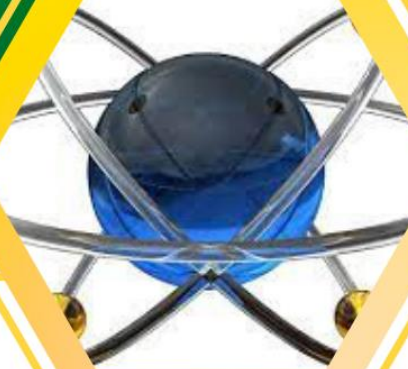
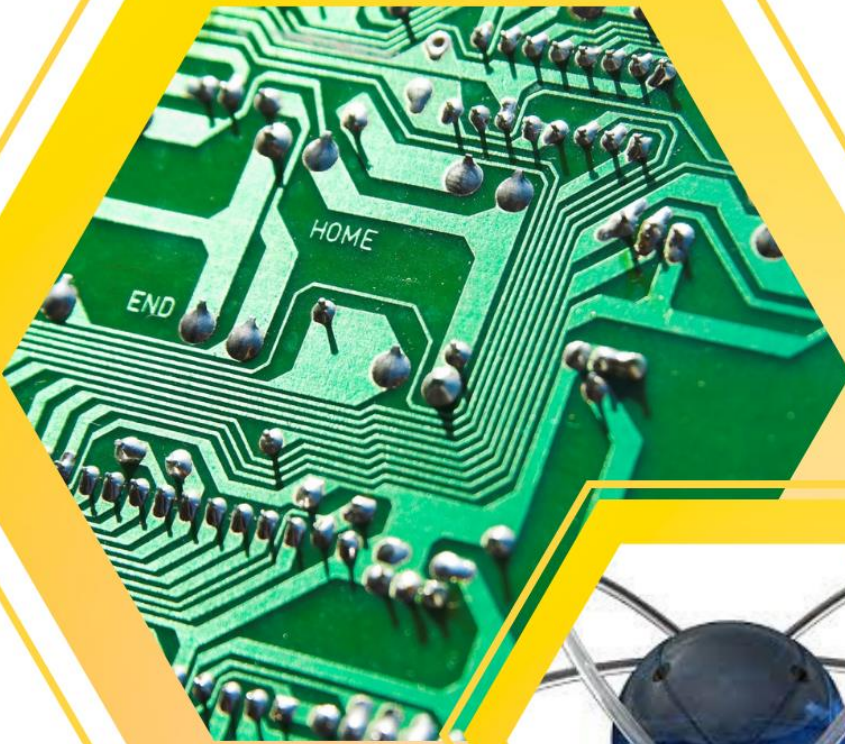


MEMBUAT PCB MENGGUNAKAN

# PROTEUS 8.7



*untuk pemula*

MOHD NIZAM BIN SAMAD  
AMILIA @ EMIL BINTI HASSAN



MEMBUAT PCB MENGGUNAKAN PROTEUS 8.7  
UNTUK PEMULA

# Perpustakaan Negara Malaysia Data Pengkatalogan-dalam-Penerbitan

PUBLISHED BY:  
Politeknik Port Dickson,  
KM 14, Jalan Pantai, 71050 Si-Rusa,  
Port Dickson, Negeri Sembilan  
**AUGUST 2023**

## ***Hakcipta***

Setiap bahagian penerbitan ini tidak boleh diterbitkan semula atau diedarkan dalam sebarang bentuk dengan sebarang cara atau sistem perolehan semula tanpa kebenaran bertulis terlebih dahulu daripada Penerbit.



Data Pengkatalogan-dalam-Penerbitan

Perpustakaan Negara Malaysia

Rekod katalog untuk buku ini boleh didapati  
dari Perpustakaan Negara Malaysia

eISBN 978-629-7643-00-7

## **Notis Cap Perdagangan**

Proteus Design Suite adalah Hak Milik Labcenter Electronics, UK.

## **Notis Bahan Visual**

Imej yang digunakan didalam buku ini adalah daripada hasil kerja penulis sendiri. Walau bagaimana pun terdapat sebahagian imej yang digunakan adalah datang dari pelbagai sumber rujukan. Setiap imej tersebut adalah di bawah hak cipta pemiliknya.

## **Penolakan Tuntutan**

Penerbit dan penulis telah berusaha untuk memastikan segala ketepatan maklumat serta mutu bahan di dalam buku ini. Walau bagaimana pun tiada jaminan diberikan sama ada secara tersurat atau pun tersirat. Penerbit dan penulis menolak sebarang tuntutan terhadap apa jua tanggung jawab serta kerugian atau apa-apa gantirugi secara langsung atau tidak langsung disebabkan daripada penggunaan maklumat daripada buku ini.

# **ACKNOWLEDGEMENTS**

## **PATRON**

Dr. Ishak Bin Mohamad  
Director, Politeknik Port Dickson

## **ADVISORS**

Abdul Rahim Bin Ibrahim  
Deputy Director (Academic), Politeknik Port Dickson

Abdul Razak Bin Ismail  
Head Of Electrical Engineering Department, Politeknik Port Dickson

## **EDITOR**

Masnora Binti Sepikun  
Electrical Engineering Department, Politeknik Port Dickson

## **FACILITATORS**

Che Azlina Binti Che Norohoseni  
PEL  
Roselinda Binti Samion  
PELJ

## **WRITERS**

MOHD NIZAM BIN SAMAD  
AMILIA @ EMIL BINTI HASAN

We would like to convey our outmost gratitude to the department of polytechnic and community college education particularly the e-learning and instructional division (BIPD) for funding our e-book project.

We hereby declare that this module is our original work. To the best of our knowledge. It contains no materials previously written or published by another person. However, if there is any, due acknowledgement and credit are mentioned accordingly in the e-book

# Senarai Kandungan

	Tajuk	Muka Surat
	Daripada Simulasi Ke Papan Litar Tercetak	1
<b>BAB 1</b>	<b>MELAKUKAN TETAPAN (<i>SETTING</i>) KEPADA PROJEK</b>	
1.0	Pra-Tetapan Awal Projek.	3
	Uji Kefahaman 1	12
<b>BAB 2</b>	<b>MELUKIS LITAR SKEMATIK</b>	
2.0	Pengenalan <i>Schematic Capture</i>	15
2.1	Melukis Litar Skematik Menggunakan Proteus	16
2.2	Memilih Komponen	17
2.3	Menyambung Komponen	21
2.4	Kaedah Selection Mode	21
2.5	Kaedah Terminal Mode	23
2.6	Terminal POWER	24
2.7	Terminal DEFAULT	27
2.8	Terminal GROUND	30
	Uji Kefahaman 2	32
<b>BAB 3</b>	<b>MEMASUKKAN KOD SUMBER KE DALAM LITAR</b>	
3.0	PENGENALAN	35
3.1	Arduino NANO	35
3.2	Memasukkan Kod Sumber Menggunakan PROTEUS	37
	Uji Kefahaman 3	41
<b>BAB 4</b>	<b>SIMULASI LITAR</b>	
4.0	Melakukan Simulasi Litar	44
	Uji Kefahaman 4	46
<b>BAB 5</b>	<b>MEMBUAT PCB LAYOUT</b>	
5.0	Menetapkan Dimensi Papan Litar	49
5.1	Memasang Komponen Pada Papan Litar	52
5.2	Menyusun Kedudukan Komponen	54
5.3	Membuat Trek Litar Pada PCB	56
5.4	Menetapkan Laluan Trek ( <i>Track-Routing</i> ).	59
5.5	Memasukkan Teks Ke Atas Papan Litar	61
5.6	Pralihat Papan Litar	63
5.7	Menetapkan Keluaran Litar Akhir	65
	Uji Kefahaman 5	68

	<b>Tajuk</b>	<b>Muka Surat</b>
<b>BAB 6</b>	<b>MEMBUAT PAPAN LITAR TERCETAK</b>	
6.0	PENGENALAN	71
6.1	Mencetak Litar	71
6.2	Proses <i>Etching</i>	73
6.3	Proses Penggerudian	76
6.4	Proses Pematerian Komponen	77
6.5	Pengujian Keterusan Litar	79
6.6	Pemeriksaan Litar Secara Visual	80
	Uji Kefahaman 6	81
	<b>RUJUKAN</b>	<b>82</b>

# Daripada Simulasi Ke Papan Litar Tercetak

Terdapat pelbagai perisian yang boleh digunakan bagi menghasilkan papan litar tercetak. Antara perisian yang biasa digunakan ialah PROTEUS. Untuk edisi kali ini, buku ini akan menerangkan cara untuk membuat sebuah papan litar tercetak untuk mereka yang baru berjinak-jinak dalam menghasilkan litar elektronik.

Secara umumnya perisian PROTEUS boleh dibahagikan kepada tiga bahagian utama iaitu *Schematic Capture*, *PCB layout* dan *Source Code* <sup>1</sup>. Proses pertama yang perlu kita lakukan adalah memasang perisian PROTEUS ke dalam komputer kita. Setelah proses pemasangan perisian tersebut berjaya dilakukan maka barulah kita boleh membuka perisian tersebut untuk peringkat seterusnya iaitu melukis litar skematik .

Kita juga boleh memasukkan coding yang sesuai sekiranya litar tersebut menggunakan mikro-pengawal. Terdapat pelbagai jenis mikro-pengawal yang disokong oleh perisian ini seperti mikro-pengawal daripada keluarga PIC dan ATMEL. Kita juga boleh melakukan simulasi litar untuk mendapatkan gambaran bagaimanakah litar tersebut beroperasi sebelum dipindahkan kepada papan litar tercetak.

Untuk membuat PCB pula kita boleh menggunakan kaedah *Etching* yang memerlukan beberapa proses tertentu sebelum direndam dalam *Ferric-Chloride*. Kita juga boleh menggunakan ciri *gerber file* yang dijana oleh perisian PROTEUS untuk membuat PCB dengan menggunakan mesin CNC bagi mendapatkan hasil yang lebih kemas. Secara umumnya buku ini akan memberi panduan langkah demi langkah bagi memudahkan pembaca memahami kaedah untuk menghasilkan PCB dengan betul.

# BAB 1

## MELAKUKAN TETAPAN (*SETTING*) KEPADA PROJEK



## 1.0 Pra-Tetapan Awal Proyek.

- a. Untuk membuka *software* PROTEUS tersebut, kita perlu *double-click* pada ikon PROTEUS yang terdapat pada bahagian *desktop* komputer seperti **Rajah 1.0**.



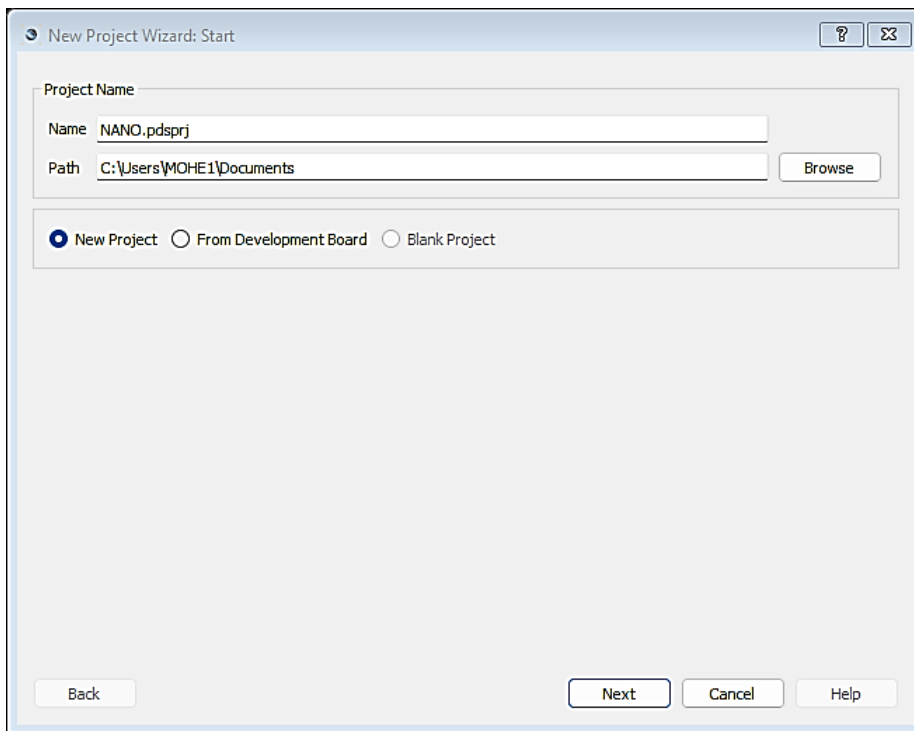
**Rajah 1.0:** Ikon PROTEUS

- b. Kemudian setelah proses dan perkara tersebut, kita akan dibawa kepada antaramuka utama PROTEUS DESIGN SUITE 8.7 seperti pada **Rajah 1.1**.



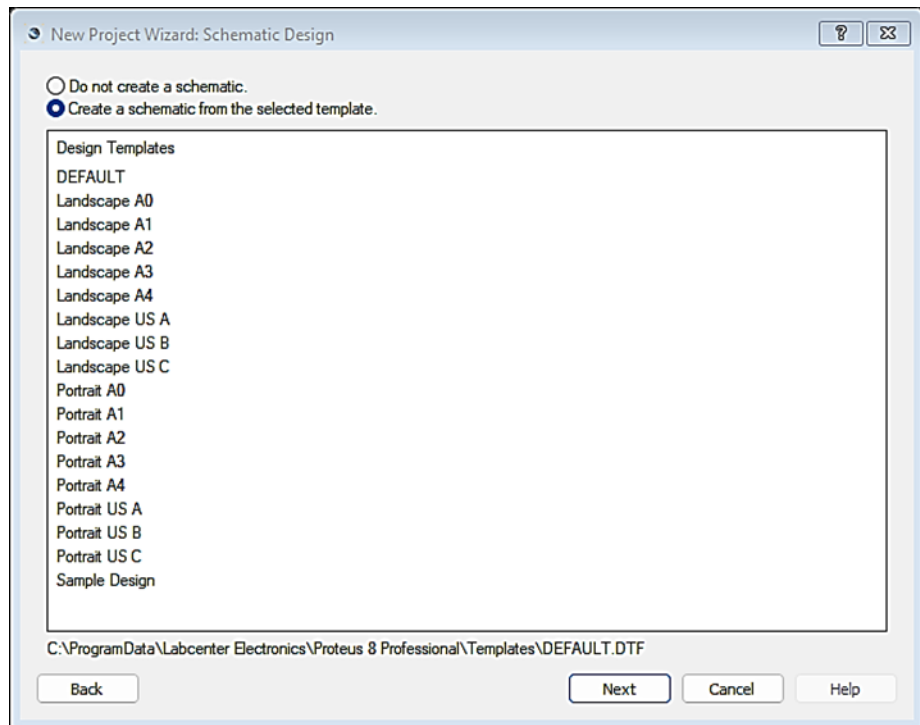
**Rajah 1.1:** Antaramuka Utama PROTEUS DESIGN SUITE 8.7

- c. Untuk antara muka pertama ini seperti **Rajah 1.2**, klik pada bahagian **New Project**.
- d. Pada bahagian ini kita tetapkan nama projek tersebut dan tetapkan ke manakah **Destination Folder** bagi projek tersebut.
- e. Untuk contoh kali ini, kita namakan projek tersebut sebagai **Nano.pdsprj**.
- f. Kemudian klik **Next**.



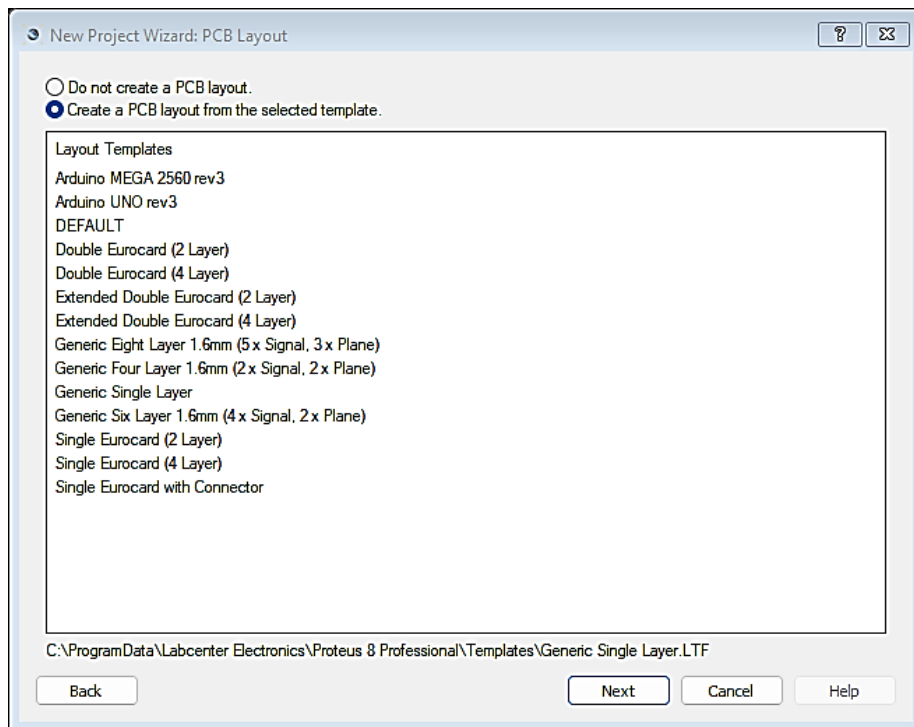
**Rajah 1.2:** Antaramuka Bahagian *New Project*

- g. Selepas menekan butang *Next*, kita akan dibawa kepada tettingkap ***New Project Wizard : Schematic Design*** pada **Rajah 1.3**. Pada bahagian ini, kita klik pada ***Create a schematic from the selected template***.



**Rajah 1.3:** Tetingkap *New Project Wizard : Schematic Design*

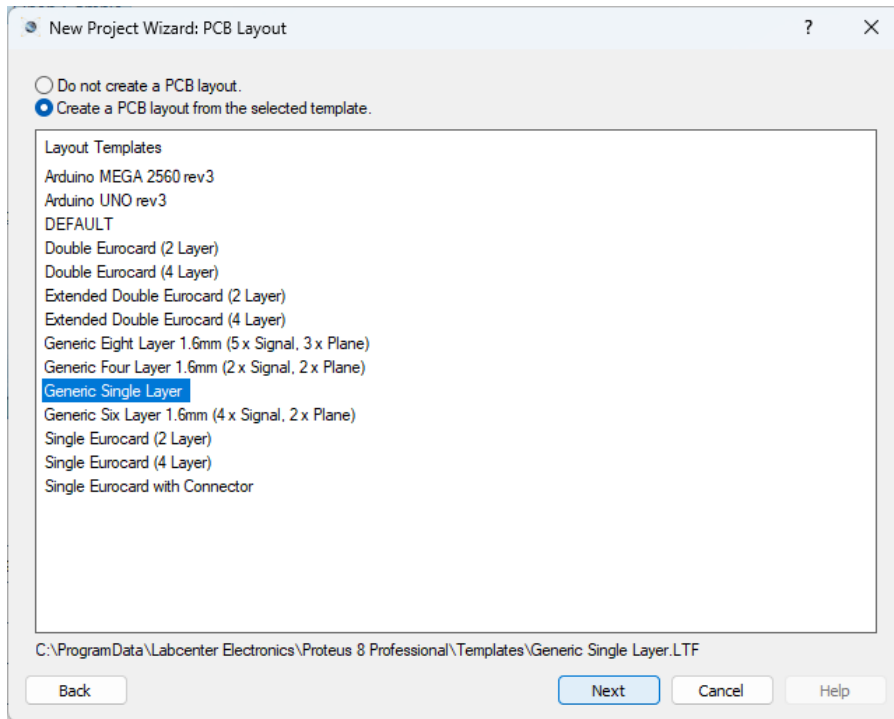
- h. Kemudian, klik **Default**. Seterusnya, klik **Next**.
- i. Setelah menekan butang **Next**, kita akan dibawa pada tettingkap **New Project Wizard : PCB Layout** seperti **Rajah 1.4**.



**Rajah 1.4:** Tettingkap *New Project Wizard : PCB Layout*

- j. Pada tettingkap ini, klik pada bahagian **Create PCB layout from the selected template**.

- k. Kemudian, kita klik pada **Generic Single Layer** seperti pada **Rajah 1.5**, disusuli dengan menekan butang **Next**.

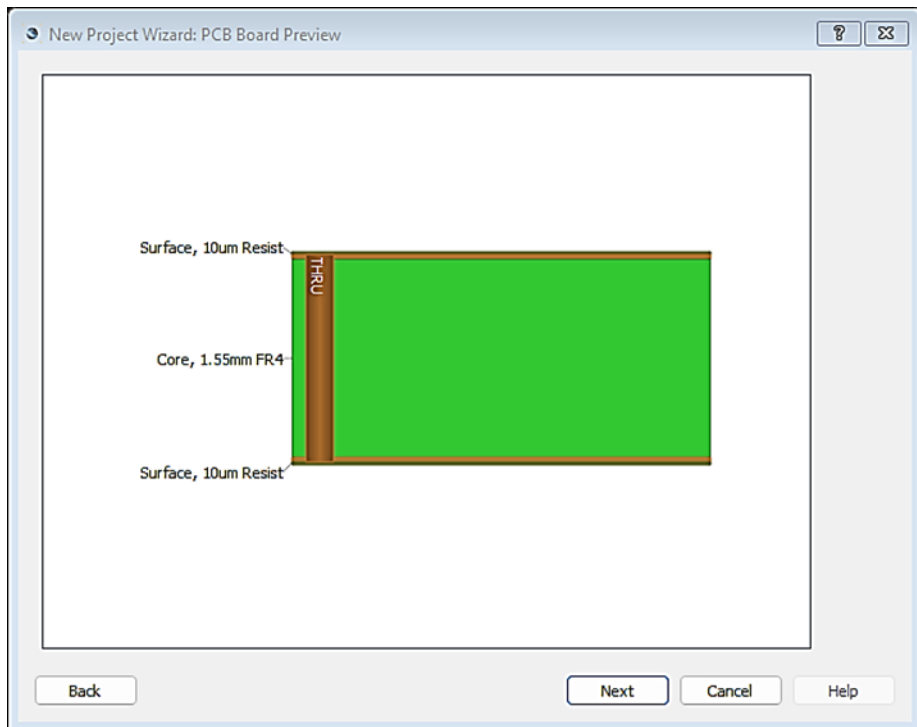


**Rajah 1.5:** Hasil Tetingkap *New Project Wizard : PCB Layout*



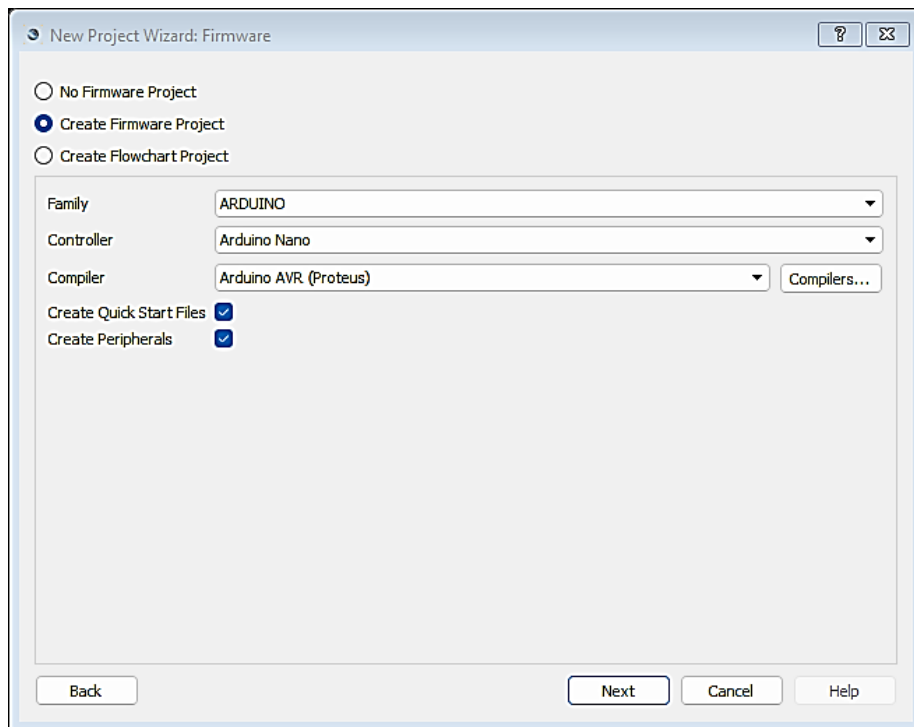


- q. Seterusnya pada **Rajah 1.8**, tettingkap **New Project Wizard : PCB Board Preview** akan terbuka. Kita hanya perlu klik *Next* sahaja memandangkan PCB yang akan kita buat nanti merupakan PCB asas.



**Rajah 1.8:** Tetingkap *New Project Wizard : PCB Board Preview*

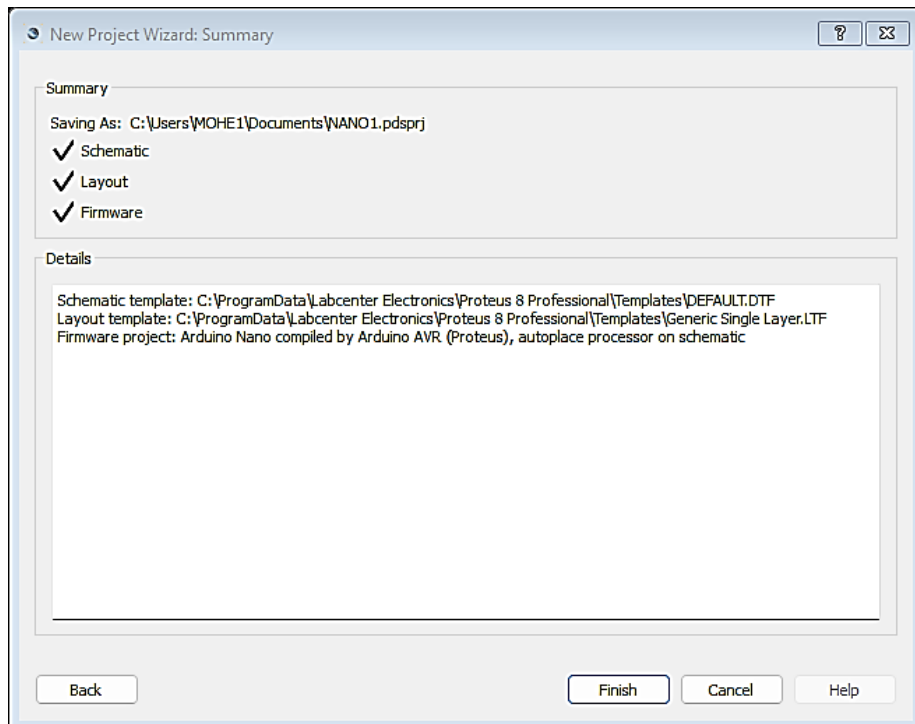
- r. Setelah melangkaui dua tettingkap tersebut kita akan dibawa pula pada **Rajah 1.9** iaitu tettingkap ***New Project Wizard : Firmware***. Memandangkan kita akan menggunakan **Arduino Nano** sebagai komponen utama, maka kita perlu membuat tetapan pada tettingkap tersebut.



**Rajah 1.9:** Tetingkap *New Project Wizard : Firmware*

- s. Pada tettingkap ***New Project Wizard : Firmware*** tersebut, kita klik pada ***Create Firmware Project***. Kemudian setkan ***Family*** kepada Arduino.

- t. Pada ruangan **Controller** tetapkan kepada Arduino Nano dan pada bahagian **Compiler** kita setkan kepada **Arduino AVR (PROTEUS)**, diikuti dengan mengklik butang **Next**.
- u. Akhir sekali, kita akan dibawa kepada tettingkap **New Project Wizard : Summary** pada **Rajah 1.10**. Pada bahagian ini, rangkuman keseluruhan tetapan akan ditunjukkan dengan tanda **check** yang menandakan tetapan yang telah dibuat sebelum ini.



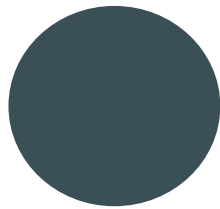
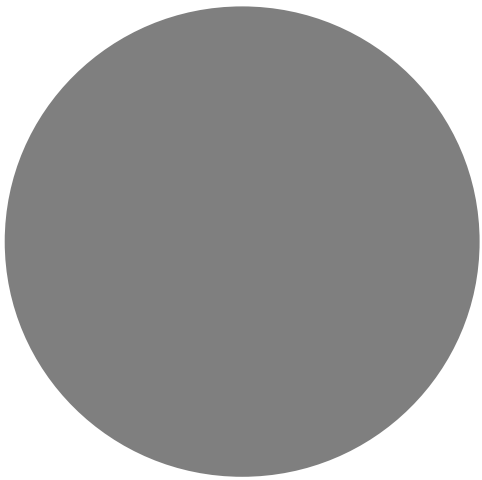
**Rajah 1.10:** Tetingkap *New Project Wizard : Summary*

## Uji Kefahaman 1

1. Apakah fungsi medan **Family** di dalam tettingkap **New Project Wizard : Firmware** ?
2. Bagaimanakah cara untuk kita melakukan tetapan supaya PCB yang terhasil merupakan PCB lapisan tunggal?
3. Nyatakan fungsi **THRU** dan **VIA** di dalam pembuatan PCB.

Imbas **QR-Code** untuk menyemak jawapan anda





## **BAB 2 : MELUKIS LITAR SKEMATIK**

Penangkapan skematik dalam perisian Proteus 8.7 membolehkan pengguna untuk merancang litar mereka secara visual dengan menggunakan ciri yang ada seperti perpustakaan komponen, simbol, dan sambungan elektrik yang telah direka bentuk. Dengan adanya antara muka drag-and-drop yang terdapat dalam perisian ini, kita boleh meletakkan dan menghubungkan semua komponen dengan mudah. Ia juga membolehkan kita untuk meniru perwakilan logik litar fizikal.

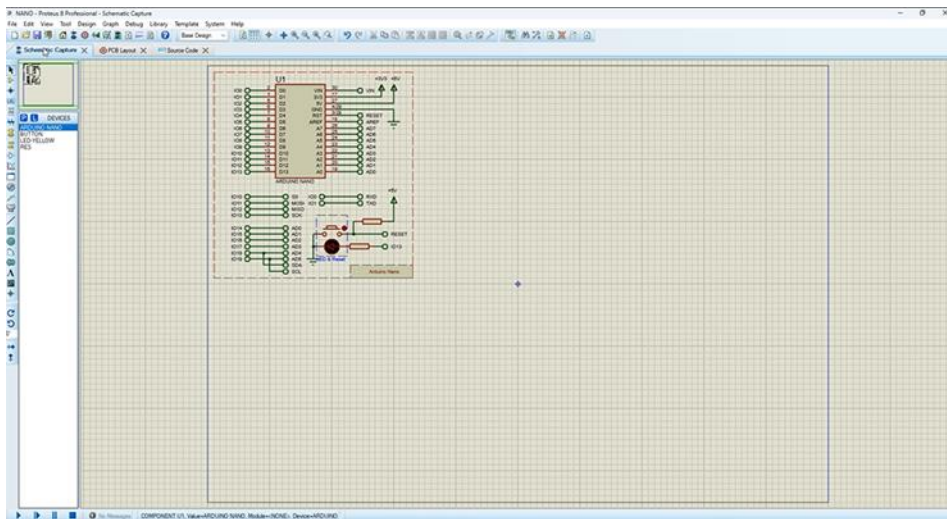
## 2.0 Pengenalan *Schematic Capture*

Langkah yang seterusnya adalah melukis litar skematik menggunakan perisian **PROTEUS Design Suite**. Setelah selesai melakukan tetapan awal sebelum ini, kita akan dibawa kepada tetingkap yang mengandungi tiga *tab* utama iaitu **Schematic Capture**, **PCB Layout** dan **Source Code** seperti pada **Rajah 2.0**.

Dapat diperhatikan disini pada kanvas utama, mikro-pengawal Arduino Nano telah diletakkan secara **default**. Setiap ikon yang terdapat pada ruangan kiri-atas dan kiri-tepi dapat diketahui fungsinya dengan melakukan **mouse-over** pada setiap ikon. Apabila **mouse-over** terhadap ikon dilakukan, akan

terpapar fungsi setiap ikon tersebut secara ringkas yang akan dihuraikan lebih lanjut semasa kita menyiapkan litar ini.

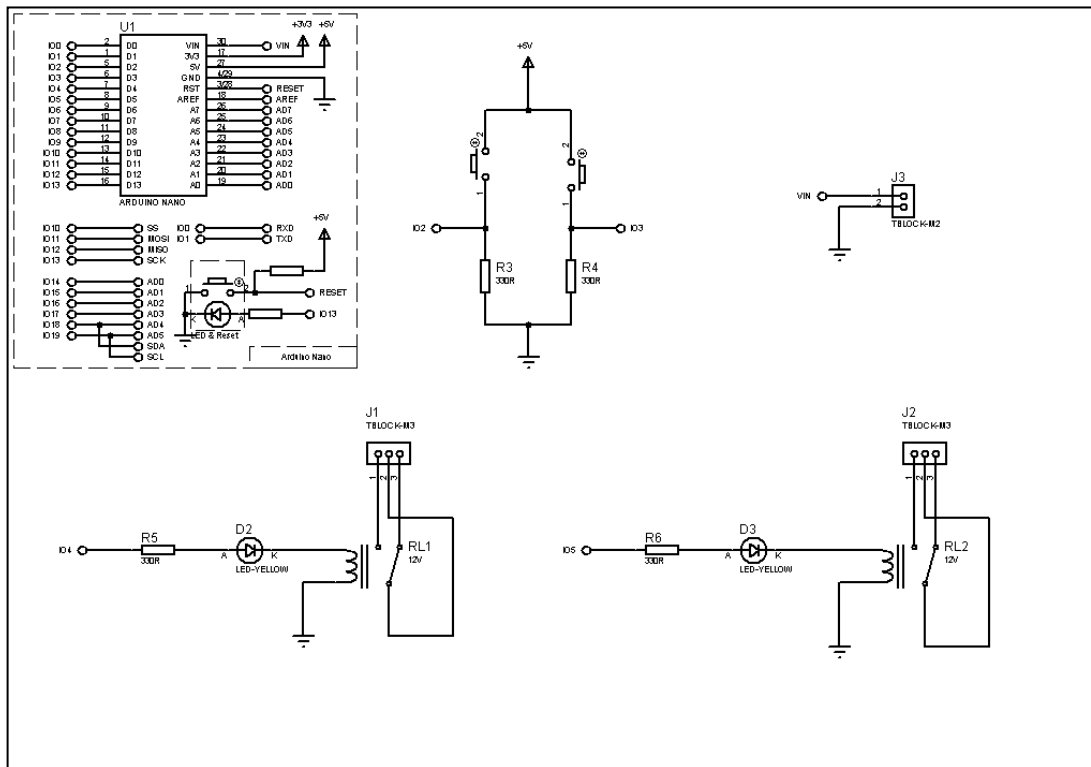
Untuk projek kali ini, kita akan membuat satu projek yang terdiri daripada dua masukan dan dua keluaran yang akan dikawal secara berasingan dengan menggunakan **Arduino Nano** sebagai mikro-pengawal. Dengan kata lain Suis 1 akan mengawal LED 1 manakala Suis 2 akan mengawal LED 2. Kawalan ini akan dilakukan oleh mikro-pengawal **Arduino Nano** yang bertindak memproses isyarat masukan dan memprosesnya sebagai isyarat keluaran.



**Rajah 2.0:** Tetingkap *Schematic Capture*, *PCB Layout* dan *Source Code*

## 2.1 Melukis Litar Skematik Menggunakan Proteus

Untuk contoh di dalam buku ini, kita akan melukis satu litar skematik seperti yang ditunjukkan pada **Rajah 2.1** di bawah dengan menggunakan perisian PROTEUS.



Rajah 2.1: Contoh Litar Skematik

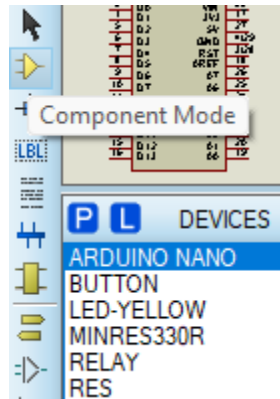
Komponen yang akan digunakan di dalam litar tersebut dinyatakan pada **Jadual 2.1** ini.

**Jadual 2.1:** Senarai Komponen pada Litar Skematik

<i>Category</i>	<i>Quantity</i>	<i>References</i>	<i>Value</i>
Resistors	1	R1	10k
Resistors	1	R2	1k
Resistors	4	R3-R6	330R
Integrated Circuits	1	U1	ARDUINO NANO
Diodes	3	D1-D3	LED-YELLOW
Miscellaneous	1	BTN1	
Miscellaneous	2	J1-J2	TBLOCK-M3
Miscellaneous	1	J3	TBLOCK-M2
Miscellaneous	2	RL1-RL2	12V

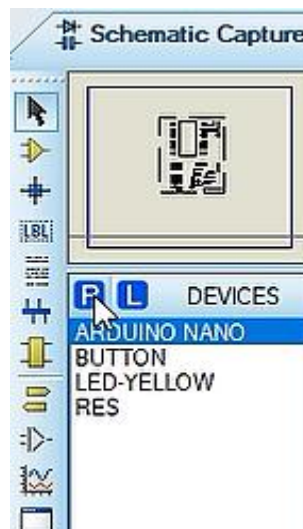
## 2.2 Memilih Komponen

- a. Untuk mula memilih komponen yang terdapat di dalam PROTEUS, Langkah pertama ialah setkan kepada **Component Mode** seperti **Rajah 2.2**.



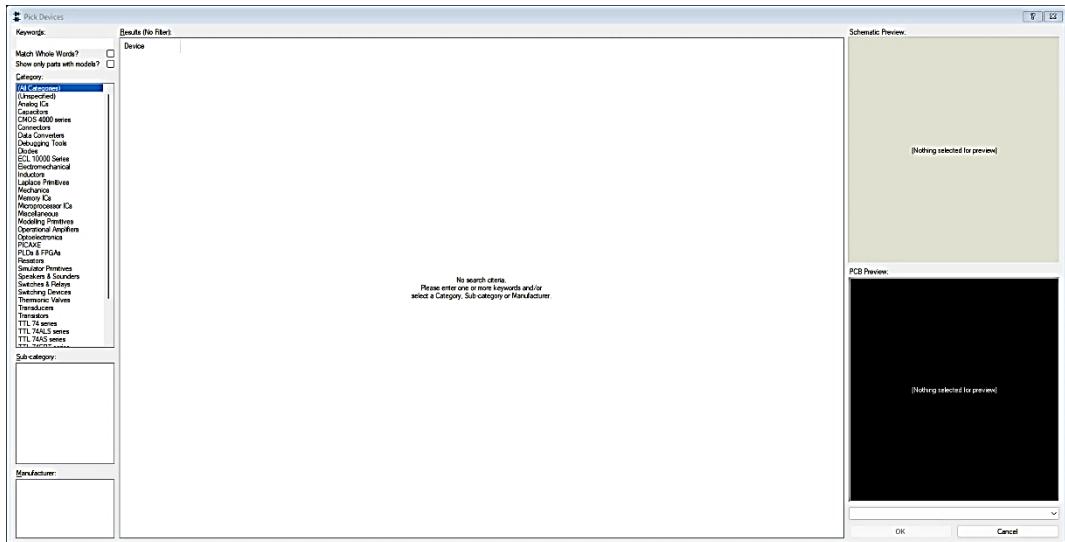
Rajah 2.2: Tetingkap *Component Mode*

- b. Ini boleh dilakukan dengan mengklik ikon **Component Mode** pada bahagian sebelah kiri-atas. Kemudian, di dalam **Component Mode** barulah kita boleh memilih komponen dengan mengetik ikon seperti yang ditunjukkan pada **Rajah 2.3** di bawah.



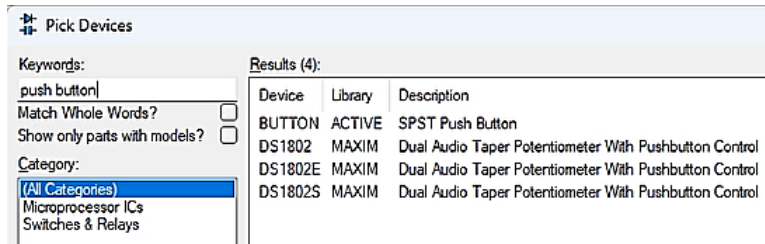
Rajah 2.3: Memilih Komponen

- c. Selepas mengetik ikon tersebut, kita akan dibawa kepada tettingkap **Pick Devices** pada **Rajah2.4**. Di dalam tettingkap **Pick Devices** tersebut, kita akan dapat melihat senarai komponen yang ada di dalam PROTEUS.



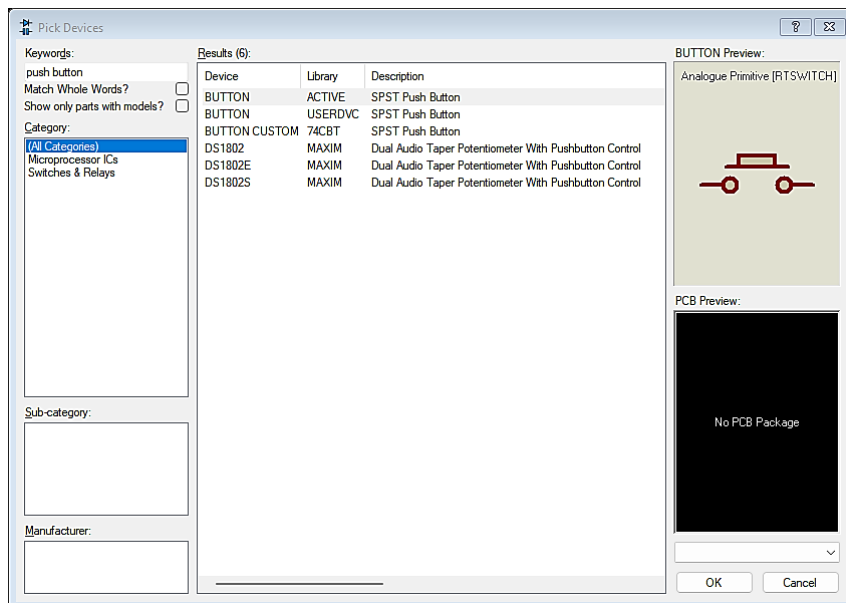
**Rajah 2.4:** Tettingkap *Pick Devices*

- d. Untuk litar kita pada kali ini, kita akan memilih suis tekan hidup sebagai *input* masukan. Oleh itu, pada tettingkap **Pick Devices** kita taip kata kunci pada medan **Keywords** dengan perkataan **Push Button**. PROTEUS akan segera menyenaraikan semua komponen yang mungkin berdasarkan kepada kata kunci yang kita masukkan seperti **Rajah 2.5**.



Rajah 2.5: Tetingkap *Keywords*

- e. Kemudian, kita klik butang **OK** yang terletak pada bahagian bawah-kanan pada tettingkap **Pick Devices** tersebut. Pada ruangan **PCB Preview** lazimnya akan dipaparkan **PCB footprint** seperti pada paparan **Rajah2.6**.



Rajah 2.6: Tetingkap *Pick Devices Push Button*

f. Walau bagaimana pun, terdapat juga komponen yang tiada mempunyai **PCB footprint**. Kita boleh mengatasi masalah ini dengan membuat **PCB footprint** tersebut secara *custom-made* dengan menggunakan ciri packaging tools yang terdapat di dalam perisian PROTEUS.

g. Kemudian, kita ulangi proses yang sama untuk memasukkan komponen yang lain seperti perintang 330 ohm, LED, dan *Relay*.

h. Setiap komponen yang kita pilih boleh diubah kedudukannya mengikut kesesuaian.

i. Malah, kita juga boleh mengubah orientasi setiap komponen kepada keadaan menegak atau pun melintang seperti yang ditunjukkan pada **Rajah 2.7**.

j. Apabila kita mengetik-kanan pada setiap komponen akan terpapar sub-menu untuk komponen tersebut.

	Drag Object	
	Edit Properties	Ctrl+E
	Delete Object	
	Rotate Clockwise	Num--
	Rotate Anti-Clockwise	Num++
	Rotate 180 degrees	
	X-Mirror	Ctrl+M
	Y-Mirror	
	Cut To Clipboard	
	Copy To Clipboard	
	Goto Child Sheet	Ctrl+C
	Goto Part in Design Explorer	
	Highlight Part in PCB Layout	
	Highlight Net on Schematic	
	Highlight Net on PCB Layout	
	Display Model Help	Ctrl+H
	Display Datasheet	Ctrl+D
	Show Package Allocation	
	Operating Point Info	
	Configure Diagnostics	
	Make Device	
	Packaging Tool	
	Decompose	
	Increment	Page-Up
	Decrement	Page-Down
	Toggle	Space

**Rajah 2.7:** Sub Menu pada Komponen

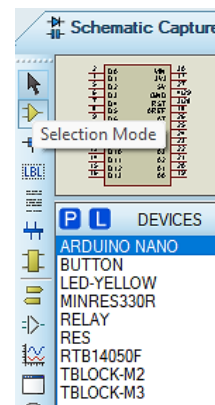
- k. Di dalam sub-menu tersebut kita boleh memutar orientasi setiap komponen sama ada mengikut atau pun melawan arah jam.
  
- l. Selain itu, kita juga boleh mengubah orientasi setiap komponen dengan menggunakan ciri pantulan (***mirror***) berdasarkan kepada paksi-x atau paksi-y mengikut kepada kesesuaian.

## 2.3 Menyambung Komponen

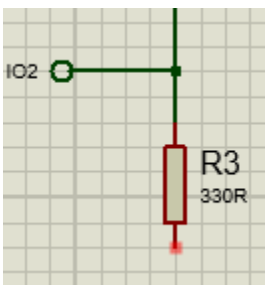
Setelah selesai memasukkan semua komponen ke dalam kanvas utama, langkah seterusnya yang perlu dilakukan ialah membuat penyambungan setiap komponen. Penyambungan setiap komponen di dalam PROTEUS boleh dilakukan dengan menggunakan dua kaedah iaitu dengan menggunakan **Selection Mode** dan **Terminal Mode**.

### 2.3.1 Kaedah Selection Mode

- a. Untuk menyambung komponen menggunakan *Selection Mode*, langkah pertama yang perlu dilakukan ialah mengetik ikon *Selection Mode* yang terletak pada bahagian kiri-atas kanvas seperti pada **Rajah 2.8**.



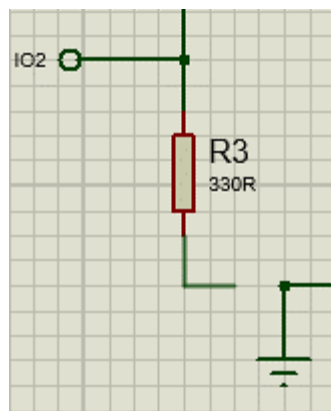
**Rajah 2.8:** Ikon *Selection Mode*



**Rajah 2.9:** Titik Merah pada Komponen yang Disambung

- b. Seterusnya pada ruangan kanvas, kita klik pada komponen yang ingin disambung. Proteus akan mengesan komponen tersebut secara automatik dengan mengeluarkan tanda titik berwarna merah seperti pada **Rajah 2.9** yang bermaksud komponen tersebut bersedia untuk disambungkan kepada komponen yang lain.

- c. Apabila tanda titik berwarna merah ini muncul maka bolehlah kita menarik kursor dari satu komponen kepada komponen yang lain.
- d. Semasa cursor tersebut ditarik garisan lurus akan keluar secara automatik seperti pada **Rajah 2.10**.
- e. Perlu diingat bahawa garisan lurus itu mewakili sambungan diantara setiap komponen.



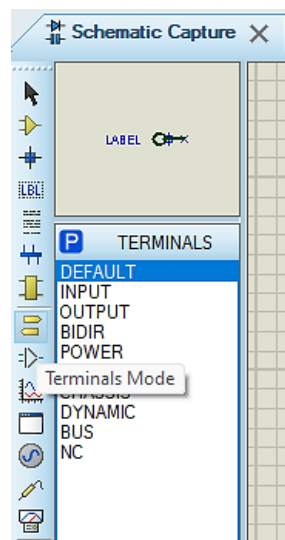
**Rajah 2.10:** Garisan Lurus Mewakili Penyambungan antara Setiap Komponen

- f. Semasa cursor tersebut ditarik garisan lurus akan keluar secara automatik seperti pada **Rajah 2.10**.
- g. Perlu diingat bahawa garisan lurus itu mewakili sambungan diantara setiap komponen.

### 2.3.2 Kaedah Terminal Mode

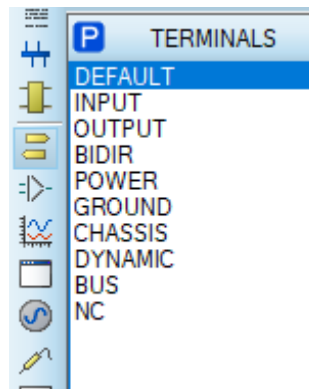
Kaedah **Terminal Mode** membolehkan kita menyambung setiap komponen secara maya tanpa kehadiran sambungan yang sebenar. Ciri ini membolehkan sambungan setiap litar dapat dilakukan dengan lebih kemas tanpa kehadiran garisan sambungan yang berselirat yang amat mengganggu pandangan. Kelebihan yang didapati dengan menggunakan **Terminal Mode** ialah kemudahan untuk melakukan **troubleshooting** sekiranya berlaku kesilapan sambungan. Disamping itu, ia juga dapat mengurangkan berlakunya kesilapan semasa melukis sambungan litar.

- a. Untuk menggunakan terminal mode langkah pertama yang perlu dilakukan ialah dengan mengetik ikon **terminal mode** yang terdapat pada bahagian sebelah kiri merujuk kepada **Rajah 2.11**.



Rajah 2.11: Ikon Terminal Mode

- b. Setelah ikon terminal mode diketik ia akan mengeluarkan sub-menu pilihan seperti ditunjukkan pada **Rajah 2.12**.

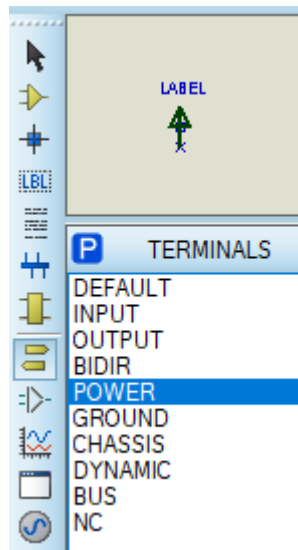


**Rajah 2.12:** Sub-menu Pilihan *Terminal Mode*

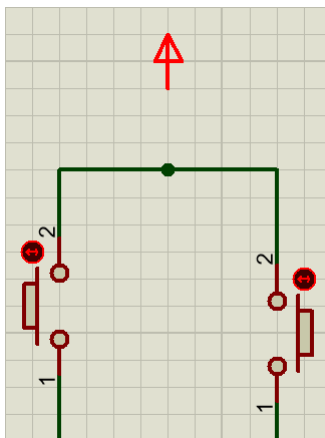
Dalam litar contoh ini , terminal yang akan digunakan ialah **DEFAULT** , **POWER** dan **GROUND**. Terminal-terminal lain juga boleh digunakan mengikut kesesuaian litar masing-masing.

### 2.3.2.1 Terminal **POWER**

- a. Untuk menggunakan *Terminal Power*, kita boleh klik pada perkataan **POWER** yang terdapat di dalam menu **TERMINALS** iaitu **Rajah 2.13**.



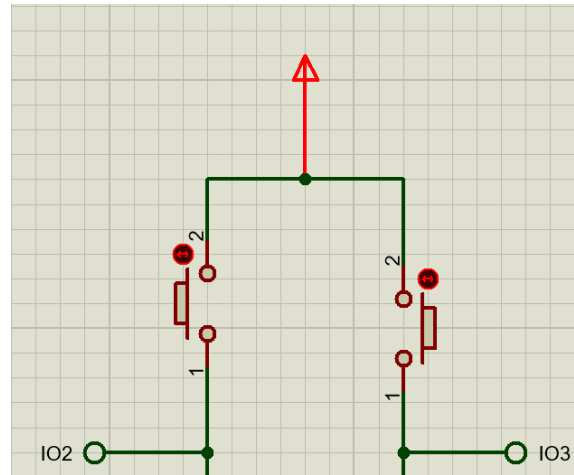
**Rajah 2.13:** Terminal **POWER**



- b. Selepas itu, bolehlah kita memasukkan terminal **Power** ke litar seperti yang ditunjukkan pada **Rajah 2.14**.

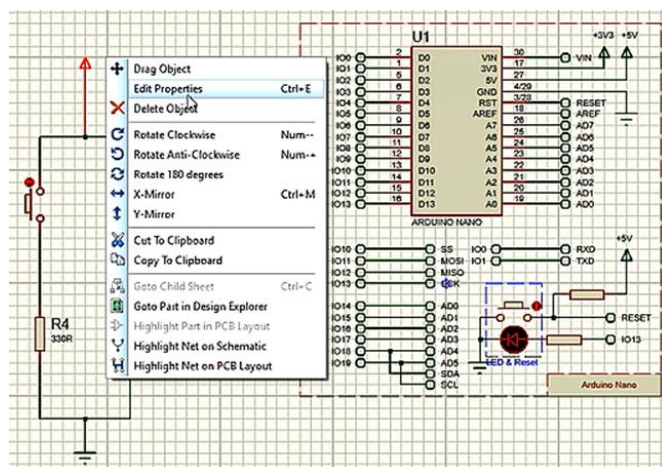
**Rajah 2.14:** Ikon Terminal **POWER**

- c. Terminal **Power** tersebut perlulah disambungkan dengan menggunakan **Selection Mode** seperti **Rajah 2.15**.



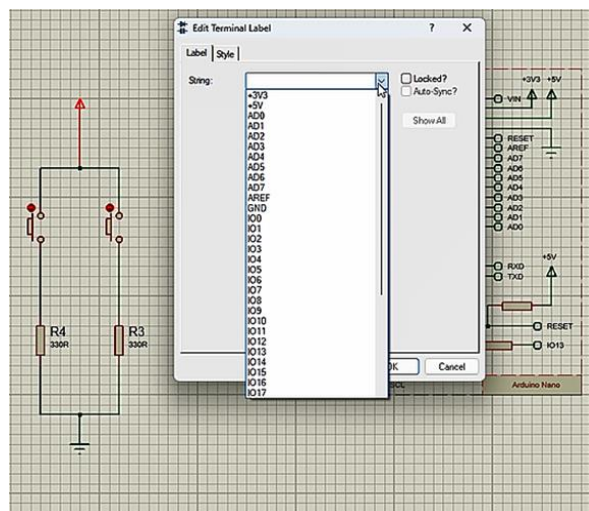
**Rajah 2.15:** Penyambungan Terminal Power secara Selection Mode

- d. Langkah seterusnya adalah untuk menyunting parameter bagi terminal **POWER**. Memandangkan litar yang digunakan menggunakan bekalan kuasa sebanyak 5 V, maka nilai ini mestilah disetkan kepada parameter terminal **POWER** tersebut. Ia boleh dilakukan dengan mengetikkan pada terminal **POWER** tersebut untuk mengeluarkan sub-menu seperti ditunjukkan pada **Rajah 2.16**.



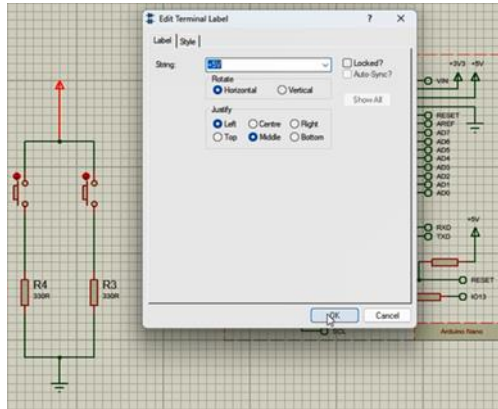
**Rajah 2.16:** Penyuntingan Parameter Terminal Power

- e. Pada bahagian sub-menu tersebut, ketik pada **Edit Properties** untuk memasukkan parameter bagi terminal **POWER**.
- f. Selepas itu, kita akan dibawa kepada tettingkap **Edit Terminal Label**. Pada tettingkap ini, akan keluar **drop down menu list** pada bahagian **String** yang memaparkan semua terminal yang ada dalam litar seperti yang dipaparkan pada **Rajah 2.17**.



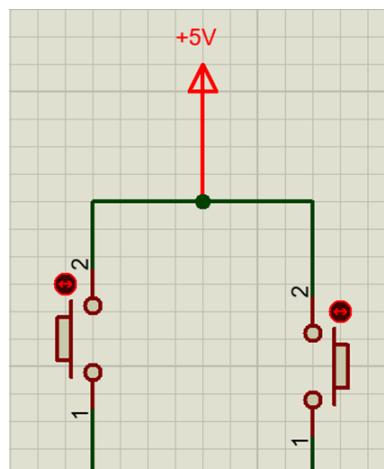
**Rajah 2.17:** Senarai *String* dalam Litar

- g. Pada bahagian *String* **Rajah 2.18** tersebut, kita tetapkan parameter kepada 5 V diikuti dengan menetek butang OK.



**Rajah 2.18:** Pemilihan 5V pada *String*

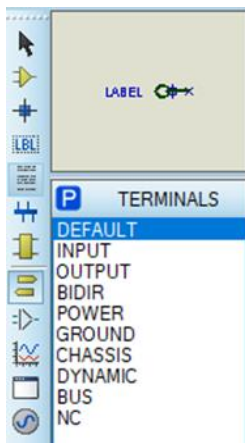
- h. Selepas itu, **Rajah 2.19** dapat diperhatikan pada bahagian terminal POWER tersebut telah mempunyai nilai yang ditandai sebagai 5V yang berfungsi sebagai bekalan kuasa bagi litar isyarat masukan.



**Rajah 2.19:** Paparan 5V pada Terminal *Power*

### 2.3.2.2 Terminal *DEFAULT*

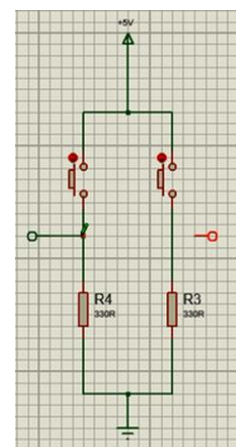
Terminal *DEFAULT* pula membolehkan kita menetapkan sebarang nilai pada bahagian string. Nilai itu mungkin sebarang nombor atau pun aksara atau pun gabungan diantara nombor dan aksara. Di dalam litar contoh ini terminal *DEFAULT* digunakan untuk menggantikan fungsi garisan sambungan. Kaedah ini membolehkan lukisan litar tersebut akan kelihatan lebih kemas dan teratur.



- a. Untuk mengaktifkan terminal *DEFAULT*, menetik ikon terminal *Mode*. Kemudian, pada sub-menu ketik pada perkataan *DEFAULT* seperti yang ditunjukkan pada **Rajah 2.20**.

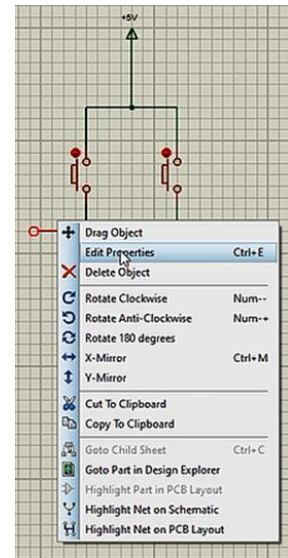
**Rajah 2.20:** Tetingkap Terminal *DEFAULT*

- b. Setelah sub-menu *DEFAULT* diaktifkan, maka **Rajah 2.21** bolehlah kita memasukkan terminal tersebut di dalam lukisan litar utama.

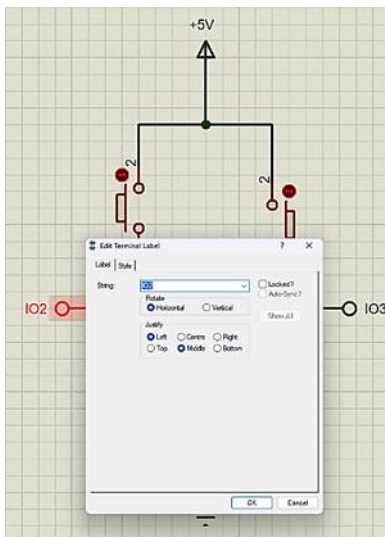


**Rajah 2.21:** Terminal *Default* Warna Merah

- c. Langkah seterusnya **Rajah 2.22** ialah untuk menyunting nilai parameter bagi terminal **DEFAULT**. Ia dilakukan dengan mengetikkan pada terminal **DEFAULT** yang akan membawa kita pada tettingkap sub-menu.



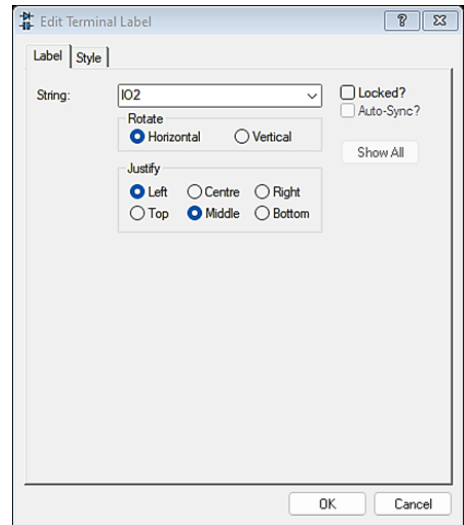
**Rajah 2.22:** Sub-menu Terminal *Default*



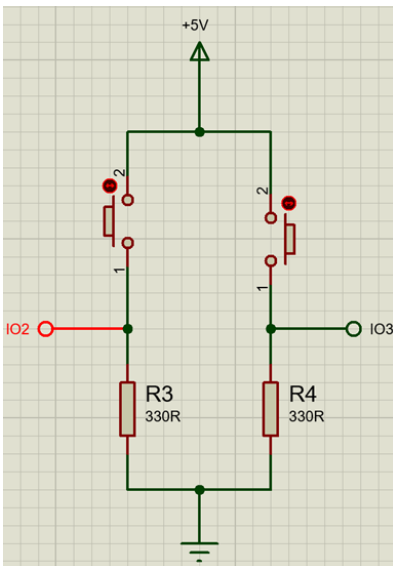
**Rajah 2.23:** Paparan *Edit Terminal Label*

- d. Pada ruangan sub-menu tersebut ketik pada **Edit Properties** untuk membawa kita pada paparan **Rajah 2.23** iaitu tettingkap **Edit Terminal Label**. Pada ruangan ini pastikan label adalah sama dengan terminal yang hendak kita sambungkan.

- e. Sebagai contoh dalam litar ini, **Rajah 2.24** terminal pertama akan disambungkan kepada pin D2 yang diwakili oleh terminal **IO2**. Oleh itu, terminal **DEFAULT** yang pertama kita akan labelkan juga sebagai **IO2**. Label ini hendaklah diubah pada medan *String* diikuti dengan menetik butang OK.



**Rajah 2.24:** Hasil Tetapan *String*



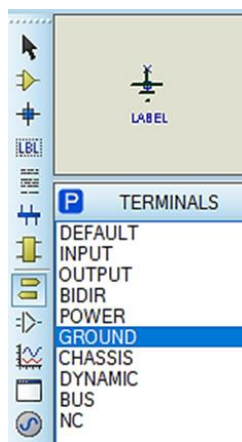
**Rajah 2.25:** Hasil Tetapan *String*

- f. Dapat diperhatikan pada **Rajah 2.25** bahawa label pada terminal **DEFAULT** yang pertama telah berubah kepada **IO2** menandakan label tersebut telah berjaya dilakukan.

- g. Proses yang sama kita lakukan untuk terminal yang lain dengan menggunakan langkah yang sama.

### 2.3.2.3 Terminal **GROUND**

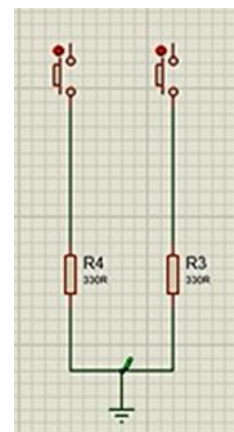
Di dalam litar elektronik, terminal **GROUND** merupakan terminal rujukan dimana voltannya adalah sentiasa bernilai sifar. Dengan kata lain ianya berfungsi seperti seolah-olah terminal negatif bagi litar arus terus. Oleh yang demikian, terminal **GROUND** mestilah dipasang pada setiap litar skematik di dalam **PROTEUS** bagi membolehkan litar yang dilukis ini boleh disimulasikan tanpa sebarang ralat. Ikuti langkah-langkah dibawah bagi memasukkan terminal **GROUND**.



- a. Untuk memasukkan terminal **GROUND** di dalam lukisan litar skematik kita perlulah berada dalam **Terminal Mode**. Kemudian ketik pada **GROUND** seperti paparan **Rajah 2.26**.

**Rajah 2.26:** Terminal **GROUND**

- b. Selepas itu, klik pada kanvas dan tetapkan kedudukan litar **GROUND** pada tempat yang sesuai. Kemudian, buat sambungan kepada komponen lain menggunakan **Selection Mode** seperti yang ditunjukkan pada **Rajah 2.27**.



**Rajah 2.27:** Terminal **GROUND**

Proses tersebut akan dilakukan secara berulang kali sehinggalah litar yang dilukis itu siap dan boleh di simulasikan. Sepanjang menyiapkan litar tersebut kita boleh juga menggunakan terminal yang lain dengan kaedah dan proses yang sama dengan menggunakan ***Terminal Mode***.

Di dalam perisian **Proteus Design Suite 8.7**, ciri ***Terminal Mode*** membolehkan pengguna berinteraksi dengan komponen maya menggunakan antara muka baris arahan. Mod ini amat berguna untuk mengautomasikan tugas, menjalankan simulasi dan melaksanakan pemeriksaan reka bentuk dengan cekap. Dengan memanfaatkan ***Terminal Mode***, kita boleh mengakses pelbagai keupayaan yang tersedia di dalam **Proteus** dengan hanya melalui arahan teks mudah.

## Uji Kefahaman 2

1. Nyatakan tiga jenis tab utama yang terdapat pada perisian PROTEUS.
2. Apakah yang perlu dilakukan untuk kita menyambung sesuatu komponen dengan menggunakan kaedah terminal **DEFAULT**?
3. Bagaimanakah cara untuk kita memastikan setiap komponen adalah bersambung di antara satu sama lain di dalam lukisan skematik?

Imbas **QR-Code** untuk menyemak jawapan anda



**BAB 3 :  
MEMASUKKAN KOD  
SUMBER KE DALAM  
LITAR.**

Penggunaan mikropengawal dalam litar memerlukan penyepaduan kod sumber ke dalam mikropengawal untuk membolehkannya beroperasi seperti yang diprogramkan.

Pengawal mikro berfungsi sebagai otak sistem yang berperanan untuk memproses input, melaksanakan kod serta mengawal keluaran yang dikehendaki. Dalam litar contoh ini, kita akan menggunakan Arduino NANO sebagai mikropengawal.

Litar contoh yang digunakan dalam buku ini mempunyai dua input dan dua output yang akan dikawal oleh Arduino Nano. Untuk mengawal setiap output secara bebas, suis kawalan khusus disepadukan ke dalam reka bentuk. Suis ini membolehkan pengguna berinteraksi dengan litar seterusnya akan mempengaruhi tingkah laku pengawal mikro dan keluaran litar.



### 3.0 PENGENALAN

Oleh kerana kita menggunakan mikro-pengawal di dalam litar, maka **source code** perlulah dimasukkan ke dalam mikro-pengawal bagi membolehkan ia beroperasi seperti yang telah diprogramkan seperti yang dinyatakan pada permulaan tadi, litar yang akan digunakan dalam contoh buku ini mempunyai dua masukan dan dua keluaran yang akan dikawal oleh mikro-pengawal. Mikro-pengawal yang digunakan dalam litar ini menggunakan Arduino NANO.

#### 3.1 Arduino NANO

Arduino Nano merupakan salah satu mikropengawal didalam keluarga Arduino. Selain arduino nano, terdapat juga mikropengawal didalam keluarga Arduino seperti Arduino Uno, Arduino Mega, Arduino Leonardo dan sebagainya.

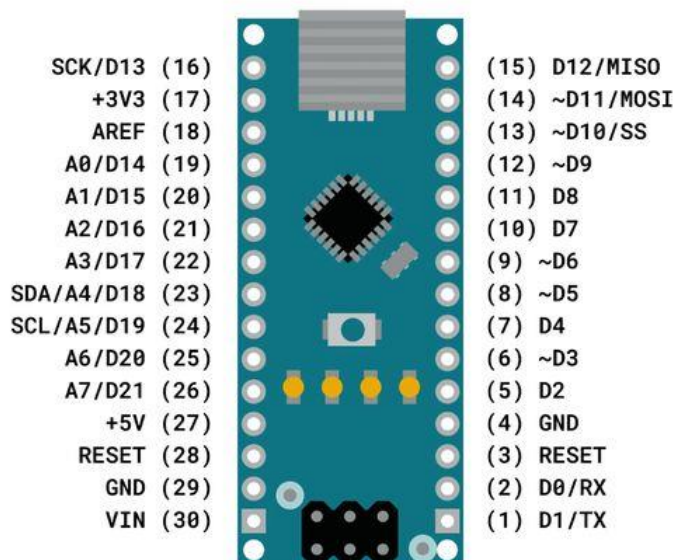


**Rajah 3.0:** ARDUINO NANO

Setiap daripada keluarga Arduino tersebut mempunyai asas yang sama iaitu menggunakan mikro-pengawal **ATmel** sebagai nadi utama. Walau bagaimana pun setiap jenis arduino tersebut mempunyai saiz dan spesifikasi yang berbeza bergantung kepada keperluan litar. Untuk litar yang bakal kita hasilkan, penggunaan Arduino Nano sudah memadai memandangkan kita hanya menggunakan dua masukan dan dua keluaran.

Untuk Arduino Nano, bekalan kuasa boleh didapati daripada dua sumber iaitu sama ada daripada **on board USB port** atau pun daripada pin **Vin**. Untuk bekalan kuasa daripada **on board USB port**, kita boleh menyambungkan port tersebut daripada mana-mana port USB. Voltan bekalan bagi USB port adalah tetap iaitu **5 VDC**.

Bagi pin Vin pula, kita boleh menyambungkan Arduino Nano kepada bekalan kuasa dalam lingkungan 5 – 12 VDC. Walaupun menerima bekalan masukan yang besar pada pin **Vin**, Papan Arduino tetap menghasilkan dua jenis **voltan keluaran iaitu 5V dan 3.3V**. Keluaran voltan ini membolehkan kita untuk menyambungkan papan Arduino sebagai punca bekalan kuasa untuk komponen elektronik yang lain.



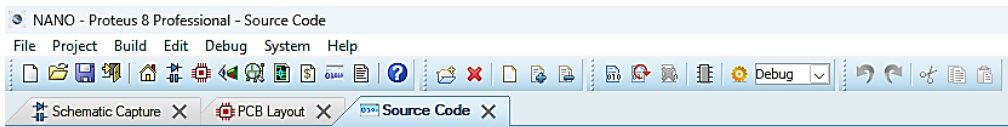
**Rajah 3.1:** Pin Keluaran ARDUINO NANO

Untuk membenamkan kod sumber kepada papan arduino secara fizikal, kita hanya perlu menyambungkan papan arduino kita kepada PC dengan menggunakan kabel USB. Kemudian kod sumber tersebut dapat dibenamkan dengan menggunakan perisian **Arduino IDE** yang telah dipasang pada komputer tersebut.

## 3.2 Memasukkan Kod Sumber Menggunakan PROTEUS

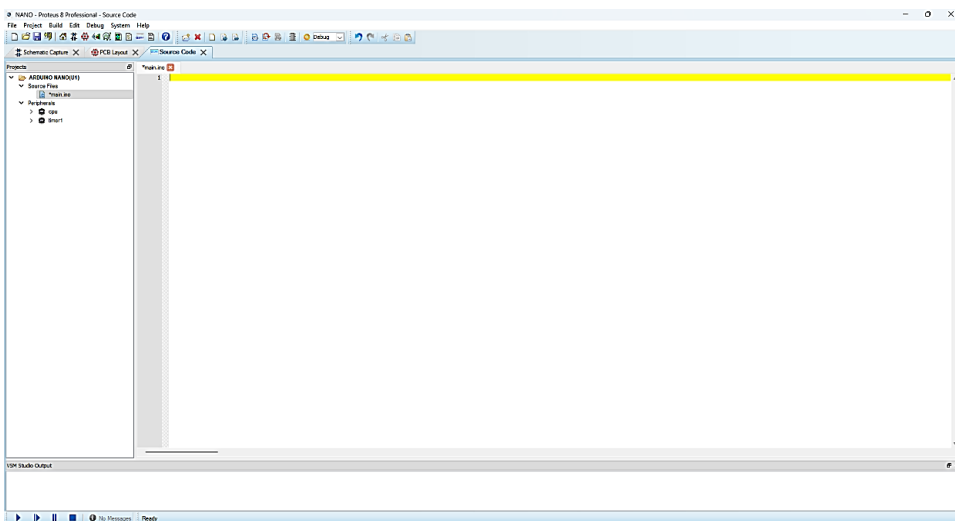
Walau bagaimana pun, untuk memudahkan kita membuat analisis, maka kita boleh melakukan simulasi terlebih dahulu dengan menggunakan perisian Proteus. Didalam contoh ini, setiap keluaran akan mempunyai suis kawalan yang berasingan. *Source code* bagi mikro-pengawal tersebut boleh dimasukkan menggunakan langkah ini.

- a. Pertama sekali kita klik tab **Source Code** bagi membolehkan medan untuk **coding** dipaparkan seperti pada **Rajah 3.0**.



**Rajah 3.2:** Tab *Source Code*

- b. Kita akan dapat melihat medan untuk **Source Code** akan dipaparkan seperti pada **Rajah 3.1**. Secara **default**, medan ini mempunyai **pre-coding** yang ditala oleh **PROTEUS**. Untuk contoh litar kali ini, **pre-coding** tersebut akan dipadamkan dan diganti dengan **coding** untuk litar contoh ini.



**Rajah 3.3:** Medan *Source Code*

- c. Kemudian pada medan **Source Code**, masukkan **Coding** seperti pada **Rajah 3.2** ini.

```
// Pin assignments
const int switchPin1 = 2; // Pin for switch 1
const int switchPin2 = 3; // Pin for switch 2
const int ledPin1 = 4; // Pin for LED 1
const int ledPin2 = 5; // Pin for LED 2

// Variables to store switch states
int switchState1 = 0;
int switchState2 = 0;

void setup()
{
  // Set the switch pins as inputs
  pinMode(switchPin1, INPUT);
  pinMode(switchPin2, INPUT);

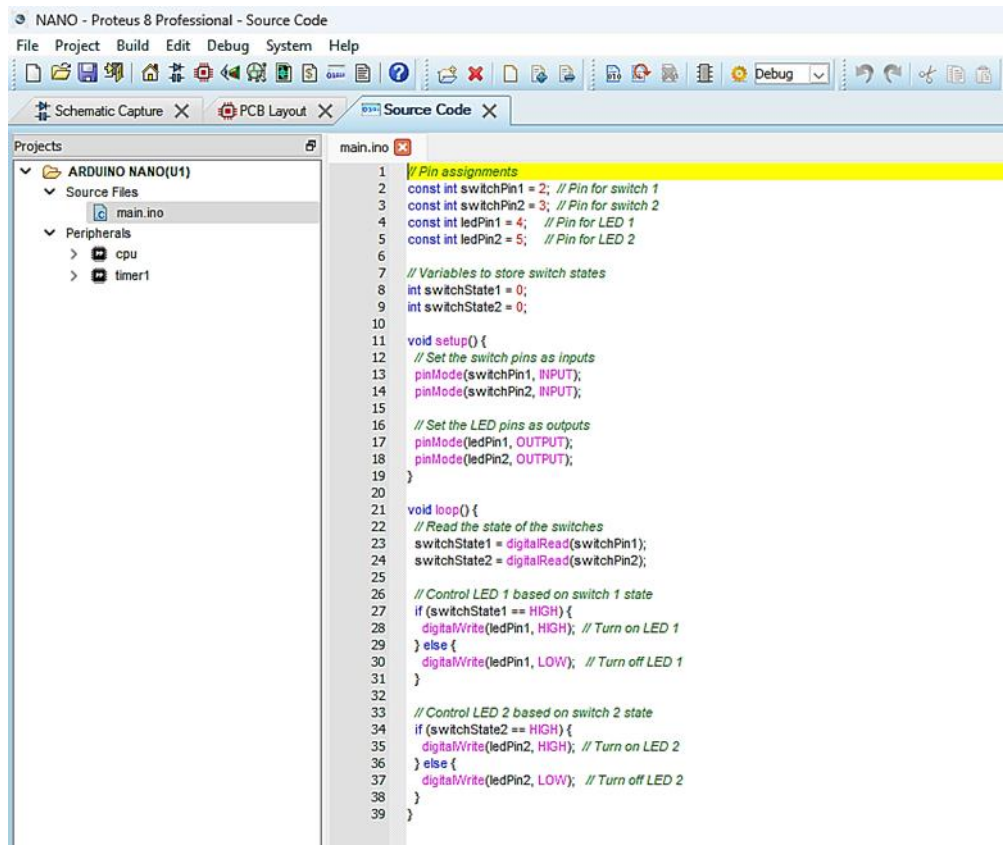
  // Set the LED pins as outputs
  pinMode(ledPin1, OUTPUT);
  pinMode(ledPin2, OUTPUT);
}
void loop() {
  // Read the state of the switches
  switchState1 = digitalRead(switchPin1);
  switchState2 = digitalRead(switchPin2);

  // Control LED 1 based on switch 1 state
  if (switchState1 == HIGH) {
    digitalWrite(ledPin1, HIGH); // Turn on LED 1
  } else {
    digitalWrite(ledPin1, LOW); // Turn off LED 1
  }

  // Control LED 2 based on switch 2 state
  if (switchState2 == HIGH) {
    digitalWrite(ledPin2, HIGH); // Turn on LED 2
  } else {
    digitalWrite(ledPin2, LOW); // Turn off LED 2
  }
}
```

**Rajah 3.4:** Coding Source Code

- d. Seterusnya, kita akan dapat melihat *Coding* yang telah selesai ditaip akan kelihatan seperti **Rajah 3.3**.

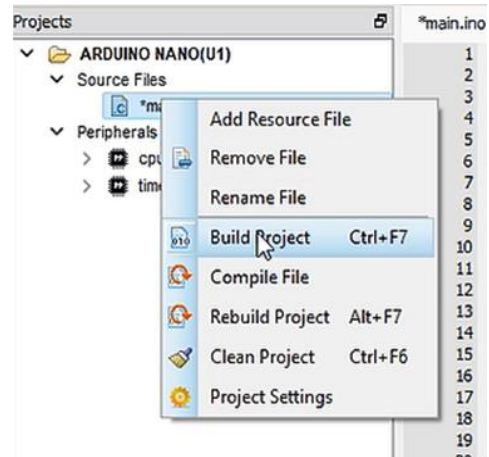


```
1 // Pin assignments
2 const int switchPin1 = 2; // Pin for switch 1
3 const int switchPin2 = 3; // Pin for switch 2
4 const int ledPin1 = 4; // Pin for LED 1
5 const int ledPin2 = 5; // Pin for LED 2
6
7 // Variables to store switch states
8 int switchState1 = 0;
9 int switchState2 = 0;
10
11 void setup() {
12 // Set the switch pins as inputs
13 pinMode(switchPin1, INPUT);
14 pinMode(switchPin2, INPUT);
15
16 // Set the LED pins as outputs
17 pinMode(ledPin1, OUTPUT);
18 pinMode(ledPin2, OUTPUT);
19 }
20
21 void loop() {
22 // Read the state of the switches
23 switchState1 = digitalRead(switchPin1);
24 switchState2 = digitalRead(switchPin2);
25
26 // Control LED 1 based on switch 1 state
27 if (switchState1 == HIGH) {
28 digitalWrite(ledPin1, HIGH); // Turn on LED 1
29 } else {
30 digitalWrite(ledPin1, LOW); // Turn off LED 1
31 }
32
33 // Control LED 2 based on switch 2 state
34 if (switchState2 == HIGH) {
35 digitalWrite(ledPin2, HIGH); // Turn on LED 2
36 } else {
37 digitalWrite(ledPin2, LOW); // Turn off LED 2
38 }
39 }
```

**Rajah 3.5:** Hasil *Coding* Medan *Source Code*

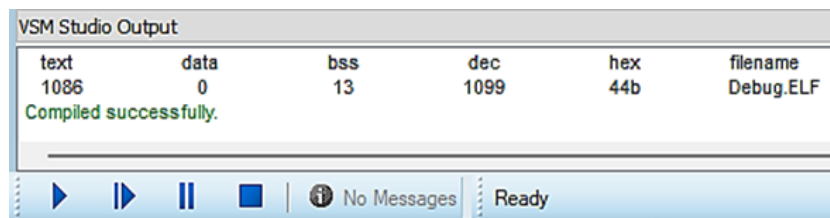
- e. Selepas *coding* dimasukkan, langkah seterusnya ialah *Build Project*. Tujuan utama kita melakukan *Build Project* adalah untuk menukarkan Bahasa pengaturcaraan kepada kod heksadesimal yang difahami oleh mikro-pengawal.

- f. Ia boleh dilakukan dengan mengetik `main.ino` yang terdapat pada bahagian tepi kiri skrin seperti **Rajah 3.4**.



**Rajah 3.6:** Pencarian *Build Project*

- g. Sekiranya *coding* yang ditaip tidak mempunyai sebarang ralat, mesej *Compiled successfully* bewarna hijau akan terpapar pada bahagian tepi-kiri bawah skrin seperti **Rajah 3.5** yang menandakan coding tersebut berjaya dibenamkan ke dalam mikro-pengawal dan sedia untuk digunakan.




**Rajah 3.7:** Mesej *Compiled successfully*

## Uji Kefahaman 3

1. Apakah yang perlu dilakukan sebelum kita memasukkan kod sumber ke dalam mikro-pengawal?
2. Mengapakah kita perlu melakukan prosedur **Build Project** selepas kod sumber dimasukkan?
3. Bagaimanakah cara untuk kita membuat perancangan untuk mendapat gambaran awal kod sumber yang bakal ditulis?

Imbas **QR-Code** untuk menyemak jawapan anda



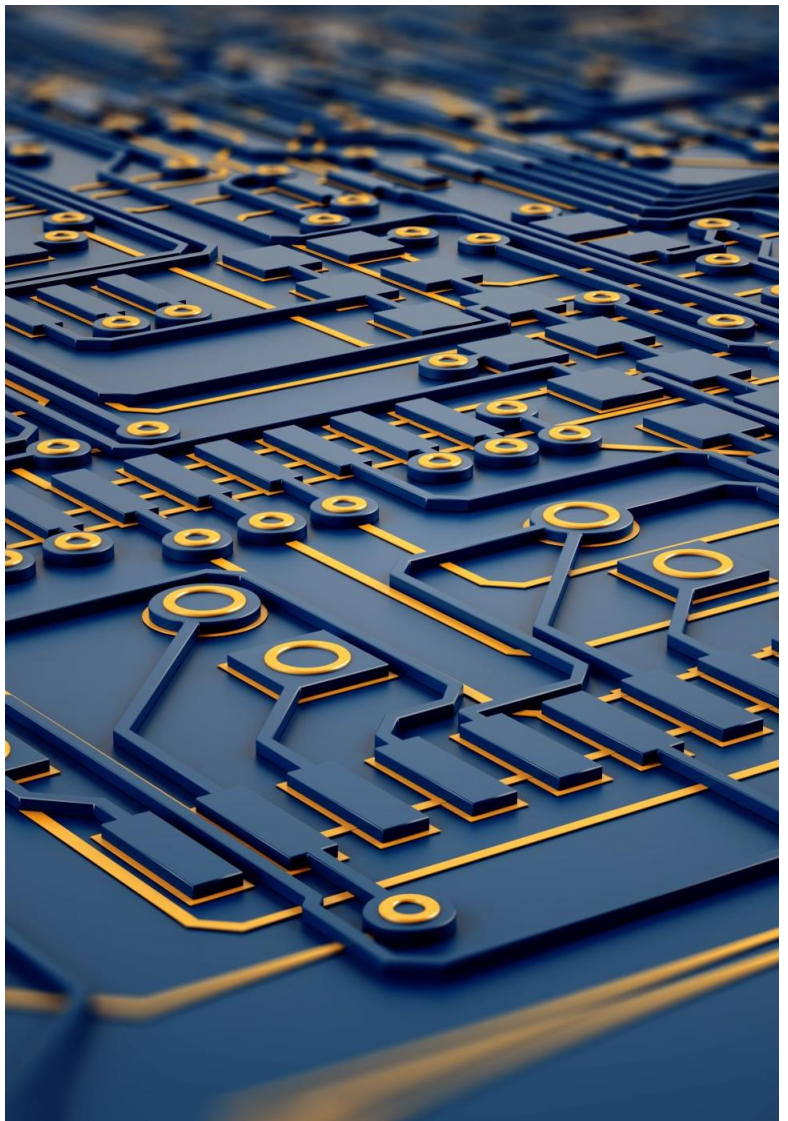


# **BAB 4 : SIMULASI LITAR**

Simulasi litar dalam Proteus Design Suite 8.7 adalah ciri yang membolehkan kita untuk mengesahkan serta menguji reka bentuk litar secara maya sebelum membuatnya secara fizikal.

Dengan adanya ciri ini, pengguna dapat menganalisis tingkah laku litar elektronik yang dihasilkan. Ia juga membolehkan kita untuk mengenal pasti sebarang masalah yang mungkin timbul dalam rekabentuk litar yang dihasilkan. Kita juga boleh memperhalusi reka bentuk litar untuk mendapatkan prestasi yang optimum.

---



## 4.0 Melakukan Simulasi Litar

Kembali semula kita kepada tab **schematic capture** untuk melakukan simulasi litar. Simulasi litar hanya boleh dilakukan setelah semua sambungan adalah betul dan *coding* telah dibenamkan ke dalam mikro-pengawal. Sekiranya syarat ini tidak dipatuhi maka perisian Proteus akan memaparkan mesej ralat kepada pengguna.

- a. Butang simulasi terletak pada bahagian kiri-bawah skrin yang terdiri daripada butang **PLAY**, **FRAME-BY-FRAME**, **PAUSE** dan **STOP** seperti yang ditunjukkan pada **Rajah 4.0**.



**Rajah 4.0:** Butang Simulasi

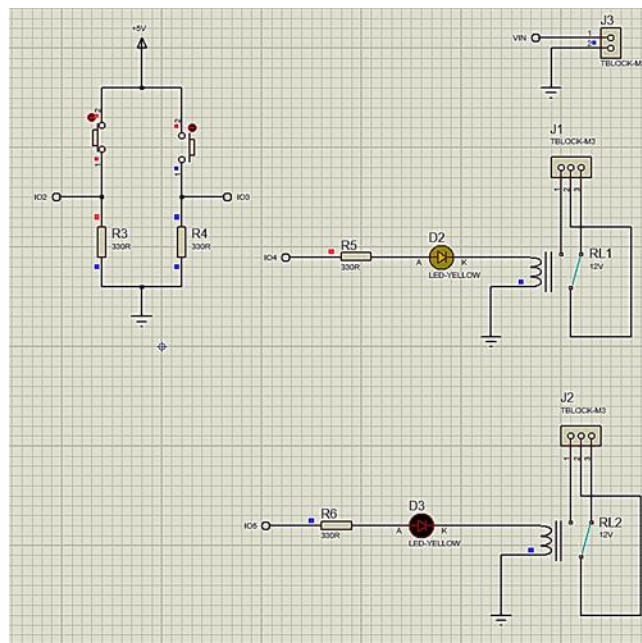
- b. Setiap butang mempunyai peranannya yang tersendiri semasa kita hendak melakukan simulasi litar.
  - i. Butang **PLAY** adalah berperanan untuk memulakan simulasi litar.
  - ii. Butang **FRAME-BY-FRAME** berperanan untuk melakukan animasi secara gerak perlahan.
  - iii. Butang **PAUSE** berperanan untuk menghentikan seketika simulasi sebelum ia disambung semula pada keadaan terakhir simulasi.
  - iv. Butang **STOP** pula berperanan untuk menghentikan simulasi litar.

- c. Semasa simulasi dijalankan butang *PLAY* akan bertukar kepada warna hijau. Paparan skrin semasa simulasi litar ditunjukkan seperti pada **Rajah 4.1**.



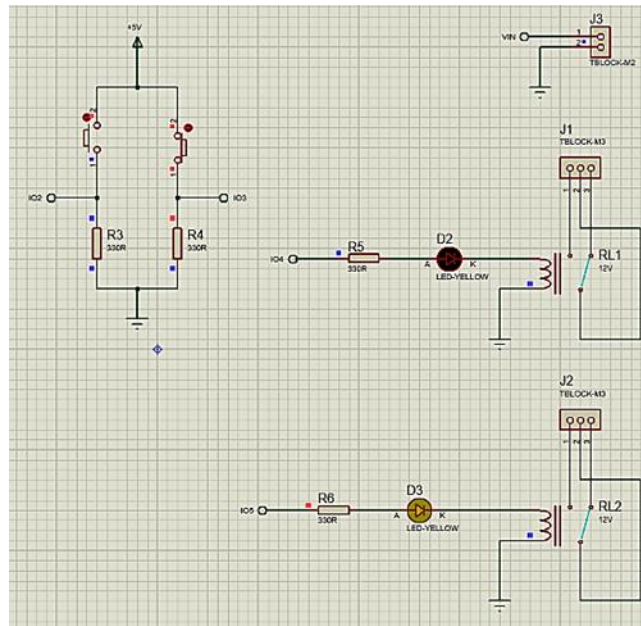
**Rajah 4.1:** Paparan Skrin Semasa Simulasi Litar

- d. Kita boleh mengedit semula litar yang dibuat sekiranya kita menemui dapatan baru setelah proses simulasi dijalankan.
- e. Daripada simulasi pada **Rajah 4.2** ini, kita dapat melihat kendalian litar dengan lebih jelas. Pada litar keluaran pertama, kita dapat melihat bahawa apabila **Suis 1** ditekan, maka **LED D2** akan menyala, manakala **Relay 1** akan berubah kedudukan daripada **Normally Close (NC)** kepada **Normally Open (NO)**.



**Rajah 4.2:** Pengendalian Litar Semasa S1 On

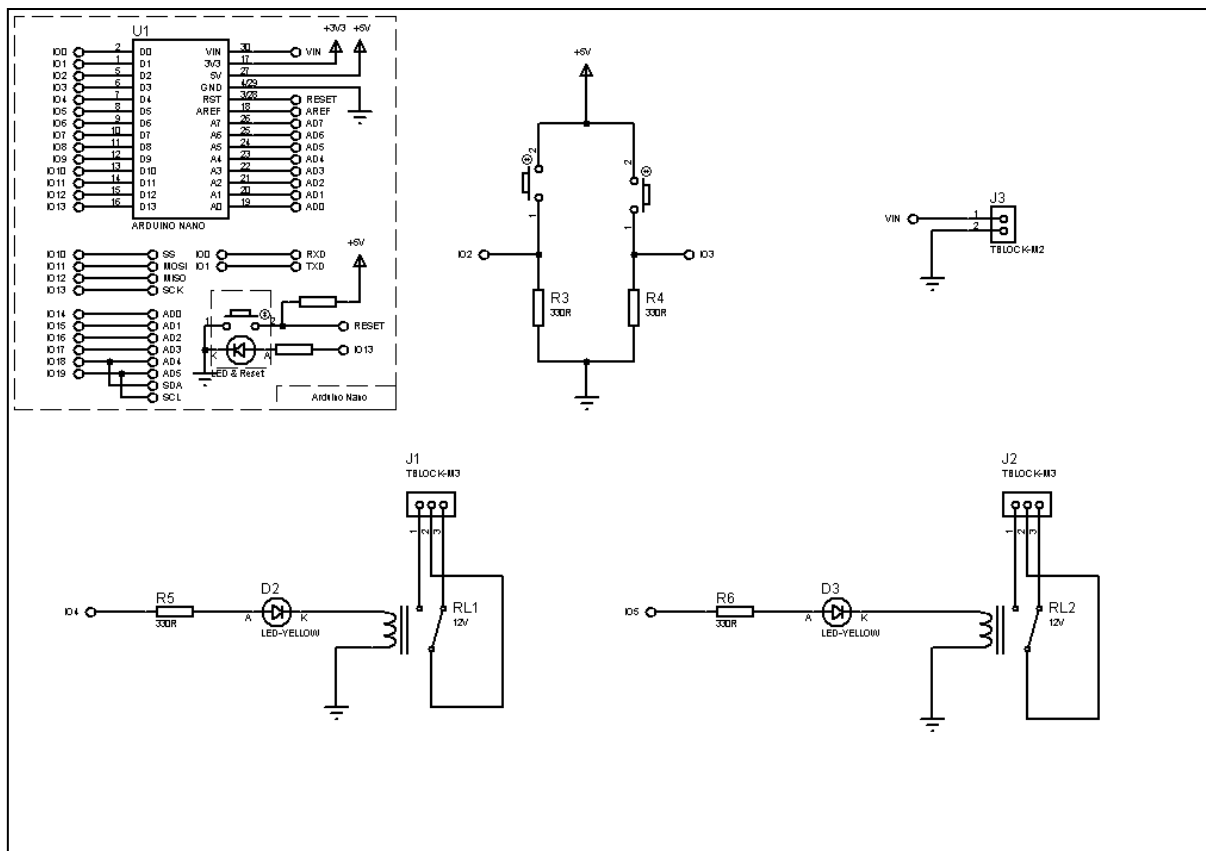
- f. Begitu juga dengan litar keluaran kedua pada **Rajah 4.3**, iaitu apabila **Suis 2** ditekan, maka **LED D3** akan menyala, manakala **Relay 2** akan berubah kedudukan daripada **Normally Close (NC)** kepada **Normally Open (NO)**.



**Rajah 4.3:** Pengendalian Litar Semasa S1 On

## Uji Kefahaman 4

1. Dengan menggunakan kaedah langkah demi langkah. Lukiskan litar di bawah ini menggunakan perisian PROTEUS.



2. Senaraikan komponen-komponen elektronik yang digunakan dalam litar di atas.



Imbas **QR-Code** untuk menyemak jawapan anda



# **BAB 5 : MEMBUAT PCB LAYOUT**

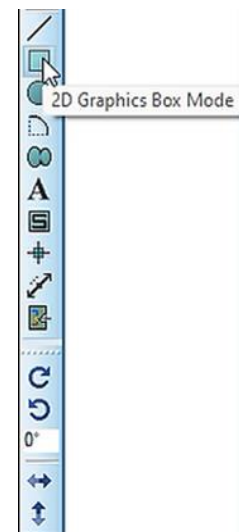
# PCB layout

Susun atur PCB adalah peringkat penting dalam proses reka bentuk elektronik. Pada peringkat ini, litar skematik berkonsep akan diubah menjadi perwakilan fizikal untuk pembuatan. Proteus Design Suite 8.7 menawarkan modul susun atur PCB yang komprehensif dan mesra pengguna yang memberi kuasa kepada pengguna untuk membuat PCB gred profesional dengan lebih mudah.

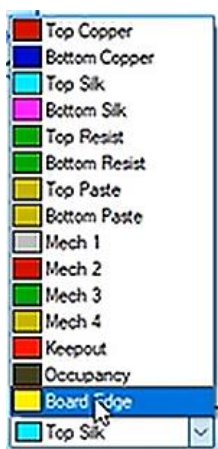
## 5.0 Menetapkan Dimensi Papan Litar

Untuk langkah seterusnya, kita akan membuat PCB *layout* berdasarkan kepada litar skematik yang telah dilukis. Di dalam tab PCB *layout*, kita akan melakukan beberapa tetapan kepada PCB tersebut supaya ia mengikut spesifikasi yang ditetapkan. Sebelum meletakkan komponen di atas papan litar, adalah penting untuk kita menetapkan saiz papan litar terlebih dahulu dengan mengikut langkah-langkah berikut.

- a. Pada tab **PCB layout**, halakan penuding kepada ikon **2D graphic box mode** yang terletak pada sebelah kiri skrin seperti pada **Rajah 5.0**. Kemudian ketik bentuk segiempat.



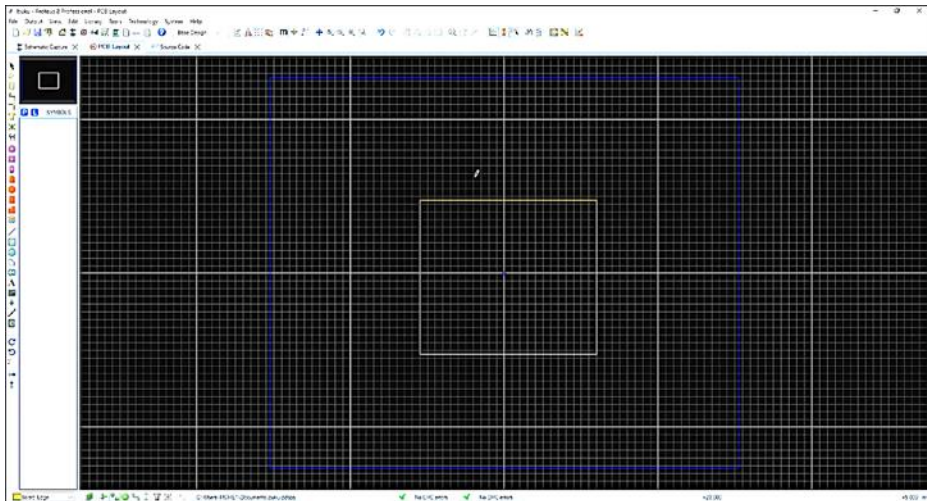
Rajah 5.0: Ikon *2D Graphic Box Mode*



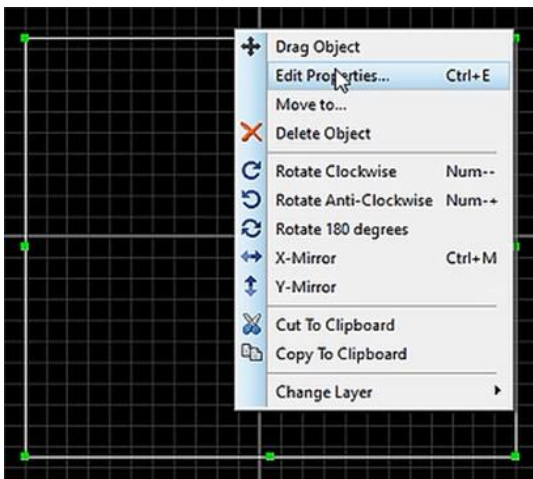
Rajah 5.1: Paparan *Layer Selector*

- b. Selepas itu, **Rajah 5.1** halakan pula penuding kepada **layer selector** di bawah-kiri skrin, dan ketik **Board Edge**.

- c. Kemudian, lukiskan bentuk segiempat tepat tadi pada ruangan kanvas. Pada peringkat ini, kita bebas untuk membuat bentuk pada ukuran bebas. Pastikan bentuk segiempat yang terhasil tadi mestilah garisan kuning menandakan lapisan **Board Edge** telah dipilih dengan betul seperti **Rajah 5.2**.



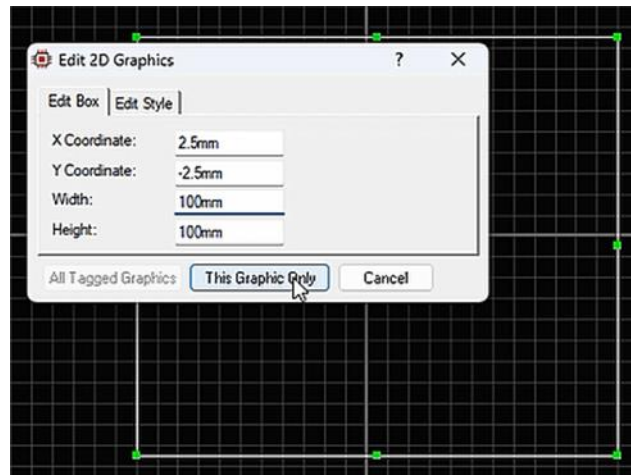
**Rajah 5.2:** Pemelihan Lapisan *Board Edge*



**Rajah 5.3:** Penetapan Saiz Papan PCB

- d. Langkah seterusnya pada **Rajah 5.3**, adalah menetapkan saiz papan kepada ukuran yang dikehendaki dalam litar contoh, ukuran yang akan disetkan ialah 100mm x 100mm. Maka, kita perlu klik-kanan pada garisan bentuk segiempat tadi untuk mengeluarkan sub-menu. Kemudian, ketik pada **Edit Properties**.

- e. Kemudian, **Rajah 5.4** kita akan dibawa kepada tettingkap **Edit 2D Graphics**. Pada tab **Edit Box** kita setkan lebar dan tinggi papan litar tersebut. Pada medan **Width**: masukkan nilai **100mm** dan medan **Height**: juga masukkan nilai **100mm**. Kemudian klik butang **This Graphic Only**.

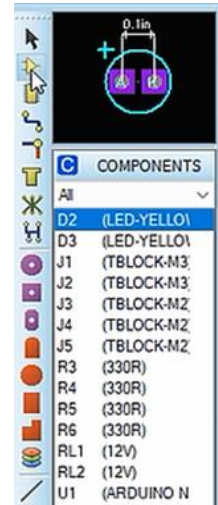


**Rajah 5.4:** Tettingkap *Edit 2D Graphics*

- f. Selepas itu, kita akan mendapati terdapat satu segiempat sama dengan garisan berwarna kuning akan terpapar pada ruangan kanvas. Ini menandakan tetapan saiz pada papan litar telah berjaya dilakukan.

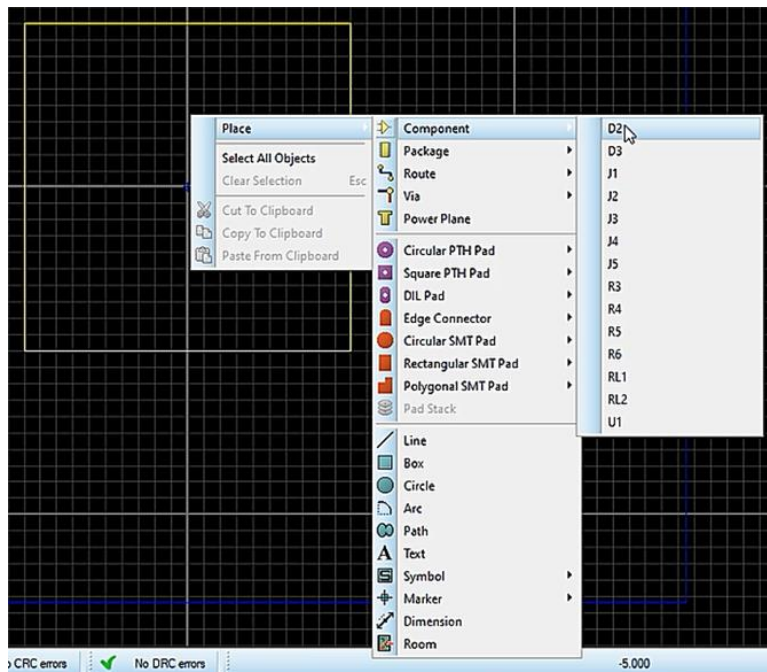
## 5.1 Memasang Komponen Pada Papan Litar

- a. Proses memasang komponen pada litar, ia boleh dilakukan dengan menetik semula ikon **Component Mode** yang terletak pada sebelah kiri-atas skrin seperti **Rajah 5.5**. Senarai komponen akan terpapar secara automatik setelah ikon **Component Mode** diketik.



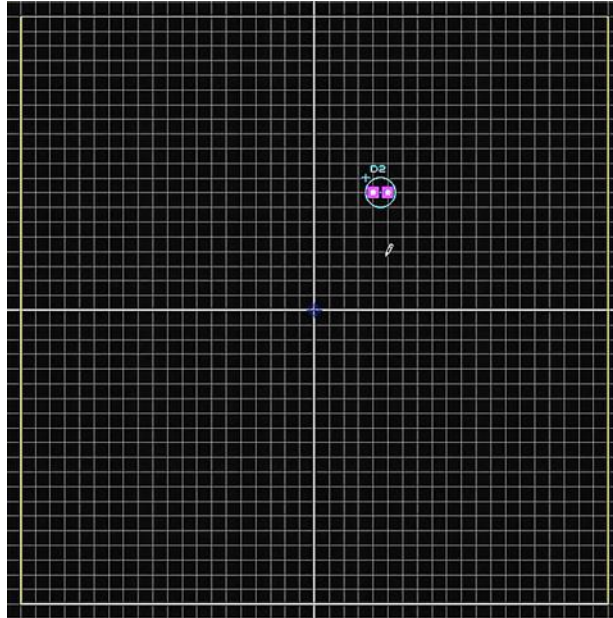
Rajah 5.5: Ikon *Component Mode*

- b. Kemudian pada ruangan kanvas seperti **Rajah 5.6**, klik-kanan untuk melihat sub-menu. Ketik pada **Place**, kemudian ketik **Component** dan akhirnya senarai komponen akan terpapar. Dalam contoh ini komponen yang dipilih ialah **Led D2**.



Rajah 5.6: Sub-menu *Place*

- c. Kemudian letakkan Led D2 tadi ke dalam ruangan papan litar seperti yang ditunjukkan pada **Rajah 5.7**.

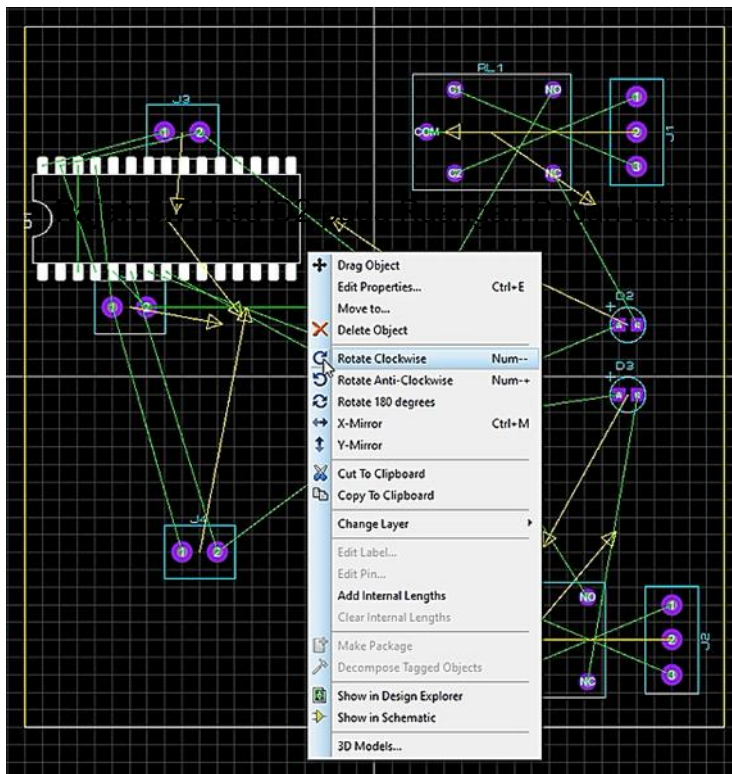


**Rajah 5.7:** Led D2 pada Ruangan Papan Litar

- d. Untuk memasukkan komponen yang lain pula kita hanya perlu mengulangi proses di atas sehingga selesai.

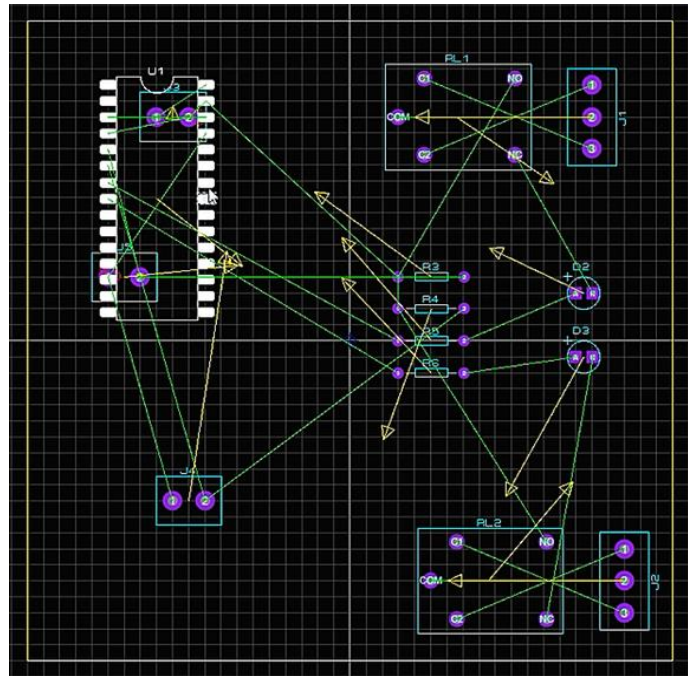
## 5.2 Menyusun Kedudukan Komponen.

- a. Selepas meletakkan semua komponen pada papan litar, kita perlu melakukan sedikit proses penyusunan iaitu menyusun semula orientasi komponen. Sebagai contoh, mikro-pengawal **Arduino NANO** akan diubah orientasinya daripada melintang kepada menegak.
  
- b. Proses mengubah orientasi boleh dilakukan dengan merujuk pada **Rajah 5.8** iaitu klik-kanan untuk memaparkan sub-menu. Pada ruangan sub-menu tersebut, ketik pada **Rotate Clockwise**.



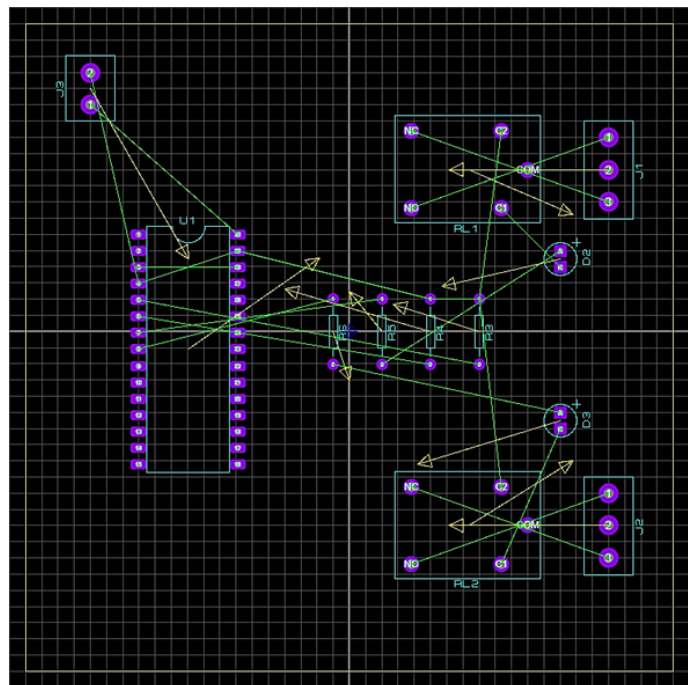
Rajah 5.8: Cara Mengubah Orientasi

- c. Dapat diperhatikan orientasi **Arduino NANO** telah pun bertukar kepada menegak selepas tetapan tersebut dilakukan seperti **Rajah 5.9**. Kita boleh mengulang proses tersebut untuk komponen-komponen yang lain juga.



**Rajah 5.9:** Hasil Orientasi Arduino NANO

- d. Langkah seterusnya, **Rajah 5.10** ialah menyusunatur kedudukan komponen pada kedudukan yang sesuai. Kedudukan komponen yang di atur mestilah berada dalam ruang lingkup segi empat berwarna kuning.



**Rajah 5.10:** Penyusunan Komponen pada Board Edge

### 5.3 Membuat Trek Litar Pada PCB

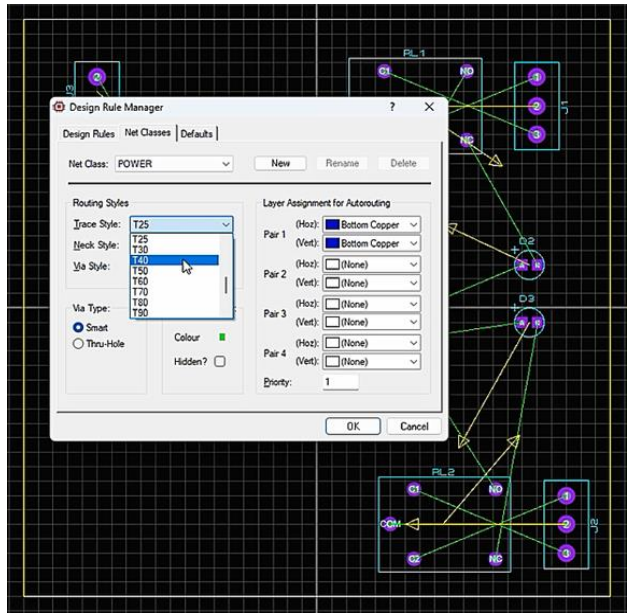
Seterusnya, setelah kita berpuas hati dengan kedudukan komponen yang telah disusun, maka bolehlah kita membuat tetapan trek litar pada PCB dengan mengikuti langkah berikut. Secara lazimnya, trek papan litar terdiri daripada trek **POWER** dan **SIGNAL**.

- a. Pertama sekali, **Rajah 5.11** halakan penuding kepada ikon **Design Rule Manager** yang terletak pada bahagian kanan-atas skrin.

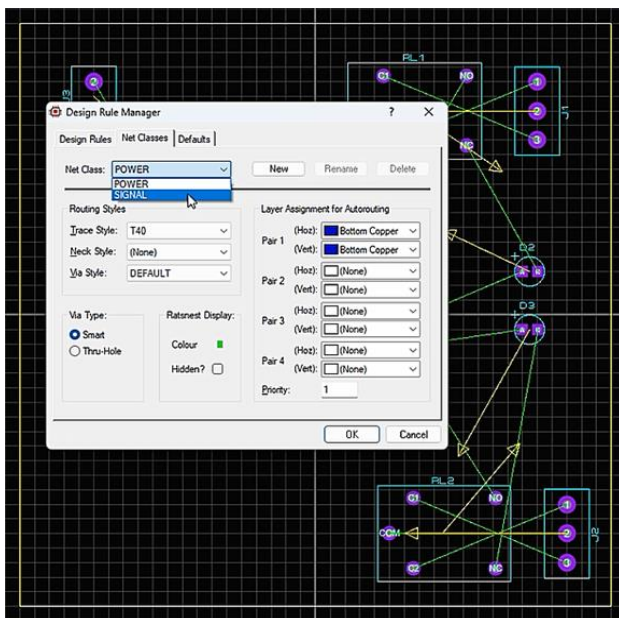


**Rajah 5.11:** Ikon *Design Rule Manager*

- b. Berdasarkan pada **Rajah 5.12** membawa kita pada tettingkap **Design Rule Manager**. Klik pada tab **Net Classes**. Pastikan Medan **Net Class : POWER** dipilih. Kemudian setkan **Trace Style: T40**. Tetapan ini merupakan tetapan untuk ketebalan trek **POWER** di dalam litar tersebut.



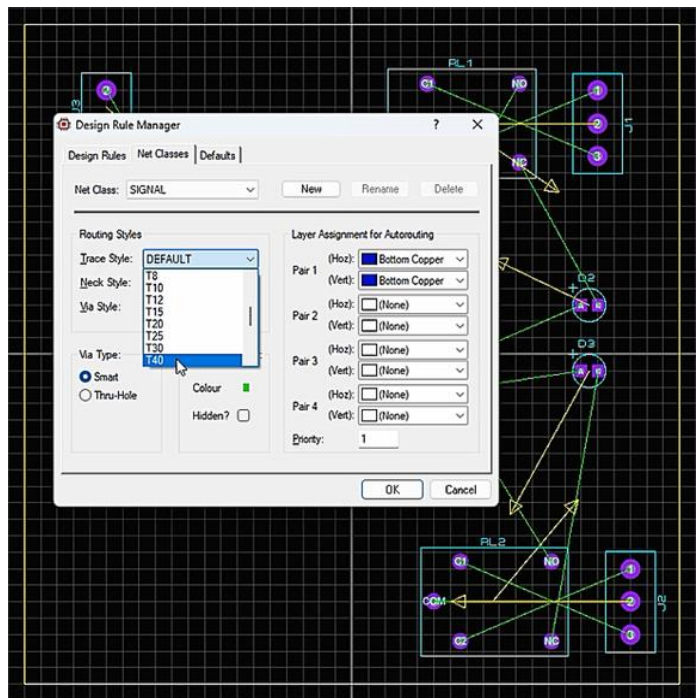
**Rajah 5.12:** Penetapan Medan **Net Class : POWER**



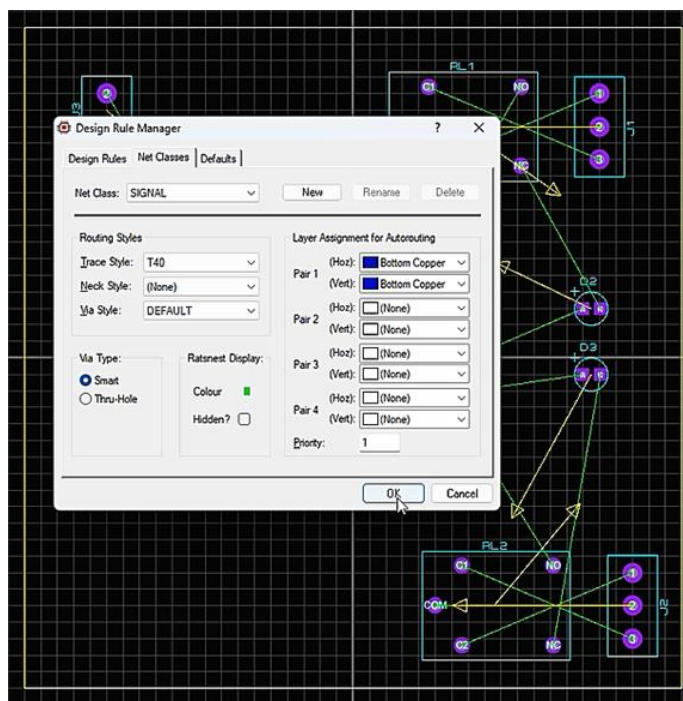
**Rajah 5.13:** Penetapan Medan **Net Class : SIGNAL**

- c. Langkah seterusnya ialah pada **Rajah 5.13** iaitu melakukan tetapan pada medan **Net Class : SIGNAL**. Halakan semula penuding pada medan **Net Class** dan pilih **SIGNAL**.

d. Kemudian **Rajah 5.14** setkan medan **Trace Style** kepada nilai **Trace Style : T40**. Tetapan ini merupakan tetapan untuk ketebalan trek **SIGNAL** di dalam litar tersebut.



**Rajah 5.14:** Penetapan Medan *Trace Style*



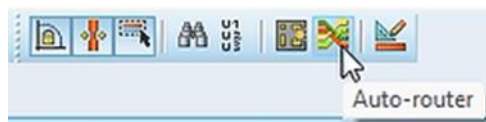
**Rajah 5.15:** Hasil Penetapan *Design Rule Manager*

e. Setelah berpuas hati dengan tetapan yang dilakukan, ketik **OK** seperti pada **Rajah 5.15**. Selepas melakukan semua tetapan yang dinyatakan tadi, maka kita boleh pergi ke peringkat seterusnya untuk melakukan proses **track routing**.

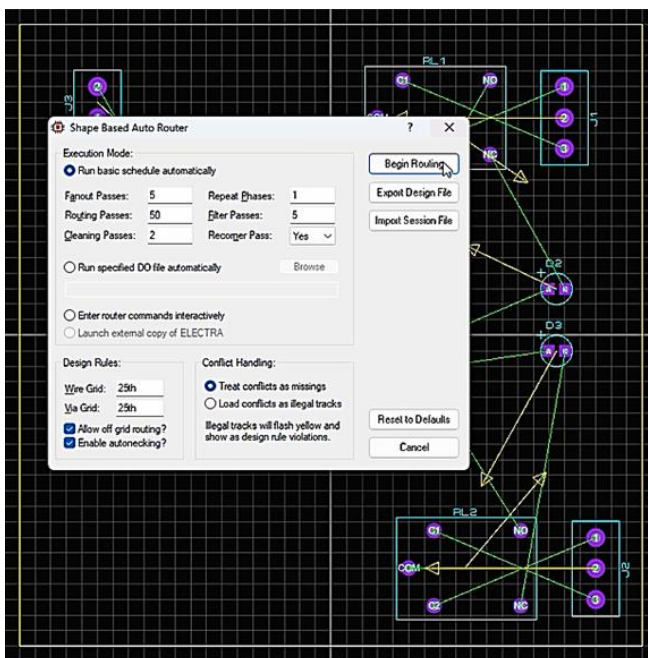
## 5.4 Menetapkan Laluan Trek (Track-Routing)

Peringkat seterusnya ialah membuat tetapan laluan trek pada papan litar. Tetapan laluan ini boleh dilakukan secara manual atau pun secara automatik. Walau bagaimanapun, untuk litar contoh ini kita akan menggunakan tetapan automatik. Melalui tetapan automatik, komputer akan membuat keputusan untuk merintis laluan yang terbaik dengan ralat yang minimum atau tanpa ralat. Tetapan laluan trek boleh dilakukan dengan mengikuti langkah yang dinyatakan di bawah ini.

- a. Halakan penuding kepada ikon **auto-router** yang teletak pada bahagian kanan-atas skrin seperti pada **Rajah 5.16**.



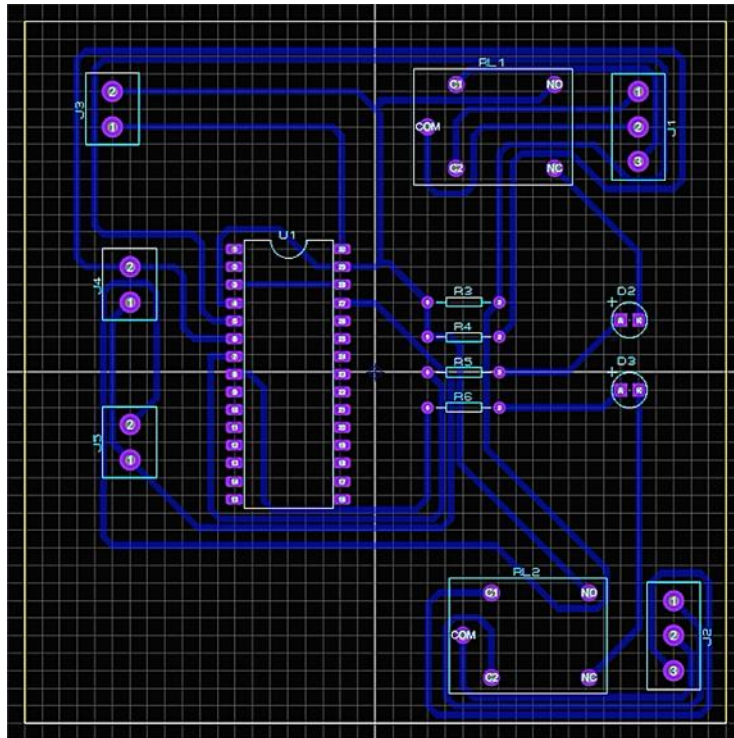
**Rajah 5.16:** Ikon *Auto Router*



**Rajah 5.17:** Tetingkap *Shape Based Auto Router*

- b. Kemudian kita akan dibawa kepada tetingkap **Shape Based Auto Router** merujuk pada **Rajah 5.17**. Pada tetingkap ini kita perlu mengetik butang **Begin Routing** untuk memulakan proses tetapan laluan trek secara automatik.

- c. Selepas butang **Begin Routing** ditekan, komputer akan menjana laluan secara automatik. Biarkan sehingga proses ini selesai. **Rajah 5.18** memaparkan hasil laluan trek litar elektronik secara automatik.



**Rajah 5.18:** Hasil Laluan Trek Litar Elektronik Secara Automatik

- d. Semak skrin bawah-tengah skrin. Pastikan tiada berlakunya **CRC error** dan **DRC error** pada trek yang dihasilkan seperti **Rajah 5.19**.



**Rajah 5.19:** Paparan **DRC error**

- e. Sekiranya terdapat ralat **CRC** dan **DRC** maka proses tersebut perlu dilakukan semula mengikut langkah-langkah yang dinyatakan tadi.

## 5.5 Memasukkan Teks Ke Atas Papan Litar

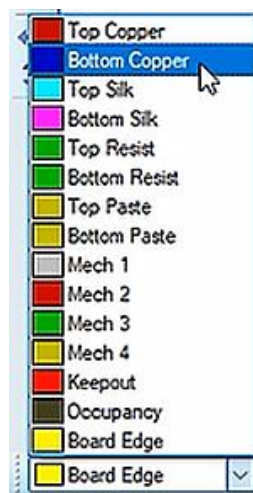
Proses memasukkan teks merupakan satu pilihan dan ianya tidak wajib. Teks ini merupakan maklumat asas seperti nama litar atau pun nama institusi.

- a. Untuk memasukkan teks ke dalam papan litar, kita perlulah mengetik ikon teks yang terletak pada bahagian kiri-tepi skrin seperti pada **Rajah 5.20**.



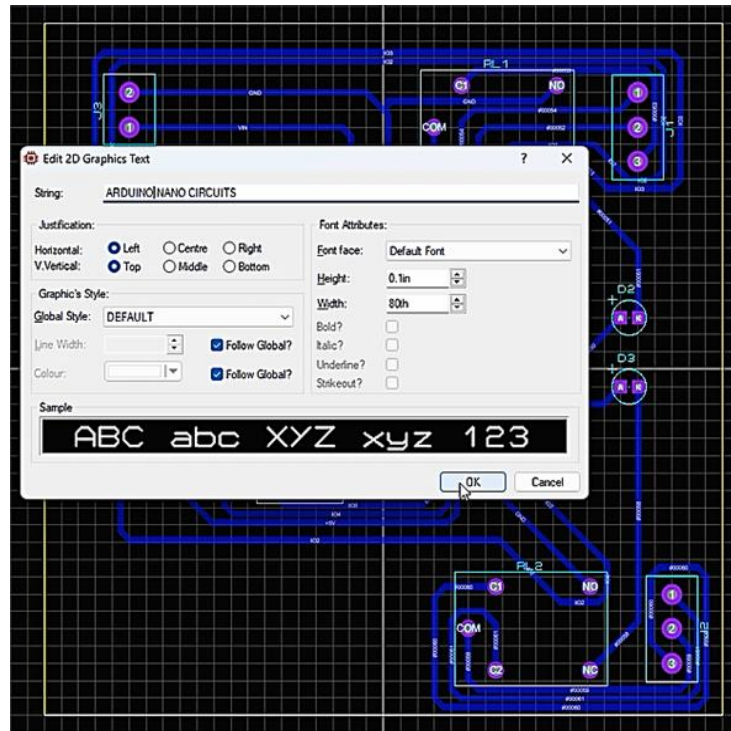
**Rajah 5.20:** Paparan 2D *Graphics Text Mode*

- b. Kemudian merujuk pada **Rajah 5.21**, tetapkan lapisan kepada **Bottom Copper**. Lapisan **Bottom Copper** diwakili dengan warna biru.



**Rajah 5.21:** Tetapan Lapisan *Bottom Copper*

- c. Kemudian, kita akan dibawa kepada tetingkap **Edit 2D Graphics Text** seperti pada **Rajah 5.22**. Pada medan *string* taip teks yang dikehendaki. Dalam contoh ini teks yang dimasukkan adalah **ARDUINO NANO CIRCUITS**.



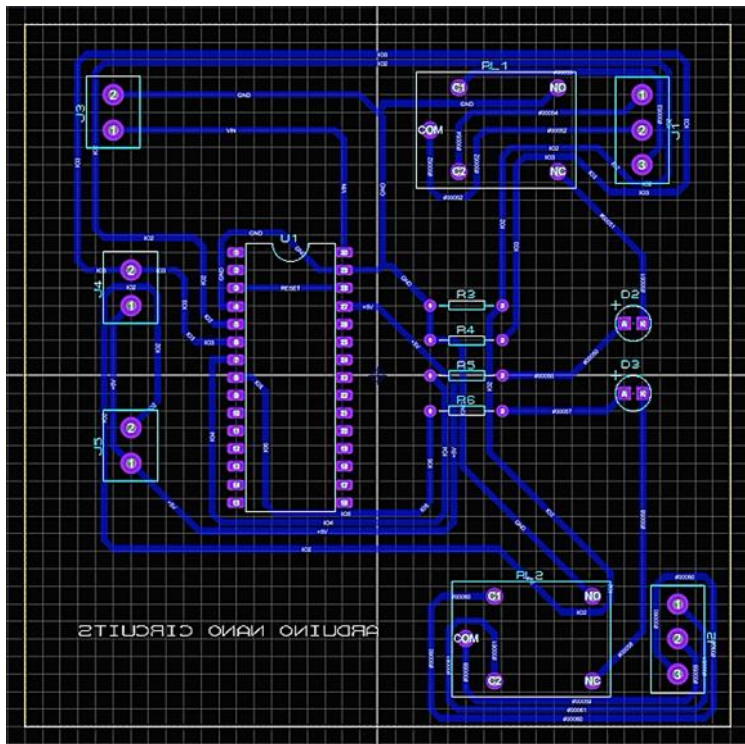
**Rajah 5.22:** Tetingkap *Edit 2D Graphics Text*

- d. Kita boleh juga mengubah semua tetapan yang ada dalam tetingkap **Edit 2D Graphics Text** pada ruangan **Justification**, **Font Attribute** dan **Graphic Style**.
- e. Setelah berpuas hati dengan tetapan yang dilakukan maka kita boleh mengetik butang **OK**.



- f. Ketik pada butang **Selection Mode** seperti **Rajah 5.23** untuk menetapkan kedudukan teks pada papan litar. Butang **Selection Mode** merupakan tanda anak panah paling atas sekali yang terletak pada bahagian kiri-atas skrin.

**Rajah 5.23:** Butang *Selection Mode*



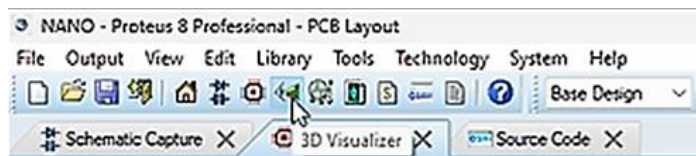
**Rajah 5.24:** Hasil Proses Memasukkan Teks di atas Papan Litar Bercetak

- g. Kedudukan ini boleh dilakukan dengan menyeret teks tersebut kepada kedudukan yang dikehendaki. **Rajah 5.24** merupakan hasil proses memasukkan teks di atas papan litar bercetak.

## 5.6 Pralihat Papan Litar

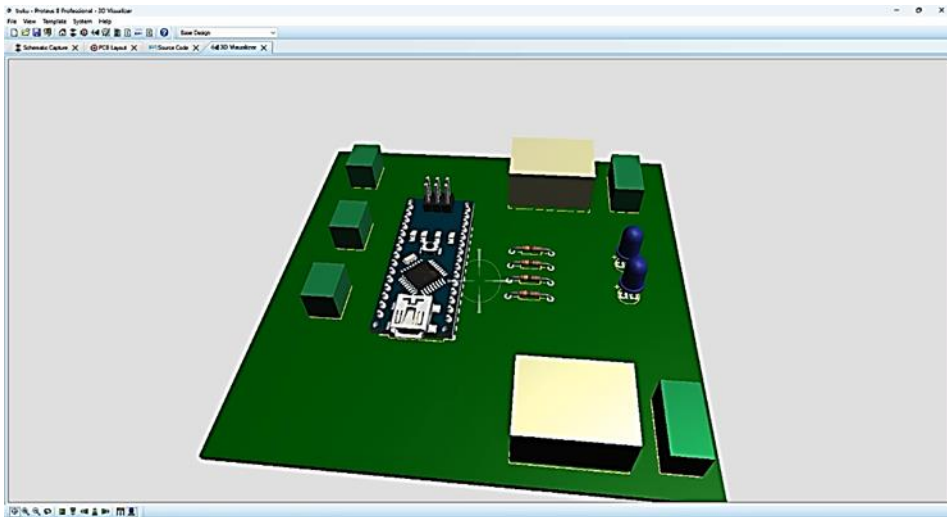
Setelah selesai melakukan proses memasukkan teks, maka bolehlah kita untuk melakukan pra-lihat papan litar yang telah dihasilkan. Tujuan utama kita melakukan proses pra-lihat ini adalah untuk memastikan bahawa susunan komponen kita adalah berada pada jarak yang betul dan munasabah. Ciri utama yang perlu diperhatikan ialah setiap komponen tidak bertindih antara satu sama lain sehingga menyukarkan proses pematerian komponen pada papan litar sebenar. Untuk itu kita boleh melakukan pralihat papan litar mengikut prosedur berikut.

- a. Halakan penuding kepada ikon **3D Visualizer** yang terletak pada bahagian atas-kiri skrin merujuk pada **Rajah 5.25**.



**Rajah 5.25:** Ikon 3D Visualizer

- b. Selepas itu kita akan dibawa kepada tettingkap **3D Visualizer** seperti **Rajah 5.26**. Pada ruangan ini kita akan dapat melihat papan litar kita secara tiga dimensi.



**Rajah 5.26:** Tettingkap 3D Visualizer

- c. Pada paparan tettingkap ini, kita boleh mengubah sudut pandangan dari pelbagai arah dengan menekan butang kawalan yang terdapat pada bahagian kiri-bawah skrin. Paparannya seperti pada **Rajah 5.27**.



**Rajah 5.27:** Paparan Mengubah Sudut Pandangan

- d. Butang kawalan ini membolehkan kita untuk melihat dari pandangan atas, bawah, depan dan belakang serta pandangan isometrik. Kita juga boleh melakukan proses **zoom-out** dan **zoom-in** bagi mendapatkan butiran yang lebih jelas.

## 5.7 Menetapkan Keluaran Litar Akhir

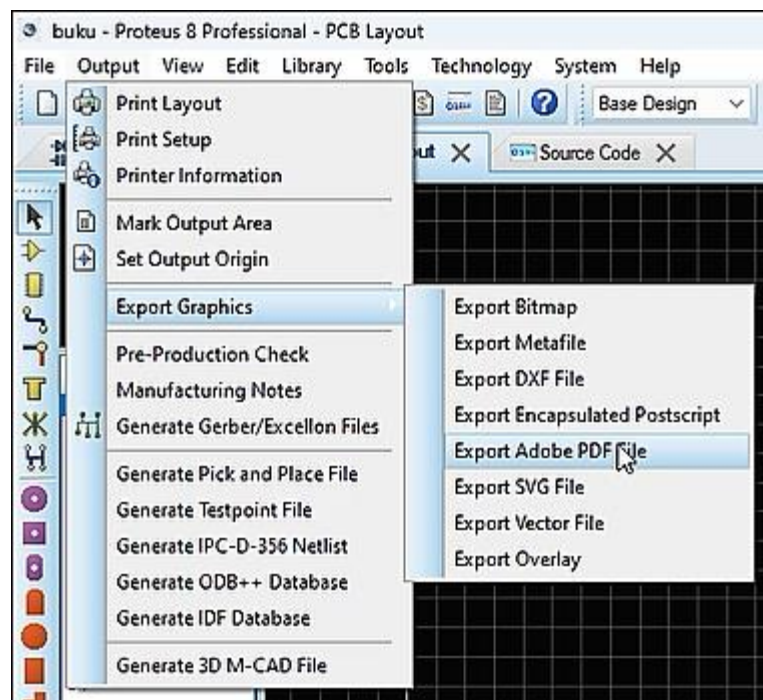
Keluaran litar akhir bagi litar yang telah siap dilukis boleh ditetapkan sama ada untuk kegunaan proses **etching** atau pun menggunakan **CNC**. Untuk litar kali ini proses keluaran adalah ditetapkan untuk kegunaan proses **Etching**. Tetapan ini boleh dilakukan dengan menggunakan langkah berikut.

- a. Ketik **Output** yang terletak pada bahagian kiri-atas skrin seperti yang ditunjukkan pada **Rajah 5.28**.



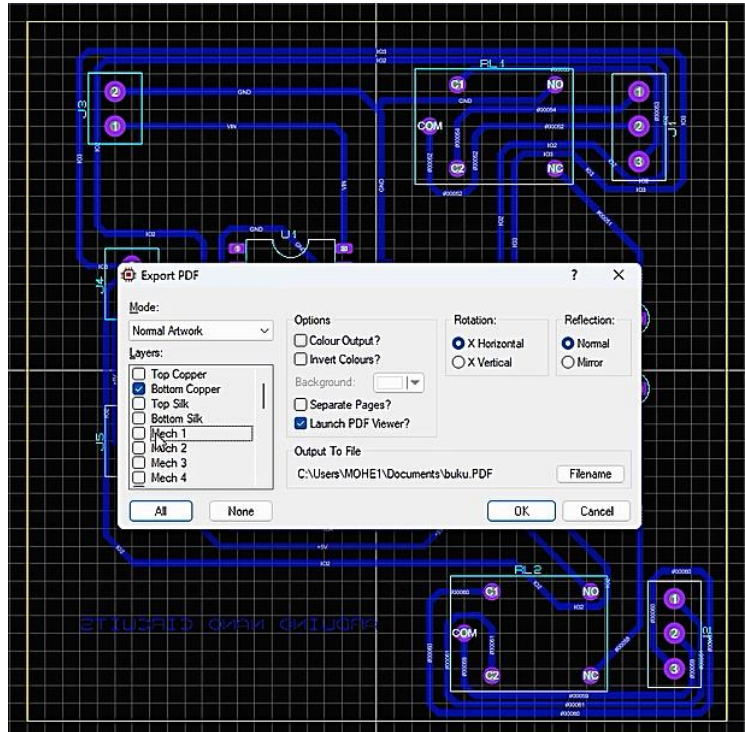
Rajah 5.28: Paparan *Output*

- b. Kemudian ketik **Export Graphics** seperti pada **Rajah 5.29**, diikuti dengan mengetik **Export Adobe PDF File**.

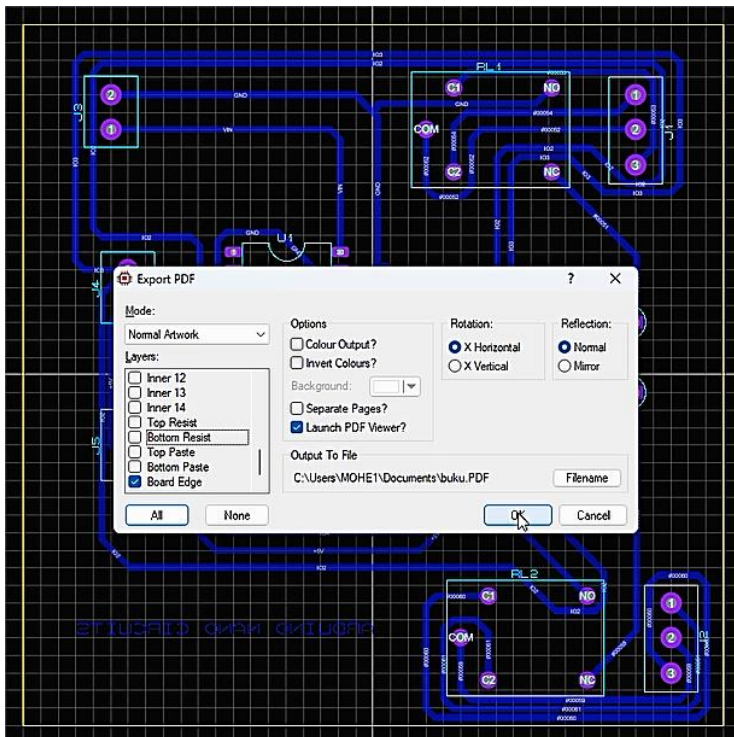


Rajah 5.29: Tetingkap *Output*

- c. Kemudian pada **Rajah 5.30**, kita akan dibawa kepada tettingkap **Export PDF**. Pada tettingkap ini kita perlu memilih lapisan yang akan dicetak nanti.



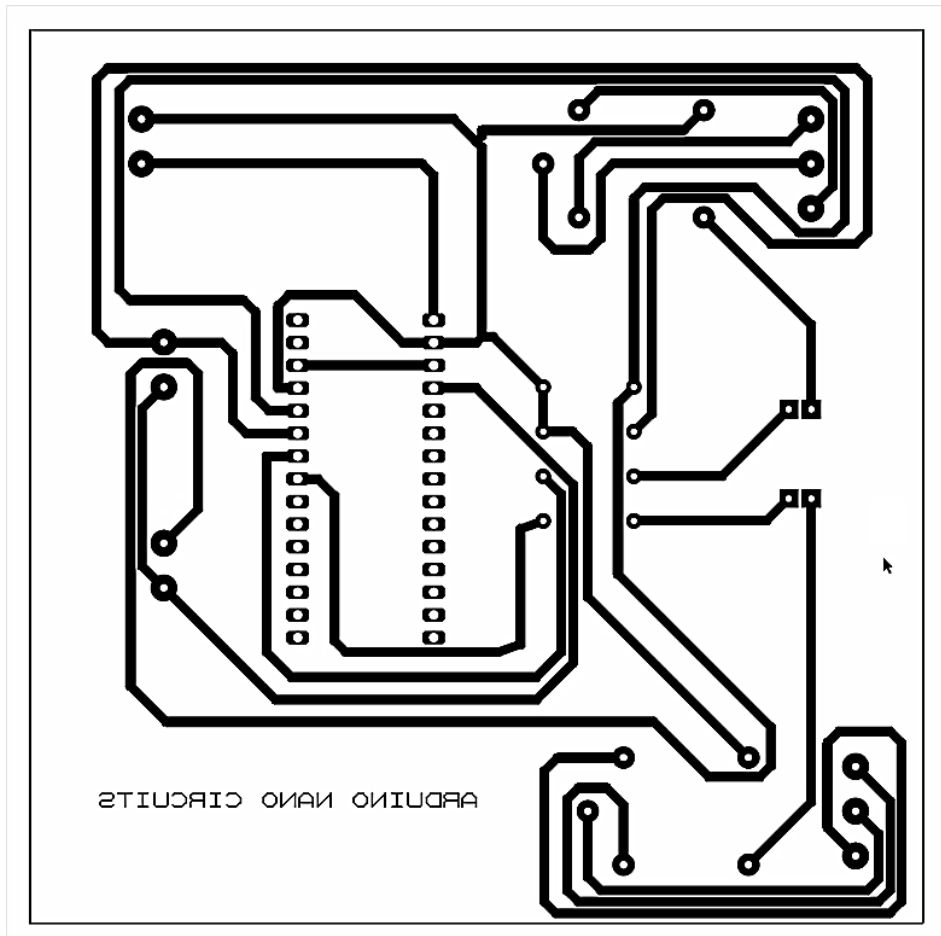
**Rajah 5.30:** Tettingkap *Export PDF*



**Rajah 5.31:** Medan *Layer*

- d. Pemilihan lapisan ini boleh dilakukan pada medan **layer** dengan mengetik **Bottom Copper** dan **Board Edge** sahaja seperti pada **Rajah 5.31**. Selepas itu, tekan butang **OK**.

- e. Kemudian **output** dalam format PDF akan dijana secara automatik. Fail PDF tersebut mestilah dicetak menggunakan pencetak laser dan dicetak di atas kertas *sticker*.



Rajah 5.32: Keluaran Litar Akhir

- f. Jika diperhatikan pada **Rajah 5.32**, perkataan **ARDUINO NANO CIRCUITS** tadi berada dalam keadaan terbalik. Ini menandakan keluaran kita adalah betul. Tulisan tersebut akan menjadi normal setelah proses transfer menggunakan mesin lamina dilakukan.

## Uji Kefahaman 5

1. Nyatakan jenis tetapan yang perlu dilakukan untuk menentukan saiz papan litar.
2. Bagaimanakah cara untuk kita memeriksa kedudukan komponen yang disusun pada papan litar tidak berlaku pertindihan?
3. Apakah perkara yang perlu diambil berat semasa membuat **track** di dalam PCB?

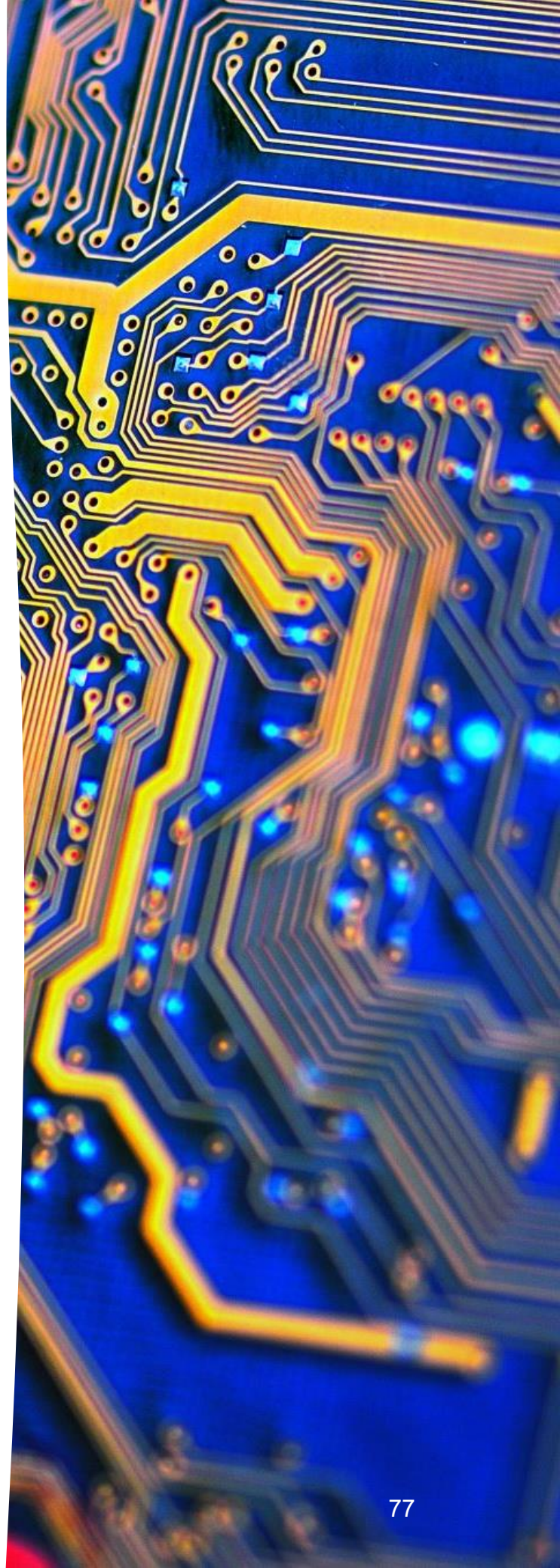
Imbas **QR-Code** untuk menyemak jawapan anda





BAB 6 : MEMBUAT  
PAPAN LITAR TERCETAK

Merancang dan mencipta Papan Litar Bercetak (PCB) adalah merupakan aspek asas kepada elektronik moden. Ianya melibatkan proses menterjemahkan litar skematik ke dalam susun atur fizikal yang berfungsi sebagai asas untuk peranti elektronik. Dengan mengatur komponen dan jejak penghalaan dengan teliti pada PCB, kita dapat memastikan sambungan elektrik dan integriti isyarat yang betul.

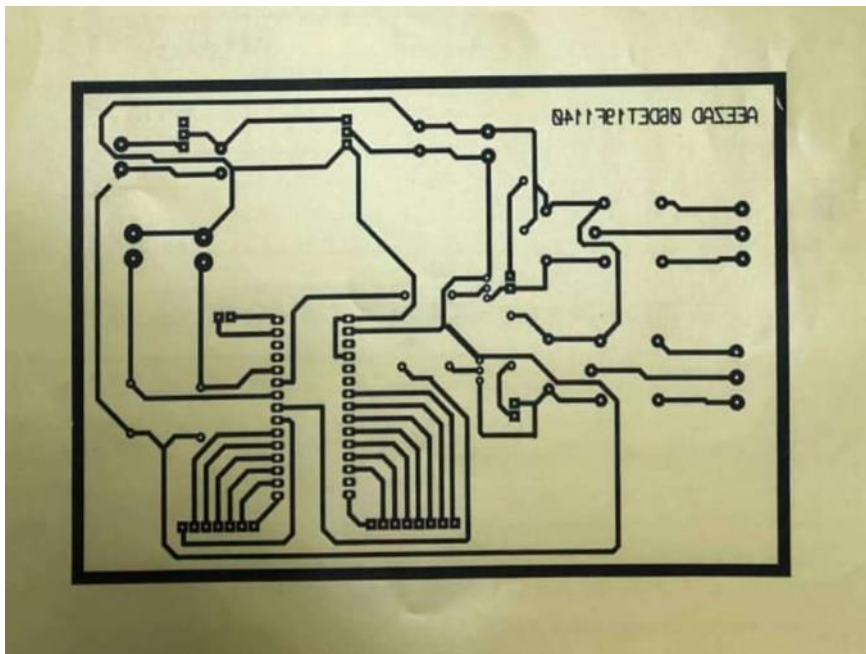


## 6.0 PENGENALAN

Proses seterusnya ialah membuat papan litar tercetak secara fizikal. Sebelum ini kita hanya dapat melihat papan tercetak yang dihasilkan hanya melalui skrin komputer sahaja. Untuk bab kali ini kita akan membincangkan cara menghasilkan Papan Litar Tercetak dengan menggunakan kaedah **Etching**. Prosedur pelaksanaan Penghasilan litar elektronik pada papan litar bercetak dinyatakan seperti langkah di bawah.

### 6.1 Mencetak Litar

- a. Keluaran grafik yang dijana daripada **PROTEUS DESIGN SUITE** disetkan kepada format PDF. Untuk membuka fail **output** tersebut kita perlulah menggunakan perisian pembaca PDF. Kemudian, barulah proses mencetak dapat dilakukan dengan menggunakan printer laser di atas kertas glos atau kertas stiker. Hasil cetakan **output** itu seperti pada **Rajah 6.0**.



**Rajah 6.0:** Hasil Cetakan *Output*

- b. Litar yang telah dicetak pada kertas stiker atau kertas glos perlu ditampalkan di atas papan litar seperti yang ditunjukkan pada **Rajah 6.1**



**Rajah 6.1:**  
Kaedah Penampalan di atas Papan Litar



**Rajah 6.2:** Kaedah *Iron-on*

- c. Selepas itu, kertas tersebut akan ditampal pada papan litar untuk melakukan proses **IRON-ON** seperti pada **Rajah 6.2**. Proses **IRON ON** ialah proses memindahkan litar dari kertas stiker atau kertas glos ke papan bercetak. Proses ini menggunakan mesin lamina yang ditetapkan pada suhu **150 darjah celcius** dengan kelajuan **4 rpm**.

## 6.2 Proses *Etching*

Proses *Etching* ialah proses pelunturan bahagian yang terdedah pada papan litar bercetak (PCB) dengan menggunakan asid **ferit klorida** yang kuat sehingga menghasilkan papan litar bercetak (PCB). Untuk melaksanakan proses *Etching* kita perlulah mengutamakan keselamatan dengan memakai Peralatan Pelindungan Diri (**PPE**) seperti pada **Rajah 6.3**. Ia bertujuan untuk mengurangkan risiko kemalangan dan kecederaan. Peralatan Pelindungan Diri (**PPE**) yang perlu dipakai ialah apron, pelitup muka, goggles, sarung tangan getah dan kasut keselamatan.



**Rajah 6.3:** Peralatan Pelindungan Diri (*PPE*)

- a. Proses *Etching* ini menggunakan mesin sembur berputar (**ROTARY SPRAY MACHINE**) pada paparan **Rajah 6.4**. Papan litar bercetak (PCB) akan didedahkan kepada cecair ferit klorida mengikut pelarasan yang sesuai.



**Rajah 6.4:** *Rotary Spray Machine*

- b. Tempoh proses **Etching** ialah satu minit 30 saat. Suhu asid pula akan dilaraskan pada suhu maksima iaitu 50 darjah Celsius meskipun proses **Etching** boleh dilaksanakan pada suhu 40 darjah Celsius. Paparan adalah seperti **Rajah 6.5**.



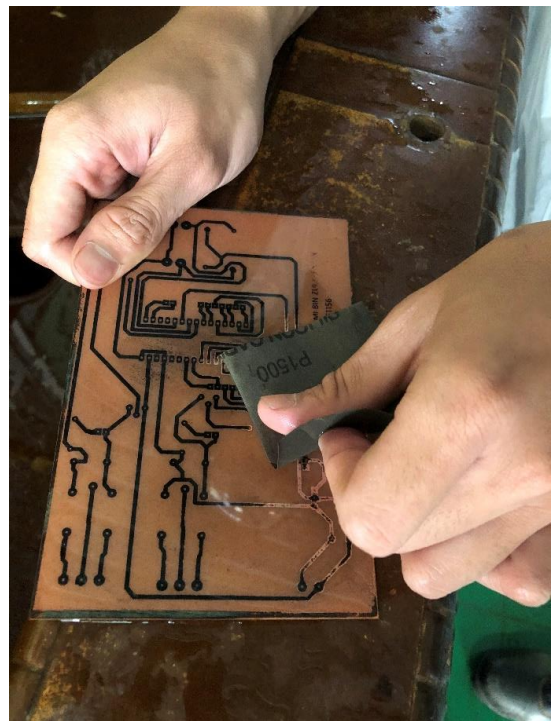
**Rajah 6.5:** Tempoh proses *Etching*

- c. Setelah proses *Etching* selesai, papan litar bercetak (PCB) akan dicuci dengan menggunakan aliran air bersih seperti yang ditunjukkan pada **Rajah 6.6**.



**Rajah 6.6:**  
Pembersihan Papan Litar Bercetak

- d. Kemudian, **Rajah 6.7** menunjukkan kertas pasir digosokkan pada papan litar bercetak (PCB) agar menanggalkan dakwat pencetak laser jet.



**Rajah 6.7:**  
Proses Menghilangkan Dakwat Pencetak pada Papan Litar Bercetak

### 6.3 Proses Penggerudian



- a. Proses penggerudian seperti pada **Rajah 6.8** dilaksanakan selepas proses *etching*. Mata gerudi akan menggerudi papan litar bercetak (PCB) sehingga menghasilkan lubang agar dapat meletakkan kaki komponen elektronik pada papan litar bercetak (PCB).

**Rajah 6.8:** Proses Penggerudian Papan Litar Bercetak

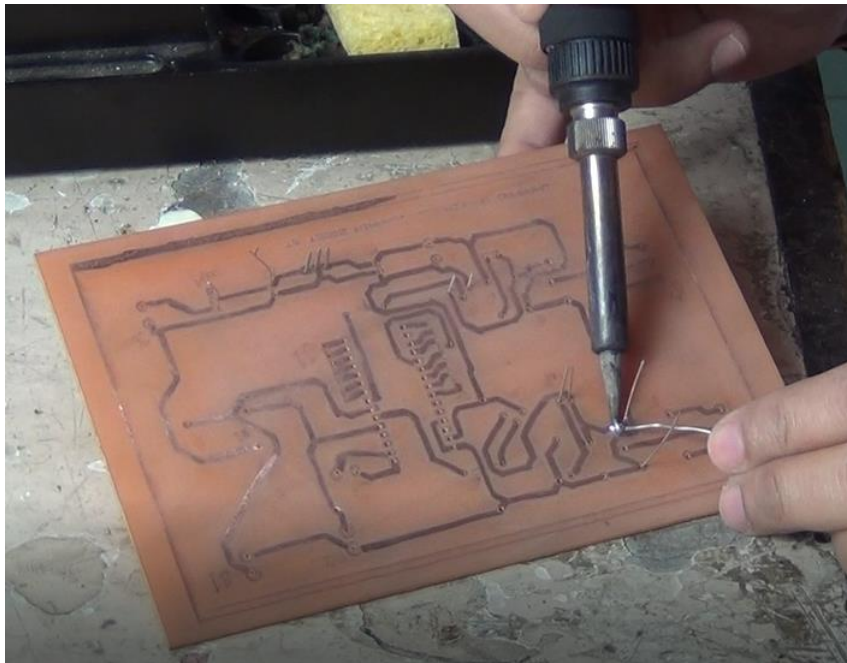
- b. Mata gerudi yang akan digunakan adalah daripada jenis **HSS Twist drill bit** yang terdiri daripada dua saiz iaitu 1.0mm untuk kaki komponen elektronik geganti, litar bersepadu (**IC**) dan pengatur manakala 0.8mm untuk kaki komponen elektronik perintang, diod dan transistor.



**Rajah 6.9:** Jenis Mata Gerudi

## 6.4 Proses Pematerian Komponen

- a. Setelah pengerudian selesai, komponen-komponen elektronik ini disisipkan pada papan litar bercetak untuk melaksanakan proses pematerian. Proses pematerian pada **Rajah 6.10** ialah proses menyambungkan kaki-kaki komponen elektronik pada papan litar bercetak dengan meleburkan logam timah.

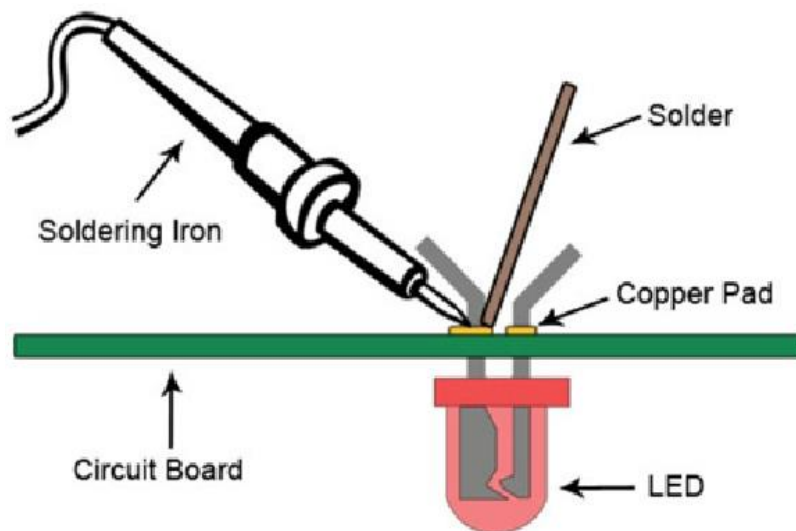


**Rajah 6.10:** Proses Pematerian Komponen

- b. Untuk melakukan proses pematerian kita mestilah memilih *soldering iron* seperti pada **Rajah 6.11** yang mempunyai jumlah kuasa yang sesuai. Lazimnya *soldering iron* dengan kuasa 20-watt hingga 30-watt adalah memadai untuk memateri komponen elektronik yang biasa.



**Rajah 6.11:** Jenis *Soldering Iron*



**Rajah 6.12:** Tunjuk Cara Pematerian Komponen

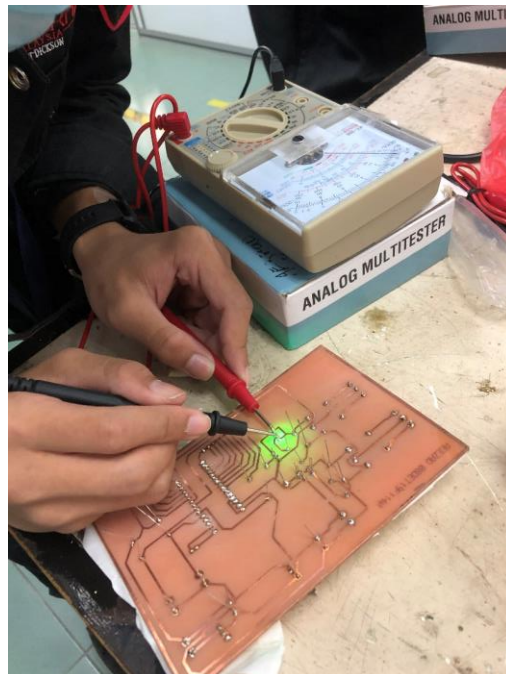
- c. **Rajah 6.12** ialah tunjuk cara untuk melakukan proses pematerian komponen, ia boleh dimulakan dengan memanaskan tempat yang hendak dipateri dengan menggunakan *soldering iron*.
- d. Kemudian, letakkan timah pateri ditempat yang hendak dipateri. Untuk menambah kekemasan hasil pematerian kita boleh menggunakan *soldering paste* atau *soldering flux* seperti pada **Rajah 6.13**.
- e. Biarkan seketika sehingga timah tersebut mencair mengikut kadaran yang sesuai.
- f. Setelah selesai proses tersebut angkat dengan segera timah dan soldering iron bagi mengelakkan track pada papan litar menjadi rosak atau pun tertanggal.



**Rajah 6.13:** Soldering Paste

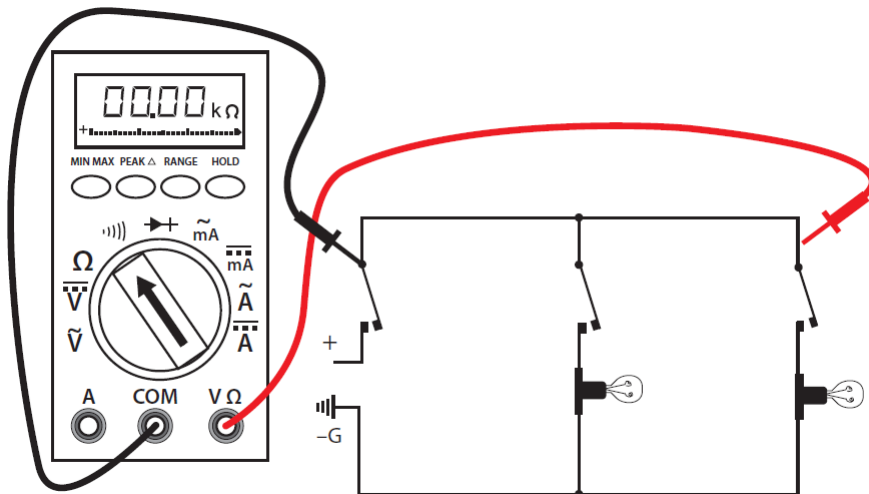
## 6.5 Pengujian Keterusan Litar

- a. Akhirnya, litar elektronik pada papan litar bercetak (PCB) boleh melakukan pengujian mengikut tugas masing-masing. Litar yang telah lengkap mesti diuji dengan ujian keterusan litar seperti pada **Rajah 6.14**.



**Rajah 6.14:** Proses Pengujian Keterusan Litar

- b. Multimeter akan dilaras pada meter ohm seperti yang ditunjukkan pada **Rajah 6.15** untuk melihat sama ada papan litar bercetak dalam keadaan litar buka atau litar pintas.



**Rajah 6.15:** Tunjuk Cara Ujian Keterusan Litar

- c. Ujian keterusan litar ini dijalankan secara **offline** iaitu tanpa bekalan kuasa. Tujuan utama ujian keterusan litar ini dijalankan adalah untuk melihat sambungan di antara **track** sama ada ia telah terputus, terpintas atau pun dalam keadaan baik. Ini kerana semasa proses **etching** kadangkala kadar hakisan oleh **ferric-chloride** adalah tidak seragam disebabkan oleh kepekatan asid yang berkurangan atau pun semburan asid yang tidak sekata.

d. Sekiranya berlaku kerosakan pada **track** maka ianya mestilah diperbaiki terlebih dahulu sebelum proses pematerian komponen dijalankan. Selepas proses pematerian komponen, litar tersebut perlu diuji sekali lagi bagi memastikan komponen yang telah dipateri mempunyai lekatan yang elok dengan papan litar. Lazimnya bacaan multimeter akan menunjukkan nilai infiniti sekiranya litar tersebut adalah litar buka. Untuk litar yang mempunya sambungan, multimeter akan menunjukkan nilai sifar.

## 6.6 Pemeriksaan Litar Secara Visual

Selain itu, pemeriksaan litar secara visual yang dilakukan selepas pematerian bertujuan untuk memastikan tiada berlakunya kesilapan pemasangan komponen bagi mengelakkan komponen tersebut rosak. Antara perkara yang diperiksa ialah kekutuban komponen berada dalam keadaan yang betul, sambungan terminal positif dan negatif berada dalam keadaan yang betul serta menggunakan kod warna yang betul. Setelah semua perkara tersebut dipatuhi barulah kita boleh memasang bekalan kuasa kepada litar tersebut untuk membuat ujian litar hidup.

Akhir sekali anda boleh mengimbas QR-code di bawah ini untuk melihat bagaimanakah proses membuat PCB dilakukan.



## Uji Kefahaman 6

1. Apakah yang dimaksudkan dengan proses *Etching* ?
2. Nyatakan jenis mata gerudi yang digunakan dalam pembuatan papan litar tercetak.
3. Mengapakah kita perlu melakukan ujian keterusan litar selepas menghasilkan papan litar tercetak?

Imbas **QR-Code** untuk menyemak jawapan anda



# RUJUKAN

1. CAI, Z. & TONG, S. Application of Proteus Simulation Software in the Teaching of Electric Courses. *DEStech Transactions on Social Science, Education and Human Science* (2017) doi:10.12783/dtssehs/ermm2017/14722.
2. Varteresian, J. CHAPTER 1 - Schematic Capture. in *Fabricating Printed Circuit Boards* (ed. Varteresian, J.) 1–10 (Newnes, 2002). doi:https://doi.org/10.1016/B978-187870750-5/50002-2.
3. Barrett, S. F. *Arduino I: Getting Started*. in *Synthesis Lectures on Digital Circuits and Systems* vol. 15 (2020).
4. Varteresian, J. CHAPTER 2 - Basic Circuit Board Placement and Routing Considerations. in *Fabricating Printed Circuit Boards* (ed. Varteresian, J.) 11–33 (Newnes, 2002). doi:https://doi.org/10.1016/B978-187870750-5/50003-4.
5. Ward, H. H. *Intermediate C Programming for the PIC Microcontroller*. *Intermediate C Programming for the PIC Microcontroller* (2020). doi:10.1007/978-1-4842-6068-5.
6. Ibrahim, D. Chapter 6 - Microcontroller Program Development. in *Designing Embedded Systems with 32-Bit PIC Microcontrollers and MikroC* (ed. Ibrahim, D.) 259–273 (Newnes, 2014). doi:https://doi.org/10.1016/B978-0-08-097786-7.00006-3.
7. Barrett, S. F. *Arduino microcontroller processing for everyone! third edition* Barrett. *Synthesis Lectures on Digital Circuits and Systems* 43, (2013).
8. Cameron, N. *Arduino Applied*. *Arduino Applied* (2019). doi:10.1007/978-1-4842-3960-5.
9. Varteresian, J. CHAPTER 5 - Making Printed Circuit Boards. in *Fabricating Printed Circuit Boards* (ed. Varteresian, J.) 69–87 (Newnes, 2002). doi:https://doi.org/10.1016/B978-187870750-5/50006-X.
10. Varteresian, J. CHAPTER 4 - Real World Guidelines for Commercial Fabrication Houses. in *Fabricating Printed Circuit Boards* (ed. Varteresian, J.) 57–67 (Newnes, 2002). doi:https://doi.org/10.1016/B978-187870750-5/50005-8.
11. Lienig, J. & Scheible, J. *Fundamentals of Layout Design for Electronic Circuits*. *Fundamentals of Layout Design for Electronic Circuits* (2020). doi:10.1007/978-3-030-39284-0.
12. Unknown. *Standard Operating Procedure for Chemical Handling and Storage*. *Colleges of Nanoscale Science and Engineering Suny Polytechnic Institute* (2015).
13. Subamia, I. D. P., Wahyuni, I. G. A. N. S. & Widiasih, N. N. *Pelatihan Penanganan Risiko Bahan Berbahaya di Laboratorium Kimia bagi Laboran*. *Prosiding SENADIMAS Ke-4 1*, (2019).
14. Kaya, M. *Printed Circuit Boards (PCBs)*. in *Minerals, Metals and Materials Series* (2019). doi:10.1007/978-3-030-26593-9\_2.
15. Lau, J. H. & Lee, N.-C. *Soldering Processes*. in *Assembly and Reliability of Lead-Free Solder Joints* (2020). doi:10.1007/978-981-15-3920-6\_3.
16. Khandpur, R. S. *Printed Circuit Boards Design, Fabrication and Assembly*. McGraw-Hill (2006).



e ISBN 978-629-7643-00-7



POLITEKNIK PORT DICKSON