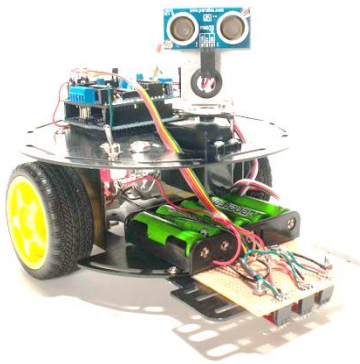


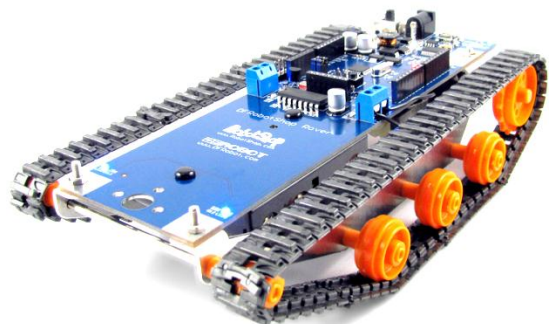
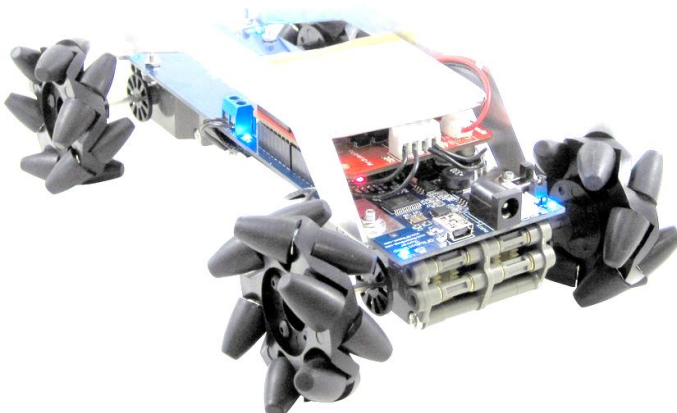
MODUL PRAKTIKUM

EL2142 Praktikum Sistem Digital dan
Mikroprosesor



Beginners Guide To The Arduino

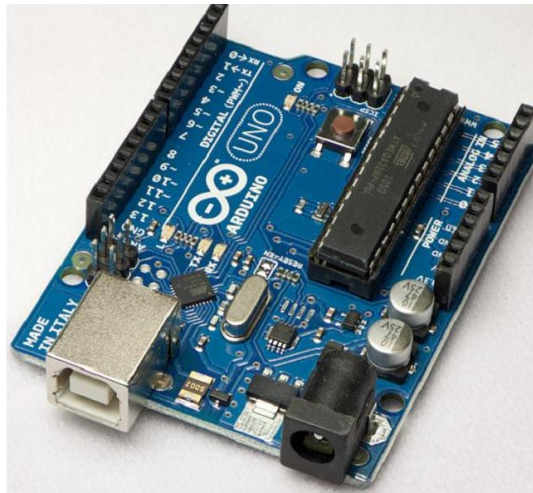
By Mukti Winanda



Perkenalan Arduino

Menurut Massimo Banzi:

Arduino is an open source physical computing platform based on a simple input/output (I/O) board and a development environment that implements the Processing language (www.processing.org). Arduino can be used to develop standalone interactive objects or can be connected to software on your computer (such as Flash, Processing, VVVV, or Max/MSP). The boards can be assembled by hand or purchased preassembled; the open source IDE (Integrated Development Environment) can be downloaded for free from www.arduino.cc.



Digital dan Analog pin

Pada Arduino terdapat dua buah jenis pin, yaitu analog dan digital. Pin digital memiliki dua buah nilai yang dapat ditulis kepadanya yaitu High(1) dan Low(0). Logika high maksudnya ialah 5 Volt dikirim ke pin baik itu oleh mikrokontroler atau dari komponen. Low berarti pin tersebut bertegangan 0 Volt. Dari logika ini, anda dapat membayangkan perumpamaan: start/stop, siap/tidak siap, on/off, dsb.

Pin-pin analog memiliki karakteristik yang berbeda dari pin digital. Informasi yang dapat ditulis atau dibaca sangat lebar. Misalnya saja untuk write, nilai dari 0-255 dapat ditulis(0V – 5V). Sedangkan untuk read, nilai dari 0-1023(0V – 5V dengan setiap kenaikan sebesar 0,005V) dapat direpresentasikan.

Berikut diperlihatkan penampang Arduino yang memperlihatkan barisan pin digital dan analog.

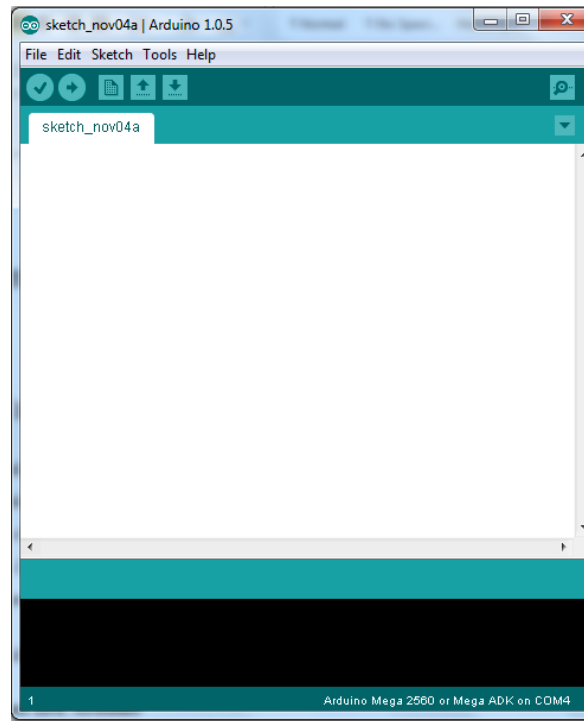


Pin-pin digital berada pada bagian atas. Pin-pin itulah yang nantinya dihubungkan ke berbagai kontrol yang berkomunikasi dengan menggunakan nilai digital. Beberapa pin digital dapat digunakan sebagai Pulse Width Modulation(PWM). Secara umum pin PWM ini dapat digunakan untuk mengirim nilai/informasi analog ke komponen. Tanda ~ pada pin 3, 5, 6, 9, 10, 11 menandakan fungsi PWM. Fungsi PWM ini merupakan tambahan dari fungsi digital. Artinya pin-pin tersebut selain fungsi digital, dapat dikonfigurasi menjadi PWM. Kemungkinan nilai dari pin-pin digital tersebut ialah IN (informasi masuk dari komponen ke kontrol) dan sebaliknya OUT(informasi keluar dari kontrol ke komponen). Pada bagian bawah, terdapat pin-pin power. Tersedia pin 5V, 3.3V, dua pin ground, Vin, dan reset.

Di sebelah kanan bagian bawah, anda akan melihat pin-pin analog input. Pin-pin ini dapat menerima masukan informasi analog dari 0 hingga 5 Volt dengan kenaikan sebesar 0.005 V. Representasi 0 V ialah 0, dan 5V ialah 1023. Di atas pin-pin analog terdapat mikrokontroler AVR. Dan di atas AVR terdapat push button reset untuk merestart program. Terdapat konektor ISP di sebelah kanan push button reset yang dapat digunakan untuk memprogram kontroler Arduino dalam kondisi tertentu(chip erase akan mengakibatkan bootloader Arduino ikut terhapus. Jadi hati-hati dalam menggunakannya).

Instalasi IDE

Untuk mulai memprogram, dibutuhkan IDE Arduino. Langsung saja download versi terbaru dari Arduino.cc. Download sesuai dengan OS yang akan digunakan. Pada modul ini digunakan windows dan IDE versi 1.0.5. Ekstrak hasil download, dan ada akan mendapatkan folder “arduino-1.0.5”, kemudian double-click “arduino.exe” untuk mulai melakukan penginstallan. Berikut penampakan IDE Arduino:



Pada gambar diatas, terdapat, beberapa tombol yang mempunyai fungsi sebagai berikut:



Verify: Cek error dan lakukan kompilasi kode



Upload: Upload kode anda ke board/kontroler. Asumsi bahwa board dan serial port telah disetting dengan benar.



New: Membuat aplikasi baru.



Open: Buka proyek yang telah ada atau dari contoh-contoh/examples.



Save: Simpan proyek anda.



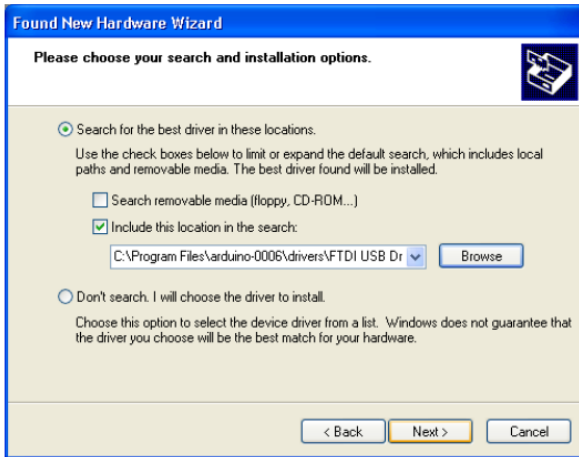
Serial Monitor: Membuka serial port monitor untuk melihat feedback/umpan balik dari board anda.

Proses kerja Arduino ialah anda melakukan pemrograman pada IDE, compile, dan upload binary/hex file ke kontroler. Berbeda dengan Processing yang kode hasil compile langsung dijalankan di komputer, kode hasil compile Arduino harus diupload ke kontroler sehingga dapat dijalankan

Install USB drivers

Untuk menjalankan Arduino, pertama-tama hubungkan USB cable yang terpasang pada board Arduino ke PC. Ketika pertama kali dihubungkan, maka akan muncul kotak dialog “Add New Hardware” wizard, kemudian arahkan folder driver kedalam drivers/FTDI USB Drivers yang terdapat pada folder Arduino berada.





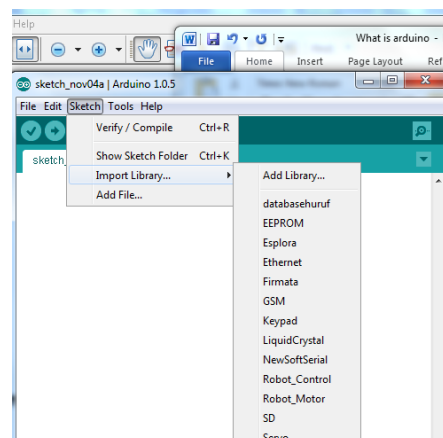
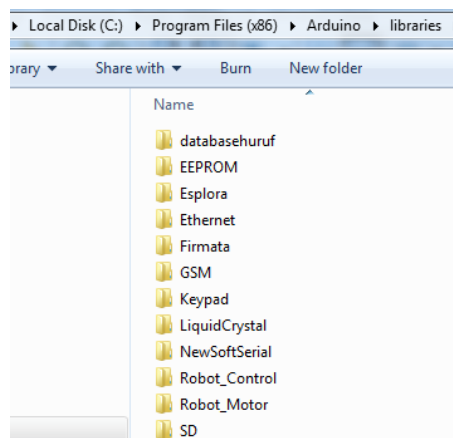
Setelah melakukan install drivers, maka kamu telah siap untuk mengupload program yang akan dibuat ke dalam mikrokontroler Arduino.

Insert libraries

Dalam beberapa kondisi, nantinya kadangkala diperlukan libraries tambahan apabila libraries yang telah ada tidak mencukupi project yang akan dibuat. Berikut cara menambahkan libraries ke dalam IDE Arduino.

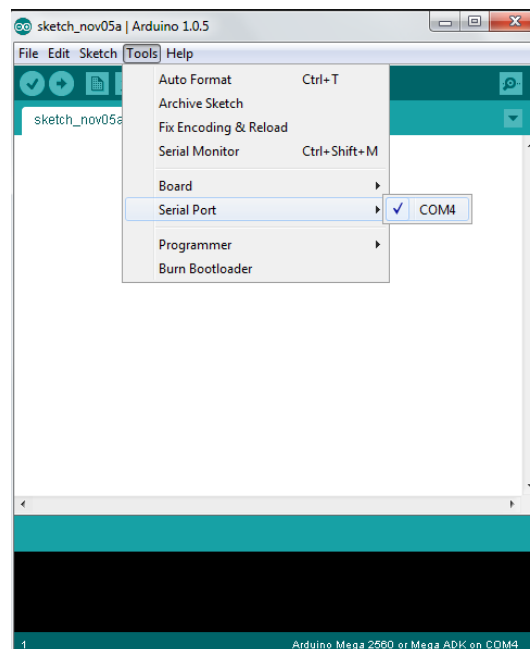
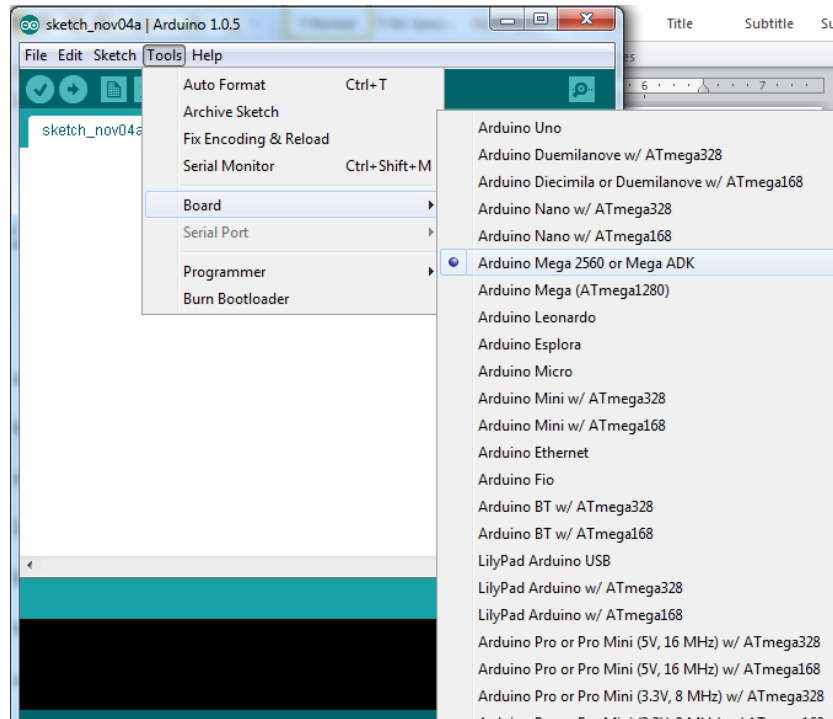
1. Copy files libraries yang telah di download
2. Letakkan pada folder libraries yang terdapat di dalam folder Arduino
3. Jalankan ulang software Arduino IDE

Pada contoh dibawah ialah menambahkan libraries keypad ke dalam libraries yang sebelumnya telah ada kemudian menjalankan ulang software IDE Arduino.



Project 1 Basic Arduino

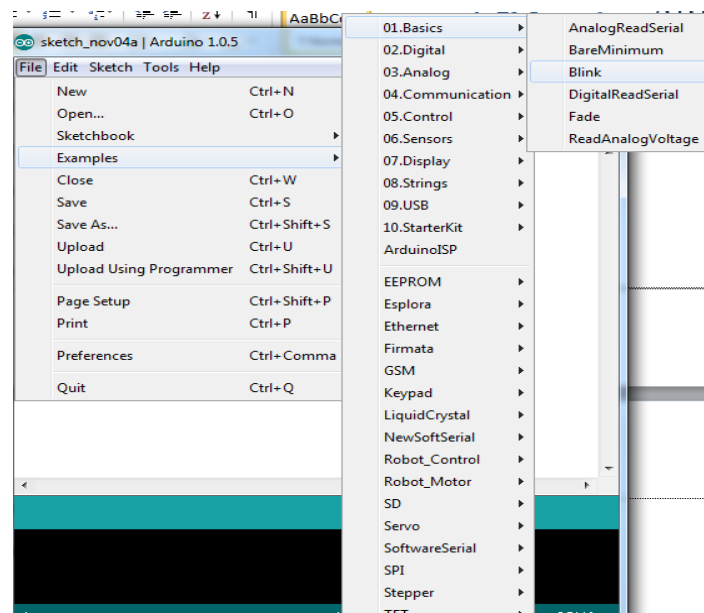
Sebelum melakukan project, pada bagian Arduino IDE dicek terlebih dahulu pada menu tools, board dan port yang digunakan pada project. Sesuaikan board dan port dengan jenis board yang digunakan serta port yang terhubung dengan USB cable pada board Arduino.



Pada project pertama ini, dilakukan pengujian terhadap board Arduino. Pengujian ini dilakukan untuk mengetes apakah board dapat berfungsi dengan baik atau tidak, selain itu untuk menguji keterkiriman source code yang di upload pada board. Pada pengujian basic ini, diujikan dua buah program yang berasal dari examples program yaitu blink dan stringlength.

Blink

Pada percobaan ini, dilakukan penyalan LED yang menempel langsung pada board Arduino. LED tersebut akan menyala dan mati secara bergantian selama satu detik. Pertama-tama buka file source code example seperti berikut.



Berikut merupakan source code yang terdapat pada basic examples blink.

```
// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

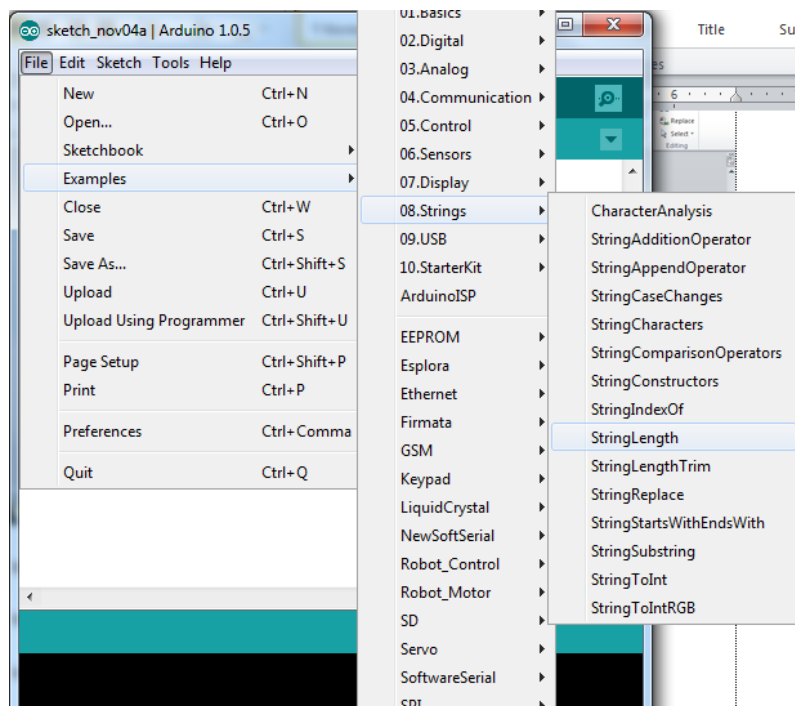
// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);               // wait for a second
  digitalWrite(led, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);               // wait for a second }
```


Berikutnya compile dan upload program ke dalam Arduino board. Perhatikan dan catat hasil yang terjadi, apakah sesuai dengan spesifikasi atau tidak.

StringLength

Pada percobaan ini dilakukan pengujian terhadap penggunaan serial monitor yang terdapat pada Arduino. Percobaan ini bertujuan untuk mengetahui panjang kata dari setiap karakter maupun string yang dikirimkan pada Arduino, untuk kemudian ditampilkan pada serial monitor.



Berikut merupakan source code yang terdapat pada basic examples Stringlength.

```
String txtMsg = ""; // a string for incoming text
int lastStringLength = txtMsg.length(); // previous length of the String

void setup() {
  // Open serial communications and wait for port to open:
  Serial.begin(9600);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for Leonardo only
  }
}
```

```
// send an intro:
Serial.println("\n\nString  length():");
Serial.println();
}





void loop() {
  // add any incoming characters to the String:
  while (Serial.available() > 0) {
    char inChar = Serial.read();
    txtMsg += inChar;
  }

  // print the message and a notice if it's changed:
  if (txtMsg.length() != lastStringLength) {
    Serial.println(txtMsg);
    Serial.println(txtMsg.length());
    // if the String's longer than 140 characters, complain:
    if (txtMsg.length() < 140) {
      Serial.println("That's a perfectly acceptable text message");
    }
    else {
      Serial.println("That's too long for a text message.");
    }
    // note the length for next time through the loop:
    lastStringLength = txtMsg.length();
  }
}
```

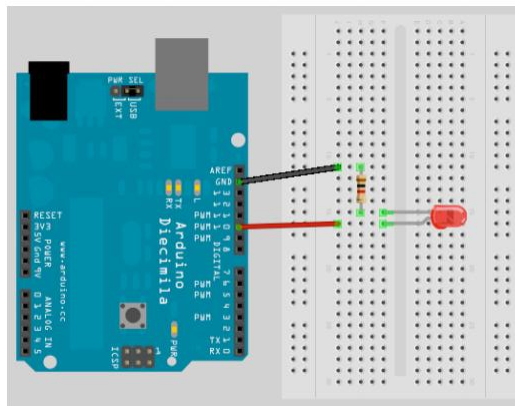
Berikutnya compile dan upload program ke dalam Arduino board. Perhatikan dan catat hasil yang terjadi, apakah sesuai dengan spesifikasi atau tidak.

Project 2 LED Flasher

Pada percobaan ini, dilakukan penyalan LED yang terpasang di luar dari board Arduino. Seperti pada percobaan blink sebelumnya, LED dapat dinyalakan dan dimatikan secara bergantian selama waktu yang diinginkan. Berikut komponen yang diperlukan pada percobaan ini:

Breadboard	
Red LED	
150 ohm Resistor	
Jumper Wires	

Berikutnya setelah komponen yang diperlukan telah terkumpul, susun rangkaian seperti terlihat pada gambar berikut:



Kemudian pasang USB cable pada Arduino dengan port USB yang terdapat pada PC untuk mengupload program. Buka Arduino IDE, lalu ketikkan program berikut.







```
// Project 2 - LED Flasher
int ledPin = 10;
void setup() {
  ! pinMode(ledPin, OUTPUT);
}
void loop() {
  digitalWrite(ledPin, HIGH);
```

```
    delay(1000);  
    digitalWrite(ledPin, LOW);  
    delay(1000);  
}
```

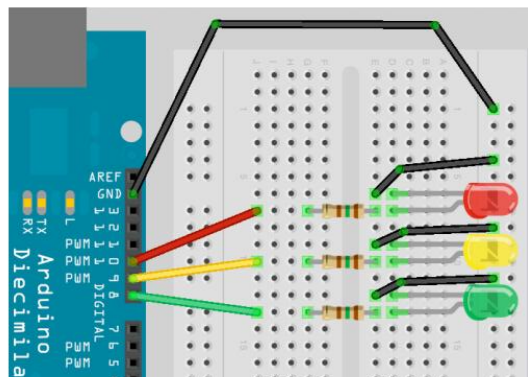
Berikutnya compile dan upload program ke dalam Arduino board. Perhatikan dan catat hasil yang terjadi, apakah sesuai dengan spesifikasi atau tidak.

Project 3 Traffic Lights

Pada percobaan ini, dilakukan pembuatan prototype lampu lalu lintas sederhana. Penyalaan lampu menggunakan LED berwarna merah, kuning, dan hijau yang kemudian disusun secara vertikal berjajar. LED akan menyala seperti halnya lampu lalu lintas yang berada di perempatan jalan. Pertama-tama lampu akan menyala merah, kemudian merah dan kuning secara bersama, lalu hijau yang menandakan kondisi jalan serta kuning dan kembali lagi ke merah. Berikut komponen yang diperlukan pada percobaan ini:

Breadboard	
Red LED	
Yellow LED	
Green LED	
3 x 220 ohm Resistor	
Jumper Wires	

Berikutnya setelah komponen yang diperlukan telah terkumpul, susun rangkaian seperti terlihat pada gambar berikut:



Kemudian pasang USB cable pada Arduino dengan port USB yang terdapat pada PC untuk mengupload program. Buka Arduino IDE, lalu ketikkan program berikut.

```
// Project 3 - Traffic Lights
int ledDelay = 10000; // delay in between changes
int redPin = 10;
int yellowPin = 9;
int greenPin = 8;
void setup() {
  pinMode(redPin, OUTPUT);
  pinMode(yellowPin, OUTPUT);
  pinMode(greenPin, OUTPUT);
}
void loop() {

  // turn the red light on
  digitalWrite(redPin, HIGH);
  delay(ledDelay); // wait 5 seconds

  digitalWrite(yellowPin, HIGH); // turn on yellow
  delay(2000); // wait 2 seconds

  digitalWrite(greenPin, HIGH); // turn green on
  digitalWrite(redPin, LOW); // turn red off
  digitalWrite(yellowPin, LOW); // turn yellow off
  delay(ledDelay); // wait ledDelay milliseconds




  digitalWrite(yellowPin, HIGH); // turn yellow on
  digitalWrite(greenPin, LOW); // turn green off
  delay(2000); // wait 2 seconds

  digitalWrite(yellowPin, LOW); // turn yellow off
  // now our loop repeats
}
```

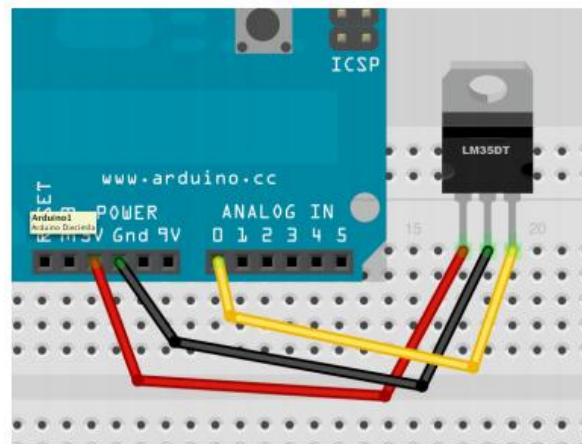
Berikutnya compile dan upload program ke dalam Arduino board. Perhatikan dan catat hasil yang terjadi, apakah sesuai dengan spesifikasi atau tidak.

Project 4 Serial Temperature Sensor

Pada percobaan kali ini dilakukan pengukuran suatu suhu ruangan dengan menggunakan sensor LM35. Suhu yang nantinya ditampilkan memiliki satuan derajat celcius dan dalam kisaran 0-100 derajat. Suhu tersebut dapat dilihat dengan menggunakan serial monitor yang terdapat pada Arduino IDE. Berikut komponen yang diperlukan pada percobaan ini:

Breadboard	
LM 35	
Jumper Wires	

Berikutnya setelah komponen yang diperlukan telah terkumpul, susun rangkaian seperti terlihat pada gambar berikut:



Kemudian pasang USB cable pada Arduino dengan port USB yang terdapat pada PC untuk mengupload program. Buka Arduino IDE, lalu ketikkan program berikut.

```
int potPin = 0;
float temperature = 0;
void setup()
{
    Serial.begin(9600);
```

```
Serial.println("LM35 Thermometer ");
analogReference(INTERNAL); //INTERNAL1V1 for Arduino Mega
}
void printTenths(int value) {
  // prints a value of 123 as 12.3
  Serial.print(value / 10);
  Serial.print(".");
  Serial.println(value % 10);
}
void loop() {
  int span = 20;
  int aRead = 0;
  for (int i = 0; i < span; i++) {
    aRead = aRead+analogRead(potPin);
  }
  aRead = aRead / 20;

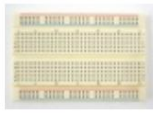
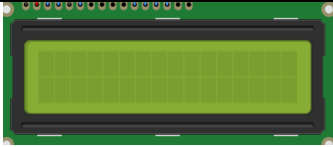


  temperature = ((100*1.1*aRead)/1024)*10;
  // convert voltage to temperature
  Serial.print("Analog in reading: ");
  Serial.print(long(aRead));
  // print temperature value on serial monitor
  Serial.print(" - Calculated Temp: ");
  printTenths(long(temperature));

  delay(500);
}
```

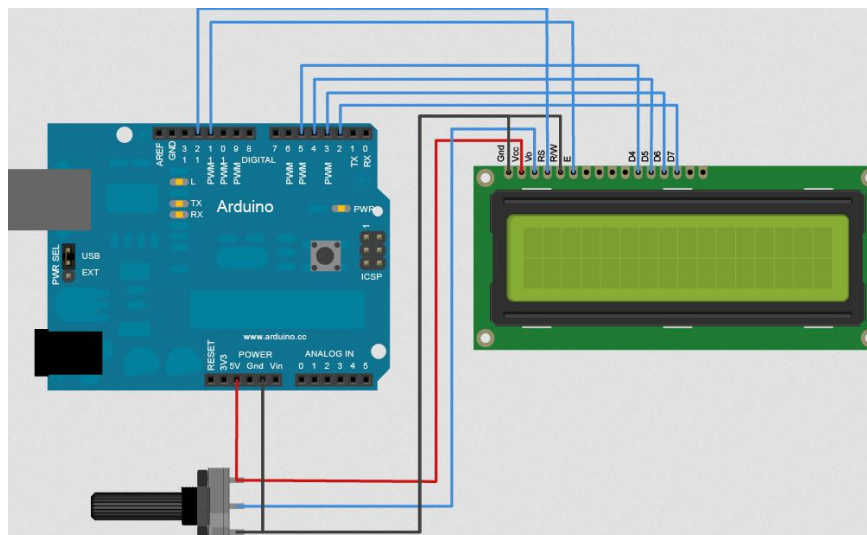
Berikutnya compile dan upload program ke dalam Arduino board. Perhatikan dan catat hasil yang terjadi, apakah sesuai dengan spesifikasi atau tidak.

Project 5 LCD

Pada percobaan kali ini dilakukan ujicoba untuk menggunakan LCD. Percobaan sederhana yang ingin ditampilkan pada layar ialah tulisan “helloworld”, kemudian tulisan tersebut nantinya dapat bergerak dari kanan ke kiri dan sebaliknya. Berikut komponen yang diperlukan pada percobaan ini:

Breadboard	
LCD	
Potensiometer 1k ohm	
Jumper Wires	

Berikutnya setelah komponen yang diperlukan telah terkumpul, susun rangkaian seperti terlihat pada gambar berikut:



Kemudian pasang USB cable pada Arduino dengan port USB yang terdapat pada PC untuk mengupload program. Buka Arduino IDE, lalu ketikkan program berikut.

```
http://www.arduino.cc/en/Tutorial/LiquidCrystal

// include the library code:
#include <LiquidCrystal.h>

// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup() {
  // set up the LCD's number of columns and rows:
  lcd.begin(16, 2);
  // Print a message to the LCD.
  lcd.print("hello, world!");
}

void loop() {
  // set the cursor to column 0, line 1
  // (note: line 1 is the second row, since counting begins with 0):
  lcd.setCursor(0, 1);
  // print the number of seconds since reset:
  lcd.print(millis()/1000);
}
```

Untuk membuat tulisan dapat bergerak dari kiri ke kanan dan sebaliknya, perlu dilakukan beberapa penambahan seperti berikut.

```
void loop() {
  // scroll 13 positions (string length) to the left
  // to move it offscreen left:
  for (int positionCounter = 0; positionCounter < 13; positionCounter++) {
    // scroll one position left:
    lcd.scrollDisplayLeft();
    // wait a bit:
    delay(150);
  }

  // scroll 29 positions (string length + display length) to the right
  // to move it offscreen right:
  for (int positionCounter = 0; positionCounter < 29; positionCounter++) {
    // scroll one position right:
    lcd.scrollDisplayRight();
    // wait a bit:
    delay(150);
  }

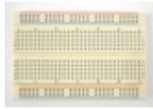





  // scroll 16 positions (display length + string length) to the left
  // to move it back to center:
  for (int positionCounter = 0; positionCounter < 16; positionCounter++) {
    // scroll one position left:
    lcd.scrollDisplayLeft();
    // wait a bit:
    delay(150);
  }

  // delay at the end of the full loop:
  delay(1000);
}
```

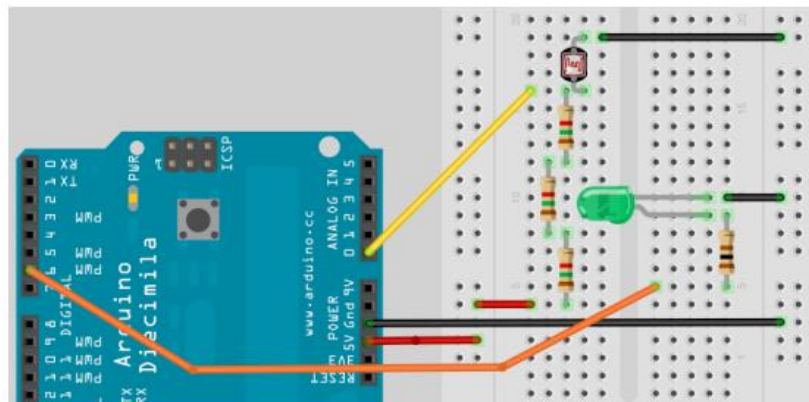
Berikutnya compile dan upload program ke dalam Arduino board. Perhatikan dan catat hasil yang terjadi, apakah sesuai dengan spesifikasi atau tidak.

Project 6 Light Sensor

Pada percobaan kali ini dilakukan uji coba terhadap penggunaan light sensor. Pada rangkaian akan dipasang dengan Light Dependent Resistor dan juga LED. Hasil pembacaan dari sensor cahaya tersebut nantinya akan menentukan lamanya cahaya yang diberikan oleh LED. Berikut komponen yang diperlukan pada percobaan ini:

Breadboard	
Light Dependent Resistor	
100 ohm Resistor	
3 x 1k ohm Resistor	
Green LED	
Jumper Wires	

Berikutnya setelah komponen yang diperlukan telah terkumpul, susun rangkaian seperti terlihat pada gambar berikut:




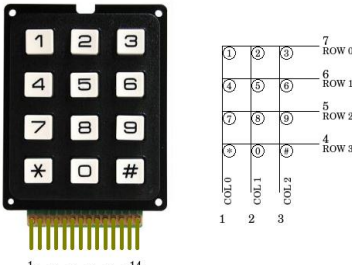


Kemudian pasang USB cable pada Arduino dengan port USB yang terdapat pada PC untuk mengupload program. Buka Arduino IDE, lalu ketikkan program berikut.

```
//Project 6 - Light Sensor
// Pin we will connect to LED
int ledPin = 6;
// Pin connected to LDR
int ldrPin = 0;
// Value read from LDR
int lightVal = 0;
void setup()
{
  // Set both pins as outputs
  pinMode(ledPin, OUTPUT);
}
void loop()
{
  // Read in value from LDR
  lightVal = analogRead(ldrPin);
  // Turn LED on
  digitalWrite(ledPin, HIGH);
  // Delay of length lightVal
  delay(lightVal);
  // Turn LED off
  digitalWrite(ledPin, LOW);
  // Delay again
  delay(lightVal);
}
```

Berikutnya compile dan upload program ke dalam Arduino board. Perhatikan dan catat hasil yang terjadi, apakah sesuai dengan spesifikasi atau tidak.

Project 7 Keypad dan Buzzer

Pada percobaan kali ini dilakukan uji coba terhadap penggunaan keypad 4x3 dan buzzer. Setiap tombol pada keypad nantinya akan merepresentasikan setiap karakter yang tertera pada keypad tersebut. Karakter angka yang ditekan nantinya akan langsung ditampilkan pada serial monitor yang terdapat pada Arduino IDE. Ketika karakter pada keypad yang ditekan ialah '*' maka buzzer akan menyala dan berbunyi, sedangkan apabila yang ditekan ialah '#' maka buzzer akan mati. Berikut komponen yang diperlukan pada percobaan ini:

Breadboard	
Keypad 4x3	
Buzzer	
Jumper Wires	

Kemudian pasang USB cable pada Arduino dengan port USB yang terdapat pada PC untuk menguplod program. Buka Arduino IDE, lalu ketikkan program berikut.

```
#include <Keypad.h>

const byte ROWS = 4; // Four rows
const byte COLS = 3; // Three columns
// Define the Keymap
char keys[ROWS][COLS] = {
  {'1','2','3'},
  {'4','5','6'},
  {'7','8','9'},
  {'#','0','*'}
}
```

```

};
// Connect keypad ROW0, ROW1, ROW2 and ROW3 to these Arduino pins.
byte rowPins[ROWS] = { 9, 8, 7, 6 };
// Connect keypad COL0, COL1 and COL2 to these Arduino pins.
byte colPins[COLS] = { 12, 11, 10 };

// Create the Keypad
Keypad kpd = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

#define buzzerpin 13

void setup()
{
  pinMode(buzzerpin, OUTPUT);
  digitalWrite(buzzerpin, HIGH);
  Serial.begin(9600);
}


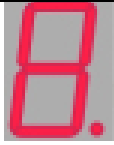


void loop()
{
  char key = kpd.getKey();
  if(key) // Check for a valid key.
  {
    switch (key)
    {
      case '*':
        digitalWrite(buzzerpin, LOW);
        break;
      case '#':
        digitalWrite(buzzerpin, HIGH);
        break;
      default:
        Serial.println(key);
    }
  }
}

```

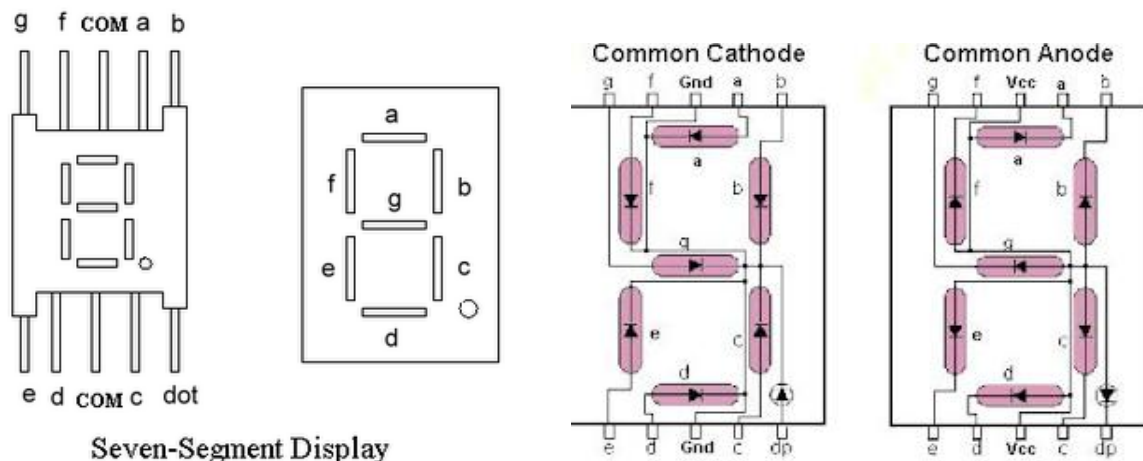
Berikutnya compile dan upload program ke dalam Arduino board. Perhatikan dan catat hasil yang terjadi, apakah sesuai dengan spesifikasi atau tidak.

Project 8 Seven Segment

Pada percobaan kali ini dilakukan uji coba terhadap lampu LED 7-segment yang dapat menampilkan representasi angka dari 0-9. Nyala pada 7-segment dapat diatur sedemikian rupa sesuai yang diinginkan, pada percobaan ini penyalaan yang terjadi ialah hitung mundur angka dari 9 ke 0 lalu kembali lagi ke angka 9. Berikut komponen yang diperlukan pada percobaan ini:

Breadboard	
LED 7-segment	
2x 1k ohm Resistor	
Jumper Wires	

Pada Umumnya seven segment terdiri 7 batang led yang disusun membentuk angka 8 (seperti pada gambar diatas), dimana setiap segmentnya terdiri dari LED yang salah satu kaki terminal lednya di jadikan satu atau yang disebut dengan common. Skema dari 7 batang led ditandai dengan huruf a - g, sebagai berikut



Common Anoda

Common Anoda merupakan bagian kaki dari anoda (+) yang dijadikan satu dan dihubungkan dengan arus positif tegangan. sedangkan untuk mengaktifkan kaki yang lainnya harus di beri tegangan negatif. atau led akan menyala jika dalam kondisi aktif low (diberi logika 0).

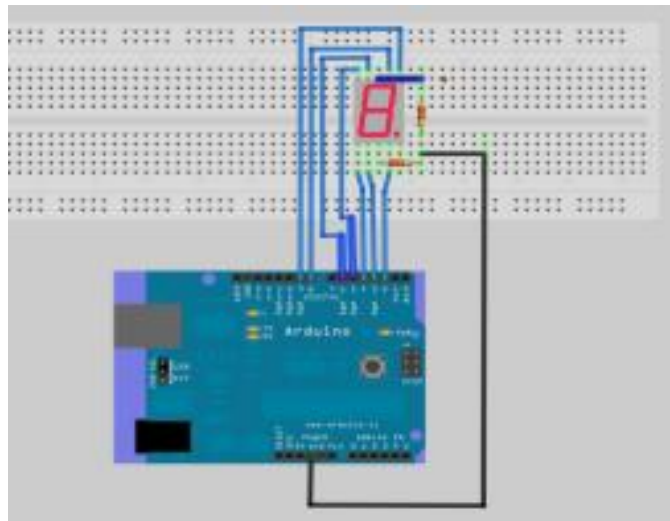
Misalkan ingin menampilkan angka 1, maka yang harus di lakukan adalah. kaki common di beri tegangan +, sedangkan kaki b dan c di beri tegangan -

Common Katoda

Common katoda ini kebalikannya dari common anoda, jadi kaki common yang disatukan adalah kaki katoda (-), sehingga untuk mengaktifkan kaki yang lain di beri tegangan (+) atau diberi logik high (1).

Berikutnya setelah komponen yang diperlukan telah terkumpul, susun rangkaian seperti terlihat pada gambar berikut:

Arduino Pin	7 Segment Pin Connection
2	7 (A)
3	6 (B)
4	4 (C)
5	2 (D)
6	1 (E)
7	9 (F)
8	10 (G)
9	5 (DP)



Kemudian pasang USB cable pada Arduino dengan port USB yang terdapat pada PC untuk mengupload program. Buka Arduino IDE, lalu ketikkan program berikut.

```
void setup() {  
  pinMode(2, OUTPUT);  
  pinMode(3, OUTPUT);  
  pinMode(4, OUTPUT);  
  pinMode(5, OUTPUT);  
  pinMode(6, OUTPUT);  
  pinMode(7, OUTPUT);  
}
```

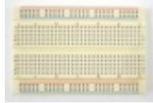



```
pinMode(8, OUTPUT);
// pinMode(9, OUTPUT);
// digitalWrite(9, 0); // start with the "dot" off
Serial.begin(9600);
}

void loop() {
    // write '0'
    Serial.println("0\n");
    digitalWrite(2, 0);
    digitalWrite(3, 0);
    digitalWrite(4, 0);
    digitalWrite(5, 0);
    digitalWrite(6, 0);
    digitalWrite(7, 0);
    digitalWrite(8, 1);
    delay(500);
    // write '2'
    Serial.println("2\n");
    digitalWrite(2, 0);
    digitalWrite(3, 0);
    digitalWrite(4, 1);
    digitalWrite(5, 0);
    digitalWrite(6, 0);
    digitalWrite(7, 1);
    digitalWrite(8, 0);
    delay(500);
    // write '3'
    Serial.println("3\n");
    digitalWrite(2, 0);
    digitalWrite(3, 0);
    digitalWrite(4, 0);
    digitalWrite(5, 0);
    digitalWrite(6, 1);
    digitalWrite(7, 1);
    digitalWrite(8, 0);
    delay(500);
}
```

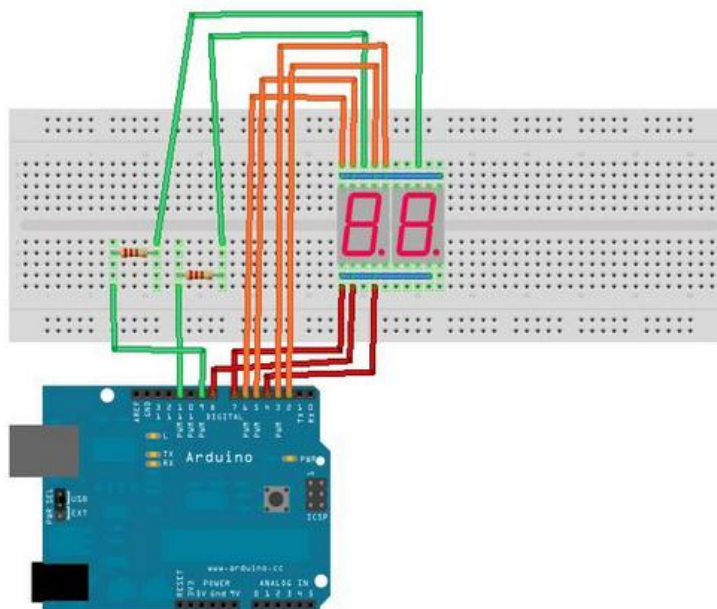
Berikutnya compile dan upload program ke dalam Arduino board. Perhatikan dan catat hasil yang terjadi, apakah sesuai dengan spesifikasi atau tidak.

Project 9 Scanning Seven Segment

Pada percobaan kali ini dilakukan uji coba terhadap dua lampu LED 7-segment yang dapat menampilkan representasi angka dari 0-9. Nyala pada 7-segment dapat diatur sedemikian rupa sesuai yang diinginkan, pada percobaan ini penyalan yang terjadi ialah hitung mundur angka dari 99 ke 00 lalu kembali lagi ke 99. Berikut komponen yang diperlukan pada percobaan ini:

Breadboard	
2 LED 7-segment	
4x 1k ohm Resistor	
Jumper Wires	

Berikutnya setelah komponen yang diperlukan telah terkumpul, susun rangkaian seperti terlihat pada gambar berikut:



Kemudian pasang USB cable pada Arduino dengan port USB yang terdapat pada PC untuk mengupload program. Buka Arduino IDE, lalu ketikkan program berikut.

```
int timer = 100;
void setup() {
  pinMode(2, OUTPUT);
  pinMode(3, OUTPUT);
  pinMode(4, OUTPUT);
  pinMode(5, OUTPUT);
  pinMode(6, OUTPUT);
  pinMode(7, OUTPUT);
  pinMode(8, OUTPUT);
  pinMode(9, OUTPUT);
  pinMode(10, OUTPUT);
  // pinMode(9, OUTPUT);
  // digitalWrite(9, 0); // start with the "dot" off
  Serial.begin(9600);
}

void loop() {
  // write '1'
  for (int i=0; i<timer; i++){
    Serial.println("1\n");
    digitalWrite(2, 1);
    digitalWrite(3, 0);
    digitalWrite(4, 0);
    digitalWrite(5, 1);
    digitalWrite(6, 1);
    digitalWrite(7, 1);
    digitalWrite(8, 1);
    digitalWrite(9, 0);
    digitalWrite(10, 1);
    delay(5);
    // write '1'
    Serial.println("1\n");
    digitalWrite(2, 1);
    digitalWrite(3, 0);
    digitalWrite(4, 0);
    digitalWrite(5, 1);
    digitalWrite(6, 1);
    digitalWrite(7, 1);
    digitalWrite(8, 1);
    digitalWrite(9, 1);
    digitalWrite(10, 0);
    delay(5);
  }
  for (int i=0; i<timer; i++){
    Serial.println("2\n");
    digitalWrite(2, 0);
    digitalWrite(3, 0);
    digitalWrite(4, 1);
    digitalWrite(5, 0);
    digitalWrite(6, 0);
    digitalWrite(7, 1);
    digitalWrite(8, 0);
    digitalWrite(9, 0);
    digitalWrite(10, 1);
    delay(5);
    // write '2'
    Serial.println("2\n");
    digitalWrite(2, 0);
    digitalWrite(3, 0);
    digitalWrite(4, 1);
    digitalWrite(5, 0);
    digitalWrite(6, 0);
    digitalWrite(7, 1);
  }
```

```
digitalWrite(8, 0);
digitalWrite(9, 1);
digitalWrite(10, 0);
delay(5);
}
for (int i=0; i<timer; i++){
  Serial.println("3\n");
  digitalWrite(2, 0);
digitalWrite(3, 0);
digitalWrite(4, 0);
digitalWrite(5, 0);
digitalWrite(6, 1);
digitalWrite(7, 1);
digitalWrite(8, 0);
digitalWrite(9, 0);
digitalWrite(10, 1);
delay(5);
  // write '3'
  Serial.println("2\n");
digitalWrite(2, 0);
digitalWrite(3, 0);
digitalWrite(4, 0);
digitalWrite(5, 0);
digitalWrite(6, 1);
digitalWrite(7, 1);
digitalWrite(8, 0);
digitalWrite(9, 1);
digitalWrite(10, 0);
delay(5);

}
}
```

Berikutnya compile dan uplod program ke dalam Arduino board. Perhatikan dan catat hasil yang terjadi, apakah sesuai dengan spesifikasi atau tidak.

Referensi

<http://www.EarthshineElectronics.com/ArduinoStarterKitManual>

<http://arduino.cc/en/Tutorial/HomePage>

<http://arduino.cc/en/Tutorial/LiquidCrystal>

<http://playground.arduino.cc/Main/LiquidCrystal>

<http://playground.arduino.cc/Main/KeypadTutorial>

<http://www.arduino.web.id/>

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

<http://www.instructables.com/id/Seven-Segment-Display-Tutorial/step3/Testing-The-Display/>

<http://www.instructables.com/id/Arduino-7-segment-countdown-timer/step4/The-programming/>

<http://www.instructables.com/id/Access-control-with-Arduino-Keypad-4x4-Servo/step5/And-now-the-programming/>

<http://bradsduino.blogspot.com/2012/10/4-x4-universal-16-key-keypad-arduino.html>

<http://insansainsprojects.wordpress.com/2011/04/18/sensor-suhu-dengan-arduino/>

<http://ngooprek.com/club/blog/2012/07/18/pengenalan-arduino/>