

BAB V

PENGUJIAN SISTEM DAN IMPLEMENTASI

5.1 HASIL IMPLEMENTASI

Pada tahap ini penulis mengimplementasikan hasil rancangan yang telah dibuat. Adapun hasil implementasi dari penelitian ini adalah sebagai berikut :



Gambar 5.1 *Prototype* Alat Pengontrol Suhu Panas Air

Gambar di atas merupakan *prototype* perancangan alat pengontrol suhu panas air yang telah di rancang penulis. Yang mana pada alat pengontrol suhu panas air tersebut terdapat terdapat sensor HCSR-04 dan sensor ds18b20, *waterheater*, LCD dan rangkaian keseluruhan dari alat pengontrol suhu panas air tersebut. Digunakannya LCD di sini adalah untuk menampilkan berapa suhu air

yang ingin di panaskan serta terdapat selisih pemanas air yang di inginkan dengan berapa suhu air yang sudah tersedia.

5.2 PENGUJIAN SISTEM

Sistem dirancang agar alat saling *terintegrasi*, artinya karena sistem sudah terdiri dari beberapa bagian yang saling mendukung menjadikan sistem dapat berdiri dan bekerja sesuai perencanaan dan rancangan pembuatan. Sehingga sistem dapat bekerja dengan baik, tentu tidak lepas dari beberapa masalah yang dilalui dalam perancangan dan pembuatan alat ini. Masih banyak hal-hal baru yang akan kita temui hingga akhirnya akan semakin meminimalkan kekurangan sistem untuk hal ini dilakukan beberapa langkah konkrit untuk tujuan pengujian sistem, yan akhirnya diharapkan untuk mendapatkan sistem yang lebih sempurna.

Pengujian alat ini menggunakan *remote control* berbasis mikrokontroller atmega 16 ini memiliki beberapa tahap, tahapan ini bertujuan untuk memperkecil kemungkinan alat tidak bekerja saat dilakukan uji coba atau perbedaan hasil yang diinginkan.

5.3 CARA MENGOPERASIKAN ALAT

Untuk alat yang dirancang dan telah direalisasikan pada tugas akhir ini akan dilakukan pengujian dengan cara :

1. Pengujian dengan perangkat lunak (*software*)
2. Pengujian dengan perangkat keras (*hardware*)

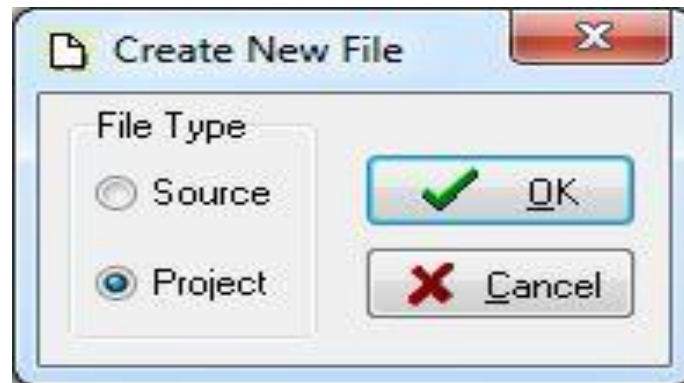
5.3.1 Pengujian Perangkat Lunak

Hal pertama dilakukan dalam pengujian perangkat lunak adalah menentukan aplikasi (*software*) yang akan digunakan untuk mengisi program pada mikrokontroler ATmega 16.

1. CodeVisionAVR

Untuk mikrokontroler ATmega 16 sinkron dengan banyak bahasa pemrograman seperti bahasa pemrograman *Assembler*, CodeVision AVR, bahasa C, Bascom AVR dan lainnya. Dalam pembuatan alat ini penulis menggunakan CodeVision AVR, sebab CodeVision AVR sangat kompetibel dengan *downloader* yang penulis gunakan. CodeVisionAVR merupakan sebuah *cross-compiler C*, *Integrated Development Environment (IDE)*, dan *Automatic Program Generator* yang didesain untuk mikrokontroler buatan Atmel seri AVR. CodeVisionAVR dapat dijalankan pada sistem operasi Windows 98, Me, dan Windows XP. Namun penulis menggunakan sistem operasi Windows 7. Cross-compiler C mampu menerjemahkan hampir semua perintah dari bahasa ANSI C, sejauh yang diijinkan oleh arsitektur dari AVR, dengan tambahan beberapa fitur untuk mengambil kelebihan khusus dari arsitektur AVR dan kebutuhan pada sistem *embedded*.

Untuk pengujian yaitu pembuatan *listing program* baru, tekan file kemudian pilih *New Project*, lalu *Checklist Project* dan OK. Untuk lebih jelasnya lihat pada gambar 5.2 di bawah ini :



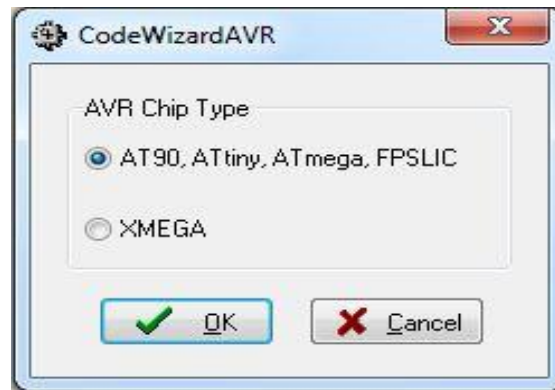
Gambar 5.2 Menu Membuat Program Baru

Maka selanjutnya akan muncul kembali jendela konfirmasi *Project*, pilih YES :



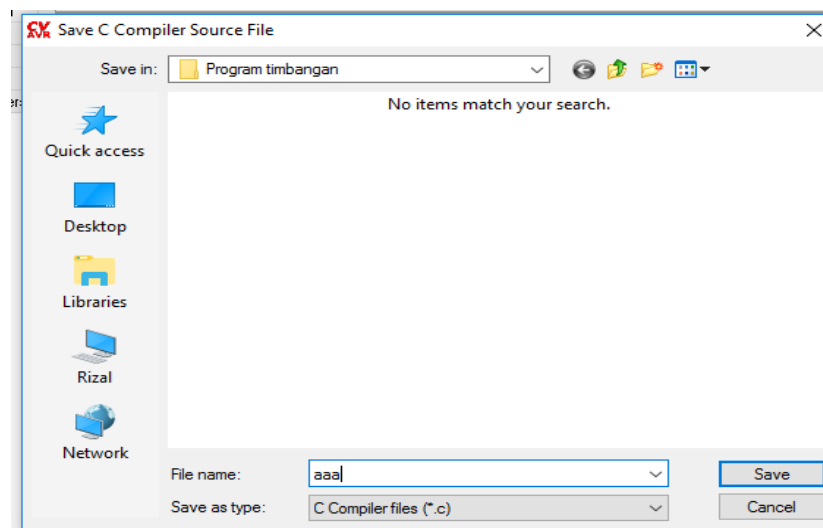
Gambar 5.3 Konfirmasi *Project*

Kemudian akan muncul kembali jendela *Code Wizard AVR*, pada bagian ini tentukan *Chip* yang akan digunakan. karena *Chip* yang akan digunakan Atmega16 maka *checklist* pilihan yang pertama yaitu *AT90*, *Attiny*, *Atmega*, *FPSLIC* lalu klik OK.



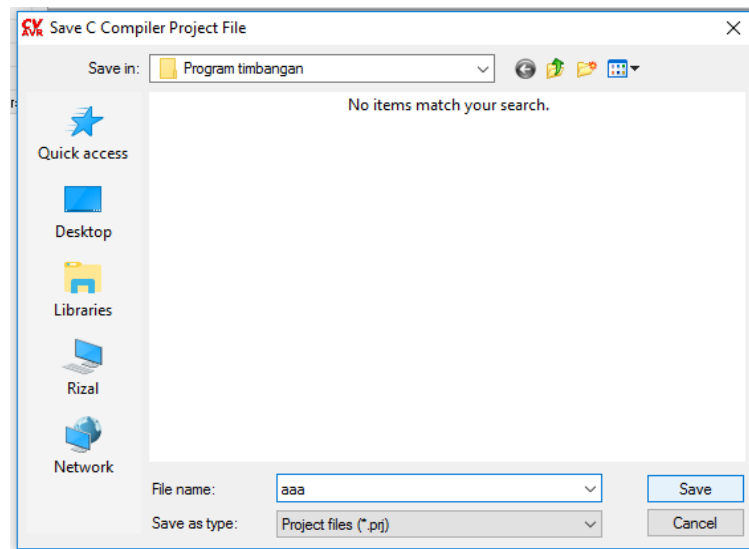
Gambar 5.4 Code Wizard AVR

Berikutnya muncul jendela *Save C Compiler Source File* yaitu jendela untuk menyimpan *file*, pilih dimana *file project* akan kita simpan. Menyimpan *file* pada *Codevision AVR* terdiri dari tiga kali penyimpanan yaitu : *Save* yang pertama berupa *file* ekstensi *.C*, yang kedua *Save file* ekstensi *.prj*, dan yang ketiga *Save File* ekstensi *.cwp*, untuk lebih jelasnya bisa dilihat pada gambar 5.10, 5.11, dan 5.12 berikut :



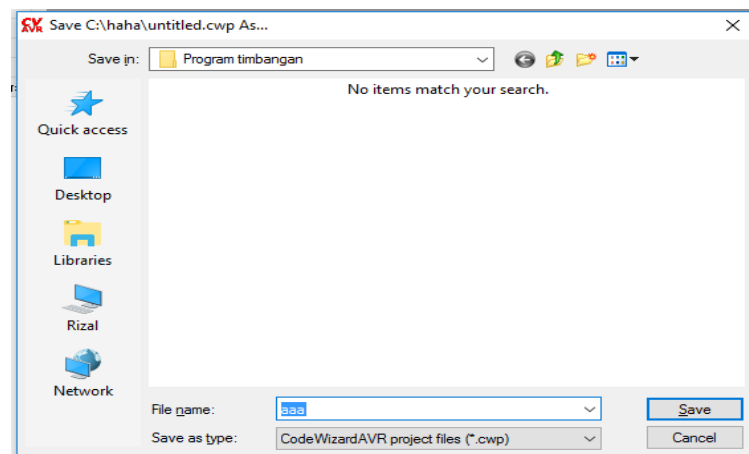
Gambar 5.5 Save Pertama file ekstensi .C

File ekstensi *.c* menyimpan data agar *Codevision AVR* diprogram menggunakan bahasa C.



Gambar 5.6 Save Kedua File Ekstensi .prj

File ekstensi.*prj* digunakan untuk menyimpan data proyek dan pengaturan dan dapat mencakup rujukan pada berkas lainnya yang digunakan oleh proyek.



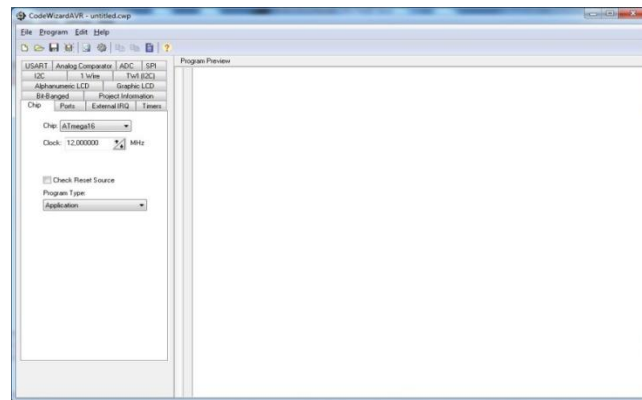
Gambar 5.7 Save Ke Tiga File Ekstensi .cwp

File ekstensi.*cwp* dipilih untuk mempermudah penulisan *source code*.

Setelah tahapan *Save File*, yang akan dilakukan selanjutnya adalah masuk ke dalam jendela *Code Wizard AVR*. Di jendela ini terdiri dari beberapa *Tab* pilihan yaitu terdiri dari :

a. Tab *Chip*

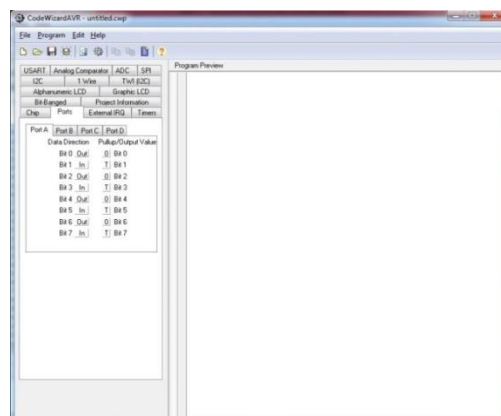
Yaitu *Tab* yang menentukan pilihan *Chip* yang digunakan Atmega16, *Clock* yang digunakan 12.000.000 *MHz*



Gambar 5.8 Tab *Chip*

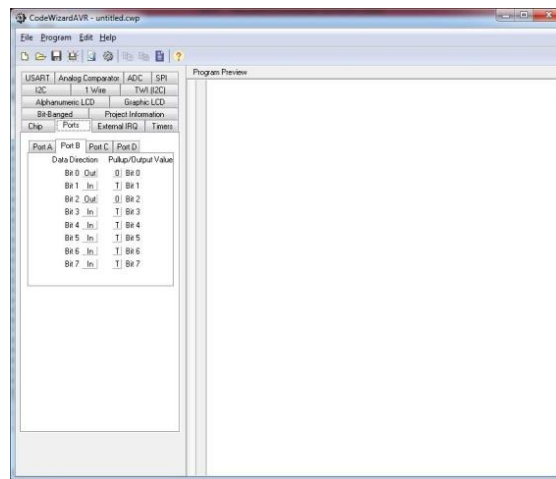
b. Tab *Ports*

Port A : Pada Tab *Ports* akan dipilih I/O Port mana saja yang akan digunakan yaitu Port A : Pin A.0, Pin A.1, Pin A.2, dan Pin A.3 *pulup/output value* dirubah menjadi out.



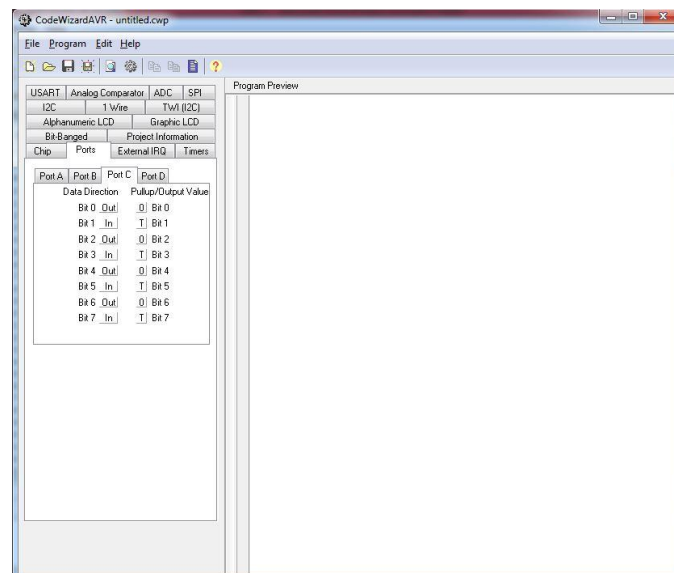
Gambar 5.9 Tab *Ports A*

Port B : Pada Tab *Ports* akan dipilih I/O Port mana saja yang akan digunakan yaitu Port B : Pin B.0, Pin B.1, Pin B.2, dan Pin B.3 *pulup/output value* dirubah menjadi out. Port B adalah letak dari LCD 16x2.



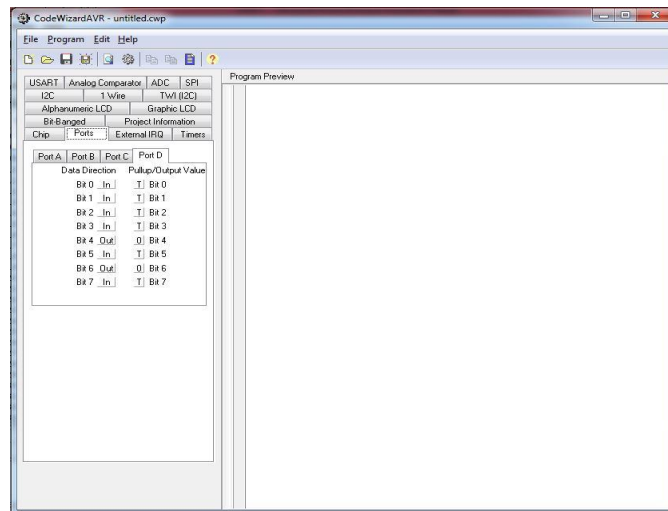
Gambar 5.10 Tab Ports B

Port C : Pada tab port C ini adalah sebagai port untuk sensor ds18b20, PORT yang digunakan adalah PORTC.2, PORTC.3, PORTC.4, PORTC.5.



Gambar 5.11 Tab Ports C

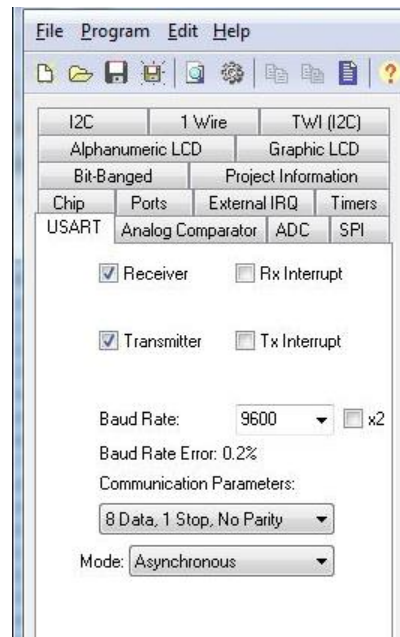
Port D : Pada Tab PORTD disini digunakan untuk tombol OK, dan tombol menaikkan suhu dan tombol menurunkan suhu, PORT yang digunakan adalah, PORTD.0, PORTD.1 dan PORTD.2 .



Gambar 5.12 Menentukan Pin Untuk Tombol

3. Tab USART

USART (*Universal Synchronous and Asynchronous Serial Receiver and Transmitter*) merupakan komunikasi yang memiliki fleksibilitas tinggi, yang dapat digunakan untuk melakukan transfer data baik antar mikrokontroler maupun dengan modul-modul eksternal termasuk PC yang memiliki fitur UART. Tab USART digunakan untuk mengaktifkan *transmitter*, *receiver*, *baud rate*, dan *communication parameters*.



Gambar 5.13 Tab USART

Transmitter data (TxD) adalah sinyal *actual* yang dikirimkan dari satu perangkat ke perangkat lain.

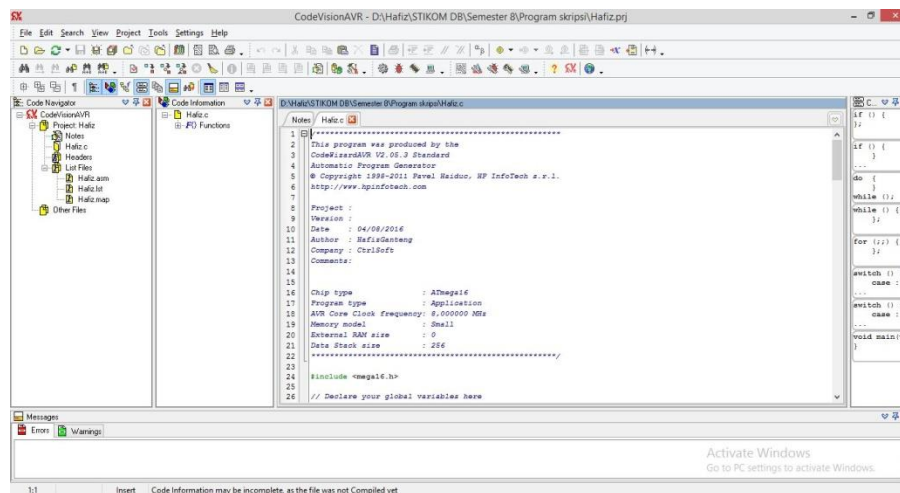
Received data (RxD) adalah sinyal yang diterima dari perangkat lain, pada perangkat lain tersebut sinyal didapat dari sinyal TxD (*Transmitted* data).

Baud Rate merupakan kecepatan pengiriman data antara perangkat dengan komputer. 1 *baud* merupakan 1 buah karakter yang dikirim. Besaran baud rate ini ada beberapa: 110, 1200 2400, 9600 19200, 38400, 57600, 115200. Satuan *baud rate* adalah bps, yang berarti *baud per second*. Pada tab USART ini *baud rate* yang digunakan adalah 9600 bps, berarti data yang dikirim memiliki laju 9600 karakter per detik.

Communication parameters digunakan untuk berkomunikasi antara perangkat/instrumen dengan komputer. Pengaturan komunikasi yang digunakan

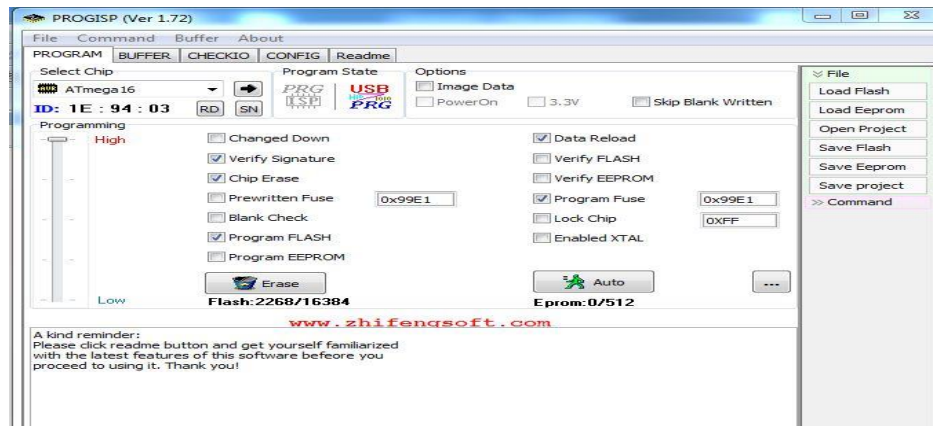
adalah 8 Data, 1 *stop*, *no parity* yang berarti jumlah data bit yang dikirim sebanyak 8, tanda bit yang dikirim sudah selesai, dan tanpa ada bit yang *error*.

Kemudian setelah menentukan *Chip*, *Port I/O*, *ADC*, akan muncul jendela *coding* program yang berupa *coding default*. Disini mulai dilakukan pengetikan *listing* program.



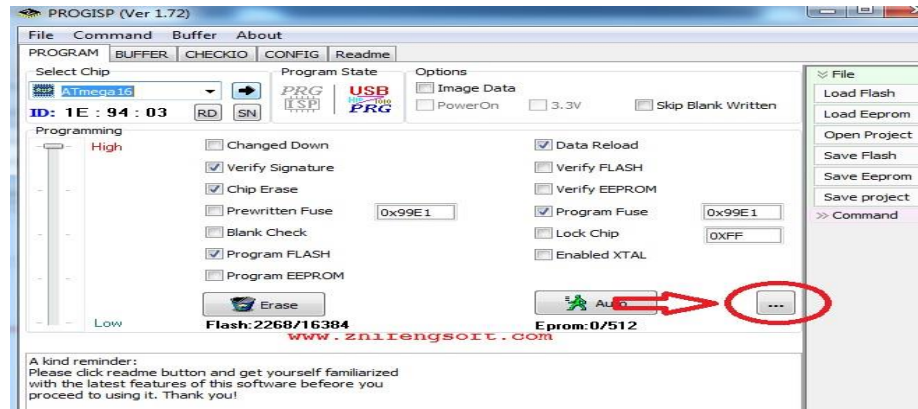
Gambar 5.14 Listing Program

Setelah *listing* program selesai dibuat, kemudian tekan tombol *Ctrl+F9* atau klik *Program* pilih *Build All* lalu OK. Tahapan berikutnya adalah memasukan program ke dalam mikrokontroler, *USB ISP Programmer* ini adalah *programmer tool* untuk mengunggah kode program terkompilasi (berkas dalam format *Intel HEX*) ke mikrokontroler Atmel yang mendukung *ISP (In-System Programming)*. Alat ini dapat digunakan dari Windows 10, dikenali sebagai *USB HID (Human Interface Device)* dengan *Vendor ID (VID) 0x03EB* dan *Product ID (PID) 0xC8B4*. Untuk menggunakan alat ini penulis menggunakan piranti lunak yang mendukung *USB ISP*, seperti *ProgISP Programmer*. Untuk lebih jelasnya perhatikan gambar 5.19 berikut.



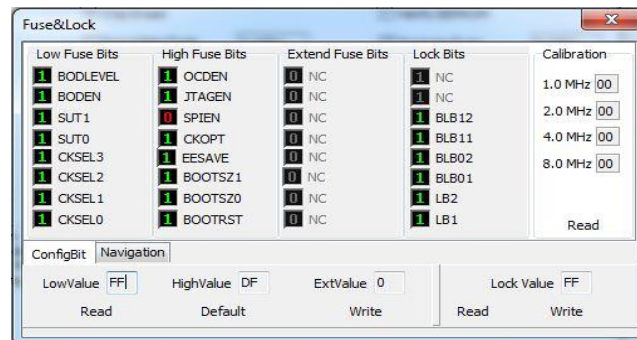
Gambar 5.15 *PROGISP (Ver 1.72)*

Pada gambar 5.19 adalah gambar tampilan awal *progisp* disini penulis menggunakan ver 1.72. Langkah berikutnya untuk memasukan program ke dalam mikrokontroler yaitu merubah *Fuse & Lock* pada *progisp*, untuk lebih jelasnya perhatikan gambar 5.20.



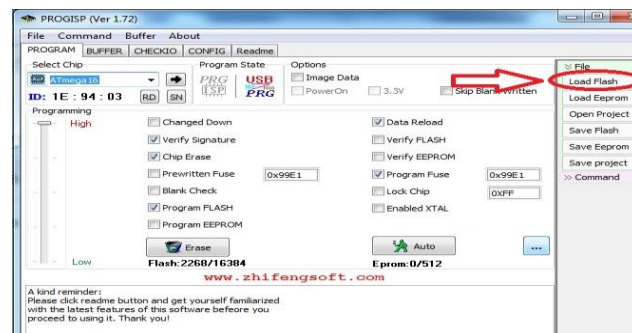
Gambar 5.16 *Setting Fuse & Lock ProgISP*

Selanjutnya setting *Fuse & Lock* seperti yang terlihat pada gambar 5.21 berikut :



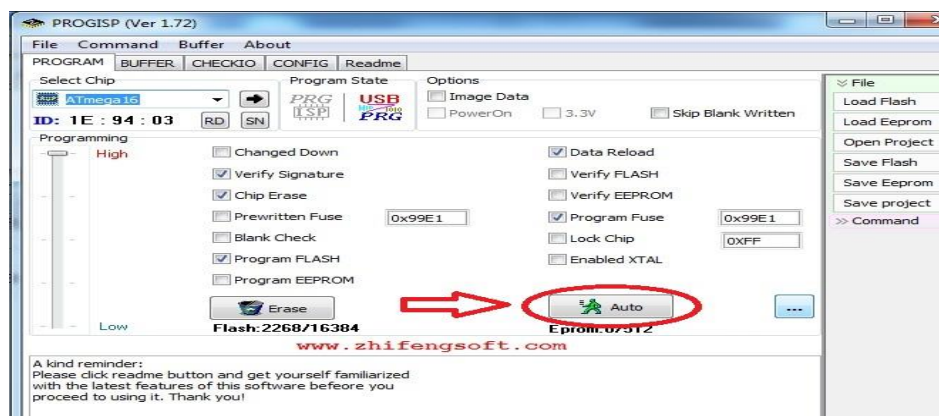
Gambar 5.17 Fuse & Lock ProgISP

Setelah setting *fuse & lock progISP* kemudian klik *Load Flash* pilih program yang akan dimasukkan kedalam mikrokontroler lalu klik OK.



Gambar 5.18 Load Flash

Setelah itu baru kemudian klik *Auto*, selanjutnya perhatikan gambar 5.23 berikut :



Gambar 5.19 Auto

Jika dibagian keterangan kiri bawah terdapat pesan *successfully done* itu tandanya program yang kita buat telah berhasil dimasukkan kedalam mikrokontroler.

5.3.2 Pengujian Perangkat Keras (*Hardware*)

1. Pengujian Tegangan Sumber

Tahap pertama yang dilakukan adalah pengecekan catu daya dengan tegangan input sebesar 12v dan output 12v. *PowerBank* yang digunakan adalah memiliki keluaran sebesar 5 V.

Hasil pengujian tegangan yang dihasilkan oleh *battery* dapat dilihat pada tabel 5.1.

Tabel 5.1 Pengujian Tegangan Sumber

Sumber Arus	Tegangan Input	Tegangan Output
<i>PowerBank</i>	5 V	5 V
Catu Daya	12 V	12 V

2. Pengujian Tegangan AT-Mega16

Rangkaian ini merupakan otak dari seluruh rangkaian. Semua rangkaian yang ada dikendalikan *input output*-nya oleh rangkaian mikrokontroler ini. Proses pengujian rangkaian ini adalah dengan menghubungkan setiap *port* dengan satu LED. Adapun hasil dari pengujian tegangan AT Mega16 ini dapat dilihat pada tabel 5.2.

Tabel 5.2 Pengujian Arus AT Mega16

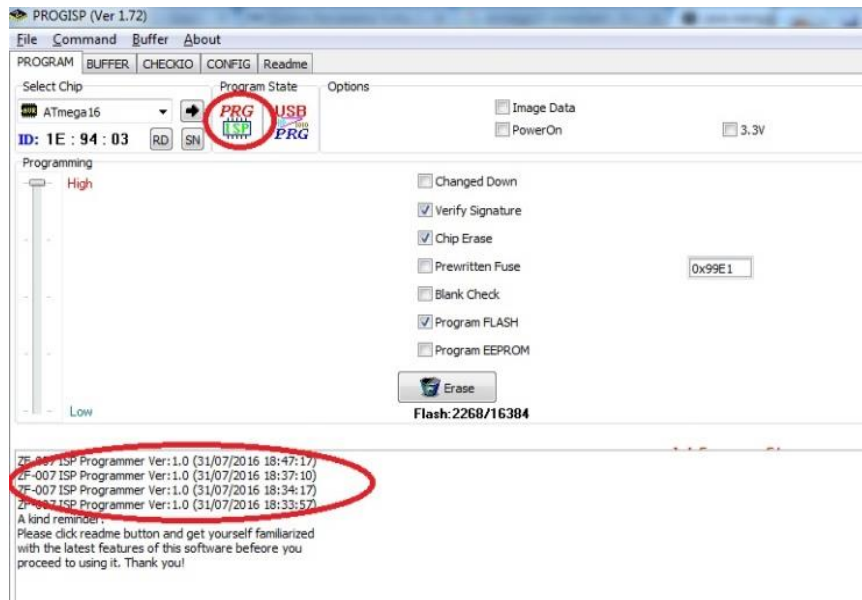
Sumber	Tegangan Input	Tegangan Output
AT Mega 16	1 A	4.9 A

3. Pengujian Downloader

Tahap pertama dilakukan *testing downloader*, yang pertama dilakukan adalah pengecekan rangkaian *regulator* ke mikrokontroler apakah sudah berukuran 5 V DC, sebab tegangan yang baik untuk mikrokontroler adalah 5 V DC, apabila kurang maupun lebih dapat mengganggu kinerja mikrokontroler bahkan dapat merusak mikrokontroler.

Hal berikutnya yang akan dilakukan adalah menghubungkan PC dengan *downloader* menggunakan *port* serial usb. Rangkaian *downloader* terdiri dari USB ISP, 1 buah mikrokontroler dan *socket 40 pin*, dan 1 buah Xtal 12 Mhz serta 2 buah *capasitor*. Untuk port ke 1 pada *header usb isp* dihubungkan ke pin ke 6 pada Atmega16, port ke 2 *header usb isp* dihubungkan ke pin Vcc Atmega 16, port ke 3 pada *header usb isp* tidak dihubungkan atau *Nc (No Connection)*, port ke 4, 6, 8, 10 pada *header usb isp* dihubungkan ke Gnd Atmega16, port ke 5 pada *header usb isp* dihubungkan ke pin ke 9 *reset* Atmega16, port ke 7 pada *header usb isp* dihubungkan ke pin 8 SCK Atmega16, dan port ke 9 pada *header usb isp* dihubungkan ke pin ke 7 MISO Atmega16.

Jika pada bagian *PROGISP* tulisannya berubah menjadi warna merah maka *USB ISP* siap dipakai. Perhatikan gambar 5.27 berikut :



Gambar 5.20 USB ISP Siap Digunakan

4. Pengujian Sensor DS18B20

Rangkaian sensor ds18b20 terdapat pada ATmega 16 yang mana sensor ds18b20 ini adalah sensor pembaca suhu air, pada rangkaian ATmega 16 sensor ds18b20 di letakkan pada PORTC.2, PORTC.3, PORTC.4 dan PORTC.5 semua sebagai Data. Adapun hasil dari pengujian sensor ds18b20 ini dapat dilihat pada tabel 5.3 di bawah ini :

Tabel 5.3 Pengujian Sensor DS18B20

Uji Coba	Hasil
Pada Saat Suhu Air Belum di Panaskan Tetapi Kondisi Alat Hidup (adanya arus listrik)	Sensor Aktif
Pada Saat Suhu Air di Panaskan	Sensor Aktif

Hasil dari percobaan di atas, pada saat suhu air belum di panaskan akan tetapi kondisi alat sudah diberikan arus listrik atau dalam keadaan hidup, maka sensor aktif, kondisi sensor aktif tersebut sedang membaca suhu air sekarang atau suhu air yang belum di panaskan hasilnya akan di tampilkan di LCD, sebaliknya sensor juga akan aktif ketika suhu air sedang di panaskan.

5. Pengujian Relay

Adapun relay ini sangat berperan penting dalam proses pemanasan air, karena pada saat proses pemanasan air, relay bekerja bagaikan saklar, yaitu membuka dan menutup arus yang masuk kedalam mikrokontoler tersebut. Adapun hasil dari pengujian relay dapat dilihat pada tabel 5.4 di bawah ini :

Uji Coba ke	Hasil/output
Pada Saat Mengatur Suhu 39° Tetapi Suhu Air sekarang 35 °	Relay Terbuka
Pada Saat Suhu Air Melebihi Dari Pengaturan Suhu	Relay Tertutup
Pada Saat Suhu Air Kurang Dari Pengaturan Suhu	Relay Terbuka

Tabel 5.4 Pengujian Relay

Hasil Uji coba relay di atas, relay akan terbuka dan memberikan arus listrik kepada *waterheater* ketika suhu air kurang dari pengaturan suhu atau pada saat hendak mengatur suhu, suhu air yang di atur melebihi dari suhu air sekarang, sebaliknya apabila suhu air sekarang melebihi dari pengaturan suhu maka relay akan tertutup dan tidak memberikan arus listrik kepada *waterheater*.

6. Perbandingan Pengujian Alat (*Prototype*) dan Termometer

Adapun maksud dari tujuan pengujian ini adalah untuk membandingkan antara Termometer dengan Alat yang sudah dibuat.

Percobaan Ke	Suhu Air Di Alat (<i>Prototype</i>)	Suhu Termometer
1	40°C	37°C
2	40°C	38,7°C
3	39°C	38,4°C
4	33°C	33,5°C
5	31°C	31,9°C

Tabel 5.5 Perbandingan Pengujian Alat dan Termometer

Pada perbandingan ini dilakukan lima kali percobaan agar dapat menentukan hasil maksimal

Pada percobaan pertama pada saat alat dihidupkan, mengatur suhu panas air yang diinginkan yaitu 40°C, setelah mencapai suhu air 40°C selanjutnya memasukkan termometer kedalam air ternyata hasil yang didapatkan pada termoter tersebut yaitu 37°C,

Pada percobaan kedua, masih sama dengan mengatur suhu air yang diinginkan yaitu sebesea 40°C, setelah mencapai 40°C, dimasukkan lagi termomter kedalam air panas tersebut, dan hasil yang didapatkan di termomter adalah 38,7°C

Pada percobaan ketiga, kali ini mengatur suhu air yaitu 39°C, setelah suhu air tersebut mencapai 39°C, di letakkan lagi termoter tersebut kedalam air, hasil yang didapatkan adalah 38,4°C

Dapat di simpulkan bahwa, terdapat kurang lebih hampir menyamai dari beberapa kali percobaan pada alat yang dibuat dengan termometer tersebut.

5.4 ANALISIS SISTEM SECARA KESELURUHAN

Untuk mendeteksi apabila terjadi kesalahan setelah uji coba, maka perlu dilakukan analisa rangkaian secara keseluruhan. Dari seluruh proses yang telah dilakukan, baik pengujian perangkat keras maupun perangkat lunak, dapat dikatakan bahwa alat ini dapat berfungsi sebagaimana yang penulis inginkan. Proses pembacaan sensor ds18b20 tidak terjadi kesalahan pada saat pembacaan data, dan juga pada saat pengujian relay, relay dalam kondisi yang bagus, sesuai dengan proses kinerja alat yang dibuat.

Pada saat kondisi alat dalam keadaan aktif, dengan mengatur berapa suhu air yang di inginkan, pada saat mengatur suhu air, dapat di lihat di LCD, setelah mengatur suhu yang di inginkan, selanjutnya kita akan melihat berapa suhu air yang sekarang, apa bila suhu air sekarang lebih besar dari pada suhu air yang di atur, maka relay tidak akan bekerja atau bisa di bilang tidak adanya pemanasan air, sebaliknya apabila suhu air sekarang lebih rendah dari pada suhu air yang di atur atau di setel, maka relay akan terbuka, *waterheater* dalam kondisi aktif sedang memanaskan air.