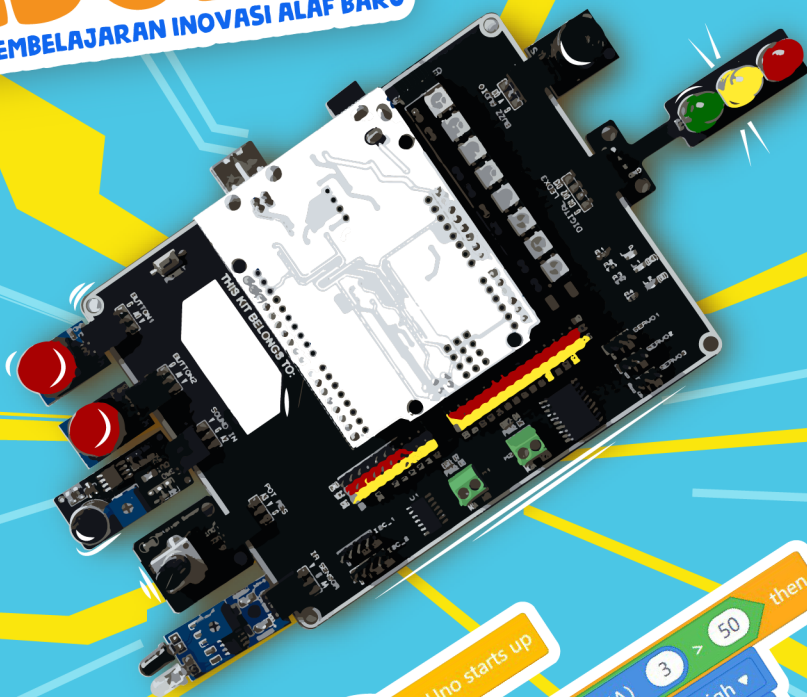


REKA EDUKIT

KIT PEMBELAJARAN INOVASI ALAF BARU



- Reka dan bangunan pelbagai inovasi berteknologi.
- Papan mikropengawal mudah dinaik taraf.
- Menggunakan perisian sumber terbuka ("open-source")
- Pengkodan grafik yang mudah dan seronok.

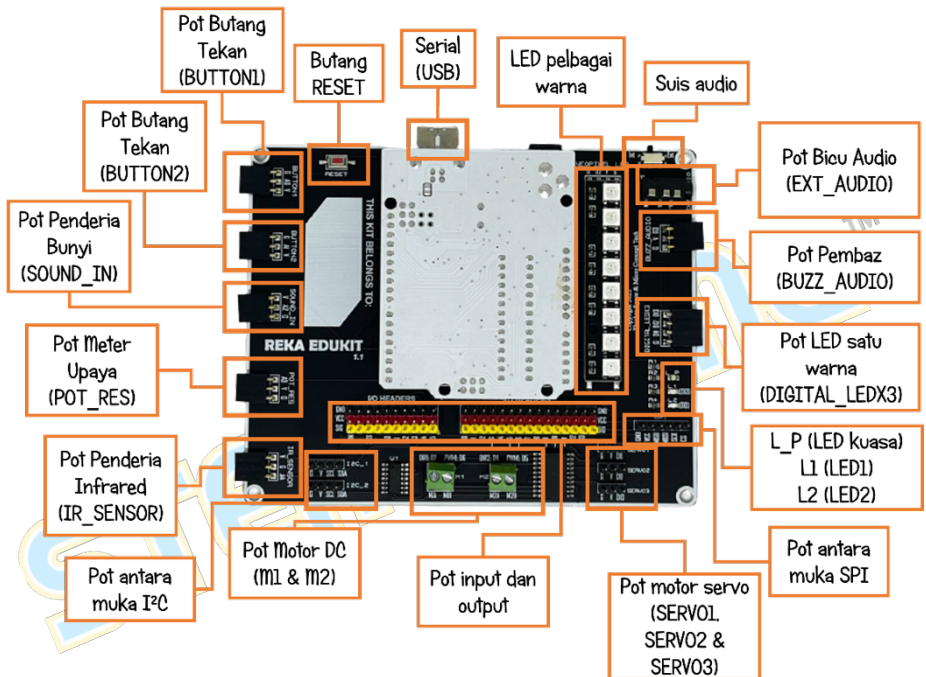


ISI KANDUNGAN

ANATOMI REKA EDUKIT	1
PEMASANGAN PERISIAN MBLOK V5.....	4
CARA MENAMBAH REKAEDUKIT	6
PENGENALAN KEPADA SISTEM ROBOTIK.....	7
1: BUTANG TEKAN KAWALAN KOMUNIKASI.....	8
2: LAMPU HARI RAYA.....	11
3: LAMPU ISYARAT LALU LINTAS	13
4: SISTEM PENGGERA.....	18
5: PESTA LAMPU.....	21
6: SISTEM PEJALAN KAKI	26
7: ALAT MUZIK ALAF BARU	31
8: SISTEM LED PELBAGAI WARNA.....	37
9: SISTEM SEKURITI.....	40
10: SISTEM BILIK DARJAH.....	48
11: SISTEM PENGHALAU BURUNG.....	64
12: SISTEM KESELAMATAN KERETA.....	78
13: PEMANTAU PENDERIA TANPA WAYAR.....	90
14: PENGAWAL LAMPU.....	102
15: KELAS FIZIK.....	130
16: KIPAS CEKAP TENAGA.....	136
17: PEMANTAU POKOK HIASAN	139
18: PENGECAMAN BAHAN KITAR SEMULA AUTOMATIK.....	149

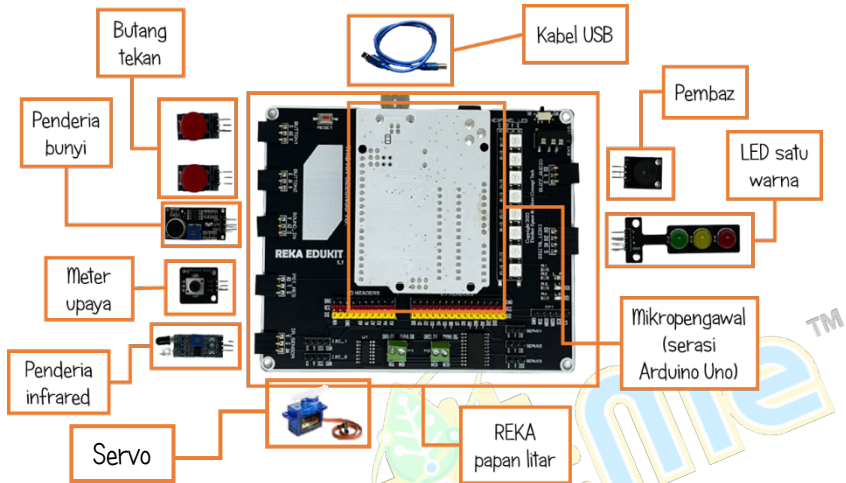
ANATOMI REKA EDUKIT

Peranti di atas papan litar.

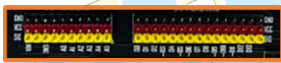


ANATOMI REKA EDUKIT

Peranti persisian Reka Edukit.



Peranti antara muka



Pot Input dan Output (I/O)

Port I/O membenarkan sambungan kepada perkakasan. Perkakasan ini mungkin dalam atau luaran. Port disambungkan dengan litar tembaga dan jalur memori yang membenarkan komunikasi data antara CPU, RAM dan port itu sendiri.



SPI

SPI (Serial Peripheral Interface) ialah bas¹ antara muka yang biasa digunakan untuk menghantar data antara mikropengawal dan peranti kecil seperti daftar peralihan, penderia dan kad SD.



I²C

I²C ialah singkatan untuk "Inter-Integrated Circuit". Ia adalah protokol sambungan antara muka bas¹ yang digunakan dalam peranti komunikasi bersiri.



Motor DC

Motor arus terus (DC) ialah sejenis motor elektrik yang berputar menukar tenaga elektrik arus terus (DC) kepada tenaga mekanikal.



Motor Servo

Motor servo ialah penggerak berputar atau penggerak lurus yang boleh mengawal kedudukan sudut atau lurus, halaju dan pecutan dengan tepat.

ANATOMI REKA EDUKIT

Butiran nombor pin peranti dan peranti antara muka di Arduino Uno

POT PERANTI	PIN ARDUINO UNO	PERANTI
BUTTON1	A0	Butang Tekan 1
BUTTON2	A1	Butang Tekan 2
SOUND IN	A2	Penderia Bunyi
POT RES	A3	Meter Upaya
IR SENSOR	A4	Penderia Infrared
BUZZ AUDIO	D3	Pembaz
DIGITAL_LEDX3 (RED)	D11	LED Satu Warna (Merah)
DIGITAL_LEDX3 (YELLOW)	D12	LED Satu Warna (Kuning)
DIGITAL_LEDX3 (GREEN)	D13	LED Satu Warna (Hijau)
NEOPIXEL_LED	D2	LED Pelbagai Warna (Neopixel)
SERVO1	D8	Motor Servo 1
SERVO2	D9	Motor Servo 2
SERVO3	D10	Motor Servo 3
MOTOR1 DIRECTION	D7	Motor DC 1 (Arah Pusingan)
MOTOR1 PWM (SPEED)	D6	Motor DC 1 (PWM / Kelajuan Motor)
MOTOR2 DIRECTION	D4	Motor DC 2 (Arah Pusingan)
MOTOR2 PWM (SPEED)	D5	Motor DC 2 (PWM / Kelajuan Motor)
I2C – SCL	A5	I2C – SCL
I2C – SDA	A4	I2C – SDA
SPI – MISO	D12	SPI – MISO
SPI – MOSI	D11	SPI – MOSI
SPI – SCK	D13	SPI – SCK
SPI – CS	D10	SPI – CS
LED1	D13	Led 1
LED2	D0	Led 2
BLUETOOTH TX*	D11	Bluetooth Tx
BLUETOOTH RX*	D12	Bluetooth Rx

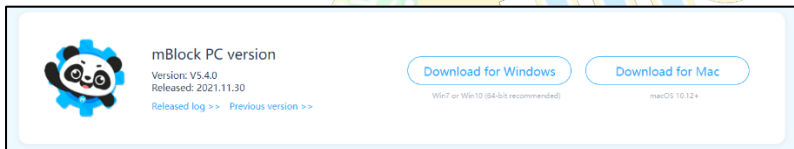
PEMASANGAN PERISIAN MBLOK V5

Langkah 1 Perisian mBlok v5 boleh didapatkan daripada

Link: <https://www.mblock.cc/en/download/> @



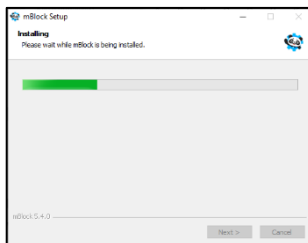
Langkah 2 Muat turun versi terkini mBlok v5 berdasarkan system pengendalian komputer.



Langkah 3 Klik mBlok v5 pada lokasi muat turun anda.

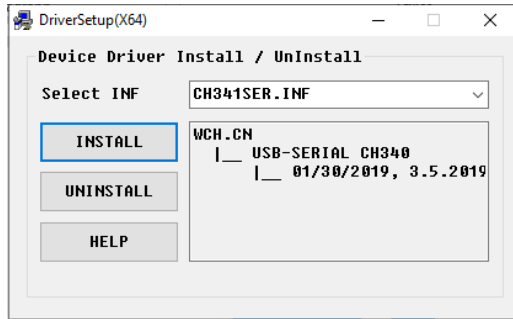


Langkah 4 Tunggu sehingga pemasangan mBlok v5 selesai.



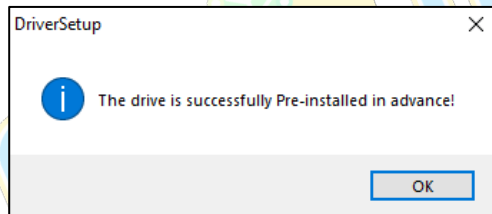
Langkah 5

Klik "INSTALL"



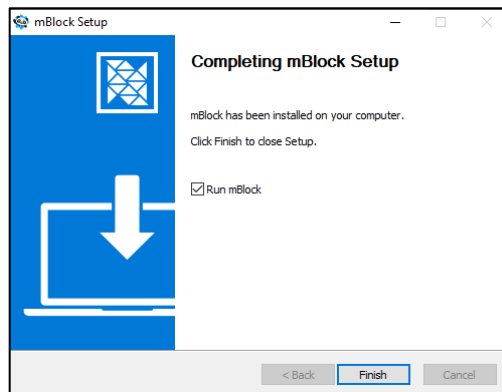
Langkah 6

Klik OK dan keluar



Langkah 7

Tandakan "Run mBlok". Klik "Finish".



CARA MENAMBAH REKAEDUKIT

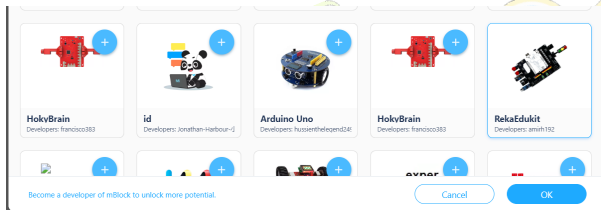


Langkah 1 Buka mBlok v5

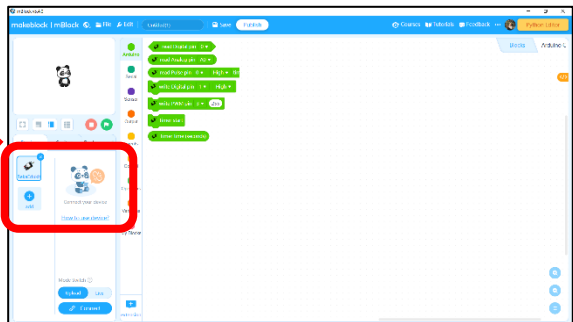
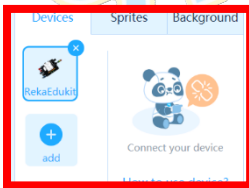


Langkah 2 tekan button add

Langkah 3 Pilih peranti RekaEduKit dan tekan ok.



Langkah 4 Sekarang anda boleh menikmati menggunakan mBlok v5!



PENGENALAN KEPADA SISTEM ROBOTIK

Definisi Sistem Robotik

- ✓ Sistem robotik adalah hasil gabungan **bidang Kejuruteraan** dan **Sains Komputer**. Ia merupakan satu bidang sains yang berkaitan dengan **penghasilan reka bentuk, pembuatan dan pemasangan** serta **operasi** yang berkaitan dengan robot.
- ✓ Penggunaan robot dapat dilihat dalam industri **automotif, pertanian, makanan, elektrik** dan lain-lain.



Penderia (Sensor)

Pergerakan (Output)



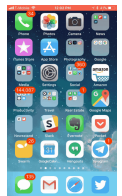
Elemen Robotik



Kuasa & Wayar



Kawalan & Pengaturcaraan



1: BUTANG TEKAN KAWALAN KOMUNIKASI

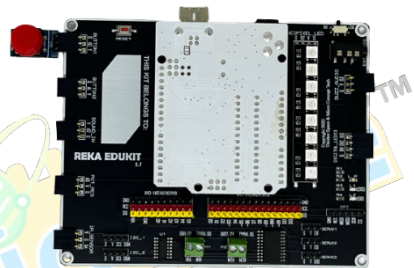
Di dalam projek ini kita akan memaparkan perkataan pada monitor bersiri.

PENGENALAN BUTANG TEKAN

Butang tekan (push button) ialah peranti mekanikal yang digunakan untuk mengawal litar elektrik. Digunakan untuk menggerakkan mekanisma pensuisan dalaman.

PERANTI YANG DIGUNAKAN :

1. Papan litar RekaEdukit
2. Butang tekan 



TUTORIAL

1. Seretkan blok **When Arduino Uno starts** berserta blok **Serial port begin**.

When Arduino Uno starts

Serial port begin 9600 ▼

Tetapkan nilai baud kepada 9600.

2. Seretkan blok **After Arduino Uno starts** berserta blok **if, then**

When Arduino Uno starts

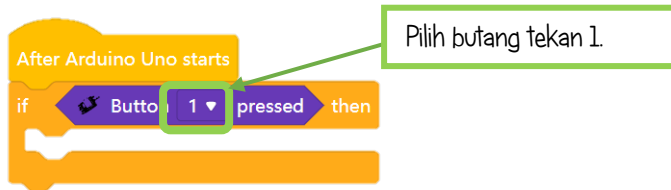
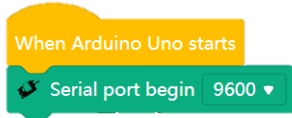
Serial port begin 9600 ▼

After Arduino Uno starts

if then

Blok ini terlaksana jika pernyataan yang telah ditetapkan adalah benar (true).

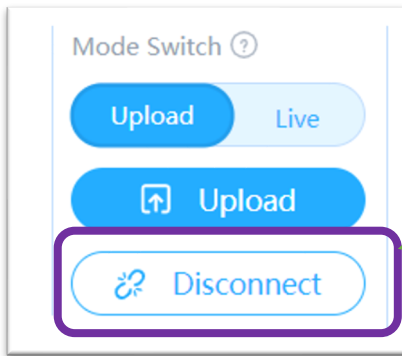
3. Seterusnya, tambah blok **Button pressed** ke dalam ruang heksagon di dalam blok **if, then**.



4. Tambah blok **Serial port print in line Reka** di dalam blok **if, then** diikuti dengan blok **delay**.

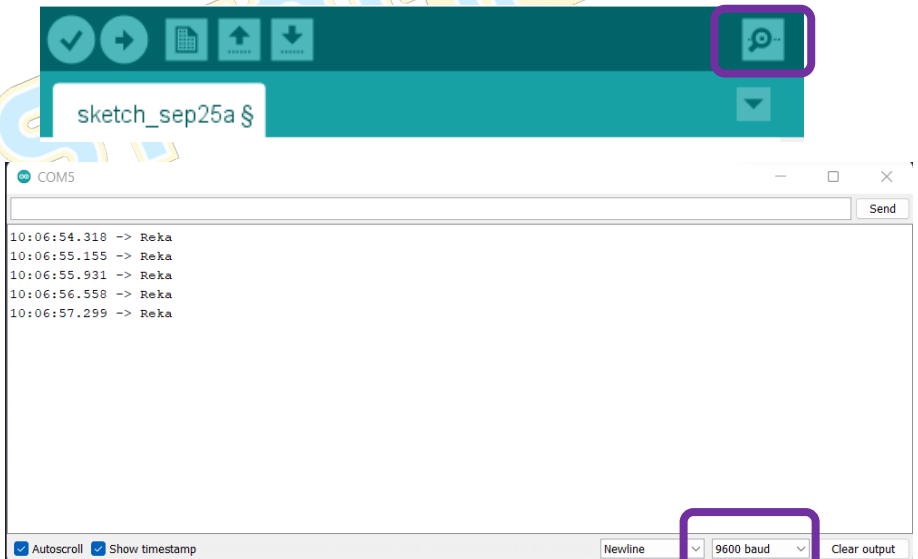


- Untuk melihat paparan pengeluaran, putus sambungan papan RekaEduKit daripada MBlok.



Tekan ikon berikut.

- Buka perisian **Arduino IDE**. Kemudian tekan ikon kanta pembesar di sebelah kanan. Ia akan membuka monitor bersiri serta memaparkan perkataan.



Tetapkan nilai baud kepada 9600 baud.

2: LAMPU HARI RAYA

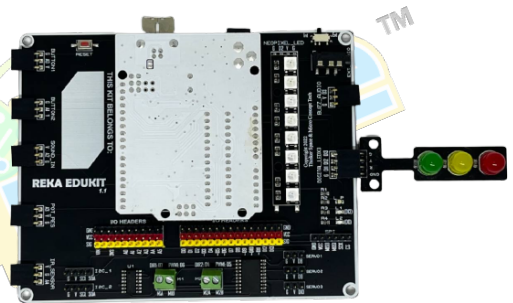
Di dalam projek ini kita akan menyalakan lampu LED satu warna (GYR LED) mengikut urutan yang telah ditetapkan.

PENGENALAN GYR LED

LED satu warna (GYR LED) merupakan modul paparan lampu isyarat mini yang mempunyai kecerahan yang tinggi, bersaiz kecil dan pendawaian mudah. Ia boleh disambung ke PWM untuk mengawal kecerahan LED.

PERANTI YANG DIGUNAKAN :

1. Papan litar RekaEdukit
2. LED satu warna (GYR LED)

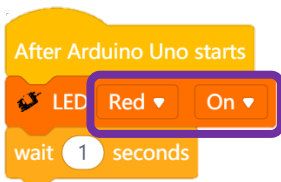


TUTORIAL

1. Seretkan blok **After Arduino Uno starts**.

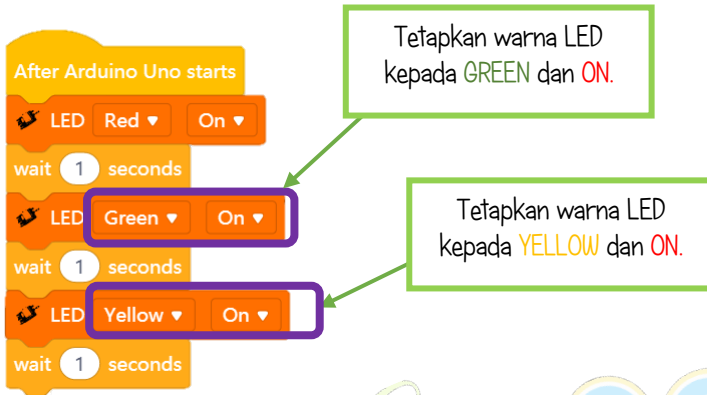
After Arduino Uno starts

2. Masukkan blok **LED** diikuti dengan blok **delay**.

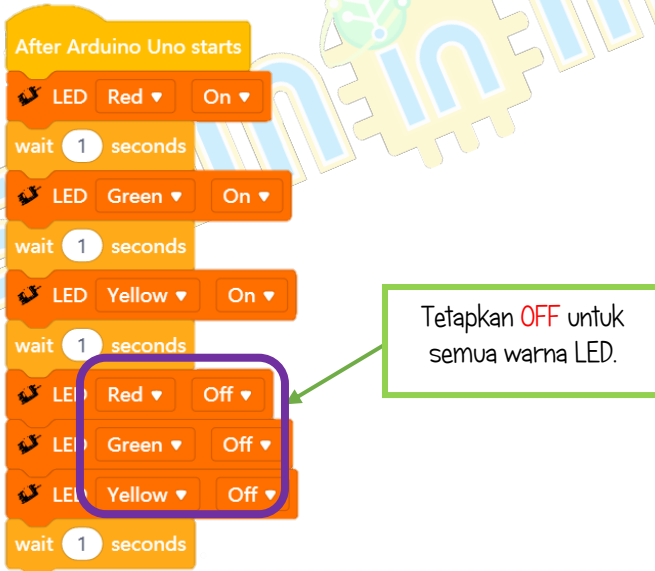


Tetapkan warna LED kepada **RED** dan **ON**.

3. Ulang Langkah 2 untuk setiap warna LED.



4. Kemudian, tambah 3 blok LED secara berturutan dan diakhiri oleh blok delay.



3: LAMPU ISYARAT LALU LINTAS

Projek ini bertujuan untuk mempraktikkan penggunaan lampu isyarat lalu lintas. Kita boleh mengawal warna LED satu warna (GYR LED) dalam satu-satu masa. Jika penderia infrared mengesan sesuatu LED berwarna hijau akan menyala. Untuk menukar kepada warna kuning, kita perlu menekan butang tekan 2. Akhir sekali, untuk menukar LED warna merah, kita perlu menekan butang tekan 1.

PERANTI YANG DIGUNAKAN :

1. Papan litar RekaEdukit

2. Butang tekan 

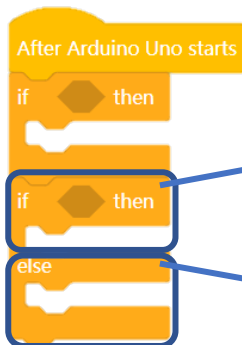
3. LED satu warna (GYR LED) 

4. Penderia infrared 



TUTORIAL

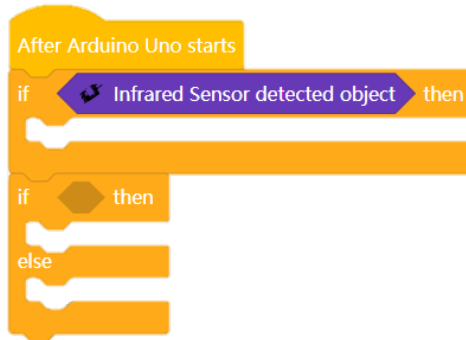
1. Seretkan **After Arduino Uno starts** berserta blok **if, then** dan blok **if, then, else**.



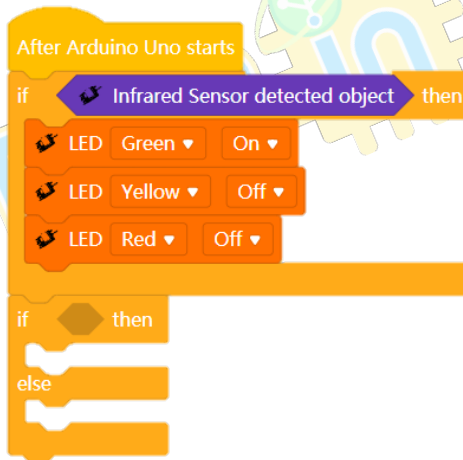
Blok ini terlaksana jika pernyataan yang telah ditetapkan adalah benar (true).

Blok ini terlaksana jika pernyataan yang telah ditetapkan di atas adalah tidak benar (false).

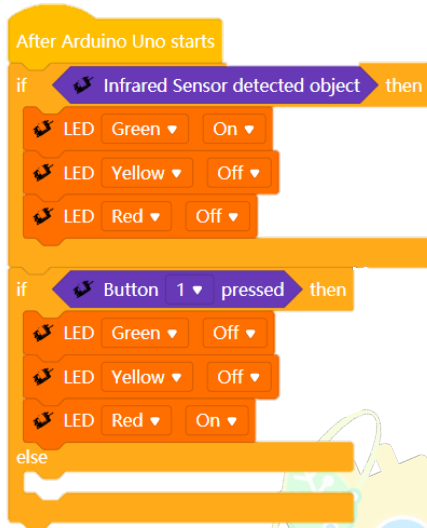
2. Masukkan blok **Infrared Sensor detected object** ke dalam ruang heksagon di dalam blok **if. then**.



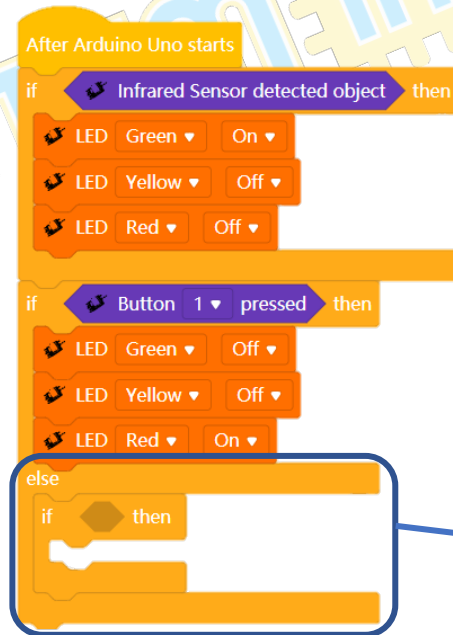
3. Kemudian, masukkan blok **LED** untuk setiap warna **Hijau**, **Kuning** dan **Merah**.



4. Seret dan masukkan blok **button 1 pressed** ke dalam ruang heksagon di dalam blok **if. then. else**.
- 5.



6. Seret dan masukkan blok **if. then** ke dalam blok **else**.



Blok ini terlaksana jika pernyataan yang telah ditetapkan di atas adalah tidak benar (false) dan jika pernyataan yang telah ditetapkan di sini adalah benar (true).

7. Masukkan blok **button 2 pressed** ke dalam ruang heksagon di dalam blok **if. then**.



8. Tambahkan blok LED untuk setiap warna Hijau, Kuning dan Merah.



4: SISTEM PENGGERA

Di dalam projek ini kita akan membina sebuah sistem penggera. Penggera akan berbunyi jika penderia infrared (IR sensor) mengesan sesuatu objek di hadapan. Seterusnya, untuk menghentikan bunyi penggera, kita perlu menekan butang 1.

PENGENALAN PENDERIA INFRARED

Penderia infrared (IR Sensor) digunakan untuk mengesan objek di hadapan.

Jarak objek yang dikesan boleh dilaras.

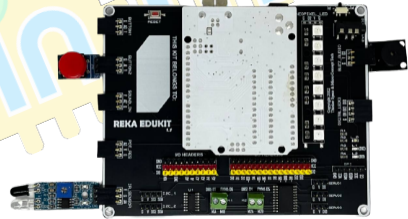
PERANTI YANG DIGUNAKAN :

1. Papan litar RekaEdukit

2. Butang tekan 

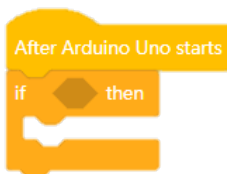
3. Penderia infrared 

4. Pembaz 

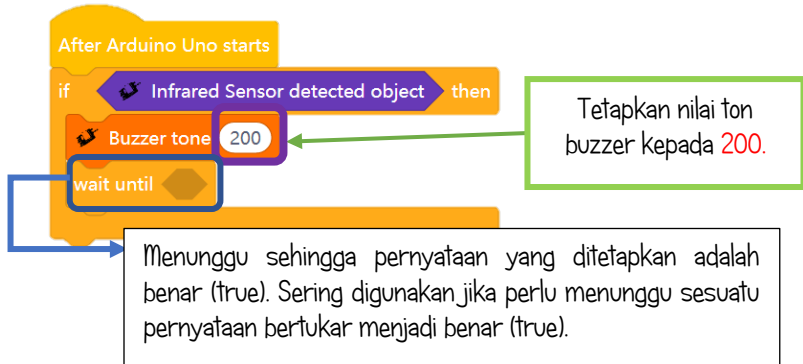


TUTORIAL

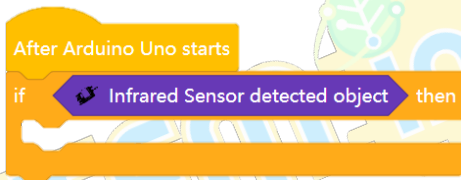
1. Seretkan **After Arduino Uno starts** berserta blok **forever** dan blok **if. then**.



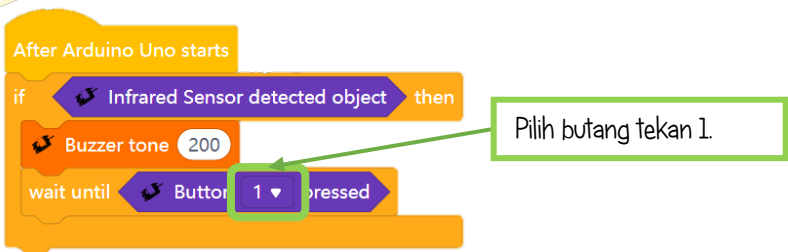
- Seterusnya, masukkan blok **Infrared sensor detect object** ke dalam ruang heksagon di dalam blok **if. then**.



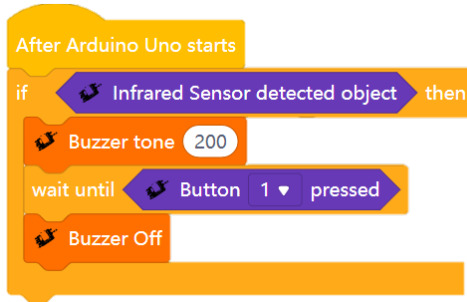
- Kemudian, tambah **buzzer tone** berserta blok **wait until** di dalam blok **if. then**.



- Seretkan blok **button pressed** ke dalam ruang heksagon di dalam blok **wait until**.



5. Tambahkan blok **Buzzer Off** di bawah blok **wait until**.



STEM in me™

5: PESTA LAMPU

Di dalam projek ini pula, kita akan menggunakan meter upaya (potentiometer) untuk menyalakan LED pelbagai warna (neopixel LED).

PENGENALAN METER UPAYA

Meter upaya (potentiometer) bertindak sebagai pembahagi voltan boleh laras. Potentiometer diubah secara manual untuk mengawal aliran arus elektrik.

PENGENALAN LED PELBAGAI WARNA (NEOPIXEL LED)

Setiap LED pelbagai warna (Neopixel LED) dikawal oleh litar bersepadu yang memproses maklumat dan menukarkannya kepada data untuk mengawal cahaya.

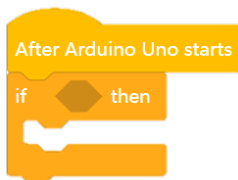
PERANTI YANG DIGUNAKAN :

1. Papan litar RekaEdukit
2. Meter upaya
3. LED pelbagai warna (Neopixel LED)



TUTORIAL

1. Seretkan blok **After Arduino Uno starts** berserta blok **if. then**.

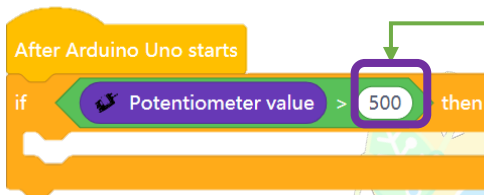


2. Tambah blok **greater than** ke dalam ruang heksagon di dalam blok **if. then**.



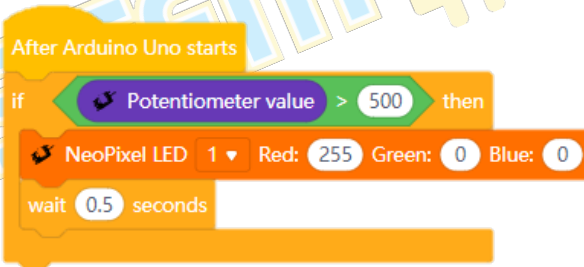
Blok ini digunakan untuk membandingkan nilai sebelah kiri dan nilai sebelah kanan. Jika nilai kiri lebih besar daripada nilai kanan, pernyataan ini dikira benar (true).

3. Kemudian, tambah blok **potential value** ke dalam blok **greater than**.

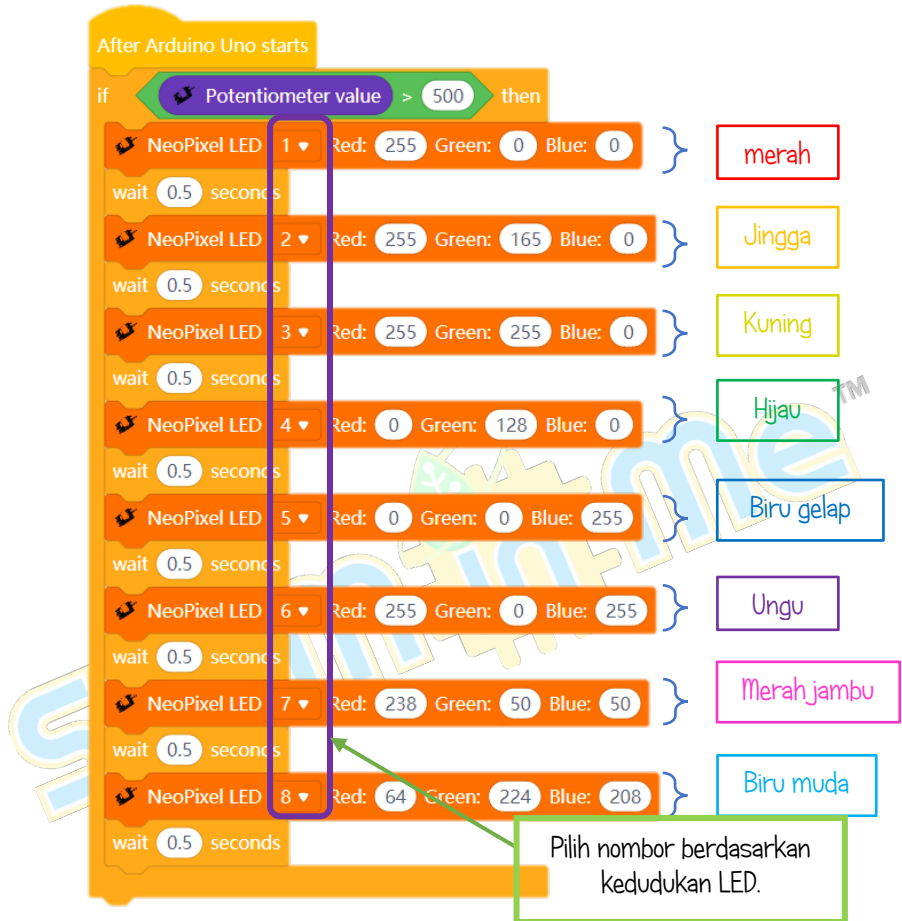


Tetapkan nilai kepada
500.

4. Seterusnya, seret dan masukkan blok **Neopixel LED** berserta **delay** ke dalam blok **if. then**.



5. Ulang Langkah 4 untuk setiap 8 blok Neopixel LED.



6. Seretkan Neopixel LED off di bawah blok if, then.



7. Ulang Langkah 6 untuk setiap 8 Neopixel LED.

```

After Arduino Uno starts
if Potentiometer value > 500 then
  NeoPixel LED 1 Red: 255 Green: 0 Blue: 0
  wait 0.5 seconds
  NeoPixel LED 2 Red: 255 Green: 165 Blue: 0
  wait 0.5 seconds
  NeoPixel LED 3 Red: 255 Green: 255 Blue: 0
  wait 0.5 seconds
  NeoPixel LED 4 Red: 0 Green: 128 Blue: 0
  wait 0.5 seconds
  NeoPixel LED 5 Red: 0 Green: 0 Blue: 255
  wait 0.5 seconds
  NeoPixel LED 6 Red: 255 Green: 0 Blue: 255
  wait 0.5 seconds
  NeoPixel LED 7 Red: 238 Green: 50 Blue: 50
  wait 0.5 seconds
  NeoPixel LED 8 Red: 64 Green: 224 Blue: 208
  wait 0.5 seconds
  NeoPixel LED 1 Off
  NeoPixel LED 2 Off
  NeoPixel LED 3 Off
  NeoPixel LED 4 Off
  NeoPixel LED 5 Off
  NeoPixel LED 6 Off
  NeoPixel LED 7 Off
  NeoPixel LED 8 Off
  
```

Pilih nombor berdasarkan kedudukan LED.

6: SISTEM PEJALAN KAKI

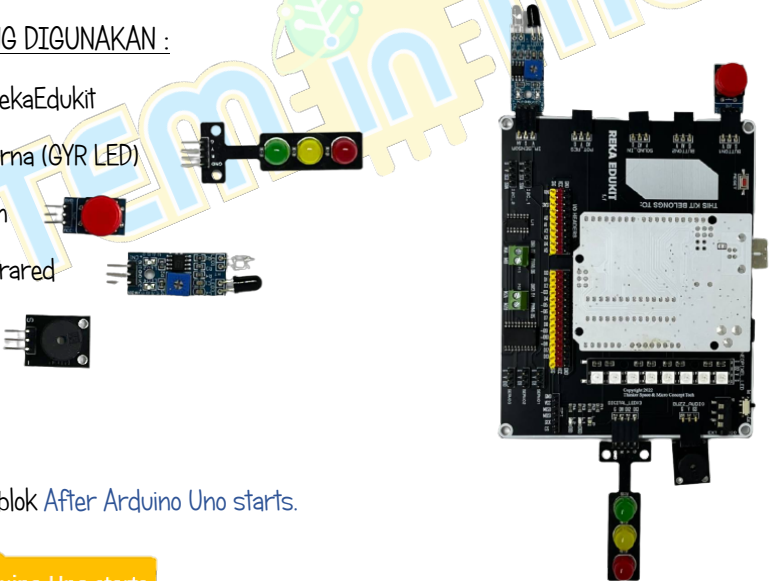
Kita akan membina sistem pejalan kaki menggunakan Peranti yang disenaraikan di bawah. Di dalam sistem ini, LED satu warna akan menyala warna **MERAH**. Apabila, penderia infrared mengesan sesuatu dan pengguna menekan butang tekan pada waktu yang sama, pembaz akan berbunyi dan LED satu warna akan menyala warna **HIJAU**.

PENGENALAN PEMBAZ

Pembaz (buzzer) merupakan peranti isyarat bunyi yang boleh menukarkan isyarat audio kepada isyarat bunyi.

PERANTI YANG DIGUNAKAN :

1. Papan litar RekaEdukit
2. LED satu warna (GYR LED)
3. Butang tekan
4. Penderia infrared
4. Pembaz



TUTORIAL

1. Seretkan blok *After Arduino Uno starts*.

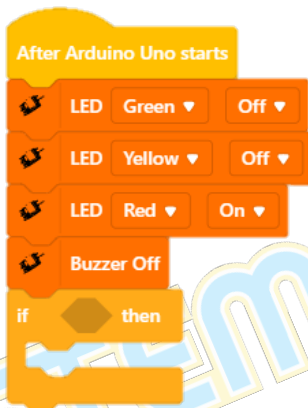
After Arduino Uno starts

- Masukkan 3 blok **LED** untuk setiap warna LED.



Pilih warna LED mengikut susunan berikut. Kemudian, pilih **ON** untuk **LED MERAH**.

- Seterusnya, seret dan masukkan blok **buzzer off** diikuti blok **if. then** di bawah blok **LED**.

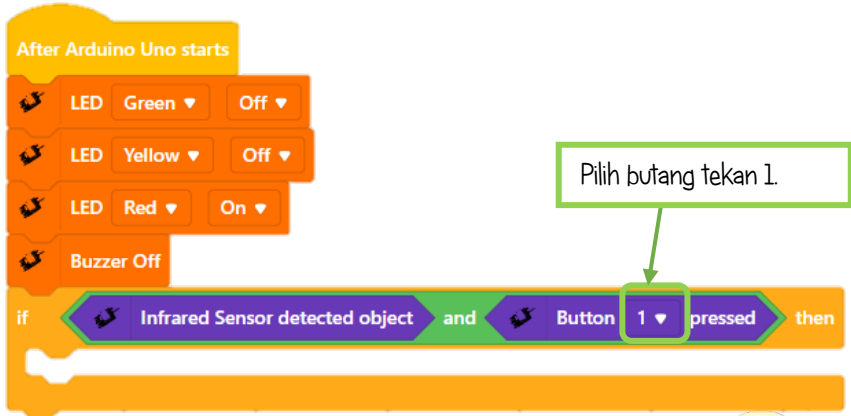


- Masukkan blok **and** ke dalam ruang heksagon di dalam blok **if. then**.



Blok ini digunakan jika ingin kedua-dua pernyataan yang ditetapkan menjadi benar(true).

5. Masukkan blok Infrared sensor detected object dan blok button pressed ke dalam blok if and.



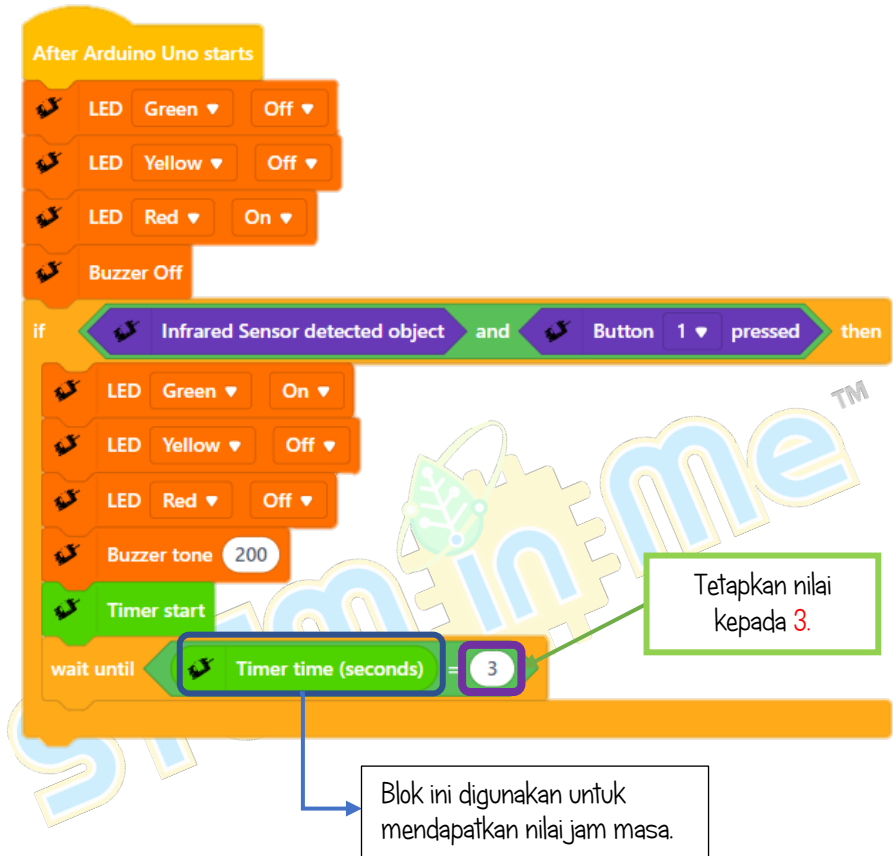
6. Kemudian, masukkan blok LED untuk setiap warna di dalam blok if, then.



7. Seterusnya, masukkan blok **buzzer tone**, **timer start** dan **wait until**.



8. Masukkan blok **timer time** ke dalam blok **equal** kemudian masukkan ke ruang heksagon di dalam blok **wait until**.

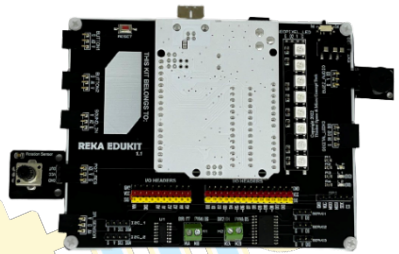
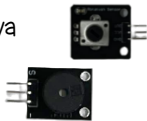


7: ALAT MUZIK ALAF BARU

Di dalam projek ini, kita akan membina alat muzik alaf baru. Pembaz akan berbunyi dengan nada yang berbeza apabila meter upaya dipusingkan mencapai nilai yang ditetapkan.

PERANTI YANG DIGUNAKAN :

1. Papan litar RekaEdukit
2. Meter upaya
3. Pembaz

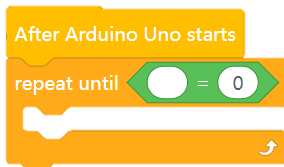


TUTORIAL

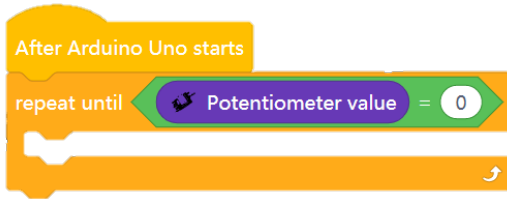
1. Seretkan **After Arduino Uno starts** diikuti blok **repeat until**.



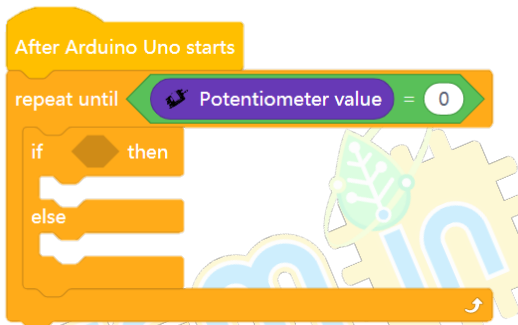
2. Tambahkan blok **Equal** ke dalam ruang heksagon di dalam blok **repeat until**.



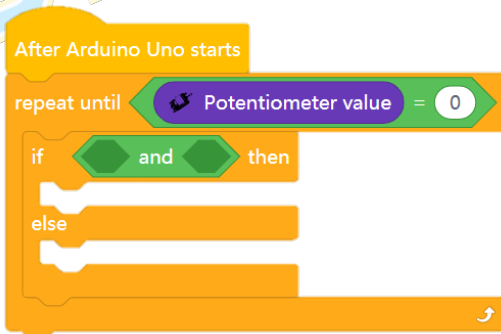
- Masukkan blok **potentiometer value** ke dalam blok **equal**.



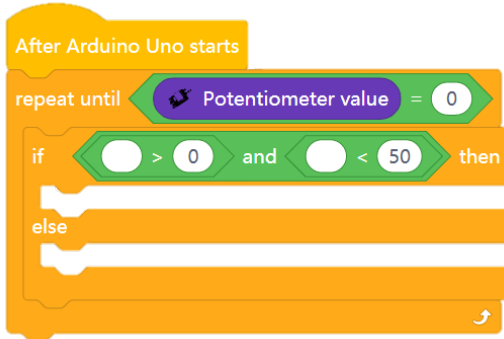
- Seterusnya, masukkan blok **if. then. else** ke dalam blok **repeat until**.



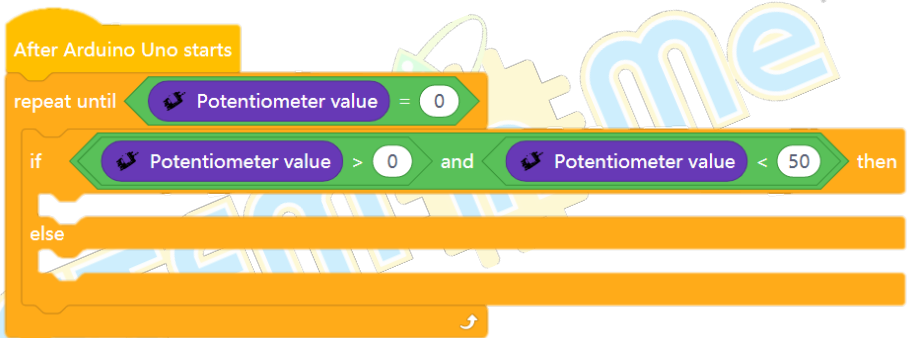
- Masukkan blok **and** ke dalam ruang heksagon di dalam blok **if. then**.



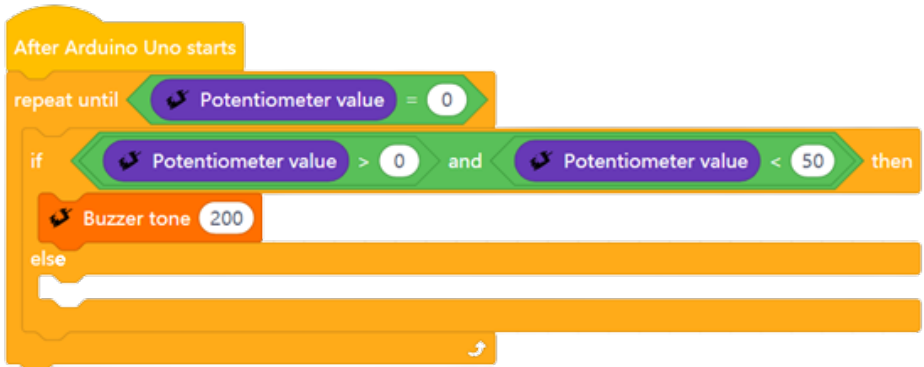
6. Masukkan blok **greater than** dan **less than** ke dalam blok **and**.



7. Kemudian, masukkan blok **potentiometer value** ke dalam ke dua-dua blok **greater than** dan **less than**.



8. Seret dan masukkan blok **buzzer tone** ke dalam blok **if, then**.



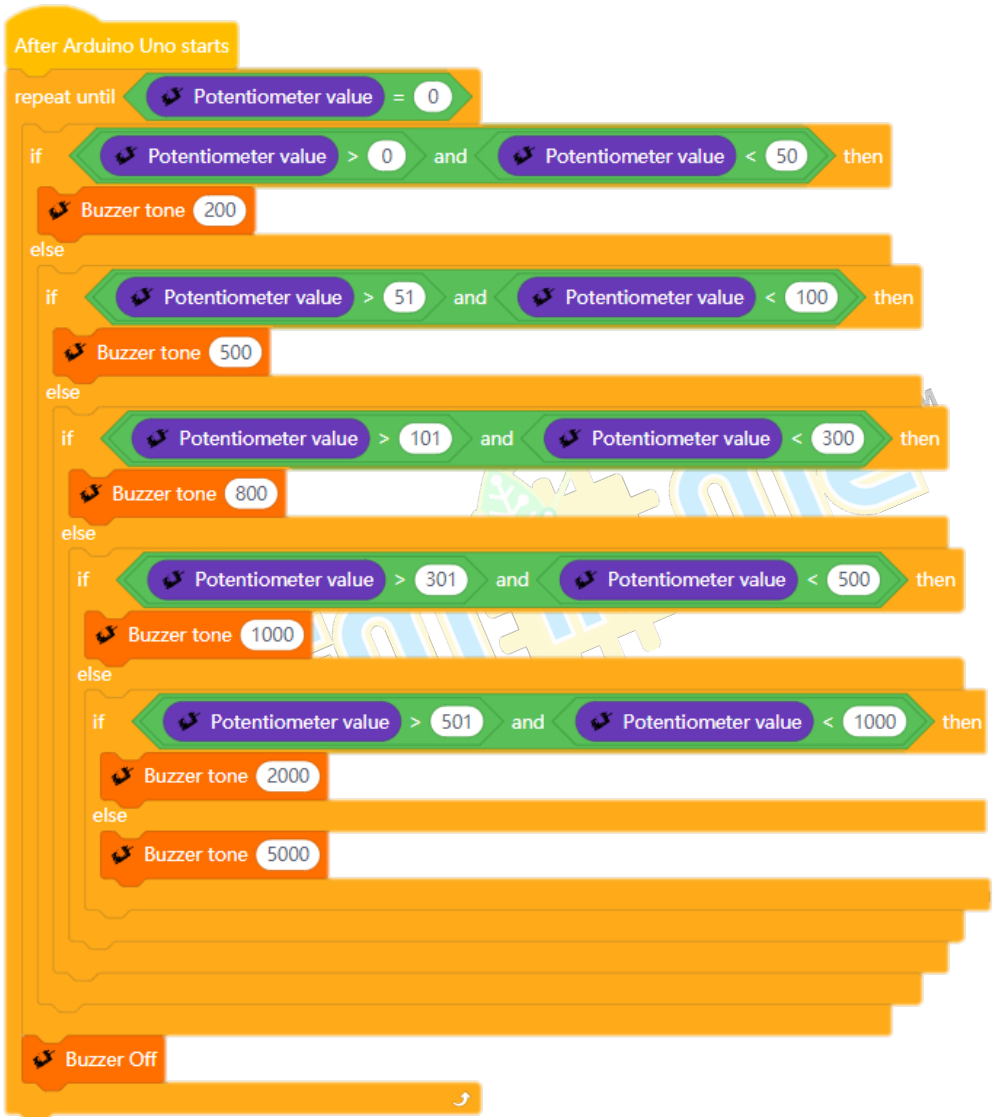
9. Ulang Langkah 4 hingga Langkah 8 dengan nilai meter upaya dan nilai nada pembaz yang berlainan. masukkan ke dalam blok setiap blok else.



10. Masukkan blok **buzzer tone** ke dalam blok **else** terakhir.



11. Seret dan masukkan blok **buzzer off** ke dalam blok **repeat until**.



8: SISTEM LED PELBAGAI WARNA

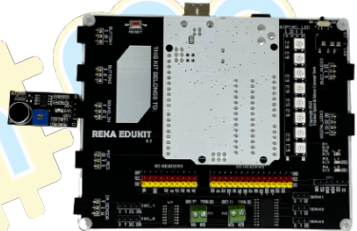
Kita akan membina projek yang boleh menyalakan LED pelbagai warna (neopixel LED) menggunakan bunyi.

PENGENALAN PENDERIA BUNYI (SOUND SENSOR)

Penderia bunyi (sound sensor) menukarkan getaran kepada isyarat audio dengan bantuan mikrofon. Ia bertindak sama seperti telinga manusia apabila bunyi dikesan. Kekuatan bunyi yang dikesan boleh dilaras

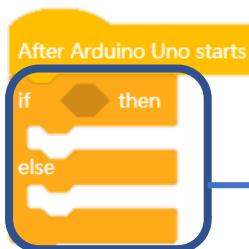
PERANTI YANG DIGUNAKAN :

1. Penderia bunyi
2. LED pelbagai warna (Neopixel LED)



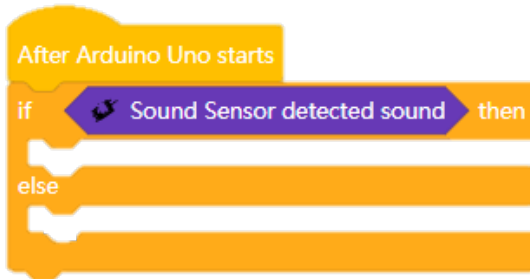
TUTORIAL

1. Seretkan **After Arduino Uno starts** berserta blok **forever** dan blok **if. then. else.**

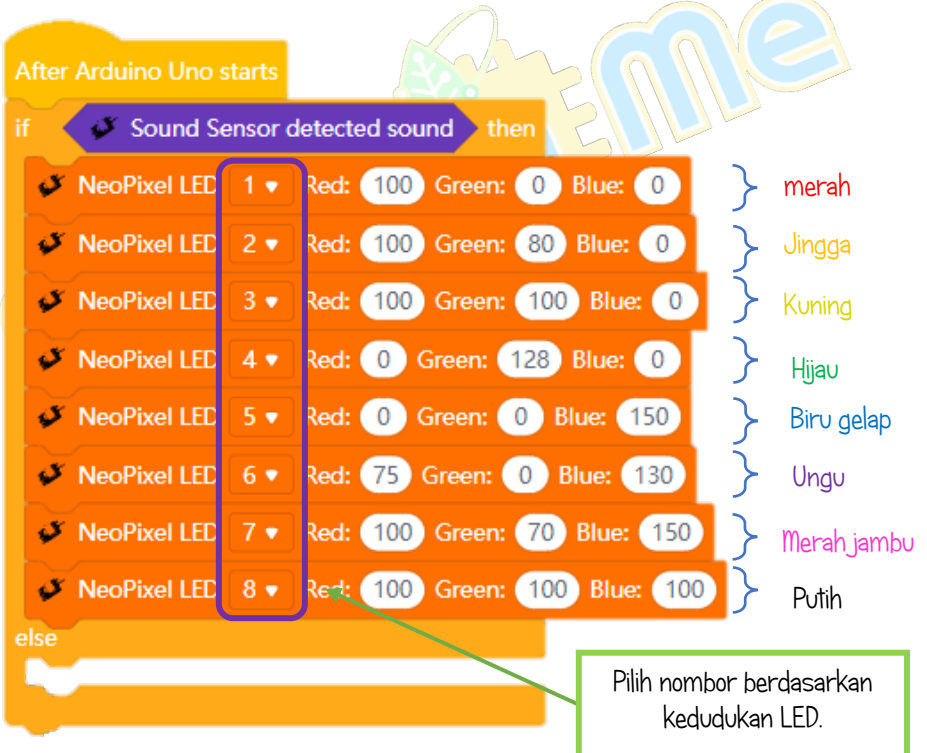


Blok ini akan memeriksa pernyataan di dalam 'if' dan akan melaksanakan blok kenyataan jika pernyataannya adalah benar (true). Namun jika pernyataannya tidak benar (false) maka blok ini akan melaksanakan blok 'else'.

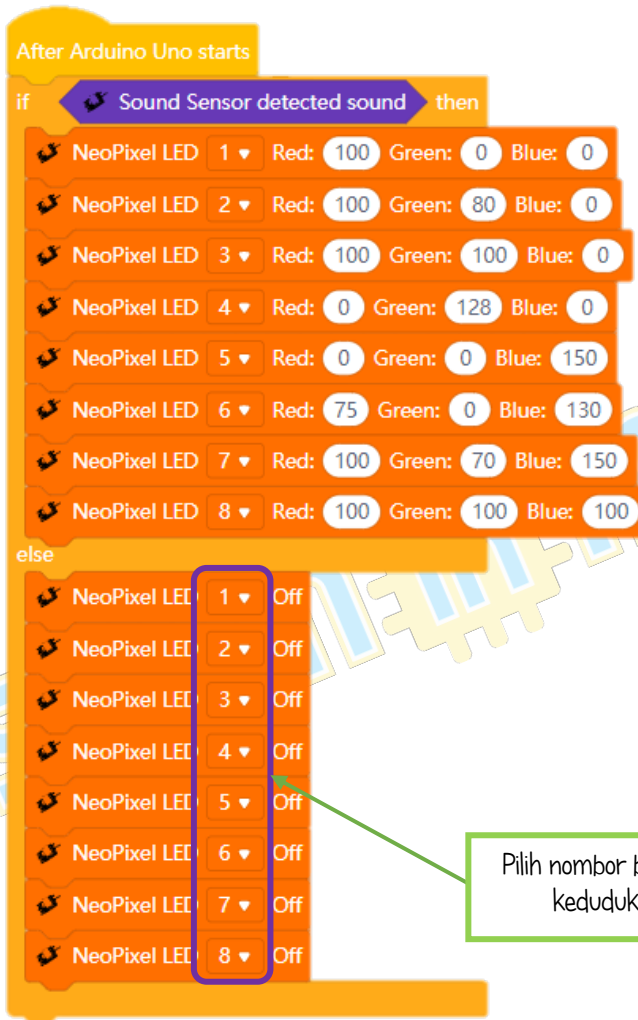
- Masukkan blok **sound sensor detect sound** ke dalam ruang heksagon di dalam blok **if**.
then.



- Tambahkan blok **Neopixel LED** untuk setkan 8 warna ke semua Neopixel LED di bawah blok **if**. **then**.



4. Seterusnya, tambahkan blok **Neopixel LED off** untuk matikan ke semua Neopixel LED.



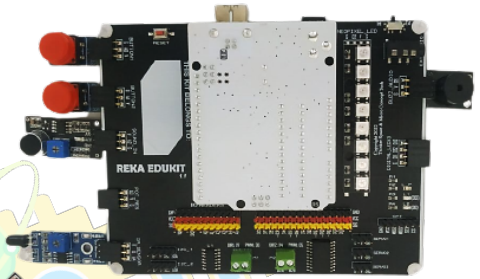
Pilih nombor berdasarkan kedudukan LED.

9: SISTEM SEKURITI

Pernah tahu bagaimana sistem sekuriti dibina? Mari kita lihat projek di bawah. Di dalam sistem ini, jika penderia infrared atau penderia bunyi mengesan sesuatu bunyi ataupun objek, pembaz akan berbunyi dan LED pelbagai warna akan menyala satu persatu untuk memberi amaran bahawa terdapat ancaman bahaya. Untuk menghentikan pembaz dan LED pelbagai warna, kita perlu menekan kedua-dua butang tekan dalam waktu yang sama.

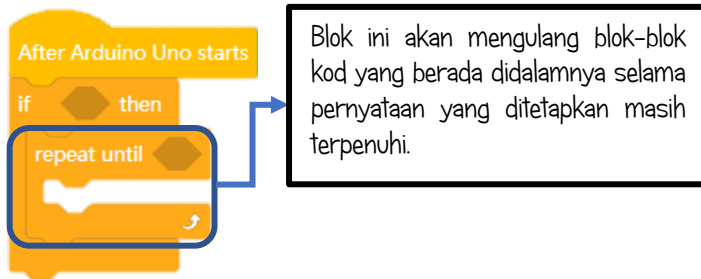
PERANTI YANG DIGUNAKAN :

1. Papan litar RekaEdukit
2. Butang tekan 
3. Penderia infrared 
4. Penderia bunyi 
5. LED pelbagai warna (Neopixel LED) 
6. Pembaz 

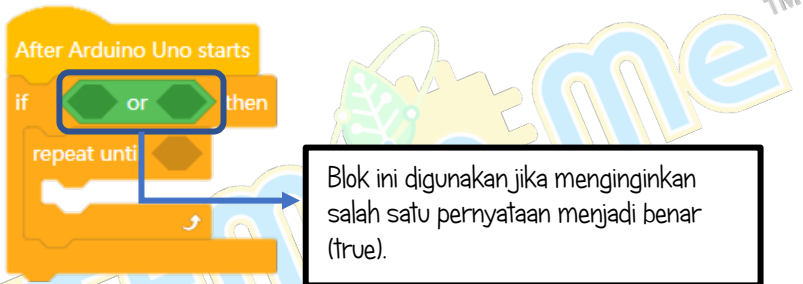


TUTORIAL

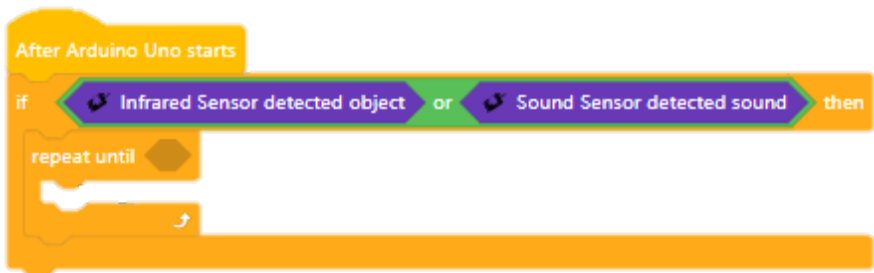
1. Seretkan *After Arduino Uno starts* berserta blok *if. then* dan blok *repeat until*.



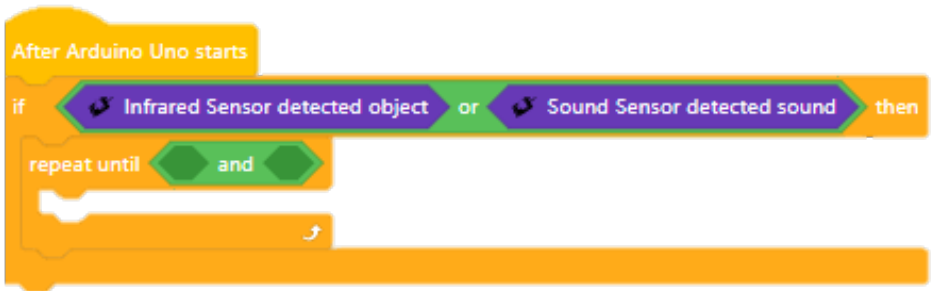
2. Masukkan blok *or* ke dalam ruang heksagon di dalam blok *if. then*.



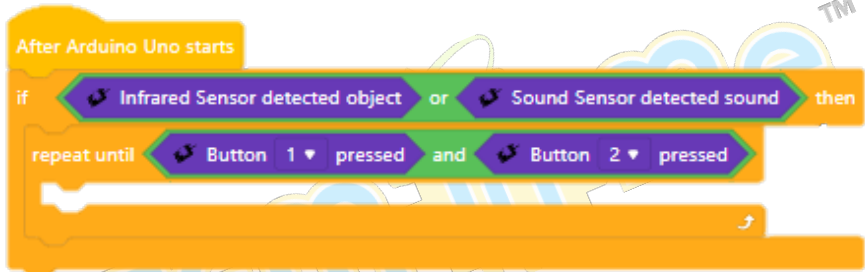
3. Kemudian, masukkan blok *Infrared sensor detected object* dan blok *sound sensor detected sound* ke dalam *or* blok.



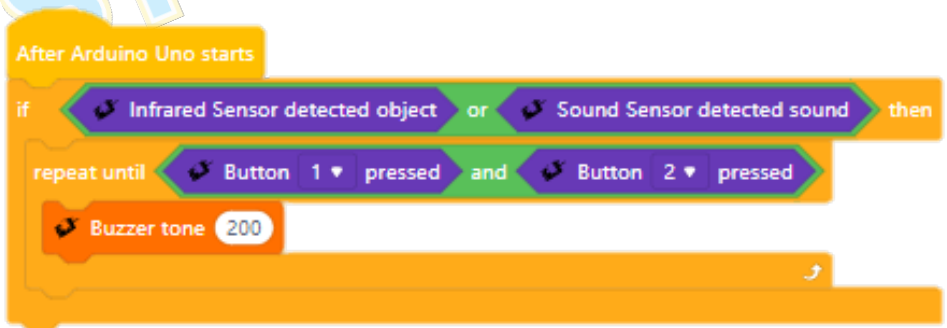
4. Tambahkan blok **and** ke dalam ruang heksagon di dalam blok **repeat until**.



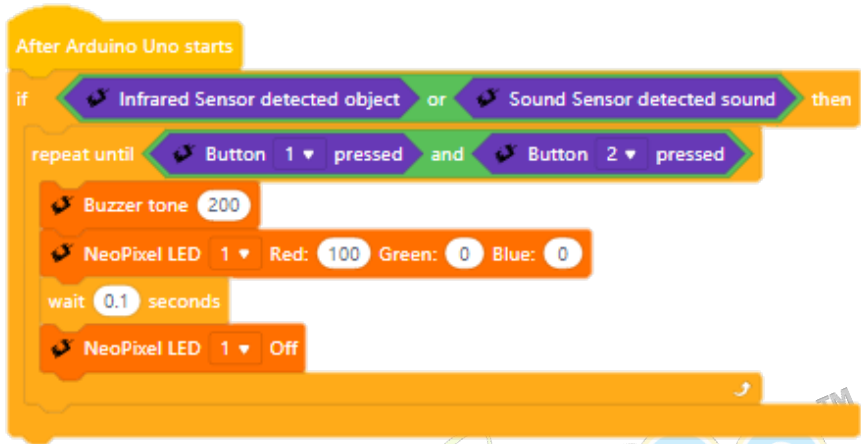
5. Seterusnya, tambahkan blok **button 1 pressed** dan **button 2 pressed** ke dalam blok **and**.



6. Tambahkan blok **buzzer tone** di bawah blok **repeat until**.

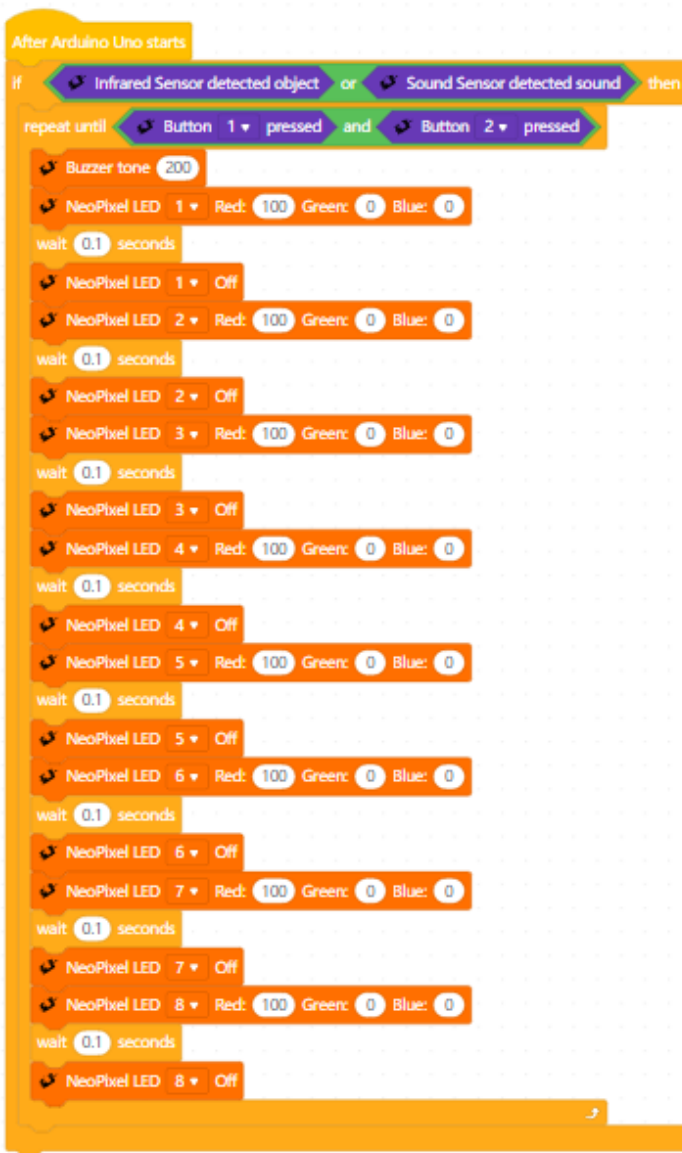


7. Tambahkan blok Neopixel LED diikuti oleh blok delay dan blok Neopixel LED off.



STEM in me

8. Ulang Langkah 7 untuk setiap 8 Neopixel LED.

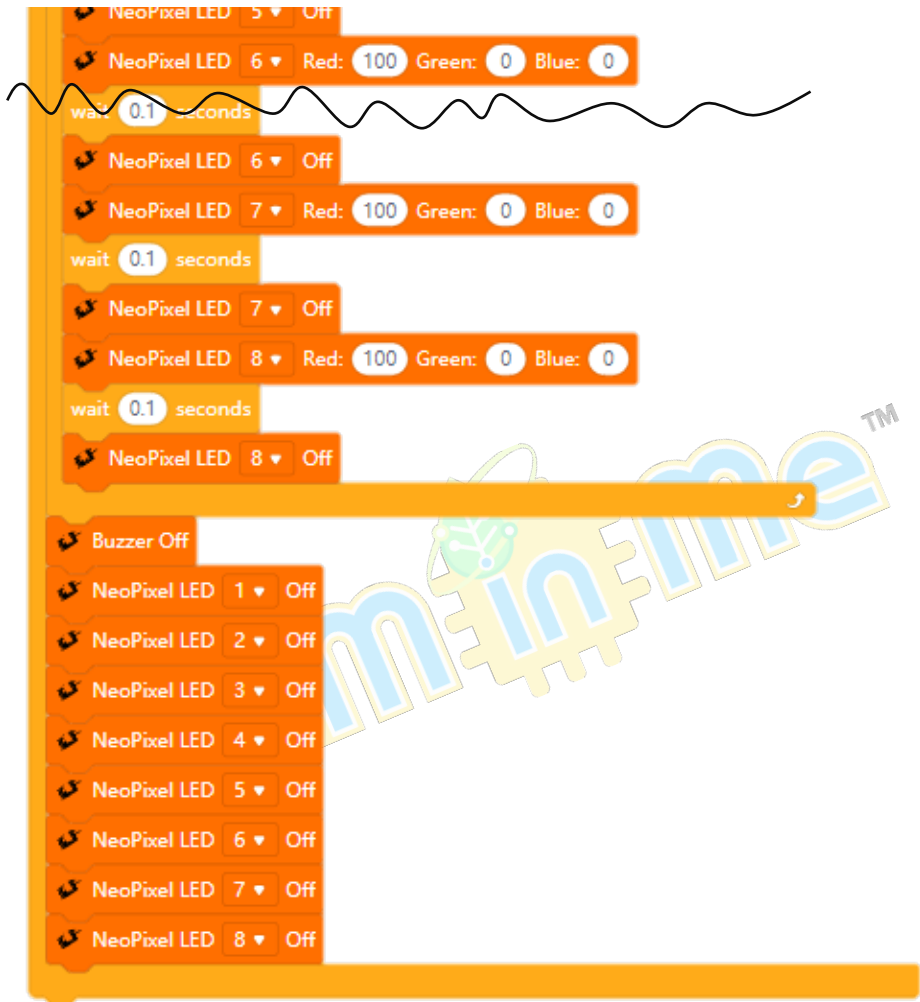


9. Kemudian, masukkan blok **buzzer off** dan blok **Neopixel LED off** di dalam blok **if, then**.



10. Tambahkan blok **Neopixel LED off** untuk setiap 8 neopixel LED.





10: SISTEM BILIK DARJAH

Jom bina sistem bilik darjah di dalam kelas! Sistem kelas bermula apabila para pelajar memasuki kelas. Penderia infrared akan menyalakan LED satu warna (GYR LED) berwarna hijau. Jika pelajar tidak hadir sekolah, kita perlu menekan butang 2 untuk menyalakan lampu kuning pada GYR LED. Apabila penderia bunyi mengesan bunyi pelajar yang riuh di dalam kelas, pembaz akan berbunyi menandakan pelajar perlu senyap. Untuk menghentikan bunyi pembaz, kita perlu menekan butang tekan 1. Seterusnya, kita boleh menyalakan lampu di kelas mengikut kecerahan yang kita kehendaki menggunakan LED pelbagai warna (Neopixel LED) dan meter upaya sebagai suis boleh laras.

PERANTI YANG DIGUNAKAN :

1. Papan litar RekaEdukit

2. Butang tekan 

3. Penderia bunyi 

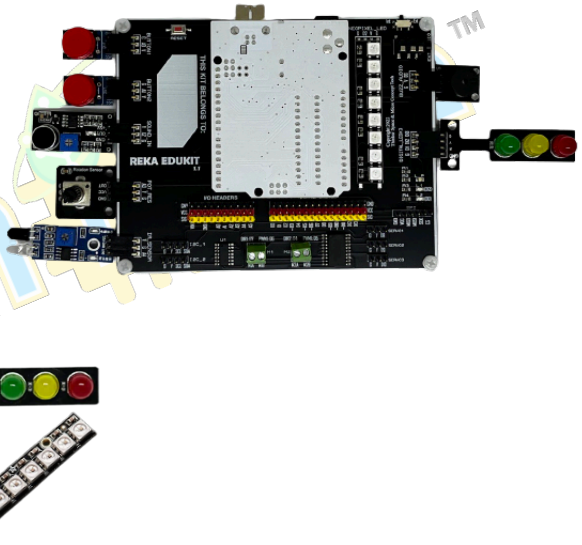
4. Penderia infrared 

5. Meter upaya 

6. LED satu warna (GYR LED) 

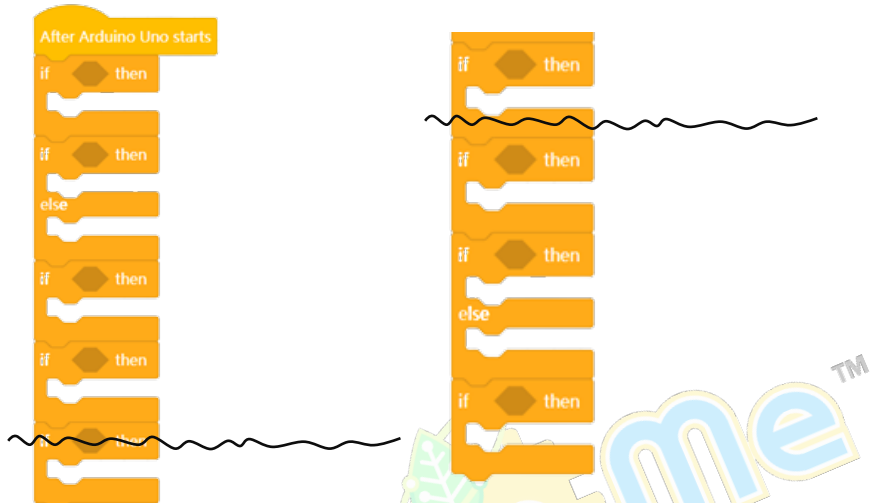
7. LED pelbagai warna (Neopixel LED) 

8. Pembaz 

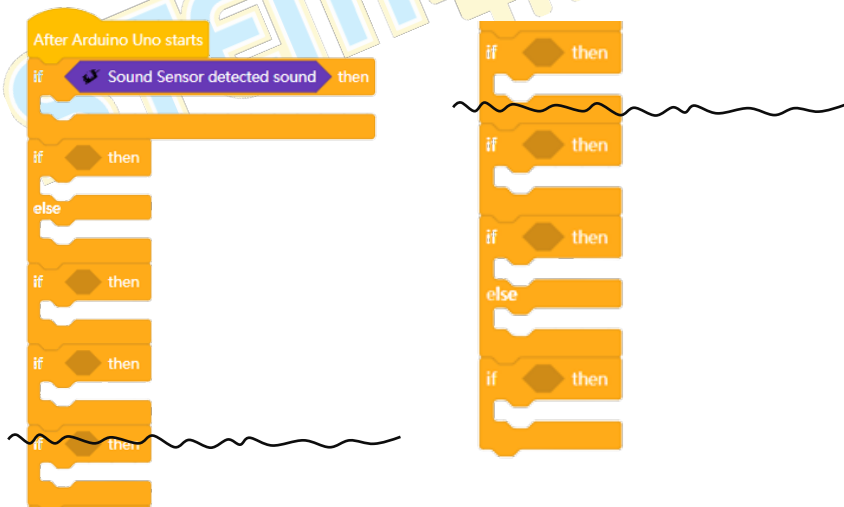


TUTORIAL

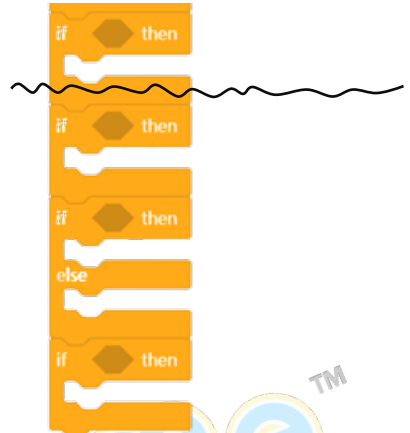
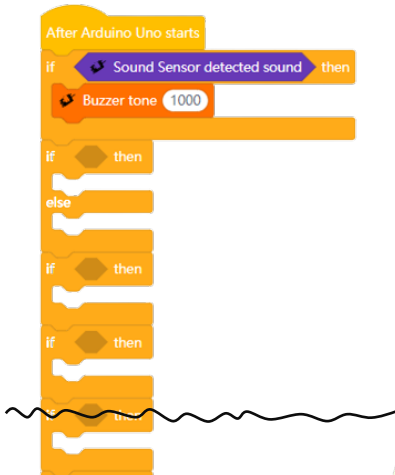
1. Seretkan **After Arduino Uno starts** berserta blok **if. then** dan blok **if. then. else**.



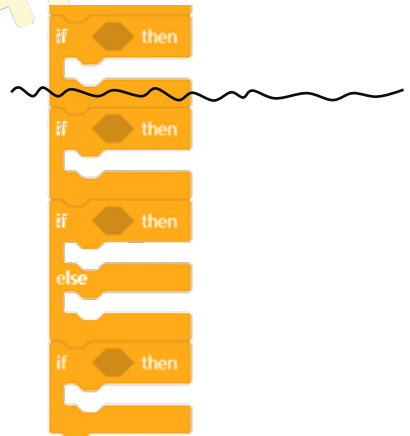
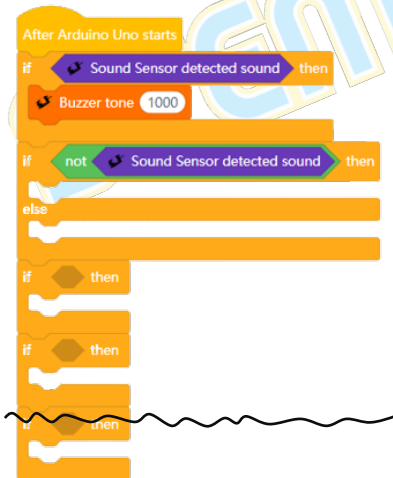
2. Masukkan **Sound Sensor detected sound** ke dalam ruang heksagon di dalam blok **if. then**.



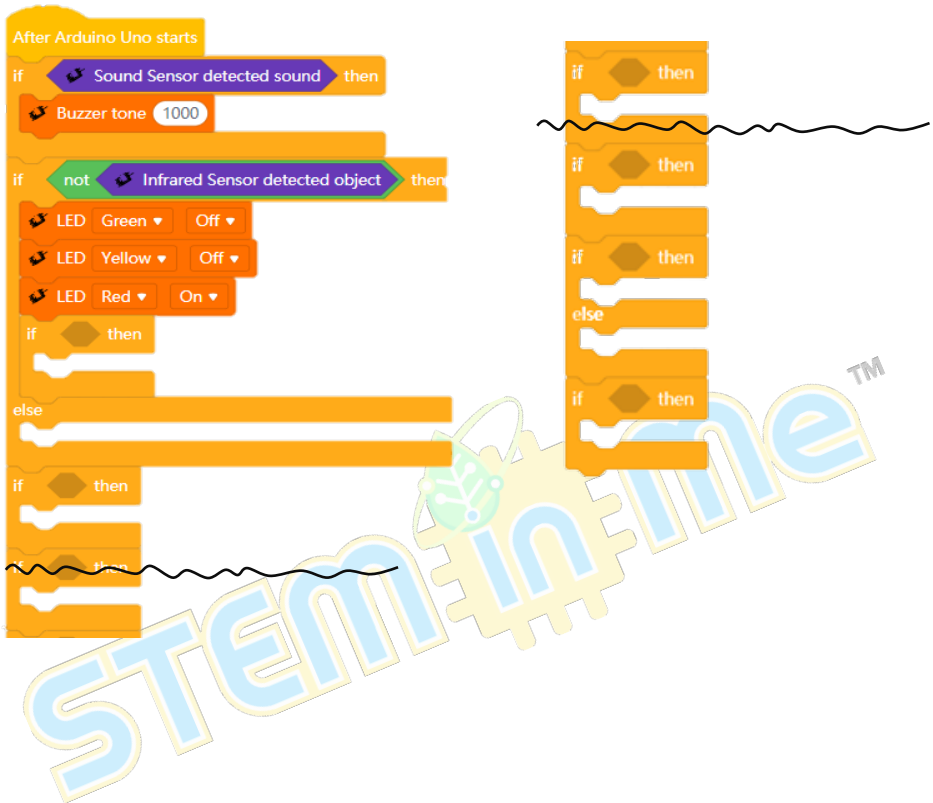
3. Masukkan blok **Buzzer tone** ke dalam blok **if. then**.



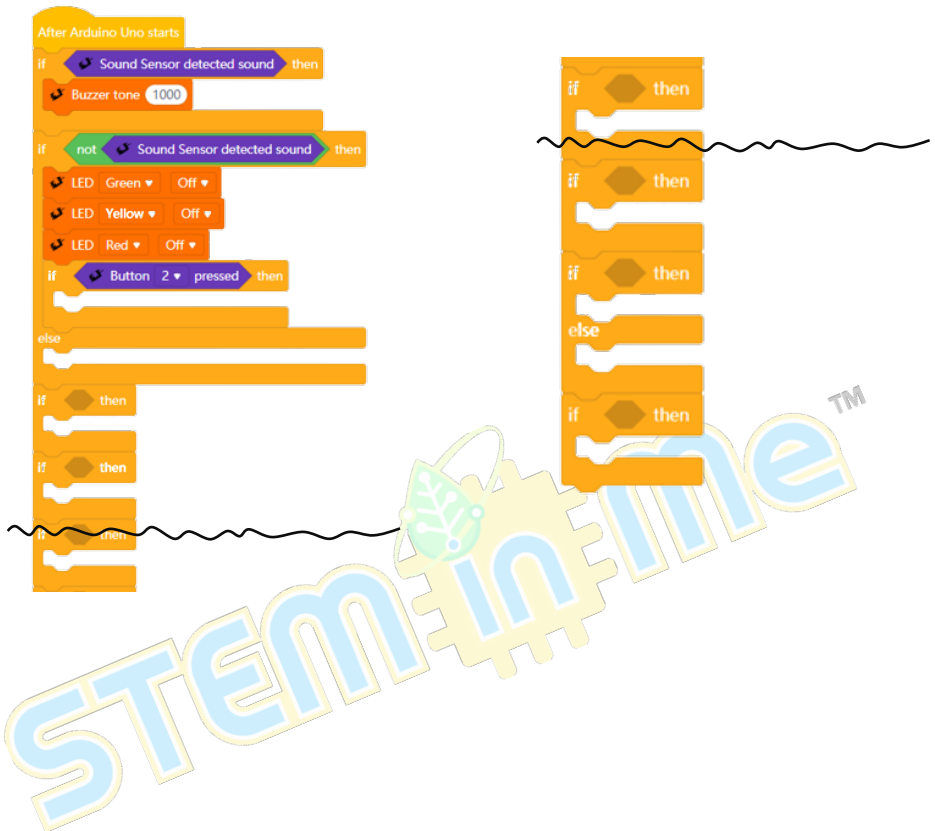
4. Kemudian, masukkan blok **not** ke dalam ruang heksagon di dalam blok **if. then** serta masukkan blok **Infrared Sensor detected object** ke dalam blok **not**.



5. Masukkan blok **LED** untuk setiap warna Hijau, Kuning dan Merah serta blok **if**, **then** ke dalam blok **if**, **then**.



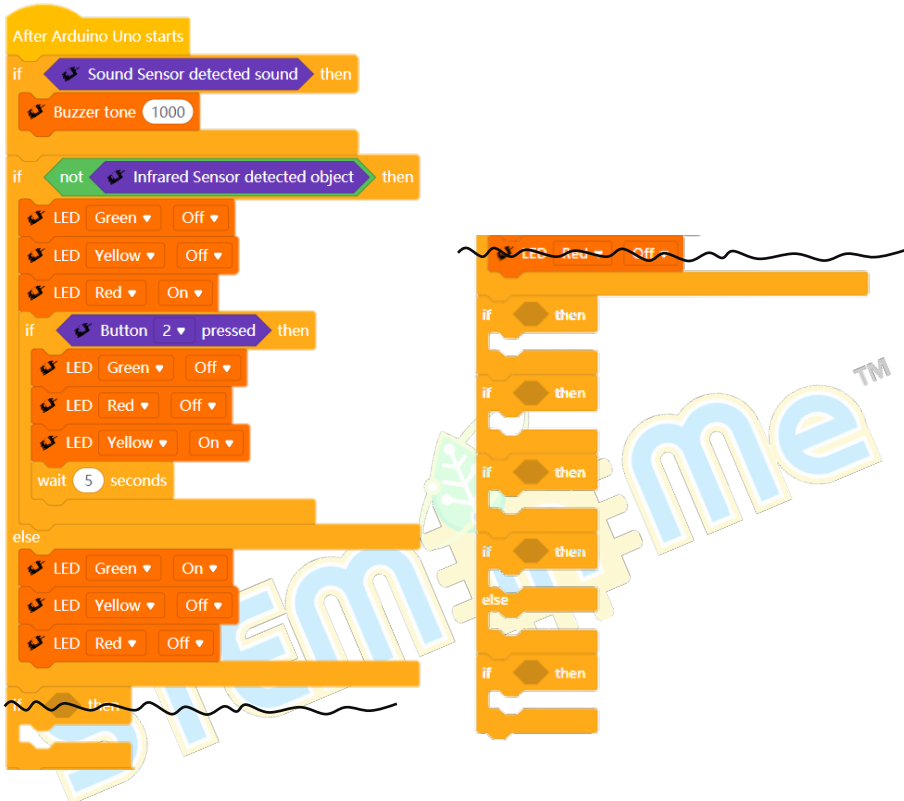
6. Masukkan blok **Button pressed** ke dalam ruang heksagon di dalam blok **if. then**.



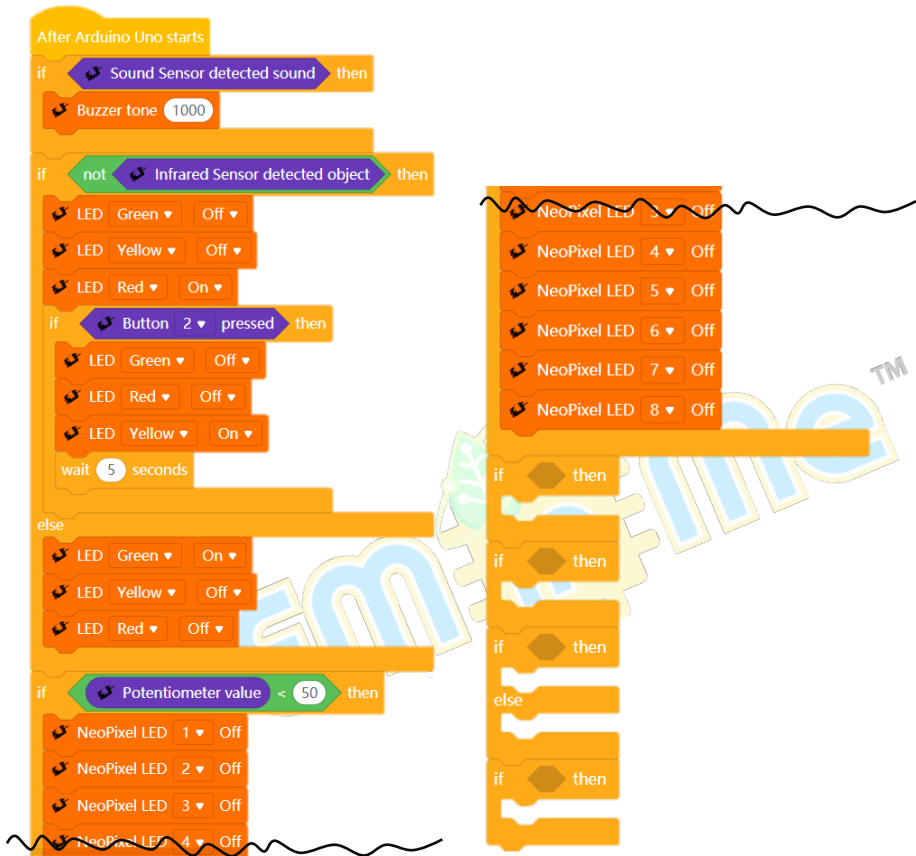
7. Tambahkan blok LED untuk setiap warna Hijau, Kuning dan Merah serta blok delay ke dalam blok if. then.



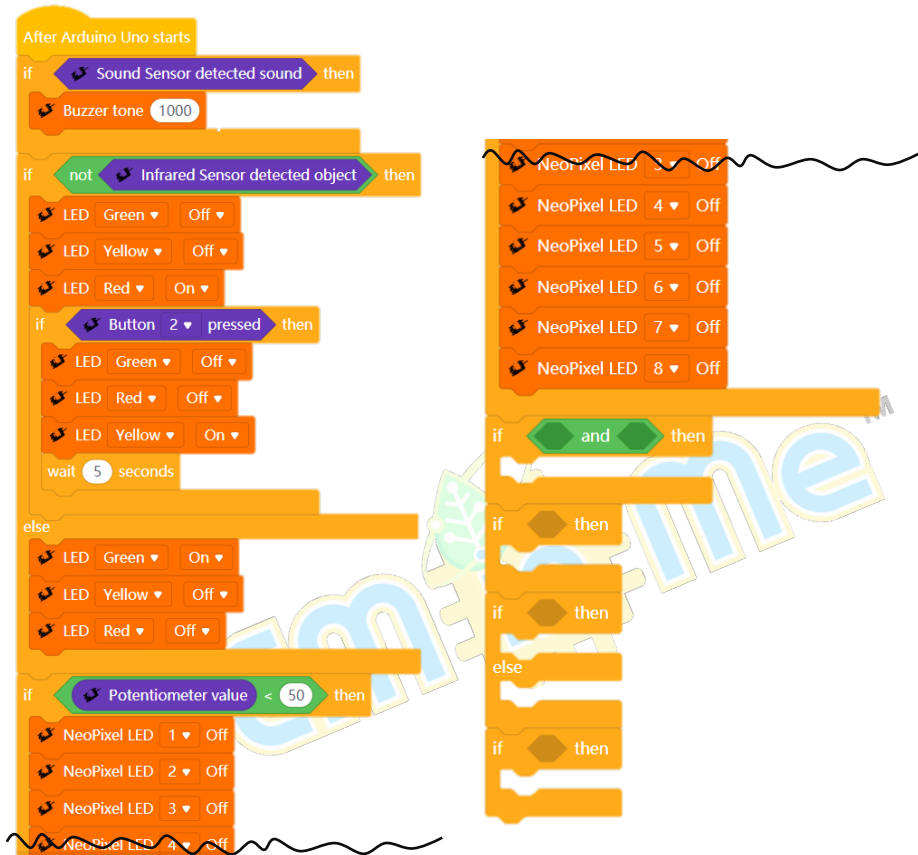
8. Masukkan blok LED untuk setiap warna Hijau, Kuning dan Merah ke dalam blok else.



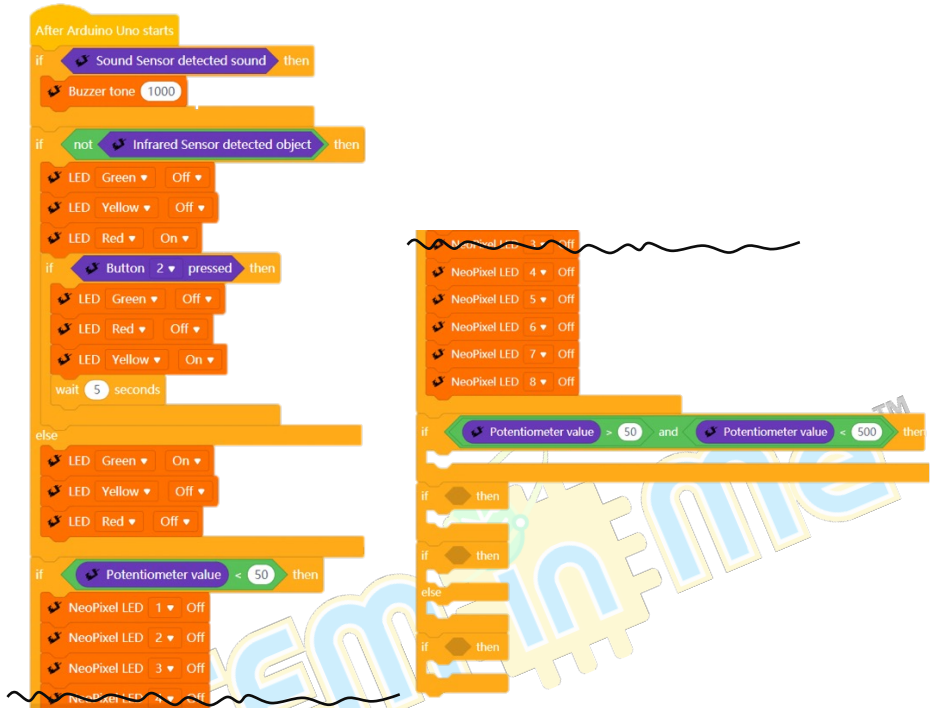
9. Masukkan blok **less than** ke dalam ruang heksagon di dalam blok **if. then** serta masukkan blok **potentiometer value** ke dalam blok **less than**.



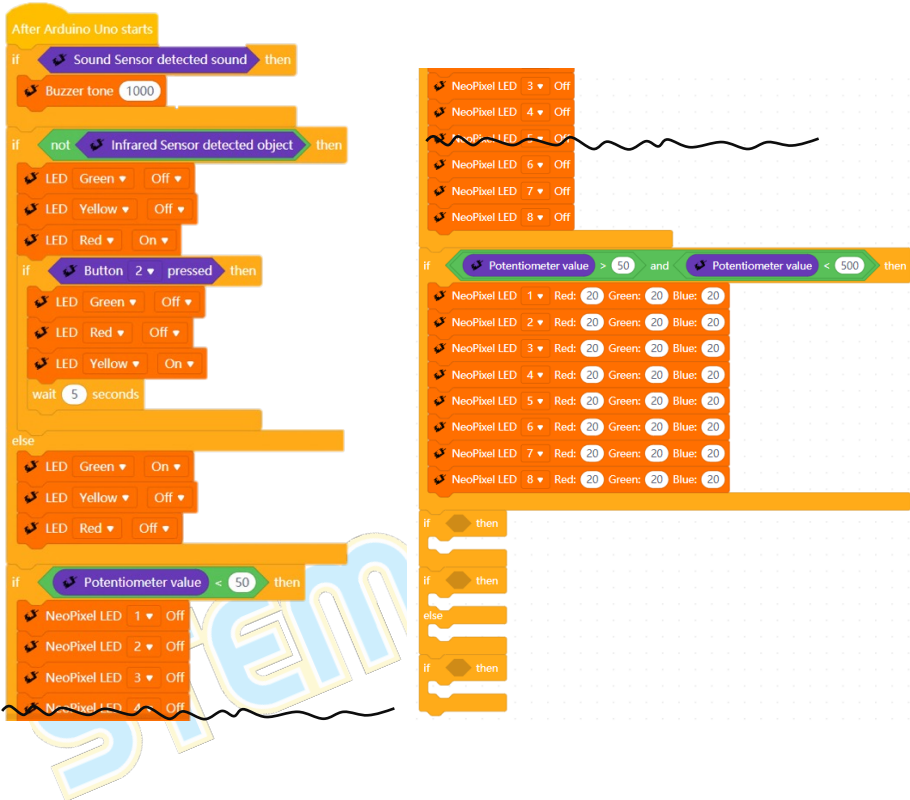
10. Masukkan 8 blok **Neopixel LED off** ke dalam blok **if. then**.



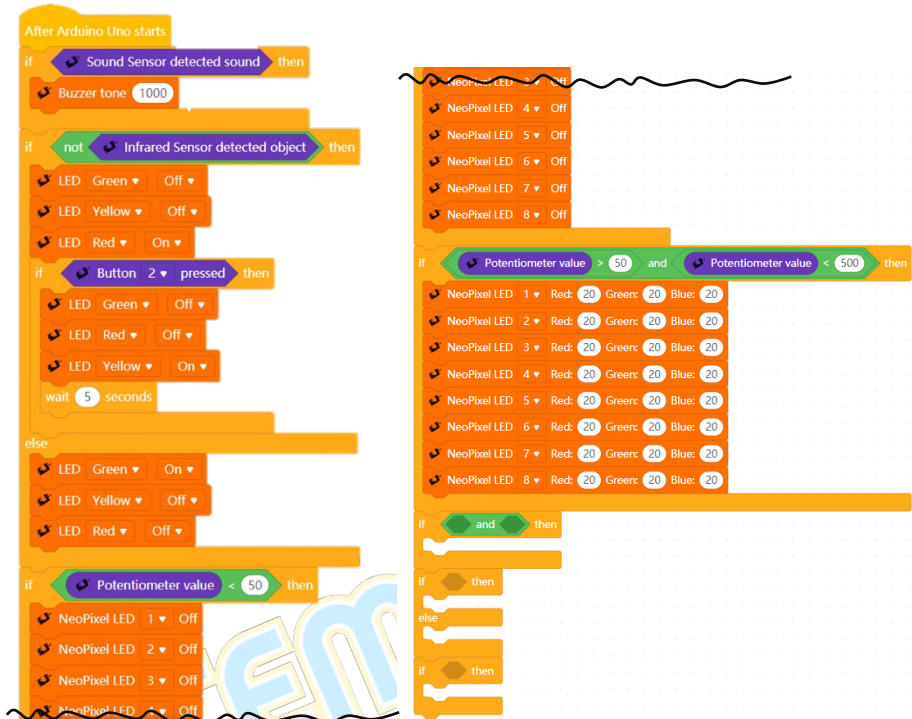
11. Masukkan blok **and** ke dalam ruang heksagon di dalam blok **if. then**.



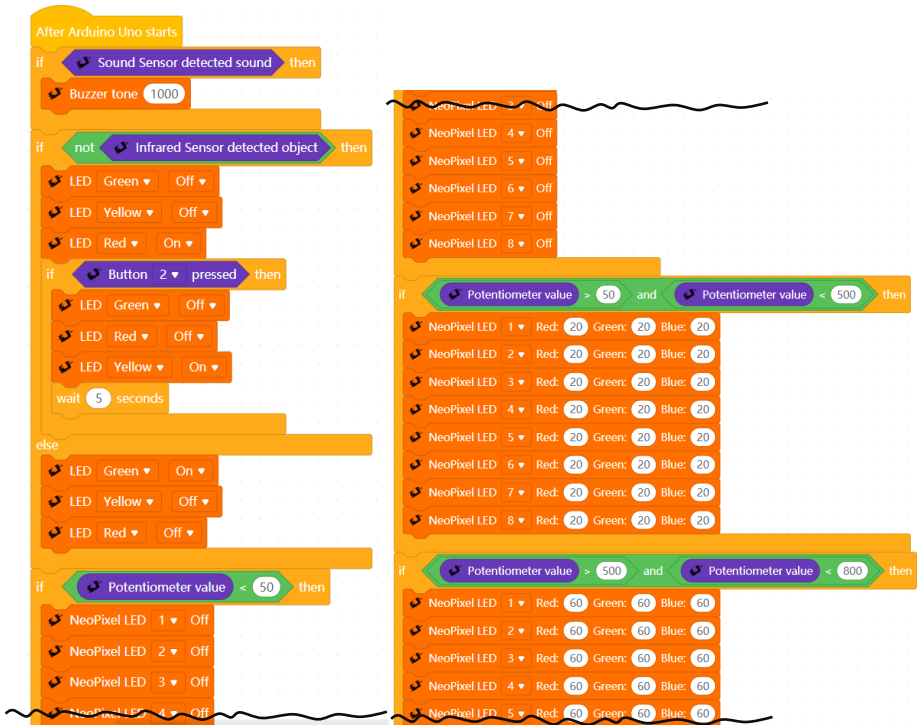
12. Masukkan blok **greater than** dan **less than** ke dalam blok **and** serta masukkan blok **potentiometer value** ke dalam blok **greater than** dan **less than**.



13. Kemudian, masukkan 8 blok **NeoPixel LED** ke dalam blok **if, then**.



14. Ulang langkah ini sehingga blok else.

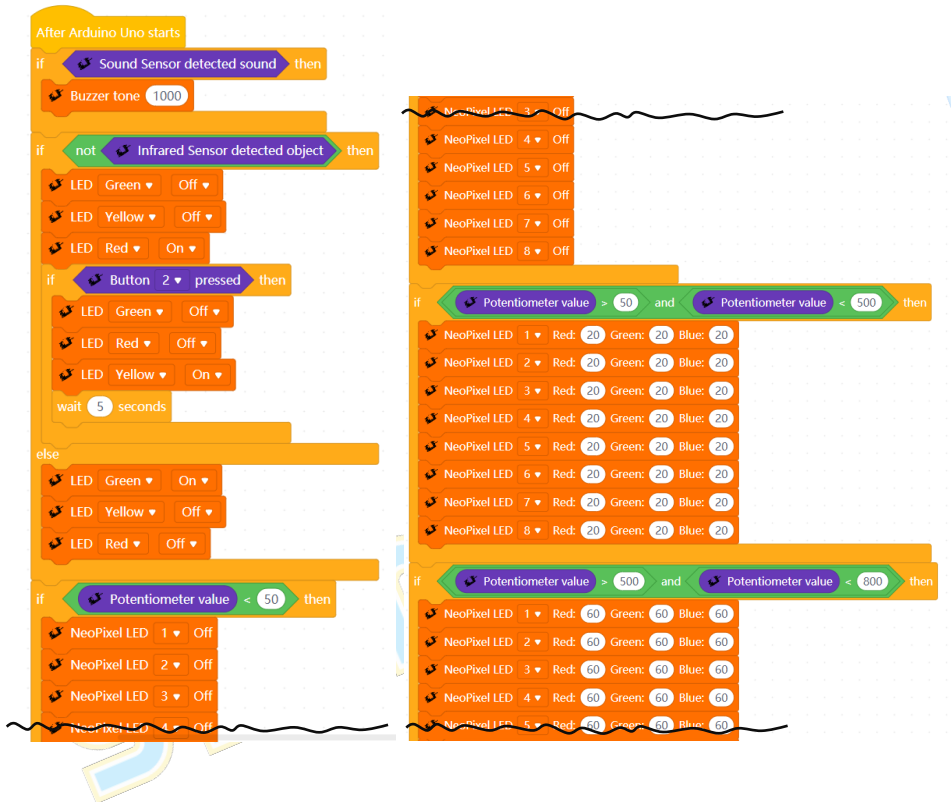


The script is a Scratch code block for controlling 8 NeoPixel LEDs based on a potentiometer's value. It starts with a 'when green flag clicked' event. The main logic is an 'if-then-else' statement. The 'if' condition is 'Potentiometer value > 800 and Potentiometer value < 1000'. If true, it sets all 8 NeoPixel LEDs to Red: 60, Green: 60, and Blue: 60. If false, it sets all 8 NeoPixel LEDs to Red: 100, Green: 100, and Blue: 100. The script ends with a 'stop all sounds' block.

```

when green flag clicked
if Potentiometer value > 800 and Potentiometer value < 1000 then
  NeoPixel LED 1 Red: 60 Green: 60 Blue: 60
  NeoPixel LED 2 Red: 60 Green: 60 Blue: 60
  NeoPixel LED 3 Red: 60 Green: 60 Blue: 60
  NeoPixel LED 4 Red: 60 Green: 60 Blue: 60
  NeoPixel LED 5 Red: 60 Green: 60 Blue: 60
  NeoPixel LED 6 Red: 60 Green: 60 Blue: 60
  NeoPixel LED 7 Red: 60 Green: 60 Blue: 60
  NeoPixel LED 8 Red: 60 Green: 60 Blue: 60
else
  NeoPixel LED 1 Red: 100 Green: 100 Blue: 100
  NeoPixel LED 2 Red: 100 Green: 100 Blue: 100
  NeoPixel LED 3 Red: 100 Green: 100 Blue: 100
  NeoPixel LED 4 Red: 100 Green: 100 Blue: 100
  NeoPixel LED 5 Red: 100 Green: 100 Blue: 100
  NeoPixel LED 6 Red: 100 Green: 100 Blue: 100
  NeoPixel LED 7 Red: 100 Green: 100 Blue: 100
  NeoPixel LED 8 Red: 100 Green: 100 Blue: 100
stop all sounds
  
```


15. Masukkan blok **button 1 pressed** ke dalam ruang heksagon di dalam blok **if**.
then. Kemudian, masukkan blok **Buzzer Off** ke dalam blok **if**. **then**.



```

when green flag clicked
  NeoPixel LED 4 ▾ Red: 60 Green: 60 Blue: 60
  NeoPixel LED 5 ▾ Red: 60 Green: 60 Blue: 60
  NeoPixel LED 6 ▾ Red: 60 Green: 60 Blue: 60
  NeoPixel LED 7 ▾ Red: 60 Green: 60 Blue: 60
  NeoPixel LED 8 ▾ Red: 60 Green: 60 Blue: 60

  if Potentiometer value > 800 and Potentiometer value < 1000 then
    NeoPixel LED 1 ▾ Red: 80 Green: 80 Blue: 80
    NeoPixel LED 2 ▾ Red: 80 Green: 80 Blue: 80
    NeoPixel LED 3 ▾ Red: 80 Green: 80 Blue: 80
    NeoPixel LED 4 ▾ Red: 80 Green: 80 Blue: 80
    NeoPixel LED 5 ▾ Red: 80 Green: 80 Blue: 80
    NeoPixel LED 6 ▾ Red: 80 Green: 80 Blue: 80
    NeoPixel LED 7 ▾ Red: 80 Green: 80 Blue: 80
    NeoPixel LED 8 ▾ Red: 80 Green: 80 Blue: 80
  else
    NeoPixel LED 1 ▾ Red: 100 Green: 100 Blue: 100
    NeoPixel LED 2 ▾ Red: 100 Green: 100 Blue: 100
    NeoPixel LED 3 ▾ Red: 100 Green: 100 Blue: 100
    NeoPixel LED 4 ▾ Red: 100 Green: 100 Blue: 100
    NeoPixel LED 5 ▾ Red: 100 Green: 100 Blue: 100
    NeoPixel LED 6 ▾ Red: 100 Green: 100 Blue: 100
    NeoPixel LED 7 ▾ Red: 100 Green: 100 Blue: 100
    NeoPixel LED 8 ▾ Red: 100 Green: 100 Blue: 100

  if Button 1 ▾ pressed then
    Buzzer Off
  
```

11: SISTEM PENGHALAU BURUNG

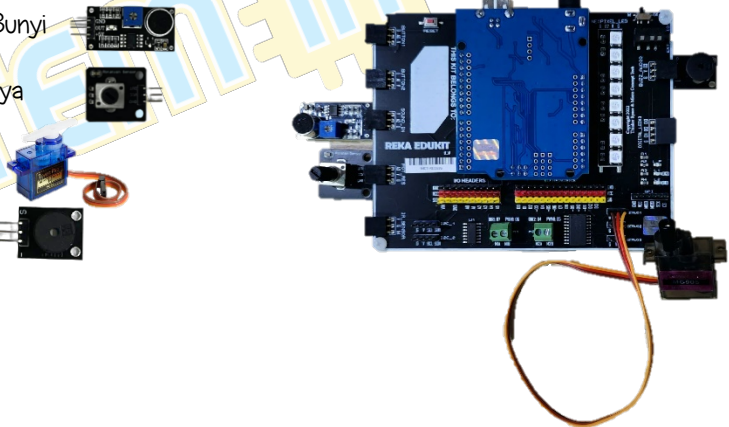
Sistem penghalau burung berfungsi apabila terdapat sebarang bunyi. Dalam sistem ini, jika penerima bunyi mengesan sebarang bunyi dan meter upaya dihidupkan kepada nilai yang ditetapkan. Pembaz akan mengeluarkan bunyi dengan nada yang berbeza sambil menggerakkan servo dari 0 hingga 180 darjah. Walau bagaimanapun, pembaz akan berhenti dan servo akan kembali ke 0 darjah jika tiada bunyi atau jika nilai meter upaya ditetapkan kepada 0.

PENGENALAN SERVO

Servo berfungsi menukar isyarat keluar dari mikropengawal kepada mekanikal dengan menukar posisi mengikut posisi sudut yang ditetapkan.

PERANTI YANG DIGUNAKAN :

1. Penderia Bunyi
2. Meter Upaya
3. Servo
4. Pembaz



TUTORIAL

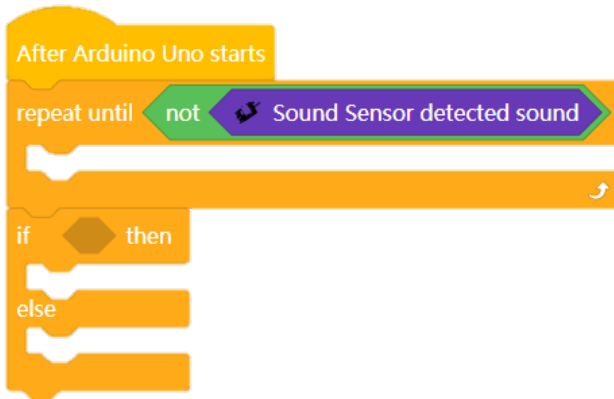
1. Seretkan **After Arduino Uno starts** berserta blok **repeat until** dan blok **if. then. else.**



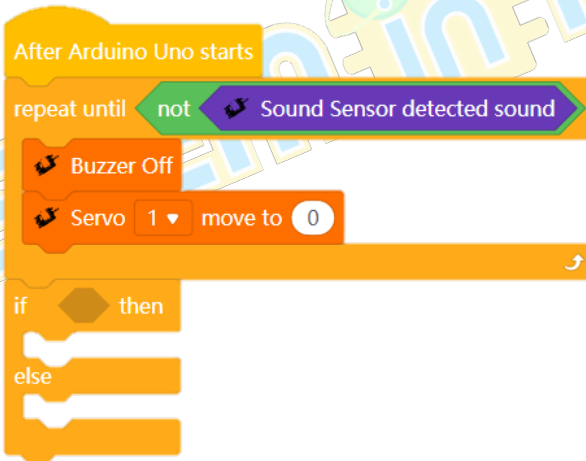
2. Masukkan blok **not** ke dalam ruang heksagon blok **repeat until**.



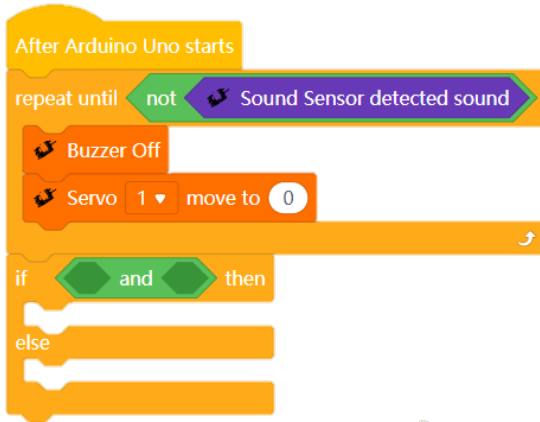
3. Kemudian, masukkan blok **sound sensor detected sound** kedalam blok **not**



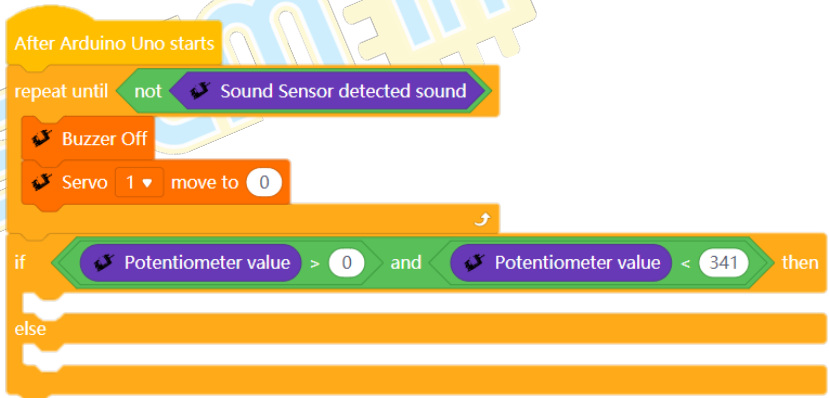
4. Tambahkan blok **buzzer off** berserta blok **servo** dibawah blok **repeat until**.



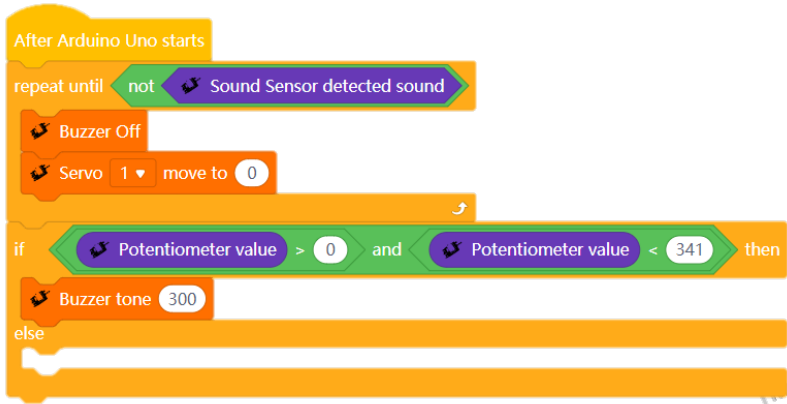
5. Masukkan blok **and** kedalam ruang heksagon blok **if, then, else**.



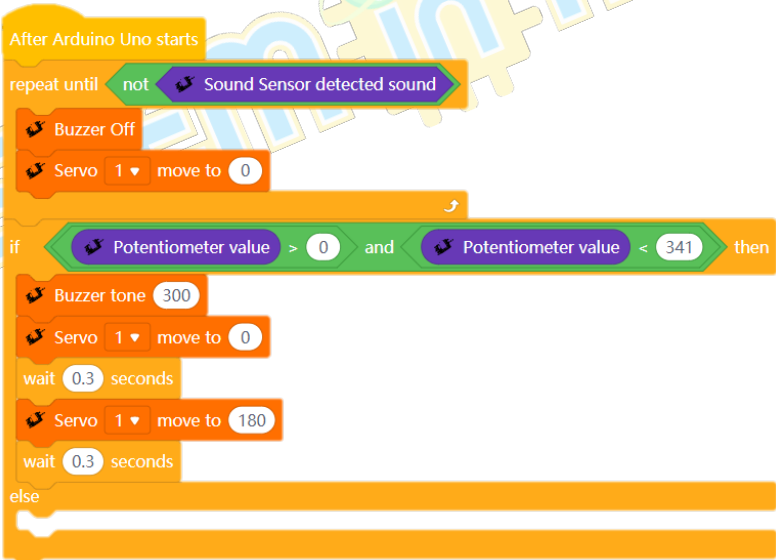
6. Masukkan blok **greater than** dan **less than** ke dalam blok **and** dan masukkan blok **potentiometer value** ke dalam **greater than** dan **less than**.



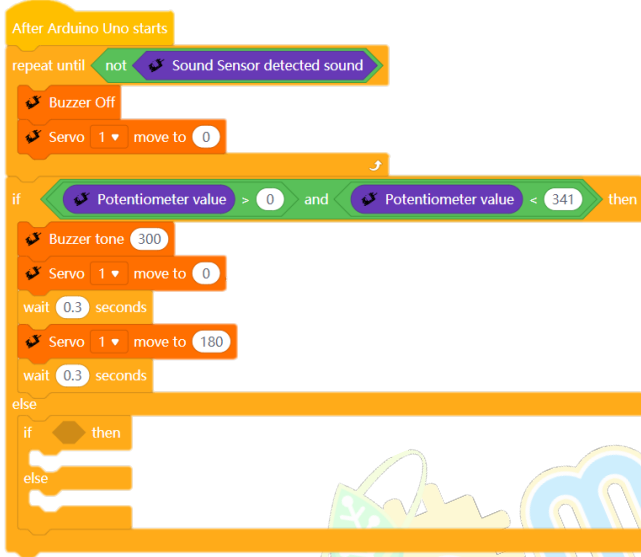
7. Tambahkan blok **buzzer tone** dibawah blok **if. then**.



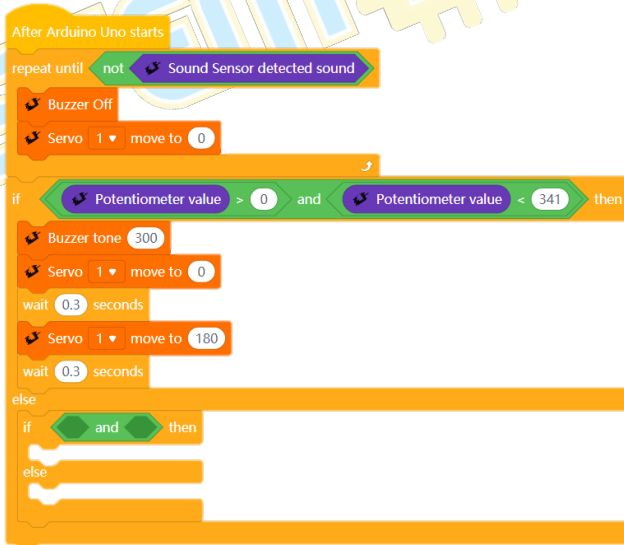
8. Tambah blok **servo** berserta blok **delay** dan blok **servo** lagi dan **delay**.



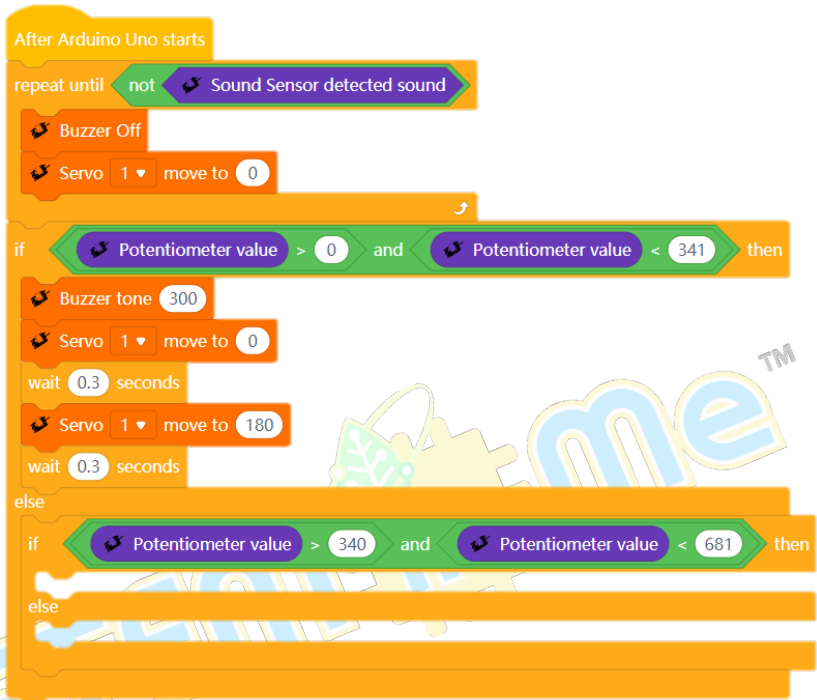
9. Tambahkan blok **if, then, else** dibawah blok **else**.



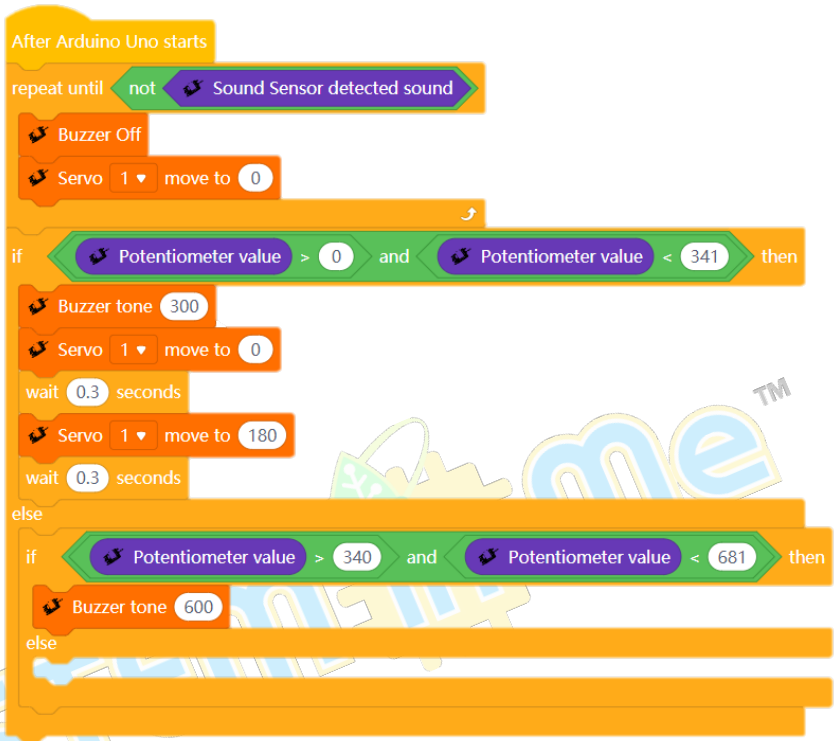
10. Masukkan blok **and** ke dalam ruang heksagon blok **if, then**.



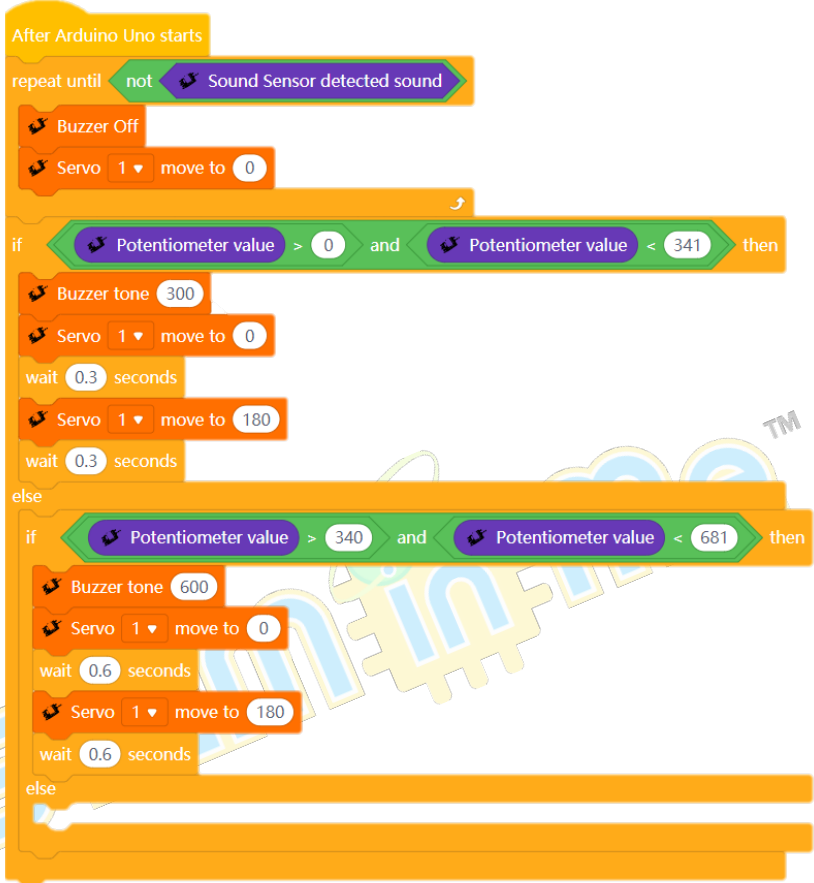
11. Masukkan blok **greater than** dan **less than** ke dalam **and** dan masukkan blok **potentiometer value** ke dalam **greater than** dan **less than**.



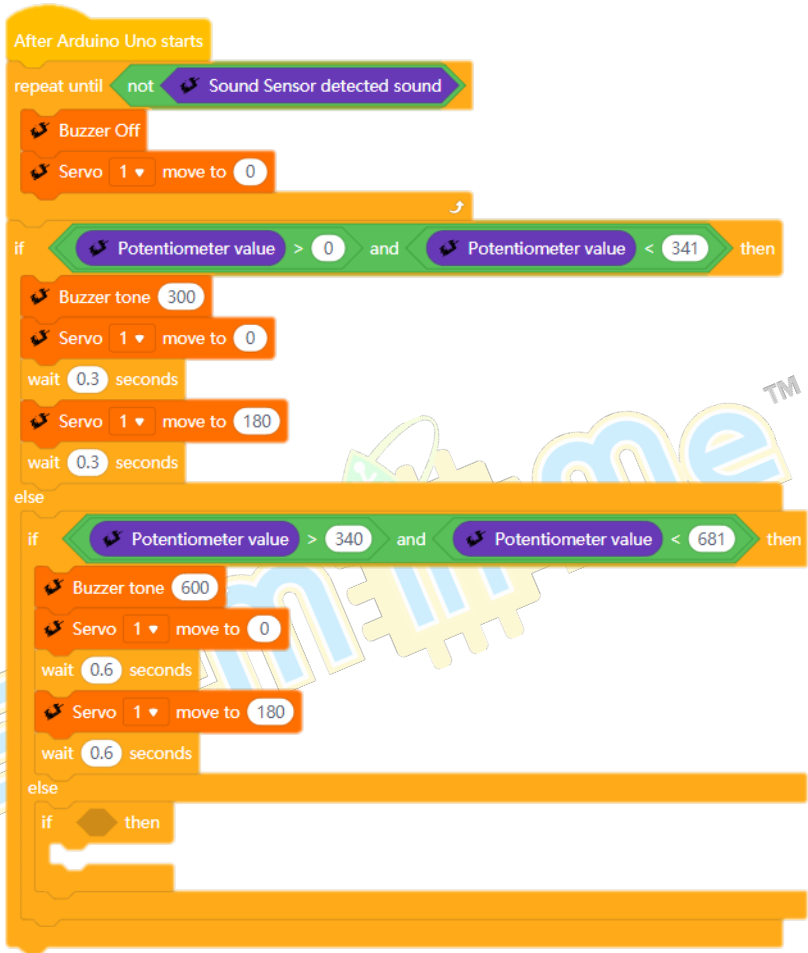
12. Tambahkan blok **buzzer tone** dibawah blok **if. then**.



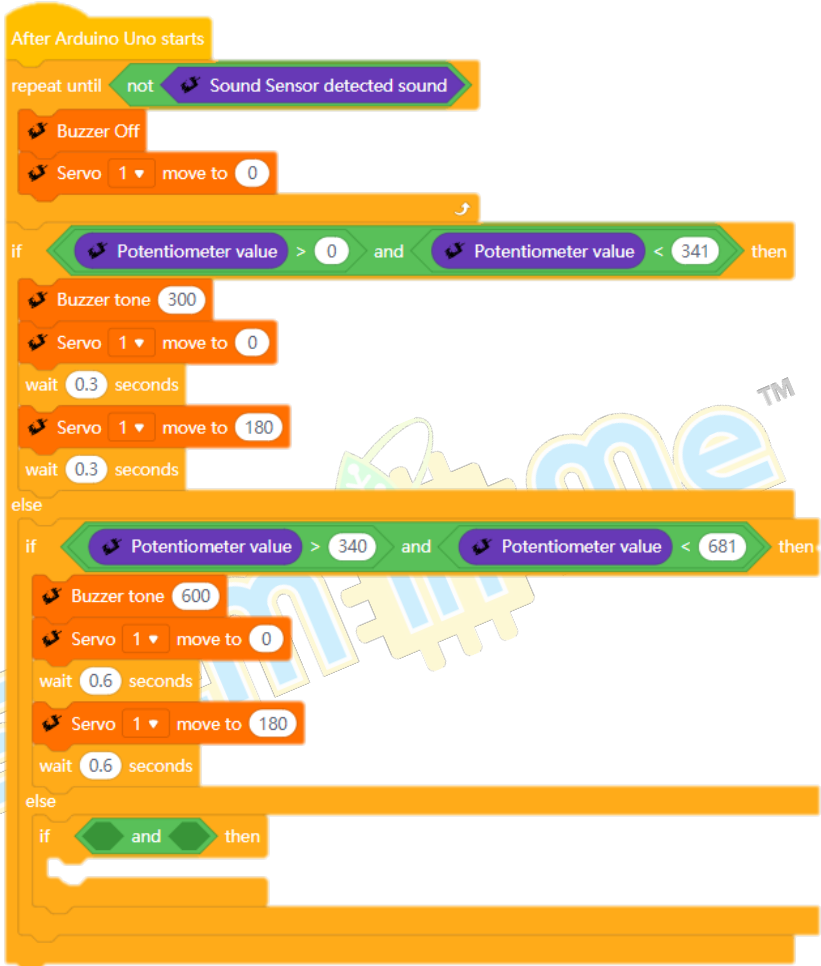
13. Tambah blok **servo** beserta blok **delay** dan blok **servo** lagi dan **delay**.



14. Tambahkan blok **if, then** dibawah blok **else**.



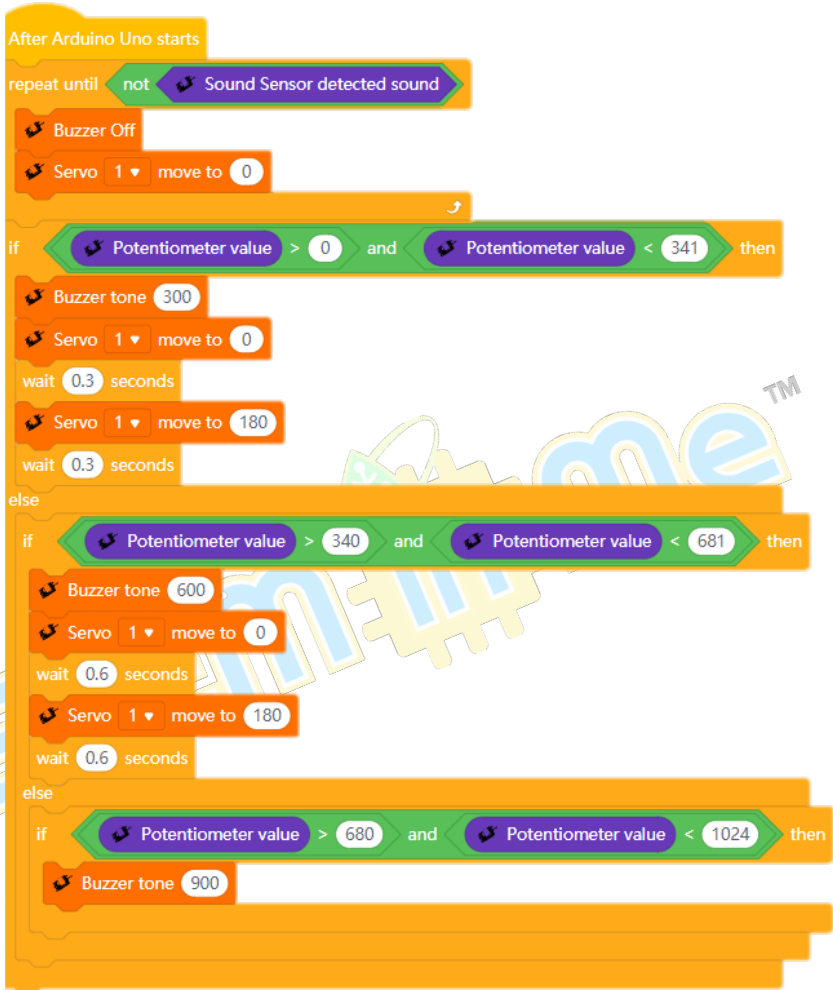
15. Masukkan blok **and** ke dalam ruang heksagon blok **if. then**.



16. Masukkan blok **greater than** dan **less than** ke dalam blok **and** dan masukkan blok **potentiometer value** ke dalam **greater than** dan **less than**.



17. Tambahkan blok **buzzer tone** dibawah blok **if. then**.



18. Tambah blok **servo** berserta blok **delay** dan blok **servo** lagi dan **delay**.



12: SISTEM KESELAMATAN KERETA

Projek kali ini untuk mengaplikasikan penggunaan interaksi dengan peranti lain secara wayarles menggunakan Bluetooth. Lampu kereta (neopixel LED) dan hon (pembaz) dikawal menggunakan aplikasi di telefon pintar ketika kereta tidak bergerak. Ini akan memudahkan pengguna untuk membuka dan menutup lampu serta membunyikan hon tanpa perlu ke kereta. Pengguna boleh mengawal lampu dalam 2 keadaan iaitu lampu rendah (Neopixel LED 1 dan 4) dan lampu tinggi (Neopixel LED 2 dan 3). Lampu rendah dikawal menggunakan potentiometer di dalam aplikasi telefon pintar manakala lampu tinggi dikawal melalui butang tekan. Hon pula dibunyikan menggunakan butang buzzer di dalam aplikasi. Bunyi hon boleh diubah mengikut nada yang diinginkan.

Pengenalan Modul Bluetooth

Modul Bluetooth membolehkan komunikasi tanpa wayar antara peranti. Ia membenarkan pemindahan data pada jarak yang dekat, menghapuskan keperluan untuk kabel fizikal dan lebih fleksibiliti dalam sambungan peranti. Antara jenis modul Bluetooth yang digunakan adalah HC-05 dan AT-09.

Pengenalan Aplikasi Reka Edukit dan Reka Edukit BLE



Figure 2: Aplikasi Reka Edukit



Figure 1: Aplikasi Reka Edukit BLE

Aplikasi Reka Edukit dan Reka Edukit BLE merupakan satu perisian yang boleh digunakan untuk mengawal Reka Edukit secara wayarles dengan menggunakan sambungan Bluetooth. Antara muka dan cara penggunaan untuk kedua – dua aplikasi adalah sama kecuali jenis modul yang sesuai dan boleh disambungkan ke aplikasi.

Aplikasi Reka Edukit hanya boleh disambungkan ke modul Bluetooth HC-05 manakala Reka Edukit BLE hanya boleh disambungkan ke modul Bluetooth AT-09.

PERANTI YANG DIGUNAKAN:

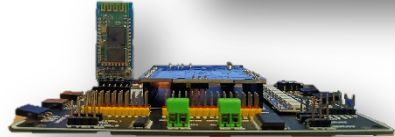
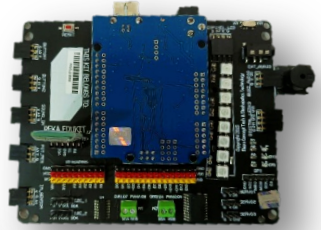
1. Modul Bluetooth



2. Pembaz

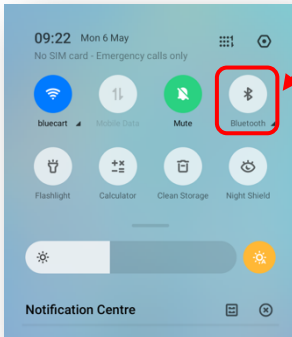


3. Neopixel LED



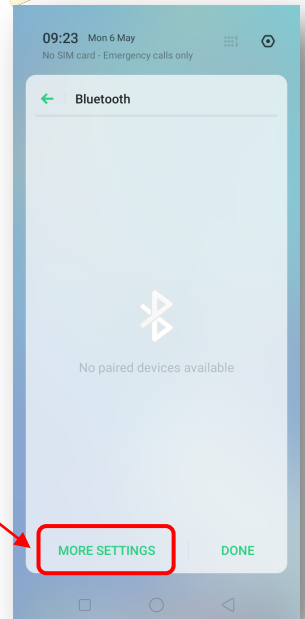
CARA PENYAMBUNGAN MODUL BLUETOOTH DENGAN APLIKASI REKA EDUKIT™

Langkah 1



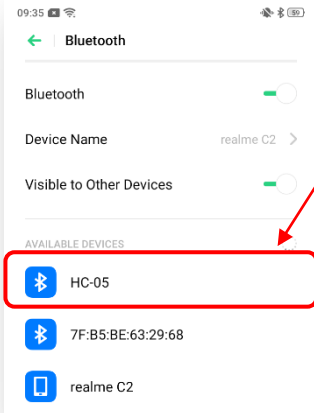
Tekan dan hidupkan menu Bluetooth.

Langkah 2



Tekan pada More Settings.

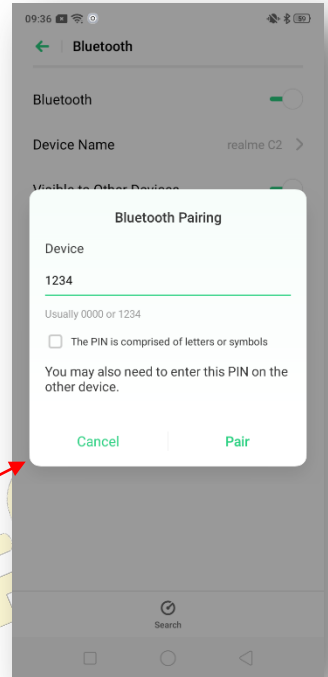
Langkah 3



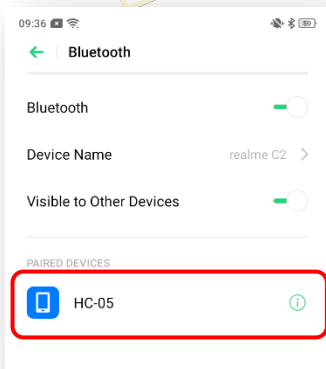
Kemudian cari nama modul Bluetooth pada Available Devices dan tekan pada nama tersebut.

Kemudian masukkan kata laluan '1234'. Untuk setiap modul Bluetooth, kata laluan adalah sama. Kemudian tekan pada butang Pair.

Langkah 4



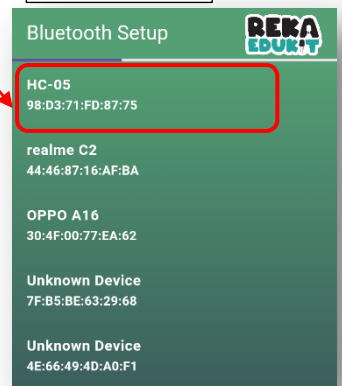
Langkah 5



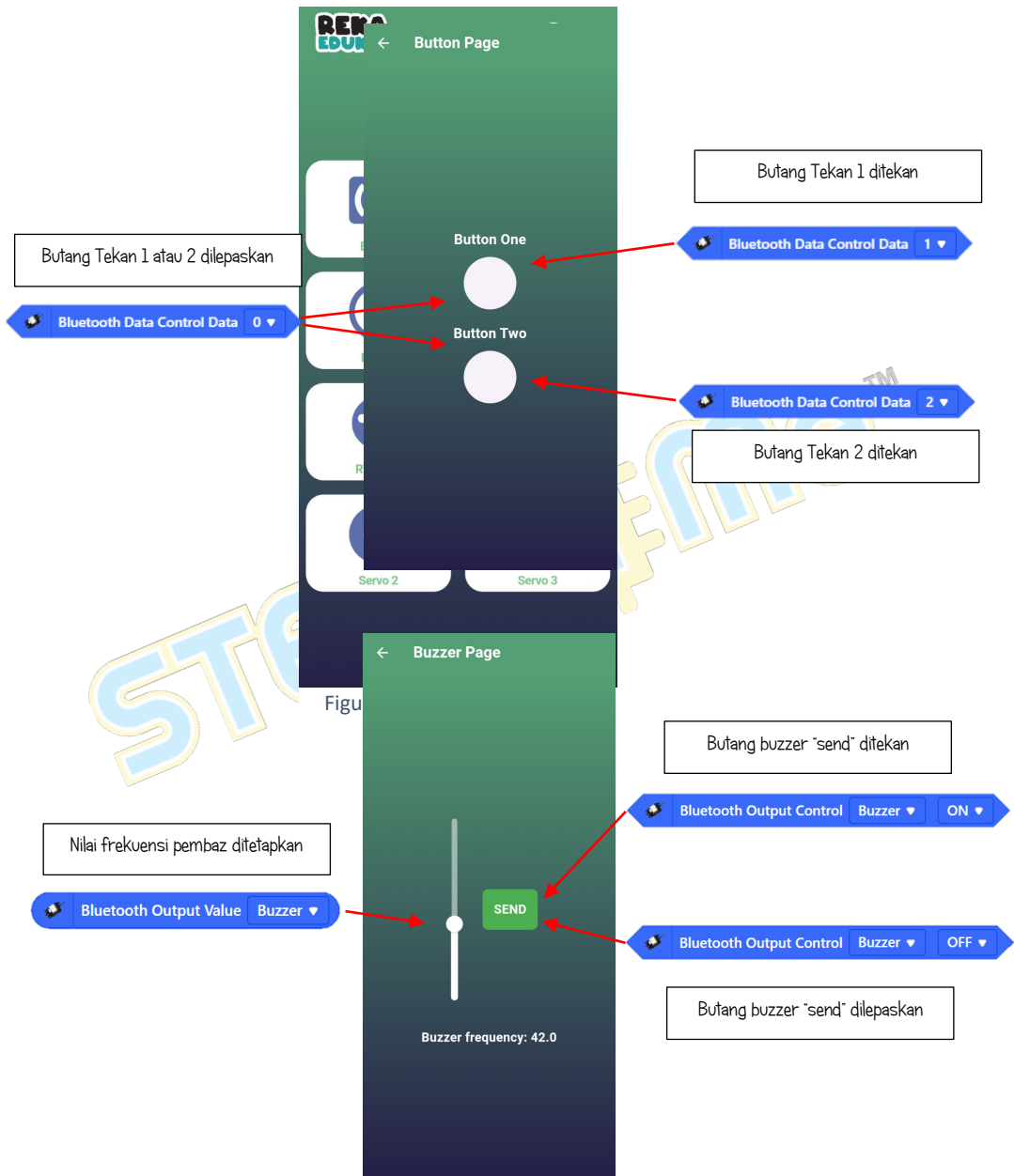
Buka aplikasi Reka Edukit dan pilih nama modul Bluetooth tadi. Modul Bluetooth telahpun Berjaya disambungkan ke aplikasi Reka Edukit.

Modul Bluetooth telah disambungkan pada telefon pintar.

Langkah 6



PENGENALAN KAITAN BLOK PENGKODAN DAN APLIKASI REKA EDUKIT

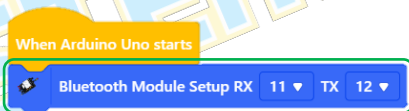




Blok pengkodean mBlok	Fungsi di Aplikasi Reka Edukit
Bluetooth Data Control Data [1]	Butang Tekan 1 ditekan
Bluetooth Data Control Data [2]	Butang Tekan 2 ditekan
Bluetooth Data Control Data [0]	Butang Tekan 1 atau 2 dilepaskan
Bluetooth Output Control [Buzzer] [ON]	Butang buzzer "send" ditekan
Bluetooth Output Control [Buzzer] [OFF]	Butang buzzer "send" dilepaskan
Bluetooth Output Control [NeoPixel LED #] [ON]	NeoPixel LED [#] On
Bluetooth Output Control [Neopixel LED #] [OFF]	NeoPixel LED [#] Off
Bluetooth Output Control [LED Red] [ON]	LED Merah On
Bluetooth Output Control [LED Red] [OFF]	LED Merah Off
Bluetooth Output Control [LED Yellow] [ON]	LED Kuning On
Bluetooth Output Control [LED Yellow] [OFF]	LED Kuning Off
Bluetooth Output Control [LED Green] [ON]	LED Hijau On
Bluetooth Output Control [LED Green] [OFF]	LED Hijau Off
Bluetooth Output Value [Potentiometer]	Nilai potensiometer ditetapkan
Bluetooth Output Value [Buzzer]	Nilai frekuensi pembaz ditetapkan
Bluetooth Output Value [LED Red]	Nilai kecerahan LED ditetapkan
Bluetooth Output Value [Servo 1]	Nilai kedudukan Servo 1
Bluetooth Output Value [Servo 2]	Nilai kedudukan Servo 2
Bluetooth Output Value [Servo 3]	Nilai kedudukan Servo 3

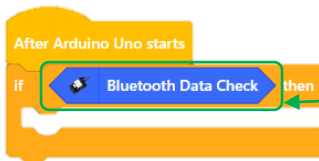
TUTORIAL

1. Seretkan *When Arduino Uno starts* serta blok *Bluetooth Module Setup*.



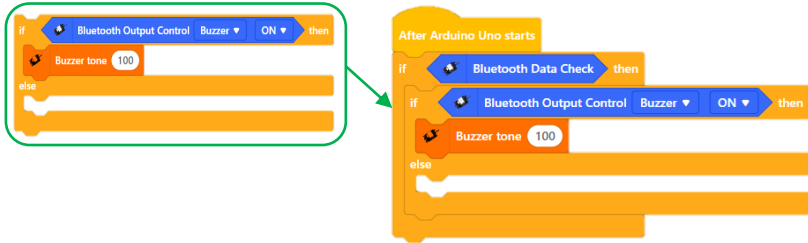
Untuk menetapkan sambungan modul Bluetooth.

2. Seterusnya, seretkan blok *After Arduino Uno starts* dan blok *If then*. Masukkan blok *Bluetooth Data Check* ke dalam ruang heksagon di dalam blok *if then*.

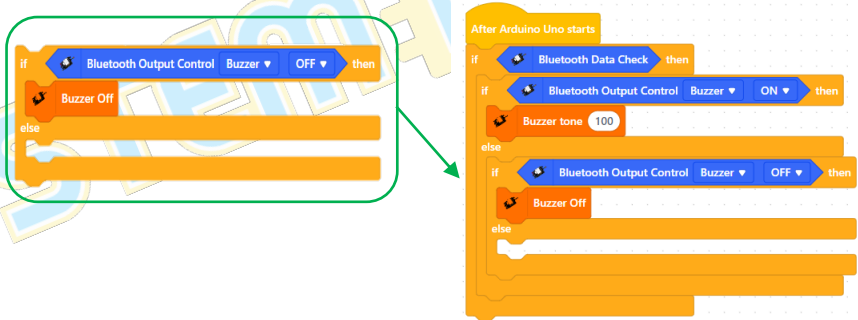


Berfungsi untuk menerima data daripada kontroler (aplikasi telefon).

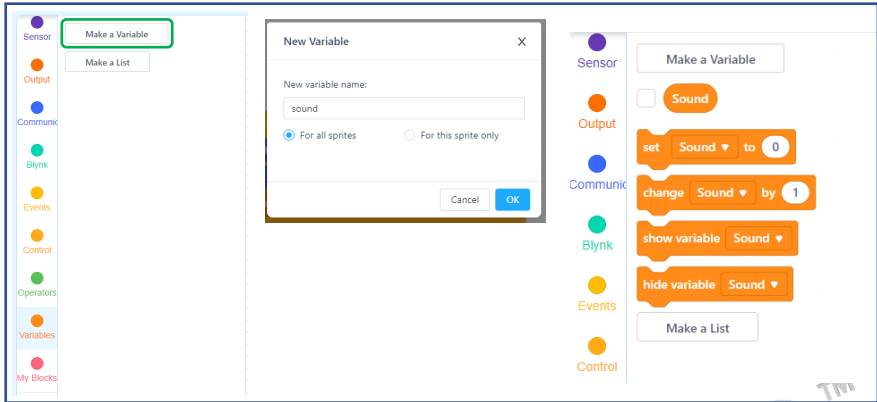
3. Seretkan blok *if: then else* dan masukkan blok *Bluetooth Output control* ke dalam ruang heksagon. Pastikan pilihan adalah "buzzer" dan "on". Kemudian masukkan blok *Buzzer tone* ke dalam blok *if then else*. Tetapkan nilai kepada 100. Masukkan blok *tadi* ke dalam blok *if then*.



4. Seretkan blok *if then else* dan masukkan blok *Bluetooth Output Control* ke dalam ruang heksagon. Tetapkan pilihan kepada "buzzer" dan "off". Seretkan blok *Buzzer off* ke dalam blok *if: then else*. Masukkan blok *tadi* ke dalam ruangan *else*.



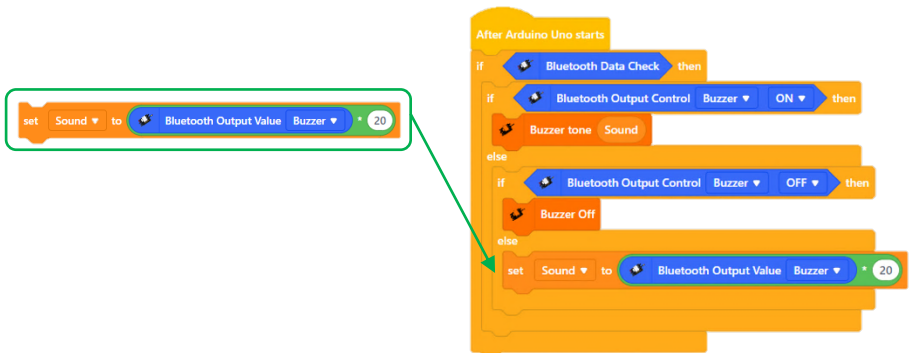
5. Tekan pada menu *Variables* dan *make a variable* untuk membuat pembolehubah. Kemudian blok untuk pembolehubah akan muncul seperti yang ditunjukkan.



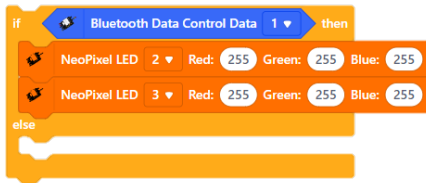
Tahukah anda?

Pembolehubah digunakan untuk menyimpan maklumat untuk dirujuk dan dimanipulasi dalam atur cara komputer. Mereka juga menyediakan cara untuk melabelkan data dengan nama deskriptif supaya program dapat difahami dengan lebih jelas.

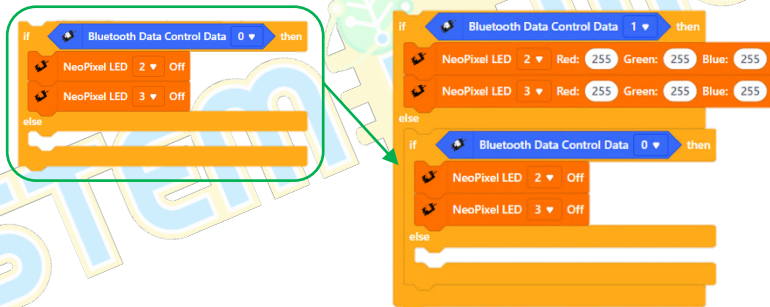
6. Seretkan blok *Bluetooth Output Value* dan masukkan blok tersebut ke dalam blok * (multiply). Kemudian masukkan blok tersebut dalam blok *set sound to*. Masukkan blok tersebut ke dalam ruangan blok *else*.



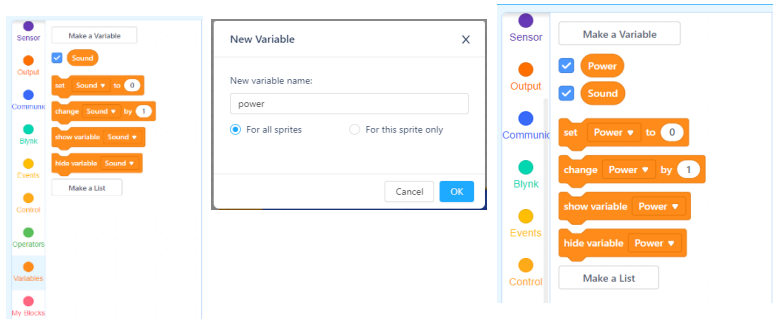
7. Seretkan blok *Bluetooth Data Control Data* dan masukkan ke dalam blok *if: then else*. Tetapkan kepada 1. Kemudian masukkan 2 blok *NeoPixel LED* ke dalam blok *if then else*. Tetapkan NeoPixel LED kepada 2 dan 3.



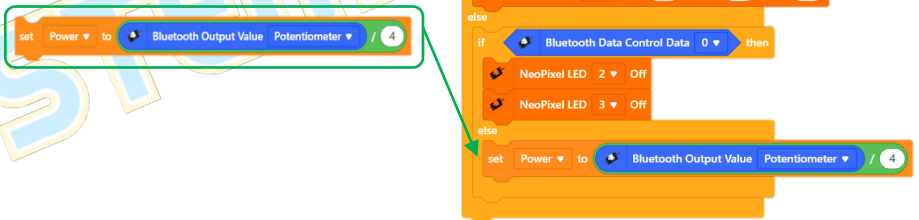
8. Seretkan blok *Bluetooth Data Control Data* dan masukkan ke dalam blok *if: then else*. Tetapkan kepada 0. Kemudian masukkan 2 blok *NeoPixel LED off* ke dalam blok *if then else*. Tetapkan kepada 2 dan 3. Masukkan blok tersebut ke dalam ruangan blok *else*.



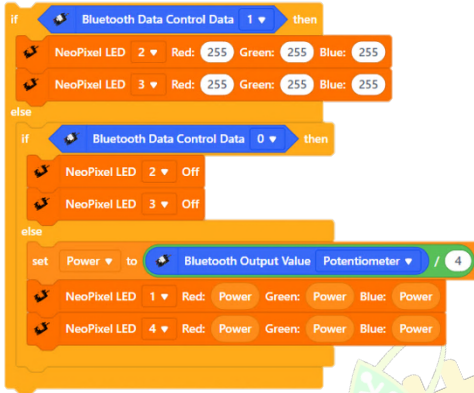
9. Tekan pada menu *Variables* dan *make a variable* untuk membuat pembolehubah. Kemudian blok untuk pembolehubah akan muncul seperti yang ditunjukkan.



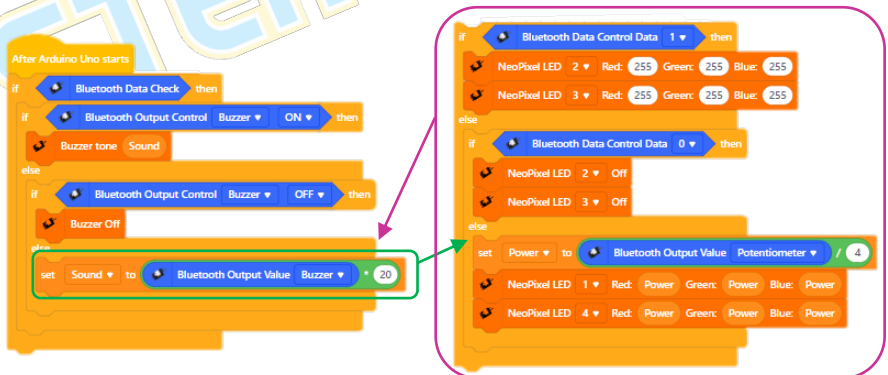
10. Seretkan blok *Bluetooth Output Value* dan masukkan blok tersebut ke dalam blok / (divide). Letak pilihan sebagai Potentiometer dan nilai / kepada 4. Kemudian masukkan blok tersebut dalam blok *set power to*. Masukkan blok tersebut ke dalam ruangan blok *else*.



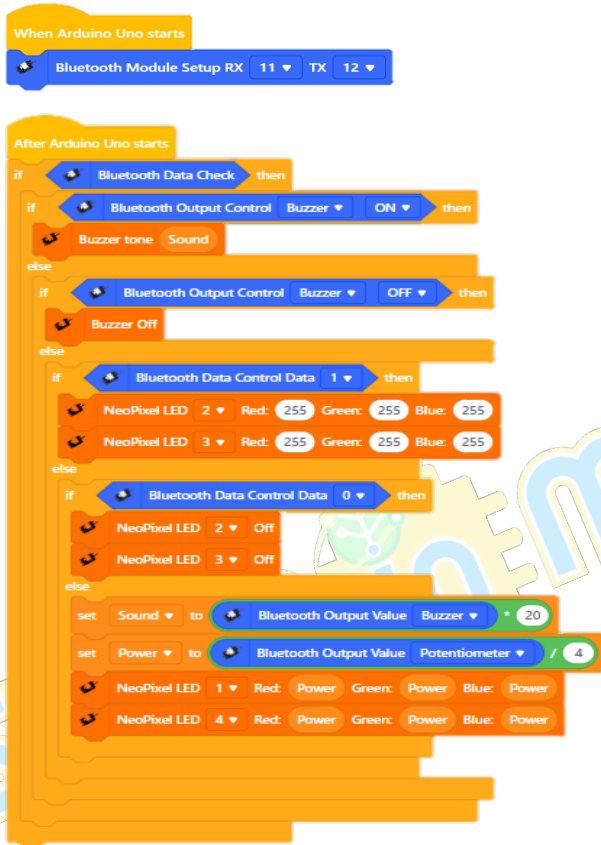
11. Kemudian seretkan 2 blok *NeoPixel LED* dan letakkan dibawah blok *set power to* dalam bahagian *else*. Tetapkan *NeoPixel LED* kepada 1 dan 4. Seretkan blok *power* dari *variables* dan masukkan ke dalam setiap ruang *red, green dan blue* untuk kedua – dua blok.



12. Seretkan blok *set sound to* ke atas blok *set power to*. Kemudian masukkan semua blok tadi ke dalam ruangan *else* dan menggantikan kedudukan *set sound to*.



13. Ini merupakan keseluruhan blok pengkodan untuk projek ini.

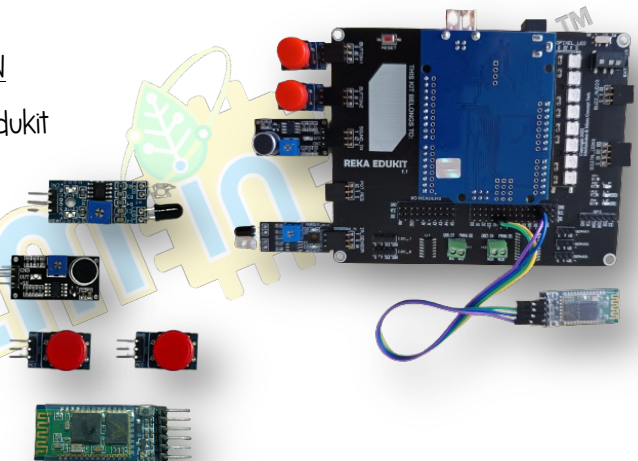


13: PEMANTAU PENDERIA TANPA WAYAR

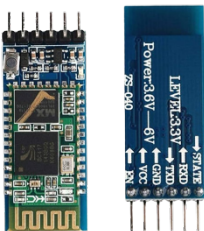
Projek inovatif ini menggunakan Reka Edukit untuk mencipta sistem pemantauan penderia yang komprehensif, mengintegrasikan dua butang tekan, sensor bunyi, dan penderia inframerah (IR). Dilengkapi dengan modul Bluetooth HC-05, sistem ini membolehkan komunikasi tanpa wayar dengan telefon bimbit menggunakan aplikasi RekaEdukit. Pengguna boleh memantau status semua sensor yang disambungkan dan melihat data semasa serta aktiviti sensor melalui aplikasi tersebut. Selain itu, tab "Variable" dalam aplikasi memaparkan jumlah masa setiap sensor mengesan input, menyediakan maklumat yang berguna. Projek ini menawarkan platform yang intuitif dan mesra pengguna untuk mengurus dan berinteraksi dengan pelbagai sensor melalui teknologi mudah alih.

PERANTI YANG DIGUNAKAN

1. Papan litar Reka Edukit
2. Penderia Inframerah
3. Penderia Bunyi
4. Butang Tekan
5. Modul Bluetooth



Jadual di bawah adalah penyambungan modul Bluetooth ke papan litar Reka Edukit menggunakan jumper wire.



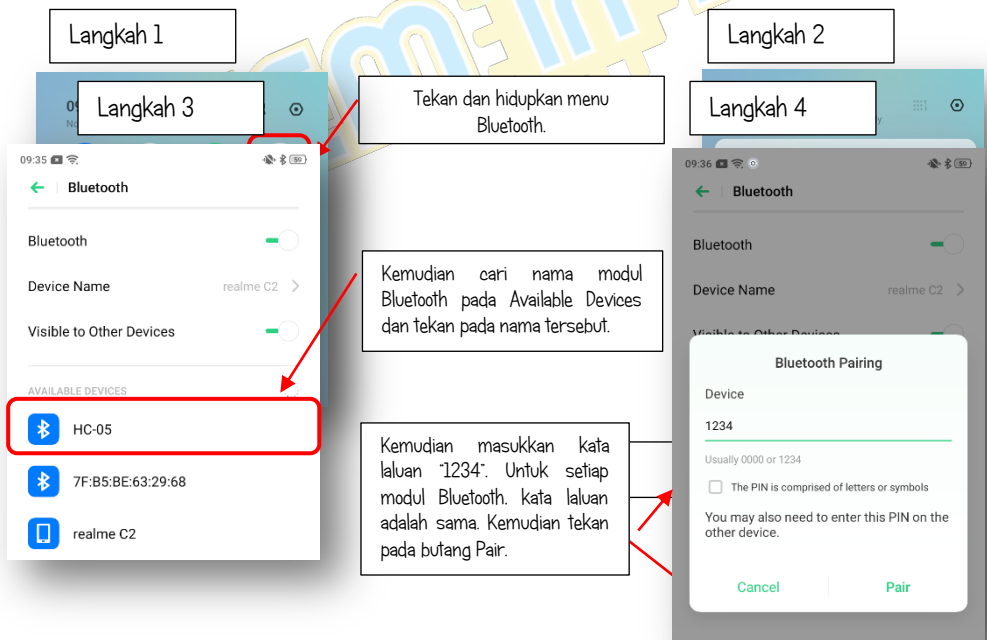
Pin Modul Bluetooth	Pin "IO Header" papan Reka Edukit
RXD	5
TXD	4
GND	GND
VCC	VCC

MENGAPA SAMBUNGAN INI DIGUNAKAN?

Dalam RekaEdukit versi 1.1. tiada port khusus yang ditetapkan untuk sambungan Bluetooth. Ini bermakna, tidak ada port tertentu yang direka khas untuk tujuan ini. Manakala, dalam Reka Edukit versi 1.2 port yang biasa digunakan untuk sambungan Bluetooth, iaitu port 11 dan 12, juga digunakan untuk RGY LED. Oleh itu, port 11 dan 12 tidak dapat digunakan untuk sambungan Bluetooth jika RGY LED digunakan.

Pengguna boleh memilih mana-mana port RXD (Receiver) dan TXD (Transmitter) yang lain untuk sambungan Bluetooth. Port RXD dan TXD ini adalah port yang digunakan untuk komunikasi serial. Namun, adalah sangat penting untuk memastikan bahawa port RXD dan TXD yang dipilih tidak bertindih dengan mana-mana input atau output lain yang sedang digunakan dalam projek. Jika port yang dipilih untuk Bluetooth sudah digunakan untuk fungsi lain, ia akan menyebabkan konflik dan gangguan dalam fungsi Reka Edukit. Pastikan juga port yang dipilih digunakan di dalam coding Bluetooth Module Setup.

CARA PENYAMBUNGAN MODUL BLUETOOTH DENGAN APLIKASI REKA EDUKIT



Langkah 1

Langkah 2

Langkah 3

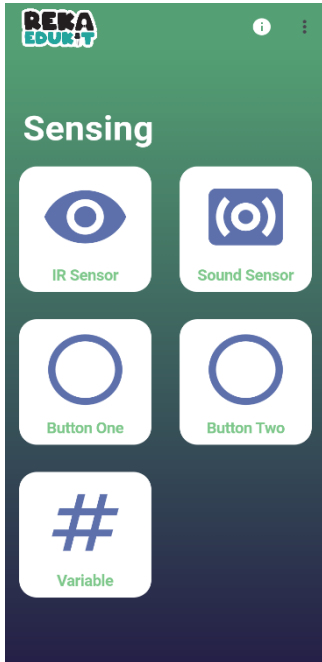
Langkah 4

Tekan dan hidupkan menu Bluetooth.

Kemudian cari nama modul Bluetooth pada Available Devices dan tekan pada nama tersebut.

Kemudian masukkan kata laluan '1234'. Untuk setiap modul Bluetooth, kata laluan adalah sama. Kemudian tekan pada butang Pair.

PENGENALAN KAITAN BLOK PENGKODAN DAN APLIKASI REKA EDUKIT

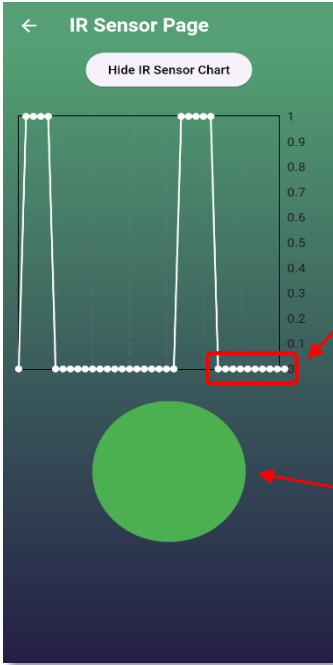


Pilihan Sensing merupakan antara muka yang membenarkan pengguna boleh memantau status sesuatu secara langsung (masa nyata) dan mencatat tempoh masa setiap kali sensor itu dihidupkan atau diaktifkan.



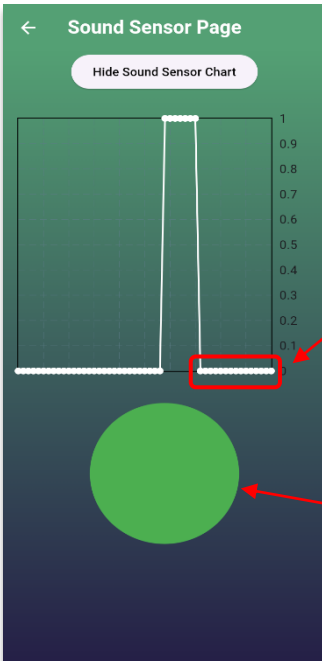
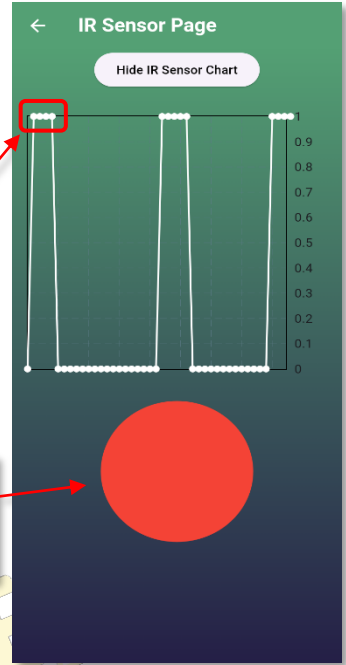
Tekan pada 3 titik pada bahagian kanan atas dan pilih pilihan





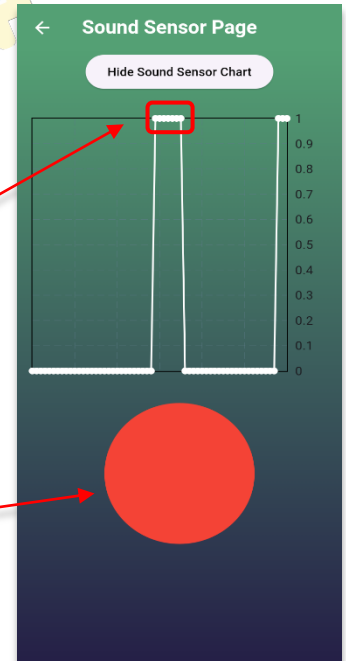
Graf menunjukkan perubahan yang berlaku apabila penderia infrared mengesan objek. Graf menunjukkan nilai 0 jika tidak mengesan objek dan nilai 1 jika mengesan objek.

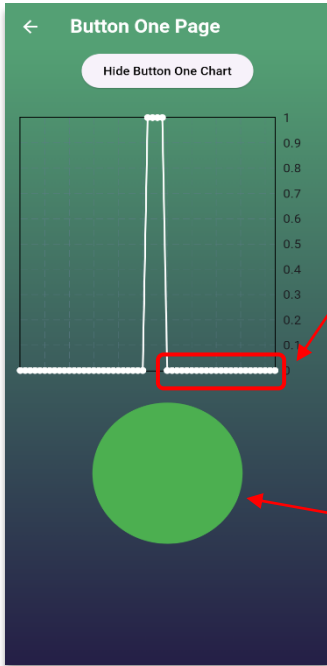
Penunjuk akan berwarna hijau jika penderia infrared tidak mengesan objek dan berwarna merah jika mengesan objek.



Graf menunjukkan perubahan yang berlaku apabila penderia bunyi mengesan bunyi. Graf menunjukkan nilai 0 jika tidak mengesan bunyi dan nilai 1 jika mengesan bunyi.

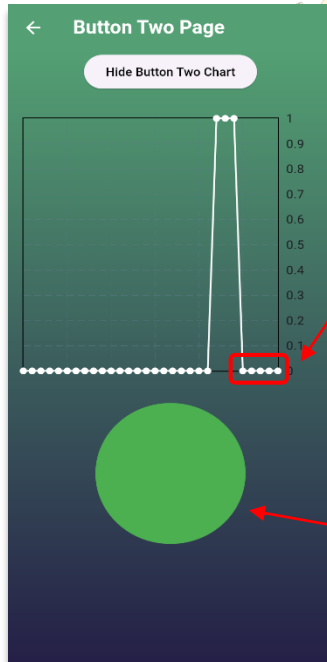
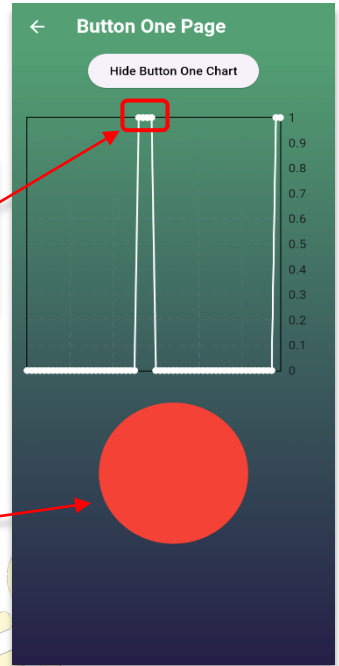
Penunjuk akan berwarna hijau jika penderia bunyi tidak mengesan objek dan berwarna merah jika mengesan objek.





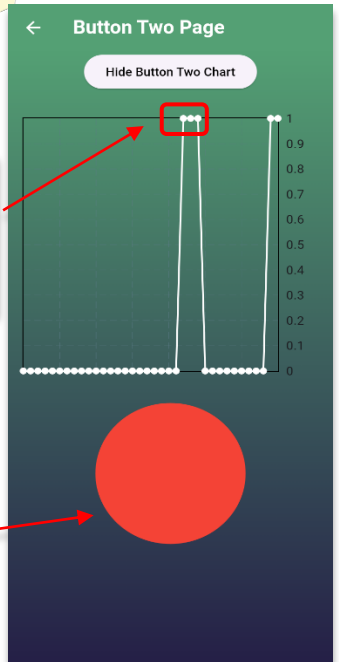
Graf menunjukkan perubahan yang berlaku apabila butang tekan 1 ditekan. Graf menunjukkan nilai 0 jika tidak ditekan dan nilai 1 jika ditekan.

Penunjuk akan berwarna hijau jika butang tekan 1 tidak ditekan dan berwarna merah jika ditekan.



Graf menunjukkan perubahan yang berlaku apabila butang tekan 2 ditekan. Graf menunjukkan nilai 0 jika tidak ditekan dan nilai 1 jika ditekan.

Penunjuk akan berwarna hijau jika butang tekan 2 tidak ditekan dan berwarna merah jika ditekan.



← Variable Page

V1: 0	V2: 0
V3: 0	V4: 0
V5: 0	V6: 0
V7: 0	V8: 0
V9: 0	V10: 0

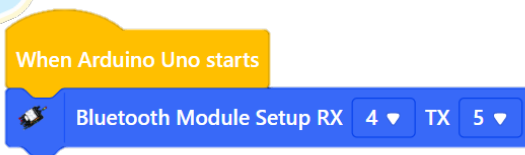
Ini menunjukkan berapa lama (saat) penderita dicetuskan.

← Variable Page

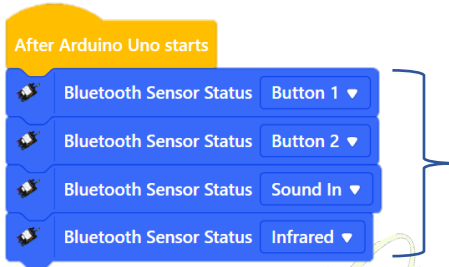
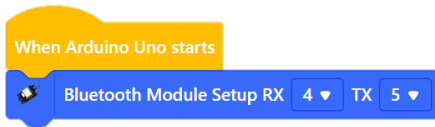
V1: 1	V2: 3
V3: 2	V4: 4
V5: 0	V6: 0
V7: 0	V8: 0
V9: 0	V10: 0

TUTORIAL

1. Seretkan blok *When Arduino Uno starts* dan blok *Bluetooth Module Setup*. Tetapkan *Rx* kepada 4 dan *Tx* kepada 5.

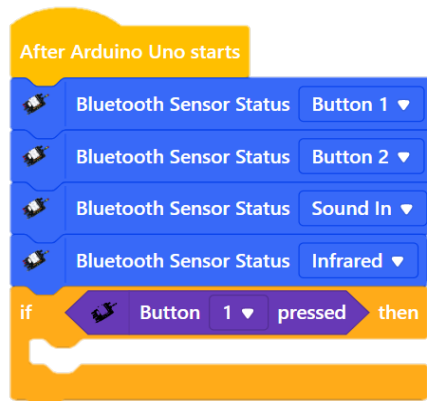
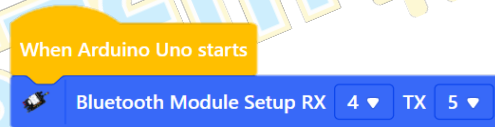


2. Seretkan blok *After Arduino Uno starts* dan 4 *Bluetooth Sensor Status*. Tetapkan kepada *Button 1*, *Button 2*, *Sound In* dan *Infrared*.

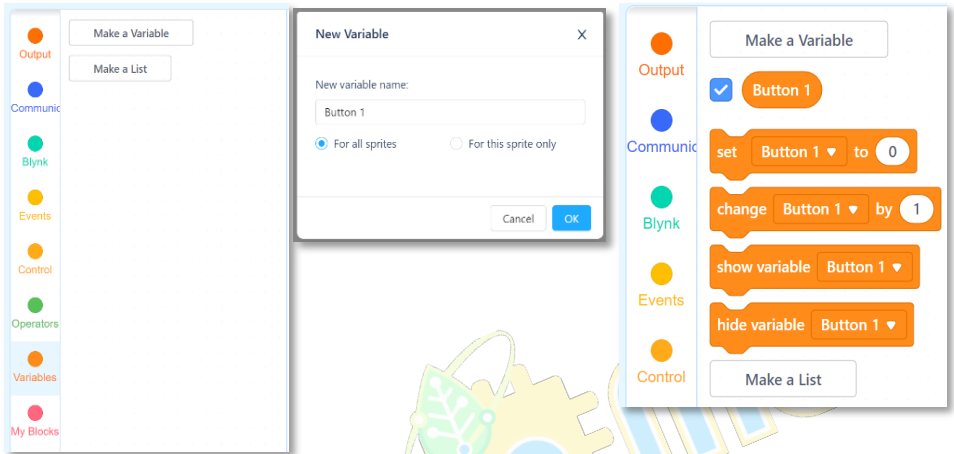


Blok ini berfungsi untuk menghantar data atau status pot penderia kepada kontroler (aplikasi telefon). Aplikasi RekaEdukit akan menunjukkan data dalam bentuk graf dan penunjuk.

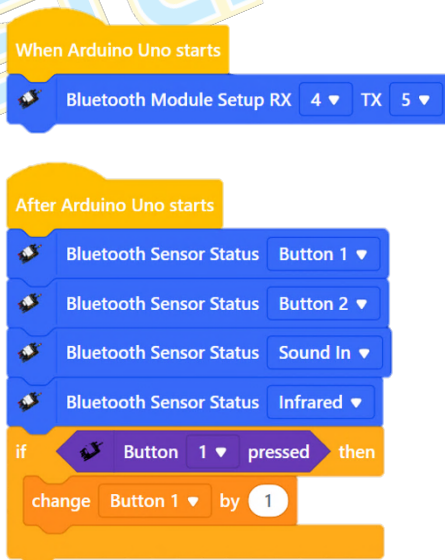
3. Seretkan blok *If then* dan sambungkan. Kemudian seret dan masukkan blok *Button 1 pressed* ke dalam ruang heksagon dalam blok *if then*.



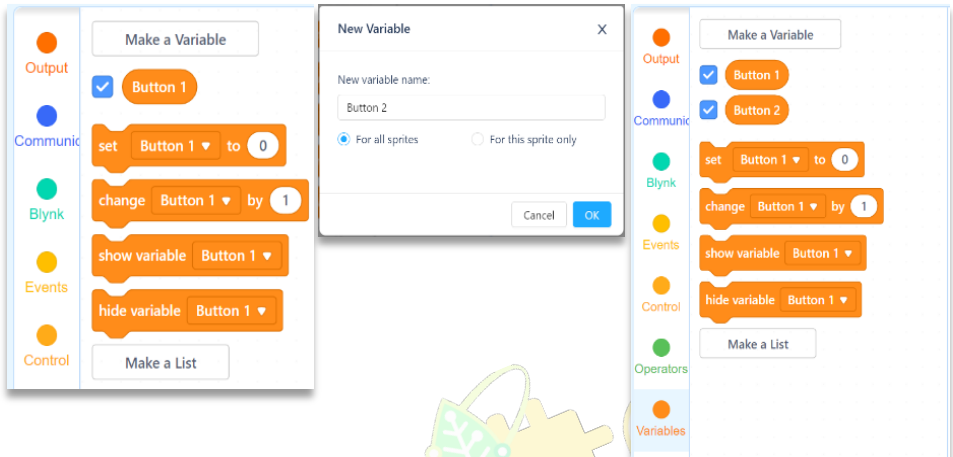
4. Tekan pada menu *Variables* dan *make a variable* untuk membuat pembolehubah. Kemudian blok untuk pembolehubah akan muncul seperti yang ditunjukkan.



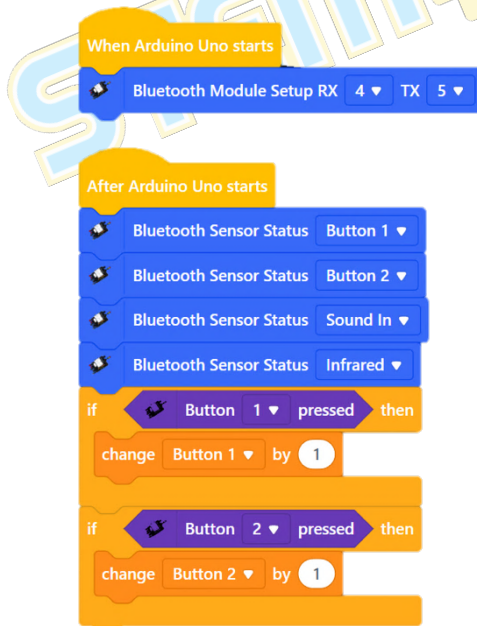
5. Sertakan blok *change Button 1 by 1* dan masukkan kedalam blok *if then*.



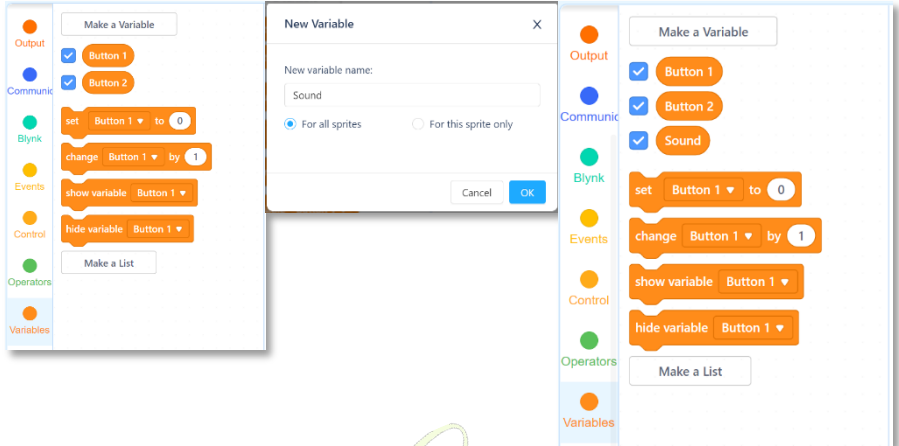
6. Tekan pada menu *Variables* dan *make a variable* untuk membuat pembolehubah. Kemudian blok untuk pembolehubah akan muncul seperti yang ditunjukkan.



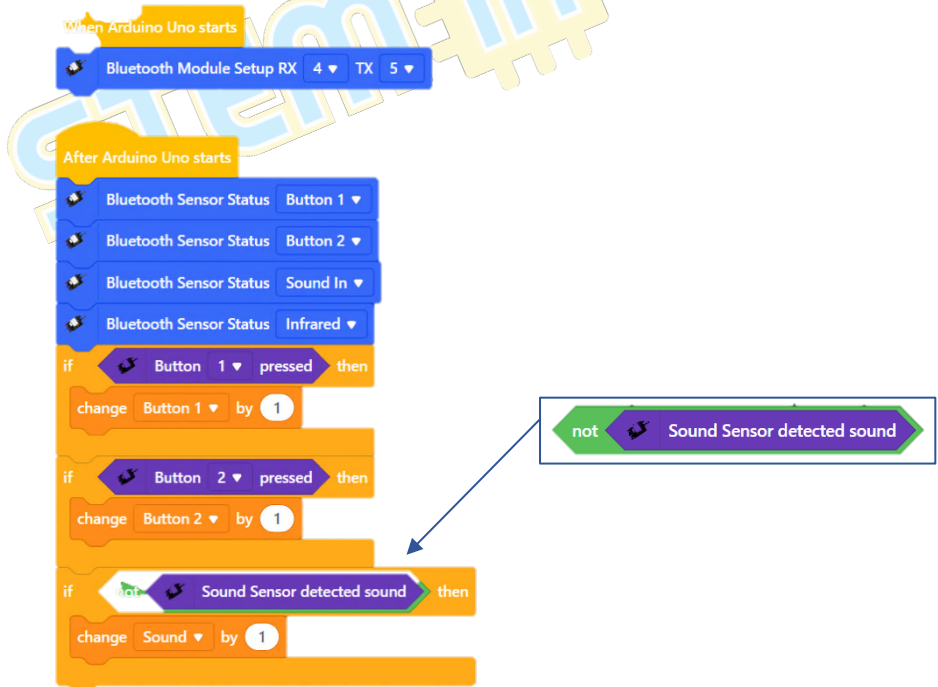
7. Duplicate blok *if then* dan sambungkan ke bawah blok *if then* yang pertama. Tukarkan kepada *Button 2*.



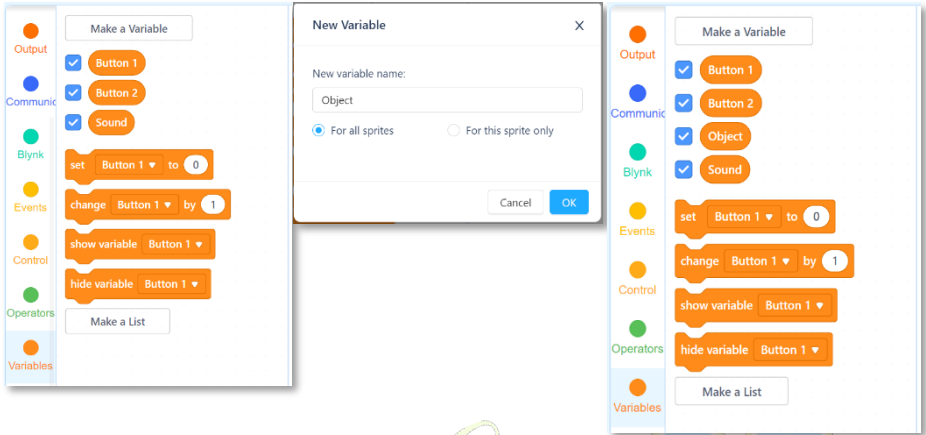
8. Tekan pada menu *Variables* dan *make a variable* untuk membuat pembolehubah. Kemudian blok untuk pembolehubah akan muncul seperti yang ditunjukkan.



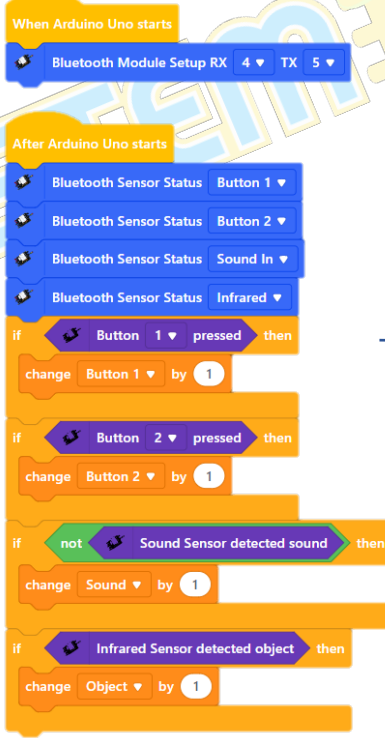
9. Seretkan blok *Sound Sensor detected sound* dan masukkan kedalam blok *not*. Kemudian masukkan kesemua blok ke dalam ruang heksagon dalam blok *if then*. Seretkan blok *change Button 1 by 1* dan tukarkan *Button 1* ke *Sound*.



10. Tekan pada menu *Variables* dan *make a variable* untuk membuat pembolehubah. Kemudian blok untuk pembolehubah akan muncul seperti yang ditunjukkan.

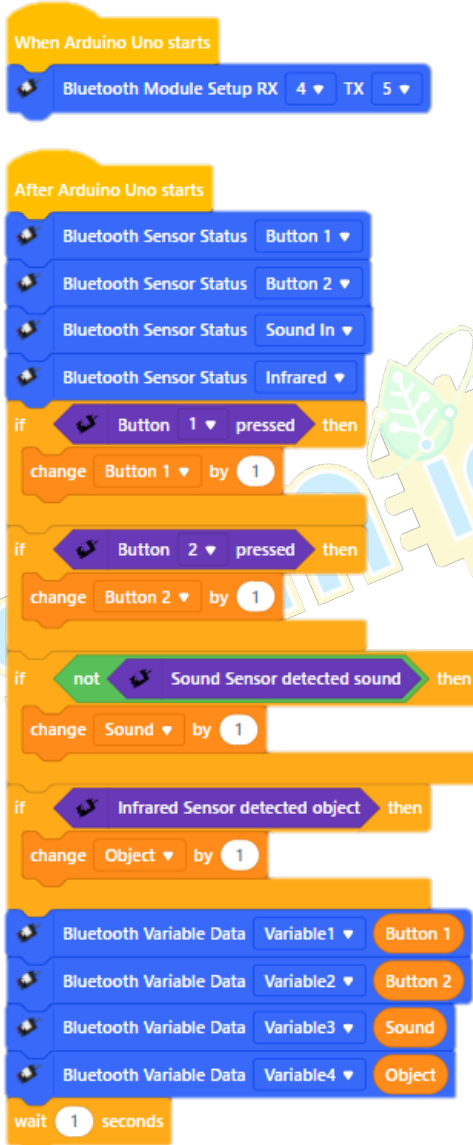


11. Seretkan blok *Infrared Sensor detected object* dan masukkan ke dalam ruang heksagon dalam blok *If then*. Kemudian seret dan masukkan blok *change Button 1 by 1*. Tukarkan *Button 1* kepada *Object*.



Blok ini digunakan sebagai pembilang (counter) untuk setiap penerima.

12. Seretkan 4 blok *Bluetooth Variable Data* dan tetapkan kepada *Variable1*, *Variable2*, *Variable3* dan *variable4*. Kemudian masukkan blok *Button 1*, *Button 2*, *Sound* dan *Object* dari menu *Variables* ke dalam setiap blok *Bluetooth Variable Data*. Yang terakhir, seretkan blok *wait 1 seconds*.



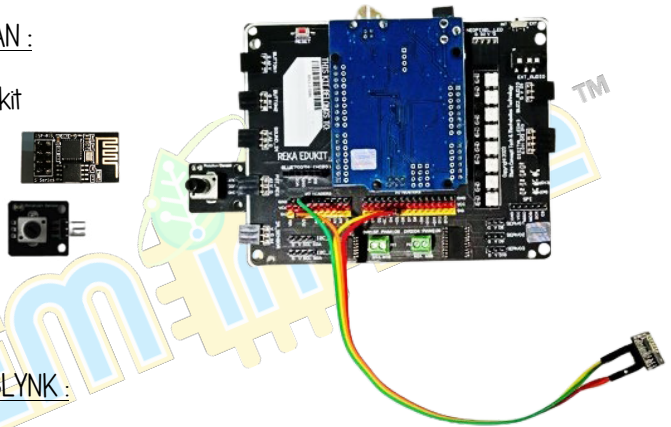
Blok ini berfungsi untuk menghantar nilai pembolehubah sebagai nilai Variable1, Variable2, Variable3 dan Variable4, kepada kontroler (aplikasi telefon) variable page.

14: PENGAWAL LAMPU

Dalam projek ini, kita akan mengawal cahaya lampu menggunakan teknologi Internet of Things (IoT). Kita akan menggunakan sebuah modul WiFi sebagai perantara antara papan litar RekaEdukit dan aplikasi Blynk. Apabila pengguna memusingkan meter upaya, kecerahan cahaya akan berubah. Melalui perisian Blynk, kita akan dapat memantau nilai kecerahan cahaya tersebut. Selain itu, kita juga dapat mengendalikan fungsi hidup dan mati lampu melalui perisian yang sama.

PERANTI YANG DIGUNAKAN :

1. Papan litar RekaEdukit
2. Wifi Modul
3. Meter upaya



PENGENALAN APLIKASI BLYNK :

Blynk adalah platform yang membolehkan pembangun membuat aplikasi Internet of Things (IoT) untuk mengendalikan perkakasan dari jauh menggunakan telefon pintar atau tablet. Ia menyediakan antara muka pengguna yang mudah digunakan untuk mengawal dan memantau pelbagai peranti IoT dan sensor. Blynk menyokong pelbagai papan mikropengawal seperti Arduino, Raspberry Pi, ESP8266, dan lain-lain, menjadikannya alat serba boleh untuk prototaip dan pembangunan IoT. Dengan Blynk, pengguna dapat membuat projek otomasi rumah, peralatan pintar, pemantauan alam sekitar, dan lain-lain, tanpa perlu pengaturcaraan yang rumit.

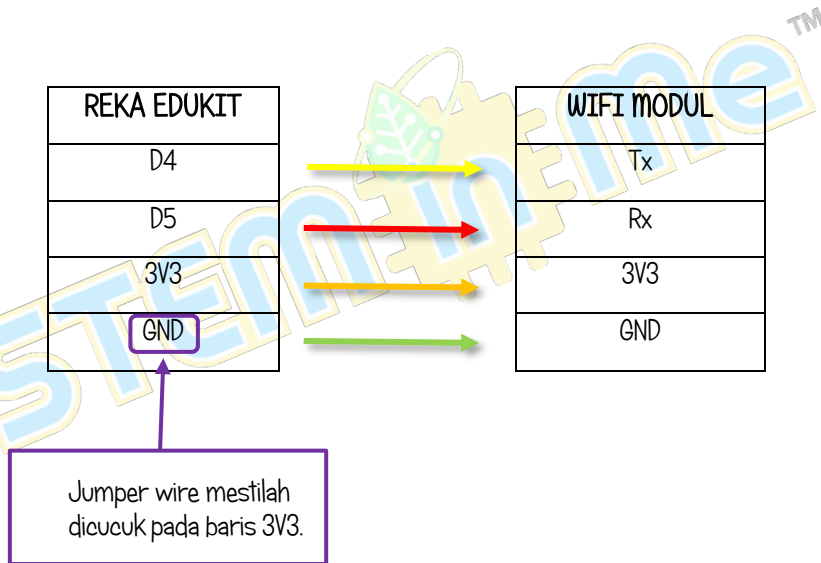


PENGENALAN WIFI MODUL :

Sebuah WiFi Module adalah komponen perkakasan yang membolehkan peranti untuk menyambung kepada rangkaian kawasan tempatan tanpa wayar (WLAN) menggunakan piawai WiFi. Biasanya, ia terdiri daripada litar terpadu kecil dengan fungsi WiFi serta komponen-komponen berkaitan seperti antenna, penyambung, dan kuasa pemprosesan tambahan atau memori.

TUTORIAL PENYAMBUNGAN REKA EDUKIT DAN WIFI MODUL :

Jadual di bawah adalah penyambungan jumper wire dari papan litar Reka Edukit kepada komponen WiFi Module.



TUTORIAL DAFTAR BLYNK

1. Layari laman sesawang Blynk dengan menggunakan pautan atau link yang disediakan.

Link : <https://blynk.io/>

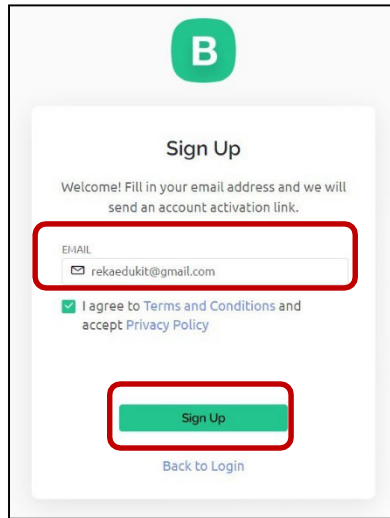
@



2. Klik "Log Masuk" atau "Log In" yang terletak di halaman muka atau antara muka laman sesawang tersebut.. Jika belum mempunyai akaun untuk laman sesawang ini, sila klik atau menekan butang yang bertuliskan "Create New Account" atau "Buat Akaun Baru".

Klik Create new account untuk ke ruangan Sign Up

3. Isi alamat e-mail dan klik Sign Up.



B

Sign Up

Welcome! Fill in your email address and we will send an account activation link.

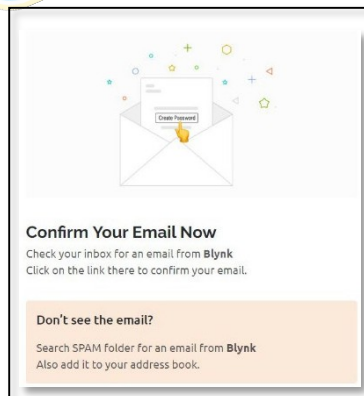
EMAIL

☒ I agree to [Terms and Conditions](#) and [accept Privacy Policy](#)

Sign Up

[Back to Login](#)

4. Untuk mula menggunakan akaun, sila sahkan alamat e-mail dengan membuka e-mail yang dihantar dan menekan pautan pengesahan yang disediakan.



Confirm Your Email Now

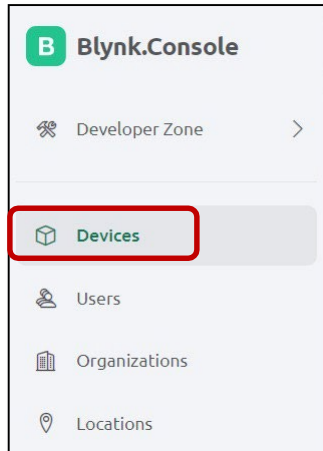
Check your inbox for an email from Blynk
 Click on the link there to confirm your email.

Don't see the email?

Search SPAM folder for an email from Blynk
 Also add it to your address book.

TUTORIAL MEMULAKAN PROJEK BARU:

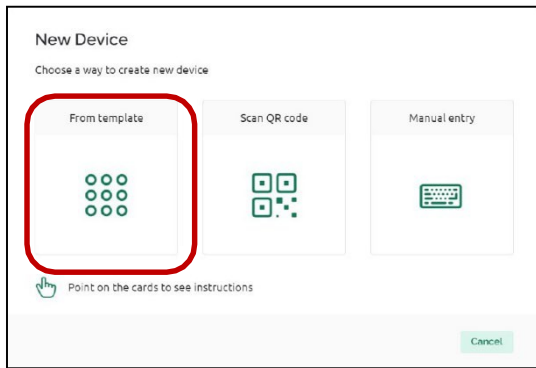
1. Setelah berjaya Log In ke dalam Blynk, langkah seterusnya adalah untuk navigasi ke ruangan atau bahagian yang dikenali sebagai [Blynk.Console](#). Kemudian klik [Devices](#).



2. Pada ruangan ini, klik [+ New Devices](#).



3. Seterusnya, klik **From template**



New Device

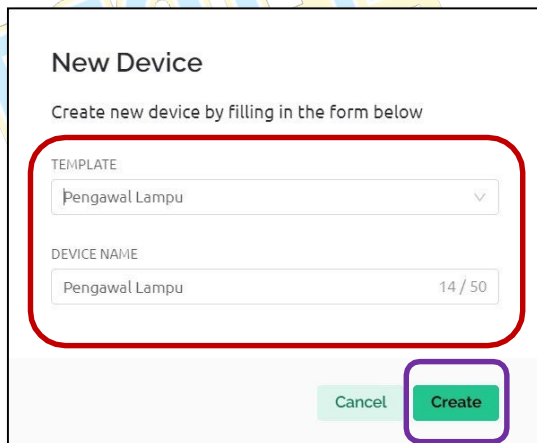
Choose a way to create new device

From template Scan QR code Manual entry

Point on the cards to see instructions

Cancel

4. Pilih projek **Pengawal Lampu** dan isi **DEVICE NAME**. Kemudian klik **Create**.



New Device

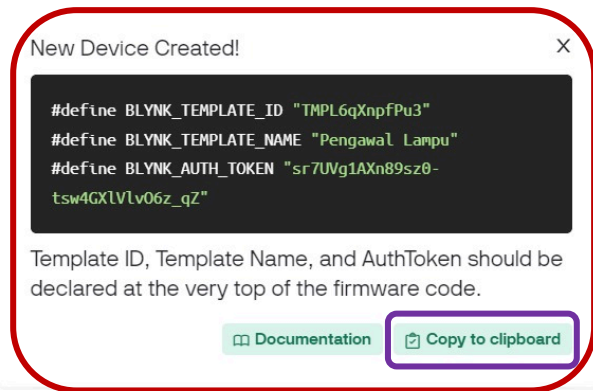
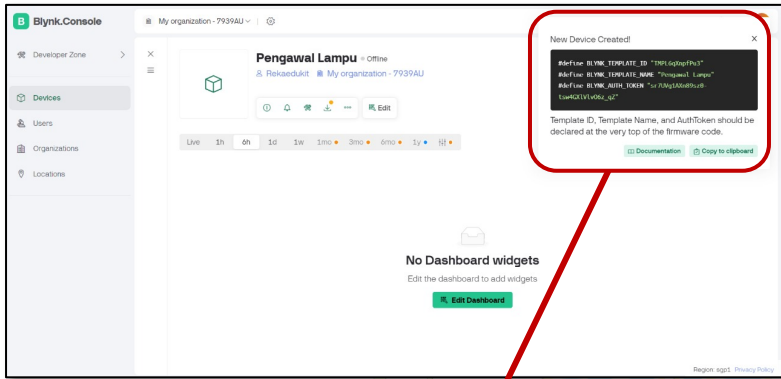
Create new device by filling in the form below

TEMPLATE
Pengawal Lampu

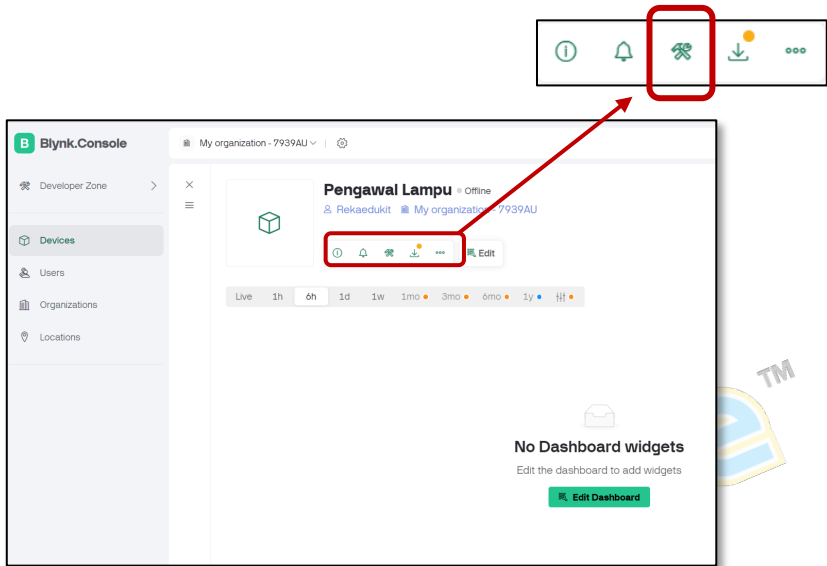
DEVICE NAME
Pengawal Lampu 14 / 50

Cancel **Create**

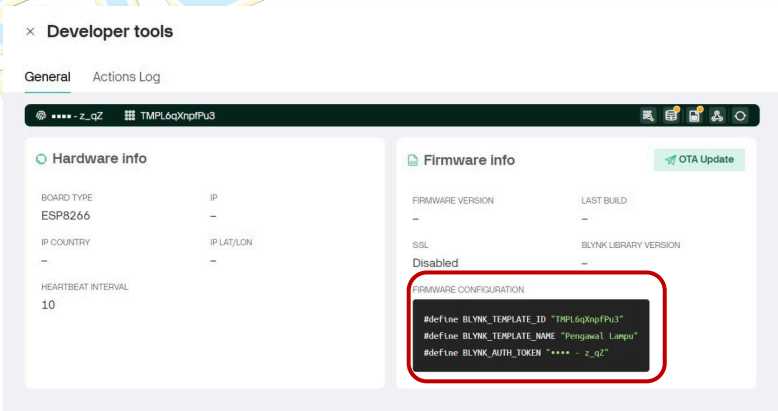
- Paparan skrin akan keluar firmware code seperti yang ditunjukkan di bawah. Klik [Copy to clipboard](#) untuk mendapatkan **TEMPLATE ID**, **TEMPLATE NAME**, dan **AUTHTOKEN**.



6. Selain itu, boleh juga dapatkan di Developers Tools dengan klik butang di bawah.



7. Maklumat mengenai **TEMPLATE ID**, **TEMPLATE NAME**, dan **AUTHTOKEN** akan dipaparan dibahagian Firmware Info.



TUTORIAL MBLOK :

1. Seretkan blok When Arduino Uno starts

When Arduino Uno starts

2. Seretkan blok ESP8266 Configuration SSID. Kemudian isi nama WiFi SSID dan WiFi Password anda. Seterusnya. Pilih nombor 4 di Rx (Receiver) dan nombor 5 di Tx (Transmitter).

When Arduino Uno starts

ESP8266 Configuration SSID: Password: RX: TX: Baud Rate: 115200

3. Seretkan blok Blynk ID, Name, Token.

When Arduino Uno starts

ESP8266 Configuration SSID: Password: RX: TX: Baud Rate: 115200

Blynk ID: Name: Token:

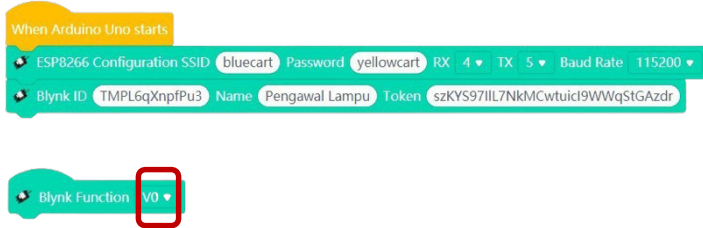
4. Seterusnya, masukkan maklumat Template ID, Nama Template, dan Auth Token di dalam maklumat peranti di Blynk. Ini biasanya boleh diakses melalui menu atau bahagian yang bertajuk "Firmware Info" di dalam peranti Blynk.

When Arduino Uno starts

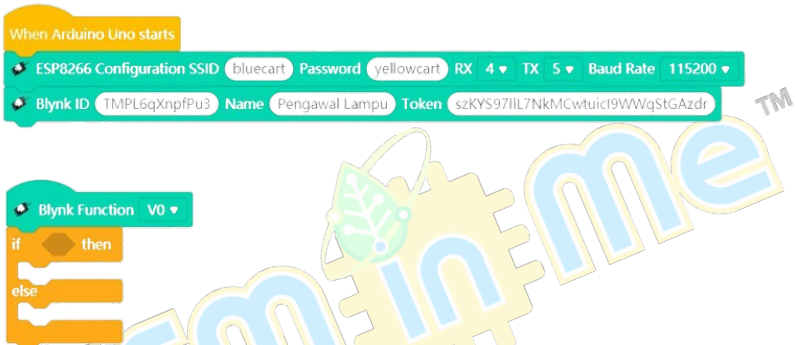
ESP8266 Configuration SSID: Password: RX: TX: Baud Rate: 115200

Blynk ID: Name: Token:

5. Seretkan blok Blynk Function dan tetapkan kepada V0



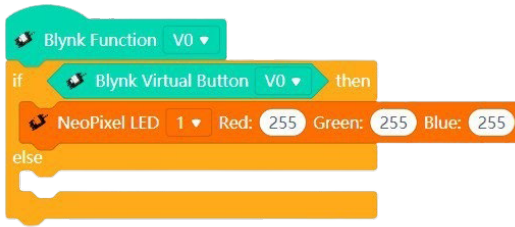
6. Seretkan blok if, then, else



7. Masukkan blok Blynk Virtual Button ke dalam blok if, then, else.



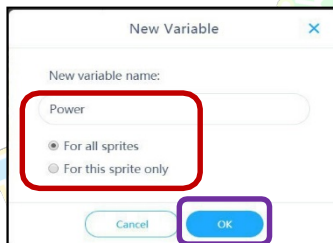
8. Seret blok **NeoPixel LED** ke dalam blok **if, then, else**.



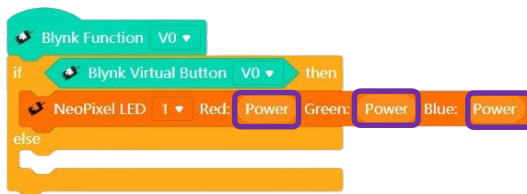
9. Pilih **Make a Variable** di **Variables**.



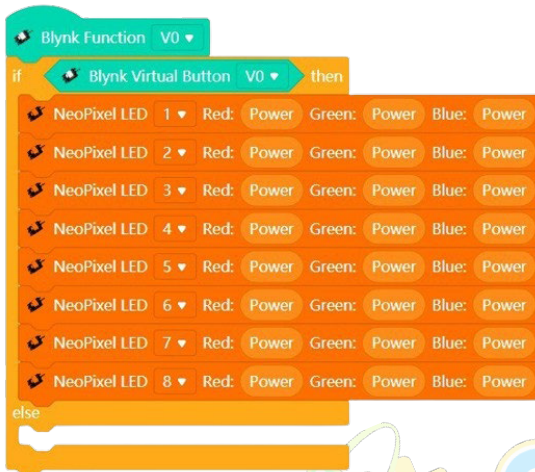
10. Namakan variable **Power** dan pilih **For all sprites**. Kemudian **OK**.



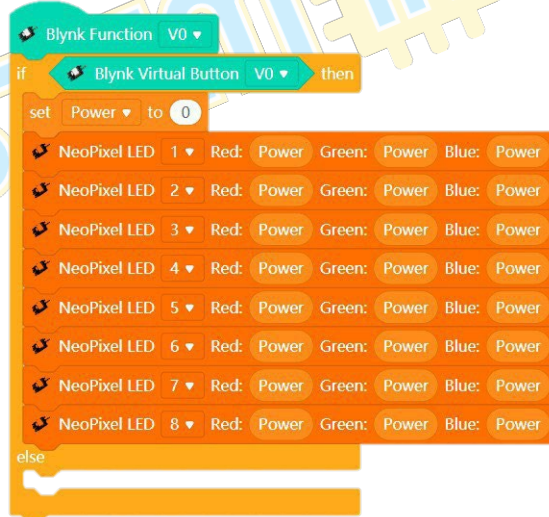
11. Masukkan blok **Power** ke dalam **Red**, **Green**, dan **Blue**.



12. Ulang langkah 11 untuk setiap 8 blok NeoPixel LED.



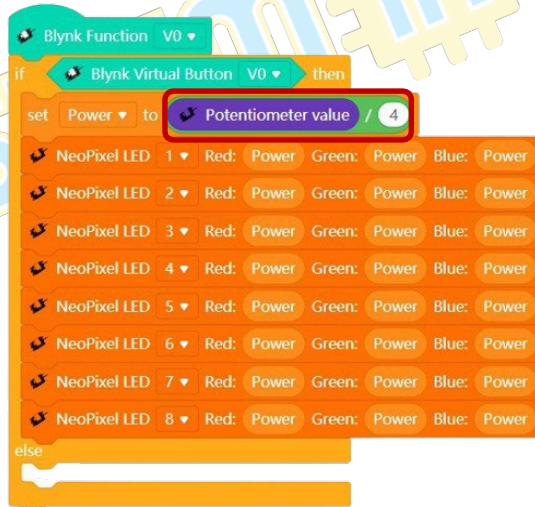
13. Seretkan blok set. Power. to.



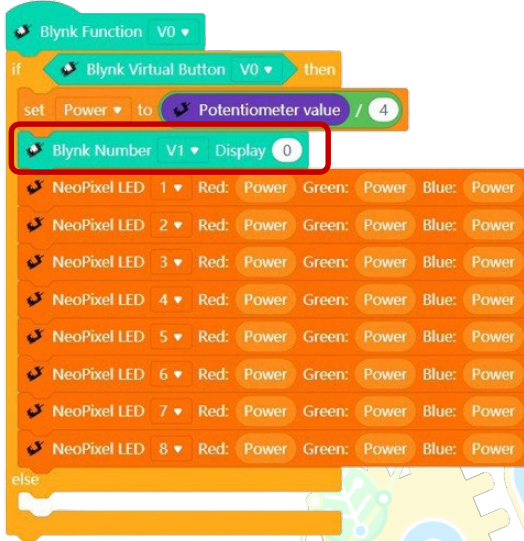
14. Masukkan blok divide ke dalam blok set. Power. to.



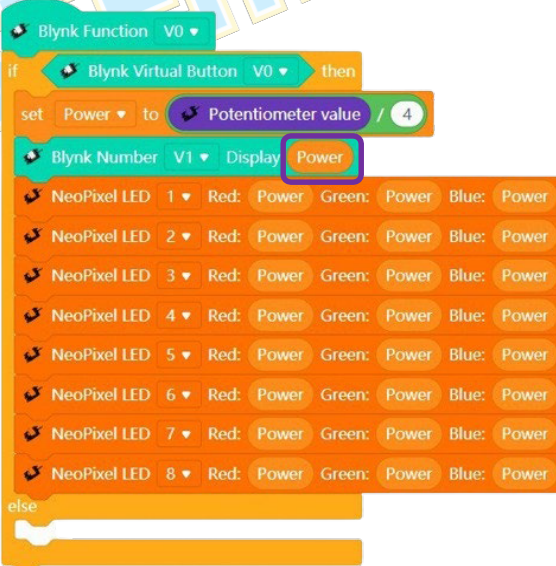
15. Masukkan blok Potentiometer value ke dalam blok set. Power. to. Kemudian isi nomor 4.



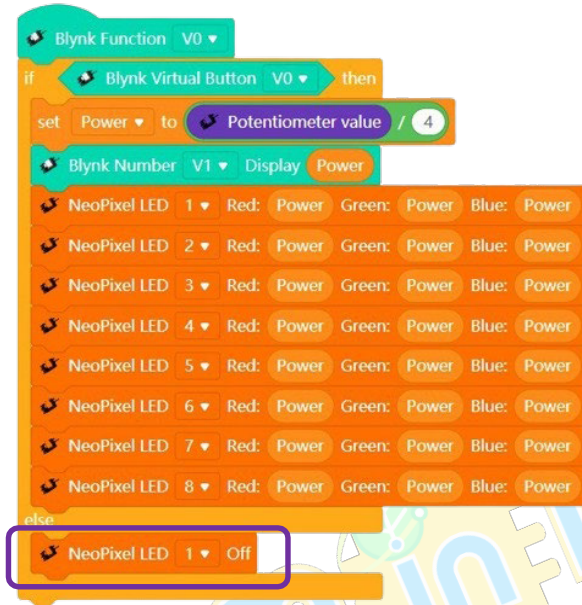
16. Seretkan blok Blynk Number: Display dan tetapkan kepada V1.



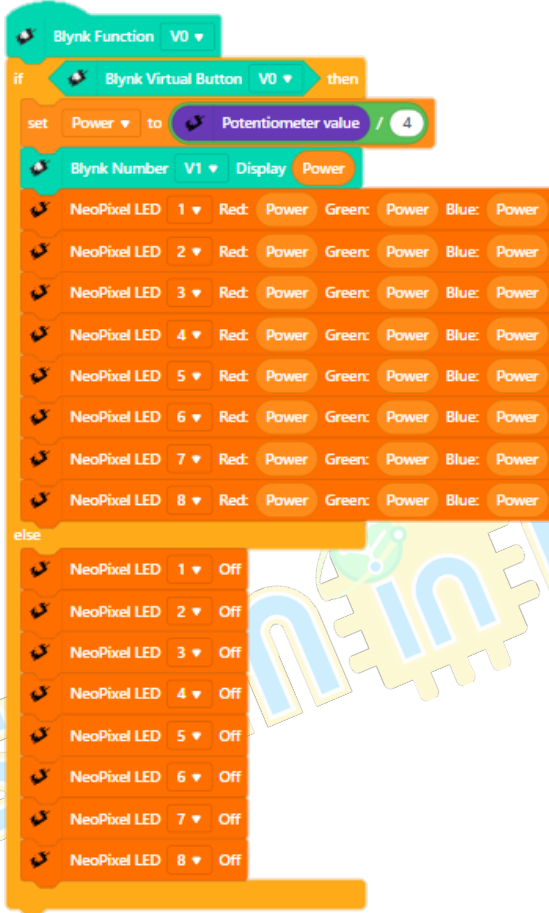
17. Masukkan blok Power ke dalam blok Blynk Number pada bahagian Display.



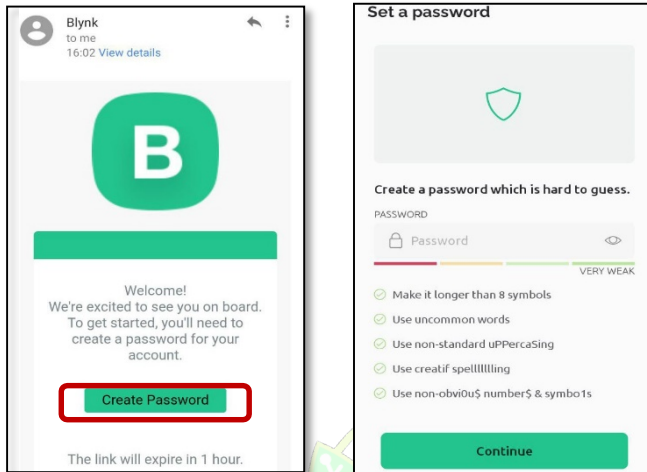
18. Seretkan blok NeoPixel LED Off di bawah else.



19. Ulang langkah 17 untuk setiap 8 blok NeoPixel LED Off.



20. Muat turun aplikasi Blynk IoT di telefon pintar anda. Kemudian Log in dan Create Password. Untuk create password boleh semak di E-mail yang digunakan.



Blynk
to me
16:02 View details

B

Welcome!
We're excited to see you on board.
To get started, you'll need to
create a password for your
account.

Create Password

The link will expire in 1 hour.

Set a password

Create a password which is hard to guess.

PASSWORD

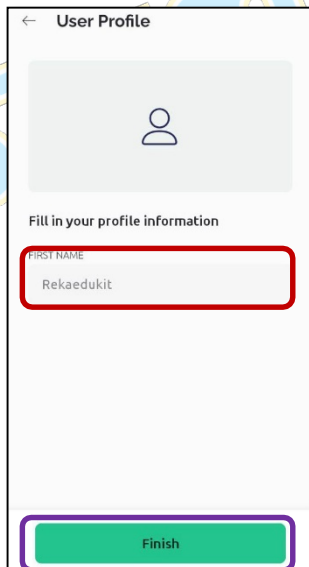
Password

VERY WEAK

- Make it longer than 8 symbols
- Use uncommon words
- Use non-standard uPPercasing
- Use creatif spellllllling
- Use non-obviou\$ number\$ & symbo1s

Continue

21. Isi maklumat profil. Kemudian, klik Finish.



← **User Profile**



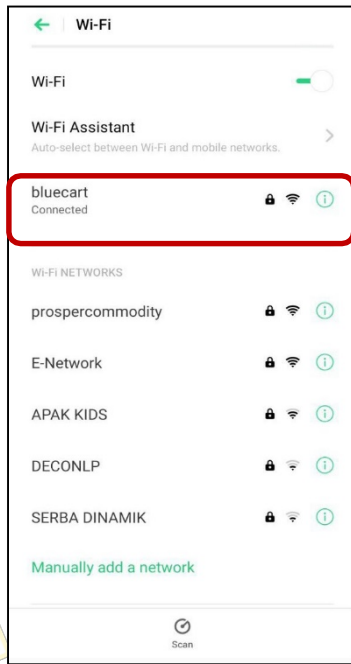
Fill in your profile information

FIRST NAME

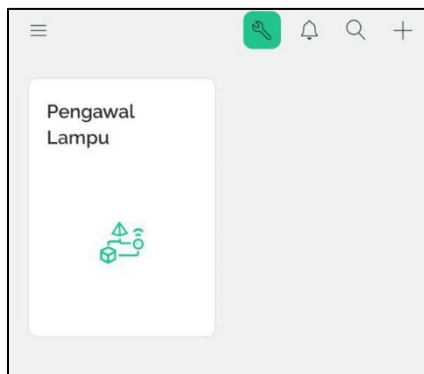
Rekaedukit

Finish

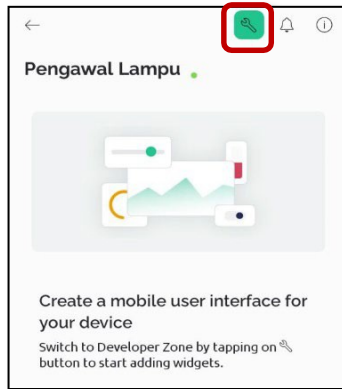
22. Seterusnya, WiFi dihubungkan pada telefon pintar. Pastikan sama seperti di blok ESP8266 Configuration SSID di langkah 2.



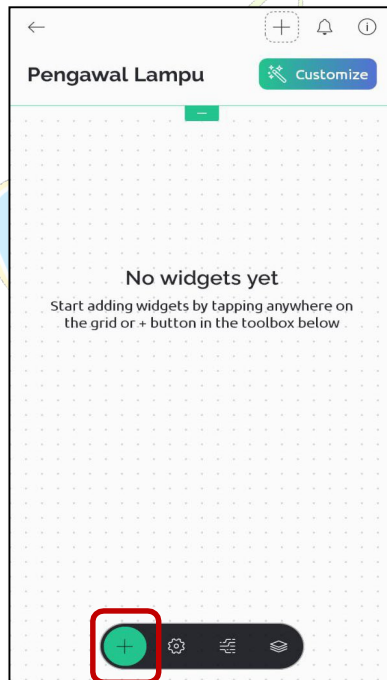
23. Di ruangan hadapan aplikasi Blynk, klik projek **Pengawal Lampu**.



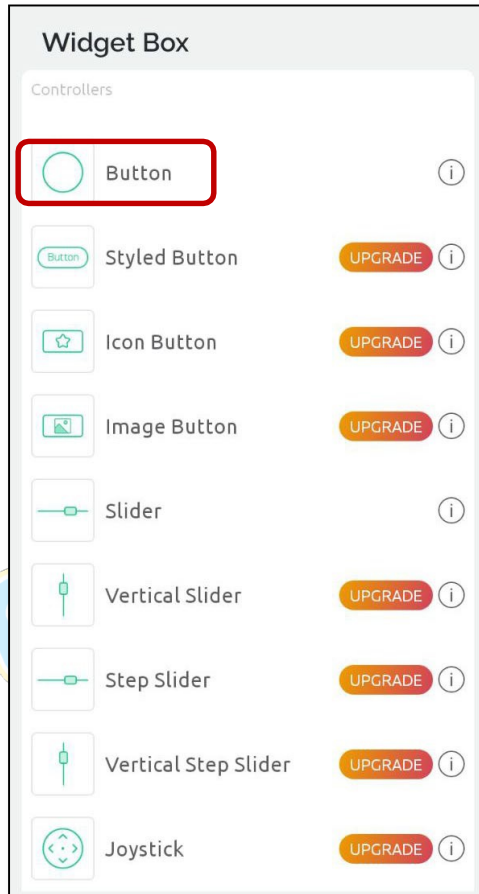
24. Klik pada **butang hijau** seperti di dalam rajah di bawah.



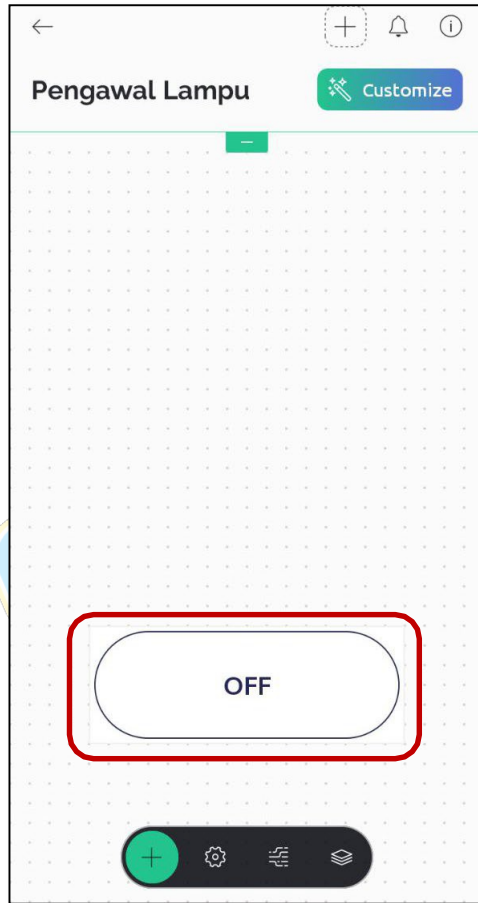
25. Klik symbol "+" di bahagian bawah.



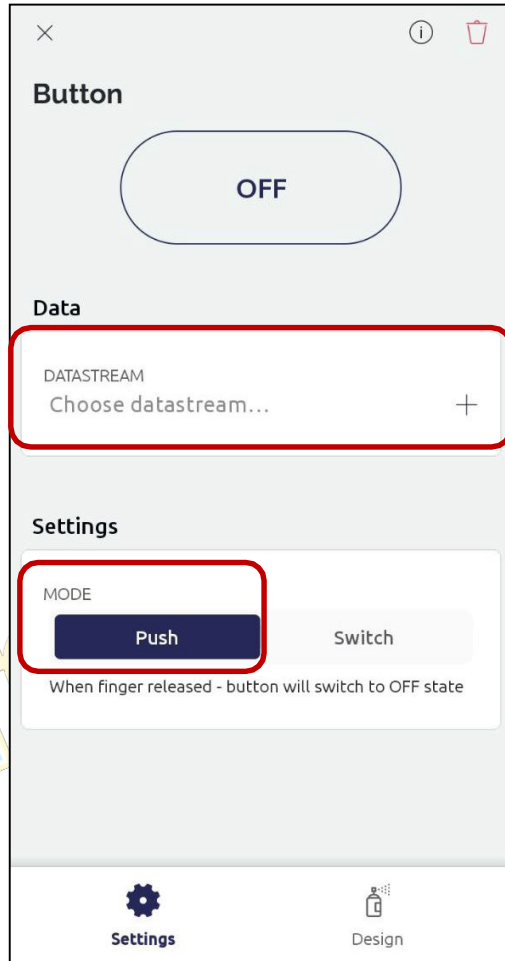
26. Pilih dan klik pada **Button** .



27. Susun kedudukan **Button** mengikut kesesuaian.



28. Tetapkan **MODE** kepada **Push**. Kemudian klik **DATASTREAM**.



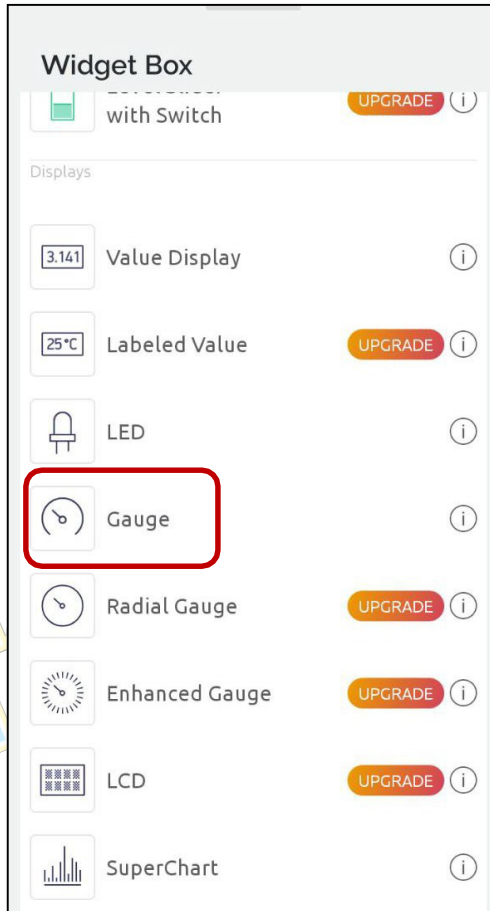
29. Klik pada **Create New**.

30. Pilih **VO** di **VIRTUAL PIN NUMBER** dan **Integer** di **DATA TYPE**. Kemudian, isi **0** di dalam **MIN** dan **1** di dalam **MAX**. Setelah itu, klik **butang "✓"** di bahagian atas.

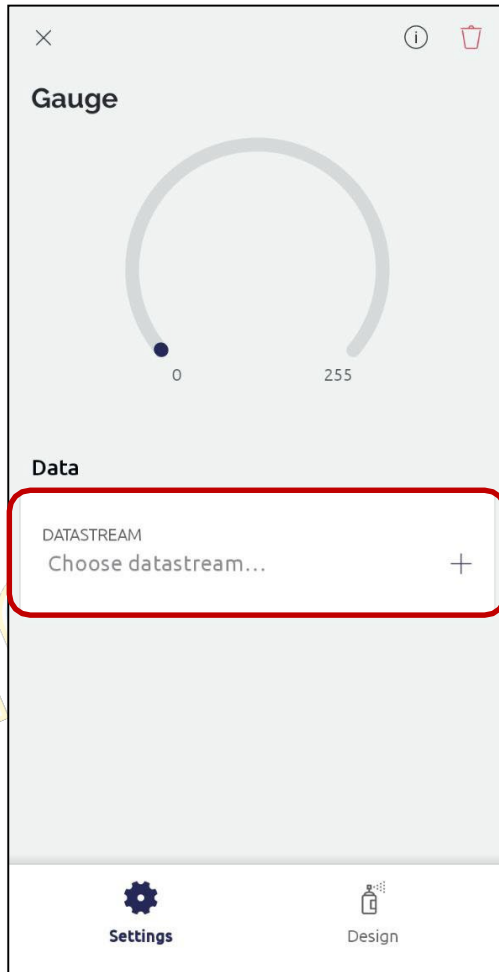
Nilai minimum (MIN) ditetapkan kepada 0. ia mewakili keadaan "OFF".

Nilai maksimum (MAX) ditetapkan kepada 1. ia mewakili keadaan "ON".

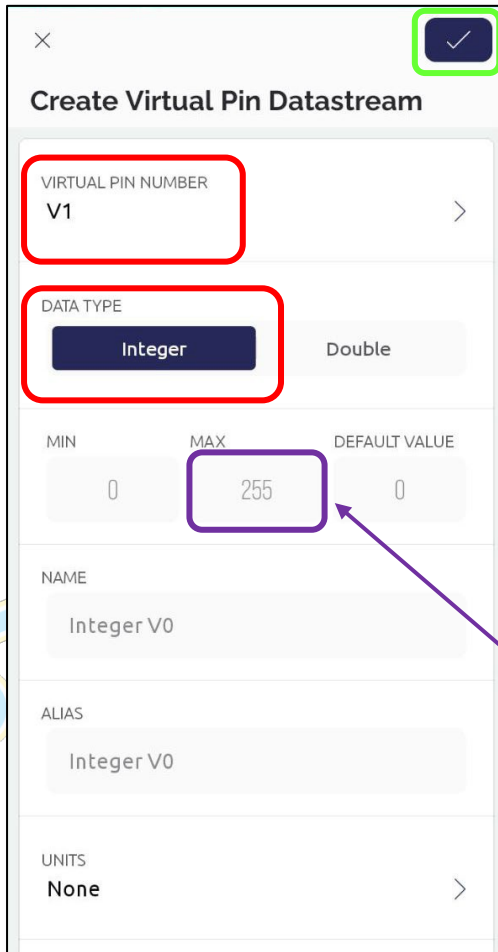
31. Ulang langkah 24. Kemudian, pilih dan klik pada Gauge.



32. Seterusnya, klik **DATASTREAM**.



33. Pilih **V1** di **VIRTUAL PIN NUMBER** dan **Integer** di **DATA TYPE**. Kemudian, klik tombol "✓" di bahagian atas.



Create Virtual Pin Datastream

VIRTUAL PIN NUMBER
V1

DATA TYPE
Integer Double

MIN MAX DEFAULT VALUE
0 **255** 0

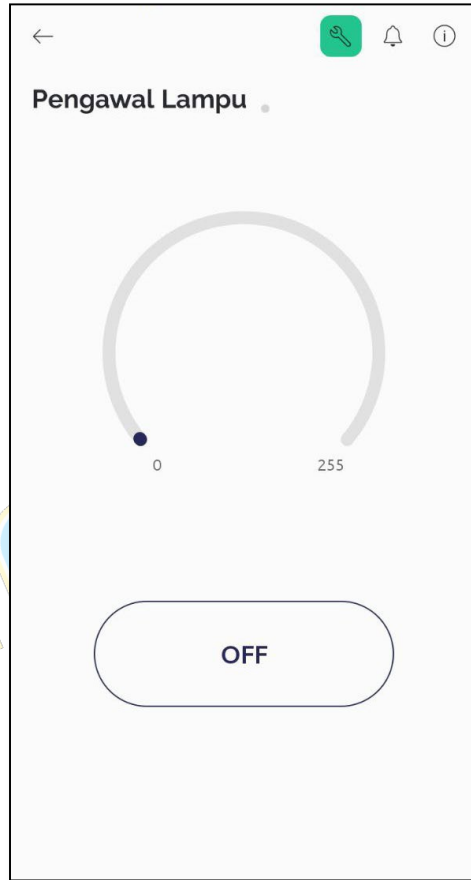
NAME
Integer V0

ALIAS
Integer V0

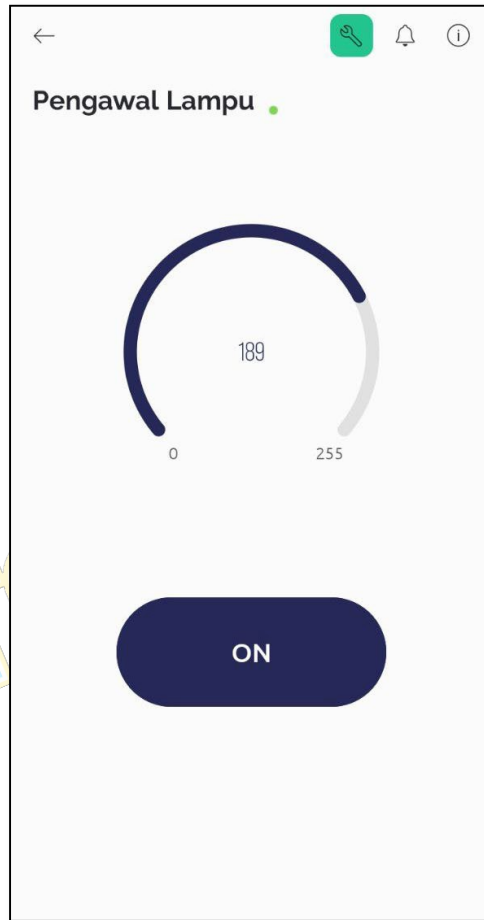
UNITS
None

Nilai maksimum (MAX) bagi NeoPixel adalah 255.

34. Rajah di bawah adalah paparan projek yang telah siap dan sedia untuk digunakan.



35. Rajah menunjukkan perubahan nilai apabila meter upaya dipusingkan. Kemudian, kecerahan cahaya lampu NeoPixel LED akan semakin bertambah.

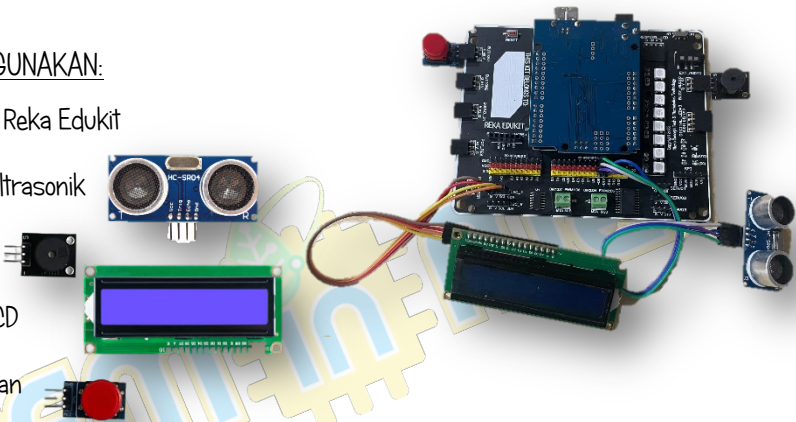


15: KELAS FIZIK

Dalam projek ini, kita akan membangunkan satu sistem yang menggunakan penderia ultrasonik untuk mengukur jarak objek. Sistem ini akan memaparkan nilai jarak yang diukur pada paparan LCD I2C. Setiap kali jarak diukur, buzzer akan mengeluarkan bunyi untuk memberi maklum balas kepada pengguna. Butang tekan digunakan untuk mengosongkan paparan teks pada paparan LCD.

PERANTI YANG DIGUNAKAN:

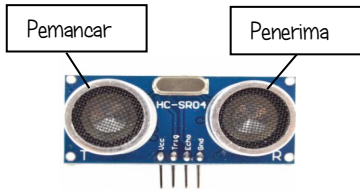
1. Papan litar Reka Edukit
2. Penderia Ultrasonik
3. Pembaz
4. Paparan LCD
5. Butang tekan



PENGENALAN PENDERIA ULTRASONIK

Penderia ultrasonik menggunakan gelombang bunyi ultrasonik untuk mengukur jarak objek dan kemudian mengubah gelombang bunyi yang dipantulkan kepada isyarat elektrik. Terdapat dua komponen utama penderia ultrasonik. Pemancar menggunakan kristal piezoelektrik untuk menjana gelombang bunyi ke arah halangan, dan penerima akan menerima gelombang bunyi yang dipantulkan selepas ia bergerak dari halangan. Penderia ultrasonik akan mengira jarak halangan berdasarkan gema yang diterima.

Jadual di bawah adalah penyambungan jumper wire dari papan litar Reka Edukit kepada penerima Ultrasonik.



Pin Ultrasonik	Pin "IO Header" papan Reka Edukit
Vcc	Vcc
Trig	D10
Echo	D9
Gnd	Gnd

PENGENALAN PAPARAN LCD

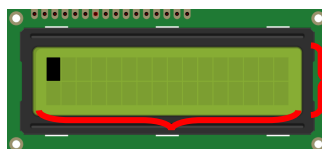
Paparan LCD I2C adalah paparan Kristal cecair (LCD) yang menggunakan protocol I2C (Inter-Integrated Circuit). Penggunaan protocol I2C dapat mengurangkan bilangan pin yang diperlukan untuk menyambungkan LCD kepada mikropengawal, menjadikannya lebih mudah untuk projek dengan pin I/O yang terhad. Untuk kenal pasti dan memberi arahan kepada setiap peranti I2C, Reka Edukit akan menggunakan alamat unik (Address 0x) yang ada pada peranti-peranti I2C.

LCD ini mempunyai dua baris dan 16 lajur, yang bermaksud ia dapat menampilkan teks dalam dua baris dengan setiap baris mampu menampung hingga 16 karakter. Ini sesuai untuk memaparkan pesan singkat atau maklumat dalam format yang teratur dan mudah dibaca.

Jadual di bawah adalah penyambungan jumper wire dari papan litar Reka Edukit kepada peranti paparan LCD.



Pin LCD	Pin I2C_1 atau I2C_2 papan Reka Edukit
VCC	V
SDA	SDA
SCL	SCL
GND	GND



16 lajur (16 karakter)

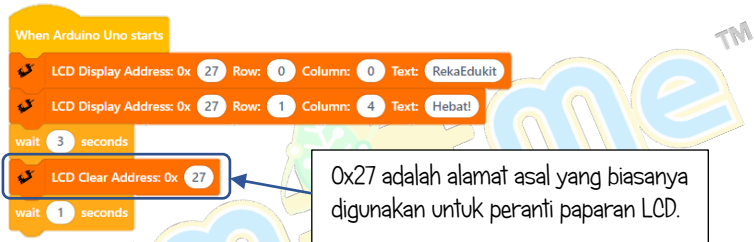
2 baris

TUTORIAL

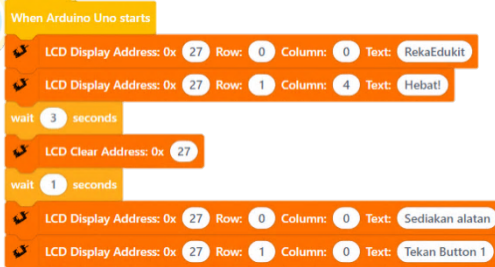
1. Seretkan *When Arduino Uno starts* diikuti 2 blok *LCD Display Address*. Kemudian seretkan blok *wait 1 seconds* dan tetapkan kepada 3.



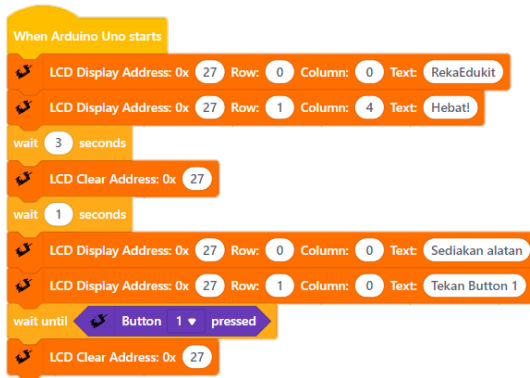
2. Seretkan blok *LCD Clear Address* dan blok *wait 1 seconds*.



3. seretkan 2 blok *LCD Display Address* dan tetapkan 0x kepada 27, row kepada 0 dan 1, column kepada 0 dan text kepada *Sediakan alatan* dan *Tekan Button 1*.



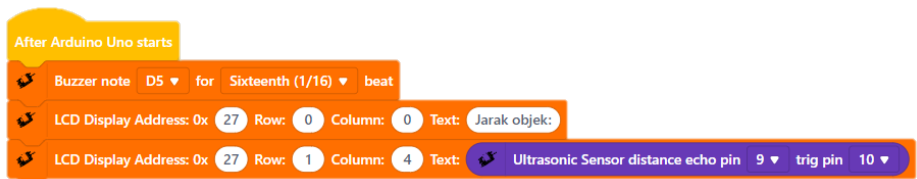
4. Seretkan blok **wait until** dan masukkan blok **Button 1 pressed** ke dalam blok **wait until**. Kemudian seretkan blok **LCD Clear Address**.



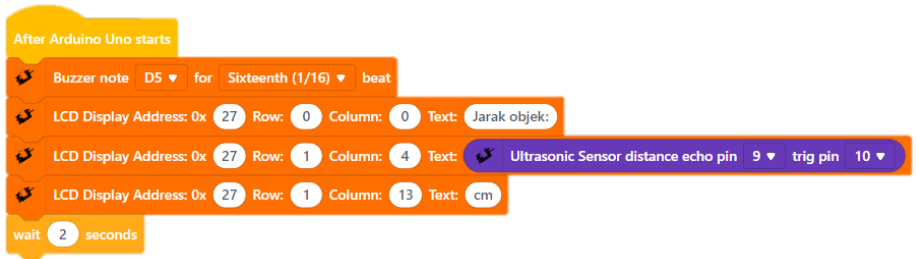
5. Seretkan blok **After Arduino Uno starts** dan blok **Buzzer note**. Tetapkan kepada D5 dan sixteenth (1/16) untuk beat.



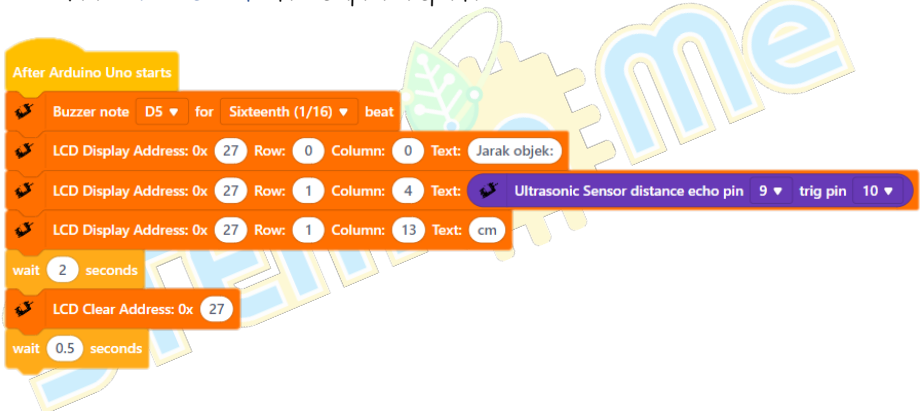
6. Seretkan 2 blok **LCD Display Address** dan tetapkan text kepada **Jarak objek**: pada blok yang pertama. Kemudian. seretkan blok **Ultrasonic Sensor distance echo pin** dan masukkan kedalam blok **LCD Display Address** yang kedua.



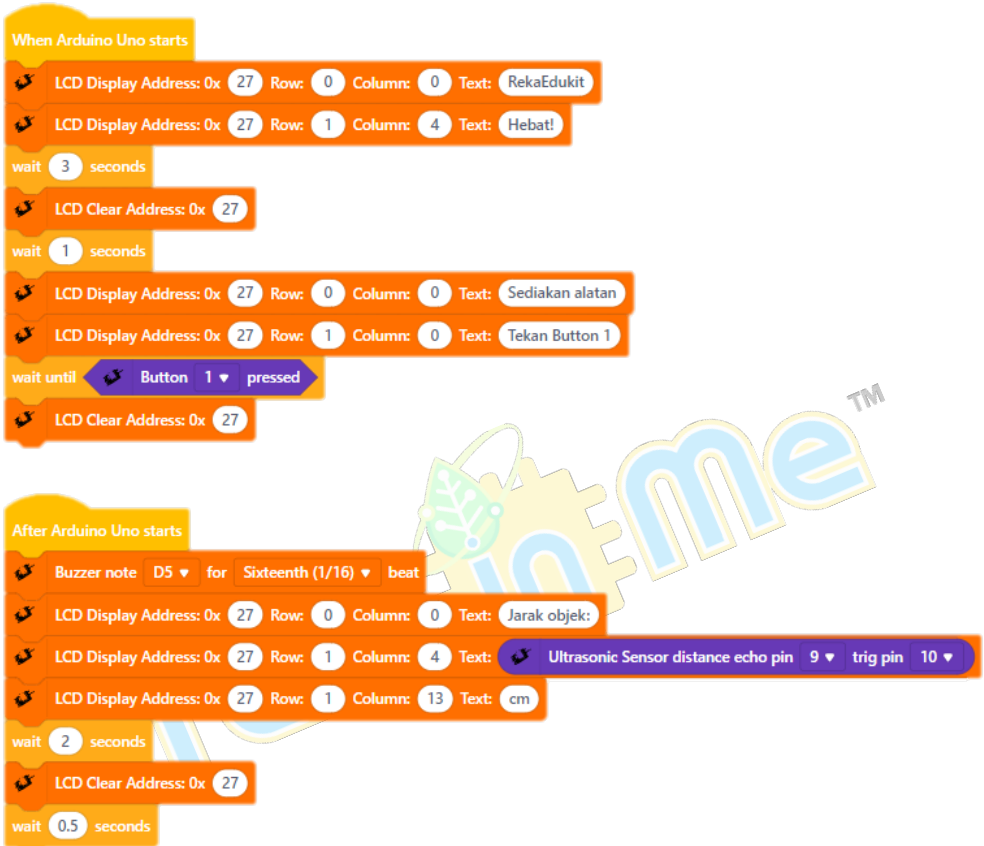
7. Seterusnya, seretkan blok **LCD Display Address** dan tetapkan **0x** kepada **27**, row **1**, column **13** dan text **cm**. Kemudian seretkan blok **wait 1 seconds** dan tetapkan kepada **2**.



8. Seretkan blok **LCD Clear Address** dan tetapkan **0x** kepada **27**. Kemudian seretkan blok **wait 1 seconds** dan tetapkan kepada **0.5**.



9. Ini merupakan keseluruhan blok pengkodan untuk projek ini.



```

When Arduino Uno starts
  LCD Display Address: 0x 27 Row: 0 Column: 0 Text: RekaEdukit
  LCD Display Address: 0x 27 Row: 1 Column: 4 Text: Hebat!
  wait 3 seconds
  LCD Clear Address: 0x 27
  wait 1 seconds
  LCD Display Address: 0x 27 Row: 0 Column: 0 Text: Sediakan alatan
  LCD Display Address: 0x 27 Row: 1 Column: 0 Text: Tekan Button 1
  wait until Button 1 pressed
  LCD Clear Address: 0x 27

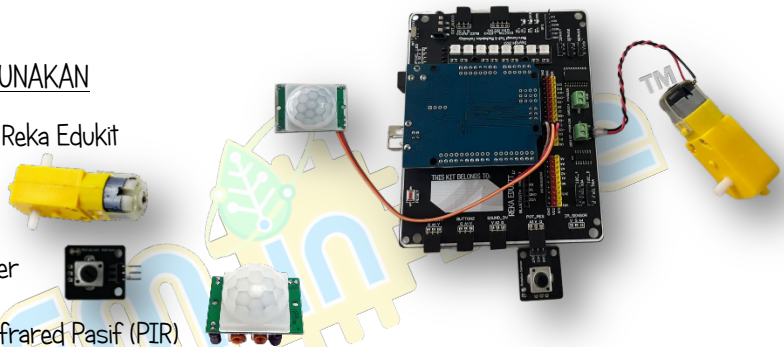
After Arduino Uno starts
  Buzzer note D5 for Sixteenth (1/16) beat
  LCD Display Address: 0x 27 Row: 0 Column: 0 Text: Jarak objek:
  LCD Display Address: 0x 27 Row: 1 Column: 4 Text: Ultrasonic Sensor distance echo pin 9 trig pin 10
  LCD Display Address: 0x 27 Row: 1 Column: 13 Text: cm
  wait 2 seconds
  LCD Clear Address: 0x 27
  wait 0.5 seconds
  
```

16: KIPAS CEKAP TENAGA

Dalam era di mana kecekapan tenaga dan kemudahan pengguna menjadi semakin penting, projek ini bertujuan untuk membangunkan sebuah sistem kipas cekap tenaga yang pintar. Dengan menggunakan gabungan motor DC, penderia infrared pasif (PIR) dan potentiometer, projek ini direka untuk meningkatkan kecekapan penggunaan tenaga dimana motor akan bergerak apabila penderia PIR mengesan pergerakan manusia dan motor akan berhenti jika tiada pergerakan dikesan manakala potentiometer digunakan untuk mengawal kelajuan motor DC.

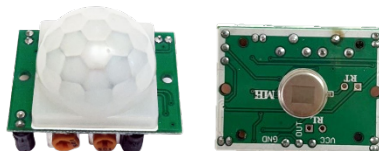
PERANTI YANG DIGUNAKAN

1. Papan Litar Reka Edukit
2. Motor DC
3. Potentiometer
4. Penderia Infrared Pasif (PIR)



PENGENALAN PENDERIA INFRARED PASIF (PIR)

Penderia Inframerah Pasif, atau Passive Infrared Sensor (PIR), adalah alat yang digunakan untuk mengesan pergerakan atau kehadiran objek yang memancarkan radiasi inframerah, seperti manusia atau haiwan. PIR sensor adalah "pasif" kerana ia tidak memancarkan tenaga dalam rangka sendiri untuk tujuan pengesanan. Sebaliknya, ia mengesan tenaga inframerah yang dipancarkan oleh objek di persekitarannya.

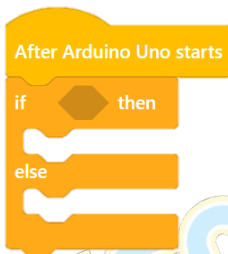


Jadual di bawah adalah penyambungan jumper wire dari papan litar Reka Edukit kepada penerima Infrared Pasif (PIR).

Pin Penerima PIR	Pin "IO Header" papan Reka Edukit
VCC	VCC
OUT	D3
GND	GND

TUTORIAL

1. Seretkan blok *After Arduino Uno starts* dan blok *if then else*.



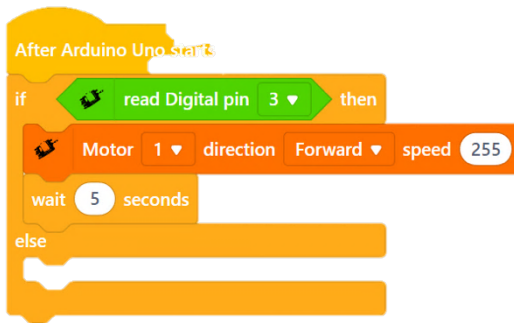
2. Seretkan blok *read Digital pin* ke dalam ruang heksagon di dalam blok *if then else*. Tetapkan pin kepada 3.



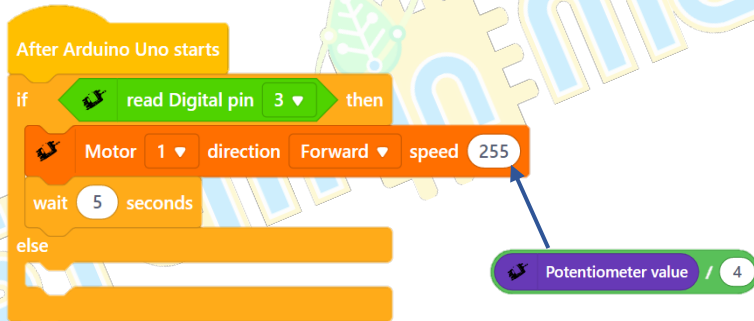
Blok ini digunakan untuk membolehkan Reka Edukit digunakan bersama penerima di luar Reka Edukit. Blok ini dipilih kerana penerima yang digunakan mengeluarkan isyarat digital.

Pastikan nombor pin yang dipilih adalah sama dengan nombor pin yang disambungkan pada penerima.

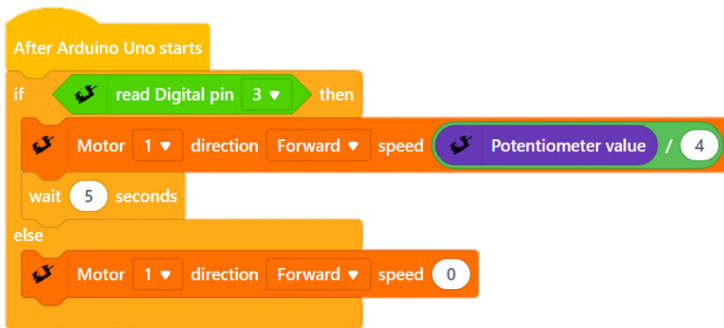
3. Seretkan blok **motor direction speed** dan tetapkan **motor** kepada 1 dan **direction** kepada forward. Kemudian seretkan blok **wait 1 second** dan sambungkan. Tetapkan kepada 5.



4. Seretkan blok **Potentiometer value** dan masukkan ke dalam blok **/**. Tetapkan nilai 4. Kemudian masukkan blok yang telah digabungkan kedalam ruangan **speed**.



5. Seretkan blok **Motor direction speed** ke dalam **else** dan tetapkan **motor** kepada 1. **direction** kepada forward dan **speed** kepada 0.

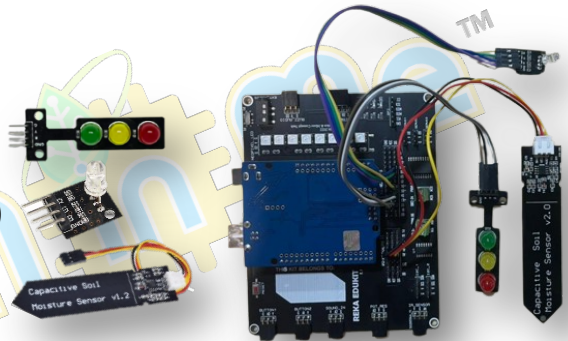


17: PEMANTAU POKOK HIASAN

Sistem "Pemantau Pokok Hiasan" mengintegrasikan sensor kelembapan tanah dengan LED RGB dan RGY, yang dihubungkan kepada mikropengawal. Mikropengawal ini memproses sensor data dan mengawal LED berdasarkan ambang kelembapan yang telah ditetapkan. Jika tanah kering, RGY LED akan menyalakan lampu merah dan RGB LED akan menyala berwarna putih. Jika sedikit lembab RGY LED kuning akan menyala dan RGB LED menyala berwarna ungu. Jika tanah mempunyai kelembapan yang mencukupi, RGY LED Hijau akan bernyala dan RGB LED akan bernyala berwarna hijau.

PERANTI YANG DIGUNAKAN

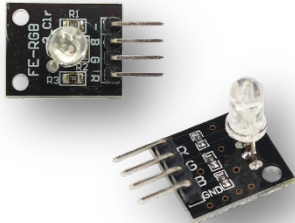
1. Papan Litar Reka Edukit
2. LED Satu Warna (GYR LED)
3. LED Pelbagai Warna (RGB LED)
4. Penderia Kelembapan Tanah



PENGENALAN LED PELBAGAI WARNA (RGB LED)

LED RGB ialah modul LED yang boleh menghasilkan hampir semua warna menggunakan tiga warna tambahan utama ini: Merah, Hijau dan Biru. Versi paling ringkas bagi LED RGB mempunyai gabungan 3 diod pemancar cahaya berasingan dalam satu pakej, ditempatkan di bawah kanta pelindung yang jelas.

Jadual di bawah adalah penyambungan jumper wire dari papan litar Reka Edukit kepada pin RGB LED.

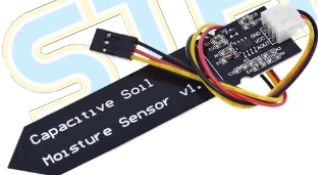


Pin RGB LED	Pin "IO Header" papan Reka Edukit
R	6
G	9
B	10
GND	GND

PENGENALAN PENDERIA KELEMBAPAN TANAH

Penderia kelembapan tanah menentukan jumlah kelembapan tanah dengan mengukur perubahan dalam kapasiti untuk menentukan kandungan air tanah. Ini boleh digunakan dalam sistem penyiraman tumbuhan automatik atau untuk memberi isyarat amaran bagi beberapa jenis apabila tumbuhan memerlukan penyiraman.

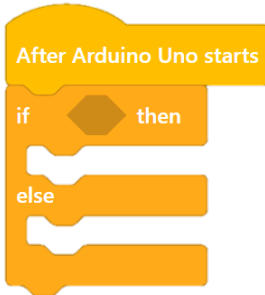
Jadual di bawah adalah penyambungan jumper wire dari papan litar Reka Edukit kepada penderia Kelembapan Tanah.



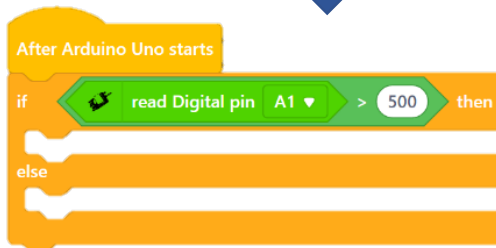
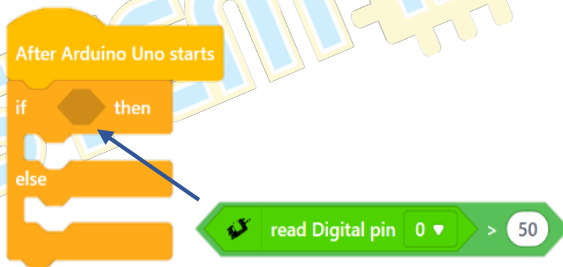
Pin Penderia Kelembapan Tanah	Pin "IO Header" papan Reka Edukit
GND	GND
VCC	VCC
AOUT	A1

TUTORIAL

1. Seretkan blok **After Arduino Uno starts** dan blok **if then else**. Kemudian sambungkan.



2. Seretkan blok **>**. Kemudian seretkan blok **read Digital pin** dan masukkan ke dalam blok **>**. Masukkan blok yang dicantumkan tadi ke dalam ruang heksagon dalam blok **if then**. Tetapkan pin kepada A1 dan nilai **>** kepada 500.



3. Seretkan 3 blok **write Digital pin** dan masukkan ke dalam blok **if then**. Tetapkan pin 2 Low untuk blok pertama diikuti 4 Low untuk blok kedua dan 5 High untuk blok ketiga.

After Arduino Uno starts

if **read Digital pin A1** > **500** then

- write Digital pin 2** Low
- write Digital pin 4** Low
- write Digital pin 5** High

else

Pastikan nombor pin yang dipilih adalah sama dengan nombor pin yang disambungkan pada penerima.

Blok ini digunakan untuk membolehkan Reka Edukit digunakan bersama penerima di luar Reka Edukit. Blok ini dipilih kerana penerima yang digunakan mengeluarkan isyarat digital.

Blok ini menunjukkan pin untuk RGY LED.
Pin 2 = Hijau
Pin 4 = Kuning
Pin 5 = Merah

Low = LED dihidupkan
High = LED dimatikan

Blok ini digunakan untuk membolehkan Reka Edukit digunakan bersama peranti keluaran di luar Reka Edukit. Blok ini dipilih kerana peranti keluaran yang digunakan perlu membaca isyarat digital.

4. Seretkan 3 blok **write PWM pin** dan tetapkan pin kepada 6, 9 dan 10. Kemudian seretkan blok **Buzzer tone** dan tetapkan kepada 500.

After Arduino Uno starts

if **read Digital pin A1** > **500** then

- write PWM pin 6** 255
- write PWM pin 9** 255
- write PWM pin 10** 255
- Buzzer tone 500**

else

Blok ini menunjukkan pin untuk RGB LED.
Pin 6 = Merah
Pin 9 = Hijau
Pin 10 = Biru

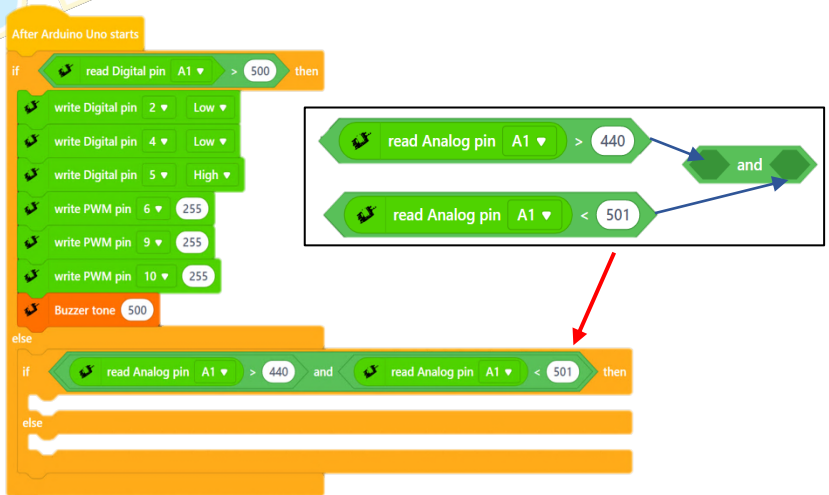
Low = LED dihidupkan
High = LED dimatikan

Blok ini digunakan untuk membolehkan Reka Edukit digunakan bersama peranti keluaran di luar Reka Edukit. Blok ini dipilih kerana peranti keluaran yang digunakan boleh membaca julat dari nilai 0 sehingga 255 dengan menggunakan teknik PWM (Pulse Width Modulation).

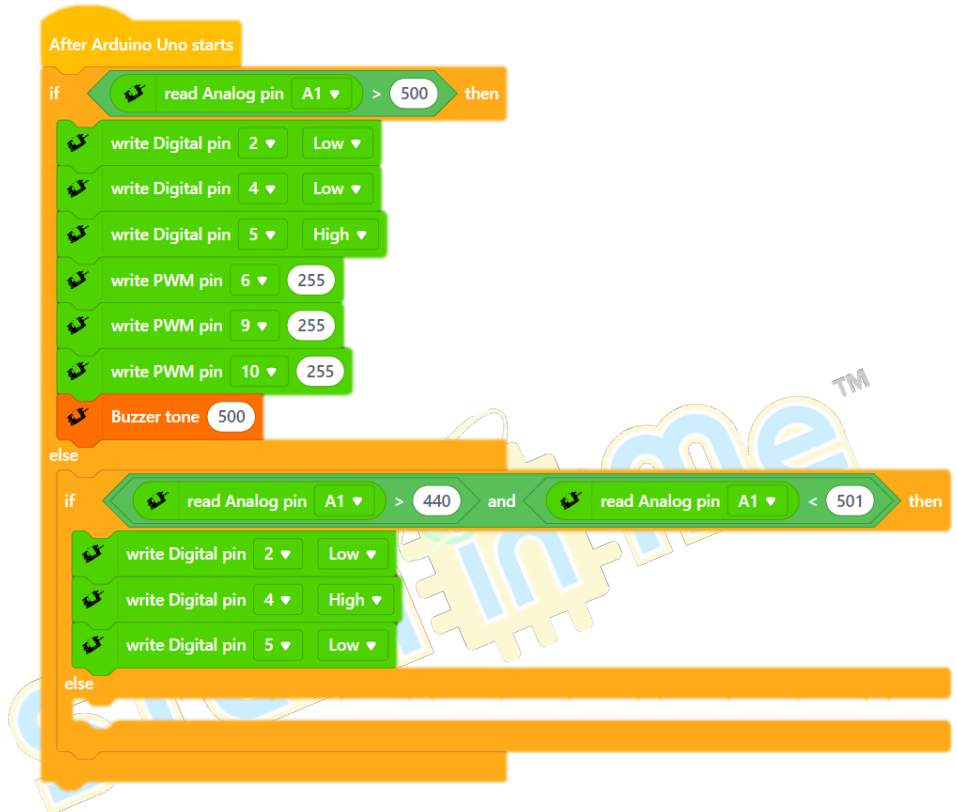
5. Seretkan blok **if then else** dan masukkan ke dalam ruang **else**.



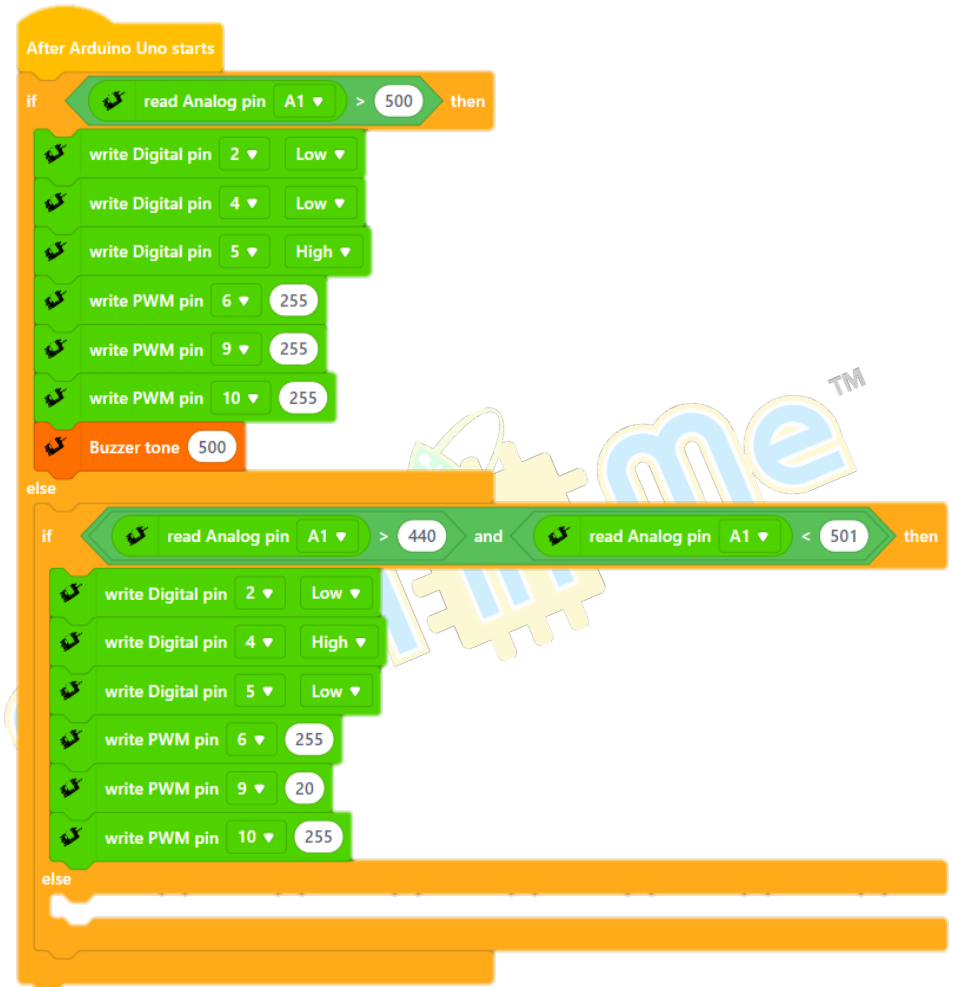
6. Seretkan blok **>** dan **<**. Kemudian seretkan 2 blok **read Analog pin** dan masukkan kedalam blok **>** dan **<**. Tetapkan pin kepada A1. nilai blok **>** kepada 440 dan nilai blok **<** kepada 501. Masukkan kedua-dua blok tadi ke dalam blok **and**. Masukkan blok tadi ke dalam ruang heksagon di dalam blok **if then**.



7. Seretkan 3 blok **write Digital pin** ke dalam ruang **if then** dan tetapkan pin kepada 2, 4, 5 dan Low, High, Low.



8. Seretkan 3 blok write PWM pin dan tetapkan pin kepada 6, 9, 10 dan 255, 20, 255.



9. Seretkan 3 blok **write Digital pin** ke dalam ruang **else** dan tetapkan pin kepada 2, 4, 5 dan High, Low, Low.

After Arduino Uno starts

```

if (read Analog pin A1 > 500) then
  write Digital pin 2 Low
  write Digital pin 4 Low
  write Digital pin 5 High
  write PWM pin 6 255
  write PWM pin 9 255
  write PWM pin 10 255
  Buzzer tone 500
else
  if (read Analog pin A1 > 440 and read Analog pin A1 < 501) then
    write Digital pin 2 Low
    write Digital pin 4 High
    write Digital pin 5 Low
    write PWM pin 6 255
    write PWM pin 9 20
    write PWM pin 10 255
  else
    write Digital pin 2 High
    write Digital pin 4 Low
    write Digital pin 5 Low
  
```

10. Seretkan 3 blok write PWM pin dan tetapkan pin kepada 6, 9, 10 dan 255, 20, 255.



11. Seretkan blok **Buzzer off** dibawah blok **if then else** dan blok **wait 1 seconds** di bahagian bawah sekali. Tetapkan nilai kepada 0.5.

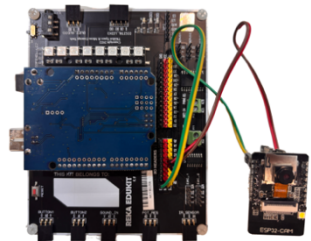


18: PENGECEKAM BAHAN KITAR SEMULA AUTOMATIK

Memisahkan bahan kitar semula secara manual mengambil masa yang lama dan sering berlaku kesilapan. Ini membuatkan proses kitar semula kurang efektif dan tidak efisien. Dengan menggunakan kamera AI, proses pengecaman dan pengasingan bahan kitar semula boleh dilakukan secara automatik. Kamera ini boleh mengenal pasti jenis bahan seperti plastik, kertas, logam, dan kaca dengan cepat dan tepat. Teknologi ini membantu mempercepatkan proses kitar semula, mengurangkan kesilapan, dan mengurangkan keperluan tenaga kerja manusia. Hasilnya, lebih banyak bahan dapat dikitar semula dengan cara yang lebih baik untuk alam sekitar.

PERANTI YANG DIGUNAKAN

1. Papan Litar Reka Edukit
2. LED Pelbagai Warna (Neopixel LED)
3. ESP32-CAM



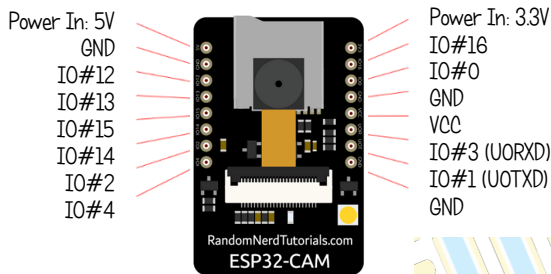
Pengenalan ESP32-Camera dan Mekanismenya

ESP32-CAM terdiri daripada 2 papan yang boleh dipisahkan iaitu modul kamera kecil yang menggunakan mikrokontroler ESP32 dan papan pengaturcaraan. Ia mempunyai Wi-Fi, Bluetooth, dan kamera OV2640 dengan resolusi 2MP. Modul ini sesuai untuk aplikasi seperti pemantauan, penstriman video, dan IoT. Dengan slot MicroSD, ia boleh menyimpan imej dan video secara tempatan. ESP32-CAM boleh diprogramkan menggunakan Arduino IDE dan digunakan untuk pengesanan wajah, keselamatan, atau robotik. Sambungan Wi-Fi membolehkan pemindahan data jarak jauh, dan GPIO-nya boleh mengawal peranti lain. Ia memerlukan bekalan kuasa 5V dan mempunyai fungsi pemprosesan data secara dalaman.



Penyambungan ESP32-CAM – Modul Kamera pada papan Reka Edukit

Pin "IO Header" papan Reka Edukit	Pin ESP32-CAM Modul Kamera
GND	GND
VCC	5V
A0	IO#3 (U0RXD)
A1	IO#1 (U0TXD)



Gambar 1: Pin pada ESP32-CAM

Nota: ESP32-CAM terdiri daripada dua komponen: Modul Kamera dan Papan Pengaturcaraan. Modul kamera perlu dipisahkan daripada Papan Pengaturcaraan untuk disambungkan papan Reka Edukit



Modul Kamera

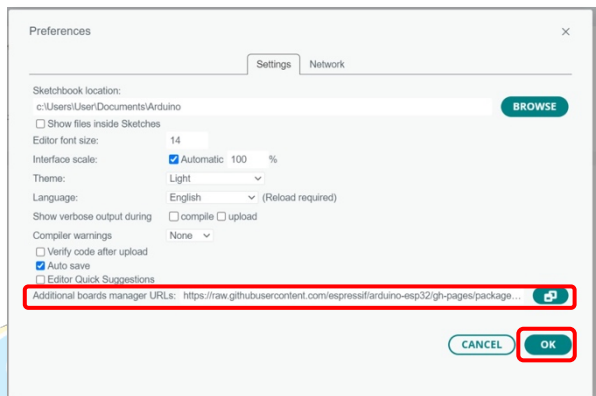
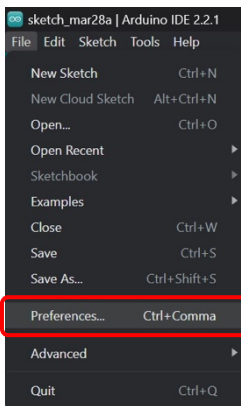


Papan Pengaturcaraan

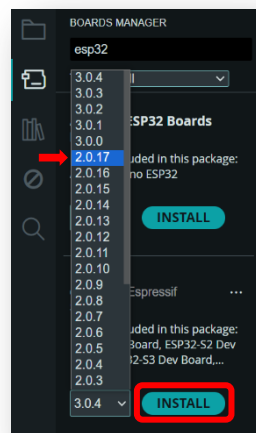
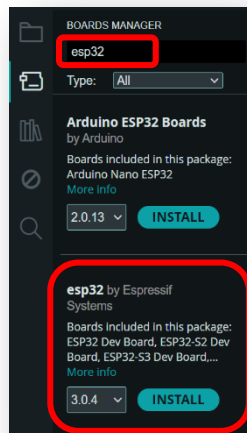
Menambah peranti ESP32-CAM pada Arduino IDE

Nota: Sila pastikan ESP32-CAM – Modul Kamera dipasangkan kepada ESP32-CAM – Papan Pengaturcaraan

Langkah 1 Buka Arduino IDE dan klik File di bahagian atas kiri. Cari dan klik Preferences. Di bahagian Additional Boards Manager URLs masukkan URL: https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json dan klik OK



Langkah 2 Buka Arduino IDE dan klik simbol fail di bahagian kiri aplikasi dan cari "esp32" seperti yang ditunjukkan. Pilih *esp32 by Expressif Systems*. Pastikan versi yang dipilih adalah 2.0.17 dan tekan install.

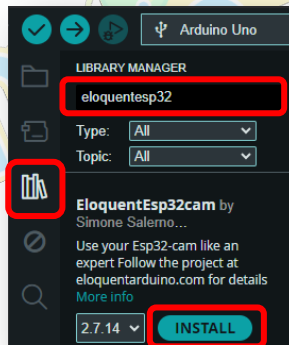


Pengumpulan Data menggunakan ESP32-CAM

Untuk melatih model AI dengan berkesan, data yang berkualiti dan relevan adalah diperlukan. Data yang diperlukan adalah dalam bentuk imej yang ditangkap oleh ESP32-CAM. Model AI memerlukan pelbagai contoh data untuk belajar mengenali corak dan berfungsi dengan baik. Dengan mengumpul data juga membolehkan kita menguji dan memperbaiki model AI berdasarkan data sebenar yang dikumpul untuk meningkatkan prestasi model. Untuk pengumpulan data, hanya ESP32-CAM dan arduino IDE yang digunakan. Tidak perlu menyambungkan ESP32-CAM pada papan Reka Edukit.

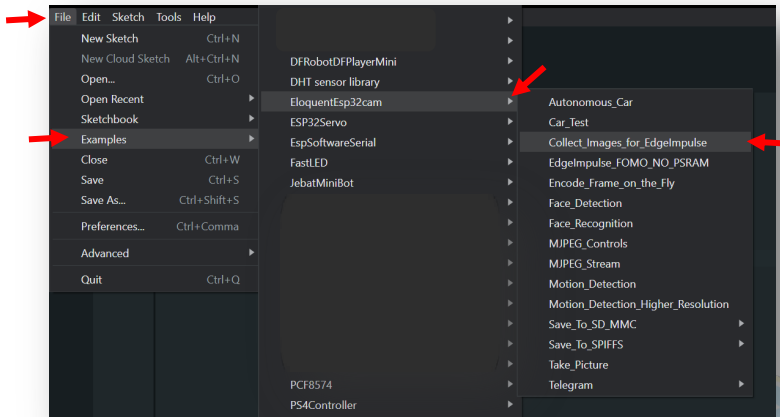
Nota: Sila pastikan ESP32-CAM – Modul Kamera dipasangkan kepada ESP32-CAM – Papan Pengaturcaraan. Sambungkan ESP32-CAM (Modul Kamera dan Papan Pengaturcaraan) kepada laptop/PC anda untuk pengumpulan data.

Langkah 1 Buka Arduino IDE dan klik simbol buku pada bahagian kiri aplikasi. Kemudian taip "eloquentesp32" pada ruangan *search*. Pastikan library yang muncul adalah *EloquentEsp32cam by Simone Salerno*. Kemudian tekan install.



Langkah 2

Klik pada menu *file* di bahagian kiri atas aplikasi. Kemudian pilih examples. Cari pilihan EloquentEsp32cam dan tekan pada menu Collect_Images_for_EdgeImpulse.



Langkah 3

Kemudian pada coding yang dipaparkan, tukarkan Wifi SSID dan PASS pada line 15 dan 16 kepada nama dan password wifi yang sedang digunakan. Tukar "wroom_s3" kepada "aithinker" pada line 36. Ia ditukar mengikut jenis model camera yang digunakan.

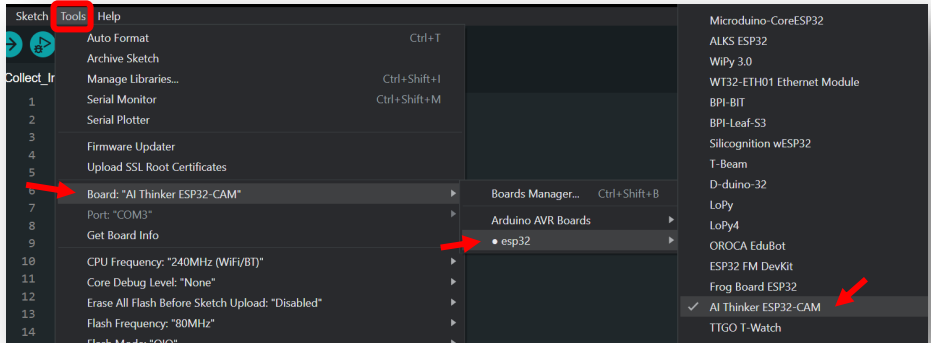
```
Collect_Images_for_EdgeImpulse.ino
13 // the camera at http://{HOSTNAME}.local
14
15 #define WIFI_SSID "SSID"
16 #define WIFI_PASS "PASSWORD"
17 #define HOSTNAME "esp32cam"
18
19
20 #include <eloquent_esp32cam.h>
21 #include <eloquent_esp32cam/extra/esp32/wifi/sta.h>
22 #include <eloquent_esp32cam/viz/image_collection.h>
23
24 using eloq::camera;
25 using eloq::wifi;
26 using eloq::viz::collectionServer;
27
28
29 void setup() {
30     delay(3000);
31     Serial.begin(115200);
32     Serial.println("__IMAGE COLLECTION SERVER__");
33
34     // camera settings
35     // replace with your own model!
36     camera.pinout.wroom_s3();
37     camera.brownout.disable();
38     // Edge Impulse models work on square images
39     // face resolution is 240x240
40     camera.resolution.face();
41     camera.quality.high();
42 }
```



```
Collect_Images_for_EdgeImpulse.ino
13 // the camera at http://{HOSTNAME}.local
14
15 #define WIFI_SSID "StemInMe"
16 #define WIFI_PASS "1234567890"
17 #define HOSTNAME "esp32cam"
18
19
20 #include <eloquent_esp32cam.h>
21 #include <eloquent_esp32cam/extra/esp32/wifi/sta.h>
22 #include <eloquent_esp32cam/viz/image_collection.h>
23
24 using eloq::camera;
25 using eloq::wifi;
26 using eloq::viz::collectionServer;
27
28
29 void setup() {
30     delay(3000);
31     Serial.begin(115200);
32     Serial.println("__IMAGE COLLECTION SERVER__");
33
34     // camera settings
35     // replace with your own model!
36     camera.pinout.aithinker();
37     camera.brownout.disable();
38     // Edge Impulse models work on square images
39     // face resolution is 240x240
40     camera.resolution.face();
41     camera.quality.high();
42 }
```

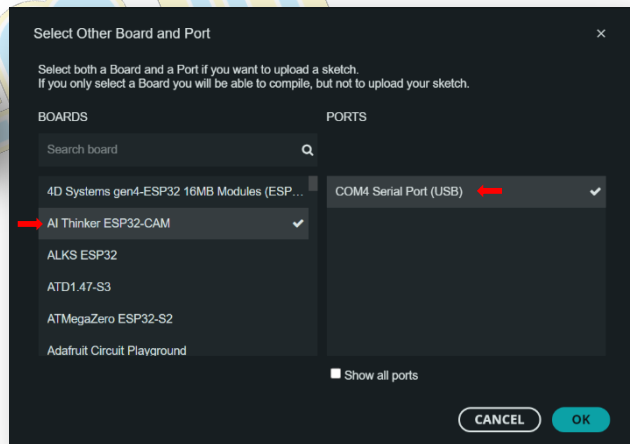
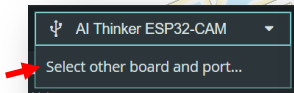
Langkah 4

Klik pada menu Tools dan pilih menu Board. Kemudian tekan pada esp32 dan cari AI Thinker ESP32-CAM



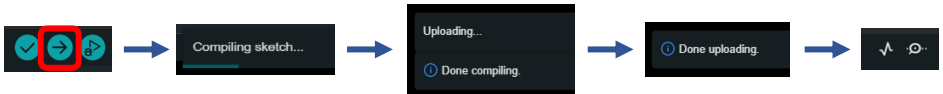
Langkah 5

Tekan pada *AI Thinker ESP32-CAM* dan pilih *select other board and port*. Pastikan *board* dan *port* dipilih dengan betul. Kemudian klik ok.



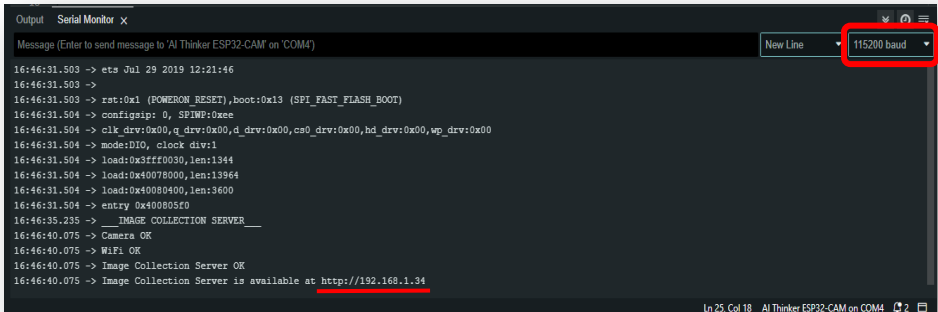
Langkah 6

Tekan pada simbol *arrow* di bahagian kiri atas untuk *compile* dan *upload*. Setelah selesai *upload*, tekan pada simbol kanta pembesar di bahagian kanan atas untuk membuka *serial monitor*.



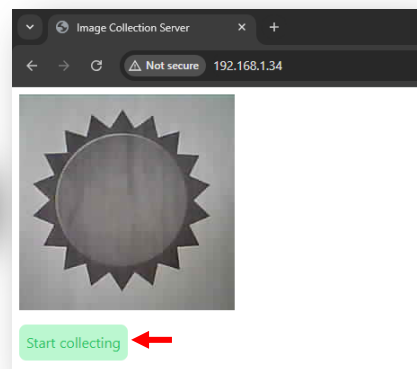
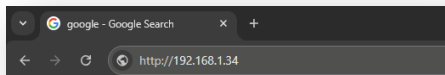
Langkah 7

Pastikan nilai baud yang ditetapkan adalah 115200 baud. Kemudian pada paparan serial monitor, salin alamat IP seperti yang ditunjukkan.



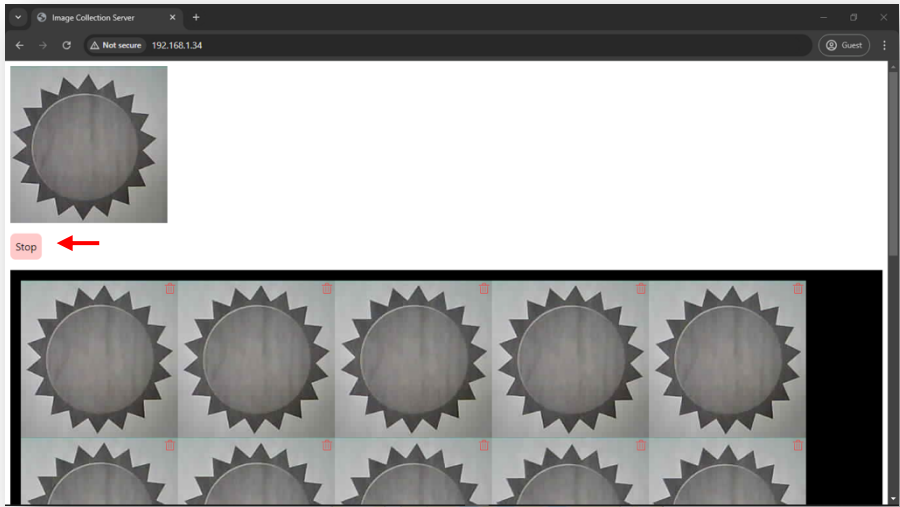
Langkah 8

Tampil alamat IP pada carian *Google Search* dan laman *Image Collection Server* akan muncul. Halakan kamera pada data imej yang ingin dikumpul seperti yang ditunjukkan dan tekan *Start collecting*.



Langkah 9

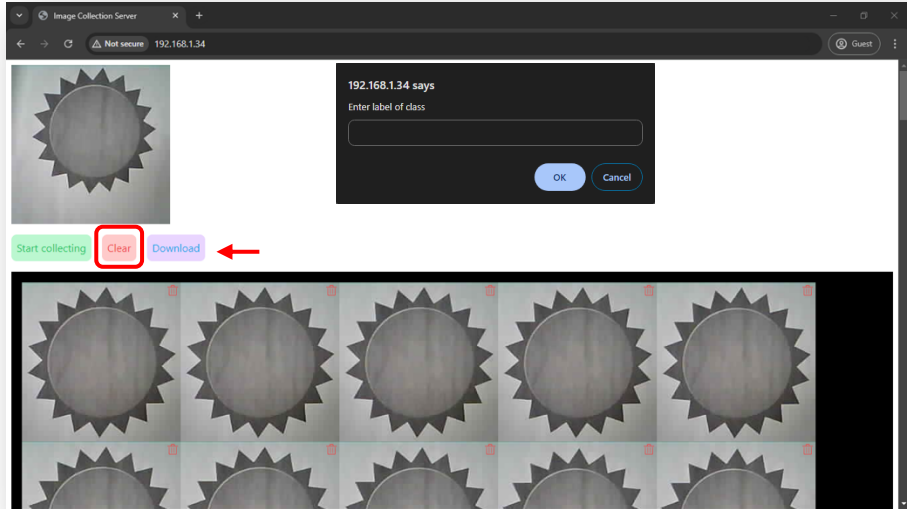
Data imej yang dikumpul akan dipaparkan pada skrin. Lebih banyak data imej yang dikumpul, lebih tepat ia berfungsi. setelah cukup data imej yang ingin dikumpul, tekan *Stop*.



STEM

Langkah 10

Kemudian tekan *Download* untuk muat turun semua data imej yang telah dikumpulkan. Masukkan nama kelas sebagai 'SUN' untuk data imej dan tekan *Ok* Untuk mengumpul data imej berlainan kelas. tekan *Clear* untuk mengosongkan data lama dan ulang dari Langkah 8.



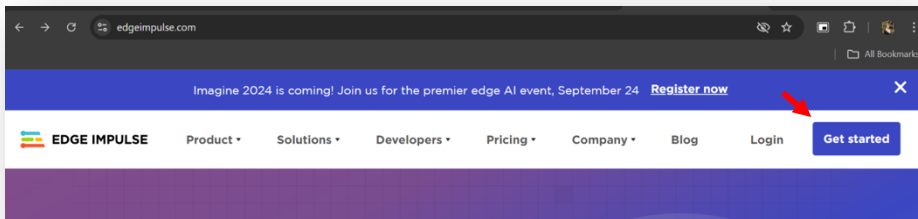
PERHATIAN!

Sila pastikan nama kelas objek yang dimasukkan di dalam Langkah 10 adalah SAMA EJAAN dan HURUF BESAR/KECIL dengan *Object Class* yang digunakan di dalam *Tutorial* di muka surat 170 - 175

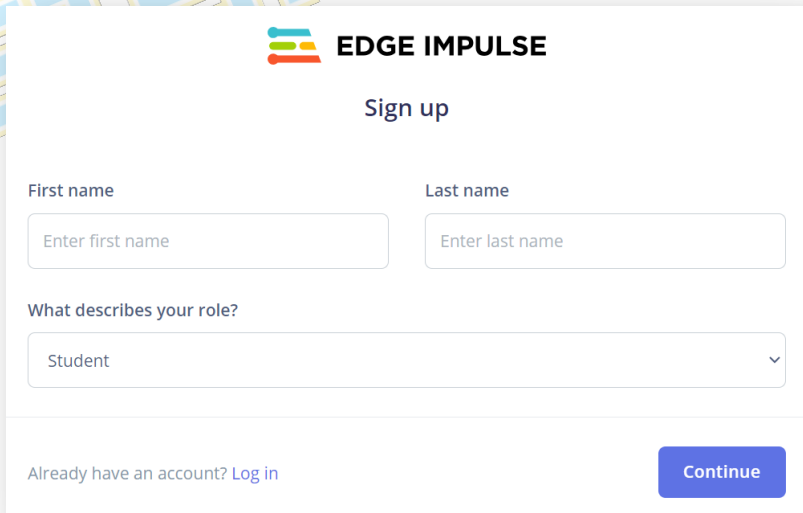
Penggunaan EdgeImpulse untuk Membangun Model Kecerdasan Buatan (AI)

Edge Impulse adalah platform yang memudahkan pembangun dalam mencipta dan menggunakan model kecerdasan buatan (AI) untuk peranti kecil seperti sensor, mikropengawal, dan kamera. Platform ini menyokong pengumpulan data, latihan model AI, dan penggunaan model pada peranti dengan kuasa rendah. Ia amat berguna untuk aplikasi peranti pintar dan projek Internet of Things (IoT).

Langkah 1 Cari *Edge Impulse* pada carian *Google Search* dan tekan *Get Started*.



Langkah 2 Masukkan nama dan pastikan *role* yang ditetapkan adalah *Student*.



Langkah 3

Kemudian masukkan maklumat yang diperlukan dan klik *Sign up*. Setelah muncul *Sign up successful*. Tekan pada *Click here to build your first ML model!*

Sign up

First name

Last name

Email

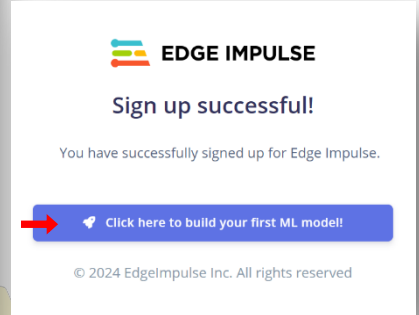
Username

Password

☐ I accept the Privacy Policy, Community Terms of Service, and Responsible AI License.

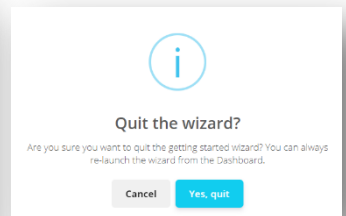
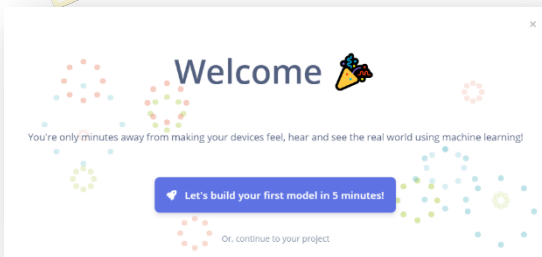
Already have an account? [Log in](#)

Sign up

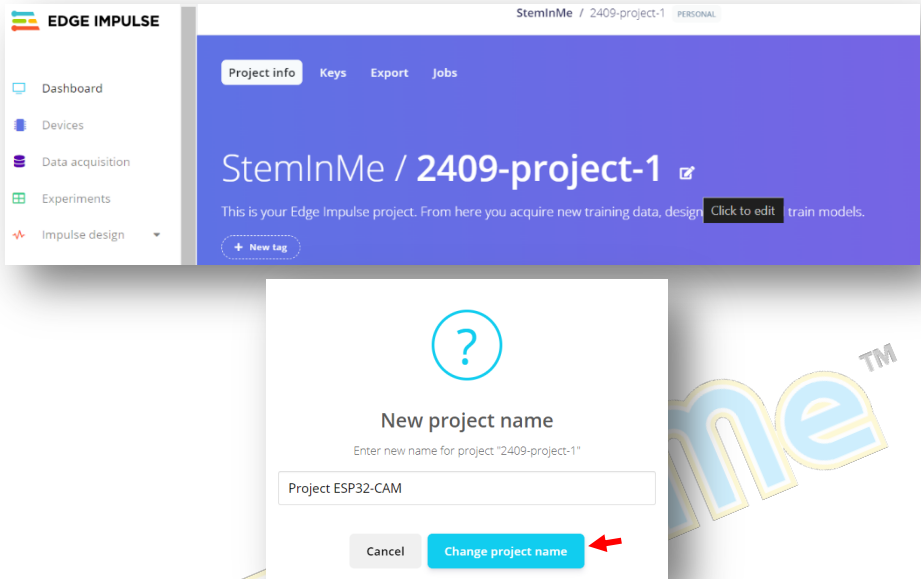


Langkah 4

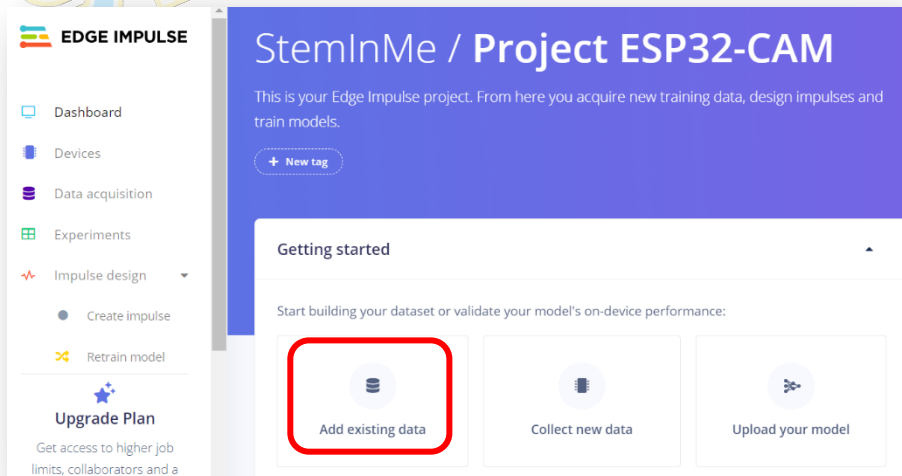
Tekan *continue to your project*. Setelah muncul *Quit the wizard?* Tekan *Yes, quit*.



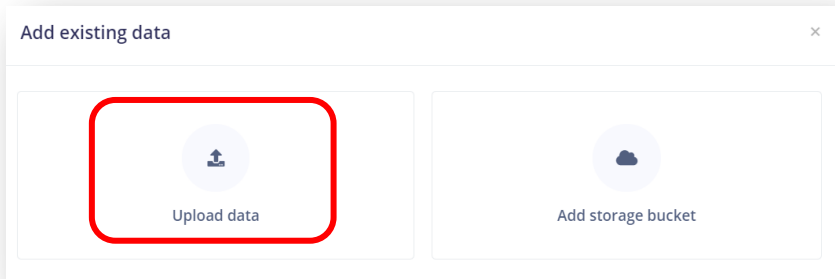
Langkah 5 Pada *dashboard* Edge Impulse, tekan logo pensel untuk mengubah nama proyek. Kemudian namakan proyek seperti yang ditunjukkan dan tekan *change project name*.



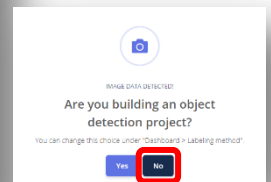
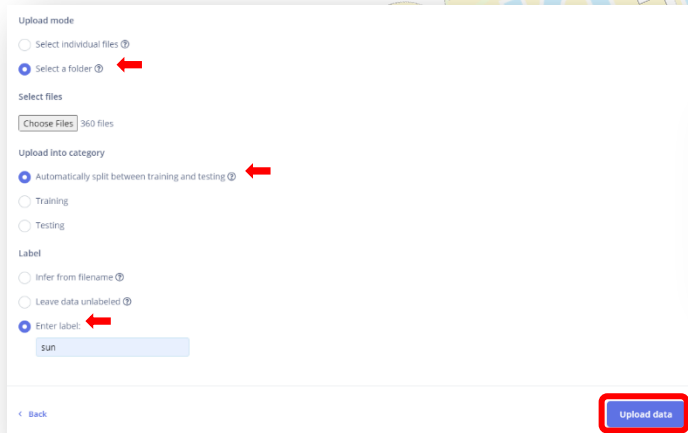
Langkah 6 Kemudian pada *dashboard*, tekan pada *Add existing data* untuk menambahkan imej data yang telah dikumpulkan sebelum ini.



Langkah 7 Tekan pada *Upload data*.



Langkah 8 Pastikan menu yang dipilih seperti yang ditunjukkan di bawah. Di bahagian *Enter label*, nama yang dimasukkan perlu mewakili setiap kelas data dan klik *Upload data*. Kemudian tekan *No* pada pada soalan *Are you building an object detection project?*



Langkah 9

Setelah imej data berjaya dimuat naik. *Job completed* akan dipaparkan. Ulang langkah 8 untuk setiap kelas data seterusnya. Pastikan nama label diubah mengikut nama kelas.

Upload mode

☐ Select individual files ⑦

☒ Select a folder ⑦

Select files

No file chosen

Upload into category

☒ Automatically split between training and testing ⑦

☐ Training

☐ Testing

Label

☐ Infer from filename ⑦

☐ Leave data unlabeled ⑦

☒ Enter label:

water

Upload output

```
[343/360] uploading sun_1721880317743.jpg OK
[344/360] uploading sun_1721880316250.jpg OK
[345/360] uploading sun_1721880318746.jpg OK
[346/360] uploading sun_1721880317256.jpg OK
[347/360] uploading sun_1721880316497.jpg OK
[348/360] uploading sun_1721880317495.jpg OK
[349/360] uploading sun_1721880318487.jpg OK
[350/360] uploading sun_1721880317992.jpg OK
[351/360] uploading sun_1721880316999.jpg OK
[352/360] uploading sun_1721880318249.jpg OK
[353/360] uploading sun_1721880318992.jpg OK
[354/360] uploading sun_1721880315242.jpg OK
[355/360] uploading sun_1721880319756.jpg OK
[356/360] uploading sun_1721880319493.jpg OK
[357/360] uploading sun_1721880319993.jpg OK
[358/360] uploading sun_1721880320243.jpg OK
[359/360] uploading sun_1721880320582.jpg OK
[360/360] uploading sun_1721880320745.jpg OK
```

Done. Files uploaded successful: 359. Files that failed to upload: 1.

Job completed

Langkah 10

Di bahagian kanan atas, klik pada *target* seperti yang ditunjukkan. Kemudian tetapkan *target device* kepada *Espressif ESP-EYE (ESP32 240MHz)* dan tekan *save*.

Target: Cortex-M4F 80MHz

S

Configure your target device and application budget

Target device

Define your target device requirements to inform model optimizations and performance calculations. No device yet? Use the default settings which you can change at any time.

Target device

Espressif ESP-EYE (ESP32 240MHz)

Processor family

ESP32

Clock rate ⑦

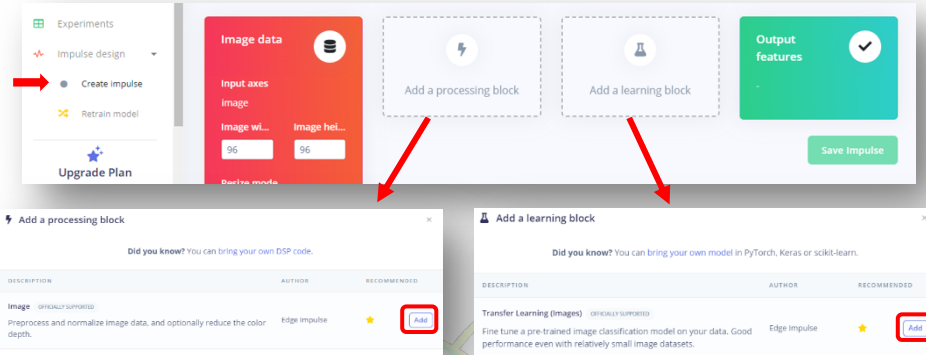
240 MHz

Max

Custom device name (optional) ⑦

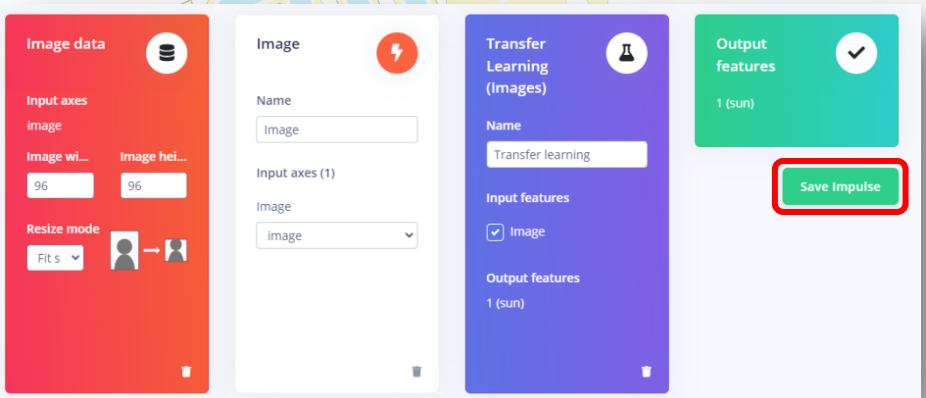
Langkah 11

Pada menu sebelah kiri, klik pada *create impulse* di dalam *impulse design*. Kemudian klik pada *Add a processing block* pilih *Image* dan klik *Add*. Seterusnya klik pada *Add a learning block* pilih *Transfer Learning (Images)* dan klik *Add*.

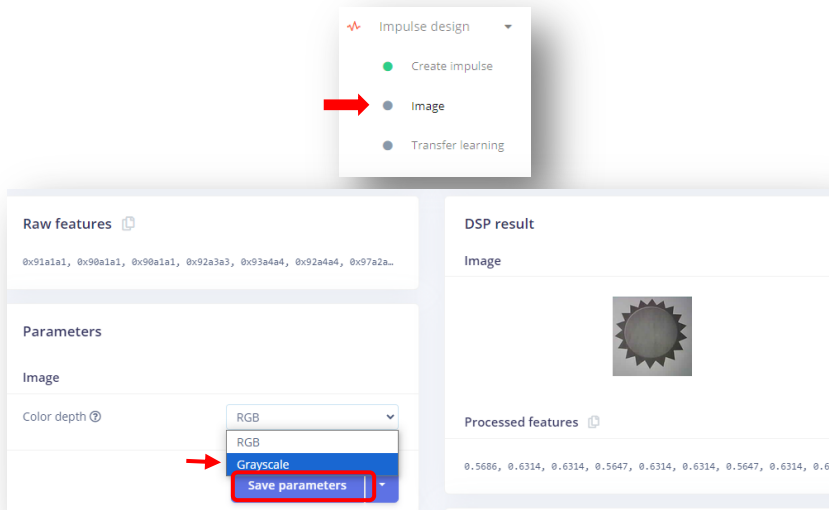


Langkah 12

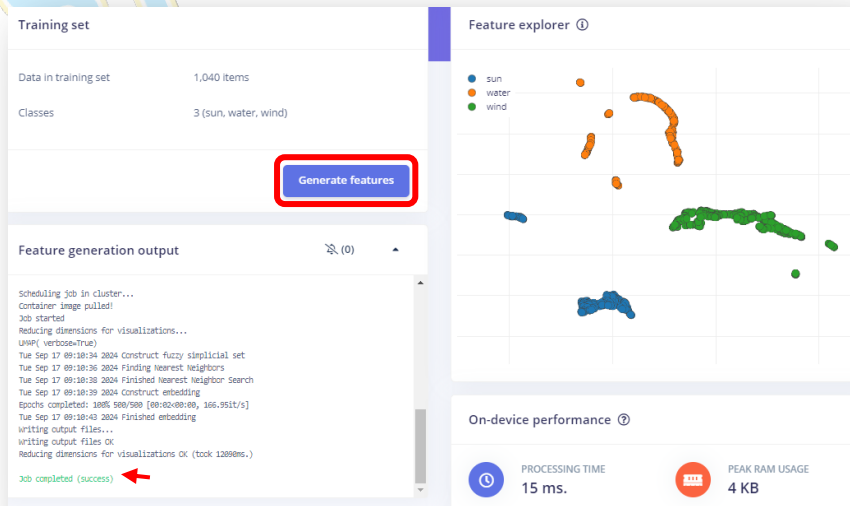
Klik pada *Save Impulse*.



Langkah 13 Tekan pada menu *Image* dan tukar *color depth* kepada *grayscale*. Ia bergantung kepada imej data yang dikumpul. Kemudian klik *Save parameters*. Imej data akan dipaparkan pada *DSP result*.



Langkah 14 Kemudian halaman *Training set* akan muncul. Tekan *Generate features* dan *Job completed (success)* akan muncul setelah selesai seperti yang ditunjukkan di bawah.



Langkah 15 Pilih menu *Transfer learning* kemudian klik *Save & train* pada bahagian *Neural Network settings*. Data akan muncul seperti yang ditunjukkan.

Impulse design

- Create impulse
- Image
- Transfer learning**
- Retrain model

Save & train

Training output

Calculating performance metrics...
Calculating inferencing time...
INFO: Created TensorFlow Lite XNNPACK delegate for CPU.
Calculating inferencing time OK
Calculating float32 accuracy...
Calculating inferencing time OK
Calculating float32 accuracy...
INFO: Created TensorFlow Lite XNNPACK delegate for CPU.
Calculating int8 accuracy...

Model training complete
Model training complete
Job completed (success)

Model
Model version: Quantized (int8)

Last training performance (validation set)

ACCURACY
100.0%

LOSS
0.00

Confusion matrix (validation set)

	SUN	WATER	WIND
SUN	100%	0%	0%
WATER	0%	100%	0%
WIND	0%	0%	100%
F1 SCORE	1.00	1.00	1.00

Metrics (validation set)

METRIC	VALUE
Area under ROC Curve	1.00
Weighted average Precision	1.00
Weighted average Recall	1.00
Weighted average F1 score	1.00

Data explorer (full training set)

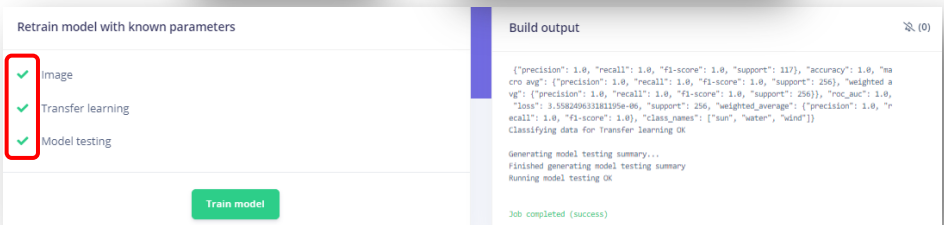
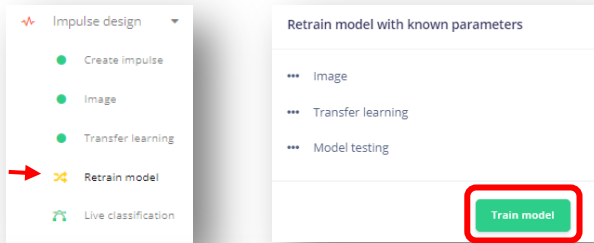
On-device performance
Engine: EON Compiler

INFERENCING TIME
1690 ms.

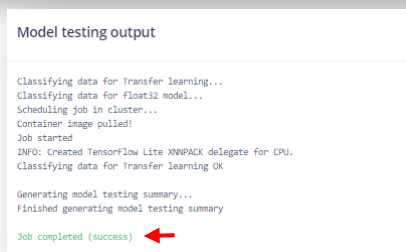
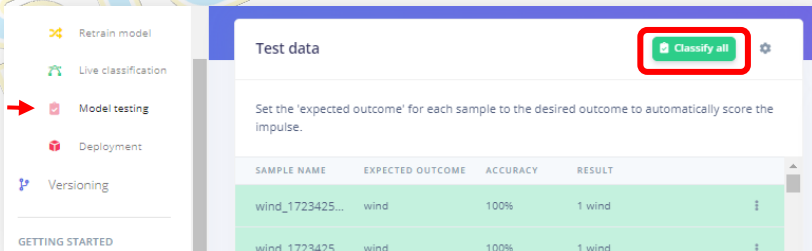
PEAK RAM USAGE
334.6K

FLASH USAGE
585.2K

Langkah 16 Pilih menu *Retrain model* dan klik pada *Train model*. Kemudian *Build output* akan memaparkan *Job completed (success)* dan semua parameter akan bertanda ✓.



Langkah 17 Klik pada menu *Model testing* dan tekan *Classify all* pada bahagian *Test data*. *Job completed (success)* akan dipaparkan pada *Model testing output*.



- Langkah 18** Klik pada menu *Deployment* dan pastikan *configure your deployment* adalah jenis *Arduino library*. kemudian pastikan pilihan model seperti yang ditunjukkan di bawah dan tekan *build*.

Retrain model
Live classification
Model testing
Deployment
Versioning

Configure your deployment

You can deploy your impulse to any device. This makes the model run without an internet connection, minimizes latency, and runs with minimal power consumption. [Read more.](#)

Search deployment options

C++ library
A portable C++ library with no external dependencies, which can be compiled with any modern C++ compiler.

Arduino library
An Arduino library with examples that runs on most Arm-based Arduino development boards.

MODEL OPTIMIZATIONS

Model optimizations can increase on-device performance but may reduce accuracy.

EON™ Compiler
Same accuracy, 17% less RAM, 14% less ROM.

Quantized (int8)

Selected

	IMAGE	TRANSFER LEARNING	TOTAL
LATENCY	15 ms.	1,690 ms.	1,705 ms.
RAM	4.0K	334.6K	334.6K
FLASH	-	585.2K	-
ACCURACY			-

Unoptimized (float32)

Select

	IMAGE	TRANSFER LEARNING	TOTAL
LATENCY	15 ms.	4,107 ms.	4,122 ms.
RAM	4.0K	893.7K	893.7K
FLASH	-	1.6M	-
ACCURACY			100.00%

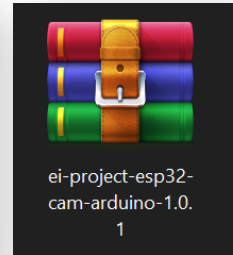
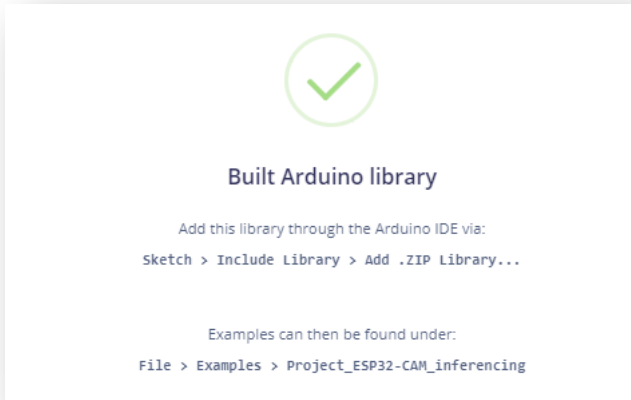
To compare model accuracy, run model testing for all available optimizations. **Run model testing**

Estimate for Espressif ESP-EYE (ESP32 240MHz) - [Change target](#)

Build

Langkah 19

Paparan seperti di bawah akan muncul dan fail Model Kecerdasan Buatan (AI) akan disimpan dalam *Download*.



Fail Model
Kecerdasan Buatan
(AI)

STEM in me

Muat turun data ke dalam ESP32-CAM

Nota: Sila pastikan modul kamera disambungkan kepada papan pengaturcaraan. Sambungkan ESP32-CAM kepada laptop/PC anda untuk muat turun data

Langkah 1

Klik pada laman: <https://www.microconcept.com.my/download/> dan cari dan klik *Reka Edukit ESP32CAM Arduino Edge Impulse v1* untuk muat turun templat pengaturcaraan. Kemudian *Extract files*. Buka folder dan klik pada fail template pengaturcaraan *RekaEdukit_ESP32CAM_EdgeImpulse_v1.ino*. Buka coding arduino didalamnya.

We offer a vibrant bilingual experience in English and Malay.

- Two Languages, Your Choice!

Reka Edukit Modul eBook v1.2 English

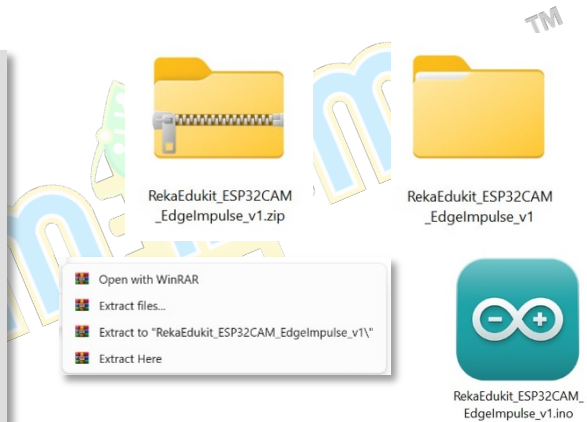
Reka Edukit Modul eBook v1.6 Bahasa Malaysia

- Dive deeper into learning with our bonus content!

Reka Edukit mBlock Project v7

Reka Edukit mBlock Troubleshooting v2

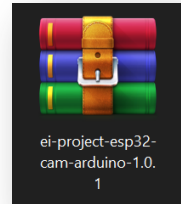
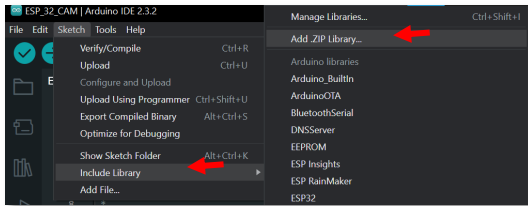
Reka Edukit ESP32CAM Arduino Edge Impulse v1



Templat
Pengaturcaraan

Langkah 2

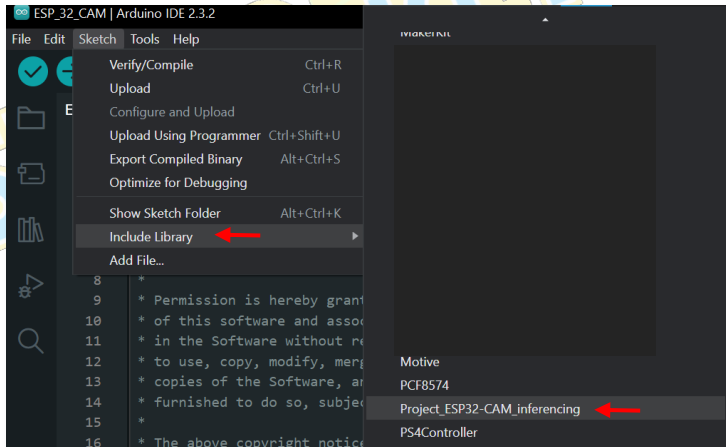
Klik pada menu *Sketch*, pilih *include Library* dan *Add .ZIP Library...*.
Tambahkan fail Model Kecerdasan Buatan (AI) yang telah dimuat turun.
Library installed akan dipaparkan pada bahagian *Output*.



Fail Model
Kecerdasan Buatan
(AI)

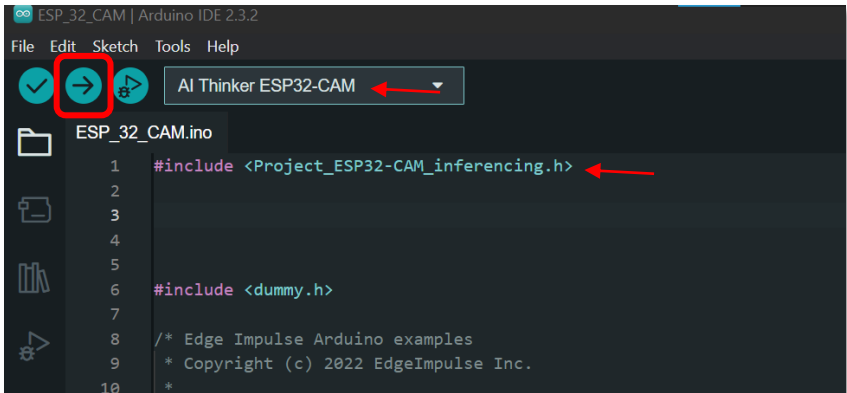
Langkah 3

Klik pada menu *Sketch*, pilih *Include Library* dan pilih *Project_ESP32-CAM_inferencing*.



Langkah 4

Nama library akan muncul pada bahagian atas coding. Kemudian sambungkan ESP32-CAM dan klik pada ikon anak panah. Pastikan *board* yang dipilih adalah *AI Thinker ESP32-CAM* dan port dipilih dengan betul.

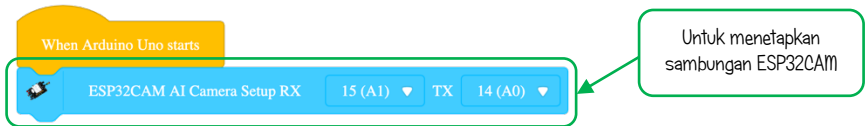


STEM in me

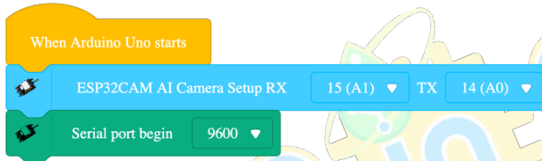
TUTORIAL

Nota: Sila pastikan modul kamera sahaja disambungkan kepada papan Reka Edukit. (Rujuk muka surat 153 untuk sambungan)

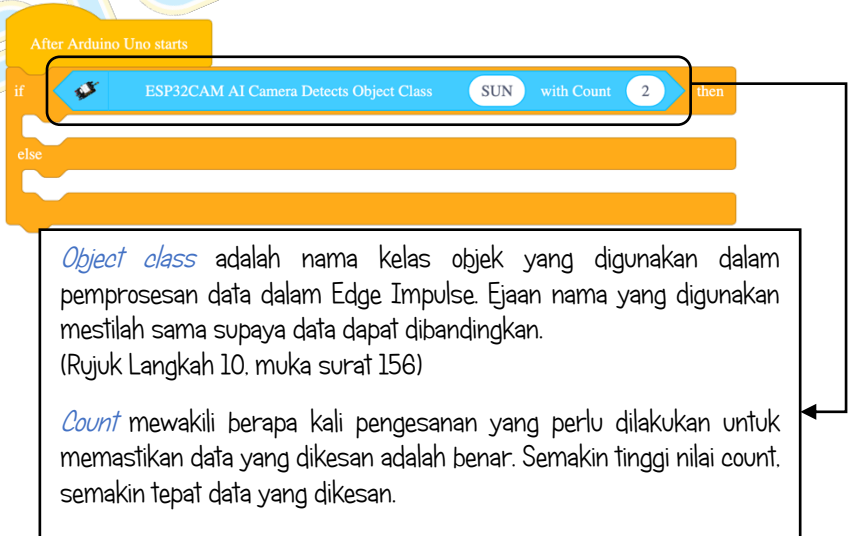
1. Seretkan blok *When Arduino Uno starts* dan sambungkan dengan blok *ESP32CAM AI Camera Setup*. Setkan *RX* kepada *15 (A1)* dan *TX* kepada *14 (A0)*



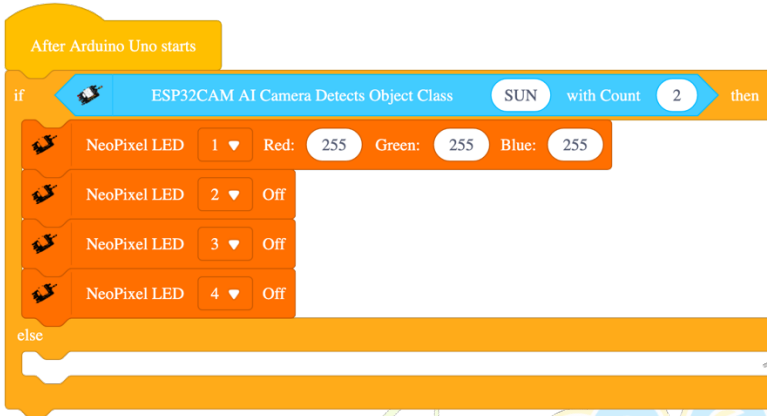
2. Kemudian sambungkan blok *Serial port begin* dan tukarkan nilai kepada *9600*



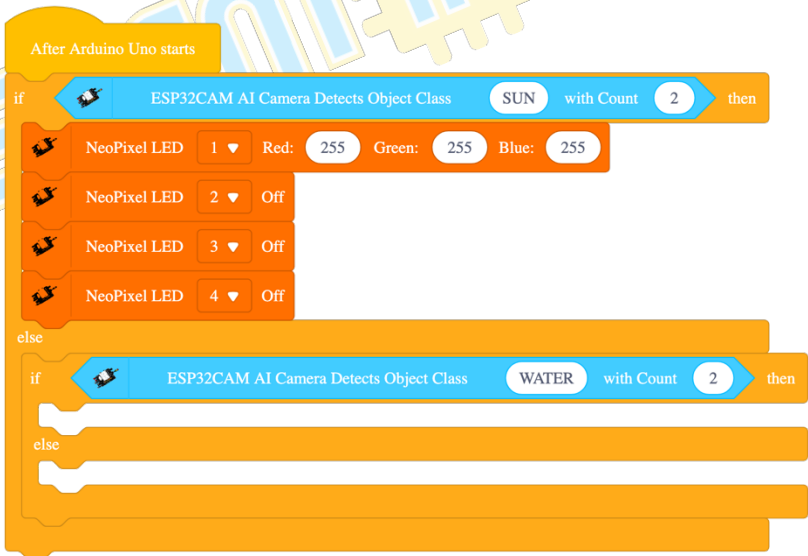
3. Seretkan blok *After Arduino Uno starts* dan blok *if then else*. Kemudian, seretkan blok *ESP32CAM AI Camera Detect* ke dalam blok *if then* tukarkan *Object Class* kepada *SUN* dan *Count* kepada *2*.



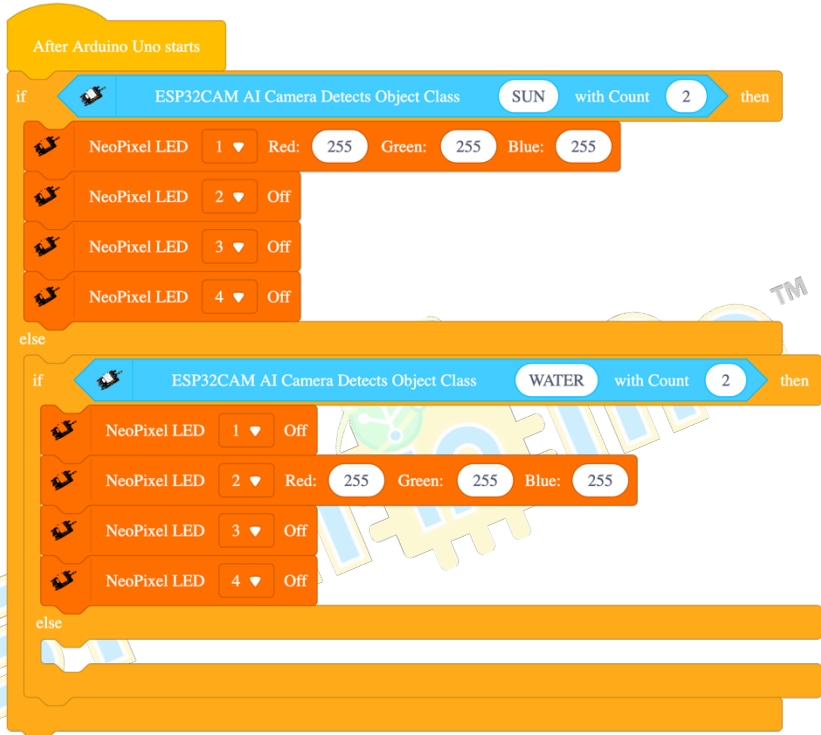
4. Seretkan blok *NeoPixel LED* dan setkan *Red* kepada *255*, *Green* kepada *255* dan *Blue* kepada *255*. Kemudian sambungkan blok *NeoPixel LED Off*



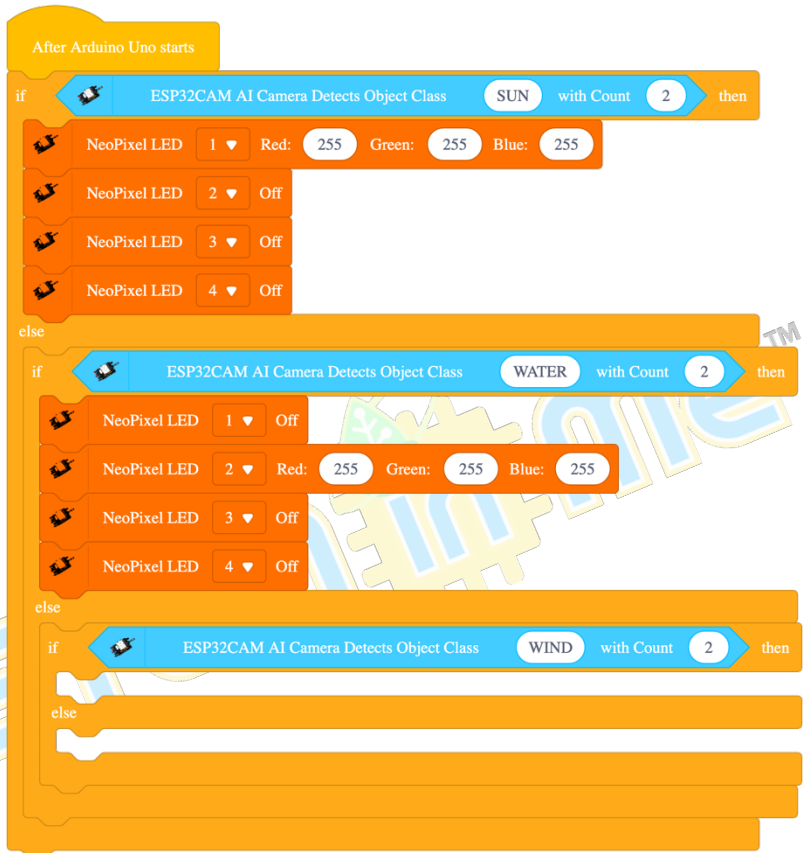
5. Seretkan blok *if, then, else* ke dalam blok *else*. Kemudian, seretkan blok *ESP32CAM AI Camera Detect* ke dalam blok *if then* tukarkan *Object Class* kepada *WATER* dan *Count* kepada *2*



6. Seretkan blok *NeoPixel LED Off* ke dalam ruangan *if then else*. Kemudian, sambungkan dengan blok *NeoPixel LED*, tukarkan kepada *NeoPixel LED* kepada 2. Red kepada 255. Green kepada 255 dan Blue kepada 255. Seterusnya sambungkan dengan blok *NeoPixel LED Off*



7. Seretkan blok *if, then, else* ke dalam blok *else*. Kemudian, seretkan blok *ESP32CAM AI Camera Detect*, ke dalam blok *if then* tukarkan *Object Class* kepada *WIND* dan *Count* kepada 2



8. Seretkan blok *NeoPixel LED Off* ke dalam ruangan *if then else*. Kemudian, sambungkan dengan blok *NeoPixel LED*. tukarkan kepada *NeoPixel LED* kepada *3 Red* kepada *255*. *Green* kepada *255* dan *Blue* kepada *255*. Seterusnya sambungkan dengan blok *NeoPixel LED Off*

The image shows a Scratch script for an ESP32CAM AI Camera. The script starts with a yellow block labeled "After Arduino Uno starts". It then enters a series of nested "if-then-else" blocks based on the object class detected by the camera.

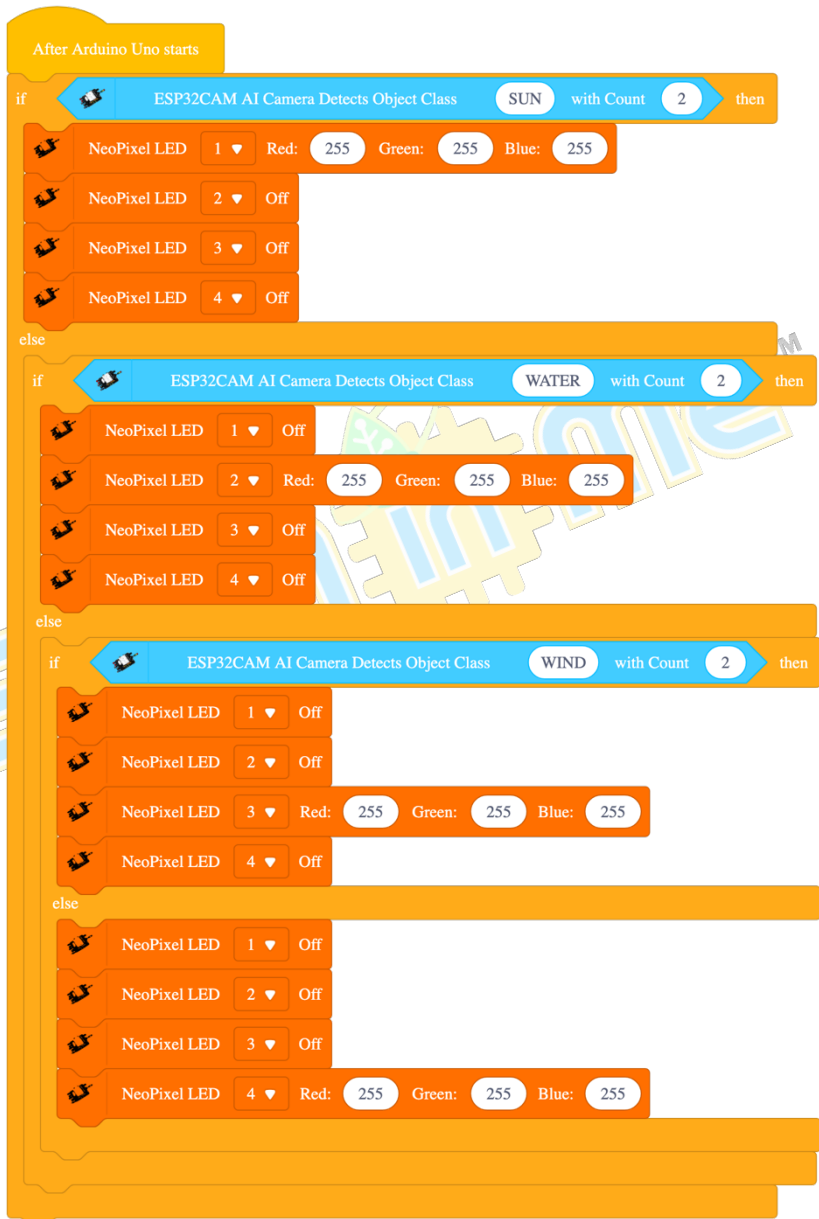
```

if (ESP32CAM AI Camera Detects Object Class: SUN with Count: 2) then
  NeoPixel LED 1: 255
  NeoPixel LED 2: 255
  NeoPixel LED 3: 255
  NeoPixel LED 4: 255
else
  if (ESP32CAM AI Camera Detects Object Class: WATER with Count: 2) then
    NeoPixel LED 1: Off
    NeoPixel LED 2: 255
    NeoPixel LED 3: 255
    NeoPixel LED 4: 255
  else
    if (ESP32CAM AI Camera Detects Object Class: WIND with Count: 2) then
      NeoPixel LED 1: Off
      NeoPixel LED 2: Off
      NeoPixel LED 3: 255
      NeoPixel LED 4: 255
    else
      // Empty else block
    end
  end
end
  
```

The script uses the following logic:

- Initial State:** All four NeoPixel LEDs (1, 2, 3, 4) are set to 255.
- Object Detection:** The script checks for three object classes: SUN, WATER, and WIND, each with a count of 2.
- LED Control:**
 - SUN:** All four LEDs are set to 255.
 - WATER:** LED 1 is set to Off, and LEDs 2, 3, and 4 are set to 255.
 - WIND:** LEDs 1 and 2 are set to Off, and LEDs 3 and 4 are set to 255.

9. Seretkan blok *NeoPixel LED Off* ke dalam ruangan *else* Kemudian, sambungkan dengan blok *NeoPixel LED*. tukarkan kepada *NeoPixel LED* kepada 4. *Red* kepada 255. *Green* kepada 255 dan *Blue* kepada 255.



MERAKYATKAN TEKNOLOGI

- Industry 4WRD
- Pemikiran Kreatif
- Pembudayaan Inovasi
- Kesejahteraan Hidup
- Kelestarian Alam
- Pembelajaran
Menyeronokkan

PENGELUAR:

MICRO CONCEPT TECH SDN BHD
1230153-W

No. 5-5, Pusat Dagangan Shah Alam,
Persiaran Damai, Seksyen 11,
40100 Shah Alam, Selangor, Malaysia

  @steminme



 <http://www.microconcept.com.my>
 steminme@microconcept.com.my