

BAB III

PERANCANGAN DAN PEMBUATAN ALAT

3.1 Perancangan Alat

Secara umum, perancangan lampu led otomatis ini terdiri dari dua bagian dasar, yaitu bagian perangkat lunak (*software*), dan bagian keras (*hardware*).



Gambar 3.1 Diagram Blok Perancangan Lampu Led Otomatis

Pada penelitian ini, tahapan awal yang dilakukan yaitu melakukan perancangan perangkat keras (*hardware*). Rancangan ini berupa desain rancangan konfigurasi LED dan sistem kendali. Konfigurasi LED yang dimaksud disini adalah penentuan jumlah LED yang digunakan, sehingga intensitas cahaya yang dihasilkan sesuai dengan yang direncanakan. Sedangkan sistem kendali adalah media pengolahan data masukan (*input data*) dari perangkat masukan (*input device*) untuk menghasilkan data keluaran (*output data*) pada perangkat keluaran (*output device*) sesuai dengan yang direncanakan.

Tahapan berikutnya yaitu perancangan perangkat lunak (*software*), sebagai tindak lanjut dari hasil rancangan *hardware*. Rancangan ini berupa suatu program sistem kendali yang berfungsi untuk mengolah data masukan (*input data*) dari perangkat masukan (*input device*) untuk menghasilkan data keluaran (*output data*) pada perangkat keluaran (*output device*) sesuai dengan yang direncanakan.

—

rancangan *hardware* dipergunakan sebagai medium untuk mensimulasikan rancangan sistem kendali yang akan digunakan.

3.1.1 Perancangan *Hardware*

Tahapan awal yang dilakukan dalam perancangan *hardware* ini adalah melakukan perhitungan untuk menentukan jumlah LED yang akan digunakan. Metode yang digunakan adalah dengan metode komulatif, yaitu dengan cara mengukur intensitas cahaya dari satu buah LED untuk dijadikan sebagai patokan. Dari hasil tersebut maka dapat dilakukan perhitungan jumlah LED yang akan digunakan dengan menjumlahkan intensitas dari satu buah LED berdasarkan batasan yang telah ditentukan sebelumnya, sehingga akan menghasilkan cahaya sesuai dengan yang telah direncanakan.

Perhitungan untuk menentukan jumlah LED adalah sebagai berikut :

Diketahui : Cahaya 1 LED = 30 Lux

Total cahaya yang diinginkan = 300 Lux

Maka : Jumlah LED = $\frac{\text{Total cahaya yang diinginkan}}{\text{cahaya 1 LED}} = \frac{300}{30} = 10$

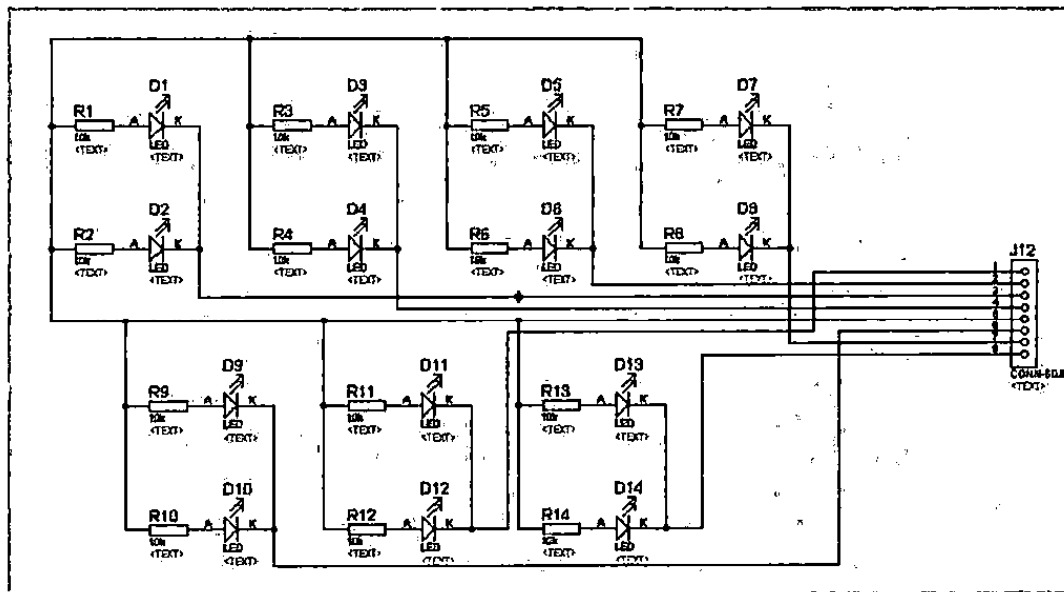
Hasil perhitungan diatas menunjukkan bahwa LED yang dibutuhkan adalah sebanyak 10 buah. Namun, penulis disini menambahkan 4 buah LED lagi sehingga total LED yang digunakan menjadi 14 buah. Hal ini dimaksudkan agar total cahaya yang dihasilkan nanti lebih besar dari batas minimal cahaya yang telah ditentukan. Selain itu, penambahan LED ini juga bermaksud untuk menggenapkan penggunaan PORT output pada

... untuk lebih mudah dalam pemrogramannya

Tahapan selanjutnya adalah perancangan rangkaian *hardware*. Dalam perancangan rangkaian *hardware* ini penulis menggunakan *software* Proteus, ISIS dan ARES. Sehingga yang dihasilkan berupa *schematic diagram* rangkaian sistem kendali dan desain *layout* PCB (*Printed Circuit Board*).

Langkah-langkah perancangan yang dilakukan adalah sebagai berikut :

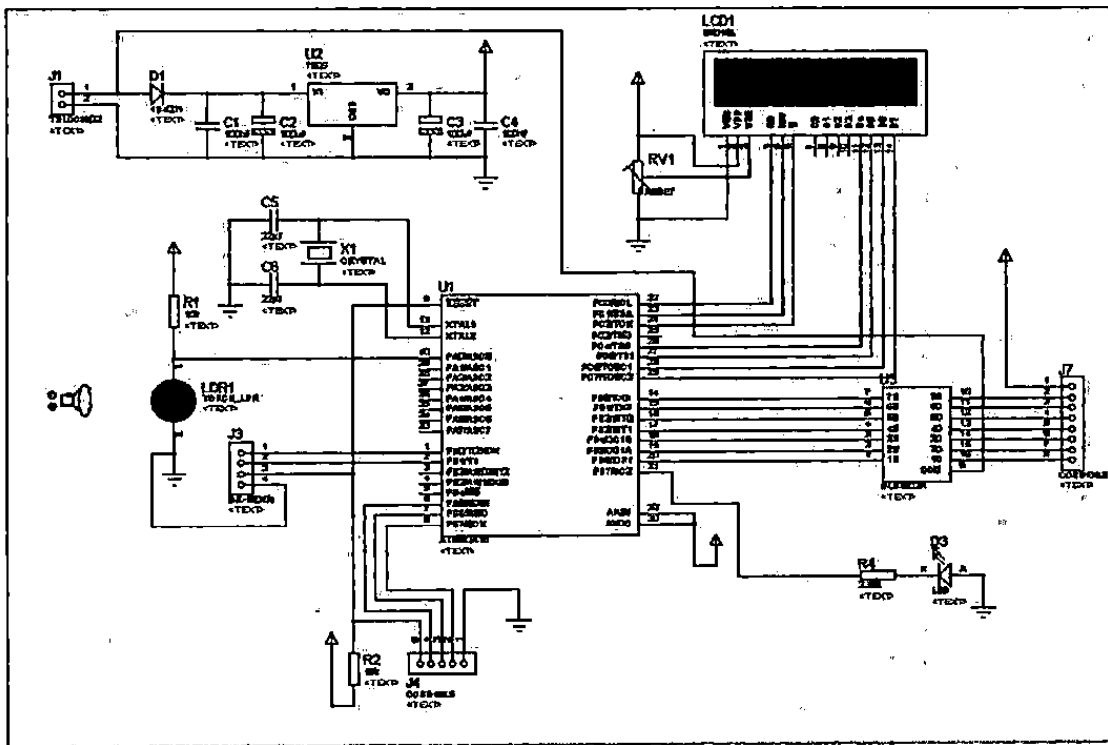
- Merancang *schematic diagram* dengan menggunakan *software* PROTEUS ISIS.
- Merancang *layout* PCB dengan menggunakan *software* PROTEUS ARES.



Gambar 3.2 Desain *Schematic Diagram* rangkaian LED pada Proteus ISIS

Schematic diagram rangkaian ini merupakan desain rangkaian elektronik LED yang akan dijadikan sebagai output pada penelitian ini. Pada rangkaian tersebut masing-masing LED diberi satu resistor dan disambung paralel. Resistor tersebut berfungsi untuk menjaga LED supaya arus yang

mengalir tidak melebihi batas kemampuan LED tersebut. Sedangkan penyambungan LED secara paralel dimaksudkan agar masing-masing LED memiliki nyala yang lebih merata dan tidak terganggu antara satu LED dengan LED yang lainnya. J12 merupakan *junction* atau sambungan yang berfungsi untuk menghubungkan rangkaian LED dengan rangkaian kontrolernya.



Gambar 3.3 Desain *Schematic Diagram* rangkaian kontroler di Proteus ISIS

Schematic diagram rangkaian kontroler ini merupakan desain sistem rangkaian elektronik yang dibuat berdasarkan sistem minimum mikrokontroler ATMEGA16, yang menjadi pusat kendali pengolahan data masukan (*input data*) dari perangkat masukan (*input device*) yang berupa

Mikrokontroler ATMEGA16 yang merupakan jantung dari rangkaian kontroler di atas akan menerima data masukan dari sensor LDR melalui PORTA.0. Data masukan dari sensor LDR berupa sinyal analog, sedangkan yang dapat diolah oleh mikrokontroler ATMEGA16 berupa sinyal digital. Sehingga data masukan itu terlebih dahulu dikonversi menggunakan fasilitas ADC (*Analog to Digital Converter*) pada mikrokontroler ATMEGA16.

Hasil konversi data analog yang telah diubah menjadi data digital tersebut, selanjutnya diolah menggunakan algoritma tertentu yang kemudian akan menghasilkan tegangan keluaran pada PORTD.0-PORTD.6. PORT-PORT tersebut terhubung dengan rangkaian LED dimana masing-masing PORT berfungsi untuk menghidupkan/mematikan 2 LED. Diantara PORTD.0-PORTD.6 dan rangkaian LED terdapat IC ULN2003A yang berfungsi untuk mencegah arus lebih mengalir yang dapat merusak mikrokontroler ATMEGA16. Sebab arus yang diperbolehkan pada mikrokontroler ATMEGA16 sangatlah kecil, yaitu hanya sebesar 20 mA.

J3 adalah *input* tegangan 5 Volt yang berasal dari *step down regulator*. Port C digunakan sebagai jalur data untuk penampil LCD. Xtal yang digunakan adalah xtal 8 MHz, agar dapat menghasilkan unit detik yang tepat. PORTB.0, PORTB.1, dan RESET, masing-masing dihubungkan dengan push button yang digunakan untuk memberikan input pada mikrokontroler. PORTD.7 output mikrokontroler dihubungkan dengan LED sebagai LED

Tabel 3.1 Pin-pin I/O yang digunakan pada ATMEGA16

Nama Port	Pin No	Pin I/O	Tipe Pin	Fungsi
PORT A	40	0	I/O	LDR
	39	1	I/O	-
	38	2	I/O	-
	37	3	I/O	-
	36	4	I/O	-
	35	5	I/O	-
	34	6	I/O	-
	33	7	I/O	-
AREF	32	-	I/O	-
GND	31	-	I/O	-
AVCC	30	-	I/O	-
PORT C	29	7	I/O	To LCD (D4)
	28	6	I/O	To LCD (D3)
	27	5	I/O	To LCD (D2)
	26	4	I/O	To LCD (D1)
	25	3	I/O	-
	24	2	I/O	To LCD (E)
	23	1	I/O	To LCD (RW)
	22	0	I/O	To LCD (RS)
PORT D	21	0	I/O	To LED

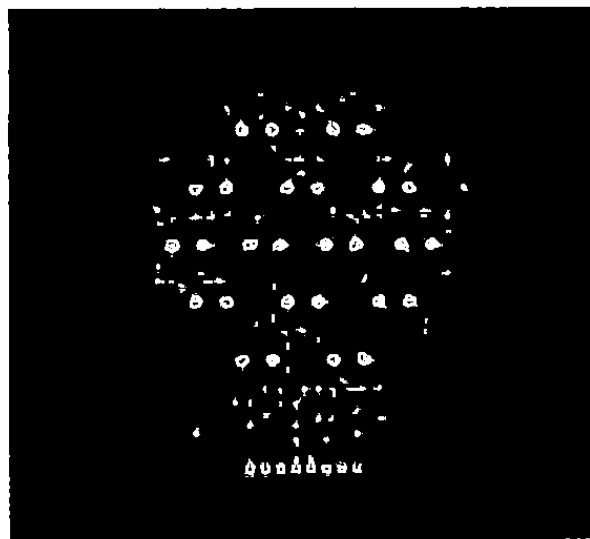
	20	1	I/O	To LED
	19	2	I/O	To LED
	18	3	I/O	To LED
	17	4	I/O	To LED
	16	5	I/O	To LED
	15	6	I/O	To LED
	14	7	I/O	LED indikator
XTAL1	13	-	I/O	-
XTAL2	12	-	I/O	-
GND	11	-	I/O	-
VCC	10	-	I/O	-
RESET	9	-	I/O	Tombol
PORT B	8	7	I/O	-
	7	6	I/O	-
	6	5	I/O	-
	5	4	I/O	-
	4	3	I/O	-
	3	2	I/O	-
	2	1	I/O	Tombol
	1	0	I/O	Tombol

Dalam pembuatan rangkaian ini diperlukan pemahaman yang utuh

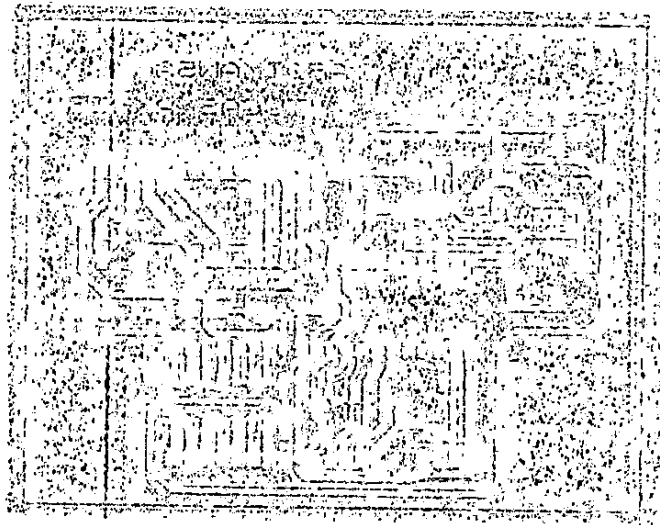
...sistem minimum dari mikrokontroler ATMEGA16 yang akan

dirancang. Sistem rangkaian yang akan dirancang diusahakan menggunakan rangkaian yang ringkas mungkin dan dengan pengkabelan yang baik. Sebab, rangkaian tersebut bekerja pada frekuensi yang relatif tinggi, sehingga peka terhadap *noise* dari luar.

Setelah menyelesaikan perancangan *schematic diagram*, tahap selanjutnya adalah membuat *layout* PCB yang dirancang menggunakan *software* PROTEUS ARES. ARES secara otomatis akan menghubungkan titik-titik komponen dari *schematic diagram* yang telah dibuat sebelumnya di PROTEUS ISIS. Berikut adalah *layout* PCB yang dihasilkan menggunakan teknik *autorouting*. Meskipun, teknik ini juga terkadang masih terdapat *error*, sehingga sangat perlu untuk mengecek kembali hasil *auto routing* tersebut dan membetulkan secara *manual* jalur-jalur yang belum benar atau kurang efisien.



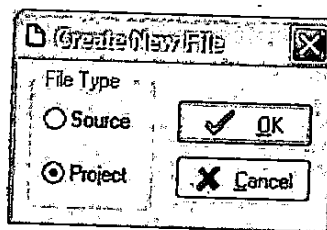
Gambar 2.4 Layout PCB rangkaian LED pada Proteus ARES



Kemudahan yang didapatkan dalam pengoperasian CVAVR jika menggunakan *code wizard*, antara lain :

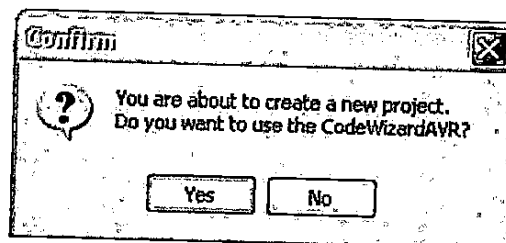
- **Membuat dan Pemilihan Chip dan Frekuensi Xtall**

Langkah pertama dalam menggunakan CVAVR adalah membuat sebuah *project* baru, dengan meng-klik *create new project*.



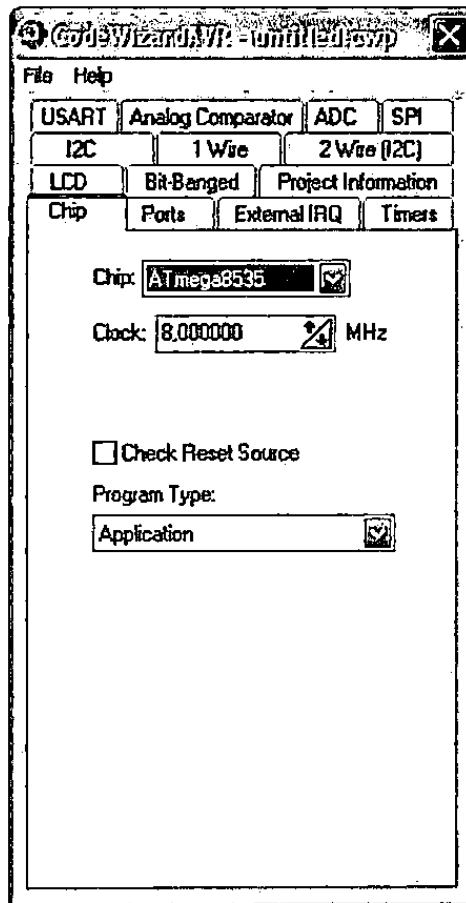
Gambar 3.6 Membuat *project* baru

Maka akan muncul pertanyaan apakah anda ingin memanfaatkan bantuan *codewizard*.



Gambar 3.7 Pemilihan bantuan *code wizard*

Jika memilih "Yes", maka akan masuk pada *codewizard*. Langkah pertama yang harus dilakukan pada *codewizard* adalah memilih jenis chip mikrokontroler dan frekuensi Xtall yang akan digunakan dalam *project*. Pemilihan chip dan frekuensi Xtall dapat dilihat pada gambar dibawah ini :



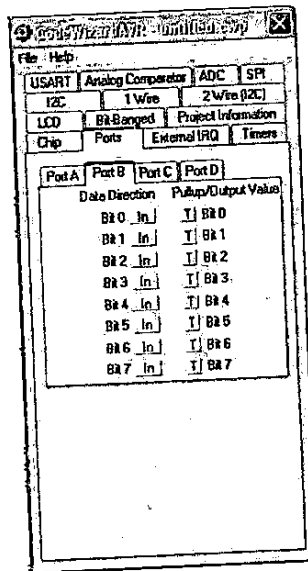
Gambar 3.8 Pemilihan Chip dan Frekuensi Xtall yang akan dipakai

- **Inisialisasi Port I/O**

Inisialisasi Port berfungsi untuk memilih fungsi port sebagai *input* atau sebagai *output*. Pada konfigurasi port sebagai *output* dapat dipilih pada saat awal setelah reset kondisi port berlogika 1 atau 0, sedangkan pada konfigurasi port sebagai *input* terdapat dua pilihan yaitu kondisi pin *input pull-down* atau *pull-up*, maka sebaiknya dipilih *pull-up* untuk memberi *default* pada input selalu berlogika 1. Setiap port berjumlah 8 bit, konfigurasi dari port

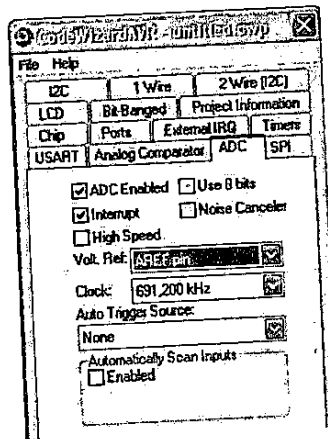
dilakukan per-bit, jadi dalam satu port dapat difungsikan sebagai input dan output dengan nilai *default*-nya berbeda-beda.

Gambar di bawah ini menunjukkan *setting* konfigurasi pada Port B dengan kombinasi *input* dan *output* yang berbeda-beda *default*-nya.



Gambar 3.9 Inisialisasi Port I/O

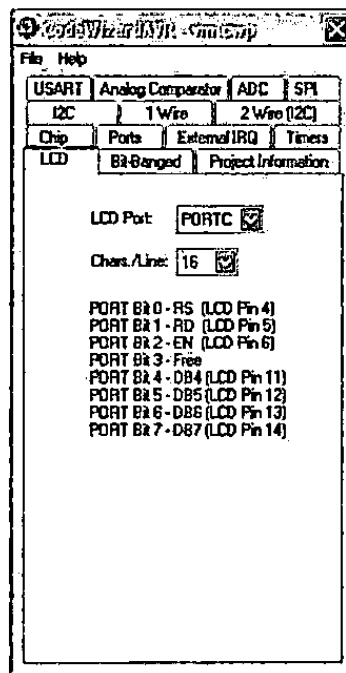
- Inisialisasi Port ADC



Gambar 3.10 Inisialisasi Port ADC

Analog to Digital Converter (ADC) berfungsi untuk mengubah tegangan analog ke data digital. Digunakan pada sensor, yang keluarannya masih analog. Fasilitas ADC internal pada ATmega16 ada 8 port, yaitu di PA.0 sampai PA.7.

- **Inisialisasi Port LCD**



Gambar 3.11 Inisialisasi Port LCD

3.2 Pembuatan

Pembuatan ini merupakan realisasi dari seluruh rancangan rangkaian tersebut di atas. Dimulai dari pengadaan bahan, persiapan alat, pengerjaan, dan pengujian. Penelitian mengenai tugas akhir ini dikerjakan selama 7 bulan, yaitu dimulai sejak bulan April 2013 hingga bulan November 2013, yang dilakukan di Universitas Muhammadiyah Yogyakarta dengan menggunakan

3.2.1 Pengadaan Bahan

Bahan-bahan yang digunakan pada tahap pembuatan adalah sebagai berikut :

- 1 lembar PCB (13cm x 10cm)
- Bahan kimia $FeCl_3$
- Fotokopi *glossylayout* PCB
- Terminal blok
- *Tools box*
- Komponen elektronika yang berupa :
 - Mikrokontroler ATMEGA16
 - Sensor LDR (*Light Dependent Resistor*)
 - LED
 - LCD 2x16 character
 - LM 7805
 - *Xtal*8 MHz
 - ULN2003A
 - LED indikator 5mm
 - Kabel
 - Beberapa komponen pendukung lainnya, seperti resistor, kapasitor, transistor, diode, dan lain-lain

3.2.2 Persiapan Alat

Alat-alat yang digunakan dalam pembuatan ini antara lain :

- Solder dan tenol

- USB ASP *downloader*
- Setrika listrik
- Bor listrik dan mata bornya
- *Cutter*
- *Tools set* (obeng, tang, tang potong)
- *Desoldering attractor*
- *Laboratory Power Supply*
- Multimeter
- *Function Generator* GW Insteck SFG-2010
- *Digital Storage Oscilloscope* GW Insteck GDS-2102
- Lux meter

3.2.3 Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan adalah :

- Aplikasi program Proteus, ISIS SCH dan ARES PCB *Layout*
- Aplikasi program Code Vision AVR
- Aplikasi program downloader eXtreme Burner AVR

3.3 Pengerjaan

Pengerjaan dimulai dengan pembuatan PCB seluruh rangkaian yang telah dirancang sebelumnya. Teknik pembuatan PCB yang dilakukan adalah teknik *transfer paper*, yaitu suatu teknik pembuatan PCB yang murah, dengan tidak mengesampingkan kualitas.

Langkah awal yang harus dilakukan adalah *layout* PCB dicetak

Hasil fotokopi tersebut dipanaskan dan ditekan pada permukaan PCB menggunakan setrika listrik. Setelah yakin semua hasil setrika berpindah tempat dari transparansi ke PCB, diamkan sejenak hingga panas pada permukaan PCB berangsur-angsur mendingin. Ketika permukaan PCB telah dingin, maka kertas fotokopi tersebut dapat dilepas secara perlahan.

Proses selanjutnya, PCB dilarutkan dalam larutan FeCl_3 , agar jalurnya dapat tercetak. Untuk mempercepat proses pelarutan, goyangkan wadah tempat pelarutan secara kontinyu. Setelah jalur tercetak, tahapan selanjutnya yaitu pengeboran lubang-lubang komponen dan pembersihan jalur tembaga pada PCB. Langkah terakhir adalah pemasangan komponen sesuai letak yang telah ditentukan