

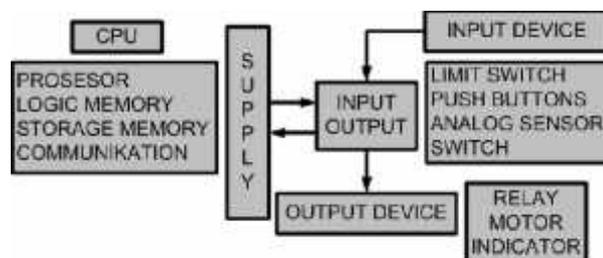
## BAB II TINJAUAN PUSTAKA

### 2.1 Pengenalan PLC

PLC merupakan singkatan<sup>1</sup> dari *Programmabel Logic Controller*, *Programmabel* artinya menunjukkan kemampuannya yang dapat dengan mudah diubah-ubah sesuai program yang dibuat dan kemampuannya dalam hal memori program yang telah dibuat. *Logic* artinya menunjukkan kemampuannya dalam hal memproses data input secara aritmatik (ALU), yaitu melakukan operasi membandingkan, menjumlahkan, mengalikan, membagi, mengurangi, dan negasi. Sedangkan *Controller* artinya menunjukkan kemampuannya dalam mengontrol dan mengatur proses sehingga menghasilkan output yang diinginkan.

PLC merupakan sebuah alat yang digunakan untuk menggantikan rangkaian sederetan *relay* yang dijumpai pada sistem kontrol proses konvensional. PLC bekerja dengan cara mengamati masukan (melalui sensor-sensor), kemudian menentukan aksi apa yang harus dilakukan pada instrumen keluaran berkaitan dengan status suatu ukuran atau besaran yang diamati. PLC banyak digunakan pada aplikasi-aplikasi industri. Semakin kompleks proses yang harus ditangani, semakin penting penggunaan PLC untuk mempermudah proses-proses tersebut.

Peralatan yang dihubungkan pada PLC yang berfungsi mengirim sebuah sinyal ke PLC disebut *input device*. Sinyal input masuk pada PLC disebut *input poin*. Input poin ini ditempatkan dalam lokasi memori sesuai dengan statusnya *on* atau *off*. Pada dasarnya PLC terdiri dari tiga bagian utama yaitu bagian input/output, bagian prosesor dan perangkat pemrograman.



**Gambar 2.1** Blok Diagram Programmable Controller



Tahap dasar untuk penyiapan awal untuk memudahkan dan memasukkan program dalam PLC dengan mempersiapkan daftar seluruh peralatan input dan output beserta lokasi I/O bit, penempatan lokasi word dalam penulisan data. Untuk pemrograman sebuah *Programmable Logic Controller* terlebih dahulu kita harus mengenal atau mengetahui tentang organisasi dan memorinya.

Ilustrasi dari organisasi memori adalah sebagai peta memori (memori map), yang *spacenya* terdiri dari kategori *User Programable* dan *Data Table*. *User Program* adalah dimana program *Logic Ladder* dimasukkan dan disimpan yang berupa instruksi – instruksi dalam format *Logic Ladder*. Setiap instruksi memerlukan satu *word* didalam memori.

### **2.1.1 Sejarah dan Perkembangan PLC**

Secara hitoris, PLC pertama kali dirancang perusahaan *General Motor* (GM) sekitar tahun 1968 untuk menggantikan *control relay* pada proses sekuensial yang dirasakan tidak *fleksibel* dan berbiaya tinggi. Pada saat itu, hasil rancangan telah benarbenar berbasis komponen *solid state* dan memiliki *fleksibilitas* tinggi, hanya secara *fungsiional* masih terbatas pada fungsi-fungsi kontrol saja. Seiring perkembangan teknologi *solid state*, saat ini PLC telah mengalami perkembangan luar biasa, baik pada ukuran, kepadatan komponen serta dari segi *fungsiionalnya*. Beberapa peningkatan perangkat keras dan perangkat lunak ini di antaranya adalah:

- Ukuran semakin kecil dan kompak.
- Jumlah input/output yang semakin banyak dan padat.

Beberapa jenis dan tipe PLC dilengkapi dengan modul-modul untuk tujuan kontrol kontinu, misalnya modul ADC/DAC, PID, modul *Fuzzy*, danlain-lain.

- Pemrograman relatif semakin mudah. Hal ini terkait dengan perangkat lunak pemrograman yang semakin *user friendly*.
- Memiliki kemampuan komunikasi dan sistem dokumentasi yang semakin baik.
- Jenis instruksi/fungsi semakin banyak dan lengkap.



- Waktu eksekusi program yang semakin cepat.

Saat ini, vendor-vendor PLC umumnya memproduksi PLC dengan berbagai ukuran, jumlah input/output, instruksi dan kemampuan lainnya yang beragam. Hal ini dilakukan untuk memenuhi kebutuhan pasar yang sangat luas, yaitu untuk tujuan kontrol yang relative sederhana dengan jumlah input/output (I/O) puluhan, sampai kontrol yang kompleks dengan jumlah I/O mencapai ribuan. Berdasarkan jumlah I/O yang dimilikinya.

Dalam hal ini PLC yang digunakan adalah salah satu jenis PLC mini LOGO *type* 0BA, dimana mini PLC ini sebagai pengontrol sistem pembatas parkir kendaraan .

## **2.2 PLC PROGRAM**

Suatu software yang berfungsi sebagai pengontrol otomatis yang berupa softcontact yang diimplementasikan kedalam suatu bentuk bilangan logika. Sehingga dapat mengatur sistem suatu alat industri elektronika dan mekanik.

Ada 2 sistem pemrograman pada PLC LOGO! :

- a. *Function Block Diagram* : Jenis Teknik Pemrograman Logic yang tersusun dari block-block diagram dalam1 fungsi blok diagram khusus.
- b. *Ladder Diagram* : Jenis Teknik Pemrograman Logic yang disusun dalam satuan-satuan kontak untuk menghasilkan fungsi tertentu dalam menghasilkan logika yang terdiri dari kontak NC, NO, Coil, Timer, Counter dll.

## **2.3 Mini PLC LOGO! 0BA6**

Merupakan Jenis dari mini PLC LOGO! Yang dapat di implementasikan pada penggerak mekanisme alat industri, alat rumah tangga, dan tugas teknik lainnya, yang mana bersifat logika elektronika. LOGO! Soft dibentuk dalam penggabungan sistem komputerisasi, elektronika, dan mekanisme mesin serta menyediakan dokumentasi profesional dengan semua informasi proyek yang



diperlukan seperti program *switching*, komentar, dan pengaturan parameter. Berikut akan dijelaskan *software LOGO! Soft Comfort?* yang digunakan untuk memprogram PLC yang dinunakan

#### **2.4 LOGO! Soft Comfort?**

*Logo! Soft Comfort* adalah salah satu jenis Module PLC dimana dapat di implementasikan pada penggerak mekanisme alat industri, alat rumah tangga, dan tugas teknik lainnya, yang mana bersifat logika elektronika. LOGO! Soft dibentuk dalam penggabungan sistem komputerisasi, elektronika, dan mekanisme mesin dimana mulai dipasarkan pada tahun 1996 oleh Siemens Manufaktur.

#### **2.5 Aplikasi LOGO! Soft Comfort**

Adapun Aplikasi yang digunakan pada LOGO! Soft adalah *Logo! Soft Comfort Verse.7.0*

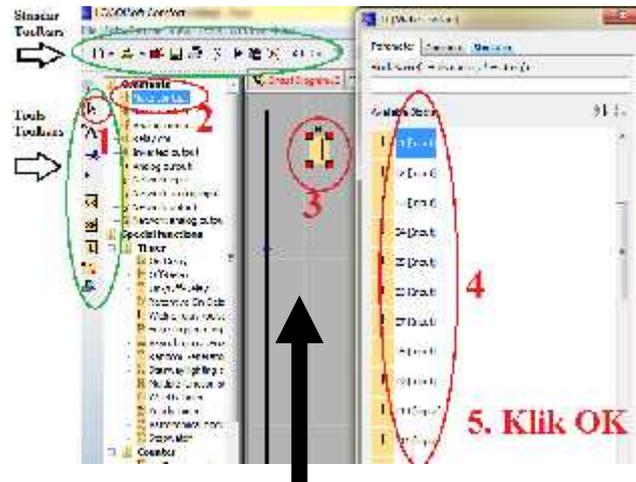
#### **2.6 Cara Pembuatan Program Logic Control (PLC)**

Untuk membuat program logic kita harus menentukan dahulu jenis blok apa yang akan kita buat apakah FBD, LAD, atau UDF. Setelah itu gunakan Icon Tools yang telah tersedia pada masing-masing tampilan.

Untuk menggunakan icon tools dapat dilakukan dengan cara mengklik icon tools yang diinginkan lalu sambil menekan tarik ke dalam areal *Drawing Board* (Contoh dibawah menggunakan tampilan Ladder Diagram) atau lebih jelasnya ikuti tahap berikut:

- 1. Klik Ponter pada Tools Toolbars**
- 2. Pilih Tools Pada Function Tools Constant, Basic, Specials dan Others Function**
- 3. Drag atau Tarik Icon Tools pada Drawing Board**
- 4. Pilih Spesifikasi Icon Tools.**
- 5. Lalu Klik OK**

Pada gambar dibawah ini dapat kita lihat penggunaan pada Constant Tools > Make Contact Normally Open (NO) pada Tampilan Ladder.



### DRAWING BOARD

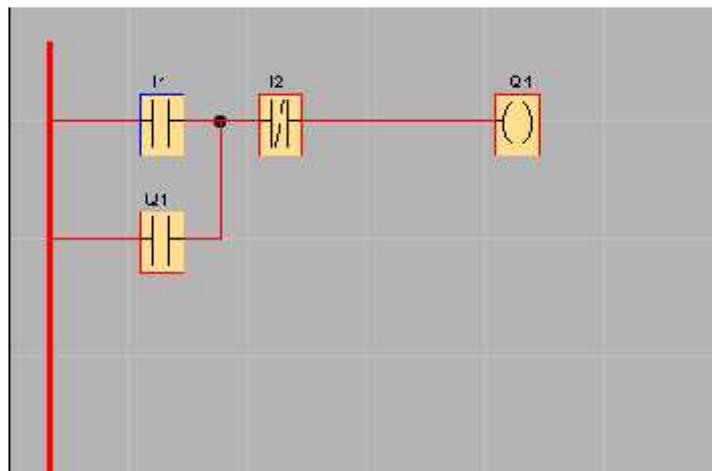
#### Gambar 2.2 Penggunaan Block Function

(Sumber : PT. Tambang Batu Bara Bukit Asam)

## 2.7 Contoh Program Logic Control (PLC) *Ladder Diagram*

### 2.7.1 *Latching* (Rangkaian Pengunci)

Rangkaian yang bersifat mengingat kondisi sebelumnya sering kali dibutuhkan dalam kontrol logic. Pada rangkaian ini hasil keluaran dikunci atau di latch dengan menggunakan kontak hasil keluaran itu sendiri, sehingga walaupun input sudah berubah, kondisi output tetap



#### Gambar 2.3 Latching Diagram

(Sumber : PT. Tambang Batu Bara Bukit Asam)



Ketika I1 di disconnect maka Q1 akan tetap berlogika 1, karena mendapat inputan langsung dari sumber. Sampai I2 sebagai pemutus di beri logika 1 yang mana kontak sebagai NC akan ber logika 0 ketika mendapat sumber tegangan sehingga memutus Q1 sebagai output.

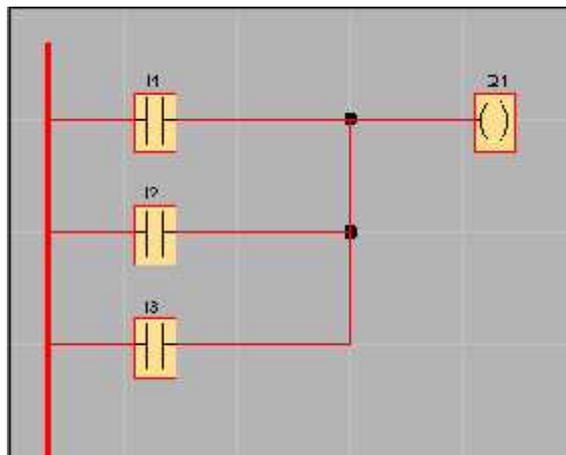
### 2.7.2 Operasi Logika OR

**Tabel 2.1 Data Kebenaran Logika OR**

(Sumber : PT. Tambang Batu Bara Bukit Asam)

I1	I2	I3	Q1
1	0	0	1
1	1	0	1
1	1	1	1
0	0	0	0

Pada logika OR, apabila salah satu inputan berlogika 1, maka Output akan bernilai logika 1. apabila semua input bernilai 0 maka akan bernilai 0. Berikut gambar Ladder diagramnya:



**Gambar 2.4 Diagram Rangkaian OR**

(Sumber : PT. Tambang Batu Bara Bukit Asam)



### 2.7.3 Operasi Logika AND

**Tabel 2.2 Data Kebenaran Logika AND**

(Sumber : PT. Tambang Batu Bara Bukit Asam)

I1	I2	I3	Q1
1	0	0	0
0	1	0	0
0	0	1	0
1	1	1	1

Operasi logika AND adalah apabila semua inputan berlogika 1 maka Output akan berlogika 1, begitu pula sebaliknya, apabila salah satu inputan berlogika nol maka output akan berlogika 0. karena operasi logika AND bersifat pengali. Berikut Ladder diagramnya:



**Gambar 2.5 Diagram Rangkaian AND**

(Sumber : PT. Tambang Batu Bara Bukit Asam)

### 2.7.4 Operasi Logika NOT

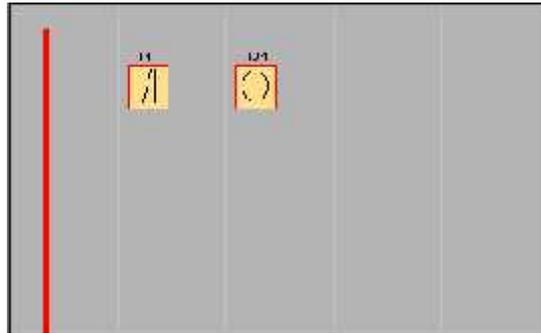
**Tabel 2.3 Data Kebenaran Logika NOT**

(Sumber : PT. Tambang Batu Bara Bukit Asam)

I1	Q1
1	0
0	1



Operasi logika NOT adalah apabila I1 bernilai 1 maka keluaran akan bernilai 0, begitu pula sebaliknya apabila I1 bernilai inputan 0 maka output akan bernilai logika 1. karena kontak yang digunakan hanya 1 yaitu kontak NC. Berikut gambar rangkaiannya:



**Gambar 2.6 Diagram Rangkaian NOT**

(Sumber : PT. Tambang Batu Bara Bukit Asam)

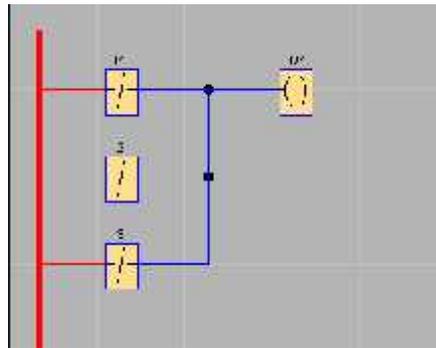
### 2.7.5 Operasi Logika NOT OR (NOR)

**Tabel 2.4 Data Kebenaran Operasi Logika NOT OR (NOR)**

(Sumber : PT. Tambang Batu Bara Bukit Asam)

I1	I2	I3	Q1
1	0	0	0
1	1	0	0
1	1	1	0
0	0	0	1

Untuk operasi logika NOR, semua kontak menggunakan Kontak Normally Close (NC) berlogika 1, apabila mendapat inputan tegangan maka akan berlogika 0. jadi operasi logika berbanding terbalik dengan operasi logika or, dimana apabila salah satu input berlogika 1 maka output akan bernilai logika 0 dan apabila semua inputan berlogika 0 maka keluaran akan berlogika 1. berikut gambar diagram ladder operasi logika NOT OR (NOR)



**Gambar 2.7 Diagram Rangkaian NOR**

(Sumber : PT. Tambang Batu Bara Bukit Asam)

### 2.7.6 Operasi Logika NOT AND (NAND)

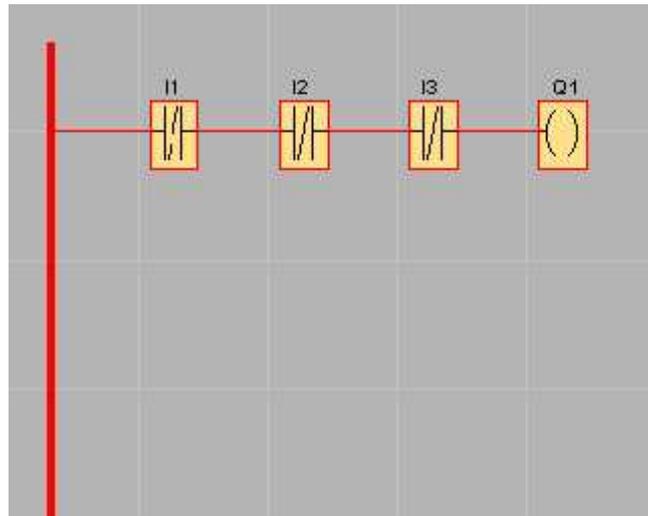
Untuk operasi logika NAND, semua kontak menggunakan Kontak Normally Close (NC) berlogika 1, apabila mendapat input tegangan maka akan berlogika 0. jadi operasi logika NAND berbanding terbalik dengan operasi logika AND, dimana apabila salah satu input berlogika 0 (Kontak NC mendapat Input sumber tegangan) maka output akan bernilai logika 0 dan apabila semua inputan berlogika 1 (Kontak)

**Tabel 2.5 Data Kebenaran Logika NAND**

(Sumber : PT. Tambang Batu Bara Bukit Asam)

I1	I2	I3	Q1
1	0	0	0
1	1	0	0
1	1	1	0
0	0	0	1

NC tidak Mendapat input sumber tegangan dan berlogika 1), maka keluaran akan berlogika 1. berikut gambar diagram ladder operasi logika NOT AND (NAND).Berikut gambar Ladder Diagramnya:



**Gambar 2.8 Diagram Rangkaian NAND**

(Sumber : PT. Tambang Batu Bara Bukit Asam)

### 2.7.7 Operasi Logika XOR

Logika ini merupakan pengembangan dari logika AND, OR, dan NOT

dimana apabila I1 dan I2 dalam kondisi yang sama seperti I1 = 0 dan I2 = 0 maka output akan berlogika 0 sedangkan I1 = 1 dan I2 = 1 maka output akan berlogika 0, dan apabila salah satu *switch* berlogika 1 maka output akan berlogika 1.

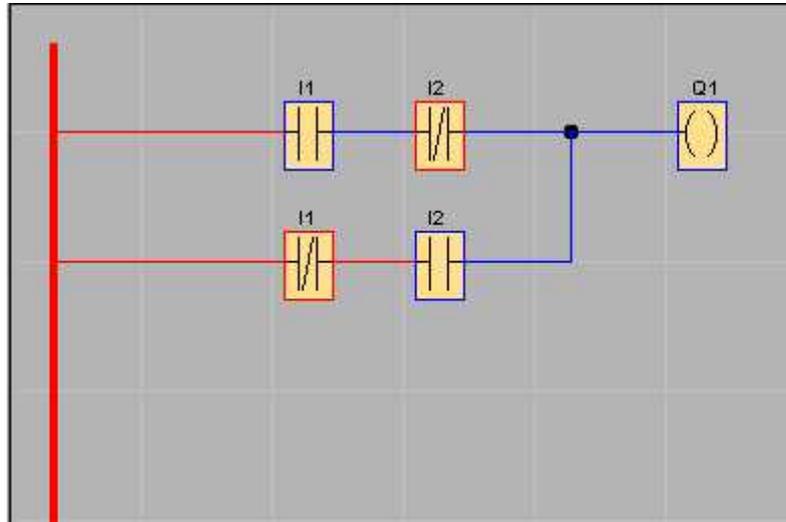
**Tabel 2.6 : Data kebenaran Operasi Logika XOR**

(Sumber : PT. Tambang Batu Bara Bukit Asam)

I1	I2	Q1
0	0	0
0	1	1
1	0	1
1	1	0



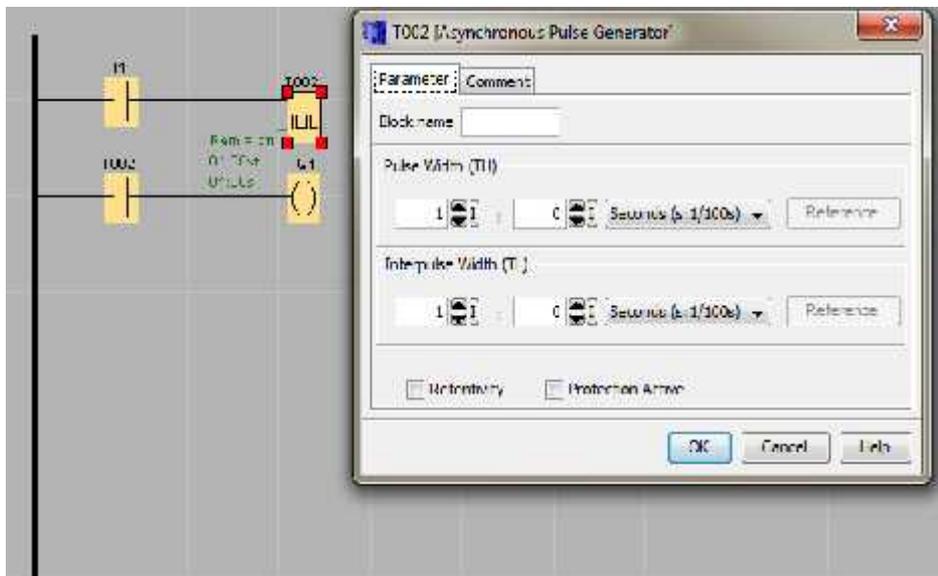
Berikut gambar Ladder Diagramnya:



**Gambar 2.9 Diagram Rangkaian XOR**

(Sumber : PT. Tambang Batu Bara Bukit Asam)

### 2.7.8 Diagram Rangkaian Pulsa Generator



**Gambar 2.10 Diagram Rangkaian Pulsa Generator**

(Sumber : PT. Tambang Batu Bara Bukit Asam)



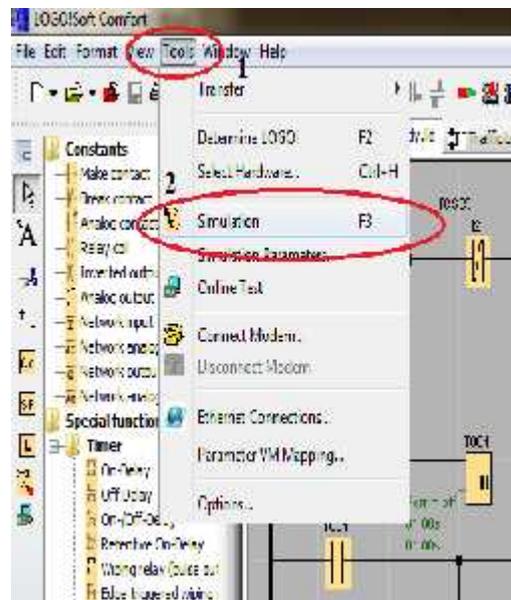
Rangkaian Pulsa, adalah bentuk diagram yang berfungsi untuk menghasilkan sinyal pulsa yang bersifat timing 0, 1 sesuai dengan waktu yang kita atur. Sesuai dengan kondisi gambar diatas, kita lihat bahwa pewaktu yang diatur adalah 1 detik untuk *Pulse Width* ( Lebar Pulsa ) dimana kondisi keluaran = 1 dan 1 detik untuk *Interpulse Widht* ( Lebar Pulsa ) Dimana kondisi keluaran = 0.

## 2.8 Simulasi ( Uji Diagram Rangkaian )

Setelah selesai membuat rangkaian diagram Ladder pada LOGO! Dapat dilakukan uji rangkaian atau simulasi dengan cara mengikuti tahap berikut:

1. **Klik Tools**
2. **Klik Simulation (F3)**

Lihat gambar berikut!

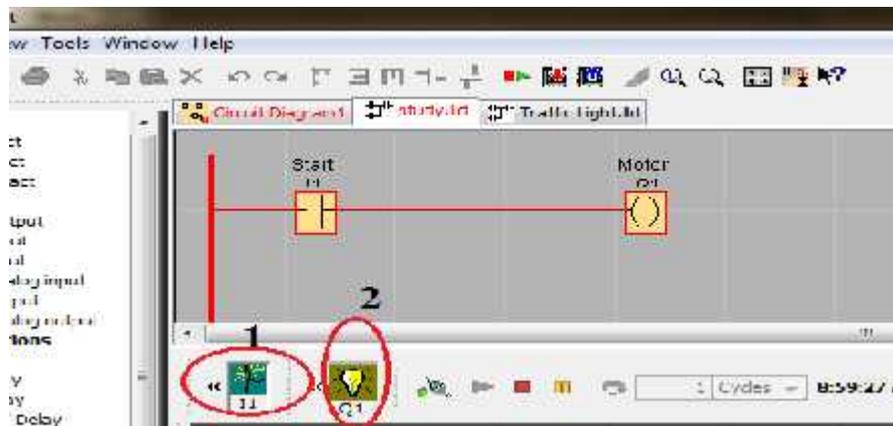


**Gambar 2.11 Uji Tes Simulasi Rangkaian**

(Sumber : PT. Tambang Batu Bara Bukit Asam)



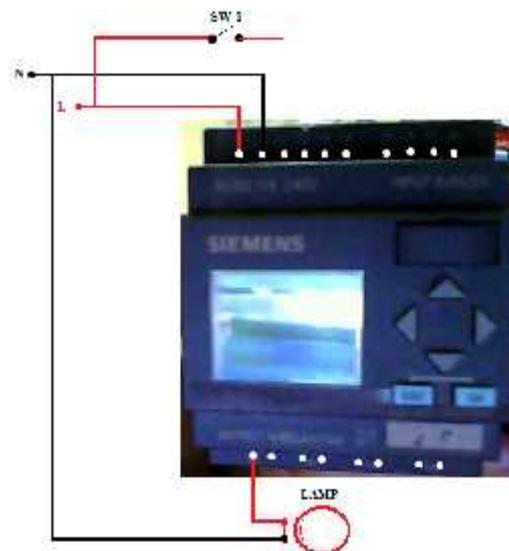
Setelah melakukan test simulation maka diagram ladder akan berwarna merah, yang berarti ada inputan tegangan, lihat gambar:



**Gambar 2.12 Contoh Pengujian Pada Rangkaian**

(Sumber : PT. Tambang Batu Bara Bukit Asam)

Pada gambar diatas, point 1 merupakan kontak yang digunakan pada LOGO! Modul, dimana I1 sebagai inputan yang digunakan. Jadi kalau kita lihat berdasarkan konfigurasi alat sebagai berikut:



**Gambar 2.13 Contoh Konfigurasi Hardware Pada Modul LOGO!**

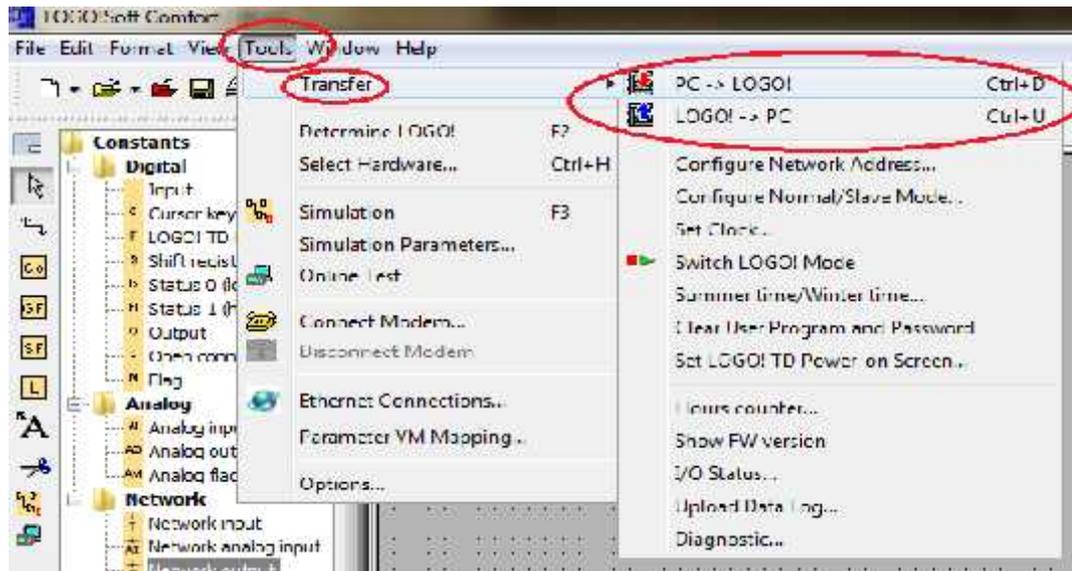
(Sumber : PT. Tambang Batu Bara Bukit Asam)



## 2.9 Cara Download dan Upload Program

Adapun cara men**Download**(Kirim Program dari PC Ke LOGO!) dan **Upload**(Ambil Program dari LOGO! Ke PC).

Lihat gambar dan lakukan tahap dibawah ini:



**Gambar 2.14 Cara Transfer LOGO! To PC dan PC to LOGO!**

(Sumber : PT. Tambang Batu Bara Bukit Asam)

### **Tahapan Download dan Upload Program!**

- **Klik Tools > Select Hardware**

Pilih Hardware sesuai dengan hardware yang dimiliki atau yang akan digunakan.

- **Klik OK**

dilanjutkan dengan >

- **Klik Tools > Transfer > PC -> LOGO! (Ctrl+D)**

Perintah diatas untuk Kirim Program dari Komputer ke Modul LOGO!

- **Klik Tools > Transfer > LOGO!->PC (Ctrl+ U)**

Perintah diatas untuk Kirim Program dari Modul LOGO! Ke PC.



### 2.10 Power Supply

Catu daya adalah sebuah<sup>2</sup> piranti elektronika yang berguna sebagai sumber daya untuk piranti lain, terutama daya listrik. Pada dasarnya pencatu daya bukanlah sebuah alat yang menghasilkan energi listrik saja, namun ada beberapa pencatu daya yang menghasilkan energi mekanik, dan energi yang lain.

Catu daya PLC berfungsi untuk memberikan sebuah tegangan masukan pada Mini PLC, yaitu tegangan DC sebesar 12 hingga 24 Vdc, biasanya tegangan minimum pada catu daya PLC ini adalah 12-24Vdc, namun tegangan yang dipakai untuk merancang alat ini menggunakan tegangan sebesar 12Vdc.



**Gambar 2.15 Power Supply 12 Vdc**

(Sumber : Dokumen Pribadi)

### 2.11 Terminal box cable

Merupakan suatu penghubung kabel dari lapangan ke ruang *relay* dengan alasan agar dapat dengan mudah melakukan perawatan serta melakukan pelacakan untuk mengatasi suatu halangan.

### 2.12 Sensor

Sensor (juga disebut detktor) adalah *converter* atau perubah / pemindah yang mengukur besaran fisik dan mengubahnya menjadi sinyal yang dapat dibaca oleh pengamat atau instrument (terutama alat elektronik).



**Sensor** adalah sesuatu yang digunakan<sup>3</sup> untuk mendeteksi adanya perubahan lingkungan fisik atau kimia. Variabel keluaran dari sensor yang diubah menjadi besaran listrik disebut Transduser. Pada saat ini, sensor tersebut telah dibuat dengan ukuran sangat kecil dengan orde nanometer. Ukuran yang sangat kecil ini sangat memudahkan pemakaian dan menghemat energi.

### 2.12.1 *Proximity Switch*

*Proximity Switch* atau disingkat TL atau ada juga yang menyebutnya **PR**, lain-lain memang tapi intinya sama. Secara bahasa *Proximity Switch* berarti, **proximity** artinya *jarak* atau *kedekatan*, sedangkan **switch** artinya saklar jaded efinisinya adalah sensor atau saklar otomatis yang mendeteksi logam berdasarkan jarak yang diperolehnya, artinya sejauhmana kedekatan object yang dideteksinya dengan sensor, sebab karakter dari sensor ini, mendeteksi object yang cukup dekat dengan satuan mili meter, umumnya sensor ini mempunyai jarak deteksi yang bermacam-macam seperti 2,5,7,10,12, dan 20 mm tergantung dari type sensor yang digunakan, semakin besar angka yang tercantum pada typenya, maka semakin besar pula jarak deteksinya. Jenis sensor proximity yang digunakan pada alat ini adalah jenis *Autonic Proxymity Sensor type PR12 2DP*.



**Gambar 2.16 Sensor *Proximity* PR12 2DP**

**(Sumber : Dokumen Pribadi)**



### **2.11.2 Cara kerja *Proximity Switch***

Seperti yang telah disebutkan diatas, sensor ini bekerja berdasarkan jarak objek terhadap sensor, ketika ada object yang mendekat kepadanya dengan jarak yang tidak telalu jauh 0-80cm, maka sensor akan bekerja dan menghubungkan kontaknya, kemudian melalui kabel yang tersedia bisa dihubungkan ke perangkat lainnya seperti lampu indikator, *relay* dll. Pada saat sensor ini sedang bekerja atau mendeteksi adanya objek maka akan ditandai dengan lampu kecil berwarna merah atau hijau yang ada dibagian atas sensor, sehingga memudahkan kita dalam memonitor kerja sensor atau ketika melakukan *preventive maintenace*.

Hampir setiap mesin - mesin produksi yang ada di setiap industri, baik itu industri kecil ataupun besar, menggunakan sensor jenis ini, sebab selain praktis sensor ini termasuk tahan terhadap benturan ataupun guncangan, selain itu mudah pada saat melakukan perawatan ataupun penggantian, sebab telah dirancang demikian oleh produsennya.

### **2.13 Motor DC**

Motor DC adalah suatu mesin yang berfungsi untuk mengubah energi listrik arus searah menjadi energi gerak atau energi mekanik. Motor DC terdiri dari dua bagian utama, yaitu rotor dan stator. Rotor adalah bagian yang berputar atau armature, berupa koil dimana arus listrik dapat mengalir. Stator adalah bagian yang tetap dan menghasilkan medan magnet dari koilnya.

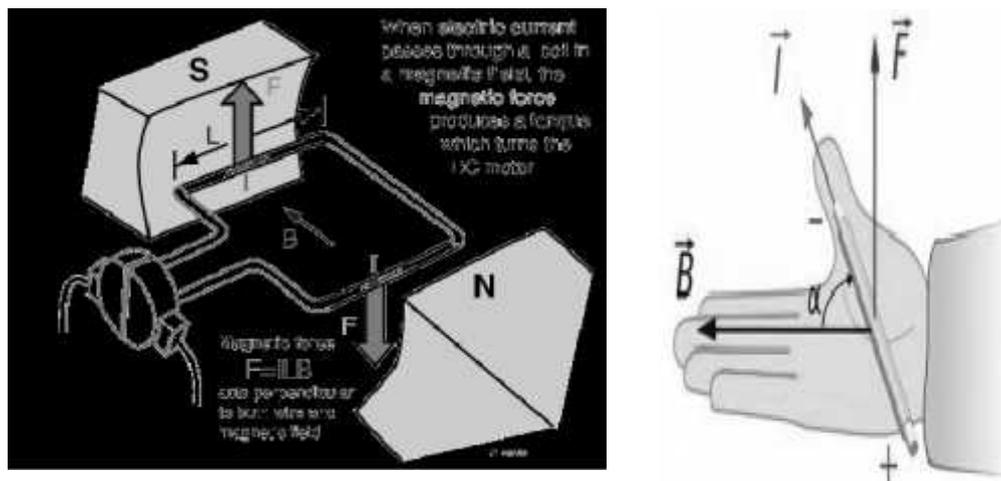
Motor DC memerlukan suplai<sup>4</sup> tegangan yang searah pada kumparan medan untuk diubah menjadi energi mekanik. Bagian utama motor DC adalah stator dan rotor dimana kumparan medan pada motor dc disebut stator (bagian yang tidak berputar) dan kumparan jangkar disebut rotor (bagian yang berputar). Bentuk motor paling sederhana memiliki kumparan satu lilitan yang bisa berputar bebas di antara kutub-kutub magnet permanen.

---



Catu tegangan dc dari baterai menuju ke lilitan melalui sikat yang menyentuh komutator, dua segmen yang terhubung dengan dua ujung lilitan. Kumparan satu lilitan disebut angker dinamo. Angker dinamo adalah sebutan untuk komponen yang berputar di antara medan magnet.

Prinsip kerja motor DC adalah jika kumparan dialiri arus listrik maka pada kedua kumparan akan bekerja gaya Lorentz. Pada gambar 2.12 dapat dilihat prinsip kerja gaya lorent, dimana gaya yang jatuh pada telapak tangan ( $F$ ), jari yang direntangkan menunjukkan arah medan magnet ( $B$ ), ibu jari menunjukkan arah arus listrik ( $I$ ). Motor DC biasanya mempunyai kecepatan putar yang cukup tinggi dan sangat cocok digunakan untuk roda robot yang membutuhkan kecepatan gerak yang tinggi.

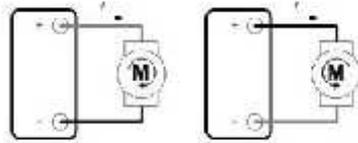


**Gambar 2.17** Prinsip gaya Lorentz

(Sumber: <http://hyperphysics.phy-astr.gsu.edu>)



Dengan berdasarkan pada prinsip gaya Lorentz, memberikan tegangan pada motor DC akan membuat motor berputar secara kontinyu kearah tertentu. Membalik arah putaran motor dapat dilakukan dengan mengubah polaritas arus yang mengalir pada motor. Gambar 2.13 memperlihatkan arah putaran motor DC berdasarkan polaritas arus yang mengalir.



**Gambar 2.18** Arah Perputaran Motor DC