้าจอแรกของ Arduino IDE เรียกว่า Arduino splash

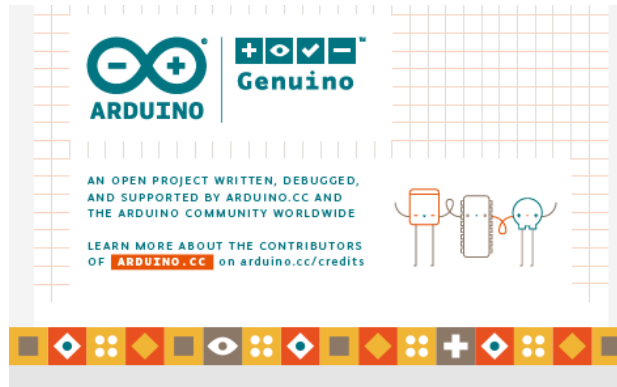


Figure 1. The Arduino splash screen

3. การใช้งานบอร์ด Arduino UNO เริ่มต้นจากการตรวจสอบว่าอุปกรณ์ไม่เสียหาย โดยการเสียบสายไฟเข้าไปในบอร์ดดังแสดงใน Fig 2 การเสียบสายไฟใช้สายยูเอสบี (USB) ที่แจกให้ เชื่อมต่อบอร์ดเข้ากับพอร์ตยูเอสบีบนเครื่องคอมพิวเตอร์ของคุณ หากอุปกรณ์ไม่เสียหายคุณ将会เห็นไฟ LED ส่วน power ติดสว่าง ดังแสดงใน Fig 3

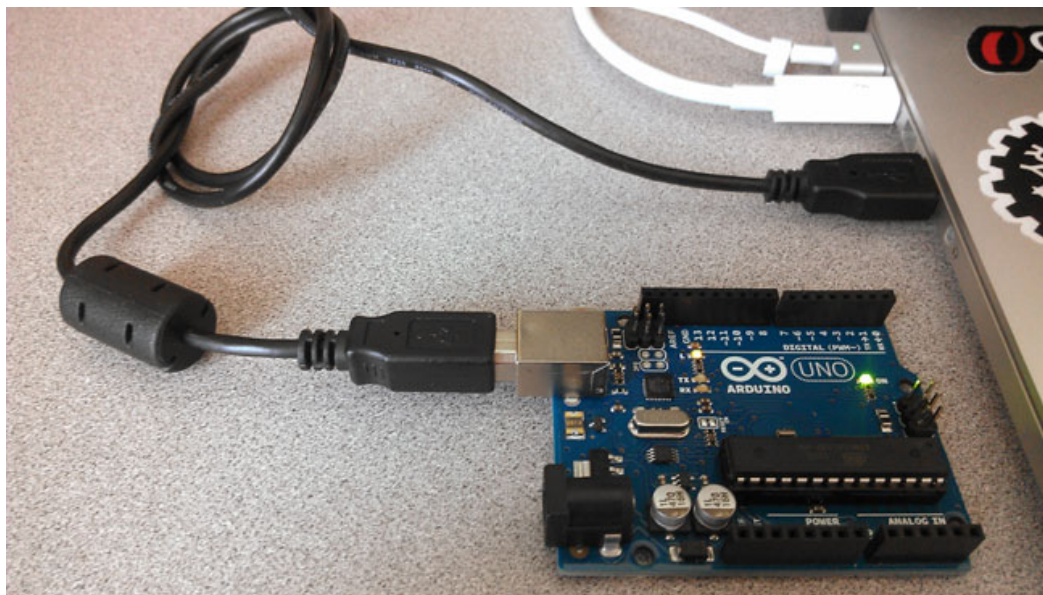


Fig. 2 the USB cable to power up the Arduino UNO



Figure 3. the LED status on board

4. เมื่อคุณเชื่อมต่อบอร์ดเข้ากับเครื่องคอมพิวเตอร์แล้ว ไดรเวอร์อุปกรณ์ (Device driver) ที่เหมาะสมต้องถูกติดตั้งบนเครื่องคอมพิวเตอร์เพื่อให้สามารถใช้งานบอร์ดได้ อย่างไรก็ตาม การดำเนินการติดตั้งดังกล่าวไม่ยุ่งยากคุณเพียงทำตามคำสั่งที่แสดงบนจอเท่านั้น Fig 4. แสดงหน้าจอการจัดการอุปกรณ์เมื่อคุณติดตั้งไดรเวอร์ถูกต้องบนระบบปฏิบัติการวินโดวส์ ทั้งนี้ บางเครื่องอาจจะแสดงการเชื่อมต่อที่พอร์ต COM3 ส่วนบางเครื่องอาจจะแสดงการเชื่อมต่อที่พอร์ต COM4

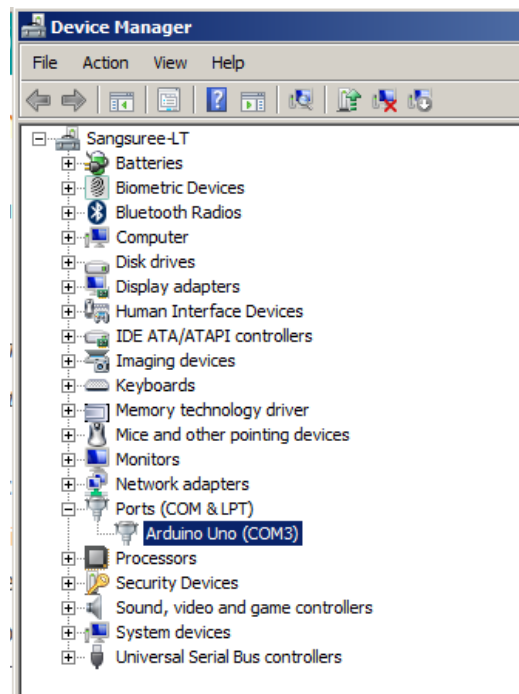


Figure 4. Device manager window

5. จากจุดนี้เราจะเริ่มต้นโหลดโปรแกรมเข้าไปบนบอร์ด Arduino โดยการโหลดโปรแกรมตัวอย่างชื่อว่า Blink โดยโปรแกรม Blink ที่คุณโหลดเข้าไปบนบอร์ด Arduino จะเป็นการสั่งให้ LED บนบอร์ด Arduino กระพริบ

โดยเริ่มต้นจากการเปิด IDE ขึ้นมา คุณจะเห็นหน้าจอแสดงใน Fig 5 จากนั้นให้ไปที่เมนู File ตามด้วยเมนู Examples และเลือกหัวข้อ Basics จากนั้นเลือกโปรแกรม Blink ดังแสดงใน Fig 6

หากคุณทำถูกต้องคุณ将会เห็นโปรแกรมแสดงบนหน้าจอ ณ เวลานี้ โปรแกรมถูกโหลดจากฮาร์ดดิสก์ของเครื่องคอมพิวเตอร์ของคุณมายัง IDE แต่ยังไม่ถูกบรรจุลงบนบอร์ด เนื่องจากคุณต้องทำการแปลภาษาของโปรแกรมให้เป็นภาษาที่บอร์ด Arduino เข้าใจก่อน เพื่อให้สามารถประมวลผลบนบอร์ดได้

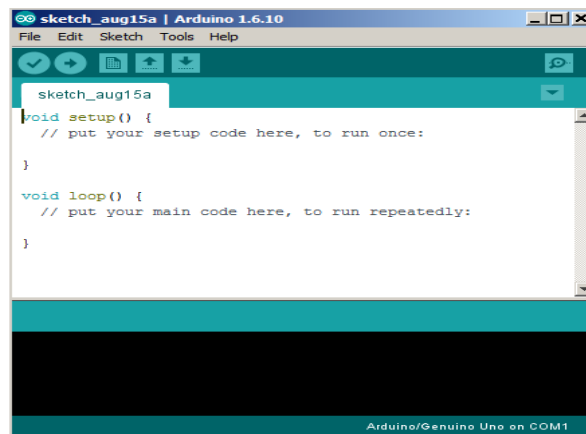


Figure 5. Arduino IDE window

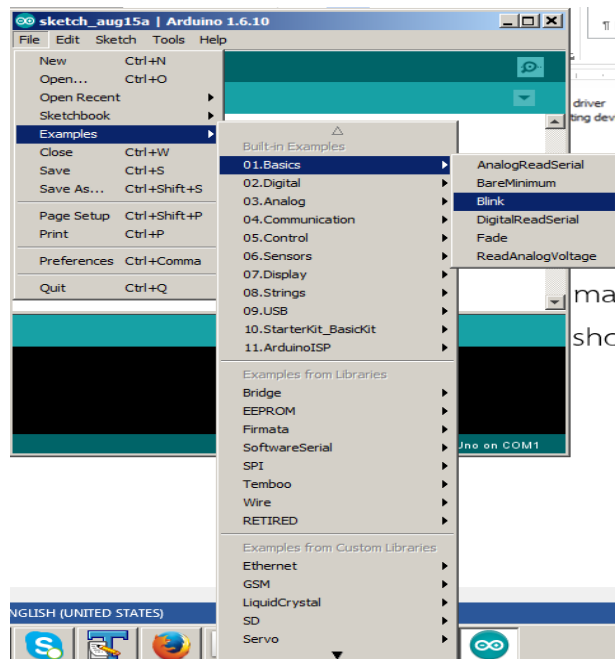


Figure 6. Menu of Basic Blink code

6. การสั้งแปลภาษา (Compile) ทำได้โดยการเลือกเมนูคำสั่ง Sketch ตามด้วย Verify/Compile คุณจะเห็นสถานะการแปลภาษาแสดงบนจอที่ตั้งแสดงใน Fig 7 เมื่อกระบวนการแปลภาษาเสร็จสิ้นแล้วส่วนด้านล่างของจอภาพจะปรากฏข้อมูล ดังนี้

Sketch uses 928 bytes (2%) of program storage space. Maximum is 32,256 bytes.

Global variables use 9 bytes (0%) of dynamic memory, leaving 2,039 bytes for local variables.

Maximum is 2,048 bytes.

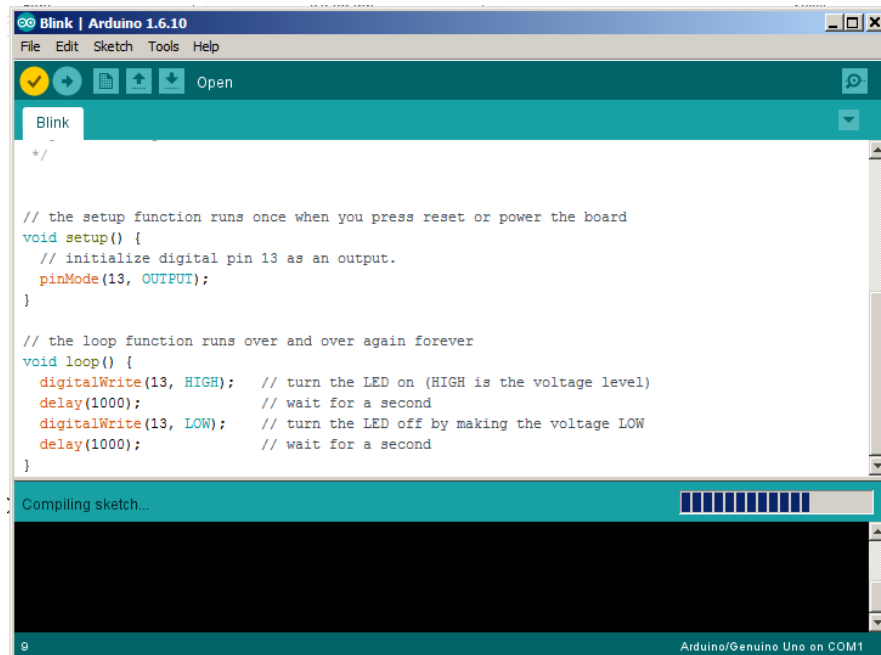


Figure 7. The compiling status

7. ขั้นตอนต่อไปคือการนำผลการแปลภาษาที่ได้จากขั้นตอนที่ 6 ไปบนบอร์ด Arduino UNO ที่เชื่อมต่ออยู่กับเครื่องคอมพิวเตอร์ของคุณ ต่อไปคุณควรจะตรวจสอบข้อมูลต่อไปนี้ทุกครั้ง เพื่อป้องกันข้อผิดพลาด
- ตรวจสอบว่าการกำหนดบอร์ดที่ต่ออยู่กับเครื่องคอมพิวเตอร์ของคุณถูกต้อง โดยการไปที่เมนู Tools และหัวข้อ Board เพื่อให้มั่นใจว่าบอร์ด Arduino UNO เป็นหัวข้อที่ถูกเลือก
 - ตรวจสอบว่าบอร์ดที่เชื่อมต่อกับเครื่องคอมพิวเตอร์ของคุณจริง โดยการไปที่เมนู Tools และหัวข้อ Port เพื่อให้มั่นใจว่าบอร์ดเชื่อมต่ออยู่ที่พอร์ตของคอมพิวเตอร์

การอัปโหลดผลการแปลภาษาไปบนบอร์ดทำได้โดยการไปที่เมนู Sketch และคำสั่ง Upload

หากคุณทำทุกขั้นตอนถูกต้องแล้ว คุณจะเห็นหลอด LED บนบอร์ด Arduino UNO กระพริบ ดังแสดงใน Fig 8 แสดงส่วนประกอบของบอร์ด Arduino UNO ที่เราใช้ในรายวิชานี้

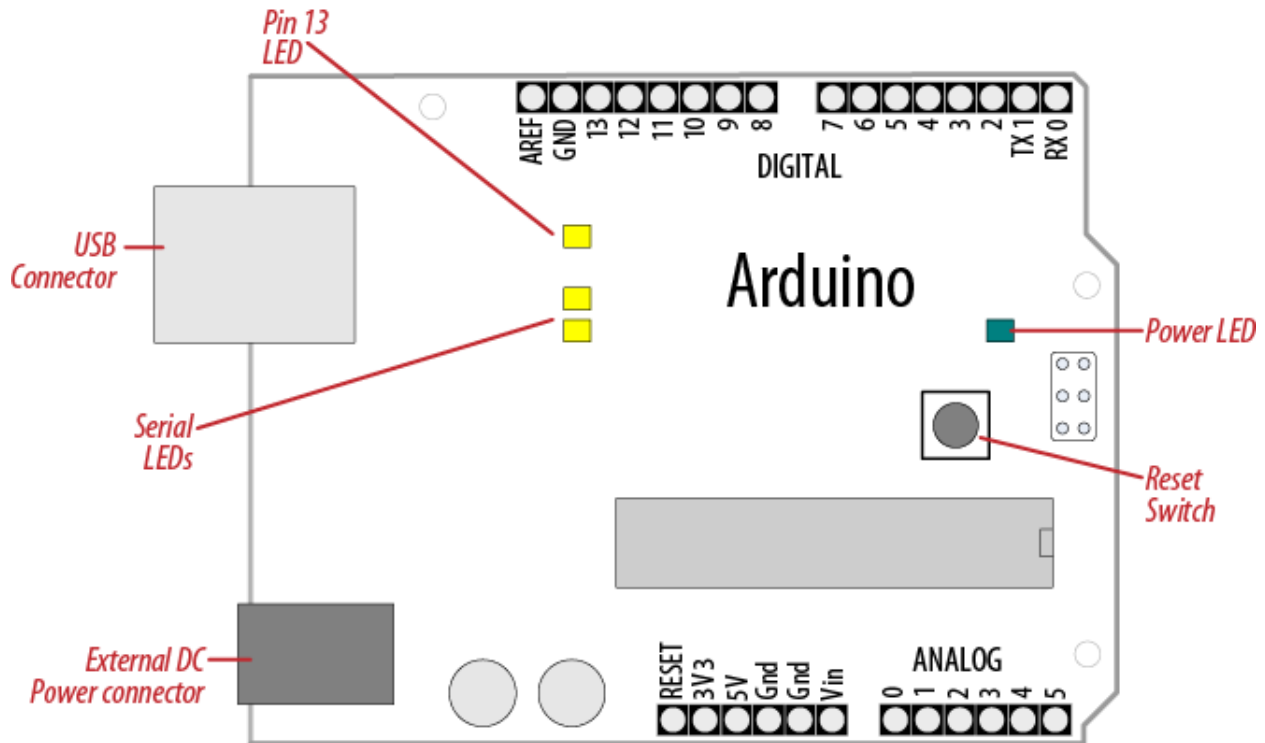


Fig. 8 Basic Arduino Board Components

Source: M. Margolis, "Arduino Cookbook", O'Reilly, 2011. ISBN: 978-0-596-80247-9

8. ขั้นตอนต่อไปนี้ เราจะลองเขียนโค้ดเองจาก sketch ว้างๆ โดยการไปที่เมนู File คำสั่ง New คุณจะได้นหน้าจอใหม่ว่างๆ ขึ้นมา ให้พิมพ์โปรแกรมต่อไปนี้ลงไป (ข้อควรระวัง: ให้พิมพ์ทุกอักขระที่ให้ไปข้างล่าง)

```
const int ledPin = 13; // LED connected to digital pin 13
void setup() {
  pinMode(ledPin, OUTPUT);
}
void loop(){
  digitalWrite(ledPin, HIGH); // set the LED on
  delay(2000);                // wait for two seconds
  digitalWrite(ledPin, LOW);  // set the LED off
  delay(1000);                // wait for one second
}
```

เมื่อพิมพ์เสร็จแล้วให้ทำการแปลภาษา (ดูคำสั่งจากการทดลองข้อก่อนหน้านี้) และโหลดโปรแกรมลงบนบอร์ด Arduino UNO หาก你做ถูกต้อง คุณจะเห็นว่าการกะพริบของ LED จะแตกต่างไปจากเดิม

ทำการบันทึกโค้ดที่คุณเขียนลงในฮาร์ดดิสก์ โดยใช้คำสั่งเมนู File หัวข้อ Save และให้ตั้งชื่อโปรแกรมเป็น Blink-1-10

9. ขั้นตอนต่อไป เราจะมาเรียนรู้ไวยากรณ์บางส่วนของภาษาซีบน Arduino โดยเริ่มต้นจากการเอา
เครื่องหมาย // จากบรรทัดที่ 1, 6, 7, 8, และ 9 ออก จากนั้นทดลองแปลภาษาและโหลดลงบนบอร์ด
จากนั้นตอบคำถามต่อไปนี้

```
1. const int ledPin = 13; // LED connected to digital pin 13
2. void setup() {
3.   pinMode(ledPin, OUTPUT);
4. }
5. void loop(){
6.   digitalWrite(ledPin, HIGH); // set the LED on
7.   delay(2000);                // wait for two seconds
8.   digitalWrite(ledPin, LOW);  // set the LED off
9.   delay(1000);                // wait for one second
10 }
```

11.1 คุณได้รับผลที่แตกต่างจากผลที่ได้ในขั้นตอนที่ 10 หรือไม่? ☐ Yes ☐ No

11.2 ข้อความที่ต่อท้าย // หมายถึงอะไร? _____

11.3 ใส่ // ลงไปให้ถูกต้อง จากนั้น ให้เอาโค้ดบรรทัดที่ 10 ออก แล้วแปลภาษาเพื่อโหลดไปบนบอร์ด
คุณได้ error หรือไม่ เพราะเหตุใด อธิบายสั้นๆ

11.4 ใส่โค้ดบรรทัดสุดท้ายเข้าไปให้ถูกต้อง จากนั้น ให้เปลี่ยนโค้ดบรรทัดที่ 1 เป็นดังนี้
// const int ledPin = 13;
และแปลภาษาเพื่อโหลดไปบนบอร์ด คุณได้ error หรือไม่ เพราะเหตุใด อธิบายสั้นๆ

11.5 คุณได้เรียนรู้ข้อจำกัด ข้อกำหนดอะไรของการเขียนโปรแกรมบ้าง อธิบายสั้นๆ

10. ขั้นตอนต่อไปนี้จะเรียนรู้ส่วนแสดงผลของบอร์ด Arduino UNO เรียกว่า Serial Monitor โดยการเริ่มต้นเขียนโปรแกรกดังต่อไปนี้บน sketch ว่าง และบันทึก (save) ชื่อไฟล์ Blink-1-12

```
const int ledPin = 13;  // LED connected to digital pin 13
void setup() {
  pinMode(ledPin, OUTPUT);
  Serial.begin(9600);
}
void loop(){
  digitalWrite(ledPin, HIGH);  // set the LED on
  Serial.println("HIGH");      // print on the monitor
  delay(2000);                 // wait for two seconds
  digitalWrite(ledPin, LOW);   // set the LED off
  Serial.println("LOW");       // print on the monitor
  delay(1000);                 // wait for one second
}
```

ทำการแปลภาษาและโหลดบนบอร์ด จากนั้นให้เปิดส่วนแสดงผลโดยการไปที่เมนู Tools และคำสั่ง Serial Monitor คุณจะเห็นหน้าต่างดังแสดงใน Fig 9 ซึ่งหน้าต่างนี้จะแสดงข้อความที่ถูกส่งผ่านคำสั่ง Serial.println()

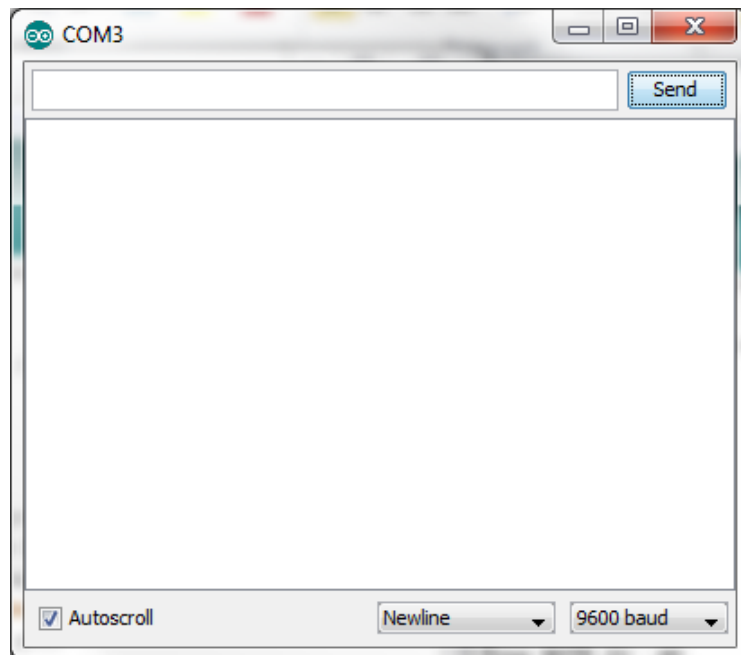


Fig. 9 Serial monitor display

Check point: Small project

ลองเอาหลอดไฟ LED ต่อกับขา 13 และขา GND จากนั้นลองเขียนโปรแกรมให้หลอดไฟติดและดับ โดยปรับระยะเวลาการติดและดับให้แตกต่างจากตัวเลขในการทดลองก่อนหน้านี้

งานท้ายกิจกรรมที่ 1

- A.1.1) จงสืบค้นความหมายของค่าคงที่ต่อไปนี้ INPUT, HIGH, LOW จากเว็บของ Arduino และเขียนอธิบายว่าค่าคงที่แต่ละตัวมีความหมายอะไร โดยใช้คำพูดของตนเองสั้นๆ
- A.1.2) จงอธิบายว่าจะเกิดอะไรขึ้น หากคุณไม่ใส่ ; ต่อท้ายคำสั่งภายใน sketch ของคุณ
- A.1.3) จงอธิบายว่าจะเกิดอะไรขึ้น หากคุณไม่ใส่ () ต่อท้ายโค้ด void setup
- A.1.4) จงอธิบายว่าจะเกิดอะไรขึ้น หากคุณใส่ // หน้าบรรทัดใน sketch
- A.1.5) จงอธิบายว่า ปุ่มต่างๆ บนหน้าจอ sketch ของ Arduino IDE ต่อไปนี้ใช้ทำอะไรบ้าง



- A.1.6) คุณต้องการเขียนโปรแกรมสั่งงานให้ Arduino ทำอะไรในรายวิชานี้ คิดในลักษณะของโครงงาน โดยอธิบายสั้นๆ ให้อาจารย์เข้าใจ คุณสามารถหาแนวคิดของโครงงานจาก Internet ได้ แต่พยายามใช้คำพูดของตนเองเพื่อฝึกการเขียนอธิบาย หากเป็นไปได้ เขียนภาพประกอบการอธิบายเพื่อให้เข้าใจง่ายขึ้น