

# OPERASI LED DAN LCD

## 1.1 Percobaan Blinking LED

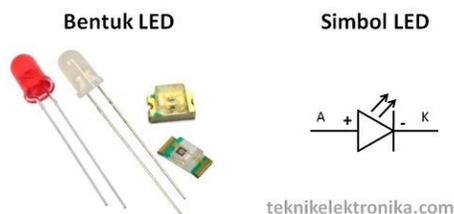
### a. Tujuan Percobaan :

1. Mahasiswa dapat memahami prinsip kerja LED
2. Mahasiswa dapat memahami mengoperasikan pin digital arduino uno sebagai output

### b. Teori dasar

**Pengertian LED (Light Emitting Diode) dan Cara Kerjanya** – Light Emitting Diode atau sering disingkat dengan LED adalah komponen elektronika yang dapat memancarkan cahaya monokromatik ketika diberikan tegangan maju. LED merupakan keluarga Dioda yang terbuat dari bahan semikonduktor. Warna-warna Cahaya yang dipancarkan oleh LED tergantung pada jenis bahan semikonduktor yang dipergunakannya. LED juga dapat memancarkan sinar inframerah yang tidak tampak oleh mata seperti yang sering kita jumpai pada Remote Control TV ataupun Remote Control perangkat elektronik lainnya.

Bentuk LED mirip dengan sebuah bohlam (bola lampu) yang kecil dan dapat dipasangkan dengan mudah ke dalam berbagai perangkat elektronika. Berbeda dengan Lampu Pijar, LED tidak memerlukan pembakaran filamen sehingga tidak menimbulkan panas dalam menghasilkan cahaya. Oleh karena itu, saat ini LED (Light Emitting Diode) yang bentuknya kecil telah banyak digunakan sebagai lampu penerang dalam LCD TV yang mengganti lampu tube.



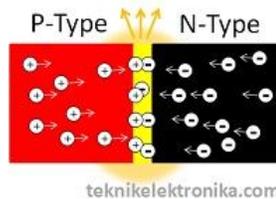
**Simbol dan Bentuk LED (Light Emitting Diode)**

### **Cara Kerja LED (Light Emitting Diode)**

Seperti dikatakan sebelumnya, LED merupakan keluarga dari Dioda yang terbuat dari Semikonduktor. Cara kerjanya pun hampir sama dengan Dioda yang memiliki dua kutub yaitu

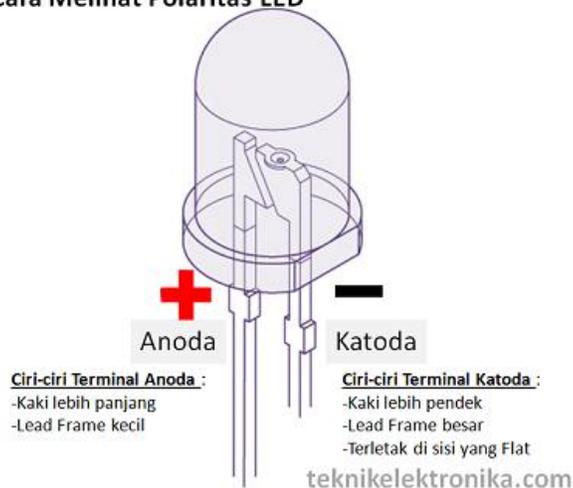
kutub Positif (P) dan Kutub Negatif (N). LED hanya akan memancarkan cahaya apabila dialiri tegangan maju (bias forward) dari Anoda menuju ke Katoda.

LED terdiri dari sebuah chip semikonduktor yang di doping sehingga menciptakan junction P dan N. Yang dimaksud dengan proses doping dalam semikonduktor adalah proses untuk menambahkan ketidakmurnian (impurity) pada semikonduktor yang murni sehingga menghasilkan karakteristik kelistrikan yang diinginkan. Ketika LED dialiri tegangan maju atau bias forward yaitu dari Anoda (P) menuju ke Katoda (K), Kelebihan Elektron pada N-Type material akan berpindah ke wilayah yang kelebihan Hole (lubang) yaitu wilayah yang bermuatan positif (P-Type material). Saat Elektron berjumpa dengan Hole akan melepaskan photon dan memancarkan cahaya monokromatik (satu warna).



LED atau Light Emitting Diode yang memancarkan cahaya ketika dialiri tegangan maju ini juga dapat digolongkan sebagai Transduser yang dapat mengubah Energi Listrik menjadi Energi Cahaya.

### Cara Melihat Polaritas LED



### Cara Mengetahui Polaritas LED

Untuk mengetahui polaritas terminal Anoda (+) dan Katoda (-) pada LED. Kita dapat melihatnya secara fisik berdasarkan gambar diatas. Ciri-ciri Terminal Anoda pada LED adalah kaki yang lebih panjang dan juga Lead Frame yang lebih kecil. Sedangkan ciri-ciri Terminal Katoda adalah Kaki yang lebih pendek dengan Lead Frame yang besar serta terletak di sisi yang Flat.

### Warna-warna LED (Light Emitting Diode)

Saat ini, LED telah memiliki beranekaragam warna, diantaranya seperti warna merah, kuning, biru, putih, hijau, jingga dan infra merah. Keanekaragaman Warna pada LED tersebut tergantung pada wavelength (panjang gelombang) dan senyawa semikonduktor yang dipergunakannya. Berikut ini adalah Tabel Senyawa Semikonduktor yang digunakan untuk menghasilkan variasi warna pada LED :

Bahan Semikonduktor	Wavelength	Warna
Gallium Arsenide (GaAs)	850-940nm	Infra Merah
Gallium Arsenide Phosphide (GaAsP)	630-660nm	Merah
Gallium Arsenide Phosphide (GaAsP)	605-620nm	Jingga
Gallium Arsenide Phosphide Nitride (GaAsP:N)	585-595nm	Kuning
Aluminium Gallium Phosphide (AlGaP)	550-570nm	Hijau
Silicon Carbide (SiC)	430-505nm	Biru
Gallium Indium Nitride (GaInN)	450nm	Putih

### Tegangan Maju (Forward Bias) LED

Masing-masing Warna LED (Light Emitting Diode) memerlukan tegangan maju (Forward Bias) untuk dapat menyalakannya. Tegangan Maju untuk LED tersebut tergolong rendah sehingga memerlukan sebuah Resistor untuk membatasi Arus dan Tegangannya agar tidak merusak LED yang bersangkutan. Tegangan Maju biasanya dilambangkan dengan tanda  $V_f$ .

Warna	Tegangan Maju @20mA
Infra Merah	1,2V
Merah	1,8V

Jingga	2,0V
Kuning	2,2V
Hijau	3,5V
Biru	3,6V
Putih	4,0V

### **Kegunaan LED dalam Kehidupan sehari-hari**

Teknologi LED memiliki berbagai kelebihan seperti tidak menimbulkan panas, tahan lama, tidak mengandung bahan berbahaya seperti merkuri, dan hemat listrik serta bentuknya yang kecil ini semakin populer dalam bidang teknologi pencahayaan. Berbagai produk yang memerlukan cahaya pun mengadopsi teknologi Light Emitting Diode (LED) ini. Berikut ini beberapa pengaplikasiannya LED dalam kehidupan sehari-hari.

1. Lampu Penerangan Rumah
2. Lampu Penerangan Jalan
3. Papan Iklan (Advertising)
4. Backlight LCD (TV, Display Handphone, Monitor)
5. Lampu Dekorasi Interior maupun Exterior
6. Lampu Indikator
7. Pemancar Infra Merah pada Remote Control (TV, AC, AV Player)

#### **c. Alat dan bahan :**

1. Board Arduino + Kabel
2. Breadboard
3. LED
4. Resistor  $220 \Omega$   $\frac{1}{2}$  watt
5. Kabel jumper

#### **d. Prosedur percobaan :**

- (i) Instalasi software Arduino
  - a. Download software arduino

File instalasi software Arduino dapat diperoleh pada alamat situs web di bawah ini yang tersedia untuk sistem operasi Windows, Mac dan Linux:

<http://arduino.cc/en/Main/Software>



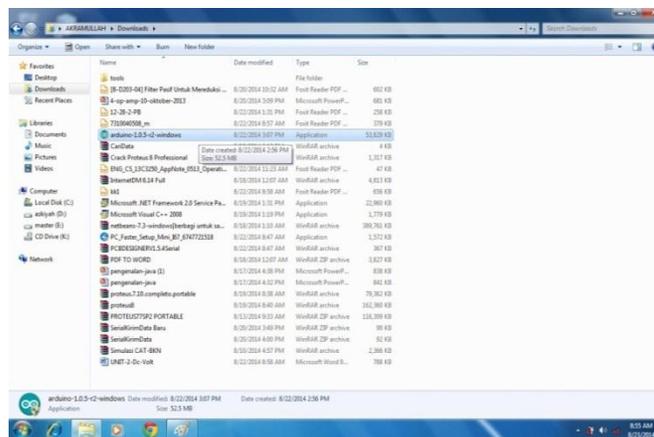
b. Cari file installer pada web tersebut seperti gambar dibawah:



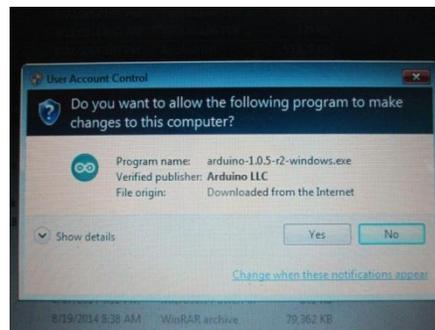
c. Klik link “ Windows Installer” untuk men-download file master IDE arduino sehingga muncul seperti gambar dibawah :



d. Setelah proses download selesai, buka folder dimana file tersebut disimpan



e. "Double klik" file tersebut untuk menginstal IDE dari arduino,



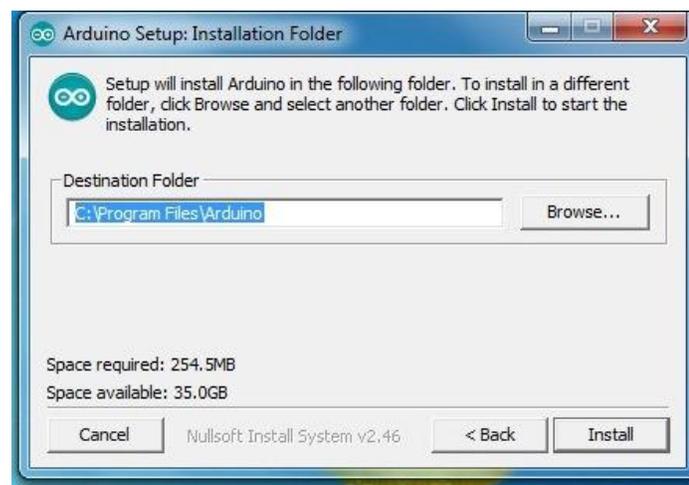
f. Jika muncul tampilan seperti gambar diatas klik "Yes",



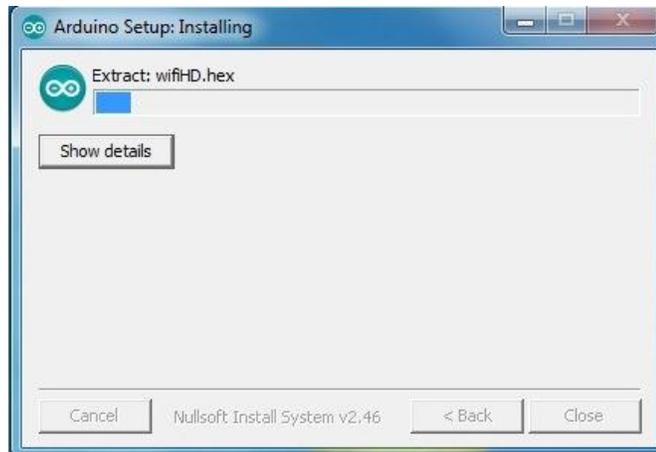
g. Kemudian klik **"I Agree"**, untuk melanjutkan proses instalasi



h. Kemudian klik **"Next >"**



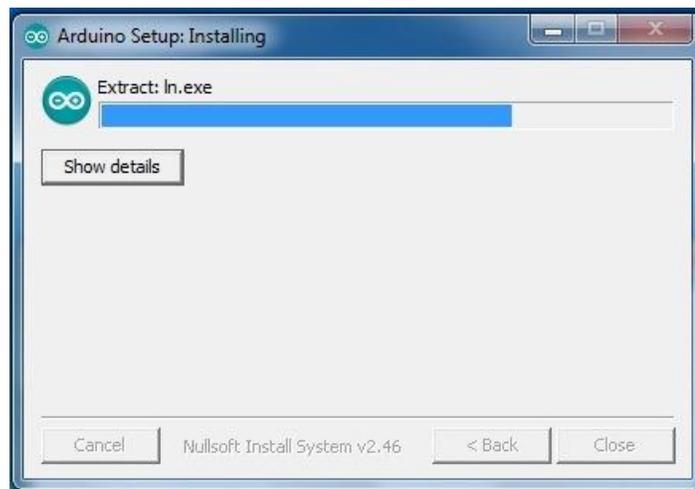
i. Kemudian klik **"Install"** sehingga tampil seperti gambar dibawah :



- j. Jika muncul tampilan seperti gambar dibawah klik **"install"**

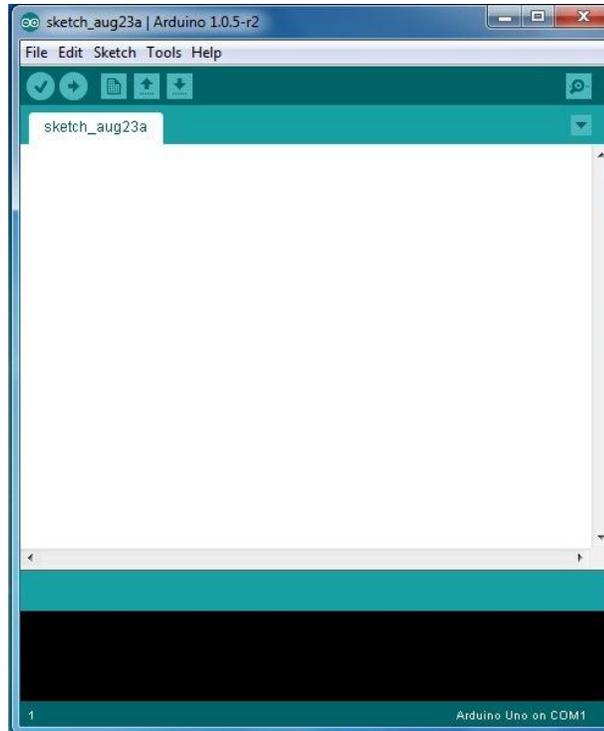


- k. Setelah itu proses instalasi akan berlanjut seperti gambar dibawah :



- l. Setelah proses **instalasi selesai** muncul tampilan seperti gambar dibawah:  
m. Kemudian klik **"close"**



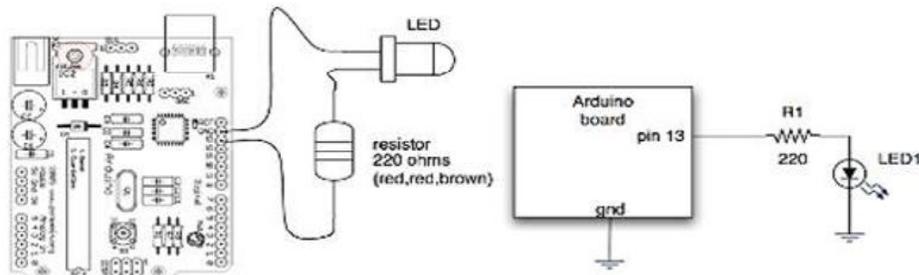


p. Gambar diatas menunjukkan tampilan sketch arduino, dengan demikian proses instalasi telah selesai dengan baik.

(ii) Langkah-langkah untuk membuat Blinking LED :

a. Pasang led pada breadboard.

*Pemasangan LED harus diperhatikan polaritasnya. Kaki yang panjang mengarah ke kutub Positif (+), sedangkan kaki yang pendek atau kaki pada permukaan LED yg datar, mengarah ke kutub negative (-) atau Ground (Gnd)*



b. Hubungkan LED dengan arduino pada Arduino seperti gambar diatas

c. Download program berikut ke board arduino

**Sketch:**

```
int led 1 = 13; // lampu LED internal pada pin 13 Arduino

void setup()
{
  pinMode(led1, OUTPUT); // pin 13 sebagai output
}

void loop()
{
  digitalWrite(led1, HIGH); // led On
  delay(1000); // tunggu 1 detik
  digitalWrite(led1, LOW); // led Off
  delay(1000); // tunggu 1 detik
}
```

- d. Amati LED setelah program diupload
- e. Tugas Buatlah led berkedip setiap 3 detik

## 1.2 Traffic Light

**a. Tujuan Percobaan :**

1. Mahasiswa dapat memahami prinsip kerja traffic light
2. Mahasiswa dapat memahami mengoperasikan arduino untuk mengatur traffic light

**b. Teori dasar**

Traffic Light (lampu lalu lintas) ditemukan pada tahun 1868 oleh seorang insinyur kereta api bernama John Peake Knight. Namun bukan lampu lalu lintas yang kita tahu sekarang, tetapi adalah lentera gas-powered revolving dengan merah dan lampu hijau. Penemuan Knight adalah serupa dengan sinyal kereta api saat itu. Lampu lalu lintas awalnya ditempatkan di dekat London's House of Commons, di persimpangan Jalan George dan Jembatan.



Pada tahun 1910, Earnest Serrine Chicago, menggunakan kata-kata STOP (berhenti) dan MOVE (melanjutkan) untuk digunakan pada lampu lalu lintas tanpa adanya cahaya. Lalu pada tahun 1912, Lester Wire dari Salt Lake City, telah menemukan lampu lalu lintas listrik yang menggunakan lampu merah dan hijau. Tahun berikutnya, James Hoge membuat sistem lampu lalu lintas menggunakan lampu listrik secara manual. Itu dipasang di Cleveland, Ohio pada tahun 1914. Lampu lalu lintas tersebut selain menggunakan lampu listrik, juga menampilkan kata-kata STOP dan MOVE. Kemudian William Ghiglieri dari San Francisco, California pada tahun 1917 menemukan rancangan lampu lalu lintas yang dapat dioperasikan secara manual atau otomatis.

Lampu kuning ditambahkan pada tahun 1920 oleh William Potts, seorang polisi Detroit. Dia benar-benar menemukan beberapa sistem lampu lalu lintas, termasuk cara menggantung empat sistem. Semua hasil temuan tersebut tidak dipatenkan. Hingga pada akhirnya orang pertama yang mengajukan permohonan paten untuk memproduksi lampu lalu lintas yang murah, adalah Garrett Morgan, yang menerima paten pada tahun 1923.



Seperti kita ketahui bahwa lampu lalu lintas yang sering kita jumpai sehari-hari memiliki 3 warna utama yaitu merah, kuning dan hijau. Warna-warna ini tentunya tidak diambil dan digunakan dengan sembarang. Dengan kata lain, warna ini diambil karena memiliki arti-arti tertentu di dalamnya. Dan berikut arti warna pada lampu lalu lintas dan sejarah singkat tentang bagaimana warna tersebut digunakan sebagai simbol untuk lampu lalu lintas.

### **Warna Merah**

Warna merah artinya larangan atau stop atau bahaya. Warna merah identik dengan warna darah, sejak jaman dahulu manusia sering berperang untuk memperebutkan sesuatu dan Berbagai hal lainnya. Berperang berarti saling membunuh, saling melukai dan saling menumpahkan darah. Banyak para korban perang tersebut yang terluka bahkan ada yang tewas. Baik korban terluka maupun tewas pasti tubuhnya akan mengeluarkan darah. Seperti kita ketahui bahwa semua manusia memiliki darah yang berwarna merah (jika darahnya bukan merah, pastinya bukan manusia. Lalu apa? "IH SEREM"). Dan jika manusia terluka pasti akan mengeluarkan darah dan terasa sakit. Dengan perkembangan jaman, ada suatu kelompok manusia yang anti dengan peperangan, dan menyatakan bahwa perang itu membahayakan, maka disepakati dan dibuatlah aturan untuk tidak saling berperang, melukai dan saling membunuh sesama manusia karena sangat membahayakan. Dengan tahapan aturan tersebut, yaitu awas bisa melukai, awas bahaya, dilarang melukai atau bahaya membuat warna merah dijadikan simbol untuk hal yang membahayakan atau larangan hingga sekarang ini.

### **Warna Kuning**

Warna Kuning artinya hati-hati, waspada atau pelan-pelan. Warna kuning identik dengan warna api, dimana api memiliki sifat antara dua pilihan, yaitu api kecil yang bisa di kendalikan, dan api besar yang sulit dikendalikan dan bisa membahayakan. Pengertian warna kuning memiliki resiko bisa aman dan bisa tidak aman atau berbahaya. Begitu juga api, baik api kecil maupun api besar memiliki sifat panas, dan manusia akan selalu hati-hati dengan api. Jaman dulu di dalam peperangan manusia selalu menggunakan api, baik untuk senjata, sinyal komunikasi, simbol dan penerangan. Dalam situasi berperang, prajurit selalu dituntut untuk waspada dan hati-hati terhadap gerakan musuhnya, apalagi di malam hari. Mereka akan menggunakan api untuk segala sesuatunya, mereka akan mengamati pergerakan musuhnya dengan melihat api yang digunakan, sehingga bila ada gerakan api atau obor musuhnya mereka akan bersiap-siap dan waspada untuk menghadapi serangan musuhnya. Sehingga sampai sekarang warna kuning telah disepakati sebagai simbol untuk hati-hati, waspada atau siap-siap. Warna kuning bisa juga diidentikkan warna daun yang sudah tua yang sebentar lagi daun tersebut akan gugur. Jadi, warna kuning diartikan sebagai warna transisi atau peralihan.

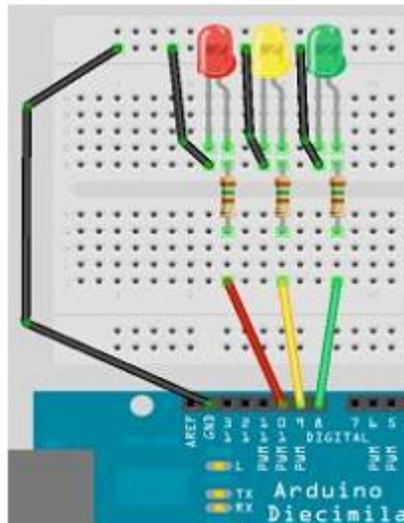
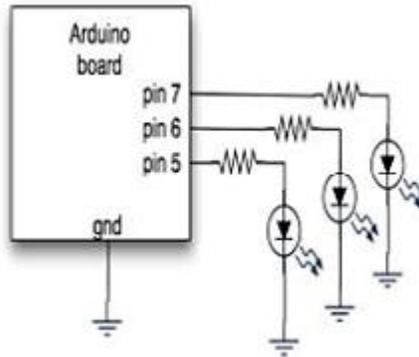
### **Warna Hijau**

Warna Hijau artinya bebas atau boleh berjalan atau aman. Warna hijau identik dengan warna alam, yaitu hutan terutama warna daun pada tumbuh-tumbuhan. Hampir semua warna daun tumbuh-tumbuhan memiliki warna yang sama yaitu hijau, meskipun sebagian kecil tumbuh-tumbuhan memiliki daun yang berwarna lain. Lantas kenapa warna hijau diidentikkan dengan kebebasan? Banyak tumbuh-tumbuhan di dunia ini berbeda jenisnya, sifatnya, ragamnya, corak dan bentuknya, golongannya serta macam-macam yang lainnya. Tetapi hampir semua daunnya memiliki warna hijau, arti kata semua bebas untuk berwarna hijau, dan tak satu pun ada yang melarangnya, baik dari tumbuh-tumbuhan itu sendiri dan yang berasal dari jenis yang berbeda. Jadi warna hijau memiliki arti suatu kebebasan. Warna hijau juga memiliki sifat sensitif terhadap penglihatan kita, memiliki warna yang menyegarkan mata terutama untuk terapi warna. Sehingga warna hijau tersebut sangat aman bagi mata kita. Dan akhirnya warna hijau disepakati sebagai simbol untuk kebebasan dan aman atau boleh dan diperbolehkan.

#### **c. Alat dan bahan :**

1. Board Arduino + Kabel

2. Breadboard
  3. LED
  4. Resistor 220  $\Omega$  ½ watt
  5. Kabel jumper
- d. Prosedur percobaan :.
1. Pasanglah LED seperti gambar berikut



Contoh pemasangan LED menggunakan breadboard pada pin 8,9,10 Arduino

2. Upload program berikut pada arduino

**Sketch:**

```
int redPin = 7; // Lampu Merah
int yellowPin = 6; // Lampu Kuning
int greenPin = 5; // Lampu Hijau
```

```
void setup()
{
pinMode(redPin, OUTPUT);
pinMode(yellowPin, OUTPUT);
pinMode(greenPin, OUTPUT);
}

void loop()
{
digitalWrite(redPin, HIGH); // Lampu Merah On
digitalWrite(yellowPin, LOW);
digitalWrite(greenPin, LOW);
delay(3000); // Jeda 3 detik
digitalWrite(redPin, LOW); // Lampu Kuning On
digitalWrite(yellowPin, HIGH);
digitalWrite(greenPin, LOW);
delay(3000); // Jeda 3 detik
digitalWrite(redPin, LOW); // Lampu Hijau On
digitalWrite(yellowPin, LOW);
digitalWrite(greenPin, HIGH);
delay(3000); // Jeda 3 detik
}
```

*Catatan:*

- *Buatlah lampu lalu lintas dengan urutan: Merah - Kuning (berkedip 3x) - Hijau*

*Petunjuk:*

**Baris program untuk Lampu kuning:**

```
digitalWrite(redPin, LOW); // Lampu Kuning On
digitalWrite(yellowPin, HIGH);
digitalWrite(greenPin, LOW);
```

```
delay(3000); // Jeda 3 detik
diganti:
for( int i=1; i<=3; i++ )
{
digitalWrite(yellowPin, HIGH);
delay(1000);
digitalWrite(yellowPin, LOW);
delay(1000);
}
```

3. Amati yang terjadi
4. Buatlah traffic light simpang empat

### 1.3 Counter

#### a. Tujuan Percobaan :

1. Mahasiswa dapat memahami prinsip kerja counter
2. Mahasiswa dapat memahami prinsip kerja *seven segment*
3. Mahasiswa dapat membangun counter dengan arduino

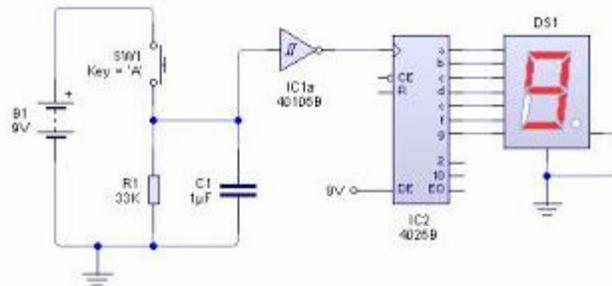
#### b. Teori dasar

##### Counter

Rangkaian Counter adalah rangkaian yang dapat berfungsi sebagai penghitung angka secara cepat, baik itu penghitungan maju maupun mundur. Penghitungan maju adalah hitungan yang di mulai dari angka yang kecil ke angka yang lebih besar, sedangkan penghitungan mundur adalah hitungan yang dilakukan dari angka yang besar ke angka yang kecil. Dalam penghitungan bisa mencapai jumlah yang tidak terbatas tergantung dari rangkaian yang kita buat dan juga kebutuhan.

Counter biasanya disebut sebagai pencacah yang tersusun dari sederet flip flop dan kemudian diperbarui sedemikian rupa dengan menggunakan karnough, sehingga angka yang masuk nantinya dapat dihitung sesuai rancangan yang kita buat. Dalam penyusunan rangkaian counter terdiri atas semua jenis flip flop, tergantung model dari masing-masing flip flop itu sendiri.

Jenis dari rangkaian pencacah (counter) dibedakan menjadi dua, yaitu rangkaian pencacah naik (up counter) dan rangkaian pencacah turun (down counter). Yang dimaksud pencacah naik atau up counter adalah cacahan dari kecil ke besar kemudian kembali ke cacahan awal secara otomatis. Sedangkan pencacah turun atau down counter adalah pencacah dari besar ke arah yang kecil kemudian kembali ke cacahan yang awal.



Contoh dari rangkaian counter diatas hanya menggunakan IC decade counter dan satu buah seven segmen sehingga hanya mewakili angka satuan. Apabila anda ingin membuat fungsi yang lebih banyak, anda hanya tinggal menambahkan IC dan seven segmennya sesuai dengan kebutuhan yang anda inginkan. IC yang digunakan sebagai pencacah merupakan IC 4026.

Di dunia ini ada banyak sekali IC pencacah yang dapat anda gunakan untuk membuat rangkaian counter, baik itu IC dari keluarga TTL maupun CMOS. Perbedaan dari kedua IC yaitu dari angka awal dimana untuk keluarga IC TTL mempunyai awal seri 74 sedangkan untuk keluarga IC CMOS adalah 40. IC yang paling sering digunakan pada setiap rangkaian adalah IC dari jenis TTL.

Pada gambar skema di atas, IC yang digunakan adalah jenis CMOS. Karena IC jenis CMOS bisa menggunakan tegangan maksimal 15 volt, sedangkan pada jenis TTL hanya mensupply tegangan maksimal 5 volt. Keuntungan lain dari penggunaan IC 4026 anda tidak perlu lagi menggunakan IC decoder sebagai interface seven segmen. Karena output yang dihasilkan sudah sesuai dengan kondisi dan fungsi dari seven segmen.

Skema rangkaian counter diatas sengaja menggunakan gerbang penyulut schmitt trigger karena berfungsi sebagai peredam bouncing dari hentakan saklar mekanik. Anda juga bisa menggunakan gerbang schmitt trigger dan menghubungkan langsung saklar input ke pin clock IC 4026.

### Seven Segment

**Pengertian Seven Segment Display** – *Seven Segment Display* (7 Segment Display) dalam bahasa Indonesia disebut dengan Layar Tujuh Segmen adalah komponen Elektronika yang dapat menampilkan angka desimal melalui kombinasi-kombinasi segmennya. *Seven Segment Display* pada umumnya dipakai pada Jam Digital, Kalkulator, Penghitung atau Counter Digital, Multimeter Digital dan juga Panel Display Digital seperti pada Microwave Oven ataupun Pengatur Suhu Digital. *Seven Segment Display* pertama diperkenalkan dan dipatenkan pada tahun 1908 oleh Frank. W. Wood dan mulai dikenal luas pada tahun 1970-an setelah aplikasinya pada LED (Light Emitting Diode).

*Seven Segment Display* memiliki 7 Segmen dimana setiap segmen dikendalikan secara ON dan OFF untuk menampilkan angka yang diinginkan. Angka-angka dari 0 (nol) sampai 9 (Sembilan) dapat ditampilkan dengan menggunakan beberapa kombinasi Segmen. Selain 0 – 9, *Seven Segment Display* juga dapat menampilkan Huruf Hexadecimal dari A sampai F. Segmen atau elemen-elemen pada Seven Segment Display diatur menjadi bentuk angka “8” yang agak miring ke kanan dengan tujuan untuk mempermudah pembacaannya. Pada beberapa jenis Seven Segment Display, terdapat juga penambahan “titik” yang menunjukkan angka koma decimal. Terdapat beberapa jenis *Seven Segment Display*, diantaranya adalah Incandescent bulbs, Fluorescent lamps (FL), Liquid Crystal Display (LCD) dan Light Emitting Diode (LED).

#### **LED 7 Segmen (Seven Segment LED)**

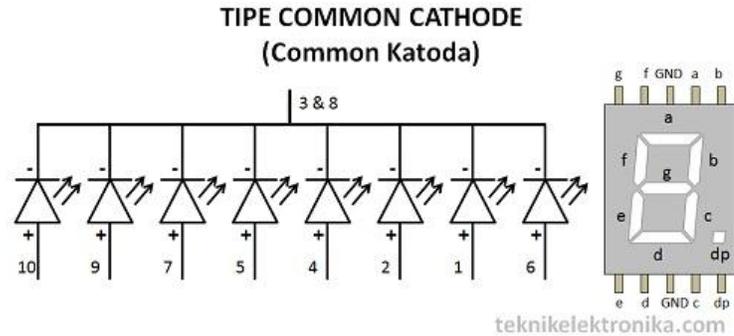
Salah satu jenis Seven Segment Display yang sering digunakan oleh para penghobi Elektronika adalah 7 Segmen yang menggunakan LED (Light Emitting Diode) sebagai penerangnya. LED 7 Segmen ini umumnya memiliki 7 Segmen atau elemen garis dan 1 segmen titik yang menandakan “koma” Desimal. Jadi Jumlah keseluruhan segmen atau elemen LED sebenarnya adalah 8. Cara kerjanya pun boleh dikatakan mudah, ketika segmen atau elemen tertentu diberikan arus listrik, maka Display akan menampilkan angka atau digit yang diinginkan sesuai dengan kombinasi yang diberikan.

Terdapat 2 Jenis LED 7 Segmen, diantaranya adalah “LED 7 Segmen common Cathode” dan “LED 7 Segmen common Anode”.

#### **LED 7 Segmen Tipe Common Cathode (Katoda)**

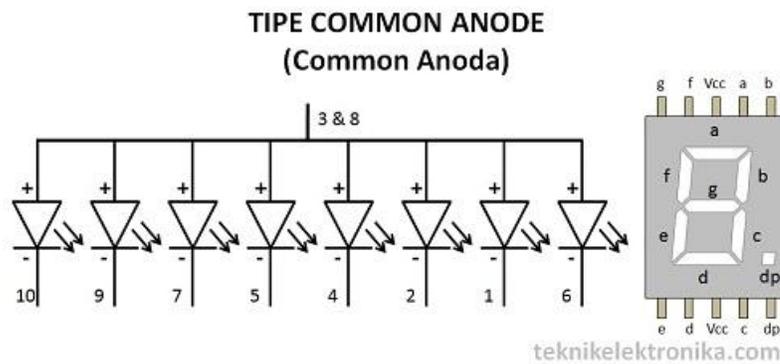
Pada LED 7 Segmen jenis Common Cathode (Katoda), Kaki Katoda pada semua segmen LED adalah terhubung menjadi 1 Pin, sedangkan Kaki Anoda akan menjadi Input untuk masing-masing Segmen LED. Kaki Katoda yang terhubung menjadi 1 Pin ini merupakan

Terminal Negatif (-) atau Ground sedangkan Signal Kendali (Control Signal) akan diberikan kepada masing-masing Kaki Anoda Segmen LED.



### LED 7 Segmen Tipe Common Anode (Anoda)

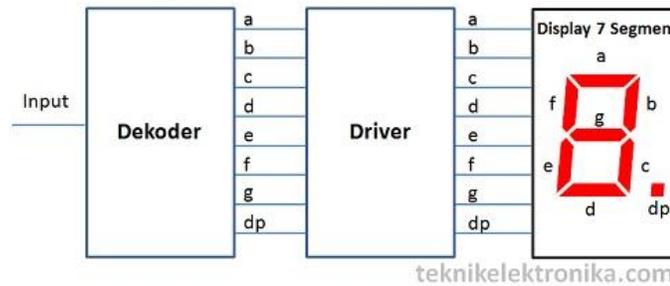
Pada LED 7 Segmen jenis Common Anode (Anoda), Kaki Anoda pada semua segmen LED adalah terhubung menjadi 1 Pin, sedangkan kaki Katoda akan menjadi Input untuk masing-masing Segmen LED. Kaki Anoda yang terhubung menjadi 1 Pin ini akan diberikan Tegangan Positif (+) dan Signal Kendali (control signal) akan diberikan kepada masing-masing Kaki Katoda Segmen LED.



### Prinsip Kerja Dasar Driver System pada LED 7 Segmen

Berikut ini adalah Blok Diagram Dasar untuk mengendalikan LED 7 Segmen :

### Blok Diagram Dasar SEVEN SEGMENT DISPLAY



Blok Dekoder pada diagram diatas mengubah sinyal Input yang diberikan menjadi 8 jalur yaitu "a" sampai "g" dan poin decimal (koma) untuk meng-ON-kan segmen sehingga menghasilkan angka atau digit yang diinginkan. Contohnya, jika output dekoder adalah a, b, dan c, maka Segmen LED akan menyala menjadi angka "7". Jika Sinyal Input adalah berbentuk Analog, maka diperlukan ADC (Analog to Digital Converter) untuk mengubah sinyal analog menjadi Digital sebelum masuk ke Input Dekoder. Jika Sinyal Input sudah merupakan Sinyal Digital, maka Dekoder akan menanganinya sendiri tanpa harus menggunakan ADC.

Fungsi daripada Blok Driver adalah untuk memberikan arus listrik yang cukup kepada Segmen/Elemen LED untuk menyala. Pada Tipe Dekoder tertentu, Dekoder sendiri dapat mengeluarkan Tegangan dan Arus listrik yang cukup untuk menyalakan Segmen LED maka Blok Driver ini tidak diperlukan. Pada umumnya Driver untuk menyalakan 7 Segmen ini adalah terdiri dari 8 Transistor Switch pada masing-masing elemen LED.

**Tabel Pengaktifan Seven Segment Display**

ANGKA	h	g	f	e	d	c	b	a
0	0	0	1	1	1	1	1	1
1	0	0	0	0	0	1	1	0
2	0	1	0	1	1	0	1	1
3	0	1	0	0	1	1	1	1
4	0	1	1	0	0	1	1	0
5	0	1	1	0	1	1	0	1
6	0	1	1	1	1	1	0	1
7	0	0	0	0	0	1	1	1
8	0	1	1	1	1	1	1	1
9	0	1	1	0	1	1	1	1

**Catatan :**

1 = ON (High)

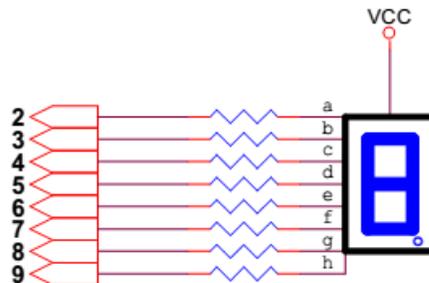
0 = OFF (Low)

**c. Alat dan bahan :**

1. Board Arduino + Kabel
2. Breadboard
3. 7 Segment
4. Resistor 220  $\Omega$  ½ watt
5. Kabel jumper

**d. Prosedur percobaan :**

1. Hubungkan seven segment dengan pin digital pada arduino seperti gambar berikut



2. Upload program berikut :

**Sketch :**

```
byte seven_seg_digits[10][7] = { { 0,0,0,0,0,0,1 },// = 0
```

```
{ 1,0,0,1,1,1,1 },// = 1
```

```
{ 0,0,1,0,0,1,0 },// = 2
```

```
{ 0,0,0,0,1,1,0 },// = 3
```

```
{ 1,0,0,1,1,0,0 },// = 4
```

```
{ 0,1,0,0,1,0,0 },// = 5
```

```
{ 0,1,0,0,0,0,0 },// = 6
```

```
{ 0,0,0,1,1,1,1 },// = 7
```

```
{ 0,0,0,0,0,0,0 },// = 8
```

```
{ 0,0,0,0,1,0,0 }// = 9
```

```
};  
  
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,HIGH);  
  nilai=10;  
}  
  
void sevenSegWrite(byte segment) {  
  byte pin = 2;  
  for (byte segCount = 0; segCount < 7; ++segCount){  
    digitalWrite(pin, seven_seg_digits[segment][segCount]);  
    ++pin;  
  }  
}  
  
void loop() {  
  nilai--;  
  sevenSegWrite(nilai);  
  if(nilai==0) nilai=10;  
  delay(1000);  
}
```

## 1.4 Menampilkan karakter di LCD

### a. Tujuan Percobaan :

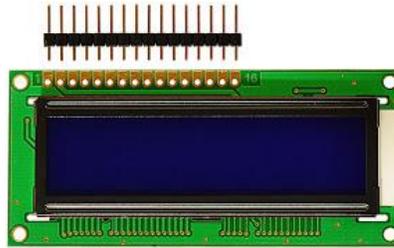
1. Mahasiswa dapat memahami prinsip kerja LCD karakter 16 x 2
2. Mahasiswa dapat memahami proses menampilkan tulisan pada LCD

### b. Teori dasar

**LCD (Liquid Crystal Display)** adalah suatu jenis media tampilan yang menggunakan kristal cair sebagai penampil utama. LCD sudah digunakan di berbagai bidang, misalnya dalam alat-alat elektronik, seperti televisi, kalkulator ataupun layar komputer.

Pada Percobaan kali ini adalah dengan menggunakan LCD 16x2 yang artinya LCD tersebut terdiri dari 16 kolom dan 2 baris karakter (tulisan). yang perlu di persiapkan adalah sebagai berikut

- ✓ LCD 16x2
- ✓ Arduino UNO (Type Lain)
- ✓ Kabel dan Konektor



Berikut adalah pin dari LCD 16 X 2

No Kaki/Pin	Nama	Keterangan
1	VCC	+5V
2	GND	0V
3	VEE	Tegangan Kontras LCD
4	RS	Register Select
5	R/W	1 = Read, 0 = Write
6	E	Enable Clock LCD
7	D0	Data Bus 0
8	D1	Data Bus 1

9	D2	Data Bus 2
10	D3	Data Bus 3
11	D4	Data Bus 4
12	D5	Data Bus 5
13	D6	Data Bus 6
14	D7	Data Bus 7
15	Anoda	Tegangan backlight positif
16	Katoda	tegangan backlight Negatif

Pin LCD nomor 4 (RS) merupakan Register Selector yang berfungsi untuk memilih Register Kontrol atau Register Data. Register kontrol digunakan untuk mengkonfigurasi LCD. Register Data digunakan untuk menulis data karakter ke memori display LCD.

Pin LCD nomor 5 (R/W) digunakan untuk memilih aliran data apakah READ ataukah WRITE. Karena kebanyakan fungsi hanya untuk membaca data dari LCD dan hanya perlu menulis data saja ke LCD, maka kaki ini dihubungkan ke GND (WRITE).

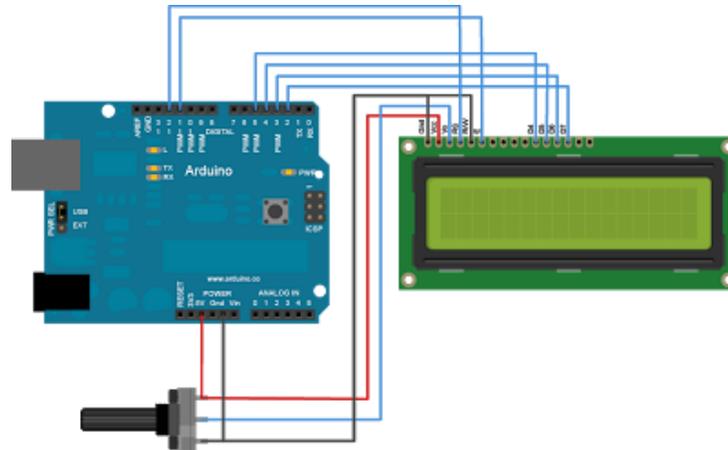
Pin LCD nomor 6 (ENABLE) digunakan untuk mengaktifkan LCD pada proses penulisan data ke Register Kontrol dan Register Data LCD.

**c. Alat dan bahan :**

1. Board Arduino + Kabel
2. Breadboard
3. LCD 16 x 2
4. Kabel jumper

**d. Prosedur percobaan :**

1. Hubungkan LCD dengan Arduino seperti langkah berikut :
  - ✓ Pin RS (kaki 4) di sambungkan dengan pin arduino digital pin 12
  - ✓ Pin E (kaki 6) di sambungkan dengan pin arduino digital pin 11
  - ✓ Pin D4 (kaki 11) di sambungkan dengan pin arduino digital pin 5
  - ✓ Pin D5 (kaki 12) di sambungkan dengan pin arduino digital pin 4
  - ✓ Pin D6 (kaki 13) di sambungkan dengan pin arduino digital pin 3
  - ✓ Pin D7 (kaki 14) di sambungkan dengan pin arduino digital pin 2
  - ✓ sambungkan potensio 10 KOhm ke +5v dan GND , dan Pin LCD 3 ke potensio
  - ✓ Pin 5 (R/W) ke Ground



2. Upload program berikut :

**Sketch :**

**// Program menulis Banner text di layar LCD**

// 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 rows and columns:

```
lcd.begin(16, 2);
```

```
lcd.clear(); // start with a blank screen
```

```
lcd.setCursor(0,0); // set cursor to column 0, row 0 (the first row)
```

```
lcd.print("NIM SAYA"); // change this text to whatever you like. keep it clean.
```

```
lcd.setCursor(0,1); // set cursor to column 0, row 1
```

```
lcd.print("Nama Saya");
```

```
}
```

```
void loop()
```

```
{
```

```
}
```

3. Amati yang tampilan LCD tersebut

4. Upload program berikut :

```
// Program Blinking Banner text
```

```
// 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 rows and columns:
```

```
lcd.begin(16, 2);
```

```
lcd.clear(); // start with a blank screen
```

```
lcd.setCursor(0,0); // set cursor to column 0, row 0 (the first row)
```

```
lcd.print("NIM SAYA"); // change this text to whatever you like. keep it clean.
```

```
lcd.setCursor(0,1); // set cursor to column 0, row 1
```

```
lcd.print("Nama Saya");
```

```
}
```

```
void loop() {
```

```
// Turn off the blinking cursor:
```

```
lcd.noDisplay();
```

```
delay(500);
```

```
// Turn on the display:
```

```
lcd.display();
```

```
delay(500);
```

```
}
```

```
// Program Banner scroll text 1234567890
```

```
// 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);

}

void loop() {

  // set the cursor to (0,0):

  lcd.setCursor(0, 0);

  // print from 0 to 9:

  for (int thisChar = 0; thisChar < 10; thisChar++) {

    lcd.print(thisChar);

    delay(500);

  }

  // set the cursor to (16,1):

  lcd.setCursor(16,1);

  // set the display to automatically scroll:

  lcd.autoscroll();

  // print from 0 to 9:

  for (int thisChar = 0; thisChar < 10; thisChar++) {

    lcd.print(thisChar);

    delay(500);

  }

  // turn off automatic scrolling

  lcd.noAutoscroll();

  // clear screen for the next loop:

  lcd.clear();

}

// Program Banner running text
```

```
// 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 rows and columns:
  lcd.begin(16, 2);
  // Print a message to the LCD.
  lcd.print("ROBOT KREATIF"); // 12 huruf
  delay(1000);
}

void loop() {
  // scroll 12 positions (string length) to the left
  // to move it offscreen left:
  for (int positionCounter = 0; positionCounter < 12; positionCounter++) {
    // scroll one position left:
    lcd.scrollDisplayLeft();
    // wait a bit:
    delay(300);
  }

  // scroll 28 positions (string length + display length) to the right
  // to move it offscreen right: 12+16 = 28
  for (int positionCounter = 0; positionCounter < 28; positionCounter++) {
    // scroll one position right:
    lcd.scrollDisplayRight();
    // wait a bit:
    delay(300);
  }
}
```

```
// 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(300);
}

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