

Pengendalian Electronic Home Appliances Berbasis IP dengan Menggunakan Modul Wiznet NM7010A

Asep Saefullah¹
Augury El Rayeb²

Email : asepsaepullah@pribadiraaharja.com; augury@pribadiraaharja.com

Diterima: 4 Agustus 2009 / Disetujui: 26 Agustus 2009

ABSTRACT

Use of remote control system has been increasing in line with the globalization era in which human movement and the broad movement and quick. At this time the public has known that for the controlling of an electronic home appliances from remote can use the remote control. The problem of remote control by the limited distance between the signal emanated by the remote and the signal received by the electronic home appliances, when the distance between home electronic appliances which are controlled with a remote / controller through limit tolerance, the electronic home appliances can not function according to the desired. To solve the problem then designed a system that can control home electronic appliances from a remote so that it does not terkendala again by distance, time and place. Design this system using the network technology that can be accessed anywhere and at any time during the availability of network. Internet network used by TCP / IP-based module starter kit network NM7010A-LF as a bridge between the low cost AVR microcontroller micro system with a computer network for controlling of the equipment electronic, low cost AVR micro system works as a web server. The result is a prototype of electronic home appliances that can control home electronic appliances from a remote that is not by the distance, place and time so that ultimately improve the effectiveness, efficiency and comfort in control.

ABSTRAKSI

Penggunaan sistem pengontrolan jarak jauh semakin meningkat sejalan dengan era globalisasi di mana perpindahan dan pergerakan manusia semakin luas dan cepat. Pada saat ini sudah dikenal masyarakat luas bahwa untuk melakukan pengontrolan terhadap suatu electronic home appliances dari jarak jauh dapat menggunakan remote kontrol. Permasalahannya penggunaan remote kontrol terkendala oleh keterbatasan jarak antara sinyal yang dipancarkan oleh remote dan sinyal yang diterima oleh electronic home appliances, apabila jarak antara electronic home appliances yang dikontrol dengan remote/pengontrol melewati batas

-
- 1. Dosen Jurusan Sistem Komputer, STMIK Raharja**
Jl. Jend Sudirman No.40 Modern Cikokol-Tangerang Telp 5529692
 - 2. Dosen Jurusan Sistem Komputer, STMIK Raharja**
Jl. Jend Sudirman No.40 Modern Cikokol-Tangerang Telp 5529692

toleransinya, maka *electronic home appliances* tersebut tidak dapat berfungsi sesuai yang diinginkan. Untuk memecahkan permasalahan tersebut maka dirancang sebuah sistem yang dapat melakukan pengontrolan *electronic home appliances* dari jarak jauh sehingga tidak terkendala lagi oleh jarak, tempat dan waktu. Perancangan sistem ini dengan memanfaatkan teknologi jaringan internet yang dapat diakses di mana saja dan kapan saja selama tersedianya jaringan. Jaringan internet dimanfaatkan TCP/IP starter kit berbasis modul jaringan NM7010A-LF sebagai jembatan antara mikrokontroler AVR low cost micro system dengan jaringan komputer untuk melakukan pengontrolan terhadap peralatan elekttronika, AVR low cost micro system berfungsi sebagai sebuah web server. Hasilnya berupa sebuah prototype pengendalian *electronic home appliances* yang dapat mengendalikan *electronic home appliances* dari jarak jauh yang tidak terkendala oleh jarak, tempat dan waktu sehingga pada akhirnya meningkatkan efektifitas, efisiensi dan kenyamanan dalam pengendalian.

Kata kunci : NM7010A, TCP/IP, AVR Low Cost Micro System, ElectronicHome Appliances

PENDAHULUAN

Secara umum masyarakat mengenal alat pengontrol jarak jauh berupa remote yang dapat mengendalikan suatu alat elektronik, seperti televisi, audio video, mobil dan lain sebagainya. Pengontrolan yang menggunakan remote terhambat oleh terbatasnya kemampuan dari remote tersebut dalam memancarkan sinyal yang akan ditangkap oleh penerima. Sehingga penggunaan remote kontrol terbatas oleh jarak, apabila jarak antara alat yang dikontrol dengan pengontrol itu melewati batas toleransinya, maka peralatan tersebut tidak dapat berfungsi sesuai yang diinginkan.

Internet merupakan jaringan luas yang terhubung secara global serta menggunakan TCP/IP sebagai protokol pertukaran paket (*packet switching communication protocol*) yang dapat diakses oleh siapa saja, dimana saja dan dapat dipakai untuk berbagai keperluan. Internet juga mempunyai pengaruh yang besar atas ilmu, dan pandangan dunia. Sebagai contoh hanya berpandukan mesin pencari seperti Google, pengguna di seluruh dunia mempunyai akses internet yang mudah atas bermacam-macam informasi. Dibanding dengan buku dan perpustakaan, Internet melambangkan penyebaran (*decentralization*) / pengetahuan (*knowledge*) informasi dan data secara ekstrim.

Untuk memecahkan permasalahan keterbatasan jarak kontrol dari remote, maka teknologi internet merupakan solusi untuk diimplementasikan. Aplikasi berikut akan memanfaatkan TCP/IP Starter Kit berbasis modul jaringan NM7010A-LF sebagai jembatan antara DT-AVR Low Cost Micro System dengan jaringan komputer. DT-AVR Low Cost Micro System akan berfungsi sebagai sebuah *web server* dalam pembuatan alat pengontrol jarak-jauh. TCP/IP Starter Kit yang merupakan sarana pengembangan TCP/IP berbasis modul jaringan NM7010A yang berfungsi sebagai jembatan anatara mikrokontroler dengan jaringan internet atau ethernet tanpa memerlukan bantuan komputer. TCP/IP ini cocok

untuk aplikasi-aplikasi embedded yang membutuhkan komunikasi dengan jaringan internet atau ethernet .

Hipotesanya adalah dengan menggunakan sistem pengontrolan jarak jauh melalui media internet akan menjadikan sebuah sistem pengontrolan yang tidak lagi terkendala lagi oleh jarak dan waktu. Proses pengontrolan dapat dilakukan dimana saja dan kapan saja. Sistem pengontrolan dengan memanfaatkan interface protocol *TCP/IP starter kit* dan dipadukan dengan teknologi mikrokontroler dapat menghasilkan suatu kinerja pengontrolan yang jauh lebih praktis dan efisien.

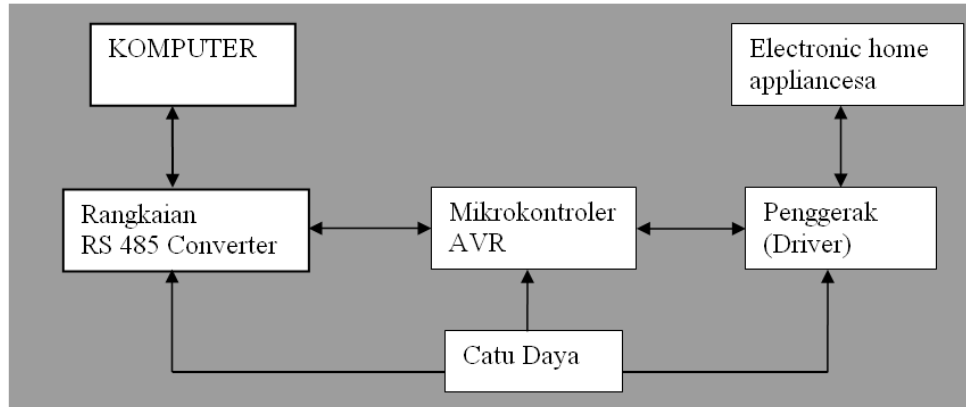
PERMASALAHAN

Keterbatasan jarak pengontrolan melalui remote karena terbatasnya daya pancar dari remote, menjadikan pengontrolan tidak dapat dilakukan dari jarak yang ekstrim lebih jauh seperti beda tempat/kota. Untuk mengendalikan peralatan elektrokronik maka antara alat yang dikontrol dan pengontrolnya harus dalam jarak jangkauan dari remote tersebut. Internet merupakan suatu solusi yang tepat karena tidak terkendala oleh jarak, di mana ada sambungan internet maka pengontrolan dapat dilakukan.

Untuk melakukan komunikasi antara peralatan yang akan dikontrol dengan pengontrolnya diperlukan suatu alat yang dapat mengendalikan komunikasi berbasis IP, dan alat tersebut adalah modul Wiznet NM7010A. Permasalahan berikutnya bagaimana modul tersebut dapat berkomunikasi dengan mikrokontroler yang merupakan alat kendali terhadap actuator (*electronic home appliances*)

PEMBAHASAN

Teknologi internet dapat dimanfaatkan untuk mengatasi jarak dalam sistem pengontrolan jarak jauh misalnya dalam hal pengontrolan peralatan listrik di sebuah industri. Tanpa dibatasi oleh ruang dan waktu, proses pengontrolan ini dapat dilakukan dari mana saja dari tempat yang memiliki akses internet. Dengan sistem pengontrolan jarak jauh melalui internet maka sistem pengontrolan tidak lagi berlingkup lokal namun berlingkup global, dimana proses pengontrolan dapat dilakukan dimana saja dan kapan saja tanpa mengharuskan petugas/operator untuk datang, namun sistem pengontrolan tersebut dapat dilakukan lewat internet untuk test dan supervisi sehingga lebih praktis dan efisien apabila memanfaatkan teknologi mikrokontroler dalam pengontrolan jarak jauh motor listrik via protokol TCP/IP Starter Kit. Diagram blok sistem pengontrolan peralatan listrik melalui internet protocol (IP) dapat dilihat di bawah ini.



Gambar 1. Diagram Blok Sistem

Penjelasan Blok Sistem

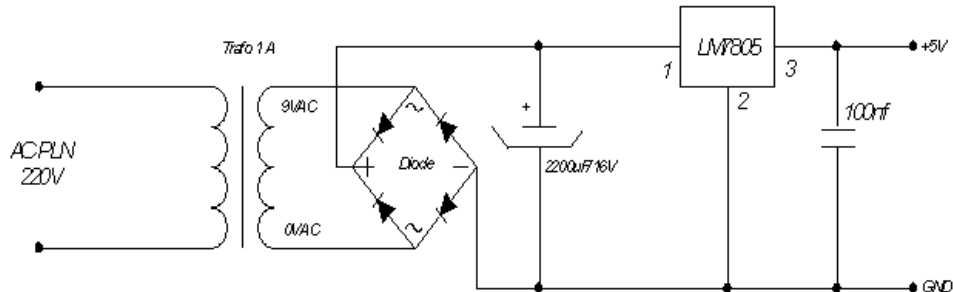
Setiap sub-sistem pada perancangan ini mempunyai fungsi dan tugas masing-masing yang saling berkaitan satu dengan lainnya, terdapat 6 blok sub-sistem yang akan dijelaskan berkaitan dengan sistem yang akan dirancang sebagai berikut :

1. Komputer
2. Catu Daya
3. Rangkaian RS 485 Converter
4. Mikrokontroler AVR
5. Penggerak (Driver)
6. Electronic home appliances

Komputer difungsikan untuk melakukan pengisian program ke mikrokontroler AVR, sedangkan rangkaian RS 485 converter sebagai perantara dari komputer ke mikrokontroler AVR untuk mengendalikan electronic home appliances. Karena program aplikasi yang dirancang berbasis windows maka kebutuhan software untuk pembuatan program adalah berikut :

1. Sistem Operasi *Windows 98, ME, XP dan 2000*
2. *Microsoft Visual basic2003*
3. *Microsoft Visio 2002*
4. *Code Vision AVR Downloader*

Setiap komponen memerlukan tegangan dan arus, baik rangkaian RS 485 converter, mikrokontroler dan driver. Oleh karena itu maka diperlukan rangkaian catu daya sebesar 5 Volt dc, agar hasil penyearahan dan filter lebih stabil maka dipergunakan IC (Integrated Circuit) regulator sebesar 5 volt yaitu IC regulator jenis LM7805. Gambar rangkaian catu daya yang digunakan dapat dilihat pada gambar berikut:



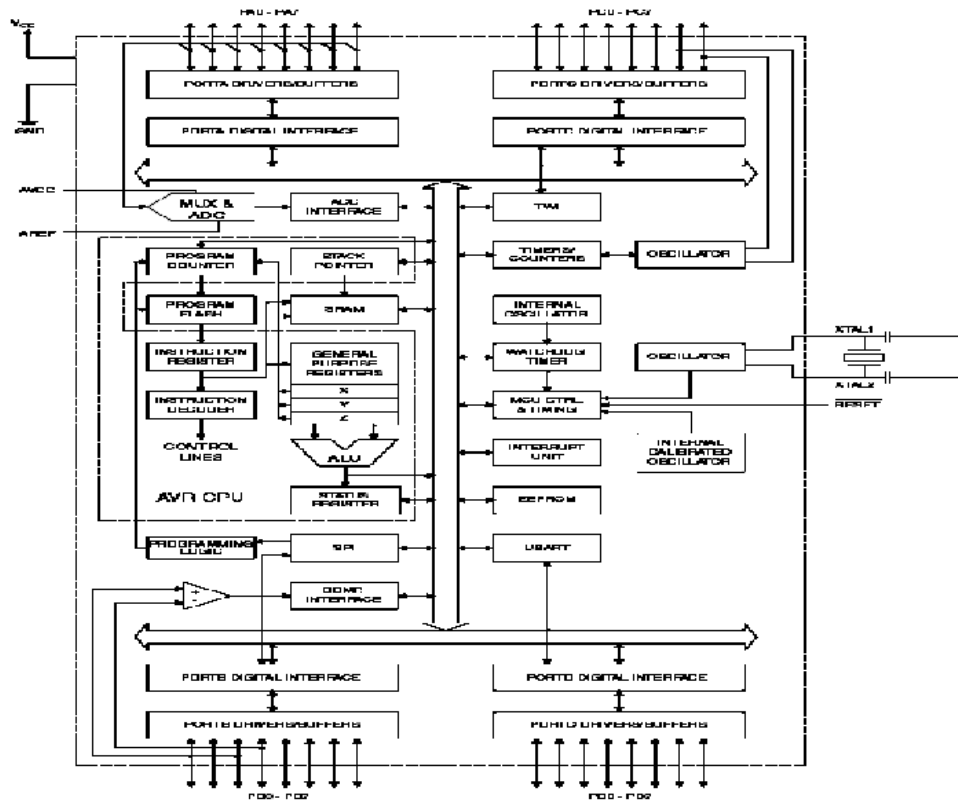
Gambar 2. Rangkaian Catu Daya 5 volt.

Rangkaian RS 485 Converter merupakan rangkaian yang digunakan sebagai interface antara mikrokontroler dengan jaringan internet atau ethernet. Rangkaian RS 485 Converter yang digunakan berbasis modul jaringan NM7010A dengan spesifikasi sebagai berikut :

- Berbasis NM7010A yang dapat menangani protokol komunikasi internal (TCP,IP,UDP,ICMP,ARP) dan ethernet (DLC,MAC).
- Menggunakan antar muka I2C untuk komunikasi dengan mikrokontroler
- Alamat I2C mini dapat dipilih dari 128 pilihan alamat yang tersedia (0, 2, 4...252, 254)
- Dilengkapi LED sebagai indikator status jaringan (collision/link, 10/100 act, full/half duplex)
- Membutuhkan catu daya 5 VDC dan telah memiliki voltage regulator 3,3 VDC/ 300Ma
- Kompatibel dengan DT-AVR *Low Cost Series* dan mendukung system kontroler lainnya.

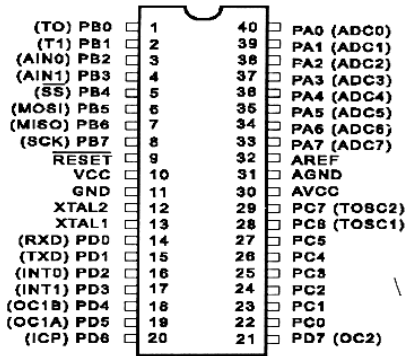
Output dari mikrokontroler mengontrol driver yang akan mengendalikan electronic home appliances, electronic home appliances ini merupakan beban akhir pengendalian. Sebelum masuk ke *actuator* maka output dari mikrokontroler perlu diperkuat terlebih dahulu melalui penggerak/driver.

Mikrokontroler AVR AT mega 8535



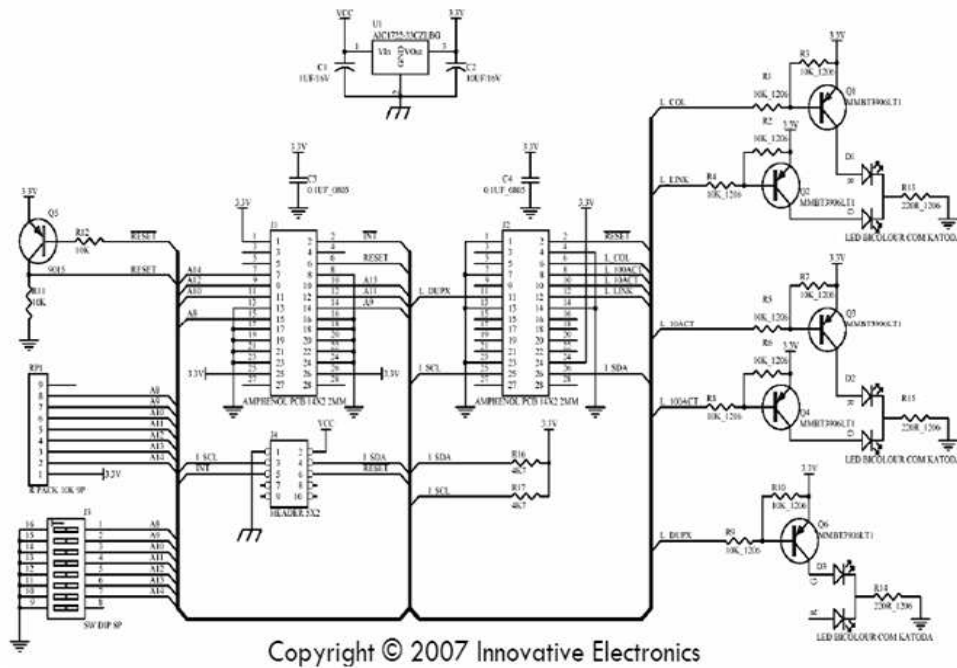
Gambar 3. Gambar diagram blok mikrokontroler AVR
(Prasimax Mikron Technology Development Center : 2007)

Mikrokontroler AVR merupakan salah satu jenis arsitektur mikrokontroler yang menjadi andalan atmel. Arsitektur ini dirancang memiliki berbagai kelebihan yang mempunyai penyempurnaan dari arsitektur mikrokontroler-mikrokontroler yang sudah ada. Salah satu kelebihan tersebut adalah kemampuan *In System Programming* sehingga chip mikrokontroler AVR langsung dapat diprogram dalam sistem rangkaian aplikasi. Selain itu AVR sudah menggunakan konsep arsitektur Harvard yang memisahkan memori dan bus untuk data dan program, serta sudah menerapkan single level pipelining, sehingga eksekusi instruksi dapat berlangsung sangat cepat dan efisien.

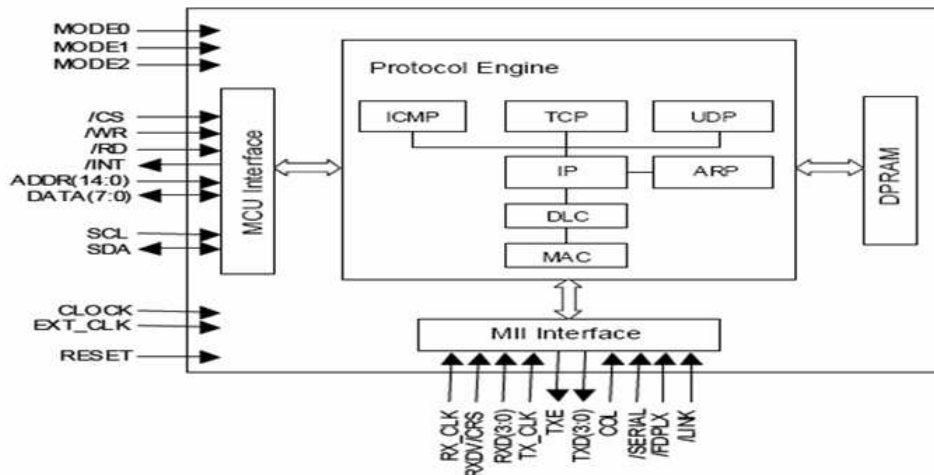


Gambar 4. Konfigurasi kaki Mikrokontroler AVR AT90S8535 PDIP 40 pin
(Prasimax Mikron Technology Development Center :2007)

Perancangan Perangkat keras
Modul jaringan Wiznet NM7010A dapat dilihat :



Gambar 5. Skema TCP/IP Starter Kit NM7010A



Gambar 6, merupakan sirkuit dari NM7010A

sedangkan blok diagramnya terdapat pada gambar 5. Perancangan ini akan memanfaatkan network modul NM7010A sebagai jembatan antara DT-AVR Low Cost Micro System dengan jaringan komputer untuk membuat sebuah web server sederhana. Programnya dikembangkan menggunakan compiler BASCOM-AVR® versi 1.11.8.1 DEMO. Pada compiler BASCOM-AVR® ini telah terdapat perintah-perintah yang mendukung antarmuka dengan modul NM7010A.

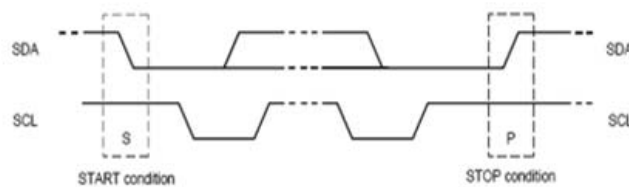
Komunikasi I2C (*Inter-Integrated Circuit*)

Untuk menghubungkan TCP/IP Starter Kit NM7010A dengan DT-AVR Low Cost Micro System digunakan I2C yang merupakan protocol dengan menggunakan dua buah kabel **SDA (Serial Data)** dan **SCL (Serial Clock)** untuk mengirimkan data secara serial. SCL merupakan jalur yang digunakan untuk mensinkronisasi transfer data pada jalur I2C, sedangkan SDA merupakan jalur untuk data. Beberapa perangkat dapat terhubung ke dalam jalur I2C yang sama dimana SCL dan SDA terhubung ke semua perangkat tersebut, hanya ada satu perangkat yang mengontrol SCL yaitu perangkat master. Jalur dari SCL dan SDA ini terhubung dengan pull-up resistor yang besar resistansinya tidak menjadi masalah (bisa 1K, 1.8K, 4.7K, 10K, 47K atau nilai diantara range tersebut).

Dengan adanya pull-up, Jalur SCL dan SDA menjadi open drain, yang maksudnya adalah perangkat hanya perlu memberikan output 0 (LOW) untuk membuat jalur menjadi LOW, dan dengan membiarkannya pull-up resistor sudah membuatnya HIGH. Dalam I2C

ada satu perangkat yang berperan menjadi master (meskipun dimungkinkan beberapa perangkat, dalam jalur I2C yang sama, menjadi master) dan satu atau beberapa perangkat slave. Dalam jalur I2C, hanya perangkat master yang dapat mengontrol jalur SCL yang berarti transfer data harus diinisialisasi terlebih dahulu oleh perangkat master melalui serangkaian pulsa clock (slave tidak bisa, tapi ada satu kasus yang disebut clock stretching). Tugas perangkat slave hanya merespon apa yang diminta master. Slave dapat memberi data ke master dan menerima data dari master setelah server melakukan inisialisasi.

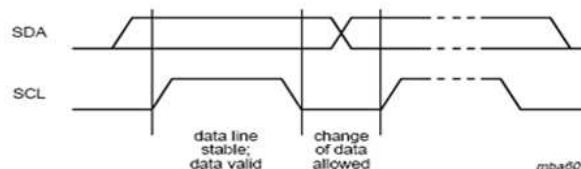
Master terlebih dahulu menginisialisasi sebelum memulai transfer data antara slavenya. Inisialisasi diawali dengan sinyal START (transisi high ke low pada jalur SDA dan kondisi high pada jalur SCL, lambang S pada gambar 2), lalu transfer data dan sinyal STOP (transisi low ke high pada jalur SDA dan kondisi high pada jalur SCL, lambang P) untuk menandakan akhir transfer data.



Gambar 7. Sinyal untuk START dan STOP

(sumber: UM10204 I2C-bus specification and user manual)

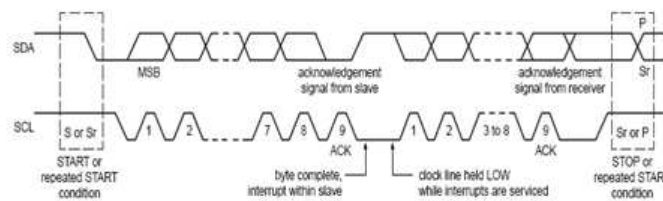
Banyaknya byte yang dapat dikirimkan dalam satu transfer data itu tidak ada aturannya. Jika transfer data yang ingin dilakukan sebesar 2 byte, maka pengiriman pertama adalah 1 byte dan setelah itu 1 byte. Setiap byte yang di transfer harus diikuti dengan bit Acknowledge (ACK) dari si penerima, menandakan data berhasil diterima. Byte yang dikirim dari pengirim diawali dari bit MSB. Saat bit dikirim, pulsa clock (SCL) di set ke HIGH lalu ke LOW. Bit yang dikirim pada jalur SDA tersebut harus stabil saat periode clock (SCL) HIGH. Kondisi HIGH atau LOW dari jalur data (SDA) hanya dapat berubah saat kondisi sinyal SCL itu LOW.



Gambar 8. Transfer bit pada jalur I2C

(sumber: UM10204 I2C-bus specification and user manual)

Setiap pulsa clock itu dihasilkan (di jalur SCL) untuk setiap bit (di jalur SDA) yang ditransfer. Jadi untuk pengiriman 8 bit akan ada 9 pulsa clock yang harus dihasilkan (1 lagi untuk bit ACK). Kronologi sebelum perangkat penerima memberikan sinyal ACK adalah sebagai berikut: saat pengirim selesai mengirimkan bit ke-8, pengirim melepaskan jalur SDA ke pull-up (ingat penjelasan open drain di atas) sehingga menjadi HIGH. Saat kondisi tersebut terjadi, penerima harus memberikan kondisi LOW ke SDA saat pulsa clock ke-9 berada dalam kondisi HIGH.



Gambar 9. Data (byte) transfer pada jalur I2C

(sumber: UM10204 I2C-bus specification and user manual)

Jika SDA tetap dalam kondisi HIGH saat pulsa clock ke-9, maka ini didefinisikan sebagai sinyal Not Acknowledge (NACK). Master dapat menghasilkan sinyal STOP untuk menyudahi transfer, atau mengulang sinyal START untuk memulai transfer data yang baru.

Dari perangkat lunak (software) untuk mengimplementasikan protokol I2C akan mengambil contoh dari rutin Peter Fleury dan dari rutin I2C CodeVision AVR (yang menggunakan bahasa C). Hal yang pertama kali terjadi dalam komunikasi ini adalah server mengirimkan sinyal START. Ini akan menginformasikan perangkat-perangkat slave yang terhubung dalam jalur I2C bahwa akan ada transfer data yang ingin dilakukan oleh master dan para slave harus siap memantau siapa yang akan dipanggil alamatnya. Selanjutnya master akan mengirimkan data berupa alamat slave yang ingin diakses. Perangkat slave yang sesuai dengan alamat yang diberikan master akan meneruskan transaksi data, slave lainnya dapat mengacuhkan transaksi tersebut dan menunggu sampai sinyal berikutnya. Setelah mendapatkan slave dengan alamat tersebut, saatnya master memberitahukan alamat internal atau nomor register yang ingin ditulis atau dibaca dari slave tersebut. Jumlah lokasi atau nomor register tersebut tergantung pada perangkat slave yang diakses. Setelah mengirim data berupa alamat slave dan kemudian data alamat internal register slave yang ingin diakses, kini saatnya master mengirim byte data. Master dapat melanjutkan mengirim byte data ke slave dan byte-byte akan ditampung di register setelahnya karena slave secara otomatis akan menaikkan alamat internal register setelah setiap byte. Ketika master selesai menulis semua data ke slave,

master akan mengirim sinyal STOP untuk mengakhiri transaksi data. Untuk implementasi kode I2C mengambil contoh rutin I2C untuk AVR oleh Peter Fleury dan dari rutin-rutin I2C yang diberikan CodeVisionAVR.

PERANCANGAN

Pada modul NM7010A, aturlah DIP Switch J3 pada TCP/IP Starter Kit untuk alamat I2C = CCH, yaitu switch 2, 3, 6, 7 posisi OFF dan switch 4, 5, 8 posisi ON. Setelah rangkaian dan catu daya terhubung dengan benar, bukalah NM7010A.BAS menggunakan BASCOM-AVR dan ubahlah baris 50 pada program tersebut agar sesuai dengan jaringan komputer yang akan digunakan. Misalnya untuk:

§ Jaringan komputer yang memiliki *gateway* dengan nilai:

Gateway = 192.168.1.2

Subnet Mask = 255.255.255.0

IP = 192.168.1.88 (nomor IP dari modul TCP/IP Starter Kit)

Maka ubahlah baris 50 tersebut menjadi:

Config Tcpi = Int0 , Mac = 12.128.12.34.56.78 , Ip = 192.168.1.88 , Submask = 255.255.255.0 , Gateway = 192.168.1.2 , Localport = 1000 , Tx = \$55 , Rx = \$55 , Twi = &HCC , Clock = 300000

§ Jaringan komputer yang tidak memiliki *gateway* dengan nilai:

Subnet Mask : 255.255.255.0

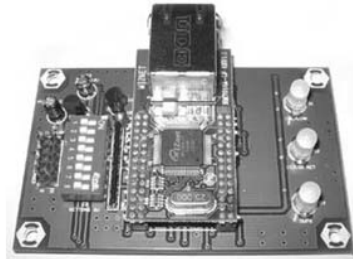
IP modul : 192.168.1.88 (nomor IP dari modul TCP/IP Starter Kit)

Maka ubahlah baris tersebut menjadi:

Config Tcpi = Int0 , Mac = 12.128.12.34.56.78 , Ip = 192.168.1.88 , Submask = 255.255.255.0 , Gateway = 0.0.0.0 , Localport = 1000 , Tx = \$55 , Rx = \$55 , Twi = &HCC , Clock = 300000

Setelah itu *compile* ulang program NM7010A.bas tersebut dan *download*-kan ke dalam DT-AVR Low Cost Micro System menggunakan *DT-HiQ AVR In System Programmer* atau divais *programmer* lain yang mendukung mikrokontroler ATmega8535. Setelah itu hubungkan sistem ke jaringan komputer dan jalankan program Microsoft Internet Explorer dari komputer yang terhubung ke jaringan komputer yang sama. Ketikkan `http://< nomor IP >/index.htm` (misalnya `http://192.168.1.88/index.htm`) pada kolom "Address" Microsoft® Internet Explorer© maka akan tampil halaman situs dari *embedded web server* ini.

[* File contains invalid data | In-line.JPG *]

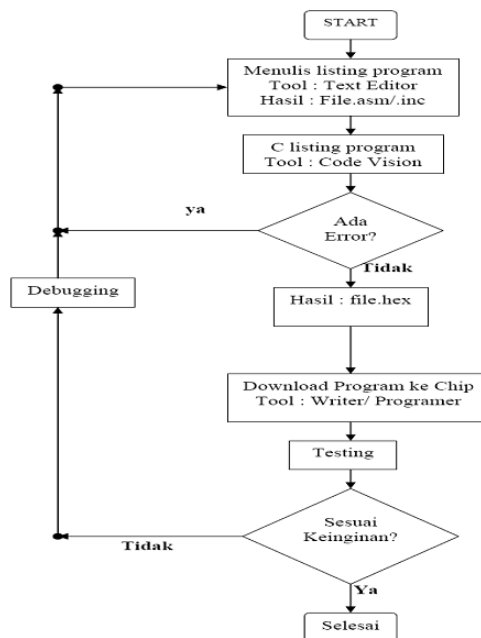


Gambar 10. Modul NM7010A-LF

Perancangan Perangkat Lunak

Mikrokontroler merupakan komponen elektronika yang kinerjanya tergantung pada program yang dimasukkan dan telah bekerja didalamnya. Sebelum mikrokontroler digunakan dalam sistem rangkaian elektronika, harus terlebih dahulu diisi program yang telah dibuat oleh pembuat program. *Software* yang digunakan untuk menuliskan listing program bahasa *assembly* adalah BASCOM-AVR.

Adapun flowchart program ke target blok adalah sebagai berikut :



Gambar 11. Flowchart Program ke Target Blok

Code Vision Atmega 8535

CodeVisionAVR merupakan software C-cross compiler, di mana program dapat ditulis menggunakan bahasa-C. Dengan menggunakan pemrograman bahasa-C diharapkan waktu disain (developing time) akan menjadi lebih singkat. Setelah program dalam bahasa-C ditulis dan dilakukan kompilasi tidak terdapat kesalahan (error) maka proses download dapat dilakukan. Mikrokontroler AVR mendukung sistem download secara ISP (*In-System Programing*).

Proses program

Proses dari program NM7010A.BAS secara garis besar adalah sebagai berikut:

1. Program akan me-*reset* modul NM7010A secara *hardware*, mengaktifkan fungsi interupsi mikrokontroler dan melakukan inisialisasi modul NM7010A pada mode komunikasi I2C.
2. Kemudian program melakukan deklarasi variabel yang akan digunakan, antara lain:
 - § Shtml sebagai *string* dengan panjang 15 karakter untuk menyimpan *suffix* dari perintah yang diterima.
 - § Ihitcounter sebagai *integer* yang berfungsi untuk menyimpan jumlah kunjungan ke webserver ini.
3. Program mengambil status dari *socket 0*.
4. Bila status *socket 0* = *established* (06h) maka:
 - a) Program akan memeriksa *buffer Rx* dari modul NM7010A, dan jika ada data yang diterima dalam *buffer Rx* maka program akan membacanya.
 - b) Bila data yang diterima adalah perintah "GET" maka program akan menyimpan *suffix* yang mengikuti perintah tersebut ke dalam variabel Shtml.
 - c) Program memeriksa apakah *buffer Rx* sudah kosong, bila belum kosong maka program akan kembali ke langkah 4. a.
 - d) Jika *buffer Rx* sudah kosong maka program mengirimkan "HTTP/1.0 200 OK< CR> < LF>" (tanda OK) dan mengirimkan "Content-Type: text html< CR> < LF>" (format *body* html yang akan dikirimkan).
 - e) Jika Shtml = "/index.htm" maka program akan mengirimkan badan index.htm dan menambah nilai variabel Ihitcounter dengan 1. Index.htm yang dikirimkan untuk nilai Ihitcounter = 1.
 - f) Program menghapus isi variabel Shtml, lalu menutup *socket 0* dan kembali ke langkah 3.
5. Bila status *socket 0* = *wait connection close* (07h) maka program akan menutup *socket 0* dan kembali ke langkah 3.
6. Bila status *socket 0* = *connection closed* (00h) maka program membuka *port 80h*

socket 0 dan mulai mendengarkan jaringan dari *socket* 0, lalu program kembali ke langkah 3.



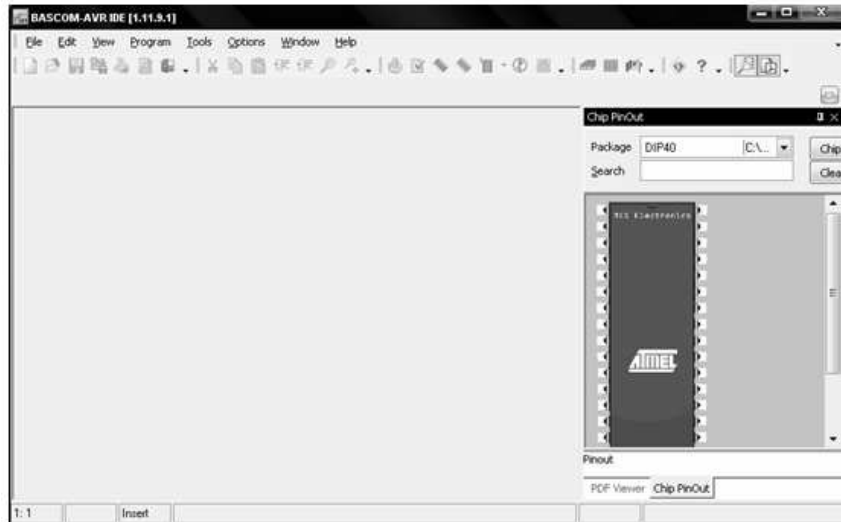
Gambar 12. Tampilan Halaman Web pada Microsoft Internet Explorer

Halaman web dari aplikasi ini terdiri dari *header*, teks, dan *counter* pengunjung seperti terlihat pada Gambar 3. Aplikasi ini dapat dikembangkan menjadi lebih kompleks misalnya untuk pengirim data sensor dan pengontrol peralatan melalui jaringan komputer.

Penulisan Listing Program Assembly pada Mikrokontroler

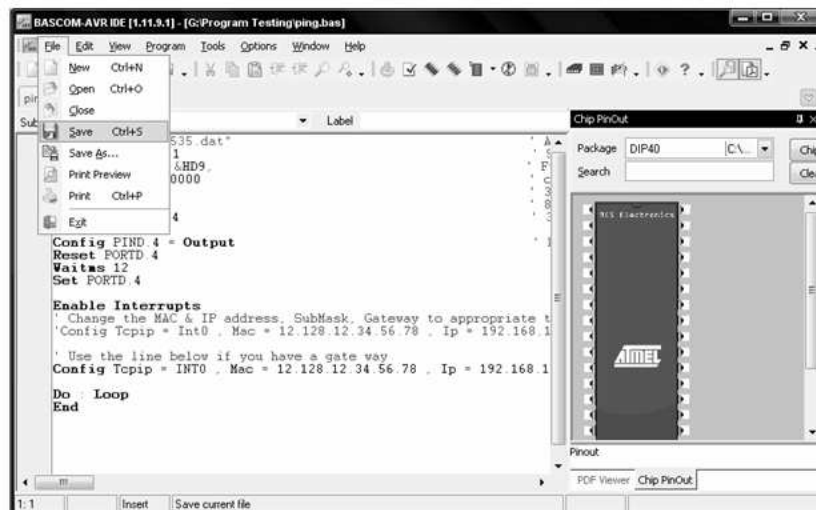
Mikrokontroler adalah salah satu jenis komponen elektronika yang kinerjanya tergantung pada program *assembly* yang dimasukkan dan telah bekerja didalamnya. Jadi agar mikrokontroler tersebut bekerja untuk mendukung sistem peralatan seperti yang diinginkan, maka harus terlebih dahulu diisi program *assembly* yang benar, baik dari segi bahasa program maupun cara pengisiannya.

Sebelum mikrokontroler digunakan dalam sistem rangkaian elektronika, harus terlebih dahulu diisi program yang telah dibuat oleh pembuat program. Hal ini bertujuan agar IC tersebut dapat bekerja sesuai dengan keinginan. *Software* yang digunakan untuk menuliskan listing program bahasa *assembly* adalah BASCOM-AVR, alasan menggunakan software ini dikarenakan mempunyai beberapa kelebihan dibanding software yang lainnya.



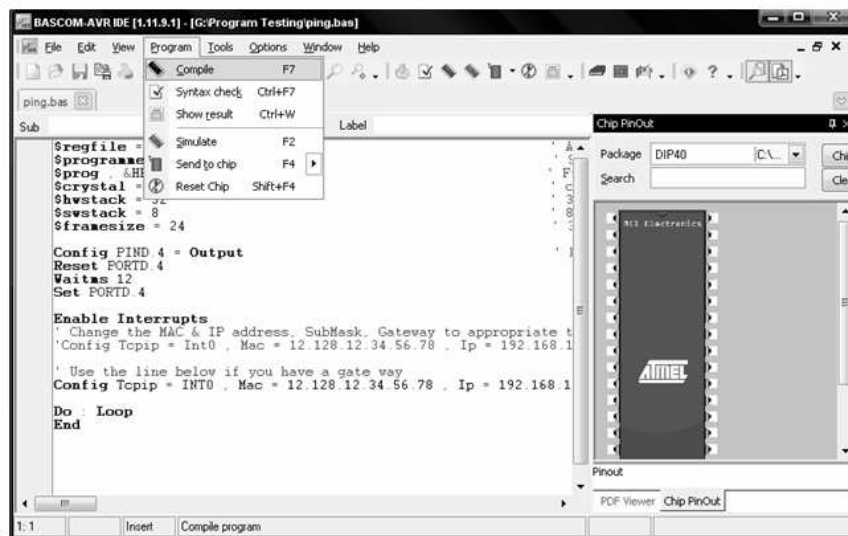
Gambar 13. Tampilan BASCOM-AVR

Setelah penulisan listing program pada BASCOM-AVR text editor selesai, kemudian *text* tersebut disimpan ke dalam file dengan nama MOTOR DC.BAS ,hal ini harus dilakukan karena *software* hanya bekerja pada file dengan nama *.BAS.*



Gambar 14. Penyimpanan File Basic

Langkah berikutnya adalah dengan mengkompilasi file Basic tersebut menjadi file heksa, sehingga file MOTOR DC.BAS akan menjadi file MOTOR DC.HEX yaitu dengan cara menekan tombol F7 pada keyboard atau melalui menu. file *.HEX inilah yang akan dimasukkan kedalam IC Atmega8535. Adapun langkah diatas dapat dilihat pada gambar berikut.

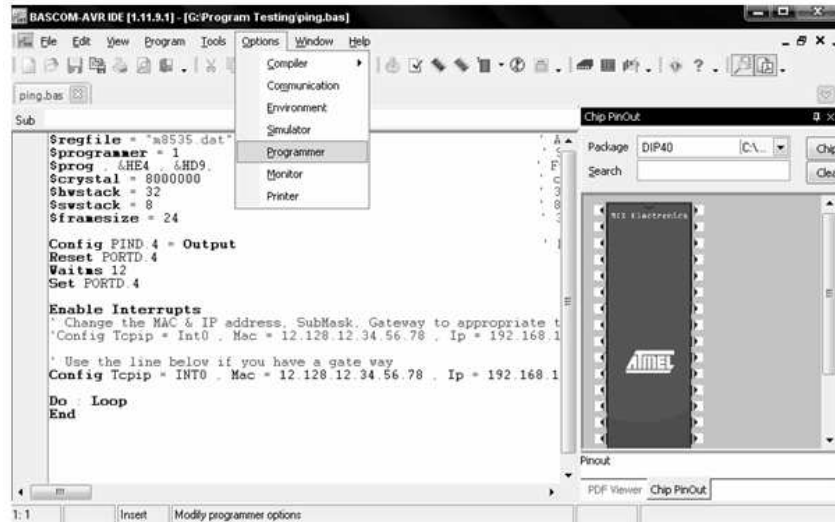


Gambar 15. Kompilasi Program basic

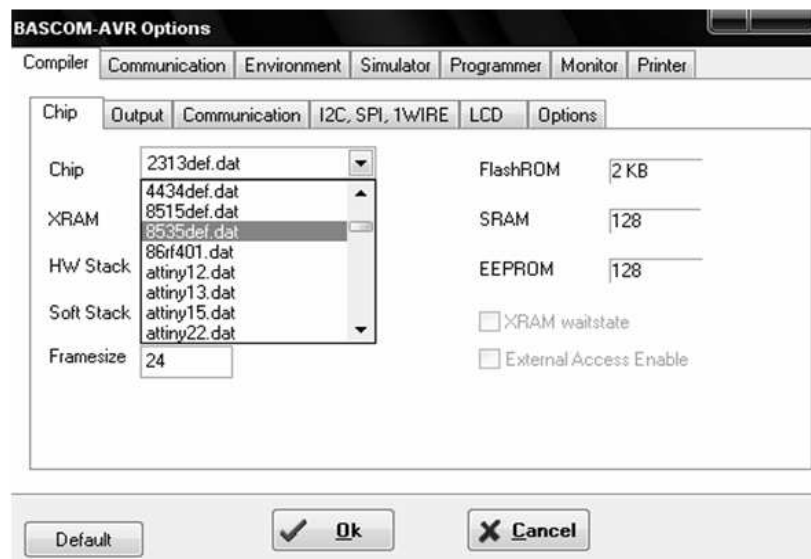
Setelah langkah – langkah tersebut dilakukan maka kita akan mempunyai beberapa file setelah langkah kompilasi, yaitu : MOTOR DC.BAS , MOTOR DC.HEX dan beberapa file pendukung lainnya. dan sampai pada tahapan ini maka proses penulisan dan kompilasi program assembly sudah selesai.

Memasukkan Program Kedalam Mikrokontroler

Pada langkah ini, IC Atmega8535 yang awalnya kosong mulai diisi dengan program. Sedangkan untuk IC yang sebelumnya telah berisi program lain, maka program tersebut dihapus terlebih dahulu secara otomatis sebelum diisi dengan program yang baru. Untuk memulainya, terlebih dahulu membuka program BASCOM-AVR yang dibuat oleh produsen mikrokontroler AT8535 tersebut yaitu ATMEL company. Kemudian memilih *device* yang akan digunakan yaitu AT8535.

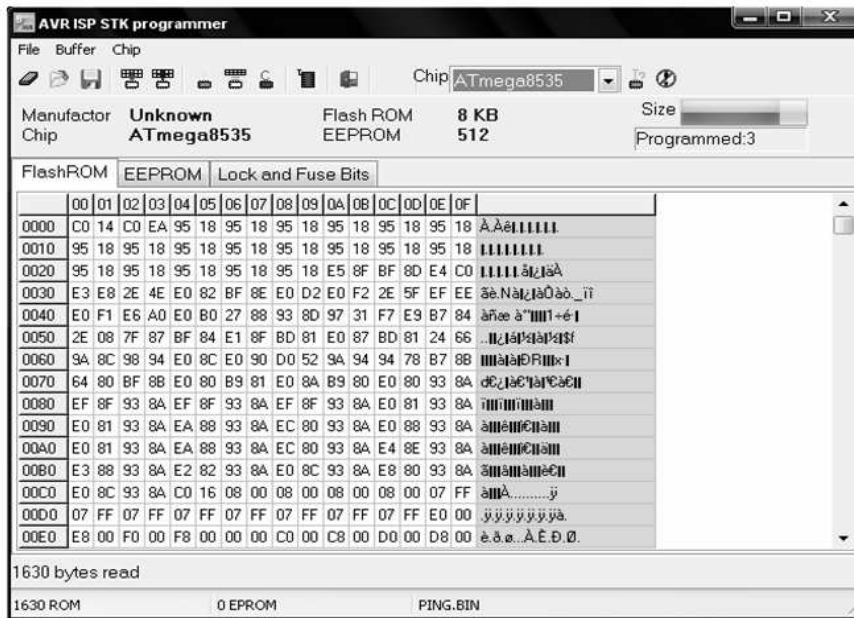


Gambar 16. Pemilihan Device yang Digunakan



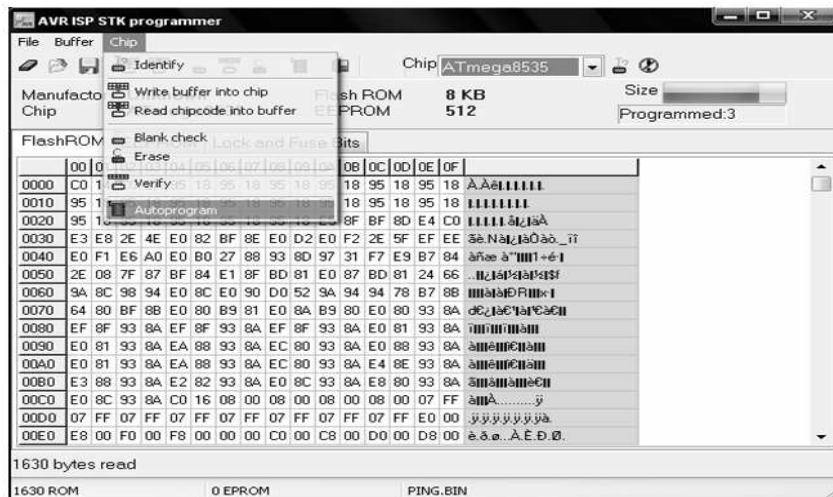
Gambar 17. Device yang Digunakan AT8535

Setelah memilih device yang digunakan, pada menu "Compiler". *Software* kemudian meminta untuk memasukan file .HEX yang akan dimasukan kedalam IC mikrokontroler, dalam hal ini adalah MOTOR DC.HEX.



Gambar 18. Memasukkan File MOTOR DC.HEX

File .HEX yang telah masuk akan dikenali oleh *software* tersebut kemudian dimasukan kedalam IC mikrokontroler. kemudian pilih menu *chip* dan cari menu *Auto Program* seperti pada gambar 3.13.



Gambar 19. Proses Pengisian IC Mikrokontroler

IC Mikrokontroler terisi seiring dengan bertambahnya presentase yang muncul pada jendela *software* setiap proses tersebut berlangsung. Proses pengisian berlangsung diawali dengan "*Erase Flash & EEPROM Memory*", yang berarti *software* melakukan penghapusan terhadap memori internal IC Mikrokontroler terlebih dahulu sebelum mengisikan program ke dalam IC tersebut. Pada proses penghapusan ini, apabila presentase telah mencapai 100% maka berarti memori internal telah sepenuhnya terhapus dan dalam keadaan kosong. Jika presentase belum mencapai 100% tetapi *software* menunjukkan tanda error, maka proses penghapusan gagal. Hal ini biasanya disebabkan oleh adanya kesalahan pada *hardware* downloadernya.

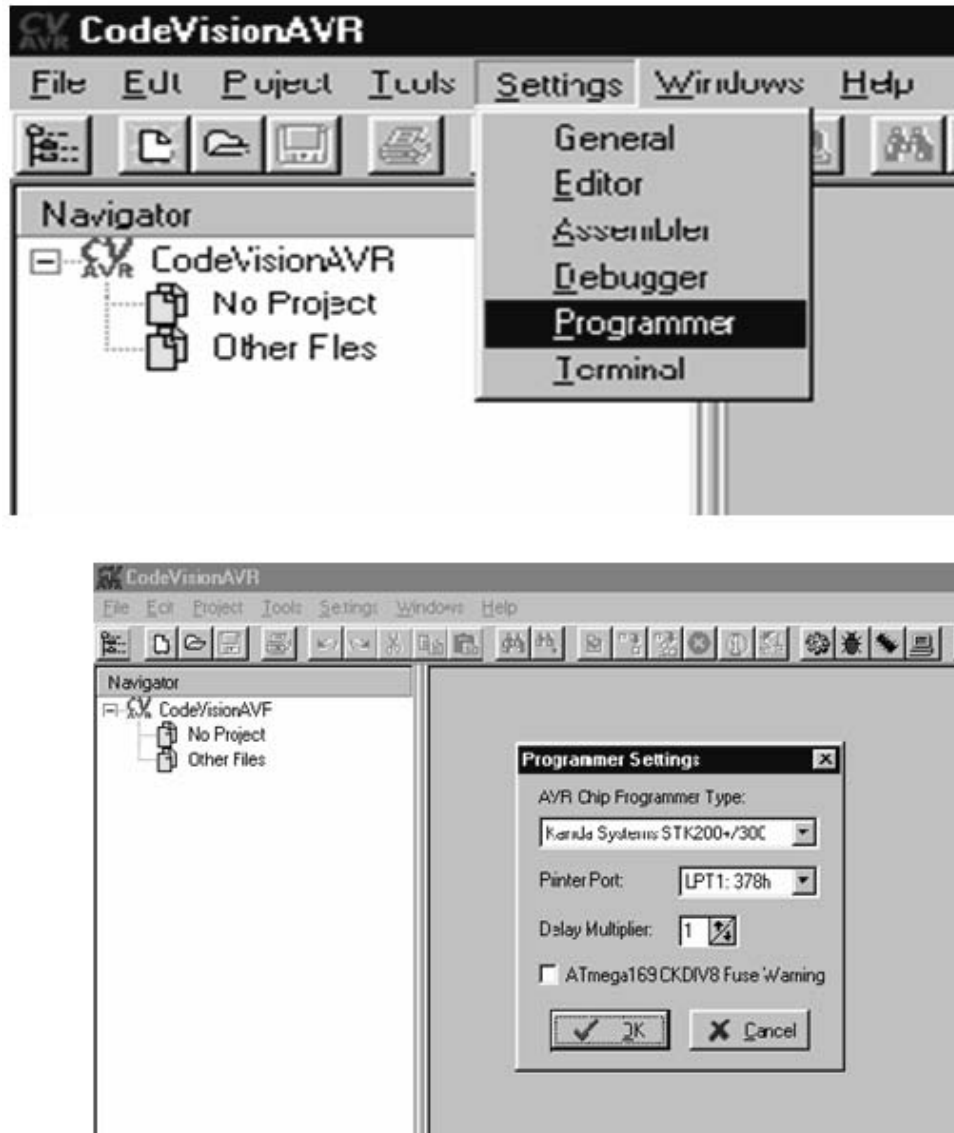
Setelah proses penghapusan selesai maka secara otomatis *software* melakukan "*Verify Flash Memory*". *Software* mulai mengisi IC Mikrokontroler dengan file. HEX. Sama dengan penghapusan, yaitu prosesnya ditunjukkan dengan penambahan presentase pengisian. 100% menunjukkan bahwa IC Mikrokontroler telah sepenuhnya terisi. Munculnya tanda error menunjukkan proses pengisian gagal, yang biasanya disebabkan oleh kesalahan pada *hardware* downloadernya. Jika langkah - langkah di atas berjalan dan selesai, maka IC Mikrokontroler yang dalam rancangan alat ini memakai jenis AT8535, sudah bisa digunakan untuk menjalankan sistem kerja alat rancangan.

Pengisian Program mikrokontroler AVR dengan Starter kit

Software Basic Compiler merupakan C Compiler untuk mikrokontroler AVR. Pada CodeVision telah disediakan editor yang berfungsi untuk membuat program dalam bahasa C, setelah melakukan proses kompilasi kita dapat mengisikan program yang telah dibuat ke dalam memori pada mikrokontroler menggunakan program yang telah disediakan oleh CodeVision AVR.

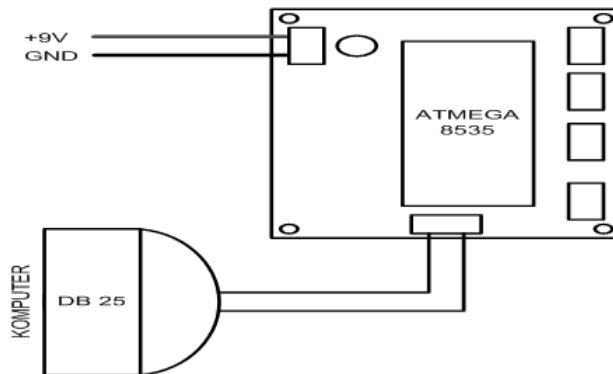
Program yang didukung oleh CodeVision banyak sekali variasinya, antara lain: *Kanda Sistem STK200+ /300, Atmel STK500/AVRISP, Dontronics DT006*, dan lain-lain. Agar *de KITS AVRISP Programmer Cable* dapat diintegrasikan dengan CodeVision AVR, terlebih dahulu harus dilakukan konfigurasi sebagai berikut :

- Jalankan Software CodeVision AVR.
- Pilih menu Setting '!' Programmer.
- Pilih tipe programmer **Kanda Sistem STK200+ / 300.**
- Lalu klik tombol OK



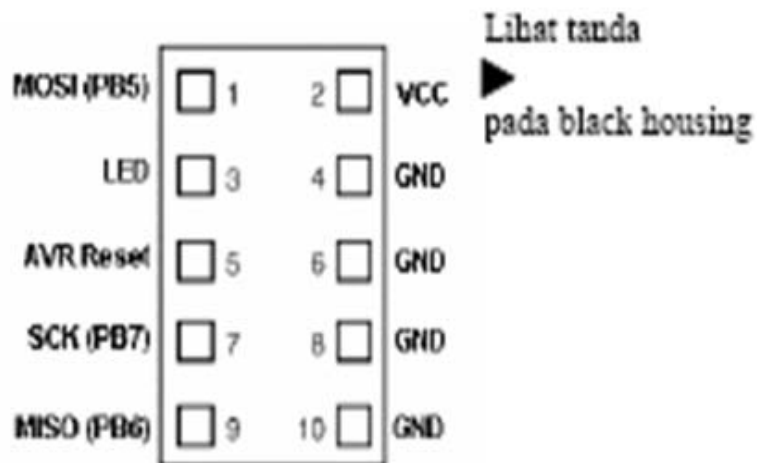
Gambar 20. Tampilan Pilihan Programmer

Setelah CodeVision dikonfigurasi, uji *de KITS AVR ISP Programmer Cable* dengan cara menghubungkannya dengan target board dan ke PC melalui port LPT seperti gambar berikut ini:



Gambar21. Koneksi Atmega 8535 Programmer Cable

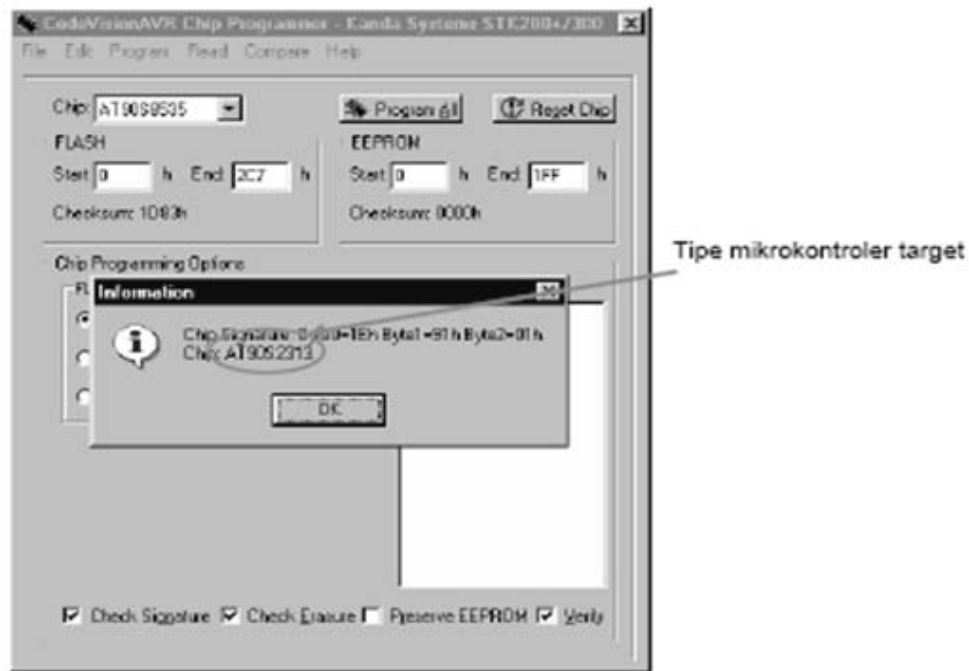
Adapun koneksi *black housing* ke header ISP pada target board disesuaikan dengan tata letak pin yang bersangkutan. Tata letak pin pada *black housing de KITS AVR ISP Programmer Cable* terdapat pada gambar 3.4. Karena *black housing* mempunyai bentuk simetris, maka satu-satunya tanda yang menjadi pedoman pemasangan adalah tanda segitiga pada salah satu sisi *black housing* di mana pin yang dekat dengan tanda tersebut adalah pin 2 yaitu VCC.



Gambar 22. Konfigurasi Pin AT 89S51

Untuk melakukan pengujian terhadap de KITS AVR ISP *Programmer Cable*, awali proyek baru dengan cara sebagai berikut:

- Pasang AVR ISP Programmer Cable pada target board yang telah berisi mikrokontroler target.
- Pilih menu Tools 'I Chip Programmer atau tekan Shift+ F4.
- Pada jendela Chip programmer pilih menu Read 'I Chip Signature.
- Apabila AVR ISP Programmer cable bekerja baik dan ID mikrokontroler tidak rusak, maka tipe mikrokontroler target akan tampak seperti gambar berikut.

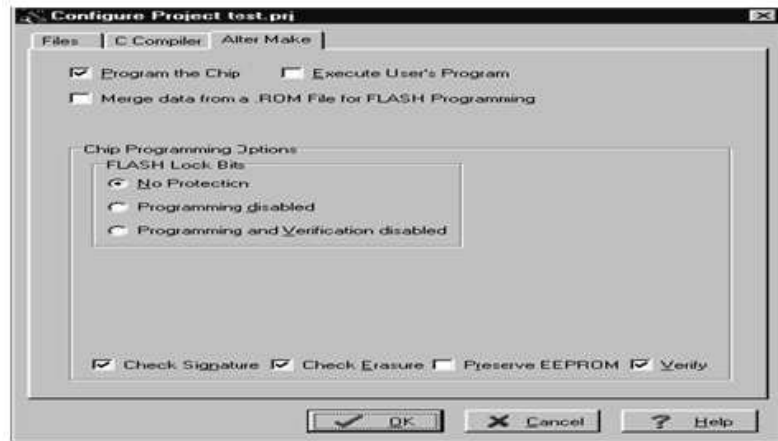


Gambar 23. Uji de KITS AVR ISP Programmer Cable dengan Read Chip Signature
(Sumber: www.innovativeelectronics.com)

Apabila ID dari mikrokontroler rusak maka tipe yang muncul adalah “*Chip: Unknown*” Setelah melakukan pemeriksaan pada de KITS AVR ISP Programmer cable, hal selanjutnya adalah membuat proyek. Buka file test.prj menggunakan CodeVision AVR.

Untuk memudahkan pemrograman, CodeVision harus dikonfigurasi terlebih dahulu. Caranya sebagai berikut:

- Pilih menu Project 'I Configure.
- Pilih tab *After Make*, kemudian centang pada kotak dengan label *Program the Chip*. Penjelasan lebih jelasnya terdapat gambar berikut ini.



Gambar 24. Pengaturan untuk Memudahkan Pemrograman
(Sumber: www.innovativeelectronics.com)

- Klik pada tombol OK.
Proses ini hanya dapat dilakukan pada saat ada project yang telah dibuat atau dibuka. Tekan Shift+F9, download ke target board dengan cara klik pada tombol program. Akhirnya seluruh rangkaian instalasi dan pengujian telah selesai, mikrokontroler sudah siap untuk dipergunakan.

KESIMPULAN

Berdasarkan hasil rancang bangun dan pengujian terhadap modul kontrol dapat disimpulkan :

- a) Electronic home appliances yang digunakan pada uji coba pengontrolan melalui media nternet dengan modul jaringan NM7010A berupa dc motor. Dengan menekan tombol F5 pada keyboard laptop/komputer yang terhubung dengan jaringan internet dapat dilakukan pengontrolan jarak jauh untuk menentukan arah putaran dc motor yaitu putaran ke kanan, putaran ke kiri dan berhenti. Banyaknya proses putaran motor dc ke kiri, ke kanan dan berhenti tercatat dan tampil pada display monitor.
- b) Memanfaatkan TCP/IP Starter Kit berbasis modul jaringan NM7010A sebagai jembatan antara DT-AVR Low Cost Micro System dengan jaringan internet dapat melakukan pengontrolan electronic home appliances yang tidak terkendala lagi oleh jarak bahkan waktu maupun tempat.

DAFTAR PUSTAKA

1. Asep Saefullah, Bramantyo Yudi W (2008). *Perancangan Sistem Timer Lampu Lalu Lintas Dengan Mikrokontroler AVR*, Journal CCIT, Vol.2 No.1, STMIK Raharja
2. Paulus Andi Nalwan (2003). *Panduan Praktis Tehnik Antar Muka dan Pemrograman Mikrokontroler AT89C51*, Gramedia, Jakarta
3. Untung Rahardja, Asep Saefullah (2009), *Simulasi Kecepatan Mobil Secara Otomatis*, Journal CCIT, Vol.2 No.2, STMIK Raharja
4. Widodo Budiharto (2005). *Perancangan Sistem Dan Aplikasi Mikrokontroler*, PT. Elex Media Komputindo, Jakarta
5. Widodo Budiharto, Sigit Firmansyah (2005). *Elektronika Digital Dan Mikroprosesor*, Penerbit Andi, Yogyakarta