

BAB V

ANALISA DAN PENGUJIAN SISTEM

5.1 PENGUJIAN SISTEM

Sistem yang dirancang merupakan suatu sistem yang sudah terintegrasi, artinya karena sistem sudah terdiri dari beberapa bagian yang saling mendukung menjadikan sistem dapat bekerja dengan baik. Perancangan alat ini tentunya tidak lepas dari beberapa masalah mulai dari perancangan sampai dengan dengan pengaplikasian di lapangan. 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, yang akhirnya diharapkan untuk mendapatkan sistem yang lebih sempurna.

Pengujian alat ini memiliki beberapa tahap. Tahapan ini bertujuan memperkecil kemungkinan alat tidak bekerja saat dilakukan uji coba atau perbedaan hasil yang diinginkan. Pada penelitian ini akan dilakukan pengujian sebagai berikut:

1. Pengujian dengan Perangkat Keras (*Hardware*)
2. Pengujian dengan Perangkat Lunak (*Software*)
3. Analisa Sistem Secara Keseluruhan

5.2 PENGUJIAN KOMPONEN ALAT

Setelah rangkaian yang telah dirancang selesai dikerjakan, maka langkah selanjutnya adalah melakukan pengujian terhadap alat tersebut. Pengujian alat ini sangat penting karena dengan pengujian inilah bisa diketahui apakah alat yang dibuat nantinya dapat berjalan dengan sempurna atau tidak.

Pengujian yang dilakukan antara lain dengan cara menghubungkan rangkaian ke satu daya dan mengukurnya dengan alat ukur seperti *multitester*. Pengujian ini dilakukan untuk mengecek apakah diantara komponen-komponennya ada yang terhubung singkat atau jalurnya ada yang terputus. Hal ini dilakukan agar seluruh komponen yang terangkai tersebut dapat berfungsi seperti yang diinginkan. Sebab apabila ada kesalahan sedikit saja akan menimbulkan resiko yang cukup besar diantaranya bisa merusak alat yang telah dibuat yang lebih parah lagi bisa juga sekaligus merusak mikrokontroler atmega 16 dan IC 555 yang berfungsi sebagai pengatur alat pemancar gelombang tersebut.

5.2.1 Pengujian Tegangan Sumber

Tahap pertama yang dilakukan adalah pengecekan baterai yang digunakan adalah memiliki keluaran sebesar 5 V. Pengujian dilakukan dengan cara menghubungkan baterai ke ATmega16 dan modul speaker.

5.2.2 Pengujian Tegangan baterai

Tujuan dari pengujian baterai adalah untuk mendapatkan tegangan masuk agar berfungsi dengan baik. Pada perancangan sistem ini berbasis mikrokontroler, catu daya berfungsi sebagai daya yang digunakan sebagai sumber daya untuk

menghidupkan seluruh komponen yang dibutuhkan. Hasil pengujian yang di dapatkan pada pengujian baterai dapat dilihat pada tabel 5.1 berikut :

Tabel 5.1 Pengujian Tegangan Baterai

Battrai	Tegangan <i>Input</i>	Tegangan <i>Output</i>
Panasonic neo	9 V	5 V
Lith-poly	5 V	4.9 V

5.2.3 Pengujian Tegangan Speaker

Speaker terdiri dari kutub positif, negatif, dan gendang magnet. positif berfungsi sebagai *input* dari rangkaian IC 555, negatif sebagai pemberi ground, sedangkan gendang magnet berfungsi sebagai pemancar gelombang audio dari IC 555. Pengujian speaker ini dengan menghubungkan modul speaker ke modul ATmega16 sesuai dengan pinya. Adapun hasil pengujian tegangan yang dihasilkan oleh tegangan speaker ini dapat dilihat pada tabel 5.2. :

Tabel 5.2 Pengujian Tegangan Speaker

Sumber	Tegangan <i>Input</i>	Tegangan Dan <i>Range Output</i>
Speaker	5 V	4.6 V

5.2.4 Pengujian Tegangan IC 555

Pada dasarnya pengaplikasian utama IC 555 ini digunakan sebagai *Timer* (Pewaktu) dengan operasi rangkaian *monostable* dan *Pulse Generator* (Pembangkit Pulsa) dengan operasi rangkaian *astable*. Selain itu, dapat juga digunakan sebagai Time Delay Generator dan Sequential Timing. adapun hasil pengujian IC555 ini dapat dilihat pada tabel 5.3. :

Tabel 5.3 Pengujian Tegangan IC555

Sumber	Tegangan Input	Tegangan Output
IC555	5 V	3.9 V

5.2.5 Pengujian Tegangan ATmega16

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 beberapa rangkaian. Adapun hasil dari pengujian tegangan ATmega16 ini dapat dilihat pada tabel 5.4. :

Tabel 5.4 Pengujian Tegangan ATmega16

Sumber	Tegangan Input	Tegangan Output
ATmega 16	5 V	4.8 V

5.2.6 Pengujian Tegangan LCD (*Liquid Crystal Display*)

Pengujian LCD (*Liquid Crystal Display*) ini untuk mengetahui apakah LCD dapat berfungsi dengan baik atau tidak. Pengujian pertama yang dilakukan dengan memberi tegangan pada *pin catu daya*. Maka LCD akan menyala, namun demikian tidak berarti LCD akan bekerja dengan baik atau tidak. Pengujian tegangan yang dilakukan LCD (*Liquid Crystal Display*) ini dapat dilihat pada tabel 5.5 berikut :

Tabel 5.5 Pengujian Tegangan LCD

Sumber	Tegangan Input	Tegangan Output
LCD (<i>Liquid Crystal Display</i>)	5 V	4.7 V

5.2.7 pengujian Tegangan SC1162

Transistor SC1162 adalah bagian dari semikonduktor dalam part elektronik di mana “transist” arus dan tegangan untuk kemudian berubah bentuk atau besaran-besarannya dalam daya tertentu. Transistor menjadi komponen aktif yang utama di semua rangkaian elektronik. Adapun hasil pengujian tegangan pada SC1162 dapat dilihat di table 5.6 sebagai berikut. :

Tabel 5.6 Pengujian Tegangan SC1162

Sumber	Tegangan Input	Tegangan Output
SC1162	5 V	35 V

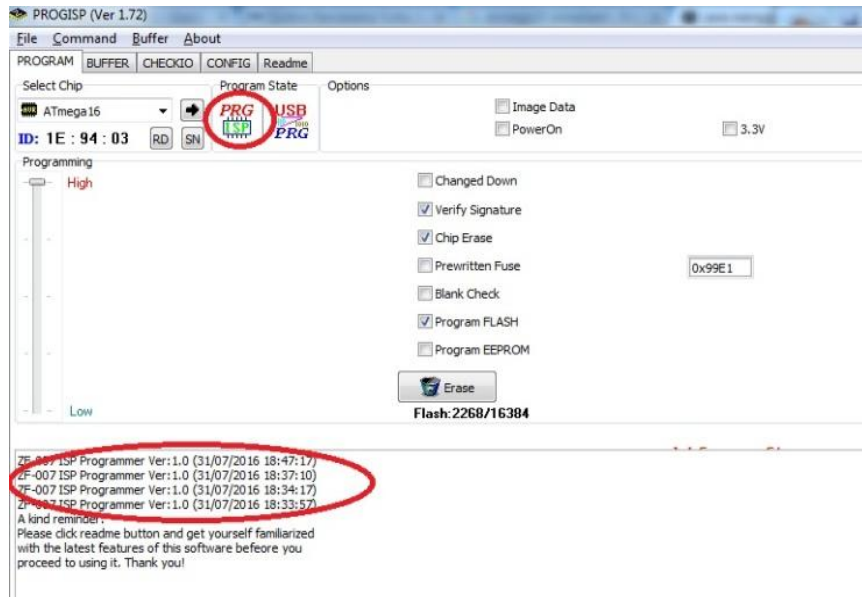
5.2.8 Pengujian Downloader

Tahap pertama dilakukan *testing downloader*, yang pertama dilakukan adalah pengecekan arus 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 5.1 berikut :



Gambar 5.1 USB ISP Siap Digunakan

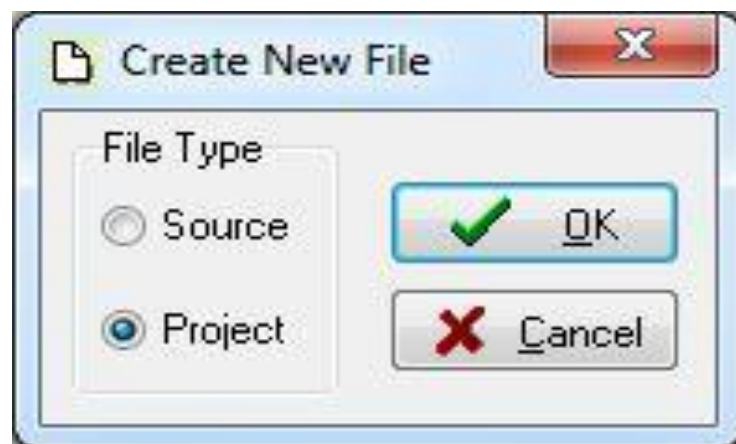
5.3 PENGUJIAN APLIKASI

Hal yang akan pertama kali dilakukan dalam pengujian perangkat lunak adalah menentukan aplikasi (*software*) yang akan di gunakan untuk mengisi program pada Sistem Alat pengusir kutu beras pada tempat penyimpanan beras dengan frekuensi audio.

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.



Gambar 5.2 Menu Membuat Program Baru

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



Gambar 5.3 Konfirmasi *Project*

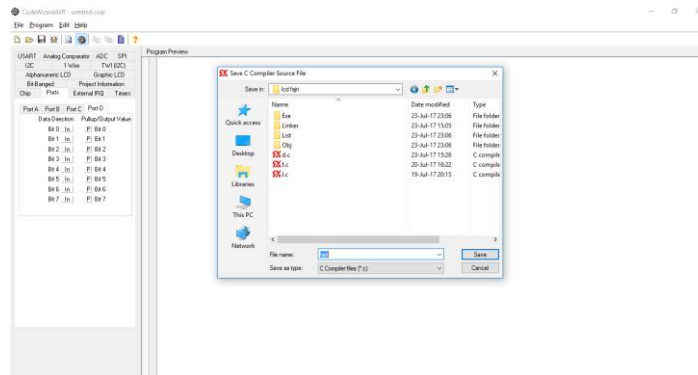
Kemudian akan muncul kembali jendela *Code Wizard AVR*, pada bagian ini tentukan *Chip* yang akan digunakan. karena *Chip* yang akan digunakan Atmega16 maka *checklist* pilihan yang pertama yaitu *AT90, Attiny, Atmega, FPSLIC* lalu klik OK.



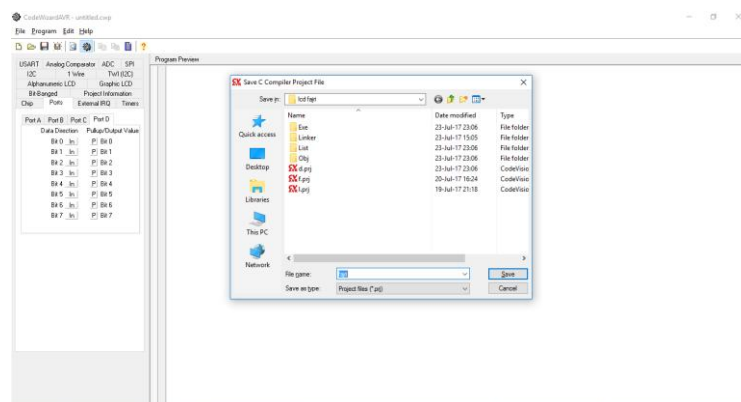
Gambar 5.4 *Code Wizard AVR*

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*

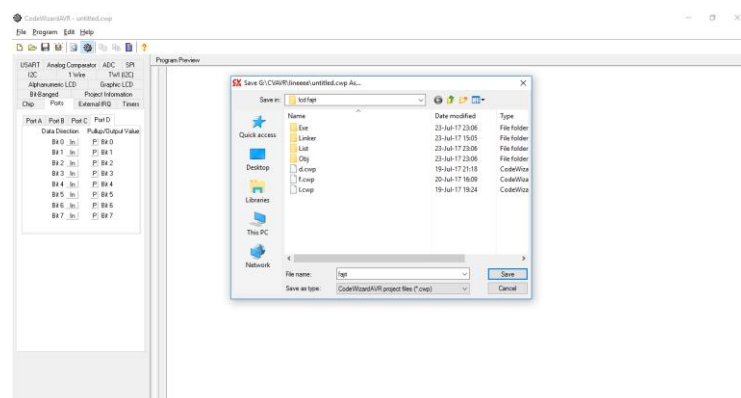
File ekstensi *.cwp*, untuk lebih jelasnya bisa dilihat pada gambar 5.5, 5.6, dan 5.7 berikut :



Gambar 5.5 Save Pertama file ekstensi .C



Gambar 5.6 Save Kedua File Ekstensi .prj



Gambar 5.7 Save Ke Tiga File Ekstensi .cwp

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 :

1. Tab *Chip*

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



Gambar 5.8 Tab *Chip*

2. Tab *Ports*

Port A : Tidak digunakan sehingga tidak perlu diatur *Port I/O*

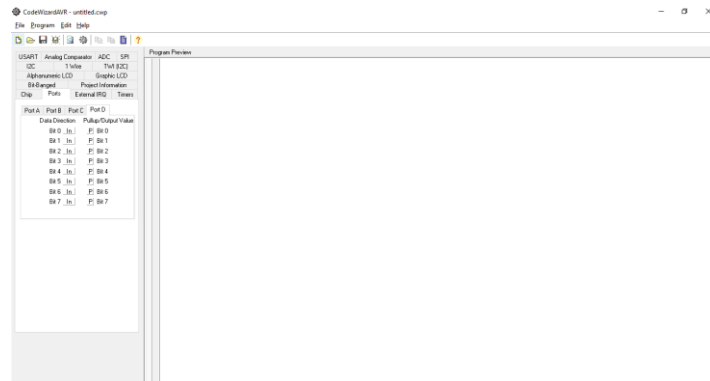
Port B : Digunakan sebagai *Output*, sehingga *Data Direction* dirubah menjadi *Out* dari *Bit 0* sampai *Bit 7* dan *Pulup/Out Value* dirubah menjadi 0 dari *Bit 0* sampai *Bit 7*



Gambar 5.9 Konfigurasi Port B

Port C : Digunakan sebagai *Output*, sehingga *Data Direction* dirubah menjadi *Out* dari *Bit 0* sampai *Bit 7* dan *Pulup/Out Value* dirubah menjadi 0 dari *Bit 0* sampai *Bit 7*.

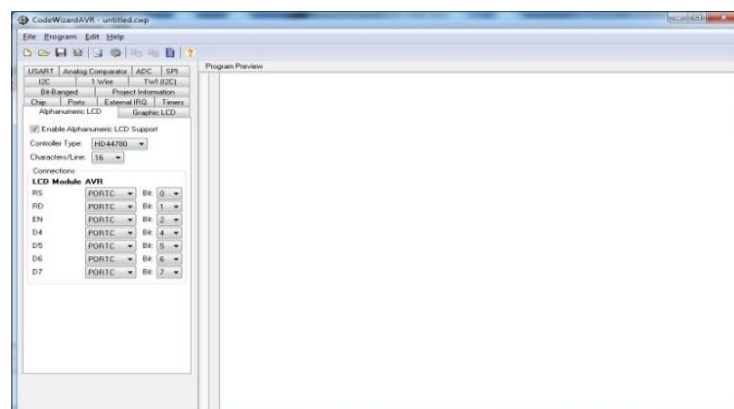
Port D : Digunakan untuk tombol



Gambar 5.10 Konfigurasi Port D

3. Tab *Alphanumeric LCD*

Pada Tab *Alphanumeric LCD* yaitu dimana Tab untuk menentukan *Port* tampilan *LCD*



Gambar 5.11 Konfigurasi Port LCD

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

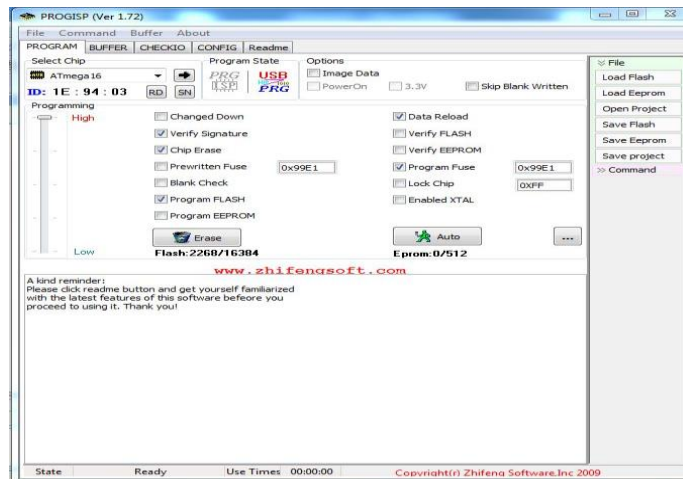
```

CodeVisionAVR - G:\C\AVR\Icd\fpj\fpj1.pj
File Edit Search View Project Tools Settings Help
CodeVisionAVR
122 int main()
123 {
124     while (1)
125     {
126
127         if (PIND.2 == 0)
128             count = 0;
129         else if (PIND.3 == 0)
130             count = 1;
131     }
132
133     if (count == 0)
134         lcd_clear();
135     lcd_gotoxy(0,0);
136     lcd_puts("Alat pengukur suhu");
137     } else if (count == 1)
138     {
139         lcd_clear();
140         lcd_gotoxy(0,0);
141         lcd_puts("Masukkan Detil");
142         lcd_gotoxy(0,1);
143         lcd_puts(waktu);
144     }
145     PORTB = 255;
146 }
147
148
149
150
151
152
153
154
155
156
157
158
159
Messages
Errors Warnings
172.1 Insert Code Information may be incomplete, as the file was not Compiled yet

```

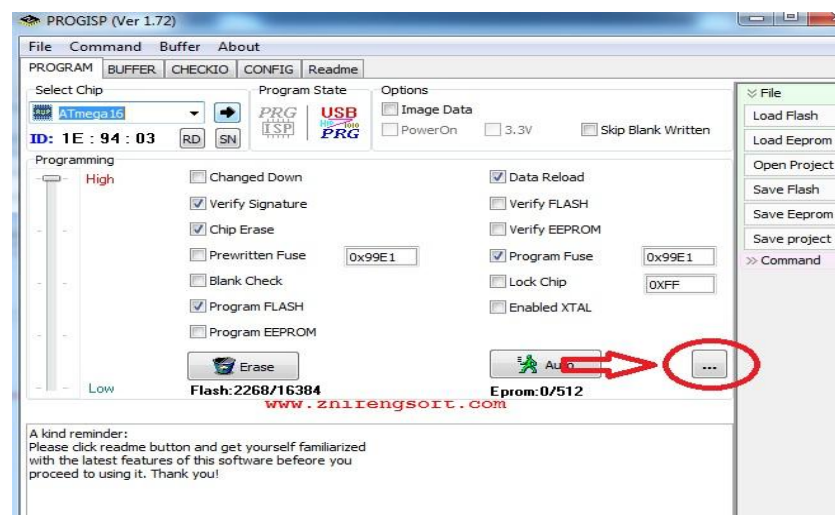
Gambar 5.12 Listing Program

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 7, 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 5.13 berikut.



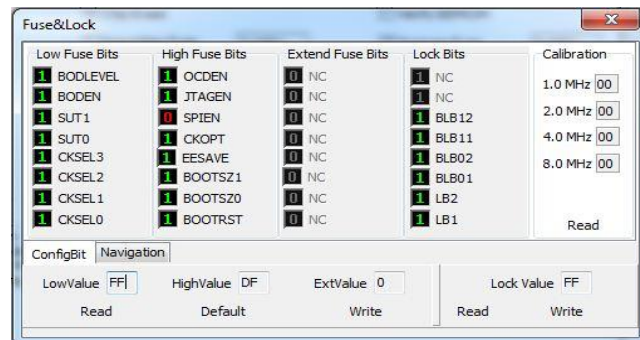
Gambar 5.13 PROGISP (Ver 1.72)

Pada gambar 5.13 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 5.14.



Gambar 5.14 Setting Fuse & Lock ProgISP

Selanjutnya setting *Fuse & Lock* seperti yang terlihat pada gambar 5.16 berikut :



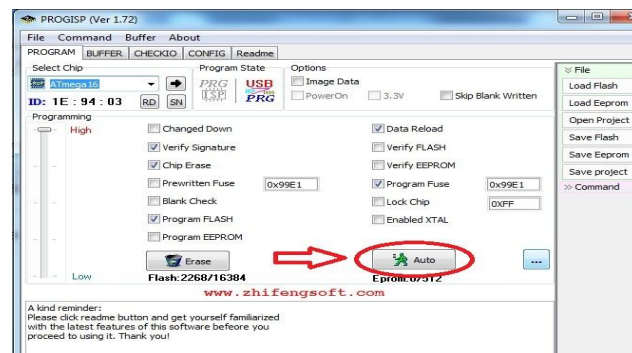
Gambar 5.15 Fuse & Lock ProgISP

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



Gambar 5.16 Load Flash

Setelah itu baru kemudian klik *Auto*, selanjutnya perhatikan gambar 5.18 berikut



Gambar 5.17 Auto

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

5.4 ANALISA KERJA ALAT SECARA KESELURUHAN

Analisis keseluruhan pada alat yang dibuat sudah terjadi sinkronisasi antara hardware dan software. pada software fitur-fitur untuk mengatur dan memancarkan gelombang dapat berjalan dengan baik. Pada hardware gelombang yang dihasilkan stabil sesuai pengaturan yang sudah dibuat. Pengamat ini dilakukan selama 2 hari berturut dengan menggunakan selang waktu 4 jam pada hari selasa tanggal 11 juli 2017 dan hari rabu tanggal 12 juli 2017.

Jumlah kutu dalam beras yang menjadi sampel berjumlah 10 ekor penghitungan jumlah sampel kutu beras pada penelitian ini secara manual yaitu dengan cara menghitung 1 per 1. Hasil pengamatan penelitian ini pada tabel 5.7.

Tabel 5.6 hasil akhir keseluruhan

Hari	Pukul	Pemancar gelombang	Kondisi kutu	Jumlah kutu	Kondisi kutu
Selasa,11 Juli 2017	7:00	Aktif	Tidak ada pergerakan kutu	10	Ada kutu
	11:00	Aktif	Bergerak menghindari	7	Ada kutu tetapi berangsur berkurang
	15:00	Aktif	Bergerak menghindari	3	Ada kutu tetapi lebih banyak berkurang
	19:00	Aktif	Tidak ada pergerakan kutu	0	Pergi semua
Rabu,12 Juli 2017	7:00	Tidak Aktif	Tidak ada pergerakan kutu	10	Ada kutu
	11:00	Tidak Aktif	Tidak ada pergerakan kutu	10	ada kutu

	15:00	Aktif	Bergerak menghindari	6	Ada kutu tetapi mulai berkurang
	19:00	Aktif	Bergerak menghindari	4	Ada kutu tetapi masih banyak

pada hari pertama dilakukan pengetesan alat selama 12 jam yang waktu pengecekannya dibagi dalam 4 jam. Hasil pengamatan pada 4 jam pertama dari jumlah kutu yang menjadi sample berjumlah 10 ekor belum memperlihatkan hasil yang diperlukan. Pada 4 jam berikutnya mulai memampakan hasil yang diinginkan terlihat dari jumlah kutu yang mulai berkurang walau tidak signifikan. Pada 4 jam berikutnya hasil yang diperlihatkan kutu yang hilang lebih banyak dibandingkan 4 jam awal semenjak alat ini dihidupkan. Dan pada 4 jam terakhir pengaktifan alat saya mendapatkan hasil yang diinginkan yaitu semua kutu beras yang ada pada sample penelitian telah pergi semua.

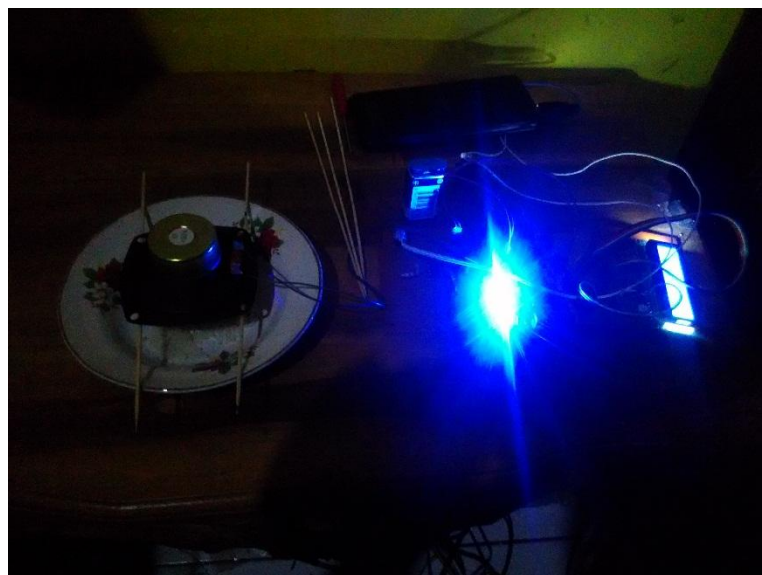
Pada hari kedua penelitian ini pada 8 jam pertama alat tidak dihidupkan dengan tujuan melihat perilaku 10 kutu sampel jika alat ini tidak diaktifkan. Pada 4 jam berikutnya alat ini diaktifkan terlihat jumlah kutu yang ada dalam beras tersebut mulai berkurang sebanyak 4 kutu. Dan pada 4jam berikutnya kutu berkurang lagi sebanyak 2 ekor.

Berikut gambar-gambar hasil penelitian yang dilakukan :

Kondisi awal saat persiapan pengujian



Gambar 5.18 Kutu Sampel Yang Sudah Disiapkan



Gambar 5.19 Alat Mulai Dijalankan

Gambar-gambar pada hari pertama



Gambar 5.20 Belum Terjadi Apa-Apa Pada Awal Pengaktifan Alat



Gambar 5.21 Kutu Mulai Berlarian Menjauhi Beras



Gambar 5.22 Kutu Masih Mencari Jalan Untuk Menjauh



Gambar 5.23 Kutu Tersisa Tinggal Sedikit



**Gambar 5.24 Tidak Ada Kutu Yang Tersisa Dari Sampel Saat Akhir
Pengamatan Hari Pertama**

Gambar-gamabr pada hari ke 2



Gambar 5.25 Tidak Terjadi Apa-Apa Karena Alat Belum Diaktifkan



Gambar 5.26 Masih Tidak Terjadi Apa-Apa



**Gambar 5.27 Kotu Yang Ada Mulai Berlarian Kabur Pada Saat Alat
Mulai Diaktifkan**



Gambar 5.28 Masih Ada Sisa Kutu Yang Berusaha Kabur



**Gambar 5.29 Sisa Sampel Kutu Yang Masih Tersisa Pada Hari
Pengamatan Kedua**