

BAB II TINJAUAN PUSTAKA

2.1 *Programmable Logic Controller*

Programmable Logic Controller (PLC) dapat didefinisikan sebagai pengendali berbasis mikro komputer yang menggunakan instruksi-instruksi yang tersimpan dalam memori yang dapat diprogram untuk menerapkan logika, pengurutan (*sequencing*), *timing*, *counting* dan fungsi-fungsi aritmatika melalui modul *input / output* (I/O) digital atau analog untuk mengendalikan mesin dan proses. Pada saat ini, PLC digunakan baik pada aplikasi kendali kontinyu maupun kendali diskrit dalam kedua macam industri proses maupun industri manufaktur [3]. PLC dirancang untuk menggantikan suatu rangkaian relay sequensial dalam suatu sistem kontrol. Selain dapat diprogram, PLC juga dapat dikendalikan dan dioperasikan oleh pengguna yang tidak memiliki pengetahuan di bidang pengoperasian komputer secara khusus [4].

PLC yang berfungsi sebagai pengendali yang programnya dapat disesuaikan dengan kebutuhan pengguna, serta penyusunan program kontrolnya berdasarkan pada suatu rangkaian kelistrikan yang diaplikasikan kedalam pernyataan logika (*logic*). Dengan cara memasukkan program kedalam *input* data yang ada pada PLC melalui *programming console* dan *programming ladder* melalui PC.

Berdasarkan namanya konsep PLC terbagi atas 3 poin, sebagai berikut [5]:

- a. *Programmable*, menunjukkan kemampuan dalam hal memori untuk menyimpan program yang telah dibuat yang dengan mudah diubah-ubah fungsi atau kegunaannya.
- b. *Logic*, menunjukkan kemampuan dalam memproses *input* secara aritmatik dan logic (ALU), yakni melakukan operasi membandingkan, menjumlahkan, mengalikan, membagi, mengurangi dan lain sebagainya.
- c. *Controller*, menunjukkan kemampuan dalam mengontrol dan mengatur proses sehingga menghasilkan *output* yang diinginkan.

2.1.1 Jenis – Jenis PLC

Secara umum PLC terbagi menjadi dua jenis yaitu, PLC *Compact* dan PLC *Modular*:

a. PLC *Compact* [7]

PLC *Compact* atau yang disebut sebagai *based* mempunyai bagian-bagian penting yaitu, komponen-komponen prosessor, I/O, dan catu daya yang melekat menjadi satu bagian per satu unit yang tidak terpisahkan. Adapun ciri-ciri PLC *Compact* antara lain:

1. Seluruh komponen yang terdapat pada PLC (modul komunikasi, modul I/O, CPU, *power supply*) menjadi satu kesatuan
2. Ukurannya lebih kecil
3. Mempunyai jumlah I/O yang relatif sedikit dan tidak dapat ditambah karena sudah ditetapkan kapabilitas I/O oleh pabrikan bukan oleh pengguna.
4. Tidak dapat menambahkan modul – modul tambahan atau khusus

Pada gambar 2.1 merupakan contoh PLC *Compact* buatan dari perusahaan *Schneider Electric*:



Gambar 2. 1 PLC *Compact* TWDLCAE40DRF [4]

b. PLC Modular

PLC *Modular* merupakan PLC yang konstruksi pada tiap komponen komunikasi dan komponen I/O terpisah yang prosesnya menjadi bagian dari modul – modul yang ada pada modular. Adapun ciri – ciri PLC *Modular*, yaitu [7]

1. Seluruh komponen yang terdapat pada PLC (modul komunikasi, modul I/O, CPU, power supply) terpisah ke dalam modul – modul.
2. Berlaku dalam ukuran besar dengan konektivitas I/O, catu daya, kemampuan komputasi, dll.
3. Pada PLC *Modular*, jumlah input dan output tidak tetap dikarenakan Input dan output dapat ditambahkan ke sistem PLC *modular* oleh pengguna.
4. Beberapa komponen penambahan dapat dipasang pada chassis atau rak atau bus dengan slot yang berbeda.

Pada gambar 2.2 merupakan contoh PLC *Modular* buatan dari perusahaan *Omron*:



Gambar 2. 2 PLC *Modular* [7]

2.1.2 Komponen Pembentuk PLC

Sebuah sistem PLC pada umumnya mempunyai dua bagian yang terdiri dari Komponen Perangkat Keras dan Komponen Perangkat Lunak [8].

a. Komponen Perangkat Keras

Komponen perangkat keras pada PLC adalah :



1. CPU (*Central Processing Unit*)

CPU merupakan bagian dari komponen penyusun PLC yang berfungsi sebagai otak bagi sistem. CPU berisi mikroprosesor yang memproses semua program dari sinyal – sinyal *input* dan melaksanakan tindakan-tindakan pengontrolan sesuai dengan program yang telah tersimpan, lalu mengkomunikasikan keputusan – keputusan yang diambilnya sebagai sinyal kontrol ke *output interface* [9].

2. Unit *Memory*

Memory yang terdapat dalam PLC digunakan untuk menyimpan data dan program. Pada umumnya *memory* menjadi satu modul dengan CPU, jika terdapat memori berbentuk eksternal, maka itu merupakan memori tambahan. Secara fisik, *memory* pada PLC berupa chip dan untuk pengaman dipasang baterai *back-up* pada PLC. Unit *memory* dapat dibedakan atas 2 jenis, yaitu [9]:

a) *Volatile Memory*

Volatile Memory merupakan suatu memori yang apabila sumber tegangannya dilepas maka data yang tersimpan akan hilang atau tidak permanen. Untuk penyimpanan data dan program dalam jangka waktu yang lebih lama maka *memory* harus mendapatkan daya terus - menerus dengan menggunakan baterai. Ada beberapa jenis *volatile memory* yaitu RAM (*Random Access Memory*), SRAM (*Static RAM*) dan DRAM (*Dynamics RAM*).

b) *Non-Volatile Memory*

Non-Volatile memory merupakan suatu memori yang jika sumber tegangan dilepas, maka data yang tersimpan tidak akan hilang. Salah satu jenis *memory* ini adalah ROM (*Read Only Memory*) yang hanya dapat dibaca dan tidak dapat ditambah ataupun diubah. Isi dari ROM berasal dari pabrik pembuatnya yang berupa sistem operasi dan terdiri dari program-program pokok yang diperlukan oleh sistem PLC. Untuk mengubah isi dari ROM tersebut, maka diperlukan *memory* dengan jenis tipe EPROM (*Erasable Programmable ROM*) yang dapat dihapus dengan mengekspos chip pada cahaya ultra violet pekat.



3. *Power Supply*

Power supply merupakan penyedia daya bagi PLC. Unit *power supply* atau unit catu daya diperlukan untuk mengkonversi tegangan masukan AC (220Volt ~ 50Hz) atau DC (24Volt) sumber menjadi tegangan rendah DC 5 Volt yang dibutuhkan oleh prosesor dan rangkaian-rangkaian dalam *input/output interface* [10].

4. *Unit Programmer*

Komponen programmer merupakan alat yang digunakan untuk berkomunikasi dengan PLC melalui *programming port*. *Programmer* mempunyai beberapa fungsi pada PLC yaitu :

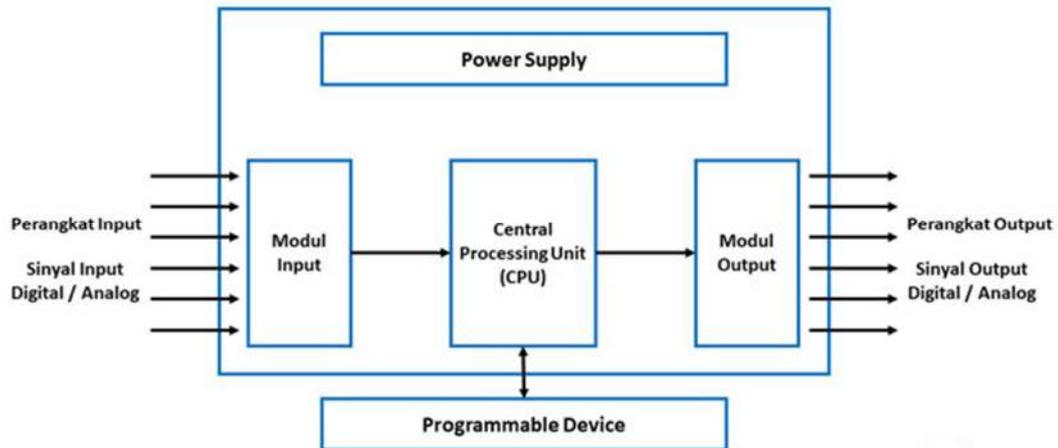
- a) *RUN*, untuk mengendalikan suatu proses saat program dalam keadaan aktif.
- b) *OFF*, untuk mematikan PLC sehingga program dibuat tidak dapat dijalankan.
- c) *MONITOR*, untuk mengetahui keadaan suatu proses yang terjadi dalam PLC.
- d) *PROGRAM*, menyatakan suatu keadaan dimana programmer/ monitor digunakan untuk membuat suatu program.

5. *Unit I/O (Input / Output)*

Unit I/O merupakan komponen PLC yang paling penting. Komponen I/O berfungsi untuk menyediakan antarmuka yang menghubungkan sistem dengan dunia luar. Keadaan tersebut memungkinkan dibuatnya sambungan-sambungan / koneksi antara perangkat-perangkat *input*, semisal sensor, dengan perangkat *output*, semisal motor dan selenoida, melalui kanal-kanal *input/output* yang tersedia.

Modul *input* pada PLC terdiri dari modul *input* analog dan modul *input* digital. Modul *input* analog mengubah tegangan (0-10 VDC) atau arus dari (0-20 mA) menjadi sinyal – sinyal yang mampu dimengerti dan diproses oleh CPU. Modul *input* analog biasa digunakan untuk peralatan input seperti *flow sensor*, Motor AC, *pressure transducer*, dan lain-lain. Sedangkan *input* digital digunakan untuk menerima sinyal *on / off* dari peralatan seperti *switch*, *push button*, dan lain-lain [11].

Modul *output* pada PLC juga terdiri dari modul *output* analog dan modul *output* digital. Modul *output* analog digunakan untuk peralatan seperti *flow controller*, *pressure regulator* atau *position controller*. Sedangkan *output* digital dapat digunakan untuk peralatan elektronika seperti LED, motor, *relay*, *alarm*, dan lain-lain [11].



Gambar 2. 3 Blok Diagram PLC [8]

2.1.3 Bahasa Pemrograman PLC

Terdapat beberapa bahasa pemrograman pada PLC yang dapat digunakan untuk berkomunikasi. Menurut *International Electrotechnical Commission* (IEC) dikenal dengan IEC 1131-3 terdapat 5 bahasa pemrograman PLC (Crispin, 1997), yaitu [12]:

1. *Ladder Diagram* / Diagram Ladder, merupakan salah satu bahasa pemrograman PLC yang memiliki garis vertikal yang disebut garis bar dan instruksi-instruksi yang digunakan berjajar dari kiri ke kanan dan dari atas ke bawah.
2. *Function Block Diagram* (FBD), merupakan salah satu bahasa pemrograman yang menitik beratkan pada hubungan antara variabel *input* dan *output* yang terdapat pada blok - blok diagram. Setiap blok memiliki logika tertentu yang dapat digunakan kembali didalam bagian yang berbeda dalam sebuah aplikasi.
3. *Structure Text* (ST), merupakan salah satu bahasa pemrograman PLC yang termasuk bahasa tingkat tinggi yang digunakan untuk beberapa kondisi yang kompleks. *Structure Text* menggunakan bahasa yang baku untuk menyatakan



kondisi setiap *step* yang berbeda dengan bahasa yang digunakan mirip dengan bahasa pemrograman pada Pascal.

Pada Laporan Akhir ini penulis menggunakan bahasa pemrograman *Ladder Diagram* untuk PLC Schneider TWDLCAE40DRF TwidoSuite yang digunakan.

2.1.4 Ladder Diagram PLC

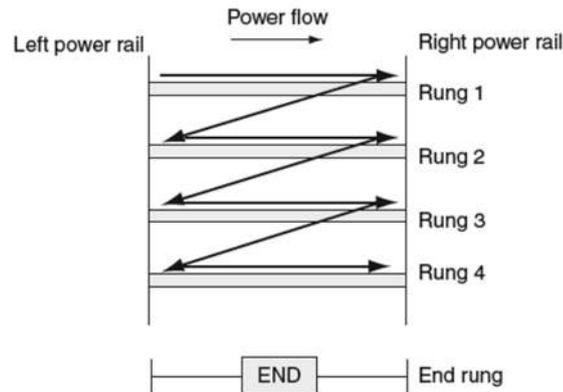
Ladder Diagram adalah bahasa pemrograman universal PLC. *Ladder Diagram* merupakan salah satu bahasa pemrograman tertua yang di gunakan untuk memprogram PLC. Pada *Ladder diagram*, bahasa pemrograman yang digunakan untuk membuat program untuk mengontrol sistem PLC disebut dengan '*Ladder Diagram Language*' atau '*Ladder Logic Language*'. Dalam menggambarkan sebuah diagram tangga, diterapkan konvensi-konvensi tertentu, yaitu [13]:

1. Garis-garis vertikal diagram merepresentasikan rel-rel daya, dimana di antara keduanya komponen-komponen rangkaian tersambung
2. Tiap-tiap anak tangga mendefinisikan sebuah operasi dalam proses kendali
3. Sebuah diagram tangga dibaca dari kiri ke kanan. Anak tangga teratas dibaca dari kiri ke kanan dan demikian seterusnya. Prosedur membaca semua anak tangga program ini disebut sebagai sebuah siklus.
4. Tiap-tiap anak tangga harus dimulai dengan sebuah *input* atau sejumlah *input* dan harus berakhir dengan setidaknya sebuah *output*.
5. Perangkat-perangkat listrik ditampilkan dalam kondisi normalnya. Dengan demikian, sebuah sakelar yang dalam keadaan normalnya terbuka hingga suatu objek menutupnya, diperlihatkan sebagai terbuka pada diagram tangga, demikian pula sebaliknya
6. Sebuah perangkat tertentu dapat digambarkan pada lebih dari satu anak tangga. Huruf-huruf atau nomor-nomor dipergunakan untuk memberi label bagi perangkat tersebut pada tiap-tiap situasi kendali yang dihadapinya.



7. *Input* dan *output* seluruhnya diidentifikasi melalui alamat-alamat yang terdaftar dan notasi yang dipergunakan bergantung pada pabrikan PLC yang bersangkutan.

Pada gambar 2.4 merupakan alur cara membaca *Ladder Diagram* pada PLC:



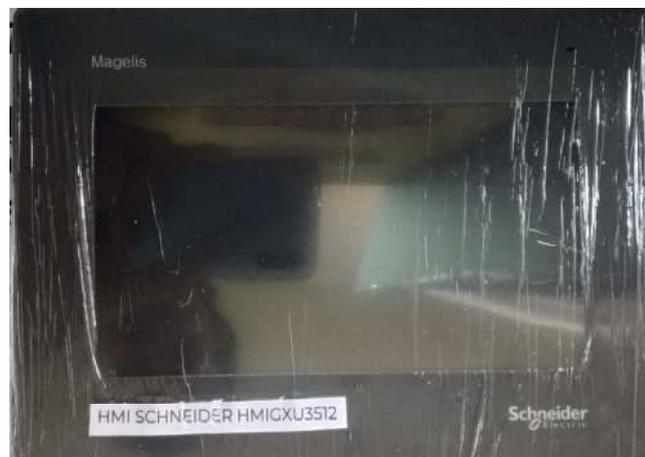
Gambar 2. 4 *Ladder Diagram* [12]

2.2 *Human Machine Interface (HMI)*

Human Machine Interface (HMI) adalah sebuah sistem yang dapat mempertemukan manusia dengan teknologi mesin. HMI merupakan antarmuka pengguna atau *dashboard* yang menghubungkan seseorang ke mesin, sistem, atau perangkat. Sementara istilah secara teknis dapat diterapkan ke layar apapun yang memungkinkan pengguna untuk berinteraksi dengan suatu perangkat, HMI yang paling umum digunakan dalam konteks dari proses industri [15]. HMI berupa pengendali dan menunjukkan status, baik dilakukan secara manual ataupun disajikan dengan visualisasi komputer yang bersifat *real time*. HMI juga dapat disebut sebagai *user interface* dan sistem kontrol untuk manufaktur.

Tugas dari HMI adalah membuat visualisasi dari sebuah teknologi atau sistem secara *real time* sehingga dengan menggunakan desain HMI yang bisa disesuaikan dapat memudahkan pekerjaan fisik. Tujuan dari HMI adalah untuk menambah tingkat interaksi antara mesin dan operator melalui tampilan yang ada di layar komputer dan memenuhi kebutuhan para pengguna terhadap informasi pada sistem [14].

Teknologi HMI digunakan oleh hampir semua organisasi industri, serta berbagai perusahaan lain seperti bidang energi, makanan dan minuman, manufaktur, minyak dan gas, daya pembangkit, daur ulang, transportasi, air dan *wastewater*, dan lain-lain untuk berinteraksi dengan mesin mereka dan mengoptimalkan proses industri mereka. Peran yang paling umum berinteraksi dengan HMIs adalah operator, integrasi sistem, dan insinyur teknik, terutama kontrol sistem insinyur. HMIs adalah sumber daya penting bagi para profesional ini, yang menggunakannya untuk meninjau dan memantau proses, mendiagnosa masalah, dan visualisasi data [15].



Gambar 2. 5 *Human Machine Interface (Magelis GXU3512)*

Berikut ini merupakan spesifikasi dari HMI GXU3512

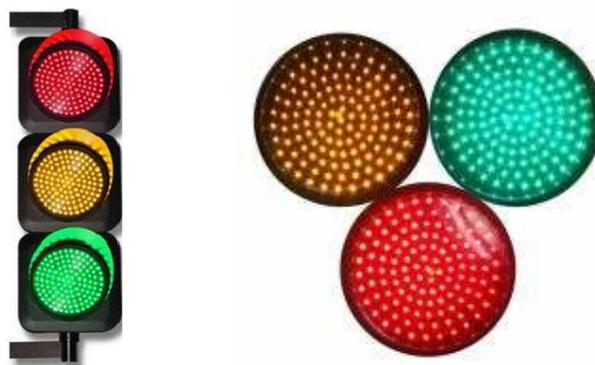
1. Range of product : Magelis HMIGXU
2. Product or component type: Advanced Touchscreen Panel
3. Display type : TFT Colour LCD
4. Display colour : 65,536 colours
5. Display resolution : 800 x 480 pixel WVGA
6. Display size : 7 inch wide screen
7. Software type : Configuration software
8. Software designation : Vijeo Designer Basic
9. Operating system : Magelis
10. Memory description : 128MB NAND flash, 128MB Main

- memory data with lithium battery backup
128KB flash user application 48MB
11. Integrated connection : Serial interface RS-232C and RS-422/485 with speed transmission 2,400 – 115,200 bps (byte per second); USB type A V2.0, USB mini_B V2.0; Ethernet port (RJ45).
12. Cut-out dimension : 207 x 152,4 mm

2.3 Lampu Lalu Lintas

Lalu-lintas (*traffic*) adalah kegiatan lalu-lalang atau gerak kendaraan atau orang dijalanan. Masalah yang dihadapi dalam perlalu-lintasan adalah keseimbangan antara kapasitas jaringan jalan dengan banyaknya kendaraan dan orang yang berlalu- lalang menggunakan jalan tersebut [16].

Lampu lalu lintas menurut UU No. 22/2009 Tentang Lalu lintas dan Angkutan Jalan: “Alat Pemberi Isyarat Lalu Lintas atau APILL adalah lampu yang mengendalikan arus lalu lintas yang terpasang di persimpangan jalan, tempat penyebrangan pejalan kaki (*zebra cross*), dan tempat arus lalu lintas lainnya” [17]. Lampu ini akan menandakan kapan kendaraan harus berjalan dan berhenti secara bergantian dari berbagai arah. Lampu lalu lintas menggunakan warna yang diakui secara universal seperti terlihat pada gambar 2.6, untuk menandakan berhenti adalah warna merah, hati-hati yang ditandai dengan warna kuning, dan hijau yang berarti dapat berjalan.



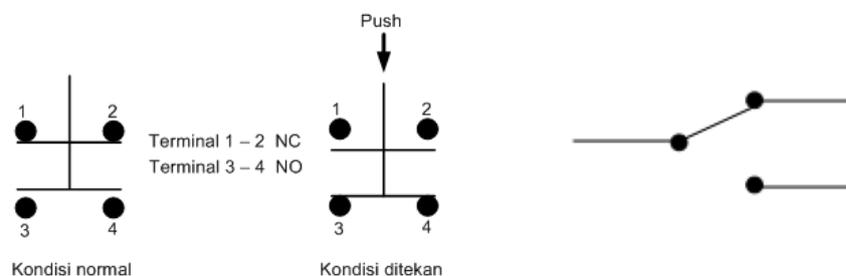
Gambar 2. 6 Modul Lampu Lalu Lintas 3 Warna [16]

2.4 Saklar Tombol Tekan (*Push Button Switch*)

Saklar Tombol Tekan atau *Push Button Switch* adalah perangkat / saklar sederhana yang berfungsi untuk menghubungkan atau memutuskan aliran arus listrik dengan sistem kerja tekan *unlock* (tidak mengunci). Sistem kerja *unlock* disini berarti saklar akan bekerja sebagai *device* penghubung atau pemutus aliran arus listrik saat tombol ditekan, dan saat tombol tidak ditekan (dilepas), maka saklar akan kembali pada kondisi normal [18].

Sebagai *device* penghubung atau pemutus, push button switch hanya memiliki 2 kondisi, yaitu *ON* dan *OFF* (1 dan 0). Istilah *ON* dan *OFF* ini menjadi sangat penting karena semua perangkat listrik yang memerlukan sumber energi listrik pasti membutuhkan kondisi *ON* dan *OFF*. Berdasarkan fungsi kerjanya yang menghubungkan dan memutuskan, push button switch mempunyai 2 tipe kontak yaitu *NC* (*Normally Close*) dan *NO* (*Normally Open*) [18].

1. *NC* (*Normally Close*), merupakan kontak terminal dimana kondisi normalnya tertutup (mengalirkan arus listrik). Dan ketika tombol saklar push button ditekan, kontak *NC* ini akan menjadi membuka (*Open*), sehingga memutus aliran arus listrik. Kontak *NC* digunakan sebagai pemutus atau mematikan sistem *circuit* (*Push Button Off*).
2. *NO* (*Normally Open*), merupakan kontak terminal dimana kondisi normalnya terbuka (aliran arus listrik tidak mengalir). Dan ketika tombol saklar ditekan, kontak yang *NO* ini akan menjadi menutup (*Close*) dan mengalirkan atau menghubungkan arus listrik. Kontak *NO* digunakan sebagai penghubung atau menyalakan sistem *circuit* (*Push Button ON*).

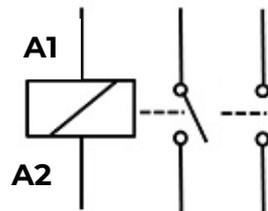


Gambar 2. 7 Prinsip Kerja *Push Button* dan Simbol SPDT *Emergency Switch* [18]

Pada Laporan Akhir ini, penulis menggunakan *Push Button Switch Emergency* bertipe SPDT (*Single Pole Double Throw*).

2.5 Relay

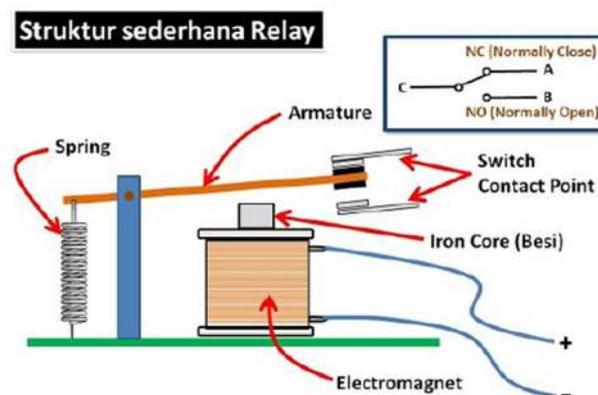
Relay adalah Saklar (*Switch*) yang dioperasikan secara listrik dan merupakan komponen Elektromekanikal (*Electromechanical*) yang terdiri dari 2 bagian utama yakni Elektromagnet (*Coil*) dan Mekanikal (seperangkat kontak saklar/*Switch*). *Relay* menggunakan prinsip elektromagnetik untuk menggerakkan kontak saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi [19].



Gambar 2. 8 Simbol *Relay* [19]

Pada dasarnya, *Relay* terdiri dari 4 komponen dasar yaitu :

1. Elektromagnet (*Coil*)
2. *Armature*
3. *Switch Contact Point* (Saklar)
4. Spring



Gambar 2. 9 Struktur Sederhana *Relay* [19]

Kontak Poin (*Contact Point*) *Relay* terdiri dari 2 jenis yaitu [19]:

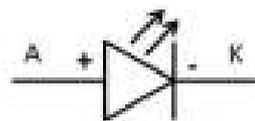
1. *Normally Close* (NC) yaitu kondisi awal sebelum diaktifkan akan selalu berada di posisi *CLOSE* (tertutup)
2. *Normally Open* (NO) yaitu kondisi awal sebelum diaktifkan akan selalu berada di posisi *OPEN* (terbuka)

Karena *Relay* merupakan salah satu jenis dari Saklar, maka istilah *Pole* dan *Throw* yang dipakai dalam Saklar juga berlaku pada *Relay*.

1. ***Pole*** : Banyaknya Kontak (*Contact*) yang dimiliki oleh sebuah *relay*
2. ***Throw*** : Banyaknya kondisi yang dimiliki oleh sebuah kontak (*Contact*)

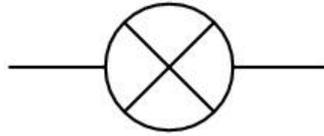
2.6 *Light Emitting Diode* (LED)

Light Emitting Diode atau disingkat dengan LED adalah komponen elektronika yang dapat memancarkan cahaya monokromatik ketika diberikan tegangan maju. LED merupakan keluarga dioda yang terbuat dari bahan semikonduktor. Warna-warna Cahaya yang dipancarkan oleh LED tergantung pada jenis bahan semikonduktor yang digunakan. LED juga dapat memancarkan sinar inframerah yang tidak tampak oleh mata seperti *remote control TV* ataupun *Remote Control* perangkat elektronik lainnya [20]. Bentuk LED mirip dengan sebuah bohlam (bola lampu) yang kecil dan dapat dipasangkan dengan mudah ke dalam berbagai perangkat elektronika. Berbeda dengan lampu pijar, LED tidak memerlukan pembakaran filamen sehingga tidak menimbulkan panas dalam menghasilkan cahaya.



Gambar 2. 10 Simbol LED [20]

Pada Gambar 2.11 adalah Pilot Lamp 12V DC sebagai LED yang digunakan pada Laporan Akhir ini dalam bentuk modul lampu lalu lintas.



Gambar 2. 11 Simbol *Pilot Lamp LED*

2.7 Hub

Untuk menghubungkan PLC dan HMI maka di perlukan Hub. Hub adalah istilah umum yang digunakan untuk menerangkan sebuah *central connection point* untuk komputer pada *network*. Hub ditunjukkan pada gambar 2.13.



Gambar 2. 12 Hub

Fungsi dasar yang dilakukan oleh hub adalah menerima sinyal dari satu komputer dan mentransmisikannya ke komputer yang lain. Sebuah hub bisa *active* atau *passive*. *Active* hub bertindak sebagai *repeater*; hub meregenerasi dan mengirimkan sinyal yang diperkuat. *Passive* hub hanya bertindak sebagai kotak sambungan; ia membagi/memisahkan sinyal yang masuk untuk ditransmisikan ke seluruh *network*. Hub adalah *central* untuk topologi star dan memungkinkan komputer untuk ditambahkan atau dipindahkan pada *network* dengan relatif mudah. Kapabilitas yang disediakan hub *central* untuk topologi star dan memungkinkan komputer untuk ditambahkan atau dipindahkan pada *network* dengan relatif mudah.

2.8 Software TwidoSuite

TwidoSuite adalah sebuah perangkat lunak yang diciptakan untuk memudahkan dalam pembuatan program otomatis kontrol listrik dari suatu proses

atau sistem. TwidoSuite menggunakan bahasa khusus yang sudah dikenal dengan baik oleh mereka yang berkecimpung di bidang teknik kontrol listrik yaitu *ladder diagram* dan *statement list*. TwidoSuite menyediakan banyak fasilitas yang memadai sehingga memberikan kemudahan dan sangat membantu dalam pembuatan kontrol otomatis yang terprogram [22]. Beberapa fasilitas itu antara lain adalah penyuntingan (*editing*), dokumentasi *hard* dan *soft*, transfer (*download* dan *upload*), verifikasi, pengujian (*running test*), pemantauan (*monitoring*). Penyuntingan (*editing*) meliputi antara lain pembuatan program (*create*), penghapusan (*clear*), penggantian (*replace*), penggandaan (*copy paste*), pencarian (*search*), pemeriksaan (*check*).

2.8.1 Pembuatan Program pada *Software* TwidoSuite

Terdapat beberapa langkah untuk menjalankan atau membuat program pada *software* TwidoSuite, yaitu

1. Klik icon TwidoSuite pada desktop atau pada start menu seperti Gambar 2.14 di bawah ini



TwidoSuite

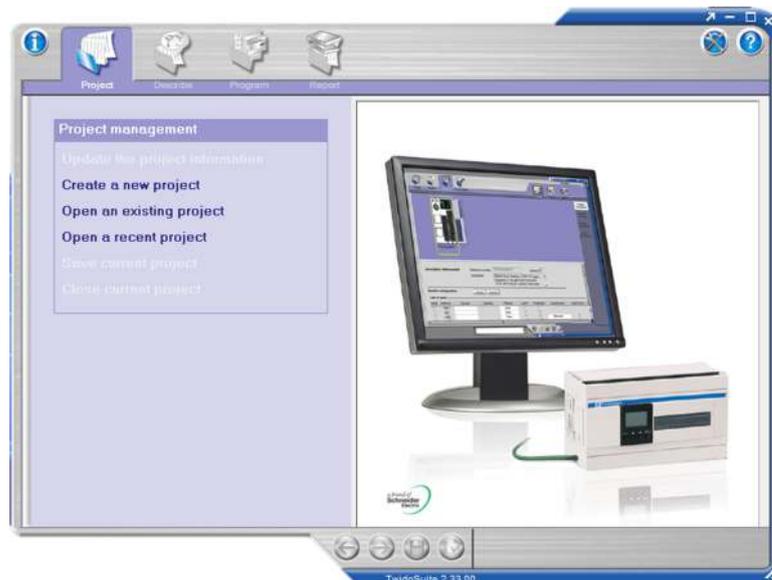
Gambar 2. 13 Icon TwidoSuite

2. TwidoSuite akan menampilkan tampilan jendela seperti pada Gambar 2.14. Pilih “Programming Mode” untuk membuat program PLC



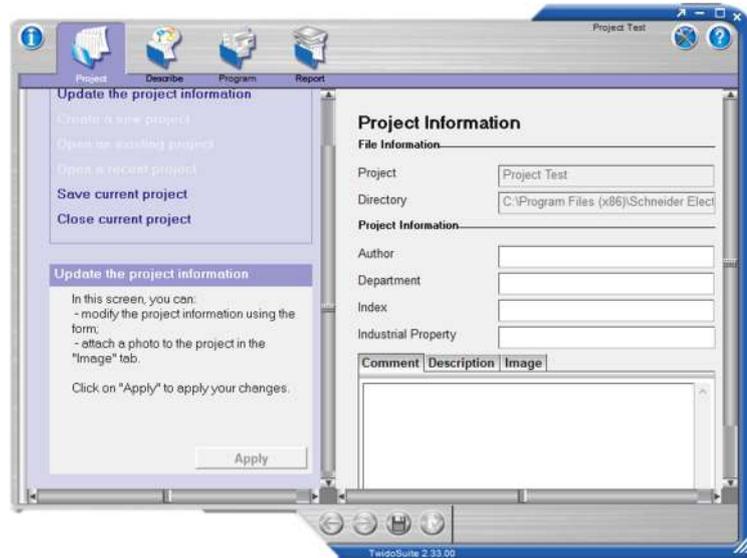
Gambar 2. 14 Tampilan Awal Jendela TwidoSuite

3. Selanjutnya akan tampil *project management* seperti Gambar 2.15. Programmer diberi pilihan untuk membuat program baru atau membuka program yang sudah jadi atau membuka program yang terakhir dibuka. Pilih *Create a new project*.



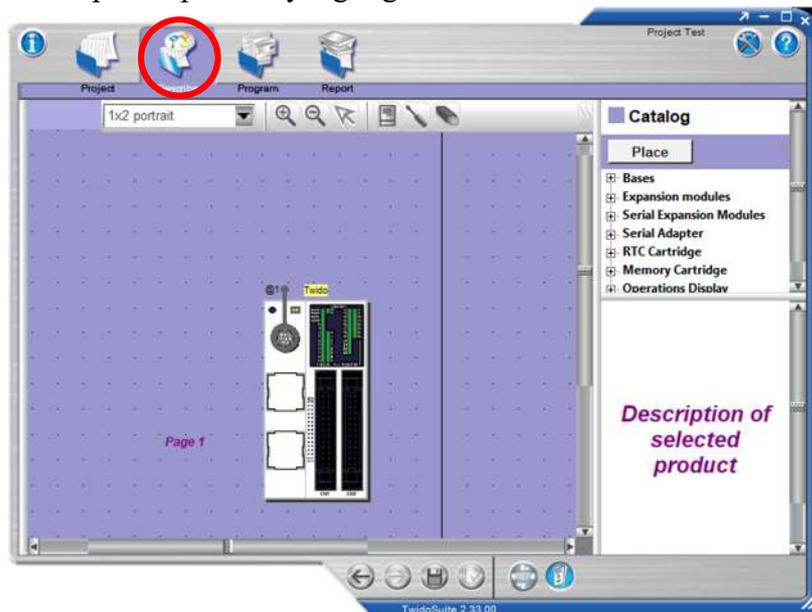
Gambar 2. 15 Tampilan *Project Management*

4. Buat nama *project* dan *directory* tempat *project* yang akan disimpan pada *menu project information* seperti pada Gambar 2.16. Kemudian klik *Create*.

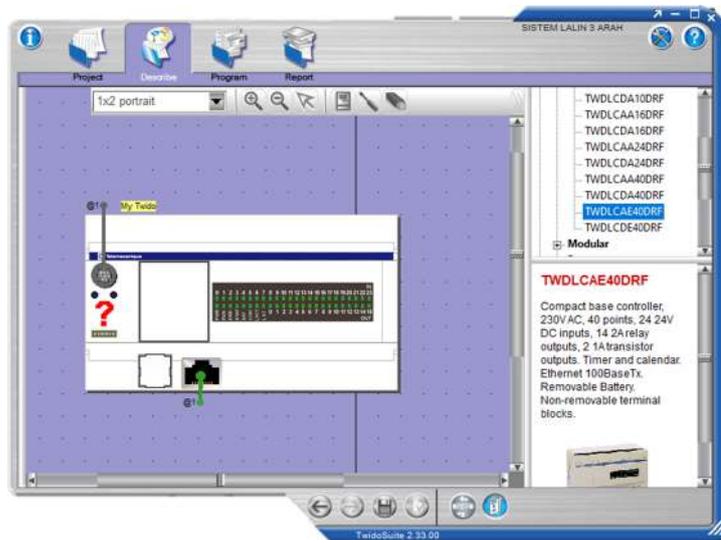


Gambar 2. 16 Tampilan *Project Information*

5. Ketika sudah muncul tampilan layar yang diperbarui atau *refresh*, klik *describe*. Untuk memasukkan tipe PLC yang digunakan pada “Bases, Compact” dan *output module* pada “Expansion Module” yang digunakan ada pada *menu catalog*. Lakukan *drag and drop* untuk menetapkan tipe PLC yang digunakan.

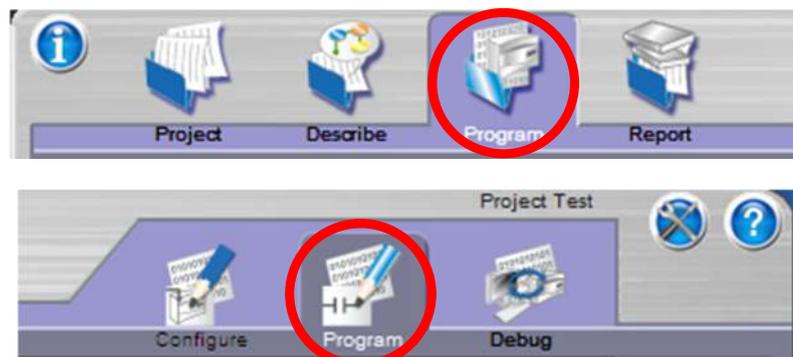


Gambar 2. 17 Tampilan *Describe*



Gambar 2. 18 Layar *Setting* Tipe PLC TWDLCAE40DRF

6. Setelah setting PLC selesai, pilih pada bagian kiri atas “Program” untuk menuju editor program PLC seperti pada Gambar 2.19 dibawah ini:



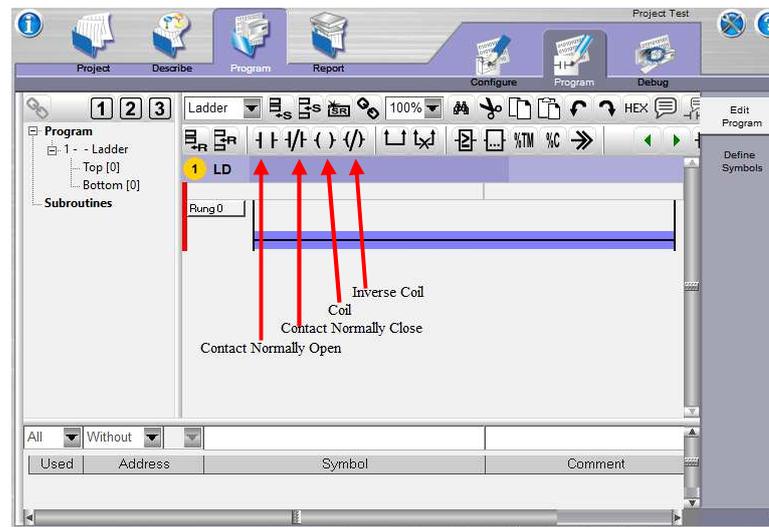
Gambar 2. 19 Menu Program

7. Setelah diklik “Program”, maka langkah selanjutnya adalah membuka *frame* untuk membuat *Ladder Diagram* dan muncul tampilan “Insert a section to begin”. Lalu klik *add a section* untuk membuat rung.



Gambar 2. 20 Tampilan Awal *Editor Ladder Diagram*

8. Setelah diklik *add a section*, maka akan muncul tampilan rung seperti pada Gambar 2.21. Untuk Mengedit ladder dilakukan menggunakan menu *program editor*, *program tool bar*, *ladder palette tool bar*, dan *extended ladder palette*.



Gambar 2. 21 *Editor Ladder Diagram* dan fungsi-fungsi dasar PLC

2.9 *Software Vijeo Designer*

Vijeo designer adalah konfigurasi berbasis tampilan dan program perangkat lunak yang digunakan untuk memprogram *Human mechine Interface* (HMI) dengan

merek magelis yang dikembangkan oleh Schneider Electric. Vijeo designer menawarkan sebuah *user interface* canggih yang mirip dengan Microsoft Visual Studio. Didalam vijeo designer terdapat aplikasi yang lengkap untuk memungkinkan proyek yang akan dikembangkan dengan cepat dan mudah diantaranya yaitu aplikasi.

2.9.1 Pembuatan Program pada *Software Vijeo Designer* [23]

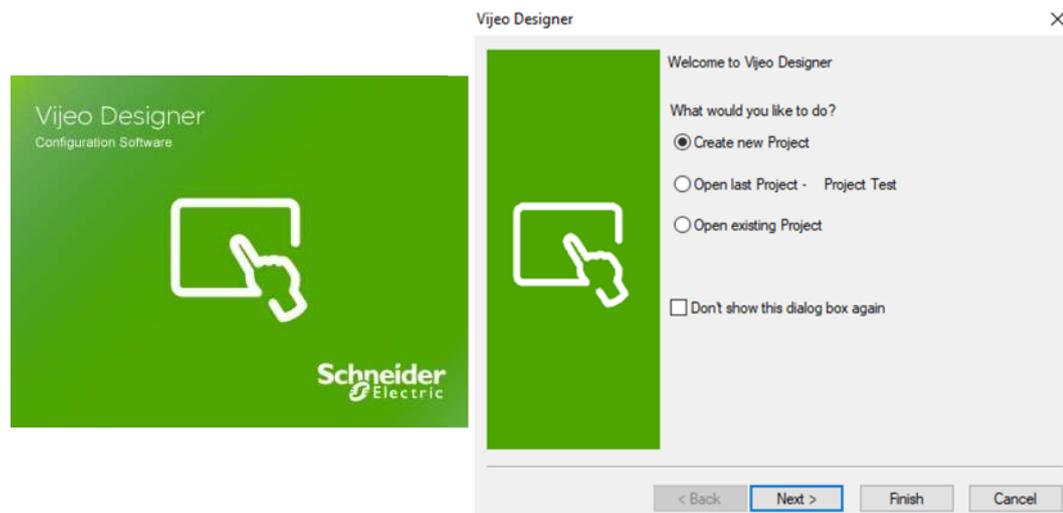
Terdapat beberapa langkah untuk menjalankan atau membuat program pada *software Vijeo Designer*, yaitu

1. Klik icon Vijeo Designer Basic V1.1 pada desktop atau pada start menu seperti Gambar 2.22 di bawah ini



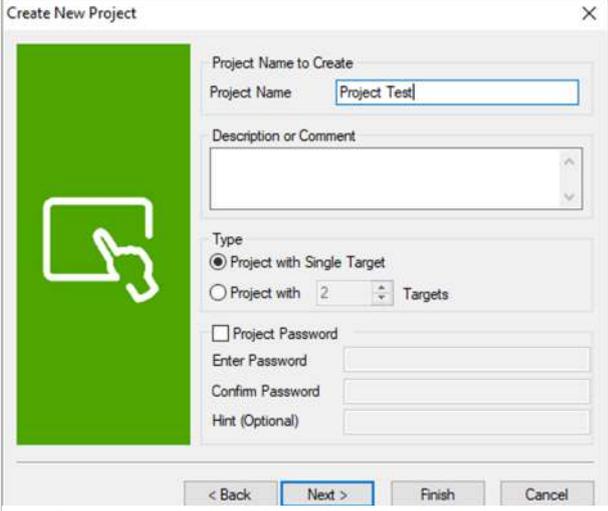
Gambar 2. 22 Icon Vijeo Designer Basic 1.1

2. Vijeo Designer akan menampilkan tampilan *loading* screen dan tampilan jendela seperti pada Gambar 2.26. Pilih “Create a New Project” untuk membuat program tampilan HMI.



Gambar 2. 23 Loading Screen dan Tampilan Awal Vijeo Designer

- Masukkan nama proyek yang diinginkan dan klik Berikutnya / *next*.



Project Name to Create

Project Name

Description or Comment

Type

Project with Single Target

Project with Targets

Project Password

Enter Password

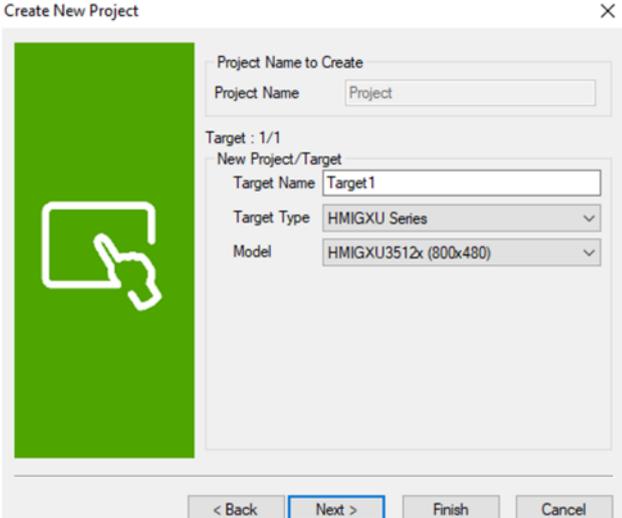
Confirm Password

Hint (Optional)

< Back **Next >** Finish Cancel

Gambar 2. 24 Tampilan *Project Setting*

- Dalam kasus penamaan target, ketik "Target 1". Pilih tipe target, seri HMIGXU Series dan model HMIGXU3512x (800x480). Klik "Next".



Project Name to Create

Project Name

Target : 1/1

New Project/Target

Target Name

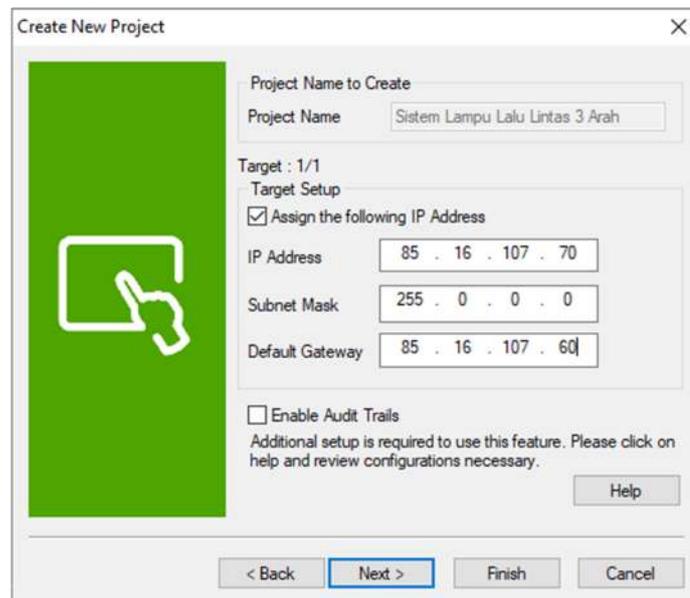
Target Type

Model

< Back **Next >** Finish Cancel

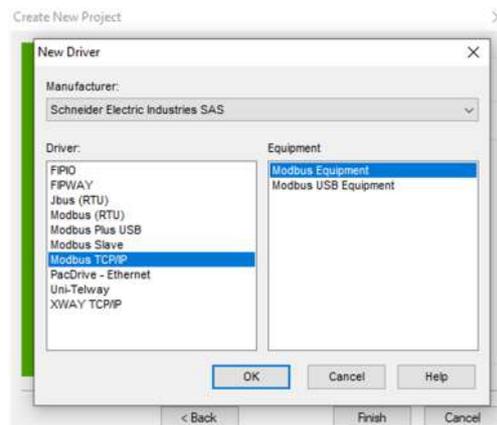
Gambar 2. 25 *Setting Target HMI*

5. Pilih alamat IP jika model memakai port *Ethernet* lalu klik Berikutnya.



Gambar 2. 26 Pengaturan IP Address pada HMI

6. Pilih driver yang relevan untuk tipe perangkat dengan menggunakan tombol Tambah. Sebagai contoh pada Gambar 2.27, pilih Schneider Electric Industries SAS sebagai produsen, Modbus TCP/IP sebagai sopir, dan Modbus Equipment sebagai peralatan. Kemudian klik Selesai.

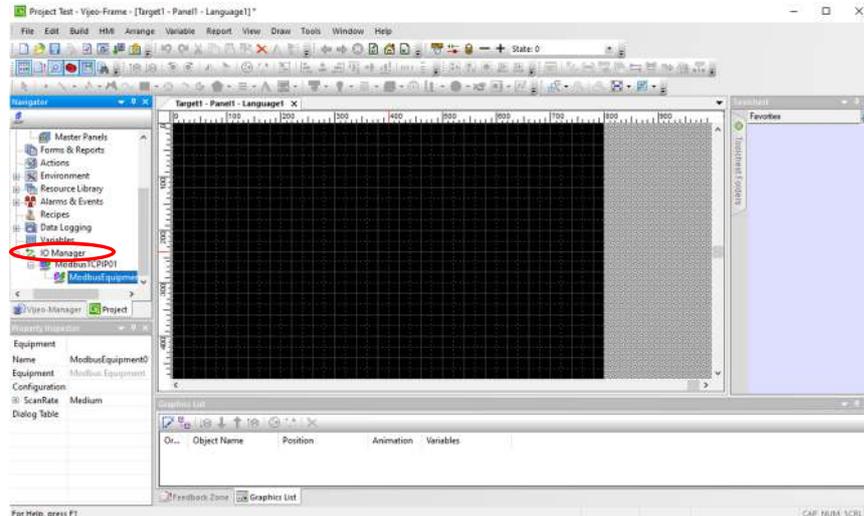


Gambar 2. 27 Pengaturan Driver HMI

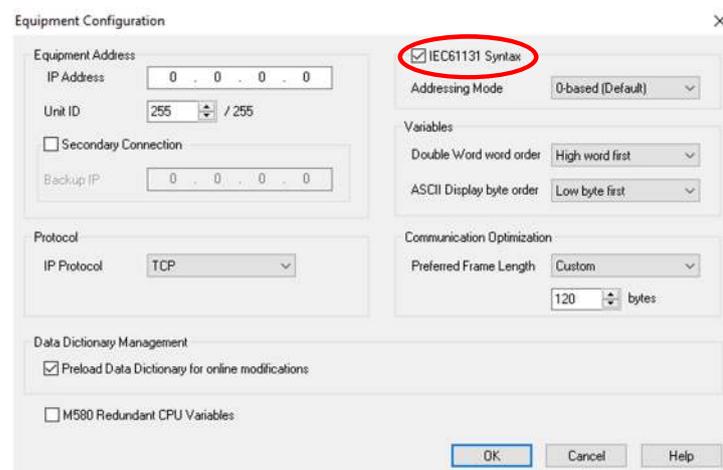
7. Atur sintaks IEC 6131 dalam kotak dialog konfigurasi peralatan. Untuk melakukan ini, di bawah [I/O Manager] di jendela [Navigator], klik



ganda peralatan untuk driver untuk membuka kotak dialog konfigurasi peralatan. Kemudian pilih [sintaks IEC61131] kotak cek.



Gambar 2. 28 Tampilan Awal *Editing*



Gambar 2. 29 Tampilan Jendela *Equipment Configuration*