

SIMPLE PROGRAMMING

‘Pengaturcaraan Mudah’

IEEE

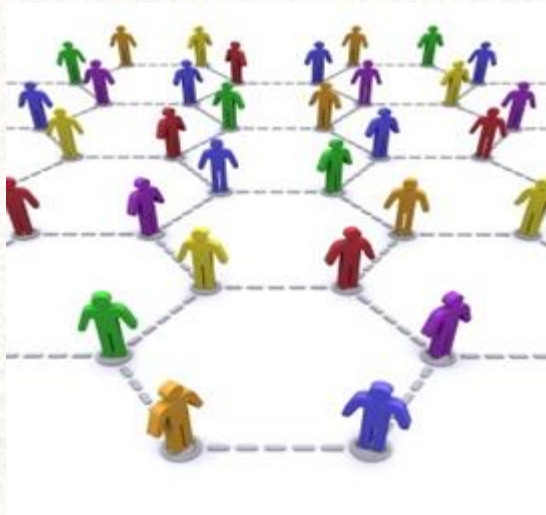
'Institute of Electrical and Electronic Engineering'

Pengenalan IEEE



Misi

Untuk memupuk inovasi dan kecemerlangan teknologi untuk faedah manusia.



Keahlian IEEE

Jurutera, saintis dan pelajar yang berminat pada bidang sains dan kejuruteraan elektrik dan komputer, dan disiplin yang berkaitan dengan profesionaliti.

400,000+ ahli di seluruh dunia:

300,000 profesional,

100,000 pelajar 160+ negara

Sektor swasta, akademik, kerajaan

Keahlian Persatuan

- Pertubuhan keahlian dalam IEEE (39 masyarakat) yang mewakili kepentingan teknikal atau profesional tertentu
- Sekitar 50% ahli IEEE juga tergolong dalam Persatuan
- Hampir semua persidangan IEEE, penerbitan, dan pembangunan standard ditaja dan diuruskan oleh Persatuan Teknikal IEEE
- Masyarakat menyokong bab teknikal yang bekerjasama dengan Bahagian IEEE tempatan anda

www.ieee.org/societies

Keahlian Persatuan

- ✓ Aerospace & Electronic Systems
- ✓ Antennas & Propagation
- ✓ Broadcast Technology
- ✓ Circuits & Systems
- ✓ Communications
- ✓ Components, Packaging, & Manufacturing Technology
- ✓ Computer
- ✓ Computational Intelligence
- ✓ Consumer Electronics
- ✓ Control Systems
- ✓ Dielectrics & Electrical Insulation
- ✓ Education
- ✓ Electromagnetic Compatibility
- ✓ Electron Devices
- ✓ Engineering in Medicine & Biology
- ✓ Geoscience & Remote Sensing
- ✓ Industrial Electronics
- ✓ Industry Applications
- ✓ Information Theory
- ✓ Instrumentation & Measurement
- ✓ Intelligent Transportation Systems
- ✓ Lasers & Electro-Optics
- ✓ Magnetics
- ✓ Microwave Theory & Techniques
- ✓ Nuclear & Plasma Sciences
- ✓ Oceanic Engineering
- ✓ Photonics
- ✓ Power Electronics
- ✓ Power & Energy
- ✓ Product Safety Engineering
- ✓ Professional Communication
- ✓ Reliability
- ✓ Robotics & Automation
- ✓ Signal Processing
- ✓ Society on Social Implications of Technology
- ✓ Solid-State Circuits
- ✓ Systems, Man, & Cybernetics
- ✓ Ultrasonics, Ferroelectrics, & Frequency Control
- ✓ Vehicular Technology

Kelebihan Ahli SPS

- ▶ PERCUMA penerbitan utama Society, Majalah IEEE *Signal Processing* elektronik dan penghantaran digital.
- ▶ Langganan bulanan PERCUMA untuk eNewsletter Pemprosesan Isyarat Masyarakat.
- ▶ Isu bulanan PERCUMA dari Warta Kandungan.
- ▶ Diskaun atas yuran pendaftaran untuk persidangan dan bengkel yang dimiliki sepenuhnya oleh Persatuan.
- ▶ Layak untuk Bantuan Geran Persidangan Perjalanan SPS

SPS Member Benefits (contd.)

▶ ENewsletter PERCUMA.



The screenshot shows an email client interface. At the top, there are navigation icons (back, forward, search, etc.) and a page indicator "3 of 39". The email title is "Careers in Signal Processing" with an "Inbox" tab. The sender is "IEEE Signal Processing Society via mail17.at11.rsgsv.net" and the date is "Mar 21 (7 days ago)".

The email content features the IEEE Signal Processing Society logo at the top left and the IEEE logo at the top right. Below the logo is a large, colorful graphic with the text "SIGNAL PROCESSING: The Science Behind Our Digital Life". A small link "Please [click here](#) if you are having trouble viewing this email" is located below the graphic.

Below the graphic is a section titled "CONNECT WITH US!" with icons for Facebook, Twitter, LinkedIn, and YouTube.


The main body of the email contains the heading "Careers in Signal Processing: Impacting Tomorrow, Today" followed by a paragraph: "When the IEEE Signal Processing Society was founded in 1948, the discipline of signal processing didn't exist, nor was it a prevalent career path. Fast forward to today, and we're living in a world enabled by signal processing, with endless possibilities in where it will take us."

Seasonal Schools: Recent Advances in Biomedical Imaging

<http://spsocmalaysia.org/winter>

→ spsocmalaysia.org/winter/contents/program.html

Winter School 2016 / 2017
Recent Advances in
Biomedical Imaging



Advancing Technology
for Humanity

[Home](#)

[Program](#)

[Venue](#)

[Register](#)

[Lodging](#)

[Committee](#)

[Contact](#)

PROGRAM

7th - 11th November 2016
@ Universiti Teknologi Malaysia (UTM) Kuala Lumpur, Malaysia

Time	Monday (7 Nov)	Tuesday (8 Nov)	Wednesday (9 Nov)	Thursday (10 Nov)	Friday (11 Nov)
0830-0900	Registration				
0900-1030	Intro to Biomedical Imaging <i>(Metin Gurcan)</i>	Big Data in Biomedical Imaging <i>(Eric Ho)</i>	Structural Imaging - X-ray and CT <i>(David Bradley)</i>	Structural Imaging - MRI <i>(Mandava Rajeswari)</i>	CAD & Image-guided Procedures <i>(Dave Hawkes)</i>
1030-1100	Coffee Break				
1100-1230	Digital Pathology <i>(Metin Gurcan)</i>	Deep Learning in Biomedical Imaging <i>(Eric Ho)</i>	Functional Imaging - PET & SPECT <i>(David Bradley)</i>	Functional Imaging - dMRI <i>(Mandava)</i>	3D Modeling & Fusion Imaging <i>(Dave Hawkes)</i>

Registration

Register by: **31 October 2016** *extended*

[Click to REGISTER!](#)

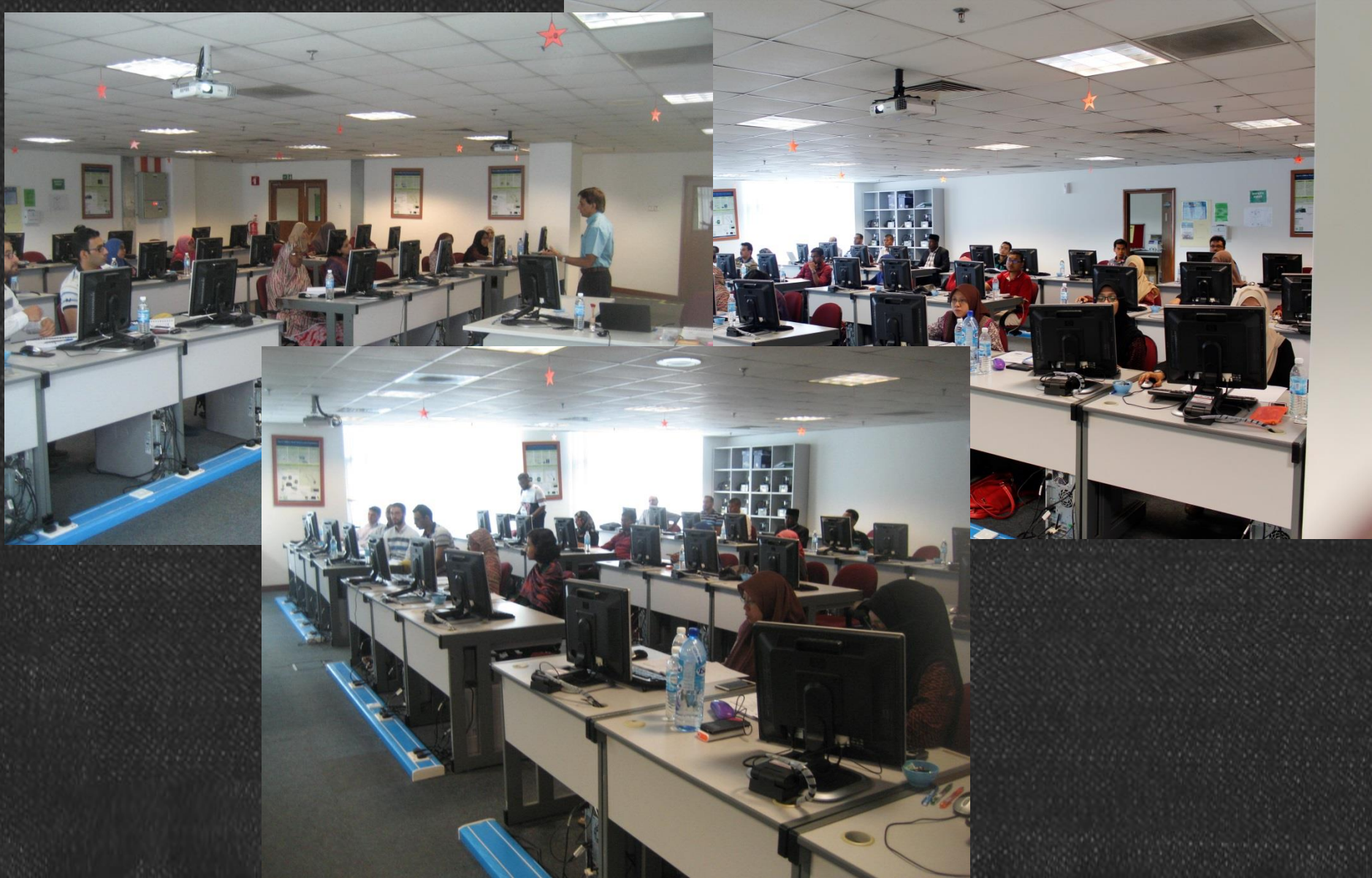
Fee Structure

Fee Structure	Total 5 days (MYR)	Daily* (MYR)



Workshops

<https://goo.gl/pqclY0>



Isi Kandungan Fasa 3

MODUL 1 :Pengenalan Asas Kejuruteraan

MODUL 2 :Projek Asas Mikropengawal

MODUL 3 :Jurutera Cilik

MODUL 1 : Pengenalan Asas Kejuruteraan

'Basic Engineering Introduction'



Hasil pembelajaran

- Peserta dapat mengenal apa itu kejuruteraan
- Peserta akan mengetahui cara jurutera bekerja dalam bidang kejuruteraan mereka
- Peserta dapat mengenalpasti sifat jurutera yang tersimpan dalam diri
- Peserta akan mendapati bahawa kejuruteraan adalah suatu bidang yang penting untuk kemajuan diri, masyarakat, dan negara.

Apa itu Kejuruteraan



Kejuruteraan ialah ...

- Perkara-perkara yang berkaitan dengan jentera atau mesin.

Sumber : Kamus pelajar edisi kedua

- Cabang sains dan teknologi berkenaan dengan rekabentuk, bangunan, dan penggunaan enjin, mesin, dan struktur.

Sumber : Kamus Oxford

Satu bidang professional yang mengaplikasi pengetahuan teknikal, saintifik dan matematik untuk menyelesaikan masalah dalam dunia sebenar.

Bidang-bidang Kejuruteraan

Bidang kejuruteraan mempunyai dapat dibahagikan kepada beberapa pengkhususan tertentu seperti :

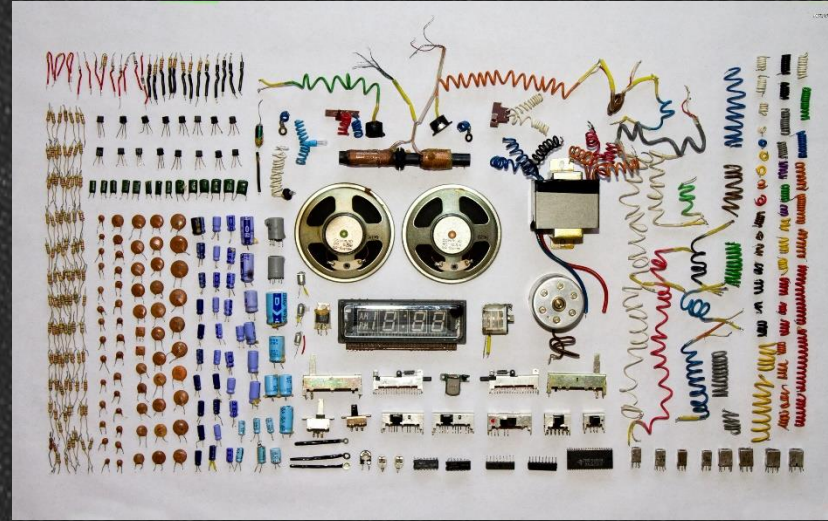
- Kejuruteraan awam
- Kejuruteraan elektrik
- Kejuruteraan elektronik
- Kejuruteraan kimia
- Kejuruteraan pembuatan
- Kejuruteraan pertanian
- Kejuruteraan sumber asli
- Kejuruteraan mekanikal
- Kejuruteraan bahan
- Kejuruteraan komputer
- Kejuruteraan aero angkasa
- Kejuruteraan telekomunikasi

Kejuruteraan Awam



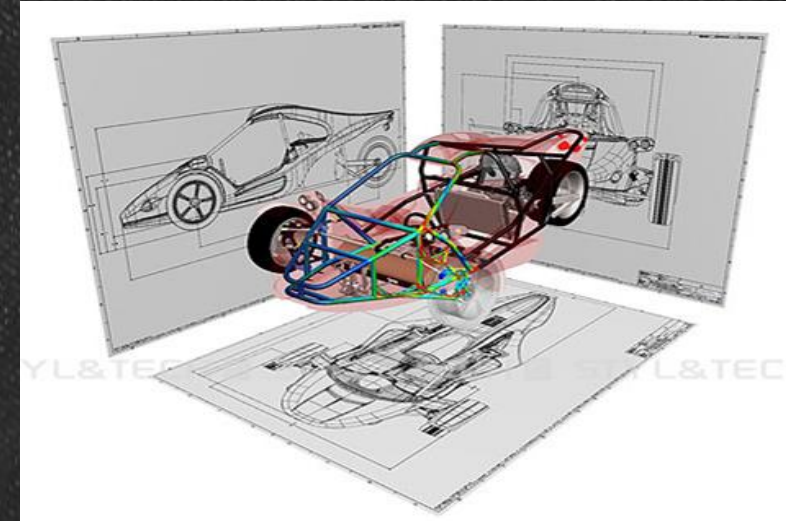
Sumber : Education Archive

Kejuruteraan Elektronik



Sumber : keyword-suggestions.com

Kejuruteraan Mekanikal



Sumber : stylIntech.com

Kejuruteraan Elektrik



Sumber : intra.edu.my

Kejuruteraan Kimia



Sumber : Seneca college

Kejuruteraan Telekomunikasi



Sumber : Utusan Online

Bagaimana Jurutera bekerja



Cara bekerja seorang jurutera

Cara bekerja seorang jurutera adalah berbeza mengikut bidang kejuruteraan yang dipilih, seperti :

Kejuruteraan awam

- Bekerja di kawasan pembinaan atau lebuhraya

Kejuruteraan elektrik

- Membuat penyambungan kuasa elektrik pada bangunan atau mesin

Kejuruteraan elektronik

- Menghasilkan komponen dan alat peranti elektronik

Kejuruteraan pertanian

- Bekerja dikawasan-kawasan pertanian

Kejuruteraan telekomunikasi

- Ditugaskan di kawasan pencawang signal

Kejuruteraan kimia

- Berada di makmal atau industry untuk penyelidikan

**Mengenal pasti
sifat jurutera
dalam diri**



Sifat seorang jurutera

Sifat-sifat dibawah adalah perlu untuk anda menjadi seorang jurutera yang memiliki nilai profesionalisme dalam bidang kejuruteraan :

- Menghargai dan menepati masa
- Kerja yang tersusun
- Bercakap berdasarkan fakta dan data, atau hipotesis secara saintifik
- Kreatif dan kritis
- Bijak menangani masalah
- Bersikap ambil peduli
- Berani dan yakin
- Bijak berkomunikasi
- Perlu kuat mental dan fizikal
- Mampu bekerja dalam apa jua kesulitan

Fakta menarik berkenaan
Jurutera
dan
Kejuruteraan



Tanpa jurutera...
apa
akan terjadi
pada dunia





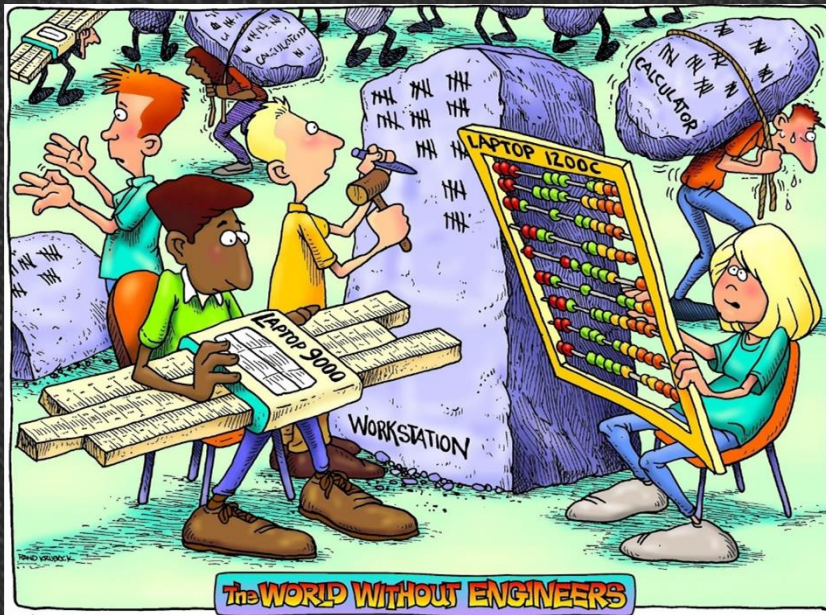
The World Without Engineers



The World Without Engineers



The World Without Engineers



The World Without Engineers



The World Without Engineers

Alamak!
Tidak ...!



Ye ke!





Oleh itu, bidang kejuruteraan ini penting untuk mengubah kehidupan yang sukar kepada yang lebih mudah. Seorang jurutera perlu ada keinginan untuk memperbaiki kehidupan kepada yang lebih baik.

Mulalah belajar dengan lebih hikmah kerana hanya dengan ilmu kemajuan akan dapat dicapai.

Semoga dengan 'hikmah' kita dapat meneroka segala penjuru kehidupan untuk kemajuan manusia sejagat.

Terima kasih...

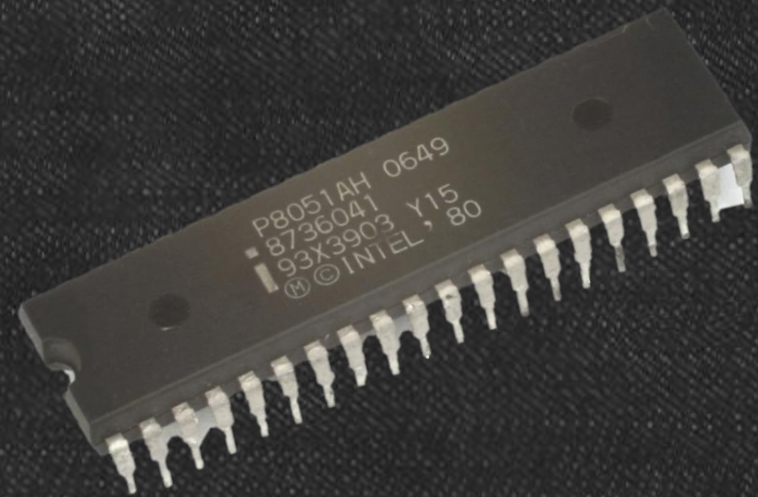
MODUL 2 : Projek Asas Mikropengawal

'MCU-Based Project'

Hasil yang akan dicapai

- Peserta dapat mengetahui apa itu mikropengawal
- Peserta dapat mempelajari ilmu berkenaan Arduino sebagai salah satu mikropengawal yang terkenal dan memahami pemprosesan isyarat bagi Arduino secara am
- Peserta dapat mempelajari bagaimana pengkodan Arduino dilakukan
- Peserta akan dapat melaksanakan projek asas Arduino dengan jayanya

Apa itu Mikropengawal (μC)



Mikropengawal atau pengawal mikro ialah ...

- Peranti kawalan yang menggunakan mikropemproses sebagai CPU-nya.

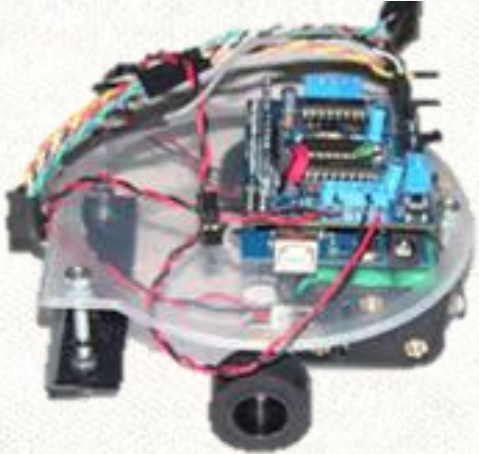
Sumber : Kamus computer Dewan Bahasa dan Pustaka

- Peranti kawalan yang menggabungkan mikropemproses.

Sumber : Kamus Oxford

Komputer pada cip (*computer on-chip*) yang telah dioptimumkan untuk mengurus dan mengawal alatan elektrik.

Contoh-contoh Aplikasi Mikropengawal



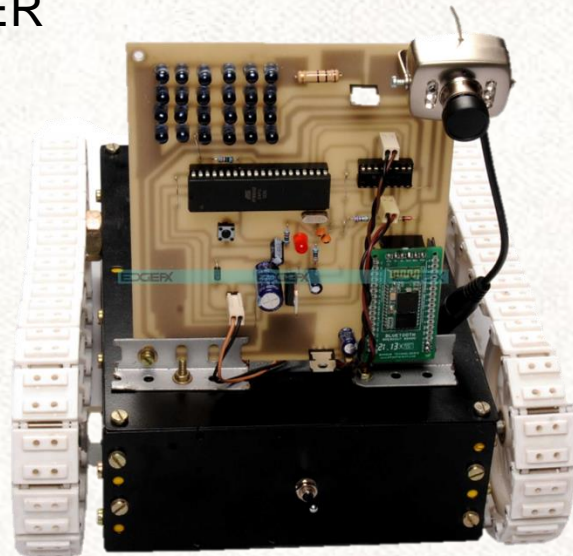
LINE FOLLOWER
ROBOT



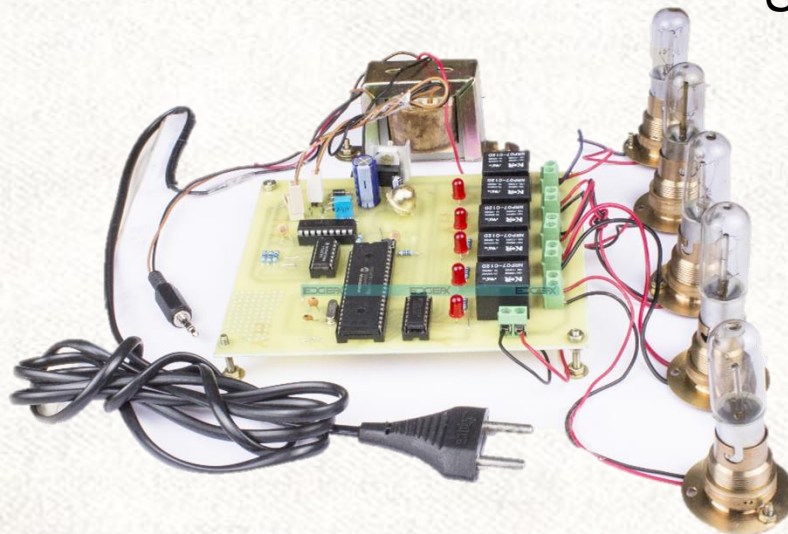
INSECT ROBOT



SMART DOORLOCK



SPY ROBOT



HOME
AUTOMATION
WIRING SYSTEM

Komponen penting dalam Mikropengawal



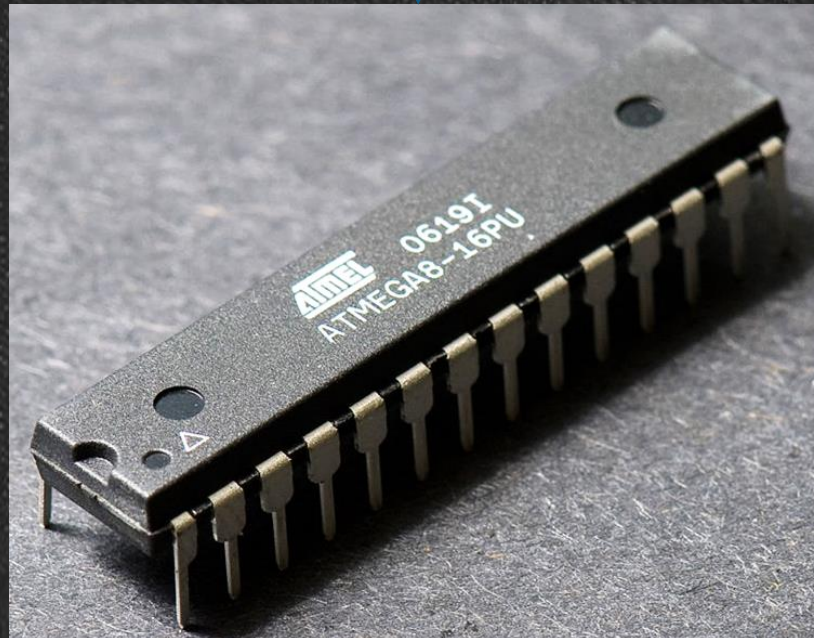
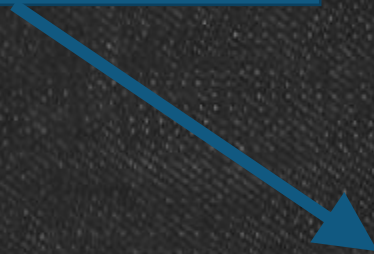
Micropemproses
(CPU)

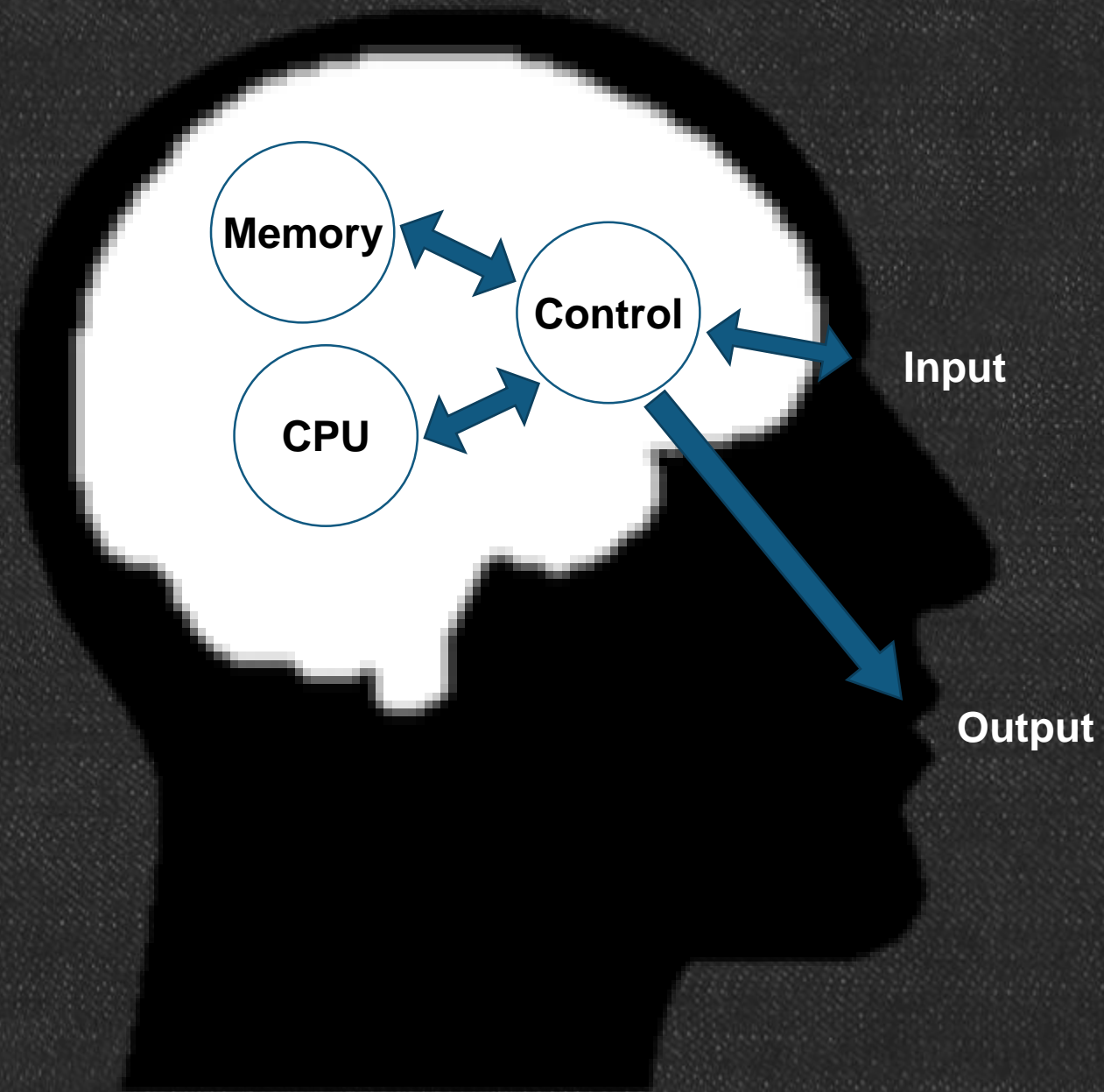


Memori

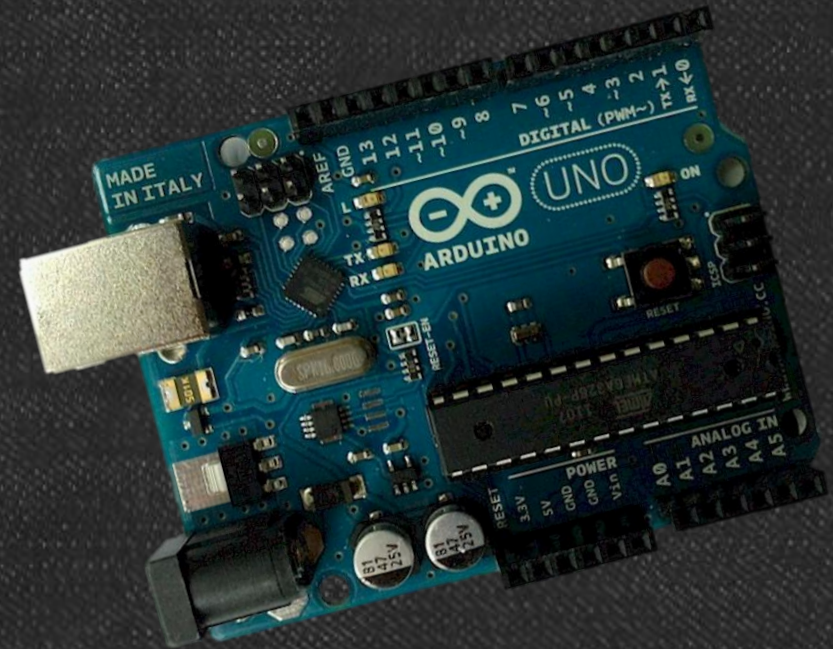


Input (masuk)
&
Output (keluar)



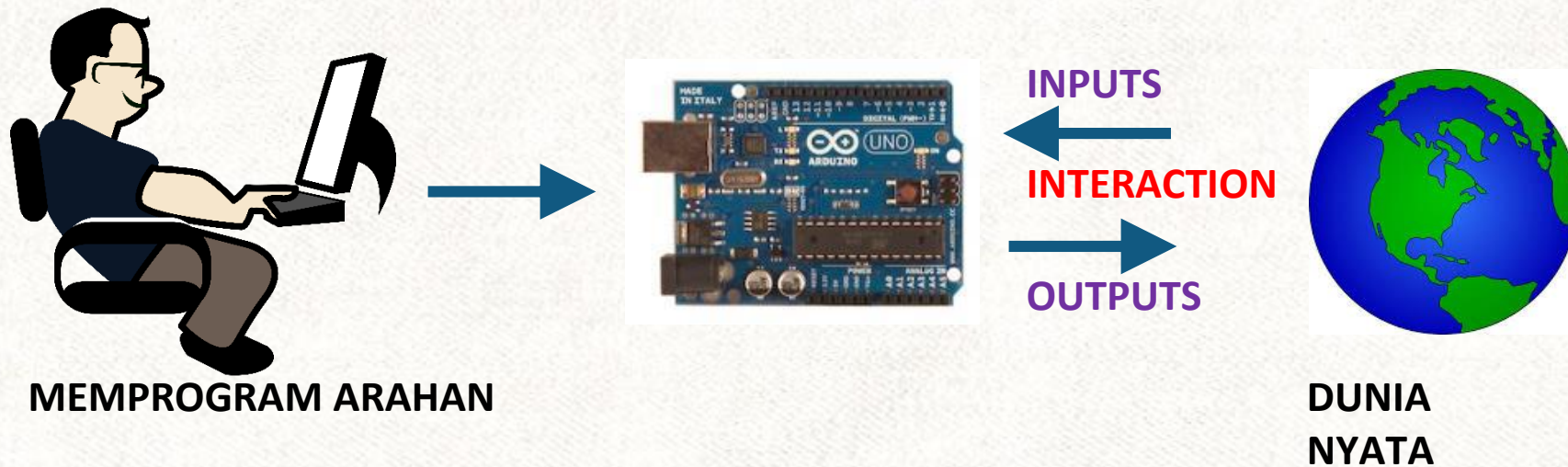


Arduino



Apa itu Arduino

- **Arduino** pada dasarnya adalah **otak suatu ciptaan** yang mampu membaca daripada input seperti tekan butang atau memantau pengukuran suhu,
- kemudian dapat diprogram untuk **menafsirkan maklumat** tersebut untuk **mengawal output** dalam bentuk cahaya, gerakan motor atau mengirim notifikasi penggera melalui SMS.



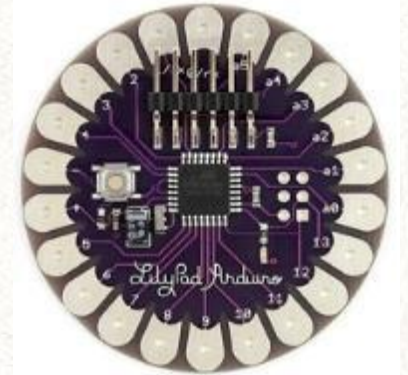
Jenis-jenis Arduino



UNO



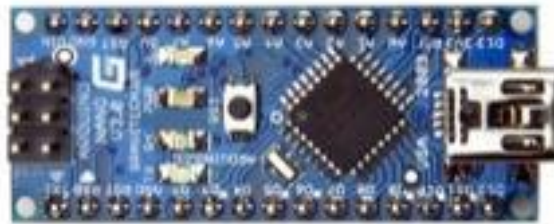
Mega



LilyPad



Arduino BT

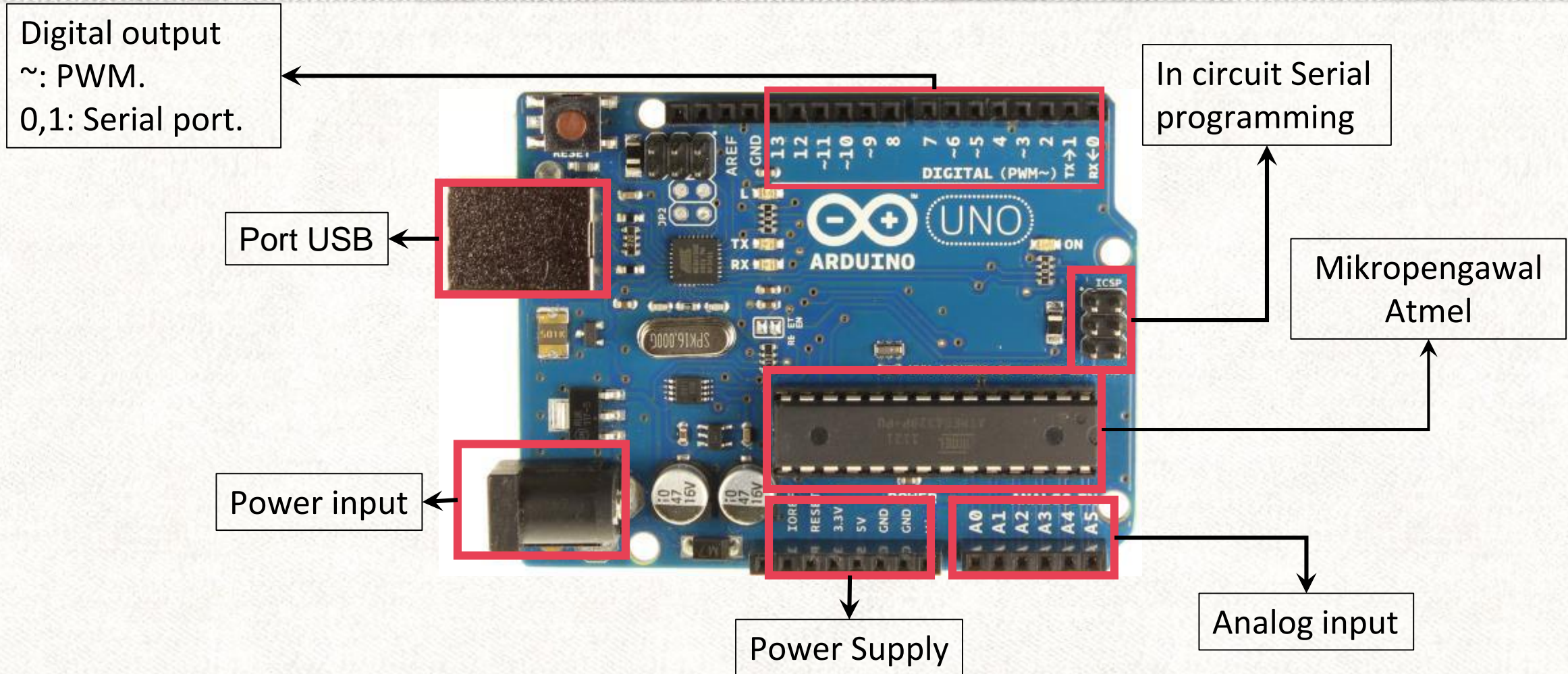


Arduino Nano



Arduino Mini

Kelengkapan Arduino UNO



Persediaan Awal Arduino

Pastikan anda telah memiliki peralatan berikut:

1

KOMPUTER

Windows, Linux or MacOS dengan port USB

2

ARDUINO BOARD dengan Wayar USB

Modul ini menggunakan Arduino UNO

3

PETI PERMULAAN ARDUINO

Mengandungi semua komponen yang diperlukan untuk modul

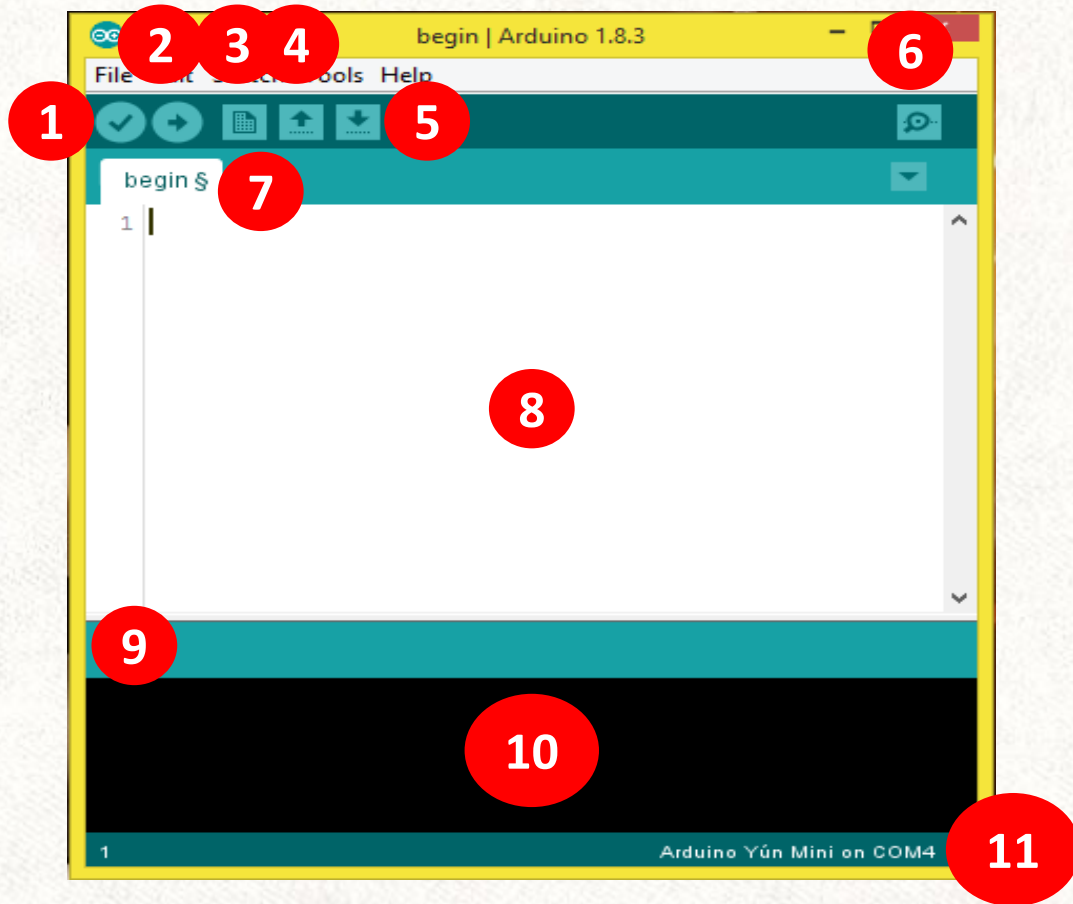
4

ARDUINO SOFTWARE

Dikaitkan dengan Integrated Developers Environment (IDE)

Persediaan Awal Arduino

Arduino IDE:



- 1 **Verify:** Menyusun kod dan mengesan ralat
- 2 **Upload:** Menghantar lakaran ke papan Arduino
- 3 **New:** Membuka lakaran yang baru
- 4 **Open:** Membuka lakaran sedia ada
- 5 **Save:** Menyimpan lakaran semasa
- 6 **Serial Monitor:** Memapar informasi serial
- 7 **Sketch Name:** Memapar nama lakaran semasa
- 8 **Sketch Area:** Ruang pengkodan
- 9 **Message Area:** Memapar kan ralat dan status

- 10 **Verify Area:** Ruang status dan ralat pada kod
- 11 **Board and Serial Port:** Memaparkan papan Arduino yang diguna bersama port serial

Persediaan Awal Arduino

Memasang Arduino *Integrated Development Environment* (IDE)

- Muat turun Arduino IDE dari <https://www.arduino.cc/en/Main/Software>

Download the Arduino IDE



ARDUINO 1.8.3

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.

This software can be used with any Arduino board. Refer to the [Getting Started](#) page for Installation instructions.

Windows Installer

Windows ZIP file for non admin install

Windows app 

Mac OS X 10.7 Lion or newer

Linux 32 bits

Linux 64 bits

Linux ARM

[Release Notes](#)

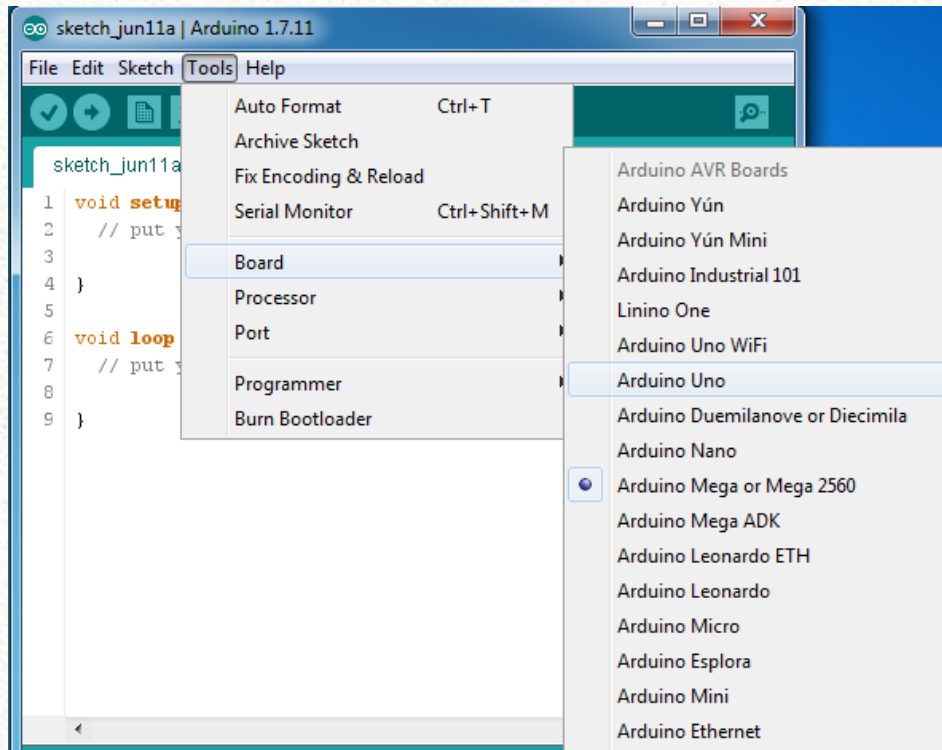
[Source Code](#)

[Checksums \(sha512\)](#)

Persediaan Awal Arduino

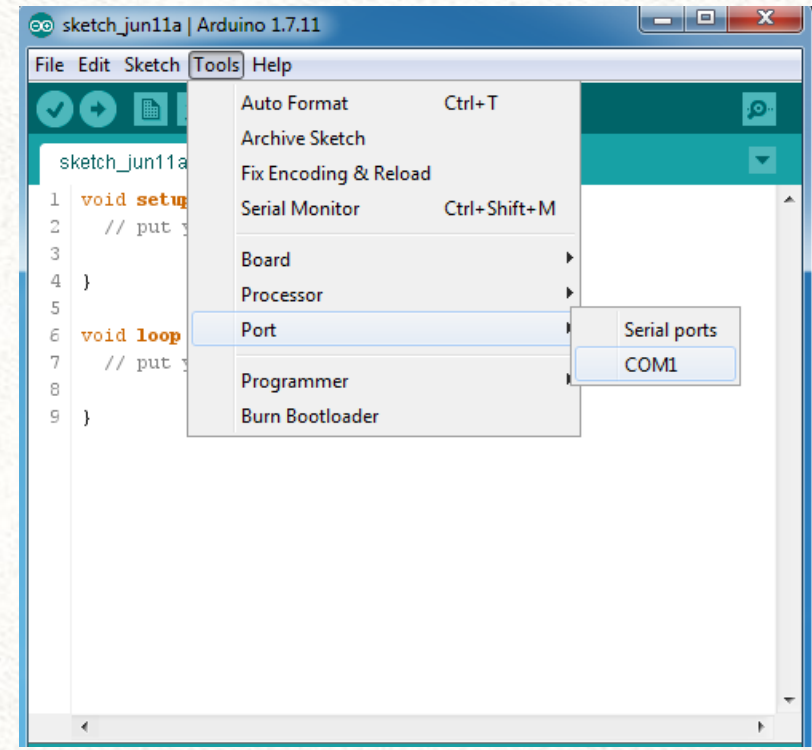
1. Pilih jenis papan Arduino

- pergi ke menu **Tools** dan pilih **Board**
- pilih jenis papan Arduino yang anda gunakan



2. Pilih Serial Port

- pergi ke menu **Tools** dan pilih **Port**
- pilih port COM untuk jenis papan Arduino yang anda gunakan



Rel Kuasa, Digital, dan Analog

1. Rel Kuasa

- terdiri daripada 5 pin
- 5V dan GND memberi kuasa dan punca bumi kepada litar luaran
- Vin and GND menerima bekalan kuasa voltan untuk menghidupkan Arduino (alternatif untuk USB dan DC jack)
- 3V3 ialah 3.3 Volt bekalan kuasa untuk litar luaran

2. Input Analog

- 6 input pin yang menerima voltan analog berlabel A0 hingga A5
- menukarkan voltan antara 0 dan 5 volt kepada nilai integer antara 0 hingga 1023
- mengambil masa 100 mikrosaat untuk membaca input analog

3. Input/Output Digital

- terdapat 13 pin digital berlabel 0 hingga 13
- setiap pin adalah *bidirectional* (2 arah), boleh di tetapkan sebagai input atau output
- pin 13 telah dihubungkan kepada LED yang sedia ada pada papan Arduino

Pengkodean Arduino

```
for (i = 0; i < 3; i++) {  
  unsigned int cp_count;  
  unsigned int len = n;  
  if (user(group_info[i]) {  
    return -EFAULT;  
  }  
  grouplist += NGS_PER_BLOCK;  
  count += cp_count;  
}
```

```
group_info = kmalloc(usr)  
if (!group_info)  
  return NULL;  
group_info->ngroups = gidsetsize;  
group_info->nblocks = nblocks;  
atomic_set(&group_info->usage, 1)
```

```
if (gidsetsize <= NGROUPS_SMALL)  
  group_info->blocks[0] = g;  
else {  
  for (i = 0; i < nb; i++) {  
    gid_t *b;  
    b = (void *)_get(GFP);  
    if (!b)  
      goto partial_alloc;  
    group_info->blo[i] = b;  
  }  
}
```

```
struct group_info  
int nblocks;
```

```
void groups_free(struct) {  
  if (group_info->blocks[0]) {  
    int i;  
    for (i = 0; i < 1; i++)  
      free_page(i);  
  }  
  free(group_info);  
}
```



Output Digital

IDEA

Diod pemancar cahaya (LED) adalah sumber cahaya untuk projek ini. Sebagai permulaan, kita akan menghasilkan kesan berkedip dengan menghidupkan LED ON dan OFF

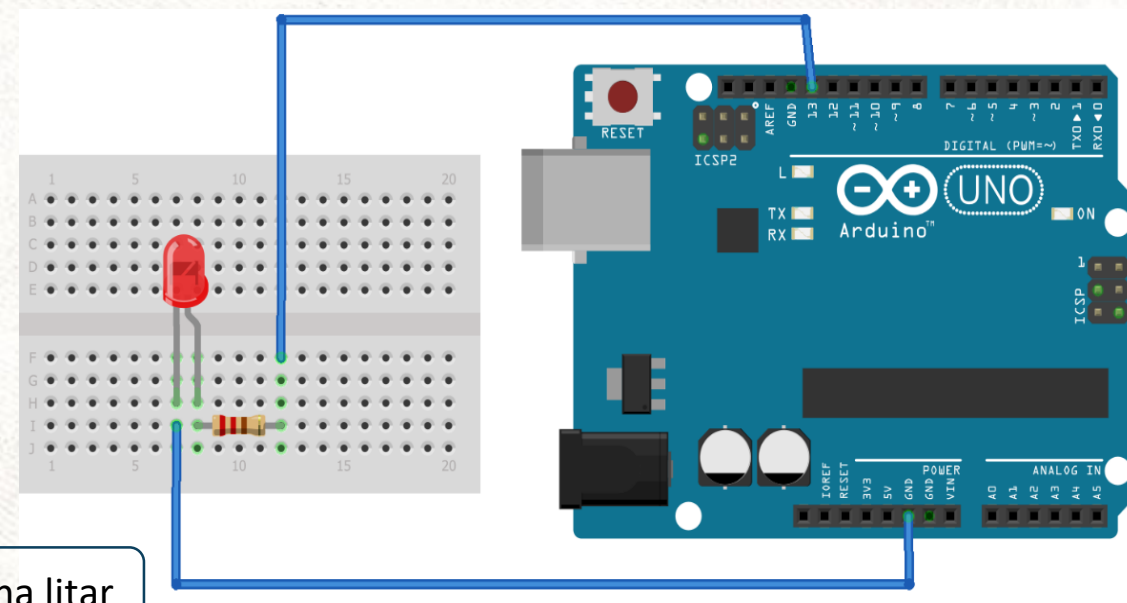
APA YANG AKAN DIPELAJARI

- Lakaran minimum Arduino
- Menetapkan pin digital dwiarah
- Menghidupkan dan mematikan pin digital output

GAMBARAJAH BLOK



GAMBARAJAH LITAR



Langkah 1: Bina litar


Output Digital

LAKARAN KOD PERTAMA ANDA

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

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

Apa-apa sahaja selepas `//` akan tidak dipedulikan oleh Arduino yang dinamakan komen

Langkah 2: Membuka Arduino IDE 

Fungsi mengandungi garis kod dalam kurungan { } yang memberitahu Arduino apa yang perlu dilakukan. Arduino mesti mempunyai dua fungsi terbina yang dipanggil `setup` dan `loop`.

Barisan kod dalam `setup` hanya berjalan **SEKALI**.

Barisan kod `loop` berjalan **BERULANG KALI**, hanya berhenti apabila Arduino dimatikan.



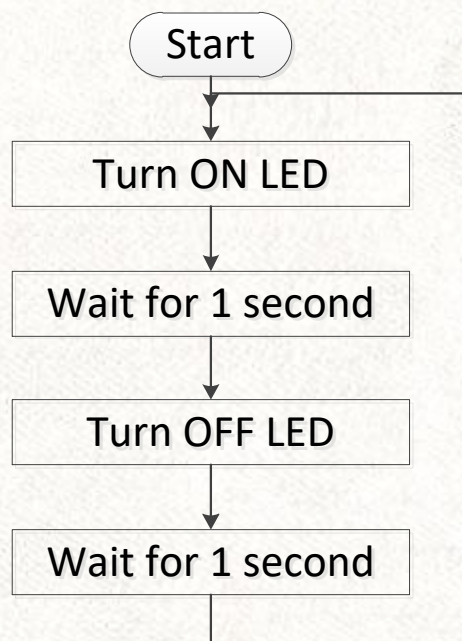
Pengaturcaraan Arduino adalah sensitif huruf. Perhatikan sintaks, contohnya: { dan (

Output Digital

KOD ARAHAN	PENERANGAN
<code>pinMode(PIN, MODE);</code>	<ul style="list-style-type: none">• Kod ini membolehkan kita menetapkan mod satu pin digital. Nilai untuk mod perlu ditetapkan sebagai INPUT atau OUTPUT
<code>digitalWrite(PIN, STATE);</code>	<ul style="list-style-type: none">• Kod ini membolehkan kita untuk ON atau OFF pin digital yang dinyatakan dalam STATE.• State yang HIGH menghantar 5 volt ke pin.• State yang LOW menghantar 0 volt.
<code>delay(1000);</code> Selang Delay untuk 1 saat.	<ul style="list-style-type: none">• Ia memberitahu Arduino untuk menunggu beberapa masa yang ditetapkan oleh nilai (dalam milliseconds) sebelum bergerak ke baris seterusnya.

Output Digital

CARTA ALIRAN



LAKARAN KOD

```

void setup()
{
  pinMode(13, OUTPUT);
}
  
```

Kita akan hidupkan cahaya LED pada pin 13, jadi tetapkannya sebagai OUTPUT

```

void loop()
{
  digitalWrite(13, HIGH);
  delay(1000);
  digitalWrite(13, LOW);
  delay(1000);
}
  
```

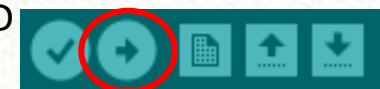
LED menyala

Tunggu selama 1 saat

LED tidak menyala

Kod yang diletakkan di dalam LOOP akan diulang

Langkah 3: Tekan pada butang UPLOAD untuk menghantar kod ke Arduino. Kamu akan melihat LED menyala dan padam setiap 1 saat



Membina Fungsi

Fungsi adalah beberapa baris kod yang dikumpulkan bersama dalam satu blok mudah. Kita boleh memanggil fungsi beberapa kali dalam lakaran dengan memanggil namanya daripada menulis setiap baris kod sekali lagi.

Kita boleh menukar garisan arahan dalam gelung ke fungsi seperti berikut:

```
void blinkLED()
{
  digitalWrite(13, HIGH);
  delay(1000);
  digitalWrite(13, LOW);
  delay(1000);
}
```

LAKARAN KOD

```
void setup()
{
  pinMode(13, OUTPUT);
}
```

```
void loop()
{
  blinkLED();
}
```

```
void blinkLED()
{
  digitalWrite(13, HIGH);
  delay(1000);
  digitalWrite(13, LOW);
  delay(1000);
}
```



Fungsi boleh dibina dimana-mana sahaja diluar lakaran fungsi SETUP dan LOOP

← Memanggil fungsi

} Fungsi yang dibina

Output Digital

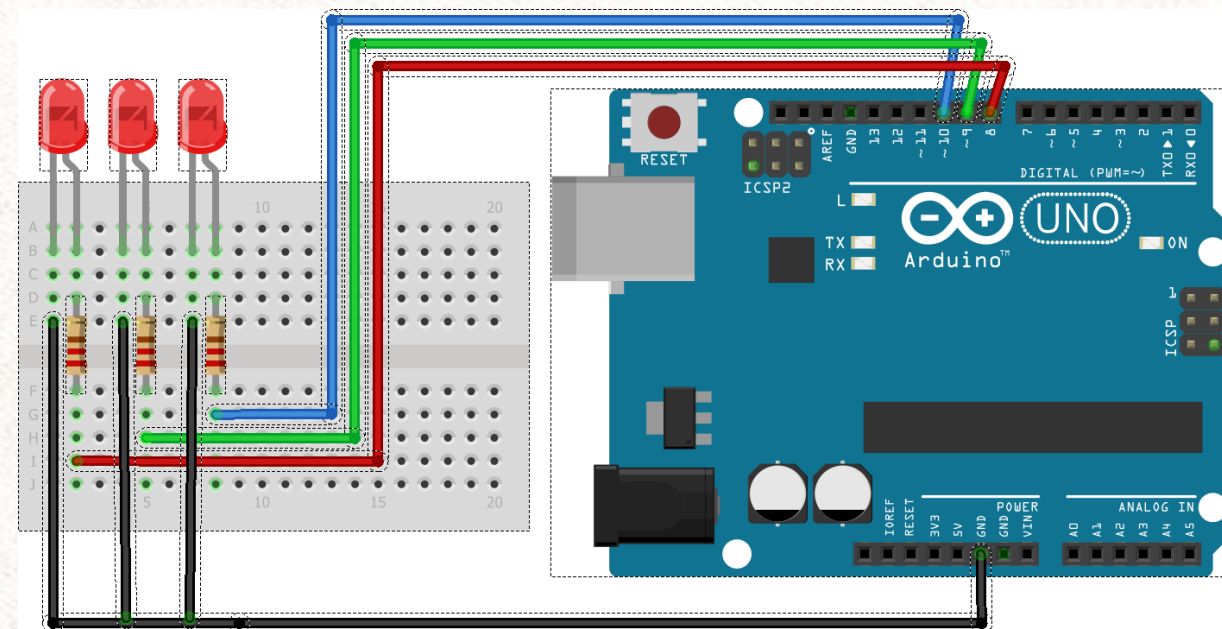
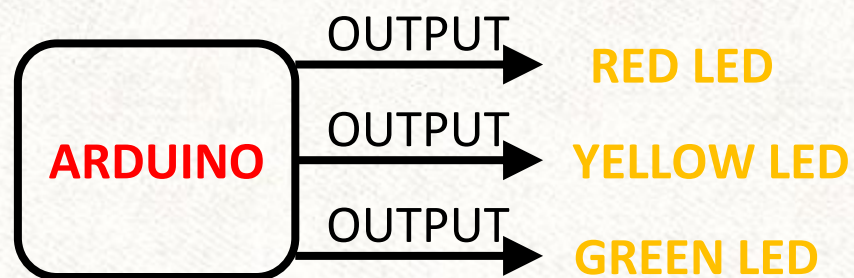
IDEA

Kawalan LED seperti lampu isyarat. Dalam latihan ini, kita akan mengawal 3 LED (merah, kuning, & hijau).

APA YANG AKAN DIPELAJARI

- Mengawal 3 LED
- Mencipta dan menggunakan pembolehubah

GAMBARAJAH BLOK



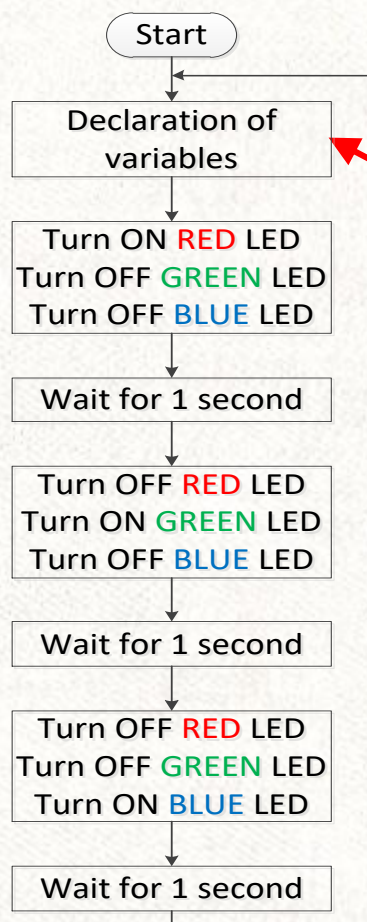
Langkah 1: Bina litar

Output Digital

KOD ARAHAN	PENERANGAN
<pre>int red = 8; int yellow= 9; int green= 10;</pre>	<ul style="list-style-type: none"> • Barisan kod ini dikenali sebagai mengisytiharkan pembolehubah yang digunakan untuk menyimpan data. Dalam kes ini, kita menggantikan nombor pin dengan nama warna yang sama. • Pemboleh ubah membuat pengkodan fleksibel di mana jika kita menukar pin red daripada pin8 kepada pin yang lain, kita hanya ubah nilai dalam perisytiharan.
<pre>pinMode(red, OUTPUT); pinMode(yellow, OUTPUT); pinMode(green, OUTPUT);</pre>	<ul style="list-style-type: none"> • Kita kini boleh menggunakan nama pembolehubah yang diisytiharkan untuk menggantikan nombor pin sebenar di seluruh lakaran. • Kita biasa dengan perkataan dan bukan nombor, lakaran juga kini lebih mudah dibaca dengan menggunakan perkataan.

Output Digital

CARTA ALIRAN



Istihar pembolehkan diluar: `void setup()` & `void loop()`

LAKARAN KOD

```
int red = 8;
int yellow = 9;
int green = 10;
```

Kita kawal LED RGB pada pin 8, 9, dan 10. Oleh itu, tetapkan ia sebagai OUTPUT.

```
void setup()
{
  pinMode(red, OUTPUT);
  pinMode(yellow, OUTPUT);
  pinMode(green, OUTPUT);
}
```

Langkah 2: Membuka Arduino IDE

```
void loop()
```

```
{
  digitalWrite(red, HIGH);
  digitalWrite(yellow, LOW);
  digitalWrite(green, LOW);
  delay(1000);
```

Mengawal 3 LED

```
  digitalWrite(red, LOW);
  digitalWrite(yellow, HIGH);
  digitalWrite(green, LOW);
  delay(1000);
```

```
  digitalWrite(red, LOW);
  digitalWrite(yellow, LOW);
  digitalWrite(green, HIGH);
  delay(1000);
```

```
}
```

Langkah 3: Tekan pada butang UPLOAD untuk menghantar kod ke Arduino.



Output Digital

IDEA

Buzzer atau beeper adalah peranti isyarat audio dalam bentuk bahan piezoelektrik. Dalam latihan ini, kita akan menghasilkan nada menggunakan Library dan buzzer Arduino.

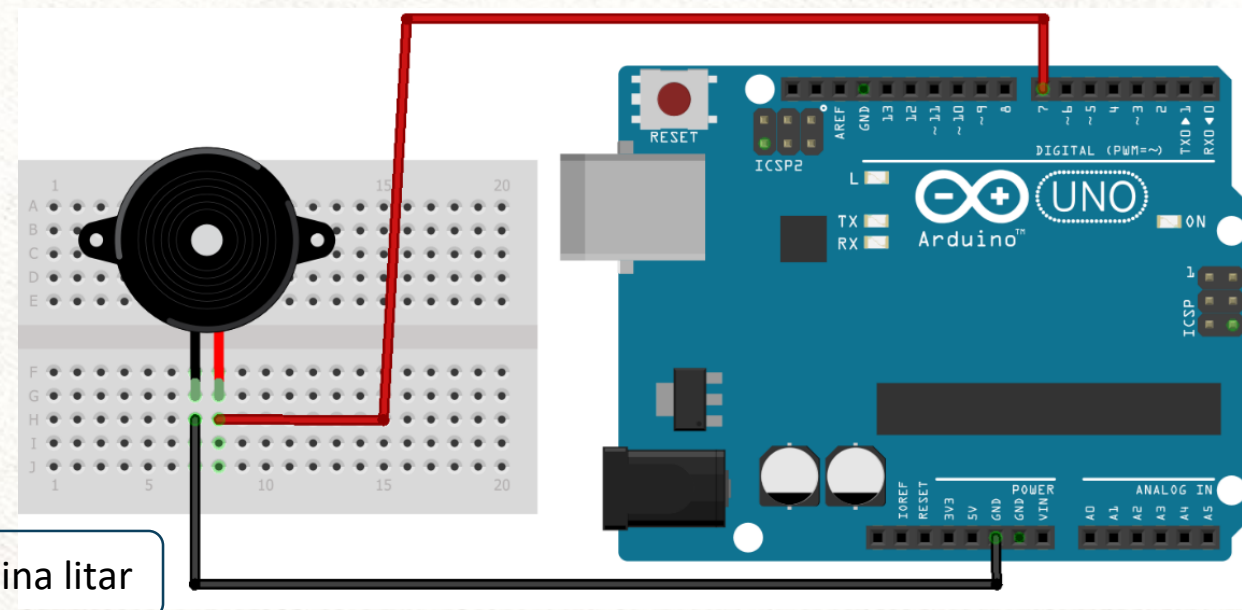
APA YANG AKAN DIPELAJARI

- Menghasilkan nada dengan pin digital
- Menggunakan Library dalam Arduino
- Menggabungkan nada untuk dijadikan melodi

GAMBARAJAH BLOK



GAMBARAJAH LITAR



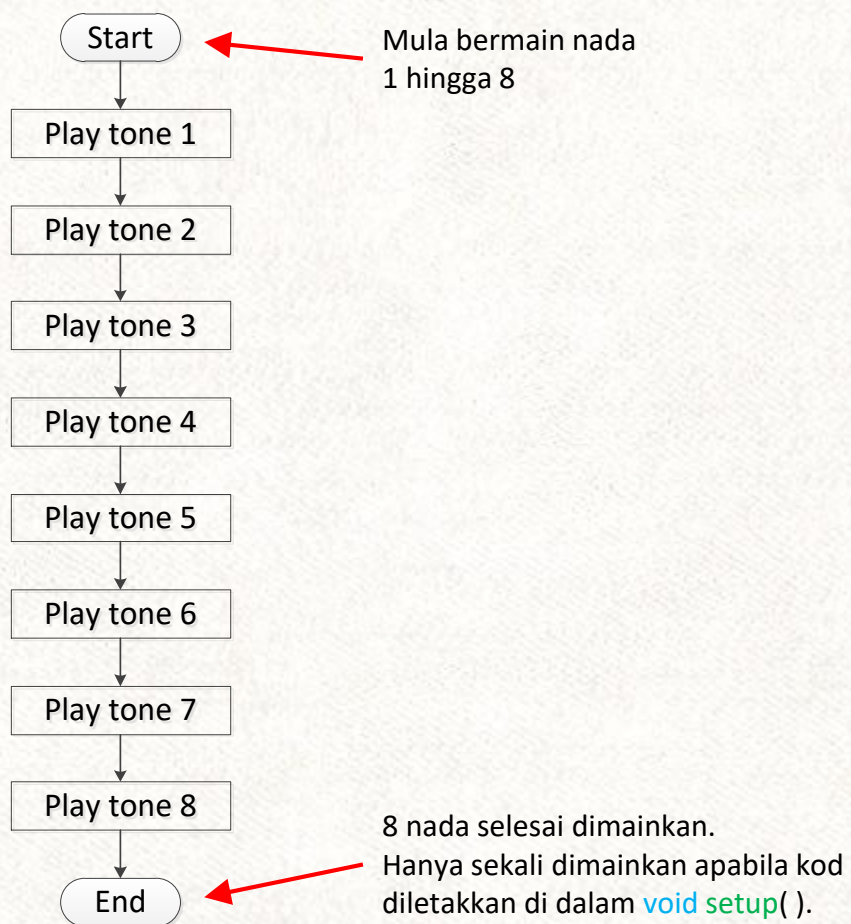
Langkah 1: Bina litar

Output Digital

KOD ARAHAN	PENERANGAN
<code>tone(7, 262, 250);</code>	Arduino perpustakaan <code>tone()</code> digunakan untuk menghasilkan output nada pada Arduino. <code>tone(pin, note, note duration);</code>
<code>delay(250 * 1.30);</code>	Dalam <code>delay()</code> , tempoh note didarabkan dengan 1.3 iaitu kelewatan 30% antara note untuk membezakan note dari satu ke yang lain.

Output Digital

CARTA ALIRAN



LAKARAN KOD

Langkah 2: Membuka Arduino IDE 

```

void setup()
{
  tone(7, 262, 250);
  delay(250 * 1.30);

  tone(7, 196, 125);
  delay(250 * 1.30);

  tone(7, 196, 125);
  delay(250 * 1.30);

  tone(7, 220, 250);
  delay(250 * 1.30);

  tone(7, 196, 250);
  delay(250 * 1.30);

  tone(7, 247, 250);
  delay(250 * 1.30);
}
  
```

- `tone(pin, note, note duration);`
- Dalam `delay()`, tempoh note **didarabkan dengan 1.3** iaitu kelewatan 30% antara note


```

tone(7, 247, 250);
delay(250 * 1.30);

tone(7, 262, 250);
delay(250 * 1.30);
}
  
```

```

void loop()
{
}
  
```

Langkah 3: Tekan pada butang  **UPLOAD** untuk menghantar kod ke Arduino.

Output Analog

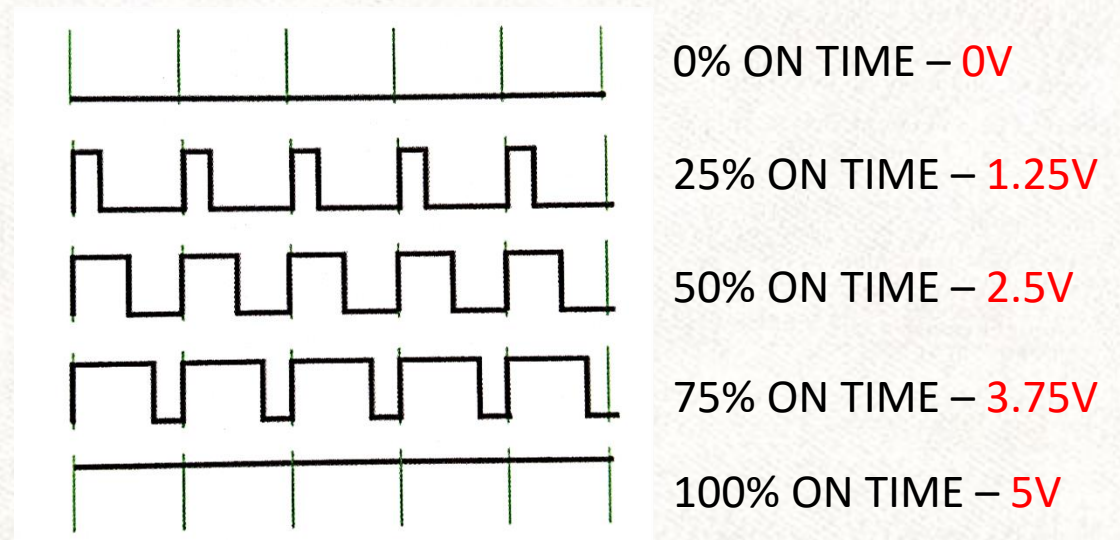


Isyarat analog boleh menghasilkan **pelbagai nilai** tidak seperti isyarat digital yang hanya mempunyai dua nilai HIGH dan LOW. Sepertimana latihan sebelumnya, **digitalWrite** (pin, **HIGH**) akan menghasilkan keluaran voltan 5V manakala **digitalWrite** (pin, **LOW**) akan menghasilkan keluaran voltan 0V.

Oleh kerana mikropengawal adalah sememangnya digital, satu teknik yang dipanggil Pulse Width Modulation (PWM) boleh mensimulasikan fungsi isyarat analog dengan mempunyai kemampuan menghasilkan output yang berbeza-beza, iaitu 2.3V atau 3.8V.



PWM menghasilkan pelbagai nilai analog dengan menukar lebar isyarat denyut digital "ON TIME" yang diulang pada kadar yang sangat cepat. Hasilnya adalah sebagai isyarat adalah voltan mantap antara 0 - 5V.

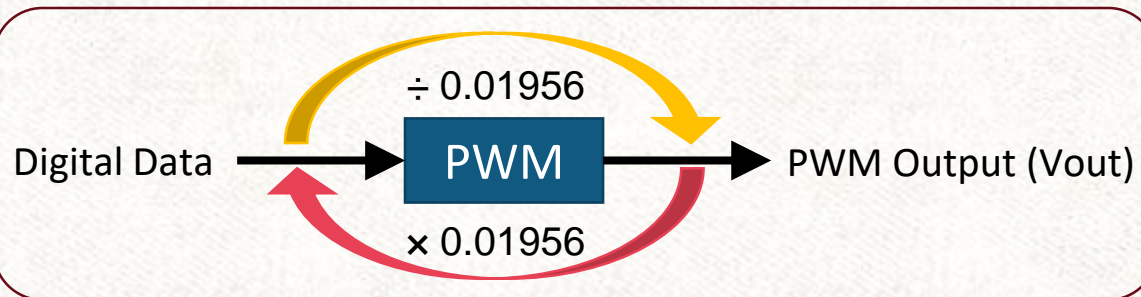
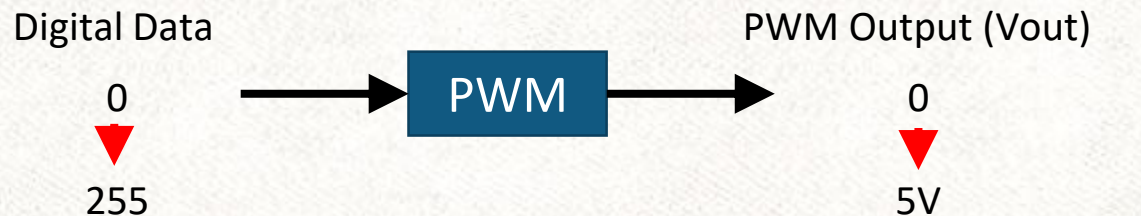


Output Analog

PULSE WITDH MODULATION (PWM)

Terdapat pin digital yang terdapat pada Arduino UNO yang mampu menghasilkan output analog melalui PWM. Pin ini ialah 3, 5, 6, 9, 10, dan 11.

Nadi digital "on time" dalam PWM ditetapkan dengan nilai antara 0 hingga 255 (integer 8-bit) yang mewakili output voltan analog 0 hingga 5V.



$$\frac{\text{PWM Output (Vout)}}{5V} = \frac{\text{Digital Data}}{255}$$
$$\text{PWM Output (Vout)} = \frac{\text{Digital Data} \times 5V}{255}$$
$$= \text{Digital data} \times 0.01956$$

Contoh jika Digital Data = 130,
 $V_{out} = 130 \times 0.01956$
 $= 2.5428V$

Output Analog

IDEA

Menggunakan `analogWrite()`, kita akan mengawal kecerahan LED. Latihan ini akan menunjukkan kemampuan Arduino dalam menghasilkan output analog.

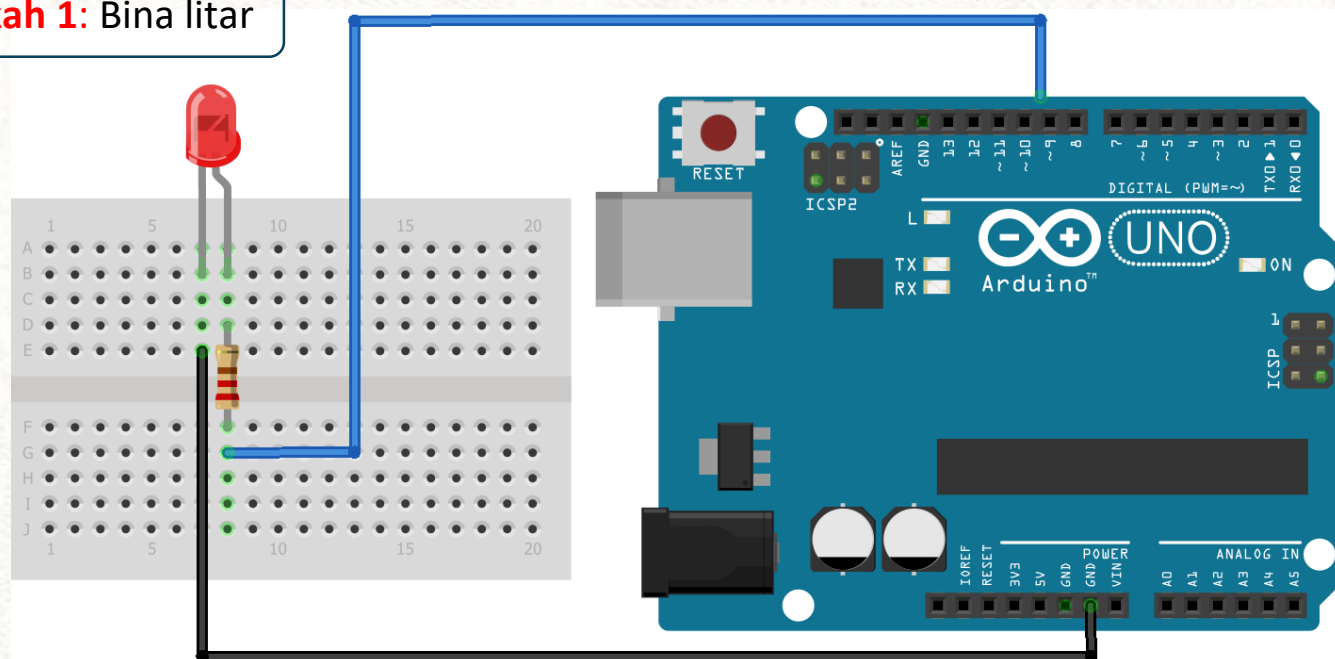
APA YANG AKAN DIPELAJARI

Menggunakan `analogWrite()`, untuk kita mengawal kecerahan LED.

GAMBARAJAH BLOK



Langkah 1: Bina litar



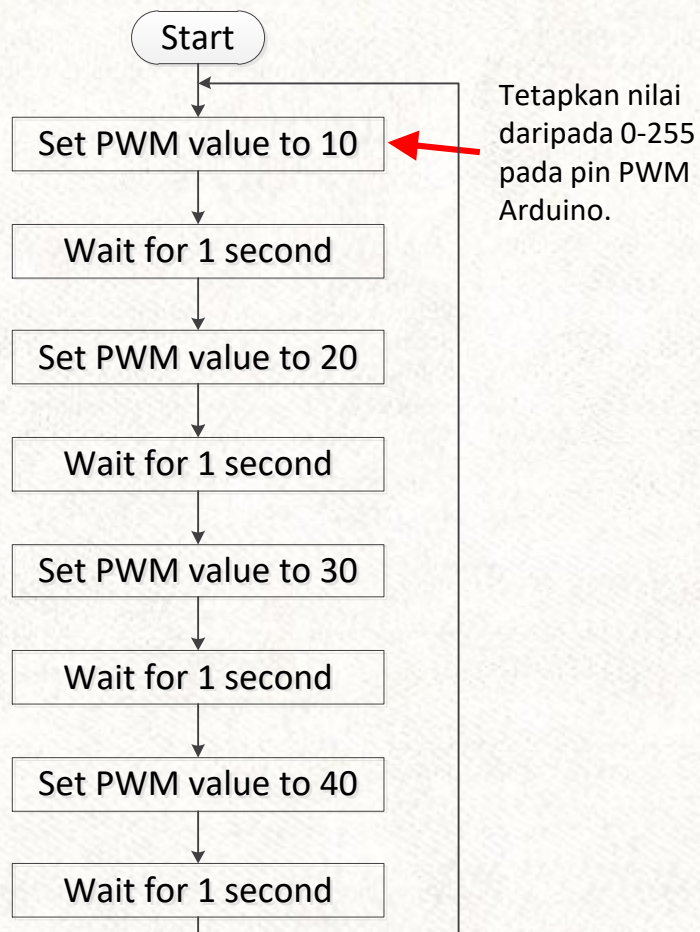
Perhatikan PIN 8, 9, dan 10 ialah PIN PWM

Output Analog

KOD ARAHAN	PENERANGAN
<pre>analogWrite(pin, value);</pre>	<p>Arahan <code>analogWrite ()</code> menghasilkan output PWM pada pin yang ditetapkan.</p> <p>PIN yang disediakan ialah 3, 4, 5, 6, 9, 10, 11. Nilai antara 0 - 255.</p> <p>Contoh:</p> <pre>analogWrite(8, 130);</pre> <p>pin = 8 Value = 130 sepadan dengan 2.5428V</p>

Output Analog

CARTA ALIRAN



LAKARAN KOD

```
int red = 9;
```

```
void setup()
```


```
{
  pinMode(red, OUTPUT);
}
```

Tetapkan pin sebagai output

```
void loop()
```

```
{
  analogWrite(red, 0);
  delay(1000);
  analogWrite(red, 10);
  delay(1000);
  analogWrite(red, 20);
  delay(1000);
  analogWrite(red, 30);
  delay(1000);
  analogWrite(red, 40);
  delay(1000);
}
```

Mengawal LED RGB berwarna merah menggunakan analogWrite

Langkah 2: Membuka Arduino IDE 

Langkah 3: Tekan pada butang UPLOAD untuk menghantar kod ke Arduino.



Pengkodan Mudah – FOR LOOP

Dalam Exercise 4, kita dapat melihat kod untuk analogWrite dan delay diulang dengan kenaikan 10 dalam nilai PWM untuk setiap pengulangan. Tujuannya adalah untuk melihat peningkatan kecerahan LED RGB berwayar.

Untuk mempermudah kod, pernyataan FOR boleh digunakan untuk mengulangi blok kod yang disertakan dalam kurungan { }.

```
for (initialization; condition; increment/decrement)
{
  //codes
}
```

Initialization : Perisytiharan pembolehubah dan dilaksanakan dahulu sekali sahaja. Titik permulaan loop.

Condition: Keadaan ini diuji. Jika benar, kod dalam kurungan akan dilaksanakan bersama dengan peningkatan/pengurangan (nilai pembolehubah akan meningkat atau berkurang). Apabila keadaan menjadi palsu, loop berakhir dan program akan terus ke baris seterusnya.

LAKARAN KOD

```
void loop()
{
  for (int i = 0; i < 40; i++)
  {
    analogWrite(9, i);
    delay(50);
  }
}
```

Pembolehubah i di 0

Selagi i kurang daripada 40, peningkatan i dengan +1 untuk setiap kali loop

Selagi i kurang daripada 40, peningkatan i dengan +1 untuk setiap kali loop

Pengkodean Mudah – FOR LOOP

CONTOH LAKARAN PENUH FOR LOOP

```
int red = 9;
```

```
void setup()
```

```
{  
  pinMode(red, OUTPUT);  
}
```

```
void loop()
```

```
{  
  for (int i = 0; i < 40; i++) //peningkatan kecerahan  
  {  
    analogWrite(red, i);  
    delay(50);  
  }  
  
  for (int i = 0; i < 40; i--) //penurunan kecerahan  
  {  
    analogWrite(red, i);  
    delay(50);  
  }  
}
```

Input Digital

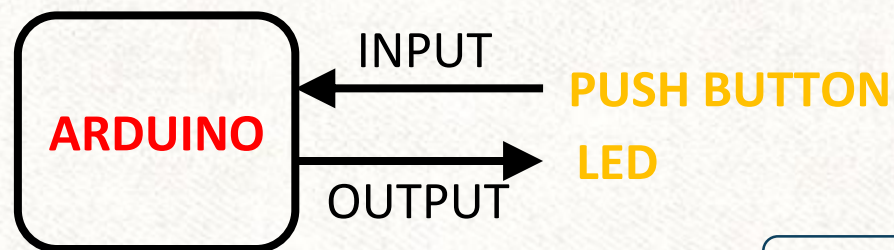
IDEA

Latihan ini akan meneroka keupayaan penderiaan digital Arduino. LED dihidupkan dengan menekan butang tekan, yang ditentukan oleh pernyataan bersyarat yang disebut IF/ELSE.

APA YANG AKAN DIPELAJARI

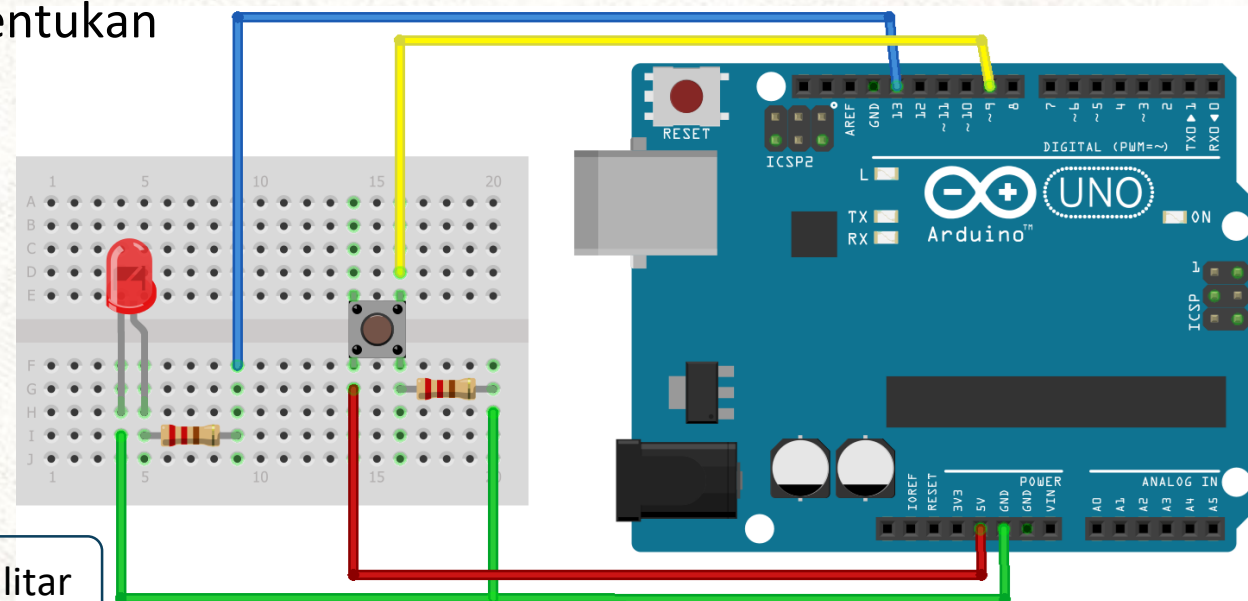
- Membaca daripada input digital untuk mengawal output
- Menggunakan pernyataan IF/ELSE untuk menentukan output logic.

GAMBARAJAH BLOK



Langkah 1: Bina litar

GAMBARAJAH LITAR

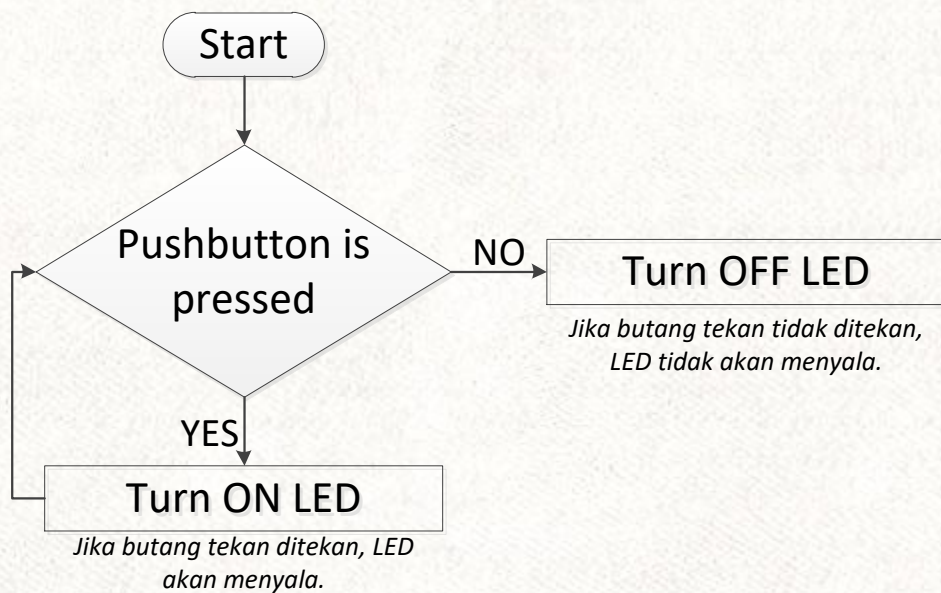


Input Digital


KOD ARAHAN	PENERANGAN
<pre>pinMode(pin, MODE);</pre>	<p>Gunakan kod ini untuk menyambungkan pin digital ke butang tekan untuk dikenal pasti sebagai input dan pin disambungkan kepada LED sebagai output.</p> <p>Contoh:</p> <pre>pinMode(9, INPUT); pinMode(13, OUTPUT);</pre>
<pre>If (conditional statement) { //action A } else { //action B }</pre>	<p>Kod arahan sampel di sebelah kiri, menunjukkan pelaksanaan “action A” apabila ia memenuhi syarat tersebut. Sekiranya tidak memenuhi syarat itu, ia akan melaksanakan “action B”</p> <p>Contoh:</p> <pre>If (digitalRead(9) == HIGH) { digitalWrite (13, HIGH); } else { digitalWrite 13, LOW); }</pre>

Output Digital

CARTA ALIRAN



LAKARAN KOD

Langkah 2: Membuka Arduino IDE 

```

void setup()
{
  pinMode(9, INPUT);
  pinMode(13, OUTPUT);
}

void loop()
{
  if (digitalRead(9) == HIGH)
  {
    digitalWrite (13, HIGH);
  }
  else
  {
    digitalWrite (13, LOW);
  }
}
  
```

Langkah 3: Tekan pada butang
UPLOAD untuk menghantar kod ke Arduino.



Jenis Data Arduino

JENIS DATA	JULAT NILAI
int	-32768 hingga 32768
float	-3.4028235 e38 hingga 3.4028235 e38
long	-2,147,483,648 hingga 2,147,483,648
unsigned int (tiada nilai negatif)	0 hingga 65535
char	-127 hingga 128
word	0 hingga 65535
byte	0 hingga 255
unsigned char	0 hingga 255

Operator Arduino

ARAHAN MATEMATIK	OPERATOR	CONTOH
Persamaan	=	$b = a$
Tambah	+	$a + b$
Tolak	-	$a - b$
Darab	*	$a * b$
Bahagi	/	a / b
Modul atau Selebihnya	%	$a \% b$

Operator Perbandingan

HUBUNGAN	OPERATOR
sama dengan	==
tidak sama dengan	!=
kurang daripada	<
lebih daripada	>
kurang daripada atau sama dengan	<=
lebih daripada atau sama dengan	>=

Operator Boolean

1. && mewakili "AND gate"

INPUT A	INPUT B	OUTPUT
Salah	Salah	Salah
Salah	Betul	Salah
Betul	Salah	Salah
Betul	Betul	Betul

2. || mewakili "OR gate"

INPUT A	INPUT B	OUTPUT
Salah	Salah	Salah
Salah	Betul	Betul
Betul	Salah	Betul
Betul	Betul	Betul

3. ! mewakili "NOT gate"

INPUT A	INPUT B
Salah	Betul
Betul	Salah

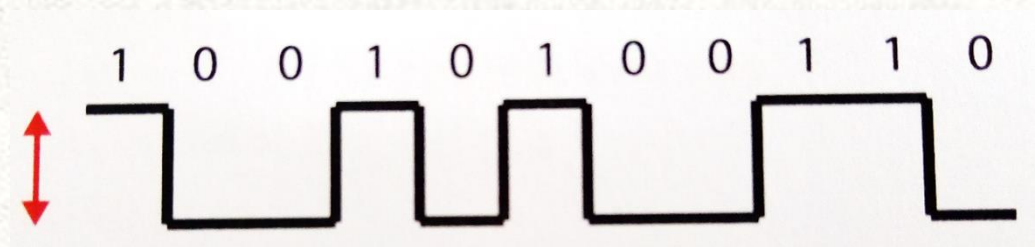
Komunikasi Serial

Arduino telah membina Library yang dipanggil Serial yang membolehkan kita berkomunikasi dari papan pengawal ke komputer melalui port USB. Dalam pemindahan data bersiri, satu bit dihantar pada satu-satu masa selepas yang lain.

Jika kita mempunyai sensor yang disambungkan ke salah satu pin analog, kita boleh menghantar nilai yang akan dipaparkan pada monitor komputer kita. Untuk menggunakan Serial Library, kod di bawah diperlukan:

```
void setup()  
{  
  Serial.begin (9600); //memulakan komunikasi serial  
}
```

Nilai 9600 menentukan kadar baud (bit per pemindahan kedua), iaitu kadar maklumat yang akan lulus dari Arduino ke komputer atau sebaliknya.



0 dan 1 juga dikenali sebagai bit, yang lalu antara komputer dan Arduino melalui wayar USB.

Fungsi Library Serial:

1. **Serial.print** () – Cetak data ke port siri
2. **Serial.read** () – Membaca data siri yang masuk
3. **Serial.write** () – Menulis data binary ke port siri
4. **Serial.available** () – Dapatkan bilangan aksara yang tersedia dari port bersiri

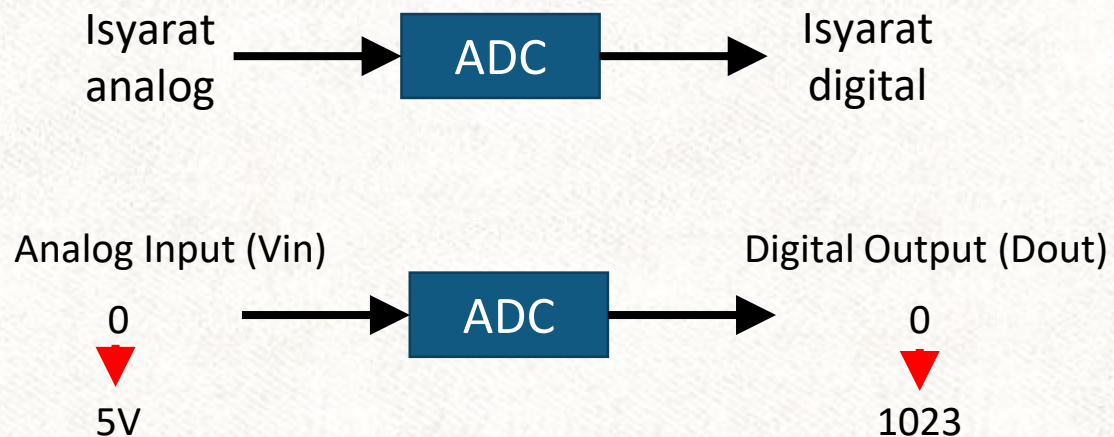
Input Analog



Kawalan sensor

Membaca input digital hanya membolehkan pengesanan dua keadaan isyarat binari HIGH dan LOW. **Dunia sebenar berfungsi dengan analog** dengan pengukuran yang berbeza-beza seperti suhu, tekanan, daya, berat, dan sebagainya. Sensor menukar sifat-sifat fizikal ini kepada sifat elektrik bersamanya, iaitu voltan, yang akan dibaca oleh mikropengawal.

Mikropengawal hanya bertindak dalam digital, oleh itu Analog to Digital Converter (ADC) diperlukan untuk menukar isyarat voltan analog ke nilai digital yang sepadan.



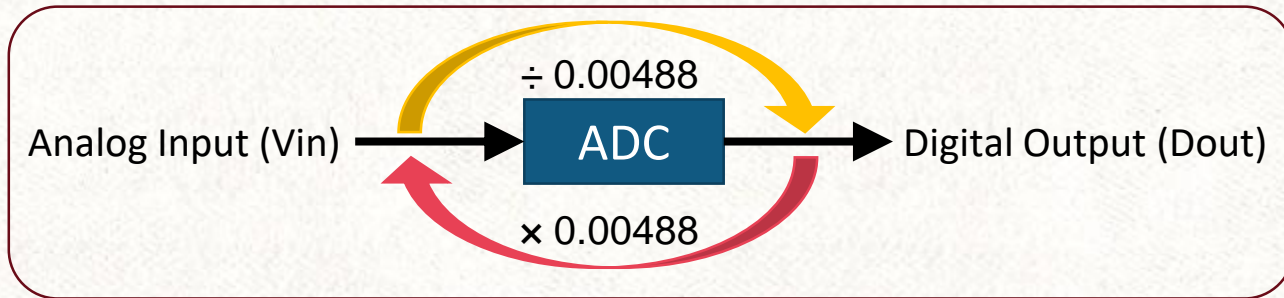
Output digital mempunyai 1024 bilangan nilai antara 0 dan 1023 untuk mewakili voltan masukan analog. Ini memberi kita resolusi berikut:

$$\text{Resolution} = \frac{\text{PWM Output (Vout)}}{5V} = \frac{5V}{1024} = 0.00488V$$

$$\text{Digital Output (Dout)} = \frac{V_{in}}{\text{Resolution}} = \frac{V_{in}}{0.00488V}$$

Input Analog

KIRAAN PRAKTIKAL ADC



Input analog (V_{in}) boleh menerima sebarang nilai antara 0 - 5V dan akan dibahagikan dengan resolusi ADC sebanyak 0.00488 untuk menghasilkan keluaran digital bersamaan (D_{out}).

$$\text{Digital Output (Dout)} = \frac{V_{in}}{\text{Resolution}} = \frac{3.3}{0.00488V} = 676$$

Input Analog

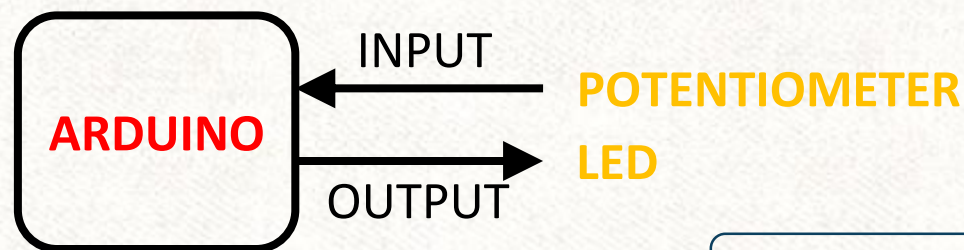
IDEA

Dalam latihan ini, kita akan membaca nilai-nilai dari pin analog tertentu yang menerima input dari potensiometer. Nilai akan dipaparkan pada monitor siri.

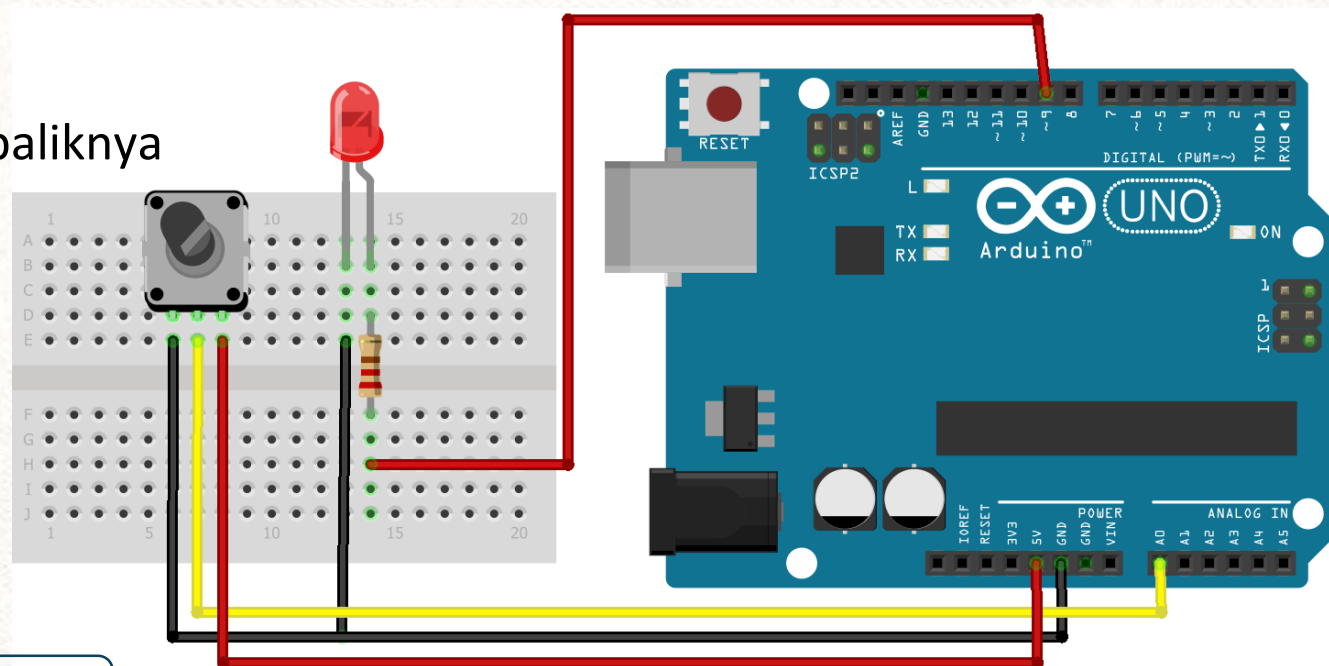
APA YANG AKAN DIPELAJARI

- Membaca input analog
- Menukar nilai analog kepada voltan dan sebaliknya
- Komunikasi siri

GAMBARAJAH BLOK



Langkah 1: Bina litar

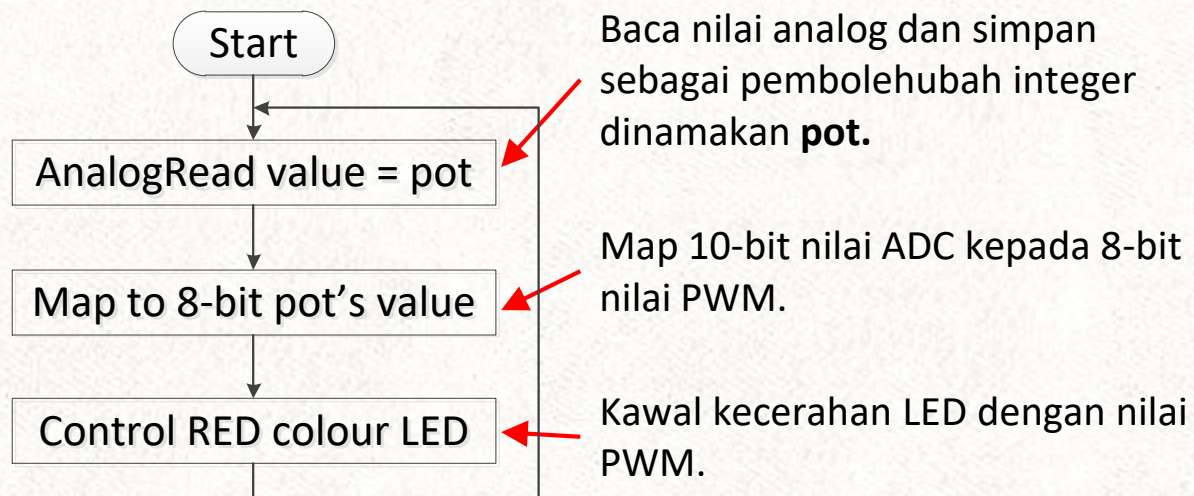


Input Analog


KOD ARAHAN	PENERANGAN
<pre>pot = map(pot, 0, 1023, 0, 255);</pre>	<p>“map” menukarkan nombor dari satu julat ke satu julat yang lain. Dalam arahan tis, nilai 0 - 1023 sedang diulang ke 0 - 255.</p> <p>Ini sedang dilaksanakan sebagai analogWrite () Arduino hanya menerima nilai 8-bit (maksimum 255) manakala analogRead () menghasilkan nilai 10 bit (maksimum 1023).</p>

Input Analog

CARTA ALIRAN




LAKARAN KOD

Langkah 2: Membuka Arduino IDE 

```

void setup()
{
  pinMode(9, OUTPUT);
  Serial.begin(9600);
}

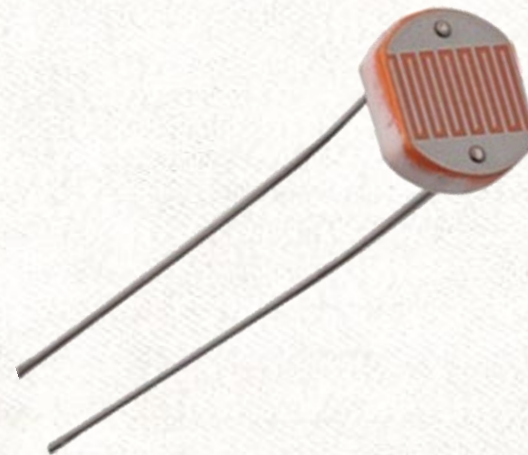
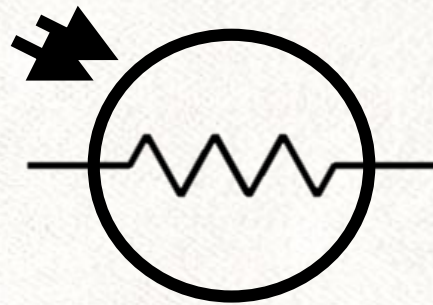
void loop()
{
  int pot = analogRead(A0);
  pot = map(pot, 0, 1023, 0, 255);
  analogWrite(9, pot);
  Serial.println(pot);
  delay (500);
}
  
```

Langkah 3: Tekan pada butang  UPLOAD untuk menghantar kod ke Arduino.

Input Analog : SENSOR [LDR]

LIGHT SENSOR (LDR)

Light Dependent Resistor (LDR) atau juga dikenali sebagai Photo-resistor adalah sensor ringan yang mudah digunakan **untuk menguji intensiti cahaya**. Mengurangkan jumlah cahaya yang diterima akan meningkatkan rintangannya, sehingga menjadikannya berkadar songsang.



LDR bukan sensor yang sangat tepat untuk mengukur nilai cahaya LUX. Tetapi, ia **boleh melakukan kerja yang baik dalam mengesan bayang-bayang dan perubahan yang cahaya yang jelas** (iaitu jika lampu di dalam bilik itu hidup atau mati). Dalam projek yang memerlukan pengesanan jelas perubahan dalam keamatan cahaya, LDR adalah pilihan yang hebat.

Input Analog : SENSOR [LDR]

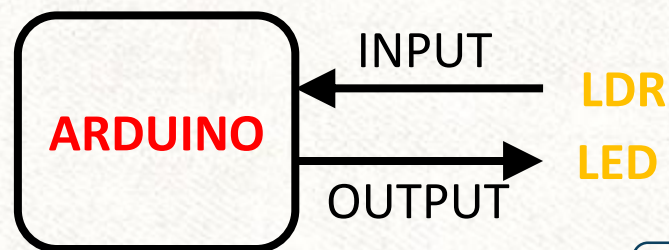
IDEA

Menghasilkan kawalan cahaya LED menggunakan LDR

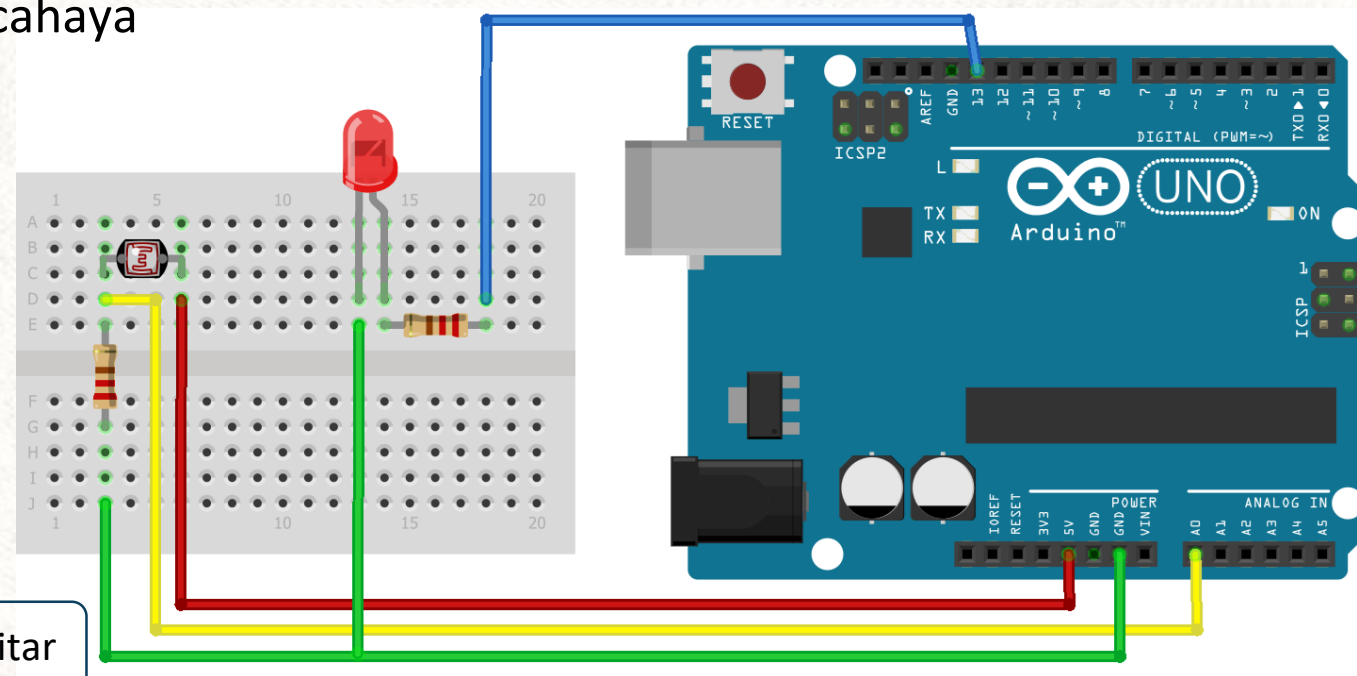
APA YANG AKAN DIPELAJARI

- Membaca analog sensor cahaya
- Mengawal LED berdasarkan nilai intensiti cahaya

GAMBARAJAH BLOK



GAMBARAJAH LITAR



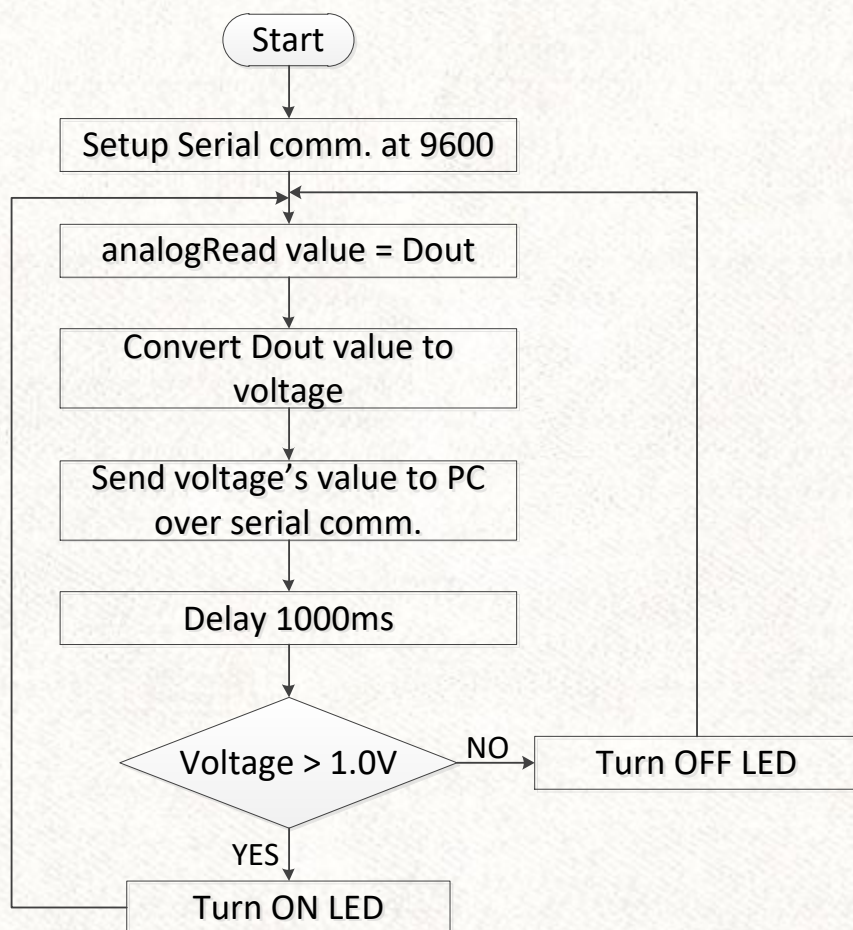
Langkah 1: Bina litar

Input Analog : SENSOR [LDR]

KOD ARAHAN	PENERANGAN
<pre>int Dout = analogRead (A0);</pre>	Kod ini dibaca dari pin analog A0 dan menyimpan nilai pembolehubah yang dipanggil dalam Dout.
<pre>float Vin = (Dout*0.00488);</pre>	Menukar nilai Dout digital ke voltan yang bersamaan. 0.00488 merujuk kepada resolusi ADC seperti yang dibincangkan sebelumnya.
<pre>Serial.print (9600);</pre>	Inisiasi komunikasi siri dengan baudrate yang ditentukan.
<pre>Serial.println ();</pre>	Arahan untuk mencetak data ke monitor siri. "ln" akan memaparkan barisan baru untuk setiap gelung dan akan dipaparkan secara menegak.
<pre>if (conditional statement) { //action A } else { //action B }</pre>	Kod arahan sampel di sebelah kiri, menunjukkan pelaksanaan "action A" apabila ia memenuhi syarat tersebut. Sekiranya tidak memenuhi syarat itu, ia akan melaksanakan "action B" Contoh: <pre>if (Vout > 1.0) { digitalWrite (13, HIGH); } else { digitalWrite (13, LOW); }</pre>

Input Analog : SENSOR [LDR]

CARTA ALIRAN




LAKARAN KOD

```

int LDR_pin = A0;    //analog pin A0
void setup()
{
  Serial.begin (9600);
  pinMode(13, OUTPUT);
}

void loop()
{
  int LDR_Reading = analogRead(LDR_pin);
  float Vout = LDR_Reading *0.00488;
  Serial.println (Vout);
  delay (1000);
  if (Vout > 0.3)
  { digitalWrite (13, LOW); }
  else
  { digitalWrite (13, HIGH); }
}
  
```

Langkah 2: Membuka Arduino IDE 

Langkah 3: Tekan pada butang UPLOAD untuk menghantar kod ke Arduin



Projek Arduino

```
for (i = 0; i < 3; i++) {  
    unsigned int cp_count;  
    unsigned int len = n;  
    if (user(group_info[i]) {  
        return -EFAULT;  
    }  
    grouplist += NGS_PER_BLOCK;  
    count += cp_count;  
}
```

```
group_info = kmalloc(usr)  
if (!group_info)  
    return NULL;  
group_info->ngroups = gidsetsize;  
group_info->nblocks = nblocks;  
atomic_set(&group_info->usage, 1)
```

```
if (gidsetsize <= NGROUPS_SMALL)  
    group_info->blocks[0] = g;  
else {  
    for (i = 0; i < nb; i++) {  
        gid_t *b;  
        b = (void *)_get(GFP);  
        if (!b)  
            goto partial_alloc;  
        group_info->blo[i] = b;  
    }  
}
```

```
struct group_info  
int nblocks;
```

```
void groups_free(struct) {  
    if (group_info->blocks[0]) {  
        int i;  
        for (i = 0; i < 1; i++)  
            free_page();  
    }  
    free(group_info);  
}
```



Projek : LAMPU RUMAH AUTOMATIK BOLEH LARAS

PERNYATAAN MASALAH

- Ibu anda kadang kala terlupa untuk mematikan suis lampu di dalam rumah apabila sudah pagi.
- Adik anda tidak suka tidur malam dalam keadaan gelap dan tidak pula terlalu terang.

TUGASAN

- Bina litar ringkas menggunakan sensor yang telah anda pelajari untuk membantu ibu anda dengan masalah yang dihadapi.
- Bina litar lengkap menggunakan potentiometer untuk mengawal intensiti cahaya bagi membantu adik anda tidur malam.

MASA YANG DIPERUNTUKKAN

1 JAM



TIPS !

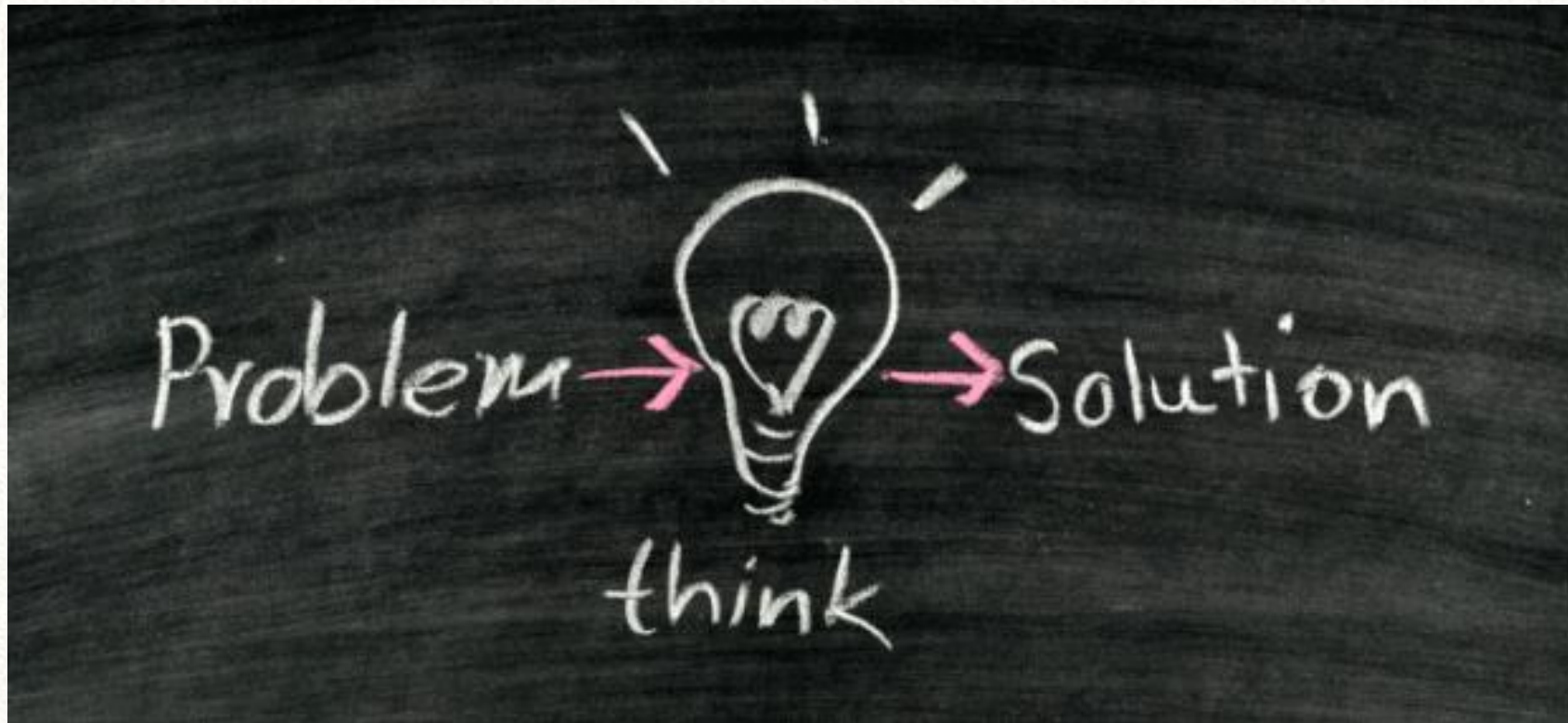
- Tugas 6
- Tugas 7

MODUL 3 : Jurutera Cilik

'Newbie Engineers'

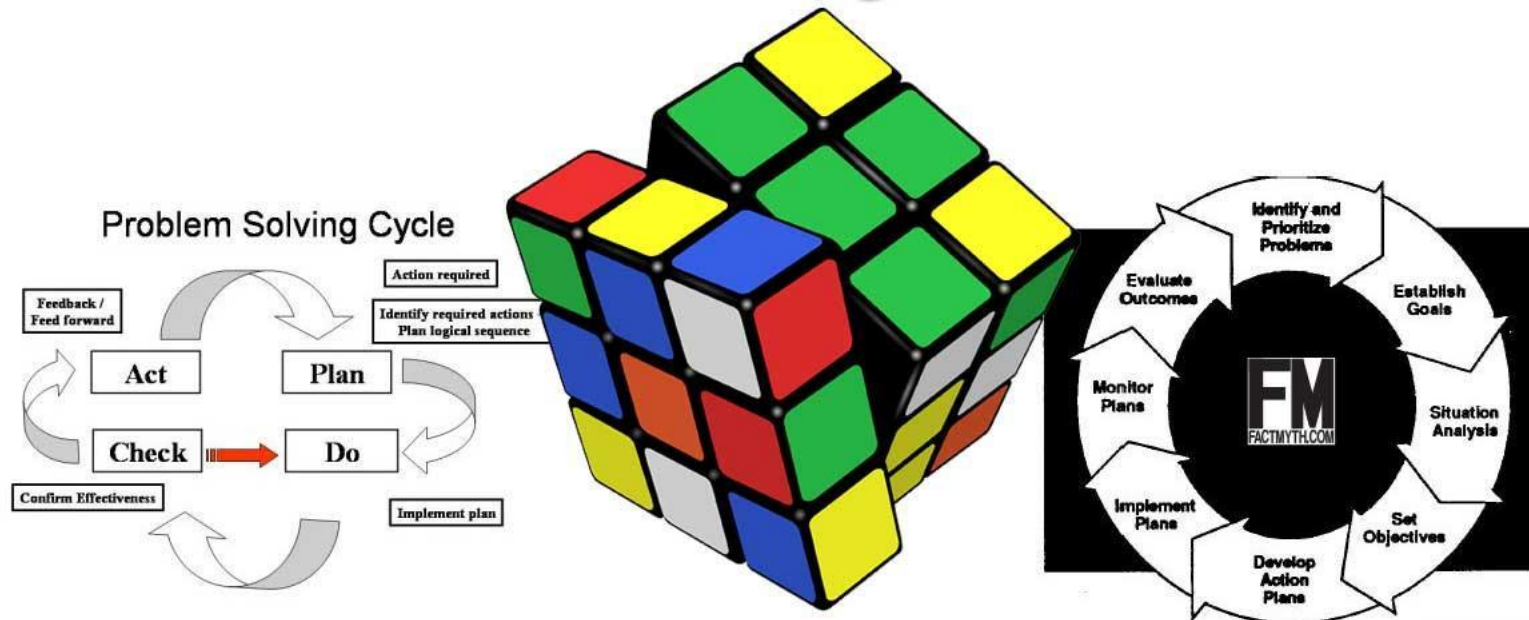
Apa yang kita perlukan sebagai jurutera?

- Kemahiran berfikir

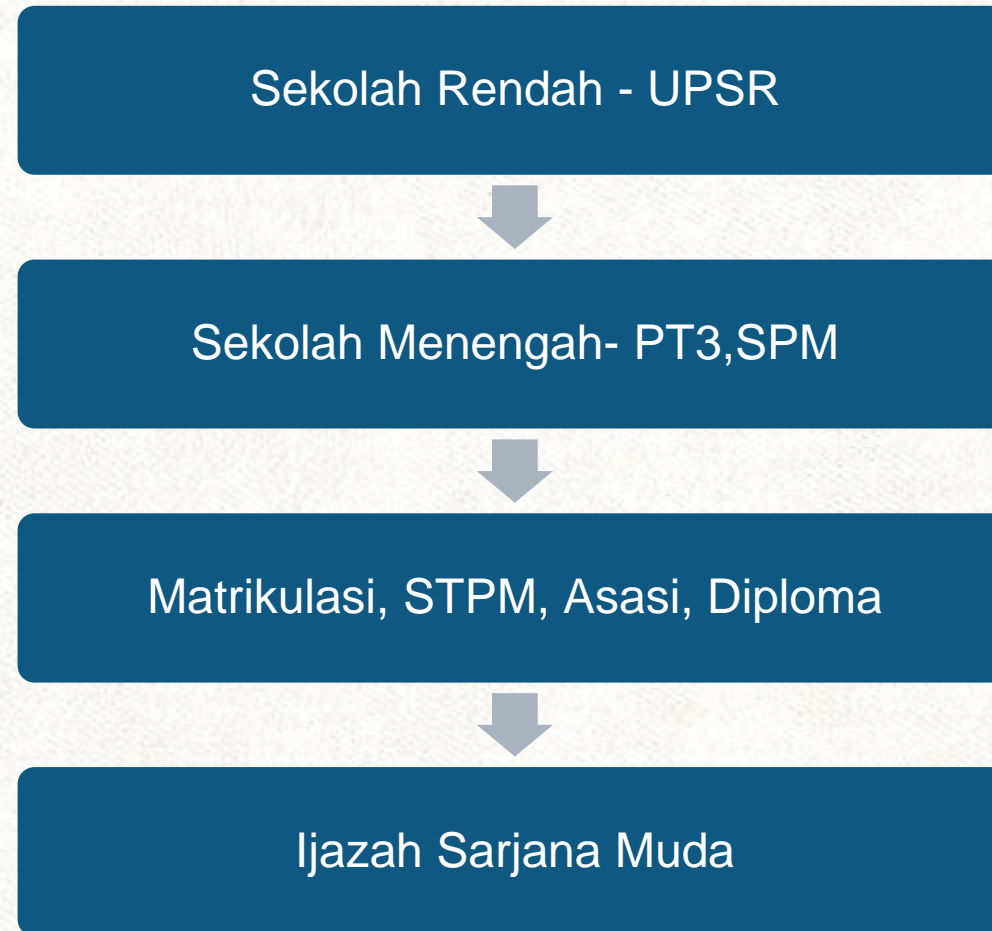


Tahukah anda?...

Is Identifying the Problem Harder than Finding the Solution?



Langkah-langkah Untuk Menjadi Jurutera



Fizik

Matematik

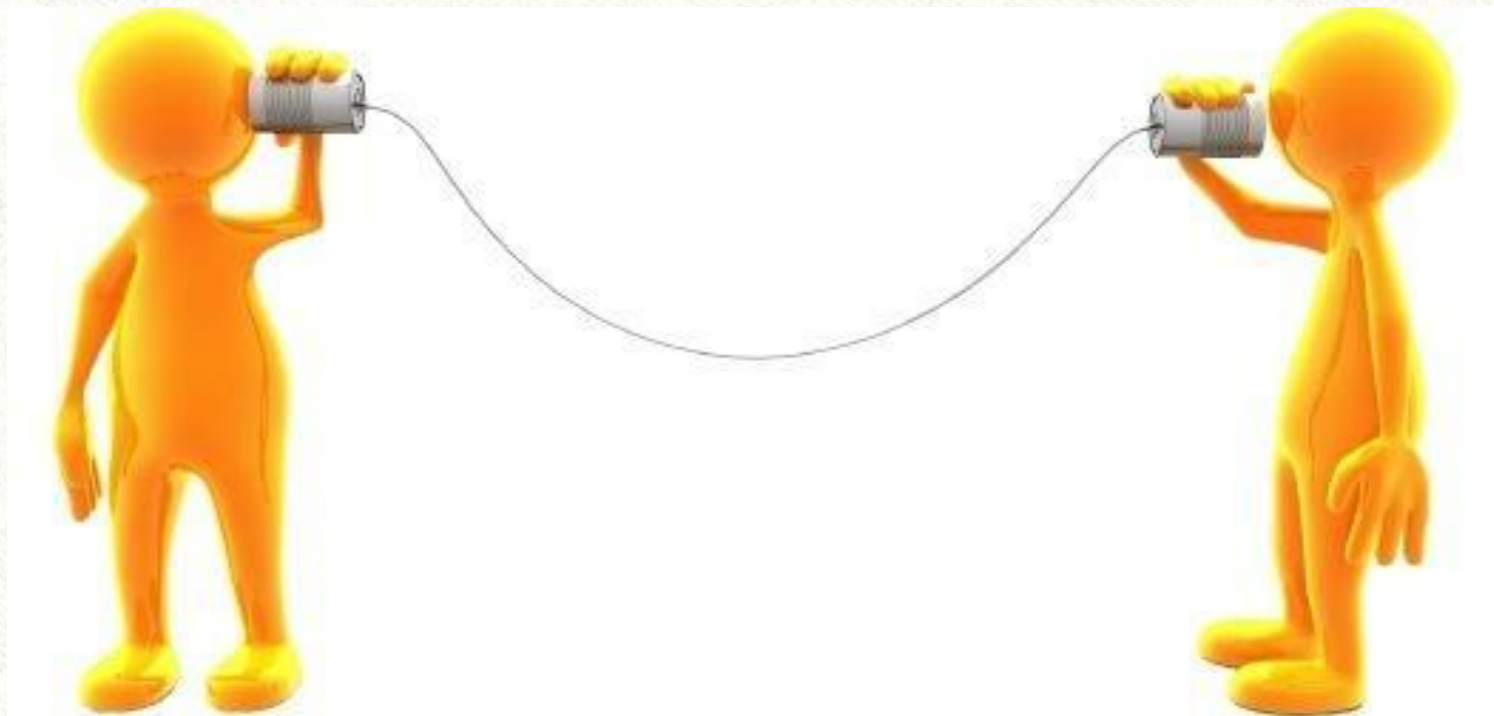
Subjek Utama

Kimia

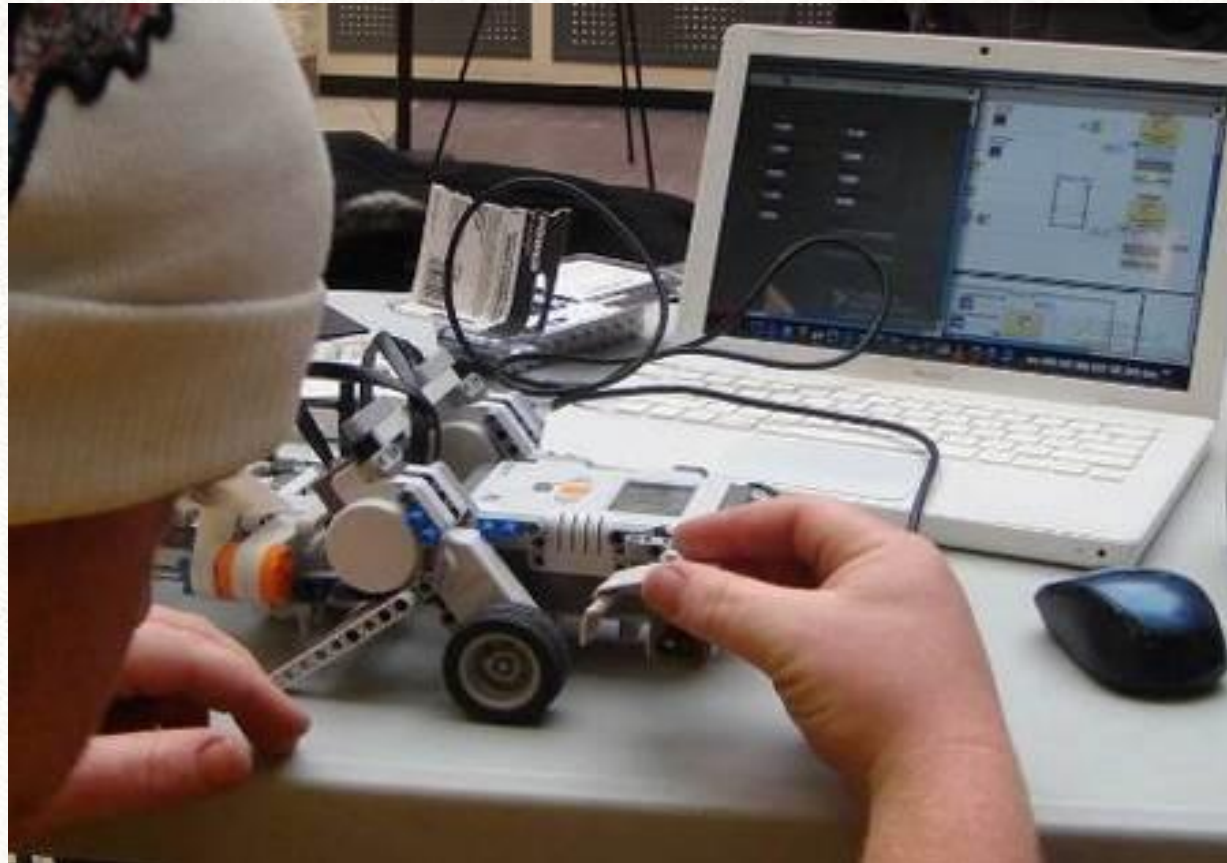
Bahasa Inggeris

Kemahiran Yang Diperlukan

1) Kemahiran berkomunikasi



- 2) Kemahiran Kejuruteraan



3) Kemahiran Teknikal



- Kemahiran berkerja secara berkumpulan



Bidang-bidang Kejuruteraan

Bidang kejuruteraan mempunyai dapat dibahagikan kepada beberapa pengkhususan tertentu seperti :

- Kejuruteraan awam
- Kejuruteraan elektrik
- Kejuruteraan elektronik
- Kejuruteraan kimia
- Kejuruteraan pembuatan
- Kejuruteraan pertanian
- Kejuruteraan sumber asli
- Kejuruteraan mekanikal
- Kejuruteraan bahan
- Kejuruteraan komputer
- Kejuruteraan aero angkasa
- Kejuruteraan telekomunikasi

Kesimpulan

- Malaysia memerlukan lebih banyak jurutera di Malaysia.
- Untuk mengambil bidang kejuruteraan, kita memerlukan persiapan rapi bermula dari prestasi akademik dan disiplin diri.
- Untuk menjadi jurutera yang baik, kebolehan untuk belajar secara berterusan contohnya menulis kod, memberi kelebihan kepada seseorang jurutera.

Terima Kasih
&
Selamat Maju Jaya

