



**PENGUKUR TINGGI TANAMAN UNTUK PENGAMATAN
PERTUMBUHAN BIJI KACANG HIJAU BERBASIS ARDUINO UNO
DENGAN TAMPILAN LCD GRAFIK**

PROYEK AKHIR

Oleh

**Muhammad Zahroni Firdaus
NIM 111903102001**

**PROGRAM STUDI DIPLOMA III TEKNIK ELEKTRONIKA
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS JEMBER
2015**



**PENGUKUR TINGGI TANAMAN UNTUK PENGAMATAN
PERTUMBUHAN BIJI KACANG HIJAU BERBASIS ARDUINO UNO
DENGAN TAMPILAN LCD GRAFIK**

PROYEK AKHIR

Diajukan guna melengkapi proyek akhir dan memenuhi salah satu syarat
Untuk menyelesaikan Program Studi Diploma III Teknik Elektronika

Oleh

**Muhammad Zahroni Firdaus
NIM 111903102001**

**PROGRAM STUDI DIPLOMA III TEKNIK ELEKTRONIKA
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS JEMBER
2015**

PERSEMBAHAN

Sembah sujud dan syukur Alhamdulillah panjatkan kehadiran ALLAH SWT yang telah memberikan anugerah, rahmat dan nikmat. Tidak lupa Sholawat serta Salam tetap tercurahkan kepada Nabi Muhammad SAW. Dengan mengucapkan syukur Alhamdulillah proyek akhir ini saya persembahkan kepada :

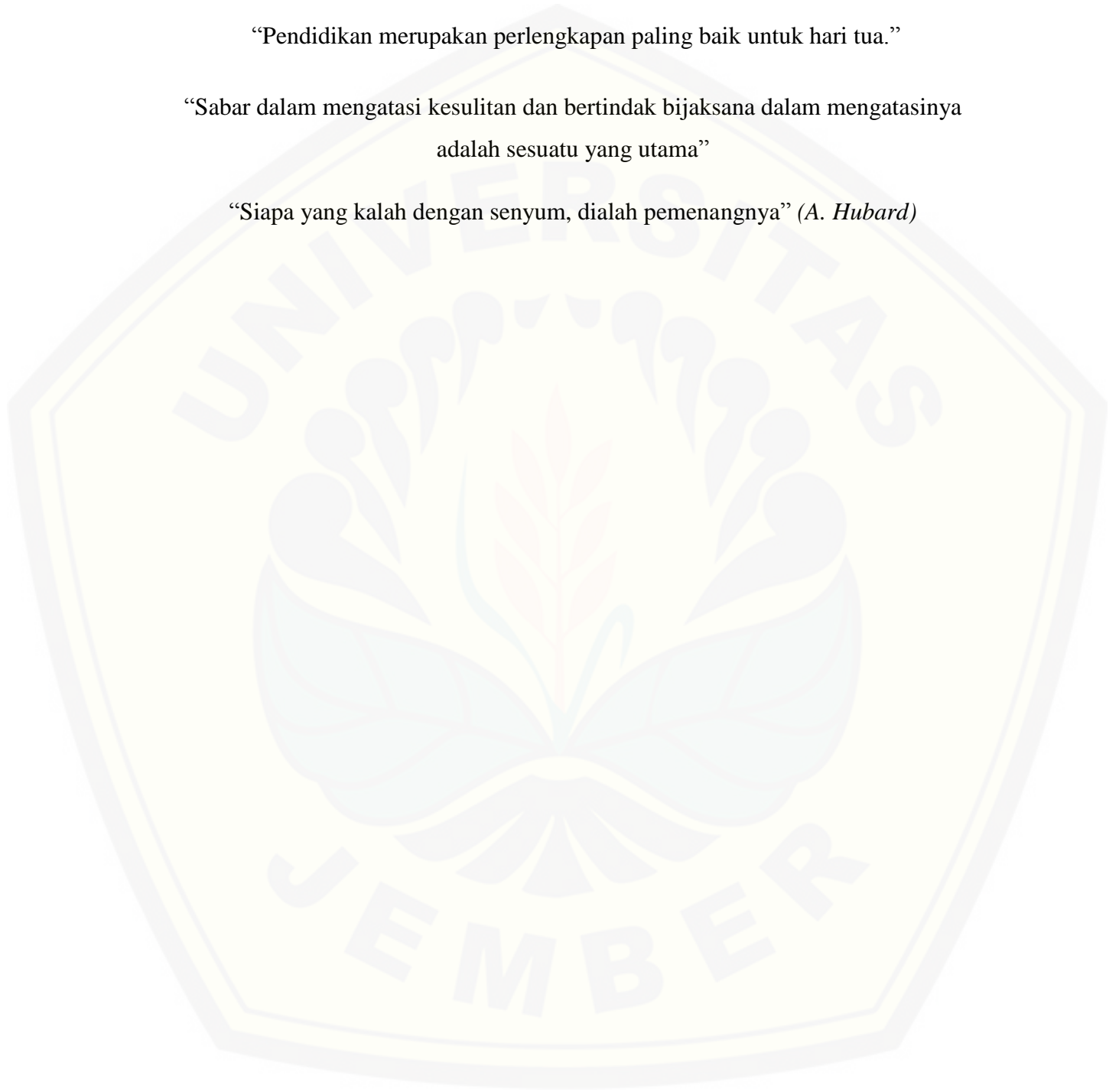
1. Bapak dan Ibuku yang senantiasa memberikan kasih sayang, semangat, nasehat dan telah berkorban sekuat tenaga demi tercapainya cita-cita buah hatinya.
2. Saudara-saudaraku yang telah memberikan nasihat, motivasi, bimbingan dan kasih sayangnya.
3. Temanku teknik elektronika yang telah memberikan semangat dan membantu menyelesaikan tugas akhir ini.
4. Guru-guruku dari TK, SD, SMP, SMA sampai Perguruan Tinggi terhormat, yang telah memberikan bekal ilmu yang bermanfaat bagi masa depanku.
5. Almamater Program Studi Teknik Elektronika Fakultas Teknik Universitas Jember yang kubanggakan.

MOTTO

“Pendidikan merupakan perlengkapan paling baik untuk hari tua.”

“Sabar dalam mengatasi kesulitan dan bertindak bijaksana dalam mengatasinya
adalah sesuatu yang utama”

“Siapa yang kalah dengan senyum, dialah pemenangnya” (*A. Hubard*)



PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Muhammad Zahroni Firdaus

NIM : 111903102001

Menyatakan dengan sesungguhnya bahwa proyek akhir yang berjudul: “Pengukur Tinggi Tanaman Untuk Pertumbuhan Biji Kacang Hijau Berbasis Arduino Uno Dengan Tampilan LCD Grafik” adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya dan belum pernah diajukan pada institusi manapun serta bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa adanya tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, 21 Juli 2015

Yang menyatakan,

Muhammad Zahroni Firdaus

NIM : 111903102001

PROYEK AKHIR

**PENGUKUR TINGGI TANAMAN UNTUK PENGAMATAN
PERTUMBUHAN BIJI KACANG HIJAU BERBASIS ARDUINO UNO
DENGAN TAMPILAN LCD GRAFIK**

Oleh

**Muhammad Zahroni Firdaus
NIM 111903102001**

Pembimbing

Dosen Pembimbing Utama : Bambang Supeno, S.T., M.T.

Dosen Pembimbing Anggota : H.R.B. Moch. Gozali, S.T.,M.T.

LEMBAR PENGESAHAN

Laporan Proyek Akhir berjudul “PENGUKUR TINGGI TANAMAN UNTUK PENGAMATAN PERTUMBUHAN BIJI KACANG HIJAU BERBASIS ARDUINO UNO DENGAN TAMPILAN LCD GRAFIK” oleh Muhammad Zahroni Firdaus NIM : 111903102001 telah diuji dan disahkan oleh Jurusan Teknik Elektro Fakultas Teknik Universitas Jember pada :

Hari, tanggal : Jum'at, 26 Juni 2015

Tempat : Ruang Sidang Fakultas Teknik Universitas Jember

Dosen Pembimbing Utama,

Dosen Pembimbing Anggota,

Bambang Supeno, S.T., M.T.
NIP. 19690630 199512 1 001

H.R.B. Moch. Gozali, S.T., M.T.
NIP. 19690608 199903 1 002

Dosen Penguji I,

Dosen Penguji II,

Suprihadi Prasetyono, ST., MT.
NIP. 19700404 199601 1 001

Dodi Setiabudi, S.T., M.T.
NIP. 19840531 200812 1 004

Mengesahkan
Dekan,

Ir. Widyono Hadi, M.T.
NIP. 19610414198902 1 001

PENGUKUR TINGGI TANAMAN UNTUK PENGAMATAN PERTUMBUHAN
BIJI KACANG HIJAU BERBASIS ARDUINO UNO DENGAN TAMPILAN LCD
GRAFIK

Muhammad Zahroni Firdaus

Jurusan Teknik Elektronika, Fakultas Teknik, Universitas Jember

ABSTRAK

Alat Bantu pendidikan mempunyai peran penting dalam pendidikan dan mempunyai banyak manfaat. Seperti menimbulkan minat sasaran pendidikan, membantu dalam mengatasi berbagai hambatan dalam proses pendidikan, membantu sasaran pendidikan untuk belajar dengan cepat dan belajar lebih banyak materi/bahan yang disampaikan, mempermudah penyampaian materi/bahan pendidikan/informasi oleh para pendidik atau pelaku pendidikan, membantu menegakkan pendidikan/informasi yang diperoleh. Alat pengukur tinggi tanaman ini merupakan alat bantu pendidikan pada salah satu mata pelajaran biologi tentang pengamatan pertumbuhan biji kacang hijau

Alat pengukur tinggi tanaman ini dibangun dari modul arduino sebagai pengendalinya dan menggunakan LCD Grafik sebagai penampilnya. Dengan pengukur ini kita dapat mengamati pertumbuhan biji kacang hijau pada pelajaran biologi yang nantinya grafik pertumbuhan akan langsung ditampilkan pada LCD Grafik

Alat pengukur tinggi tanaman yang dirancang berguna untuk pengamatan pertumbuhan biji kacang hijau dibidang pendidikan sebagai alat bantu pendidikan pada mata pelajaran biologi

Kata kunci: Kacang Hijau, Arduino, LCD Grafik,

**PENGUKUR TINGGI TANAMAN UNTUK PENGAMATAN
PERTUMBUHAN BIJI KACANG HIJAU BERBASIS ARDUINO UNO
DENGAN TAMPILAN LCD GRAFIK**

Muhammad Zahroni Firdaus

Electronics Engineering Departement, Engineering Faculty, Jember University

ABSTRACT

Aids education has an important role in education and have many benefits. Raises interest objectives such as education, help in overcoming various obstacles in the educational process, help target education to learn quickly and learn more material/materials presented, facilitate the delivery of materials/educational materials / information by educators or educational actors, help enforce education/information obtained.

Altimeter these plants are educational tools on one of the subjects on the observation of biological growth green beans. Altimeter these plants are built from modules arduino as Graphic LCD controller and use as performers. With this measure we can observe the growth of green beans on biology that will chart the growth will be directly displayed on the LCD Graphic.

Altimeter designed plants useful for observations of the growth of green beans in the field of education as a tool for education on the subjects of biology

Keywords: Green Beans, Arduino, LCD Graphic

RINGKASAN

PENGUKUR TINGGI TANAMAN UNTUK PENGAMATAN PERTUMBUHAN BIJI KACANG HIJAU BERBASIS ARDUINO UNO DENGAN TAMPILAN LCD GRAFIK; Muhammad Zahroni Firdaus NIM 111903102001; 2015: halaman; Program Studi Diploma Tiga (DIII), Jurusan Teknik Elektro, Fakultas Teknik Universitas Jember.

Pada alat pengukur tinggi tanaman biji kacang hijau ini menggunakan modul Arduino yang dilengkapi chip ATmega 328 sebagai pengendali input output untuk mengetahui nilai tinggi sensor ultrasonik SFR04 yang ditempatkan diatas biji kacang hijau untuk pengamatan selama 7 hari yang kemudian data pengukur tinggi yang diperoleh sensor ditampilkan pada LCD Grafik.

Data pertumbuhan tinggi biji kacang hijau selama 7 hari akan langsung ditampilkan pada LCD Grafik berupa grafik pertumbuhannya selama 7 hari serta tinggi aktualnya.

Alat pengukur tinggi tanaman ini sudah otomatis karena menggunakan modul Arduino yang dilengkapi chip ATmega 328 sebagai pengendali dan LCD Grafik sebagai penampilnya. Jadi kita hanya mengamati pertumbuhannya saja dikarenakan setelah alat ini melakukan pengamatan data tentang tinggi tanaman biji kacang hijau serta grafik pertumbuhannya sudah tampil pada LCD Grafik.

SUMMARY

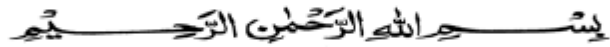
PENGUKUR TINGGI TANAMAN UNTUK PENGAMATAN PERTUMBUHAN BIJI KACANG HIJAU BERBASIS ARDUINO UNO DENGAN TAMPILAN LCD GRAFIK; Muhammad Zahroni Firdaus NIM 111903102001; 2015; pages; Program Studi Diploma Tiga (DIII), Electronics Engineering Departement, Engineering Faculty, Jember University

In the altimeter plant green beans using Arduino modules equipped ATmega 328 chip as the controller inputs and outputs to determine high value SFR04 ultrasonic sensor that is placed on green beans for observation for 7 days and then the data obtained by the sensor height gauge displayed on the LCD graph.

Data high growth green beans for 7 days will be directly displayed on the LCD graph in the form of a graph of growth for 7 days as well as the actual height.

This plant altimeter is automatically selected using the Arduino module equipped ATmega 328 chip as the controller and LCD Graphics as performers. So we just observe growth only because after this tool to make observations of high data about plant green beans and chart its growth already appears on the LCD graph.

PRAKATA



Puji syukur kehadiran Allah SWT atas segala rahmat dan hidayah-Nya sehingga laporan proyek akhir yang berjudul “PENGUKUR TINGGI TANAMAN UNTUK PENGAMATAN PERTUMBUHAN BIJI KACANG HIJAU BERBASIS ARDUINO UNO DENGAN TAMPILAN LCD GRAFIK”, dapat terselesaikan dengan baik. Laporan proyek akhir ini disusun untuk memenuhi salah satu syarat dalam menyelesaikan pendidikan Diploma Tiga (DIII) pada Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.

Terselesainya laporan proyek akhir ini tidak terlepas dari bantuan berbagai pihak, oleh karena itu disampaikan ucapan terimakasih kepada:

1. Allah SWT, dengan segala Keagungan dan Keajaiban-Nya yang senantiasa mendengar do'aku, menuntunku dari kegelapan, serta senantiasa menaungiku dengan rahmat dan hidayah-Nya.
2. Nabi Muhammad SAW, yang telah menjadi penerang di dunia dan suri tauladan bagi kita semua.
3. Bapak Ir. Widyono Hadi, M.T. selaku Dekan Fakultas Teknik Universitas Jember.
4. Bapak Syamsul Arifin, S.T., M.T. selaku Pembantu Dekan I Fakultas Teknik Universitas Jember.
5. Bapak Dr. Triwahyu Hardianto, S.T., M.T. selaku Ketua Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.
6. Bapak Stryo Budi Utomo, S.T.,M.T. selaku Ketua Program Studi Diploma Tiga (DIII) Teknik Elektro Universitas Jember.
7. Bapak Bambang Supeno, S.T., M.T. selaku Dosen Pembimbing Utama dan Bapak H.R.B. Moch. Gozali, S.T., M.T selaku Dosen Pembimbing Anggota yang telah meluangkan waktu dan pikiran serta perhatiannya guna memberikan bimbingan dan pengarahan demi terselesainya proyek akhir ini.

8. Bapak Ir. Widyono Hadi, M.T. selaku Dosen Pembimbing Akademik (DPA) yang telah memberikan bimbingan dan pengarahannya.
9. Bapak Suprihadi Prasetyono, ST., MT. dan Dodi Setiabudi, S.T., M.T. selaku Tim Penguji Proyek Akhir yang telah meluangkan waktu dan pikiran serta perhatiannya guna memberikan pengarahan demi terselesaikannya penulisan laporan proyek akhir ini.
10. Sivitas Akademika Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.
11. Teman seperjuangan Elektro 2011 Universitas Jember, tanpa kalian saya bukan apa-apa.
12. Semua pihak yang telah membantu dalam penulisan karya serta laporan proyek akhir ini.

Semoga laporan proyek akhir ini dapat bermanfaat dalam mengembangkan ilmu pengetahuan khususnya untuk disiplin ilmu teknik elektro, kritik dan saran diharapkan terus mengalir untuk lebih menyempurnakan proyek akhir ini dan diharapkan dapat dikembangkan untuk penelitian-penelitian selanjutnya.

Jember, 21 Juli 2015

Penulis

DAFTAR ISI

HALAMAN SAMPUL	i
HALAMAN JUDUL	ii
HALAMAN PERSEMBAHAN	iii
HALAMAN MOTTO	iv
HALAMAN PERNYATAAN	v
HALAMAN PERSETUJUAN	vi
HALAMAN PENGESAHAN	vii
RINGKASAN	x
PRAKATA	xii
DAFTAR ISI	xiv
DAFTAR GAMBAR	xvii
DAFTAR TABEL	xix
DAFTAR LAMPIRAN	xx
BAB 1. PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
1.6 Sistematika Penelitian.....	3

BAB 2. TINJAUAN PUSTAKA

2.1 Biji Kacang Hijau	4
2.2 Pertumbuhan Tanaman	5
2.3 Modul Arduino Uno ATmega328P	6
2.3.1 ATmega328P	11
2.3.2 Arduino-022.....	12
2.3.3 USART	13
2.3.4 Serial Monitor	13
2.3.2 ADC	14
2.4 Sensor Ultrasonik.....	15
2.5 LCD Grafik	17
2.6 RTC (<i>Real-Time Clock</i>) DS1307	19

BAB 3. METODE PENELITIAN

3.1 Tempat dan Waktu Penelitian.....	21
3.3.1 Tempat	21
3.3.2 Waktu.....	21
3.2 Alat dan Bahan.....	21
3.3 Diagram Blok Sistem	22
3.4 Rancangan Alat.....	23
3.4.1 Rangkaian <i>Power Supply</i>	23
3.4.2 Rangkaian RTC	23

BAB 4. HASIL DAN PEMBAHASAN

4.1 Pengujian Modul Arduino.....	25
4.2 Pengujian LCD Grafik	32
4.3 Pengujian Sensor Ultrasonik.....	37
4.4 Data Pengamatan Pertumbuhan Biji Kacang Hijau.....	41
4.5 Gambar Alat.....	47

BAB 5. PENUTUP

5.1 Kesimpulan	48
5.2 Saran	48

DAFTAR PUSTAKA	49
-----------------------------	-----------

LAMPIRAN.....	50
----------------------	-----------

DAFTAR GAMBAR

Gambar 2.1 Biji Kacang Hijau.....	4
Gambar 2.2 <i>Board</i> Arduino Uno.....	7
Gambar 2.3 Kabel <i>USB Board</i> Arduino Uno.....	7
Gambar 2.4 Program <i>memory map</i> ATmega328P	11
Gambar 2.5 Tampilan Arduino-0022.....	13
Gambar 2.6 Serial <i>Monitor</i> Arduino	14
Gambar 2.7 Modul dan Koneksi Sensor Ultrasonik	16
Gambar 2.8 Blok Diagram dan Gambar Kontur LCD Grafik 128x64.....	17
Gambar 2.9 Diagram pin RTC DS1307	20
Gambar 3.1 Diagram Blok Sistem	22
Gambar 3.2 Rangkaian <i>Power Supply</i>	23
Gambar 3.3 Rangkaian RTC	24
Gambar 4.1 Modul Arduino Uno.....	25
Gambar 4.2 <i>Flowchart</i> Sistem Arduino.....	26
Gambar 4.3 <i>Library</i> Program pada Arduino-1.5.5.....	28
Gambar 4.4 <i>Flowchart</i> Sistem Sensor Ultrasonik.....	29
Gambar 4.5 <i>Flowchart</i> Sistem RTC.....	30
Gambar 4.6 Tampilan Waktu dan Tanggal pada LCD	31
Gambar 4.7 LCD Grafik	32

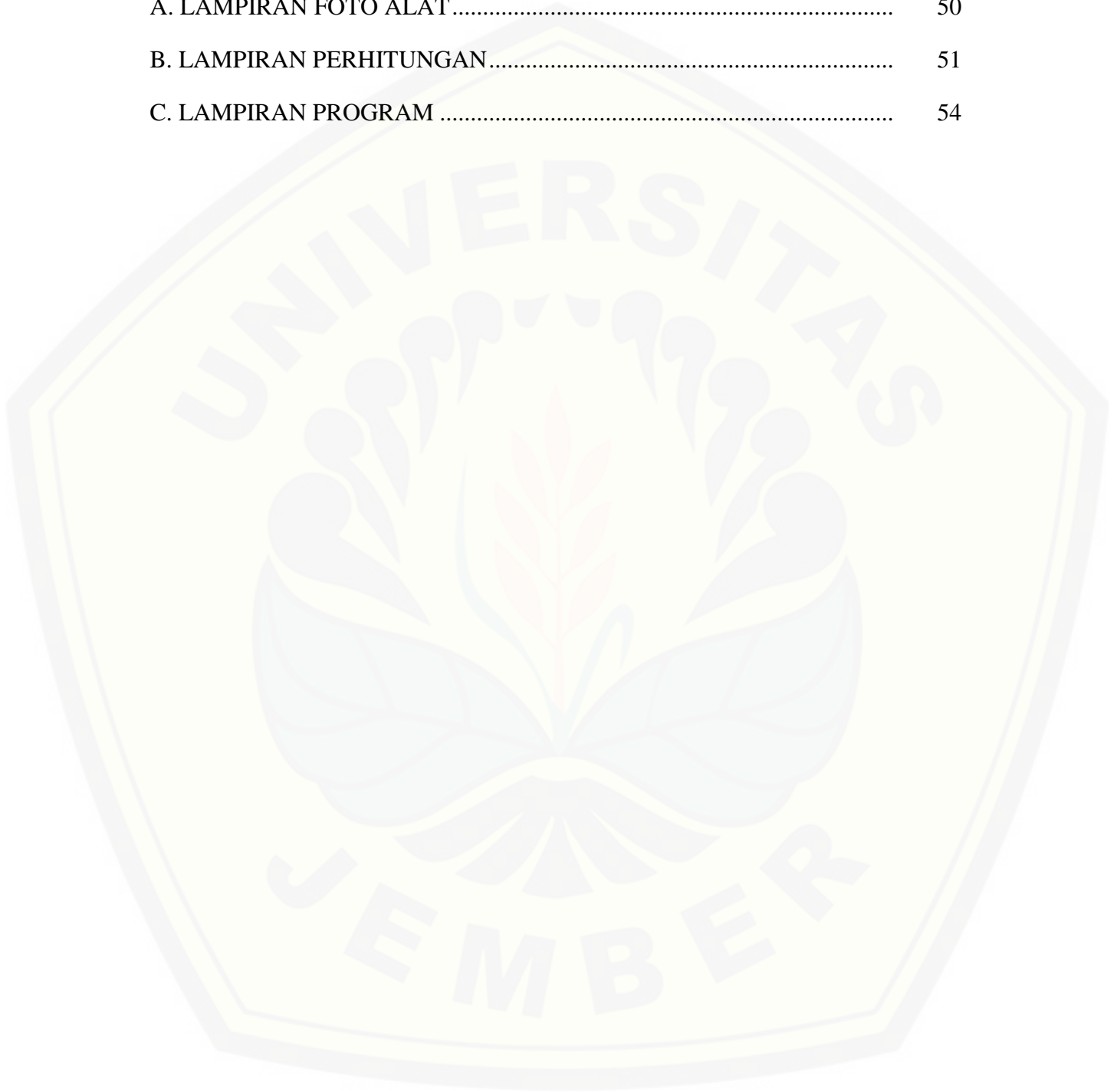
Gambar 4.8 Program pada Arduino-1.5.5	34
Gambar 4.9 Program pada Arduino-1.5.5	35
Gambar 4.10 <i>Library</i> LCD Grafik pada Arduino-1.5.5	36
Gambar 4.11 Tampilan LCD Grafik	36
Gambar 4.12 Sensor Ultrasonik	37
Gambar 4.13 Flowchart Perhitungan Sensor Ultrasonik	38
Gambar 4.14 <i>Library</i> Sensor Ultrasonik pada Arduino-1.5.5.....	40
Gambar 4.15 Tampilan Menu pada LCD Grafik	41
Gambar 4.16 <i>Flowchart</i> Sistem Alat Keseluruhan	42
Gambar 4.17 Grafik Pertumbuhan Biji Kacang Hijau Selama 7 Hari	44
Gambar 4.18 Tampilan Alat Keseluruhan	47

DAFTAR TABEL

Tabel 2.1 Diskripsi Arduino Uno.....	8
Tabel 2.2 Diskripsi Sensor Ultrasonik	16
Tabel 2.3 Diskripsi LCD Grafik.....	18
Tabel 4.1 Hasil Pengukuran (V) <i>Output</i> Modul Arduino Uno.....	27
Tabel 4.2 Port Pengujian Arduino Uno.....	29
Tabel 4.3 Hasil Pengukuran (V) Output LCD Grafik.....	33
Tabel 4.4 Pengujian pada Sensor Ultrasonik	39
Tabel 4.5 Pengamatan Pertumbuhan Biji Kacang Hijau.....	44
Tabel 4.6 Data Pengamatan Kedua	45
Tabel 4.7 Data Error % Pertumbuhan Selama 7 Hari Penamatan Pertama	45
Tabel 4.8 Data Error % Pertumbuhan Selama 7 Hari Penamatan Kedua.....	46
Tabel 4.9 Error % Rata-rata.....	47

DAFTAR LAMPIRAN

A. LAMPIRAN FOTO ALAT	50
B. LAMPIRAN PERHITUNGAN.....	51
C. LAMPIRAN PROGRAM	54



BAB I

PENDAHULUAN

1.1 Latar Belakang

Pendidikan merupakan salah satu unsur penting yang mempengaruhi kelangsungan hidup manusia, baik secara langsung maupun tidak langsung. Dalam pendidikan, media pembelajaran memiliki peran utama dalam keberhasilan pendidikan, sedangkan alat bantu pendidikan hanya menjadi perantara dalam memudahkan penyampaian informasi. Alat Bantu pendidikan mempunyai peran penting dalam pendidikan dan mempunyai banyak manfaat. Seperti menimbulkan minat sasaran pendidikan, membantu dalam mengatasi berbagai hambatan dalam proses pendidikan, membantu sasaran pendidikan untuk belajar dengan cepat dan belajar lebih banyak materi/bahan yang disampaikan, mempermudah penyampaian materi/bahan pendidikan/informasi oleh para pendidik atau pelaku pendidikan, membantu menegakkan pendidikan/informasi yang diperoleh.

Makhluk hidup semakin hari semakin tumbuh dan berkembang. Pertumbuhan merupakan proses kenaikan volume yang bersifat *irreversibel* (tidak dapat balik), dan terjadi karena adanya penambahan jumlah sel dan pembesaran dari tiap-tiap sel. Pada proses pertumbuhan biasa disertai dengan terjadinya perubahan bentuk. Pertumbuhan dapat diukur dan dinyatakan secara kuantitatif. Perkembangan adalah proses menuju dewasa. Proses perkembangan berjalan sejajar dengan pertumbuhan. Berbeda dengan pertumbuhan, perkembangan merupakan proses yang tidak dapat diukur. Dengan kata lain, perkembangan bersifat kualitatif, tidak dapat dinyatakan dengan angka.

Organisme disebut telah dewasa apabila telah mampu berkembang biak secara generatif. Pada tumbuhan, hal itu ditandai dengan terbentuknya bunga. Sedang pada manusia dan mamalia lainnya ditandai dengan telah berkembangnya *gonade* yang menghasilkan sel-sel kelamin (*gamet*). Pada pria ditandai dengan dimulainya produksi sel sperma oleh *testis*, dan pada wanita menghasilkan *ovum* (sel telur) yang dibentuk di *ovarium*.

Pengukur tinggi tanaman ini merupakan alat bantu pendidikan atau pengamatan yang dikhususkan pada pengamatan pertumbuhan biji kacang hijau. Tampilan berupa LCD Grafik dapat menunjukkan ukuran tinggi biji kacang hijau dalam centimeter untuk mendapatkan hasil pengamatan yang maksimal sekaligus grafik pertumbuhannya selama 7 hari. Alat ini menggunakan sensor jarak yang berfungsi untuk mendeteksi tinggi biji tanaman kacang hijau pada saat tumbuh.

Tanaman biji kacang hijau akan tumbuh dan berkembang dengan normal walaupun tanpa menggunakan media tanah. Pertumbuhan biji kacang hijau akan cepat terjadi pada tempat yang gelap, karena cahaya bisa mengurai hormon *auksin*, yaitu hormon yang berperan dalam perpanjangan batang biji kacang hijau.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka diambil rumusan masalah sebagai berikut :

1. Bagaimana merancang alat pengukur tinggi biji tanaman kacang hijau menggunakan modul arduino dengan tampilan LCD Grafik.
2. Bagaimana memfungsikan alat pengukur tinggi biji tanaman kacang hijau sebagai alat bantu pendidikan, atau sebagai alat pengamat pertumbuhan tanaman.

1.3 Batasan Masalah

Adapun pokok pembahasannya meliputi yaitu :

1. Nilai pengukuran berupa bilangan bulat dalam satuan centimeter.
2. Obyek yang digunakan adalah biji tanaman kacang hijau.
3. Hasil pengamatan akan langsung ditampilkan di LCD Grafik berupa grafik pertumbuhan selama 7 hari.

1.4 Tujuan

Dari pembuatan alat ini memiliki tujuan yang hendak dicapai yaitu:
Merancang/membuat alat yang dapat mengukur tinggi tanaman untuk pengamatan pertumbuhan biji tanaman kacang hijau.

1.5 Manfaat

Adapun manfaat dengan adanya alat ini adalah :
Dengan media ini diharapkan dapat membantu menciptakan alat bantu sarana pendidikan untuk mempermudah dalam pembelajaran ataupun pengamatan. Serta diharapkan dengan adanya media ini, pemerintah dapat mencantumkan kembali salah satu mata pelajaran biologi tentang pengamatan pertumbuhan biji tanaman kacang hijau.

1.6 Sistematika Penelitian

Laporan proyek akhir ini disusun berdasarkan sistematika sesuai berikut :

a. BAB I PENDAHULUAN

Berisi tentang latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat dan sistematika penelitian.

b. BAB II TINJAUAN PUSTAKA

Berisi tentang uraian teori dan alat yang dipakai dalam penelitian.

c. BAB III METODE PENELITIAN

Berisi tentang gambaran sistem penelitian secara keseluruhan baik itu berupa jadwal penelitian, diagram blok, *flowchart*.

d. BAB IV HASIL DAN PEMBAHASAN

Berisi tentang pembahasan hasil dan kinerja alat secara menyeluruh.

e. BAB V PENUTUP

Berisi tentang kesimpulan dari keseluruhan hasil penelitian, serta berisi saran.

BAB II TINJAUAN PUSTAKA

2.1 Biji Kacang Hijau

Kacang hijau adalah sejenis tanaman budidaya dan palawija yang dikenal luas di daerah tropika. Tumbuhan yang termasuk suku polong-polongan (*Fabaceae*) ini memiliki banyak manfaat dalam kehidupan sehari-hari sebagai sumber bahan pangan berprotein nabati tinggi. Biji kacang hijau lebih kecil dibanding biji kacang-kacangan lain. Biji kacang hijau terdiri atas tiga bagian utama, yaitu kulit biji (10%), *kotiledon* (88%) dan lembaga (2%). Biji Kacang hijau dapat ditunjukkan pada gambar 2.1.



Gambar 2.1 Biji Kacang Hijau
sumber : <http://blog.ub.ac.id>

Bagian paling bernilai ekonomi adalah bijinya. Di samping itu, panen kacang hijau ini harus dikerjakan beberapa kali. Peningkatan produksi kacang hijau dilakukan dengan cara memperbaiki kultur teknis petani, mendapatkan varietas-varietas yang produksinya tinggi dan masak serempak, serta peningkatan usaha pasca panen. Dari segi agronomis dapat dilakukan dengan tindakan pemupukan NPK dan pengaturan jumlah populasi, jarak tanam, *sanitasi*, pengendalian hama dan penyakit tanaman.

Di dalam kacang hijau terdapat banyak kandungan gizi antara lain *protein*, *calcium*, *fosfor*, vitamin B1, vitamin B2, vitamin E, *magnesium* dan zat antioksidan.

2.2 Pertumbuhan Tanaman

Pertumbuhan diartikan sebagai suatu proses penambahan ukuran atau volume serta jumlah sel secara *irreversible*, atau tidak dapat kembali ke bentuk semula. Perkembangan adalah peristiwa perubahan biologis menuju kedewasaan tidak dapat dinyatakan dengan ukuran tetapi dengan perubahan bentuk tubuh (*metamorfosis*) dan tingkat kedewasaan. Pertumbuhan dan perkembangan merupakan dua aktifitas kehidupan yang tidak dapat dipisahkan, karena prosesnya berjalan bersamaan.

Tumbuhan yang masih kecil, belum lama muncul dari biji dan masih hidup dari persediaan makanan yang terdapat di dalam biji, dinamakan kecambah (*plantula*). Awal perkecambahan dimulai dengan berakhirnya masa *dormansi*. Masa *dormansi* adalah berhentinya pertumbuhan pada tumbuhan dikarenakan kondisi lingkungan yang tidak sesuai. Berakhirnya masa *dormansi* ditandai dengan masuknya air ke dalam biji suatu tumbuhan, yang disebut dengan proses *imbibisi*. *Imbibisi* terjadi karena penyerapan air akibat potensial air yang rendah pada biji yang kering. Air yang *berimbibisi* menyebabkan biji mengembang dan memecahkan kulit pembungkusnya dan juga memicu perubahan metabolik pada *embrio* yang menyebabkan biji tersebut melanjutkan pertumbuhan. *Enzim-enzim* akan mulai mencerna bahan-bahan yang disimpan pada endosperma atau *kotiledon*, dan *nutrien-nutriennya* dipindahkan ke bagian *embrio* yang sedang tumbuh.

Biji dapat berkecambah karena di dalamnya terdapat *embrio* atau lembaga tumbuhan. *Embrio* atau lembaga tumbuhan mempunyai tiga bagian, yaitu akar lembaga/calon akar (*radikula*), daun lembaga (*kotiledon*), dan batang lembaga (*kaulikulus*).

Terdapat dua macam pertumbuhan, yaitu:

1. Pertumbuhan Primer

Terjadi sebagai hasil pembelahan sel-sel jaringan meristem primer. Berlangsung pada *embrio*, bagian ujung-ujung dari tumbuhan seperti akar dan batang.

2. Pertumbuhan Sekunder

Merupakan aktivitas sel-sel meristem sekunder yaitu kambium dan kambium gabus. Pertumbuhan ini dijumpai *dikotil*, *gymnospermae* dan menyebabkan membesarnya ukuran diameter tumbuhan. Mula-mula kambium hanya terdapat pada ikatan pembuluh, yang disebut kambium *vasic* atau kambium *intravaskuler*. Fungsinya adalah membentuk *xylem* dan *floem* primer. Selanjutnya *parenkim* akar/batang yang terletak di antara ikatan pembuluh, menjadi kambium yang disebut kambium *intervasis*.

2.3 Modul Arduino Uno ATmega328P

Arduino Uno adalah *board* berbasis mikrokontroler pada ATmega328P. *Board* ini memiliki 14 *digital input/output* pin (dimana 6 pin dapat digunakan sebagai *output PWM*), 6 *input analog*, 16 MHz *osilator* kristal, koneksi USB, jack listrik tombol *reset*. Pin-pin ini berisi semua yang diperlukan untuk mendukung mikrokontroler, hanya terhubung ke komputer dengan kabel USB atau sumber tegangan bisa didapat dari adaptor AC-DC atau baterai untuk menggunakannya.

Board Arduino Uno memiliki fitur-fitur baru sebagai berikut :

- 1,0 *pinout*: tambah SDA dan SCL pin yang dekat ke pin *aref* dan dua pin baru lainnya ditempatkan dekat ke pin *RESET*, dengan IO REF yang memungkinkan sebagai *buffer* untuk beradaptasi dengan tegangan yang disediakan dari *board* sistem. Pengembangannya, sistem akan lebih kompatibel dengan *Processor* yang menggunakan AVR, yang beroperasi dengan 5V dan dengan Arduino. Karena yang beroperasi dengan 3.3V. Yang kedua adalah pin tidak terhubung, yang disediakan untuk tujuan pengembangannya.

- *Circuit Reset*



Gambar 2.2 *Board Arduino Uno*

Sumber : <http://www.Arduino.cc>



Gambar 2.3 *Kabel USB Board Arduino Uno*

Sumber : <http://www.Arduino.cc>

Tabel 2.1 Deskripsi Arduino Uno

<i>Microkontroller</i>	Atmega 328P
<i>Operasi Voltage</i>	5 V
<i>Input Voltage</i>	7-12 V (Rekomendasi)
<i>Input Voltage</i>	6-20 (<i>Limits</i>)
I/O	14 pin (6 pin untuk PWM)
Arus	50 Ma
<i>Flash Memory</i>	32 KB
<i>Bootloader</i>	SRAM 2 KB
EEPROM	1 KB
Kecepatan	16 Mhz

a. Catu Daya

Arduino Uno dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal. Sumber listrik dipilih secara otomatis. Eksternal (non-USB) daya dapat datang baik dari AC-DC adaptor atau baterai. Adaptor ini dapat dihubungkan dengan cara menghubungkannya *plug* pusat-positif 2,1 mm ke dalam *board* colokan listrik. *Lead* dari baterai dapat dimasukkan ke dalam *header* pin Gnd dan *Vin* dari konektor *Power*. *Board* dapat beroperasi pada pasokan daya dari 6 - 20V. Jika diberikan dengan kurang dari 7V, jika pin 5V dapat menyuplai kurang dari 5V dan *board* mungkin tidak stabil. Jika menggunakan lebih dari 12V, regulator tegangan bisa panas dan merusak *board*. Rentang yang dianjurkan adalah 7-12V.

Pin catu daya adalah sebagai berikut:

- a. *Vin*. Tegangan *input* ke *board* Arduino ketika menggunakan sumber daya eksternal (sebagai lawan dari 5 volt dari koneksi USB atau sumber daya lainnya diatur). Dapat juga menyediakan tegangan melalui pin ini, atau, jika memasok tegangan melalui colokan listrik, mengaksesnya melalui pin ini.

- b. 5V. Catu daya diatur digunakan untuk daya mikrokontroler dan komponen lainnya di *board*. Hal ini dapat terjadi baik dari *Vin* melalui regulator *on-board*, atau diberikan oleh USB .
- c. 3,3V pasokan yang dihasilkan oleh regulator *on-board*. Menarik arus maksimum adalah 50 mA.
- d. GND

b. Memory

ATmega328 ini memiliki 32 KB dengan 0,5 KB digunakan untuk *loading file*. Kemudian juga memiliki 2 KB dari SRAM dan 1 KB dari EEPROM.

c. Input & Output

Masing-masing dari 14 pin *digital* pada Arduino Uno dapat digunakan sebagai *input* atau *output*, menggunakan fungsi *pinMode*, *digitalWrite*, dan *digitalRead*. Ketiganya beroperasi pada 5V. Setiap pin dapat memberikan atau menerima maksimum 40 mA dan memiliki resistor *pull-up* internal dari 20-50 K Ω . Selain itu, beberapa pin memiliki fungsi khusus:

- Serial: 0 (RX) dan 1 (TX). Digunakan untuk menerima (RX) dan mengirimkan (TX) data TTL serial. Pin ini terhubung ke pin yang sesuai dari chip ATmega8U2 *USB-to-Serial* TTL.
- Eksternal Interupsi: 2 dan 3. Pin ini dapat dikonfigurasi untuk memicu interupsi pada nilai yang rendah, tapi naik atau jatuh, atau perubahan nilai. Lihat *attachInterrupt* fungsi untuk rincian.
- PWM: 3, 5, 6, 9, 10, dan 11. Menyediakan 8-bit *output* PWM dengan *analog Write* fungsi.
- SPI: 10 (SS), 11 (mosi), 12 (MISO), 13 (SCK). Pin ini mendukung komunikasi SPI menggunakan perpustakaan SPI.
- LED: 13. Ada *built-in* LED terhubung ke pin digital 13. Ketika pin adalah nilai tinggi, LED menyala, ketika pin adalah rendah, itu *off*.

Arduino Uno memiliki 6 *input analog*, diberi label A0 melalui A5, masing-masing menyediakan 10 bit resolusi yaitu 1024 nilai yang berbeda. Secara *default* sistem mengukur dari tanah sampai 5V.

- TWI: A4 atau SDA pin dan A5 atau SCL pin. Mendukung komunikasi TWI
- Aref. Referensi tegangan untuk *input analog*. Digunakan dengan *analogReference*.
- Reset.

d. Komunikasi

Arduino Uno memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, arduino lain, atau mikrokontroler lain. ATmega328 ini menyediakan UART TTL (5V) komunikasi serial, yang tersedia pada pin *digital* 0 (RX) dan 1 (TX). Sebuah ATmega16U2 pada saluran *board* ini komunikasi serial melalui USB dan muncul sebagai *com port virtual* untuk perangkat lunak pada komputer. *Firmware* arduino menggunakan USB *driver* standar COM, dan tidak ada *driver* eksternal yang dibutuhkan. Namun, pada *Windows*, file .Inf diperlukan. Perangkat lunak arduino termasuk *serial monitor* yang memungkinkan data sederhana yang akan dikirim ke *board* arduino. Led pada RX dan TX di *board* akan berkedip ketika data sedang dikirim melalui chip *USB-to-serial* dan koneksi USB ke komputer. ATmega328 ini juga mendukung komunikasi I2C (TWI) dan SPI. Fungsi ini digunakan untuk melakukan komunikasi *inteface* pada sistem.

e. Programming

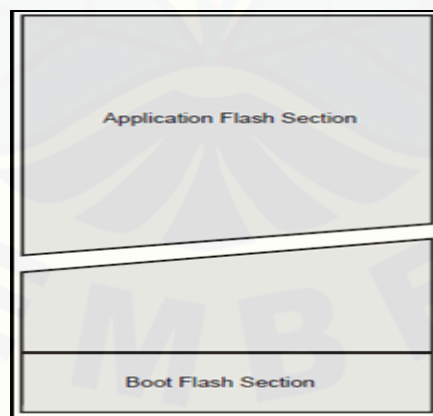
Arduino Uno dapat diprogram dengan perangkat lunak arduino. Pilih Arduino Uno dari *Tool* lalu sesuaikan dengan mikrokontroler yang digunakan. ATmega328 pada *Uno Arduino* memiliki *bootloader* yang memungkinkan untuk meng-*upload* program baru, untuk itu tanpa menggunakan programmer *hardware* eksternal. Ini berkomunikasi menggunakan protokol dari bahasa C. Sistem dapat menggunakan perangkat lunak FLIP Atmel (*Windows*) atau programmer DFU (*Mac OS X dan Linux*) untuk memuat *firmware* baru. Atau dapat menggunakan *header* ISP dengan programmer eksternal.

f. Otomatis Software Reset

Tombol *reset* Arduino Uno dirancang untuk menjalankan program yang tersimpan di dalam mikrokontroler dari awal. Tombol *reset* terhubung ke Atmega328 melalui kapasitor 100 nf. Setelah tombol *reset* ditekan cukup lama untuk me-*reset* chip, *software IDE* Arduino dapat juga berfungsi untuk meng-*upload* program dengan hanya menekan tombol *upload* di *software IDE* Arduino.

2.3.1 ATmega328P

Manajemen memori dalam mikrokontroler penting dilakukan karena memori yang dimiliki mikrokontroler sangat terbatas. Pada Atmega328P terdapat tiga jenis memori, yaitu *data memory*, *program memory*, dan EEPROM. Bus ketiga memori tersebut terpisah, sehingga dapat mengakses ketiga jenis memori tersebut dalam waktu yang bersamaan. ATmega328P menggunakan *Flash Memory* untuk *program memory*. *Flash Memory* dibagi menjadi dua bagian, yaitu *Boot Loader* dan *Application Program*. Pembagian ini bertujuan untuk keamanan perangkat lunak. *Flash Memory* memiliki ketahanan tulis atau hapus sebanyak 10.000 kali. *Program memory map* dapat dilihat pada gambar 2.4 :



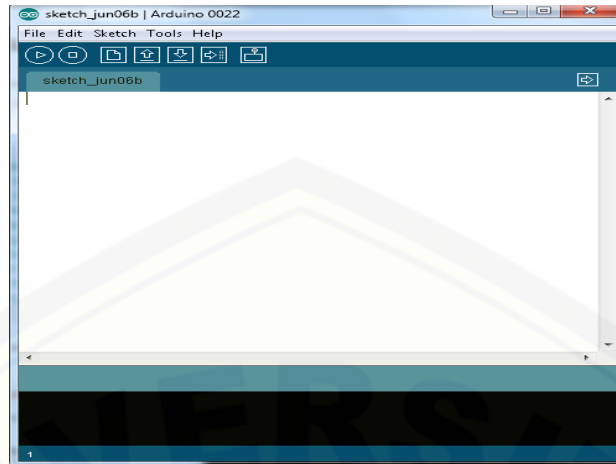
Gambar 2.4 Program *memory map* ATmega328P

Sumber : <http://repository.ipb.ac.id>

SRAM digunakan oleh ATmega328P untuk *data memory*. Kapasitas SRAM dari ATmega328P adalah 2KB. SRAM terbagi menjadi empat bagian yaitu 32 GPR (*General Purpose Register*), 64 I/O *register*, *Additional I/O register*, dan *Internal SRAM*. *Data memory map* memiliki sifat *volatile* sehingga data yang ada pada SRAM akan hilang ketika sudah tidak diberikan catu daya.

2.3.2 Arduino-0022

Arduino-022 merupakan *open-source Arduino environment* yang digunakan untuk penulisan kode. Dengan menggunakan Arduino-022 penulisan kode menjadi mudah dan kode yang ditulis dapat diunggah ke arduino. *Software* ini dapat digunakan di *Windows*, *Mac OS X*, dan *Linux*. *Arduino environment* ditulis dalam bahasa Java dengan didasarkan pada *processing*, *avr-gcc*, dan *open source software* lainnya. Bahasa pemrograman Arduino didasarkan pada bahasa pemrograman C/C++ serta terhubung dengan AVR Libc sehingga dapat menggunakan fungsi-fungsi yang terdapat pada AVR Libc. AVR Libc berisi fungsi-fungsi yang digunakan untuk menggunakan AVR, seperti pengaturan *register*. Pada Arduino-0022 penggunaan AVR Libc dipermudah karena secara *default library* pada Arduino-0022 sudah mencakup AVR Libc tanpa harus tahu AVR Libc mana yang digunakan. Jika dalam penulisan kode membutuhkan AVR Libc, maka penambahan AVR Libc pada *header* kode program dapat dilakukan. Tampilan Ardino-0022 dapat dilihat pada gambar 2.5.



Gambar 2.5 Tampilan Arduino-002
Sumber : <http://repository.ipb.ac.id>

2.3.3 USART

USART (*Universal Synchronous and Asynchronous Serial Receiver and Transmitter*) dapat difungsikan sebagai transmisi data sinkron dan asinkron. Sinkron berarti *clock* yang digunakan antara *transmitter* dan *receiver* satu sumber *clock*. Asinkron berarti *transmitter* dan *receiver* mempunyai sumber *clock* yang berbeda. USART terbagi dalam tiga bagian yaitu *clock generator*, *transmitter*, dan *receiver* (Ardi 2010). Komunikasi serial ini penting dipahami karena selain beberapa *interface* menggunakan mode serial, komunikasi ini bisa dimanfaatkan sebagai pemantau program saat berjalan (*debugging*). Misalkan akan menganalisis suatu kondisi suhu, ini bisa dilihat dari monitor dengan ditransmisikan ke komputer secara serial.

2.3.4 Serial Monitor

Pada Arduino-0022 terdapat *serial monitor*. *Serial monitor* berfungsi untuk menampilkan data yang terdapat pada arduino. Selain itu, melalui *Serial monitor* juga dapat mengirimkan data ke arduino. Mengirim data ke arduino dilakukan dengan cara memasukkan data yang ingin dikirim dan menekan tombol *send* atau menekan *enter*. Tampilan dari *Serial monitor* dapat dilihat pada gambar 2.6. *Baud-rate* pada *serial*

monitor harus sama dengan *Serial.begin* pada kode program. Hal tersebut dilakukan agar terjadi kecocokan dalam komunikasi. *Baud-rate* memiliki satuan *bit per second*.



Gambar 2.6 Serial monitor Arduino
Sumber : www.Arduino.cc

2.3.5 ADC

ADC atau *Analog to Digital Converter* merupakan alat yang digunakan untuk mengubah data *analog* menjadi data *digital*. Pada Arduino Uno sudah terdapat modul ADC, sehingga dapat langsung digunakan. Fitur ADC yang terdapat pada ATmega328P adalah sebagai berikut :

1. Resolusi mencapai 10-bit.
2. 0,5 LSB *Integral Non-linearity*.
3. Akurasi mencapai ± 2 LSB.
4. Waktu konversi 13-260 μ s.
5. Memunyai 6 saluran ADC.
6. *Optional Left Adjustment* untuk pembacaan hasil ADC.
7. 0 - vcc untuk kisaran *input* ADC.
8. Disediakan 1,1V tegangan referensi ADC.
9. Mode konversi kontinyu atau konversi.
10. Interupsi ADC.

11. *Sleep mode noise canceler.*

Sinyal *input* dari pin ADC akan dipilih oleh *multiplexer* (*register ADMUX*) untuk diproses oleh ADC. Karena *converter* ADC dalam *chip* hanya satu buah sedangkan saluran masukannya lebih dari satu, maka dibutuhkan *multiplexer* untuk memilih *input* pin ADC secara bergantian. ADC memunyai rangkaian untuk mengambil sampel dan *hold* (menahan) tegangan *input* ADC, sehingga tetap dalam keadaan konstan selama proses konversi. Sinyal *input* ADC tidak boleh melebihi tegangan referensi. Nilai *digital* sinyal *input* ADC untuk resolusi 10-bit (1024) ditulis dalam persamaan :

$$\text{Kode digital} = (V_{\text{input}} / V_{\text{ref}}) \times 1024(1)$$

2.4 Sensor Ultrasonik

Sensor ultrasonik berfungsi sebagai pendeteksi jarak yang pada aplikasinya untuk mendeteksi jarak kendaraan dengan benda di depannya. Sensor ini mampu mendeteksi jarak dengan *range* 3 cm sampai 3 m. Prinsip kerjanya yaitu pemancar ultrasonik mengeluarkan frekuensi 40 Khz yang dihasilkan oleh mikrokontroler, kemudian diterima oleh pemancar ultrasonik. Dengan menghitung selang waktu antara pengiriman sinyal sampai dapat diterima oleh penerima, dapat diketahui nilai jarak dari kendaraan dan kendaraan sekitarnya. Sensor ultrasonik yang digunakan merupakan modul jadi dari *PARRALAX* dengan tipe SRF04, spesifikasi teknis *Deventech SRF04 Ultrasonic Range Finder*.

Output dari sensor ini berupa data PWM sehingga *duty cycle* dari sinyal *output* berbanding lurus dengan data jarak. Semakin jauh jarak obyek maka semakin besar *duty cycle*. Untuk mengaktifkan sensor maka modul diberi *trigger* pulsa maka sensor akan mengeluarkan sinyal pwm dan *duty cycle* tersebut sebagai jarak obyek dengan sensor. Sensor diberi *trigger* kemudian dihitung lebar *duty cyclenya*. Jika sensor ultrasonik yang digunakan pada satu sistem lebih dari satu maka data yang dikirim secara bergantian. Data yang dikirim adalah data 8-bit dengan nilai 5-225, dimana nilai 0 digunakan sebagai tanda akhir data.

Tabel 2.2 Diskripsi Sensor Ultrasonik :

Tegangan <i>Input</i>	5 VDC
Konsumsi Arus	30 mA (rata-rata), 50 mA (<i>max</i>)
Frekuensi Suara	40 Khz
Jangkauan	3 cm – 3 m
Sensitivitas	Mampu mendeteksi gagang sapu berdiameter 3 cm dalam jarak >2 m
<i>Input Trigger</i>	10 mS min. pulsa level TTL
Pulsa <i>Echo</i>	Sinyal level TTL positif. Lebar disbanding proporsional dengan jarak yang dideteksi

Di bawah ini merupakan tampilan fisik dari rangkaian sensor ultrasonik beserta koneksi-koneksinya :



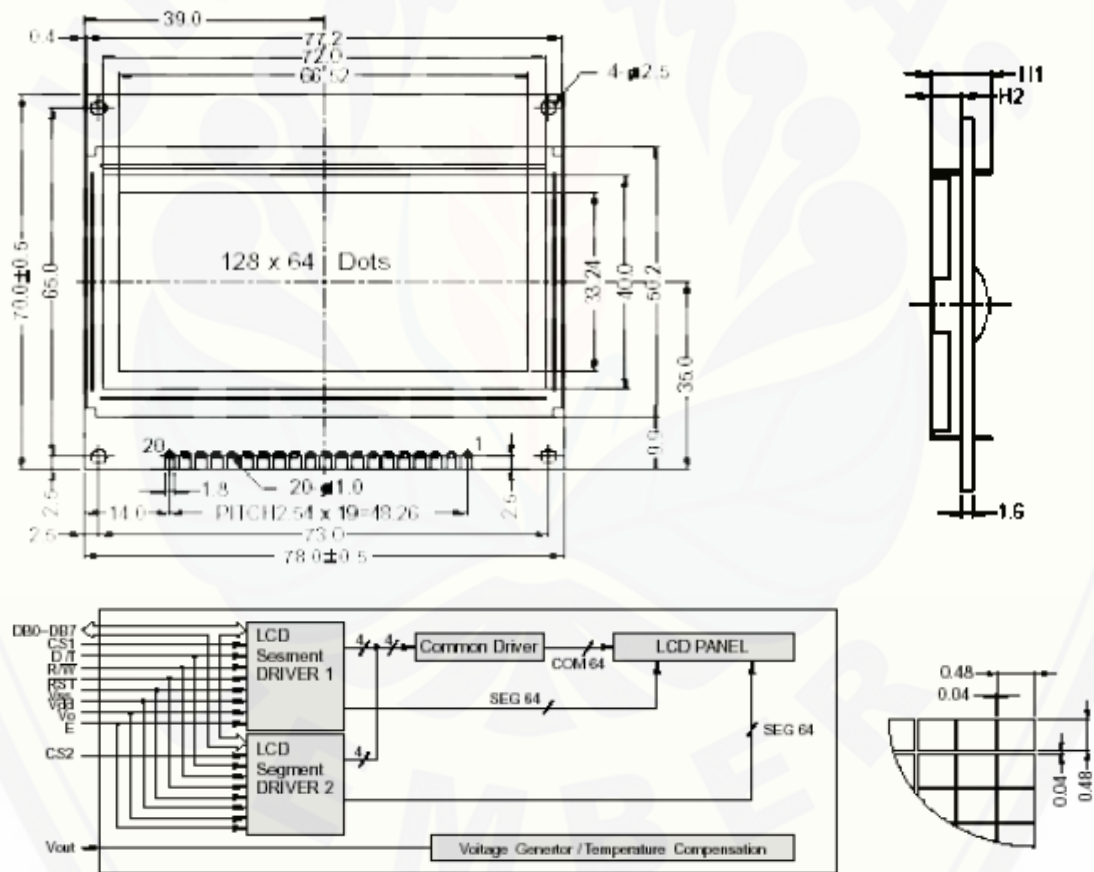
Gambar 2.7 Modul dan Koneksi Sensor Ultrasonik

Sumber : www.tokopedia.com

2.5 LCD Grafik

Komponen *display* data dapat bermacam-macam, mulai dari LED, *seven segment*, LCD (*Liquid Crystal Display*) sampai dengan monitor *analog* maupun *digital*. LCD merupakan komponen *display* yang banyak digunakan. Mulai dari LCD alpha numerik (LCD karakter) ukuran 2x16, 4x40 karakter sampai dengan LCD grafik berbagai ukuran (dalam piksel). LCD dikendalikan oleh IC yang akan mengirimkan sinyal sesuai dengan alamat memori pada layar LCD.

Pada LCD Grafik 128x64 dot digunakan dua IC KS108b *segment driver* (bagian kiri/kanan LCD) dan satu IC KS107 *common driver* yang dikombinasikan untuk *drive* sinyal kontrol ataupun membangkitkan sinyal data pada LCD.



Gambar 2.8 Blok Diagram dan Gambar Kontur LCD Grafik 128x64

Sumber : <http://image.frompo.com>

Tabel 2.3 Diskripsi LCD Grafik :

Pin No.	Symbol	Level	Description
1	V _{ss}	0 V	Ground
2	V _{DD}	5.0 V	Supply voltage for logic
3	V _o	(Variable)	Operating voltage for LCD
4	D/I	H/L	H : Data, L: Instruction
5	R/W	H/L	H:Read(MPU-Module), L:Write(MPU-Module)
6	E	H	Enable Signal
7	DB0	H/L	Data Bit 0
8	DB1	H/L	Data Bit 1
9	DB2	H/L	Data Bit 2
10	DB3	H/L	Data Bit 3
11	DB4	H/L	Data Bit 4
12	DB5	H/L	Data Bit 5
13	DB6	H/L	Data Bit 6
14	DB7	H/L	Data Bit 7
15	CS1	L	Select Column 1 - Column 64
16	CS2	L	Select Column 65 - Column 128
17	RST	L	Reset Signal
18	V _{out}		Negative Voltage
19	A	•	Power Supply for LCD backlight (+)
20	K	•	Power Supply for LCD backlight (-)

Dalam prosesnya, pin-pin tersebut dihubungkan ke mikrokontroler yang akan membangkitkan sinyal untuk memberikan *input* ke IC pada LCD. Secara umum dapat dibagi dalam *port* data (DB0-DB7) dan *port* kontrol yang terdiri atas pin E, R/W, D/I, CS1, CS2, RST.

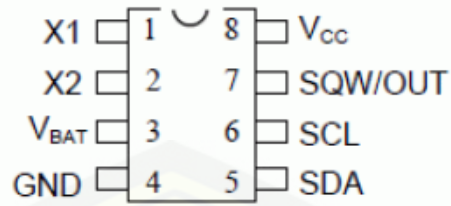
2.6 RTC (*Real-Time Clock*) DS1307

Real-time clock DS1307 adalah IC yang dibuat oleh perusahaan *Dallas Semiconductor*. IC ini memiliki kristal yang dapat mempertahankan frekuensinya dengan baik. *Real-time clock* DS1307 memiliki fitur sebagai berikut :

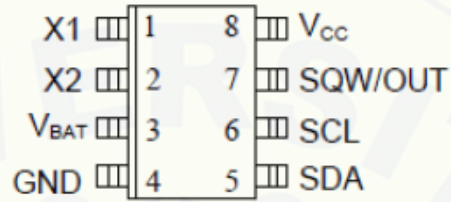
1. *Real-time clock* (RTC) menyimpan data-data detik, menit, jam, tanggal dan bulan dalam seminggu, dan tahun valid hingga 2100.
2. 56-byte, *battery-backed*, RAM *nonvolatile* (NV) RAM untuk penyimpanan.
3. Antarmuka serial *Two-wire* (I2C).
4. Sinyal keluaran gelombang-kotak terprogram (*Programmable squarewave*).
5. Deteksi otomatis kegagalan-daya (*power-fail*) dan rangkaian *switch*.
6. Konsumsi daya kurang dari 500nA menggunakan mode baterai cadangan dengan operasional osilator.
7. Tersedia fitur industri dengan ketahanan suhu: -40°C hingga +85°C.
8. Tersedia dalam kemasan 8-pin DIP atau SOIC.

Sedangkan daftar pin RTC DS1307 adalah sebagai berikut :

1. VCC – *Primary Power Supply*.
2. X1, X2 – 32.768k Hz *Crystal Connection*.
3. VBAT – +3V *Battery Input*.
4. GND – *Ground*.
5. SDA – *Serial Data*.
6. SCL – *Serial Clock*.
7. SQW/OUT – *Square Wave/Output Driver*.



DS1307 8-Pin DIP (300-mil)



DS1307 8-Pin SOIC (150-mil)

Gambar 2.9 Diagram pin RTC DS1307

Sumber : Data Sheet IC Real-Time Clock DS1307

BAB III

METODE PENELITIAN

3.1. Tempat dan Waktu Penelitian

3.1.1 Tempat

Untuk tempat pelaksanaan dilakukan di Laboratorium Dasar dan Optik Jurusan Teknik Elektro Universitas Jember.

3.1.2 Waktu

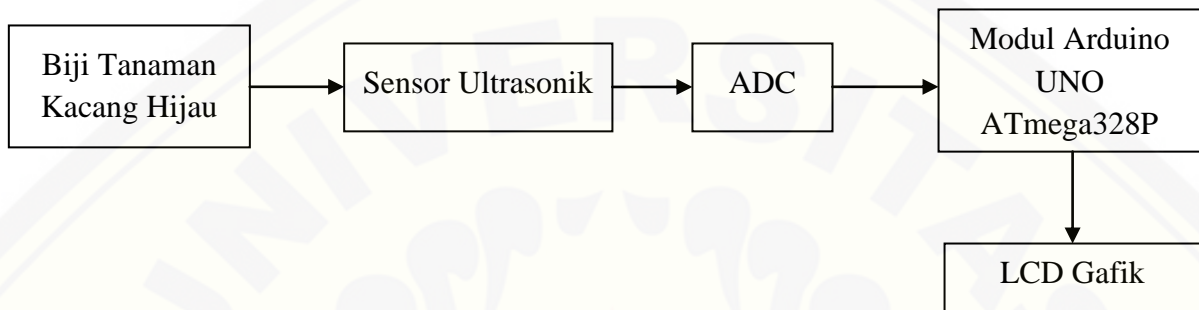
Waktu penelitian dilaksanakan selama 6 bulan mulai bulan Januari sampai bulan Juni 2015.

3.2 Alat dan Bahan

1. Biji Tanaman Kacang Hijau
2. Modul Arduino Uno
3. LCD Grafik
4. Sensor Ultrasonik/jarak
5. Modul RTC
6. *Power Supply*
7. Kabel Pelangi
8. *Push Button*
9. *Acliric*
10. Kapas

3.3 Diagram Blok Sistem

Dalam pembuatan alat ini juga disertakan diagram blok sistem yang diharapkan dapat mempermudah pembuatannya. Secara umum, diagram blok perancangan alat dapat dilihat pada gambar 3.1 di bawah ini:



Gambar 3.1 Diagram Blok Sistem

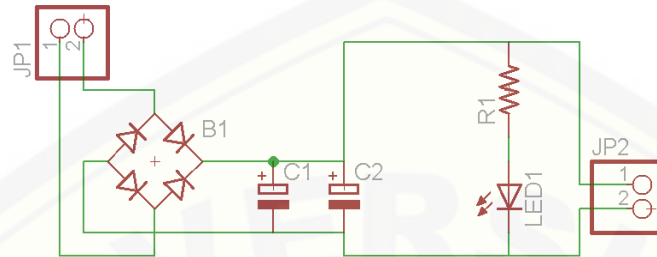
Keterangan dari diagram blok pada gambar 3.1 :

1. Sensor Ultrasonik : dimana sensor ini berfungsi untuk mendeteksi pertumbuhan tinggi biji tanaman kacang hijau.
2. Modul Arduino Uno ATmega328P : dimana mikrokontroler ini digunakan untuk mengendalikan prinsip kerja alat, memberi perintah supaya dapat ditampilkan pada LCD Grafik, sehingga alat ini bekerja seperti yang diinginkan
3. LCD Grafik : digunakan untuk tampilan alat, LCD Grafik akan menampilkan ukuran tinggi biji tanaman kacang hijau dalam satuan centimeter beserta grafik pertumbuhan selama 7 hari.

Sumber tegangan berupa adaptor yang akan menjadi penyuplai tegangan pada alat ini. Alat ini akan terus hidup untuk pengamatan pertumbuhan selama 7 hari.

3.4 Rancangan Alat

3.4.1 Rangkaian *Power Supply*

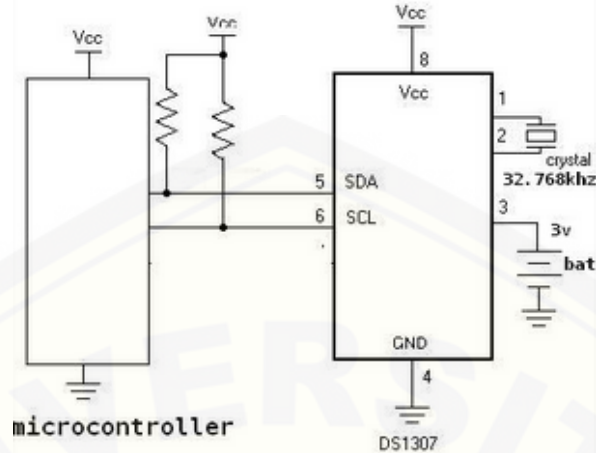


Gambar 3.2 Rangkaian *Power Supply*

Pada rangkain *power supply* ini merupakan sumber tegangan atau VCC, yang digunakan untuk memberikan tegangan pada modul arduino, sensor ultrasonik dan LCD Grafik agar dapat dioperasikan sesuai fungsinya. Tegangan yang dikeluarkan dari *power supply* adalah 9V, masuk ke modul arduino yang sebelumnya telah melewati IC *regulator* 7809 agar tegangan yang dikeluarkan menjadi 9 VDC. Tegangan tersebut nantinya juga akan masuk ke rangkaian lainnya atau keseluruhan rangkaian pada alat ini.

3.4.2 Rangkaian RTC

Rangkaian RTC digunakan untuk menunjukkan waktu dan tanggal pertumbuhan biji kacang hijau setiap harinya sampai 7 hari. Gambar di bawah ini menunjukkan rangkaian RTC menggunakan IC DS1307. Agar dapat bekerja rangkaian ini membutuhkan komponen tambahan yaitu Kristal eksternal, nilai Kristal tersebut adalah 32,768 kHz. Selain itu RTC juga dilengkapi baterai 3,3 V. Fungsi baterai pada RTC sebagai catu daya cadangan apabila catu daya utama mati.



Gambar 3.3 Rangkaian RTC

Pada pin 1 dan pin 2 dihubungkan dengan Kristal eksternal 32,768 KHz. Pin 3 Vbat di hubungkan dengan baterai eksternal 3V. Pin 4 GND dapat hubungkan dengan *ground* pada baterai eksternal. Pin 8 Vcc di hubungkan ke Vcc sistem minimum. Pada pin 5, 6, dan 7 adalah sinyal data dari RTC ke arduino, di antaranya pin 7 adalah SQM, pin 6 adalah SCL dan pin 5 adalah SDA dan dimasing-masing sinyal data tersebut diberikan resistor 1k hingga 10k sebagai *pul up*, kecuali Pin 7 yaitu SQM.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Pengujian Modul Arduino

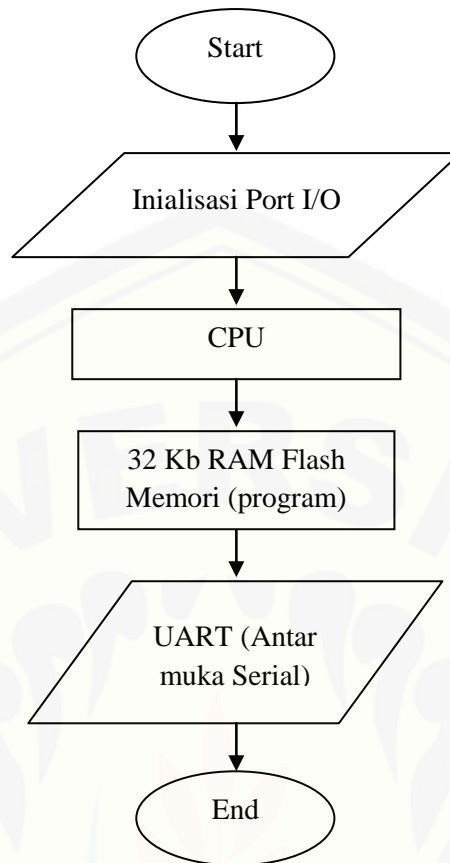
Dalam pembuatan alat pengukur tinggi tanaman ini, modul arduino dibeli di pasaran, Arduino Uno dengan ATmega328P mempunyai 14 *digital input/output* (6 di antaranya dapat digunakan untuk *PWM outputs*), 6 *analog inputs*, 16 MHz *crystal oscillator*, *USB connection*, *power jack*, *ICSP header*, dan *reset button*, pengujian rangkaian modul arduino ini dilakukan dengan cara memasukkan program kedalam mikrokontroler melalui *personal computer*, pengujian dilakukan untuk mengetahui apakah modul arduino dapat bekerja atau tidak.



Gambar 4.1 Modul Arduino Uno

Sumber : <http://www.Arduino.cc>

Untuk mengetahui sistem kerja dari arduino perlu diketahui *flowchart* sistem pada arduino seperti di bawah ini:



Gambar 4.2 *Flowchart* Sistem Arduino

Pada *flowchart* di atas menjelaskan dimulai dari inialisasi port I/O, pin-pin untuk menerima data (input) digital atau analog, dan mengeluarkan data (output) digital atau analog. Kemudian masuk ke CPU (Central Processing Unit) yang berfungsi untuk menjalankan setiap intruksi dari program. Setelah itu RAM Flash Memori yang bersifat non-volatile (hilang saat daya dimatikan) yang berfungsi untuk menyimpan program yang dimuat dari computer. Selain program, flash memori juga juga menyimpan bootloader. Bootloader adalah program inisiasi yang ukurannya kecil, dijalankan oleh CPU saat daya dihidupkan. Setelah bootloader selesai dijalankan, berikutnya program di dalam RAM akan dieksekusi. Untuk yang terakhir UART (*Universal Asynchronus Reseiver/Transmitter*) merupakan antarmuka yang digunakan untuk komunikasi serial.

Pengujian dilakukan dengan mengukur tegangan *output* dengan AVO meter. Pengujian ini dilakukan supaya mengetahui pin-pin arduino dapat bekerja dengan mengukur tegangan output setiap pin. Di bawah ini merupakan tabel data hasil pengukuran pin-pin arduino :

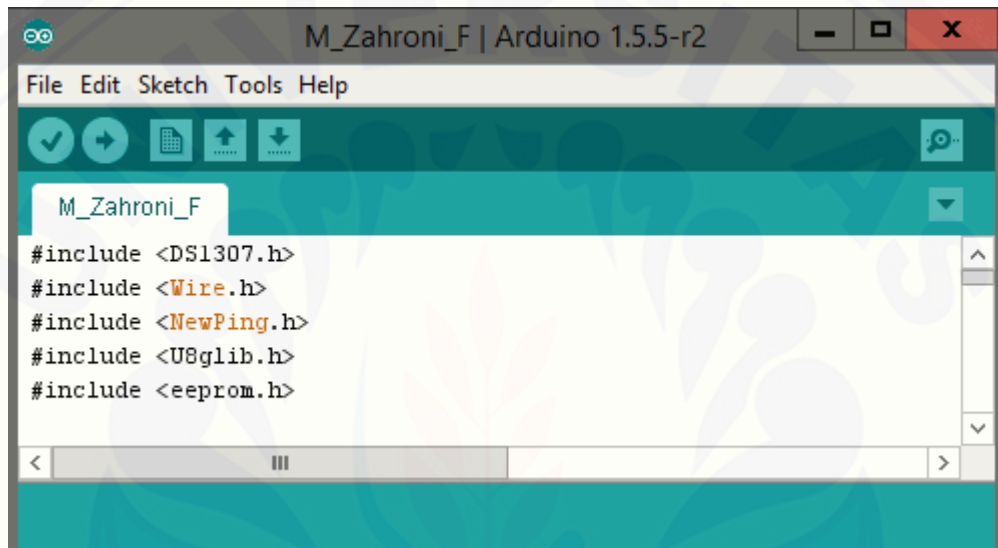
Tabel 4.1 Hasil Pengukuran (V) *Output* Modul Arduino Uno

PIN	Tegangan <i>Output</i>			
	Pengukuran 1	Pengukuran 2	Pengukuran 3	Pengukuran Rata-rata
0	4,90 V	4,91 V	4,92 V	4,91 V
1	4,90 V	4,96 V	4,95 V	4,93 V
2	4,90 V	4,92 V	4,92 V	4,91 V
3	4,93 V	4,93 V	4,94 V	4,93 V
4	4,93 V	4,91 V	4,90 V	4,91 V
5	4,93 V	4,95 V	4,96 V	4,94 V
6	4,93 V	4,91 V	4,96 V	4,93 V
7	4,93 V	4,91 V	4,90 V	4,91 V
8	4,96 V	4,91V	4,94 V	4,93 V
9	4,94 V	4,94 V	4,94 V	4,94 V
10	5,12 V	4,91 V	4,93 V	4,98 V
11	4,90 V	4,91 V	4,93 V	4,91 V
12	5,02 V	4,93 V	4,92 V	4,95 V
13	5,1 V	4,91 V	4,97 V	4,99 V

Tabel di atas merupakan data pengukuran yang dilakukan dengan 3 kali pengukuran kemudian hasilnya dirata-rata. Kemudian tegangan keluaran rata-rata pin-pin arduino 4,91V, 4,93V, 4,91V, 4,91V, 4,91V, 4,94V, 4,93V, 4,91V, 4,91V, 9,93V, 4,94V, 4,98V, 4,91V, 4,95V, 4,99V. Kemudian rangkaian ini dapat dihubungkan dengan rangkaian-rangkaian lainnya untuk mendapatkan kerja sistem

yang baik, dikarenakan tegangan yang didapatkan mendekati 5V. Pada arduino, *board* beroperasi pada tegangan 5V-12V. jika tegangan yang disuplai kurang dari 5V maka arduino akan tidak stabil, dan jika tegangan lebih dari 12V maka regulator tegangan bisa cepat panas dan merusak *board* arduino.

Untuk penulisan program untuk arduino pada *software* IDE Arduino. Pertama-tama memasukkan program *library* untuk LCD Grafik, sensor ultrasonik, RTC, dan penyimpanan data seperti :

The image shows a screenshot of the Arduino IDE interface. The title bar reads "M_Zahroni_F | Arduino 1.5.5-r2". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". Below the menu bar is a toolbar with icons for saving, running, and uploading. The main text area shows the following code:

```
#include <DS1307.h>
#include <Wire.h>
#include <NewPing.h>
#include <U8glib.h>
#include <EEPROM.h>
```

Gambar 4.3 *Library* Program pada Arduino-1.5.5

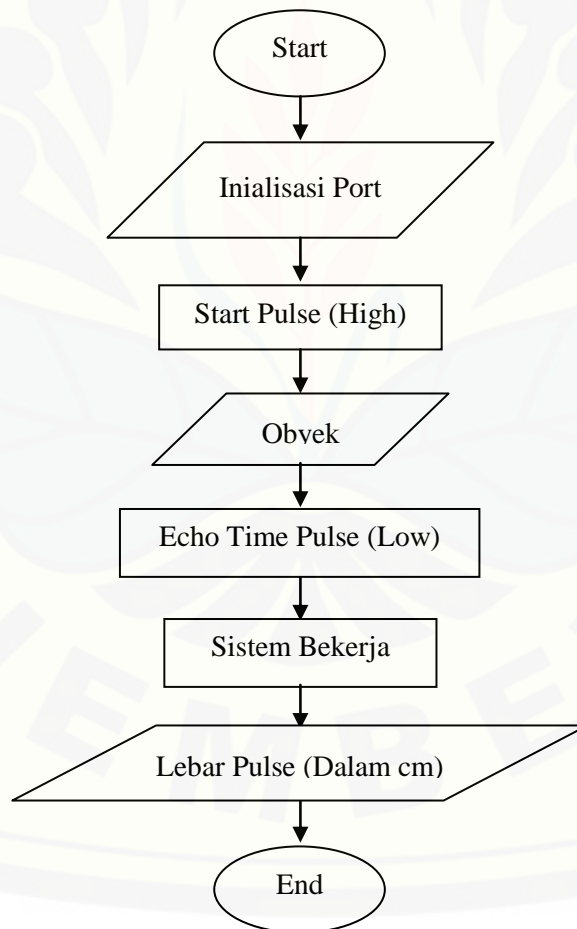
Setelah memasukkan *library* dari komponen yang dibutuhkan, kemudian dilanjutkan untuk memasukkan program-program perintah masing-masing komponen supaya alat dapat bekerja dengan baik.

Adapun juga port pengujian untuk melihat dimana komponen tidak bermasalah dan dapat bekerja sesuai yang diharapkan. Untuk yang pertama menguji modul arduino seperti yang hasilnya seperti di bawah ini :

Tabel 4.2 Port Pengujian Arduino Uno

No	Port Pengujian	Hasil Pengujian	
		Berhasil	Gagal
1.	Sensor Ultrasonik	√	-
2.	Tombol Menu (Push Button)	√	-
3.	Rangkaian RTC	√	-
4.	LCD Grafik	√	-

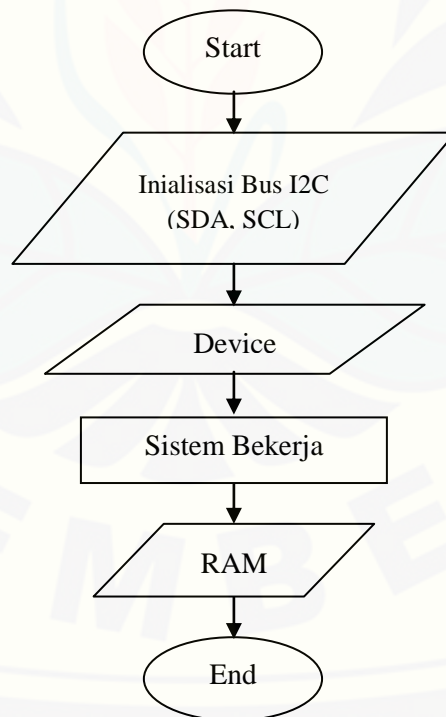
Pada sensor ultrasonik dapat dapat dijelaskan dengan melihat *flowchart* sistem sensor ultrasonik di bawah ini:



Gambar 4.4 *Flowchart* Sistem Sensor Ultrasonik

Pada dasarnya sensor ultrasonik terdiri dari sebuah chip pembangkit sinyal 40KHz, sebuah pemancar ultrasonik dan sebuah penerima ultrasonik. Pemancar mengubah sinyal 40KHz menjadi suara sementara, penerima berfungsi untuk mendeteksi pantulan suaranya. Jadi sensor ini bekerja mendeteksi obyek di depannya dengan cara mengirimkan suara ultrasonik dan kemudian menerima pantulan suara tersebut. Sensor hanya akan mengirimkan suara ultrasonik ketika ada pulsa trigger dari mikrokontroler (high pulse selama 5 μ S). Suara Ultrasonik dengan frekuensi sebesar 40KHz akan dipancarkan selama 200 μ S. Suara ini akan merambat di udara dengan kecepatan 344.424m/detik (atau 1cm setiap 29.034 μ S), mengenai objek untuk kemudian terpantul kembali ke sensor. Selama menunggu pantulan, sensor akan menghasilkan sebuah pulsa. Pulsa ini akan berhenti (low) ketika suara pantulan terdeteksi oleh sensor. Terdapat 4 pin sensor ultrasonik yaitu Vcc, Gnd, Trigg dan Echo. Kemudian kita dapat mengetahui sensor bekerja dengan mengukur pin Trigg dan Echo dengan AVO meter.

Untuk rangkaian RTC dapat dilihat dengan *flowchart* sistem RTC :



Gambar 4.5 *Flowchart* Sistem RTC

Rangkaian RTC beroperasi pada komunikasi serial I2C, yaitu pin SDA dan SCL. Cara kerja dengan mengirimkan sinyal start diikuti dengan device address dan alamat sebuah register yang akan dibaca. Beberapa register dapat dibaca sampai stop condition dikirim. data waktu dan tanggal tersimpan dalam memori RAM masing-masing 1 byte. Pembuktian rangkaian RTC dapat bekerja pada pin arduino yaitu dengan terbacanya tanggal dan waktu yang dapat dilihat pada tampilan LCD :

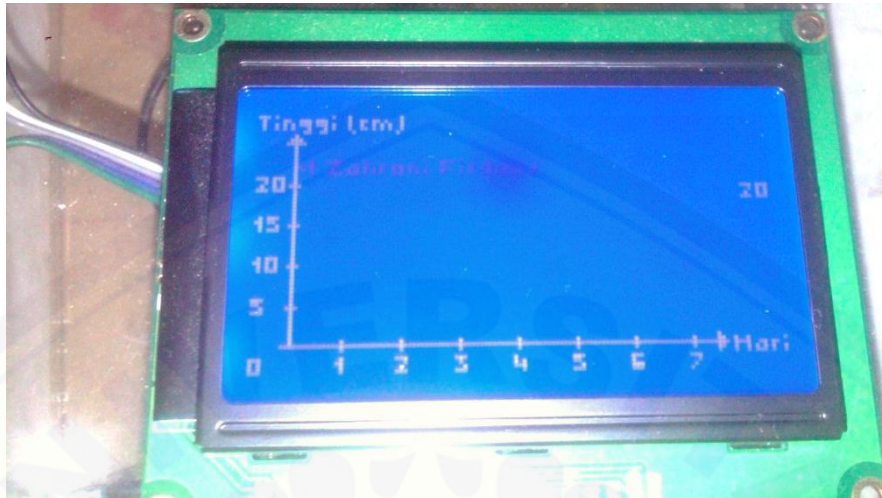


Gambar 4.6 Tampilan Waktu dan Tanggal pada LCD

Port pengujian di atas adalah bahwa setiap pin-pin yang dibutuhkan oleh komponen-komponen untuk sensor ultrasonik, tombol menu, rangkaian RTC, dan LCD Grafik dapat bekerja. Pada sensor ultrasonik dikatakan berhasil dikarenakan sensor bekerja dan dapat mengukur tinggi obyek. Kemudian untuk tombol menu juga dikatakan berhasil karena pada tombol menu yang terdiri dari 5 tombol yaitu menu, *up*, *down*, *ok*, dan *cancel*, juga dapat bekerja ketika tombol ditekan. Tombol ini dapat ditunjukkan pada tampilan LCD Grafik sebelum melakukan pengujian. Untuk berikutnya, rangkaian RTC dikatakan berhasil karena dapat dilihat juga pada tampilan LCD Grafik yang menunjukkan jam dan tanggal yang terus berjalan walaupun ketika alat dimatikan. Dan untuk LCD Grafik dapat dikatakan berhasil dikarenakan LCD dapat bekerja dengan menampilkan tinggi obyek dari sensor ultrasonik, menampilkan pilihan menu untuk tombol menu, tampilan jam dan tanggal dari RTC dan grafik pengukuran dari keseluruhan alat.

Jadi dari data tabel 4.2 dapat dilihat bahwa modul arduino bekerja pada masing-masing bagian yang diperlukan, dengan demikian rangkaian ini dapat dihubungkan pada rangkaian lainnya untuk mendapatkan kerja sistem yang baik.

4.2 Pengujian LCD Grafik



Gambar 4.7 LCD Grafik

Pengujian untuk LCD Grafik dilakukan dengan cara mengukur tegangan keluaran dari masing-masing pin pada LCD Grafik. Di bawah ini merupakan tabel pengukuran tegangan masing-masing pin:

Dari tabel di bawah ini merupakan pengukuran tegangan keluaran dari pin-pin LCD grafik yang pengukurannya dilakukan beberapa kali. Di bawah ini merupakan pengukuran sampai 3 kali kemudian hasil pengukuran dirata-rata. Pengukuran dilakukan beberapa kali dikarenakan supaya mendapati hasil tegangan keluaran yang akurat. Jadi jika pengukuran dilakukan hanya sekali dapat nantinya ditakutkan alat ini tidak bekerja dengan maksimal. Dengan melihat hasil pengukuran dari tabel dibawah ini menunjukkan LCD dapat bekerja sesuai dengan kebutuhannya. Kemudian dapat dihubungkan dengan rangkaian lainnya.

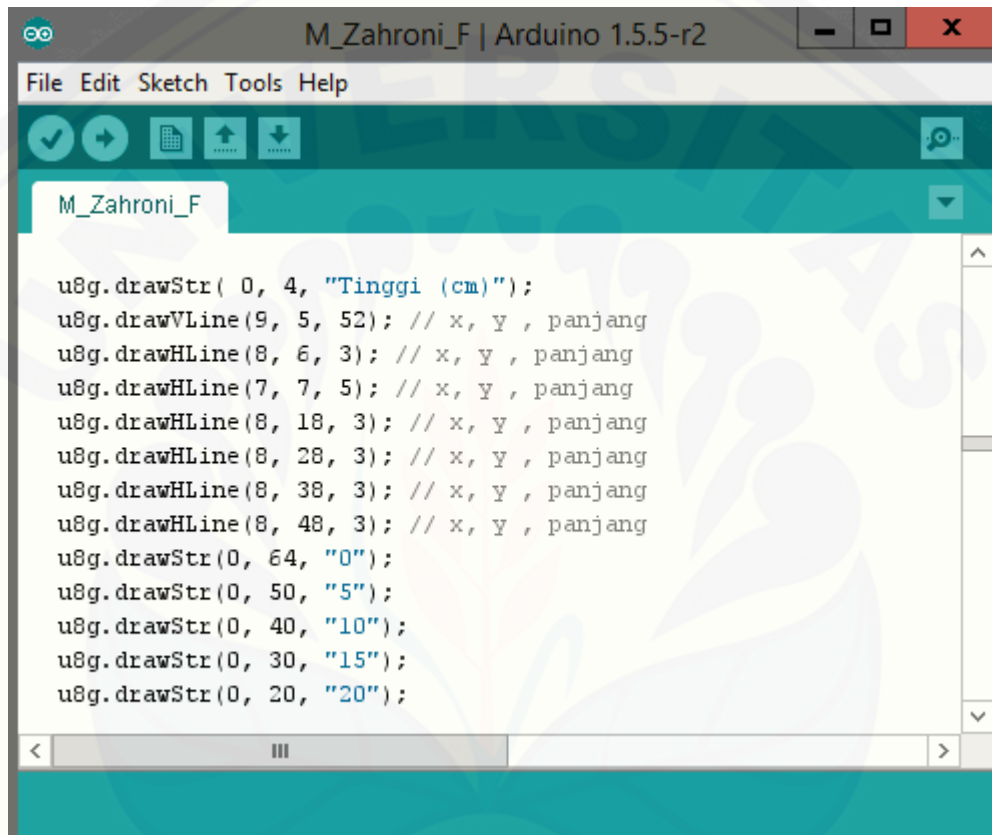
Tabel 4.3 Hasil Pengukuran (V) *Output* LCD Grafik

PIN	Tegangan <i>Output</i>			
	Pengukuran 1	Pengukuran 2	Pengukuran 3	Pengukuran Rata-rata
BLK	2,01 V	2,1 V	2,21 V	2,1 V
BLA	5,03 V	5 V	5,3 V	5,11 V
<i>Vout</i>	0 V	0 V	0 V	0 V
RST	5,2 V	5,11 V	5,25 V	5,18 V
NC	0 V	0 V	0 V	0 V
PSB	0 V	0 V	0 V	0 V
DB7	2,9 V	2,88 V	2,87 V	2,88 V
DB6	2,9 V	2,91 V	2,86 V	2,89 V
DB5	2,9 V	2,93 V	2,86 V	2,89 V
DB4	2,9 V	2,89 V	2,88 V	2,89 V
DB3	2,9 V	2,89 V	2,84 V	2,87 V
DB2	2,9 V	2,85 V	2,89 V	2,88 V
DB1	2,9 V	2,82 V	2,91 V	2,87 V
DB0	2,9 V	2,93 V	2,89 V	2,90 V
E	5,6 V	5,55 V	5,71 V	5,62 V
R/W	0 V	0 V	0 V	0 V
RS	0 V	0 V	0 V	0 V
<i>Vo</i>	0 V	0 V	0 V	0 V
VDD	0 V	0 V	0 V	0 V
VSS	0 V	0 V	0 V	0 V

Pengujian pada LCD berfungsi untuk melihat proses dalam pengujian sensor ultrasonik. Untuk mengetahui kinerja dari LCD tersebut maka dilakukan pengujian dilakukan dengan menghubungkan pin Lcdpin Pin , Db4, Db5, Db6 , Db7 , E, Rs

menuju pin-pin digital modul arduino yaitu pin 10, pin 11, pin 13, Vcc dan Gnd, LCD *display 128x64* pada alat ini digunakan untuk menampilkan grafik pertumbuhan biji tanaman kacang hijau.

Supaya dapat menampilkan grafik, perlu memasukkan program seperti di bawah ini :

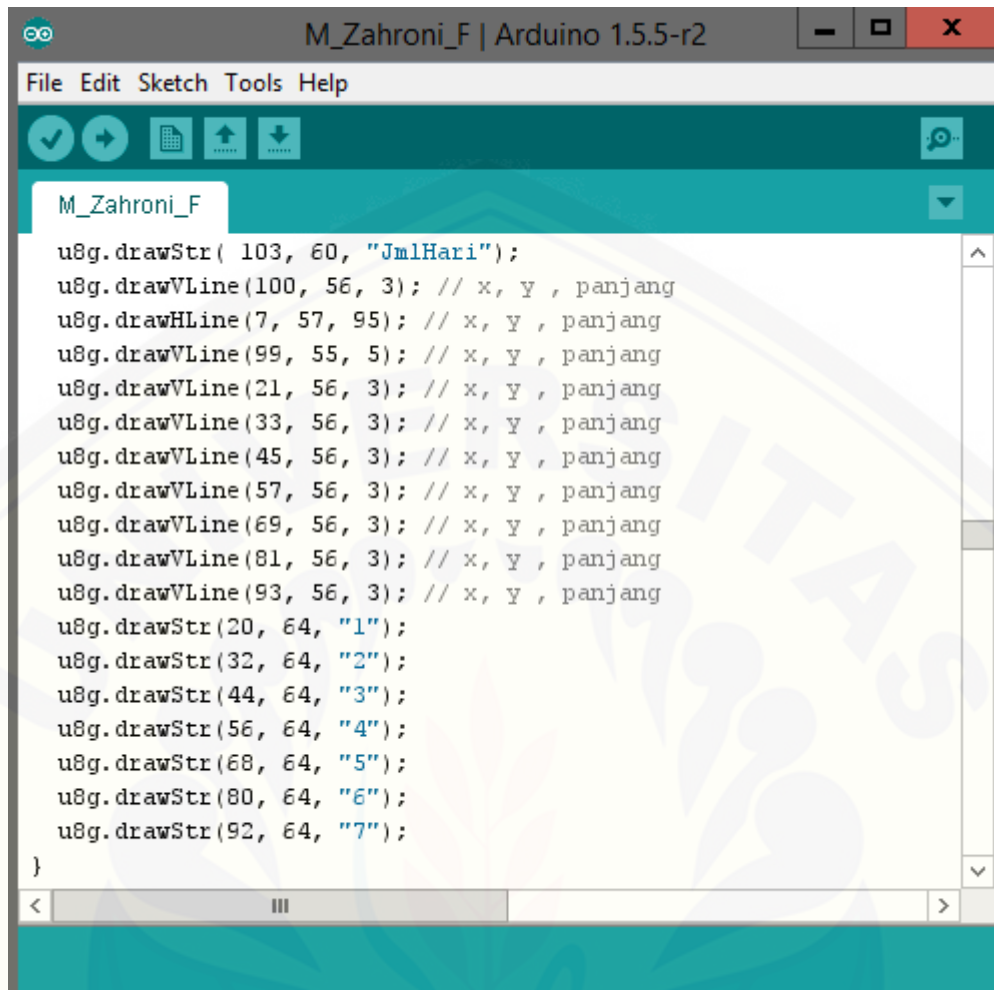


```
u8g.drawStr( 0, 4, "Tinggi (cm)");
u8g.drawVLine(9, 5, 52); // x, y , panjang
u8g.drawHLine(8, 6, 3); // x, y , panjang
u8g.drawHLine(7, 7, 5); // x, y , panjang
u8g.drawHLine(8, 18, 3); // x, y , panjang
u8g.drawHLine(8, 28, 3); // x, y , panjang
u8g.drawHLine(8, 38, 3); // x, y , panjang
u8g.drawHLine(8, 48, 3); // x, y , panjang
u8g.drawStr(0, 64, "0");
u8g.drawStr(0, 50, "5");
u8g.drawStr(0, 40, "10");
u8g.drawStr(0, 30, "15");
u8g.drawStr(0, 20, "20");
```

Gambar 4.8 Program pada Arduino-1.5.5

Program di atas akan menampilkan garis horizontal untuk menunjukkan tinggi biji tanaman kacang hijau. Karena pertumbuhan biji kacang hijau yang cepat setelah menjadi kecambah maka di LCD Grafik hanya akan tampil angka-angka 0, 5, 10, 15, dan 20.

Setelah itu memrogram untuk garis vertikal supaya dapat menjadi grafik yang diinginkan. Seperti pada gambar di bawah ini :

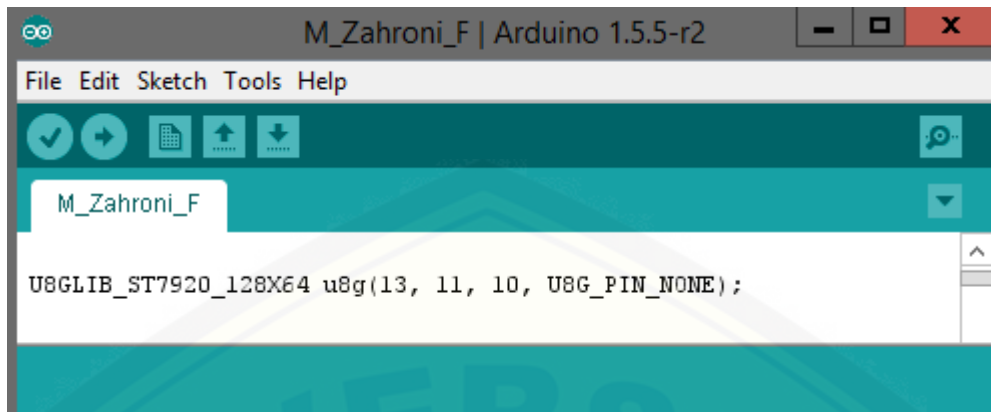


```
u8g.drawStr( 103, 60, "JmlHari");
u8g.drawVLine(100, 56, 3); // x, y , panjang
u8g.drawHLine(7, 57, 95); // x, y , panjang
u8g.drawVLine(99, 55, 5); // x, y , panjang
u8g.drawVLine(21, 56, 3); // x, y , panjang
u8g.drawVLine(33, 56, 3); // x, y , panjang
u8g.drawVLine(45, 56, 3); // x, y , panjang
u8g.drawVLine(57, 56, 3); // x, y , panjang
u8g.drawVLine(69, 56, 3); // x, y , panjang
u8g.drawVLine(81, 56, 3); // x, y , panjang
u8g.drawVLine(93, 56, 3); // x, y , panjang
u8g.drawStr(20, 64, "1");
u8g.drawStr(32, 64, "2");
u8g.drawStr(44, 64, "3");
u8g.drawStr(56, 64, "4");
u8g.drawStr(68, 64, "5");
u8g.drawStr(80, 64, "6");
u8g.drawStr(92, 64, "7");
}
```

Gambar 4.9 Program pada Arduino-1.5.5

Gambar di atas merupakan program yang nantinya menjadi garis vertikal yang akan ditampilkan di LCD Grafik.

Untuk menguji digunakan LCD *library* yang terdapat pada software Arduino 1.5.5 program tersebut lalu *diload* ke modul yang sudah terkoneksi dengan LCD Grafik. Berikut ini merupakan tampilan dari *library* LCD Grafik yang terdapat pada *software* Arduino 1.5.5 :



Gambar 4.10 *Library* LCD Grafik pada Arduino 1.5.5

Lalu program tersebut *diload* ke modul dan akan menghasilkan karakter kata, garis dan lain-lain dalam LCD yang diinginkan. Seperti pada gambar 4.10 di bawah ini:



Gambar 4.11 Tampilan LCD Grafik

Berdasarkan gambar 4.11 LCD Grafik 128x64 ini dapat bekerja untuk dapat menunjukkan grafik pengukuran nantinya.

4.3 Pengujian Sensor Ultrasonik

Modul Ultrasonik biasanya berbentuk papan elektronik berukuran kecil dengan rangkaian elektronik dan 2 buah transducer di dalamnya. Dari 2 buah *transducer* ini, salah satu berfungsi sebagai *transmitter* (pemancar) dan satu lagi sebagai *receiver* (penerima). Ada juga modul ultrasonik yang hanya mempunyai 1 buah *transducer*, *transducer* ini berfungsi sebagai *transmitter* dan *receiver* sekaligus. Pin yang tersedia pada modul ini yaitu pin VCC, TRIGGER, ECHO dan GND. Ada juga modul yang pin TRIGGER dan ECHO-nya digabung menjadi satu dan pemakaiannya dilakukan secara bergantian.

Modul Ultrasonik bekerja dengan cara menghasilkan gelombang suara pada frekuensi tinggi kemudian dipancarkan oleh bagian *transmitter*. Pantulan gelombang suara tersebut akan mengenai benda di depannya yang nantinya akan ditangkap oleh bagian *receiver*. Dengan mengetahui lamanya waktu antara dipancarkannya gelombang suara sampai ditangkap kembali, kita dapat menghitung jarak benda yang ada di depan modul ultrasonik.

Pengukur tinggi biji tanaman kacang hijau ini menggunakan sensor ultrasonik HC SR-04. Untuk Spesifikasi dari modul ini antara lain *power supply* : 5V DC, *quiescent current* : < 2mA, *effectual angle* : < 15°, *ranging distance* : 2cm – 500 cm, *resolution* : 1 cm, *ultrasonic Frequency* : 40 kHz. Di bawah ini adalah bentuk fisik dari sensor ultrasonik dengan tipe HC SR-04 yang ditunjukkan pada gambar 4.12 di bawah ini :

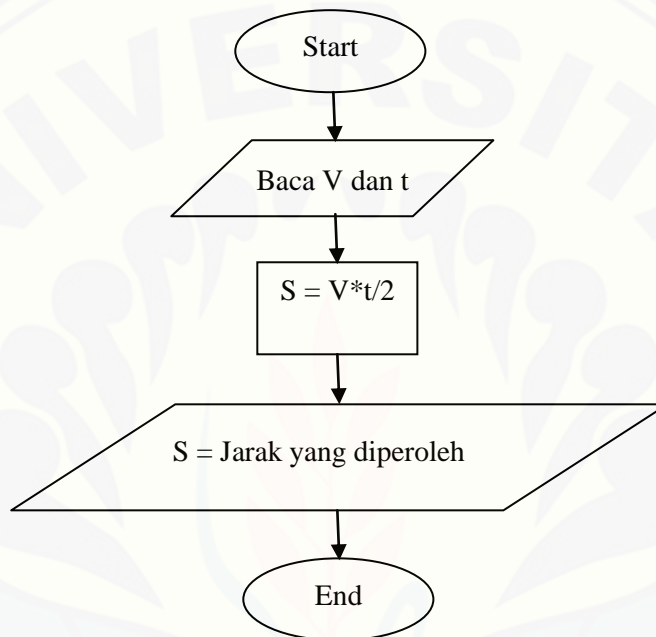


Gambar 4.12 Sensor Ultrasonik

Sumber : www.electrodragon.com

Pada gambar 4.12 merupakan bentuk fisik dari sensor jarak atau *ultrasonic*. Dalam proyek akhir ini sensor jarak digunakan untuk mendeteksi pertumbuhan tinggi biji tanaman kacang hijau. Terdapat 4 terminal pada sensor jarak yaitu *VCC*, *Ground*, *Trigger*, dan *Echo*.

Untuk mengetahui jarak dari sensor ultrasonic dapat dihitung menggunakan rumus atau dapat dilihat pada *flowchart* dibawah ini:



Gambar 4.13 *Flowchart* Perhitungan Sensor Ultrasonik

Pada gambar di atas merupakan langkah-langkah untuk mengetahui jarak dari suatu benda dengan sensor ultrasonik. S merupakan jarak yang diperoleh, kemudian V merupakan kecepatan gelombang yang dipancarkan sensor ultrasonik, dan t merupakan waktu yang ditempuh oleh gelombang ultrasonik sampai ke obyek benda.

Berikut ini merupakan tabel pengukuran tegangan keluaran dari sensor dengan jarak-jarak yang sudah ditentukan:

Tabel 4.4 Pengujian pada Sensor Ultrasonik

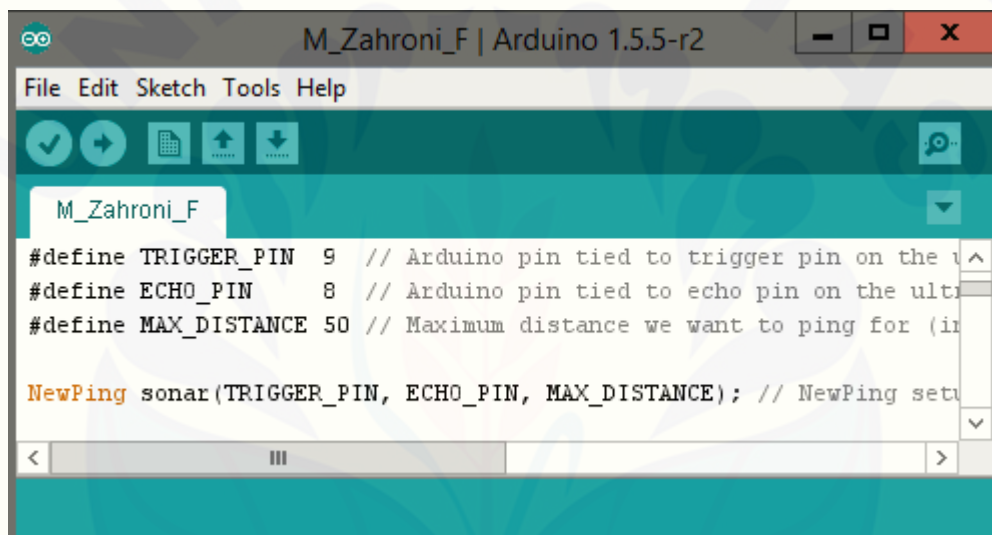
No	Jarak Obyek	Tegangan Output Sensor	Mendeteksi Obyek
1	1 cm	0,02 V	Terdeteksi
2	2 cm	0,04 V	Terdeteksi
3	4 cm	0,08 V	Terdeteksi
4	6 cm	0,12 V	Terdeteksi
5	8 cm	0,16 V	Terdeteksi
6	10 cm	0,2 V	Terdeteksi
7	12 cm	0,24 V	Terdeteksi
8	14 cm	0,28 V	Terdeteksi
9	16 cm	0,32 V	Terdeteksi
10	18 cm	0,36 V	Terdeteksi
11	20 cm	0,4 V	Terdeteksi
12	22 cm	0,44 V	Terdeteksi
13	24 cm	0,48 V	Terdeteksi
14	26 cm	0,52 V	Terdeteksi
15	28 cm	0,56 V	Terdeteksi
16	30 cm	0,6 V	Terdeteksi

Dari tabel di atas menunjukkan bahwa semakin jauh jarak obyek dengan sensor ultrasonik maka semakin besar pula tegangan keluaran yang diperoleh. Kemudian juga dapat dilihat bahwa nilai tegangan menunjukkan 20 mV/cm.

Dari tabel 4.4 di atas dapat disimpulkan bahwa sensor ultrasonik dapat mendeteksi obyek benda dari jarak 1 cm sampai 30 cm. Dapat dikatakan sensor berfungsi dengan baik atau normal, dikarenakan jarak pada sensor sama dengan jarak yang ada pada alat ukur yaitu mistar. Kemudian dari pengujian tersebut dapat dikatakan sensor dapat digunakan sesuai dengan yang di perlukan pada proyek ini yang nantinya akan di terapkan pada pengukur tinggi biji tanaman kacang hijau. Pada sensor ultrasonik ini hanya dapat mendeteksi jarak maksimal yaitu 300 cm. Tapi dalam proyek akhir ini jarak maksimal yang digunakan adalah kurang lebih 30 cm.

Pengujian ini juga ditampilkan pada LCD Grafik berupa nilai tinggi dalam centimeter. Pengujian ini dilakukan dengan cara mengkalibrasi dengan alat ukur yaitu mistar dengan ketinggian 30 cm. Sensor diletakkan di atas dengan mistar di samping sensor sebagai perbandingan. Kemudian dari bawah diletakkan obyek benda dengan ketinggian mulai dari 0 sampai 20 cm. Pada tabel di bawah ini menjelaskan bahwa jarak obyek benda yang terdeteksi oleh sensor ultrasonik.

Supaya dapat bekerja dengan baik perlu *library* untuk sensor ultrasonik, seperti di bawah ini :



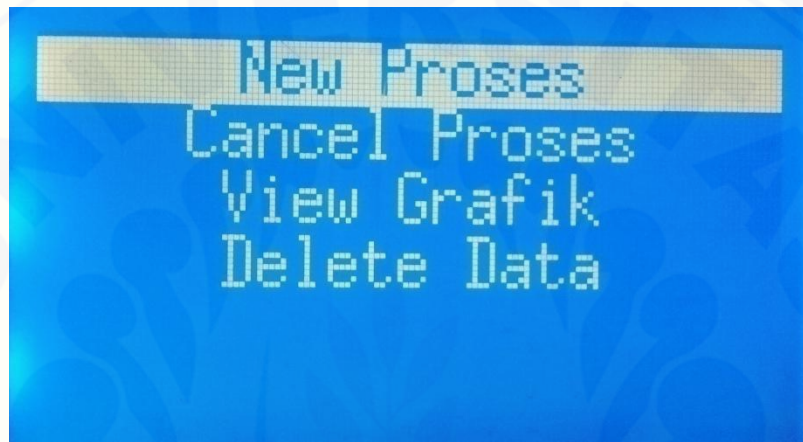
```
#define TRIGGER_PIN 9 // Arduino pin tied to trigger pin on the u
#define ECHO_PIN 8 // Arduino pin tied to echo pin on the ult
#define MAX_DISTANCE 50 // Maximum distance we want to ping for (in
NewPing sonar(TRIGGER_PIN, ECHO_PIN, MAX_DISTANCE); // NewPing setu
```

Gambar 4.14 *Library* Sensor Ultrasonik pada Arduino-1.5.5

Dalam pengujian kali ini, data yang akan diambil yaitu jarak yang dapat dideteksi oleh sensor ultrasonik yang akan diukur mulai dari jarak 1 cm sampai dengan 30 cm. Gelombang ultrasonik yang dipancarkan oleh *transmitter* telah dipantulkan terlebih dahulu sebelum bisa dideteksi oleh *Firmware* dari sensor ultrasonik itu sendiri.

4.4 Data Pengamatan Pertumbuhan Biji Kacang Hijau

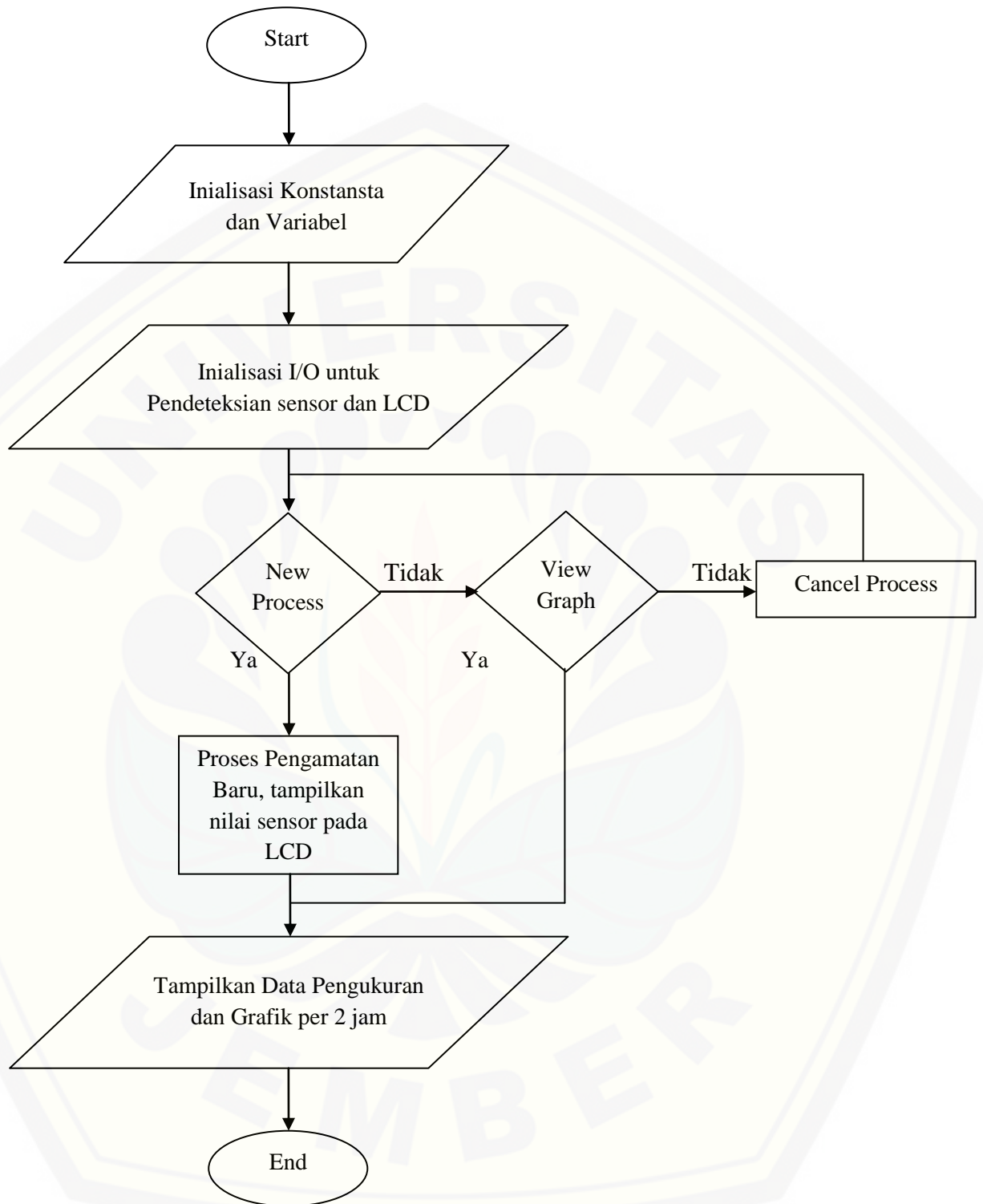
Untuk melakukan pengamatan pada biji kacang hijau perlu memperhatikan langkah-langkahnya. Untuk pertama, pada alat ini terdapat 5 buah tombol menu, di antaranya menu, *up*, *down*, *ok*, dan *cancel*. Kemudian terdapat 4 menu pilihan yang ditampilkan pada LCD Grafik yaitu *new process*, *cancel process*, *view graph*, *delete data*. Seperti pada tampilan di bawah ini :



Gambar 4.15 Tampilan Menu pada LCD Grafik

Fungsi-fungsi dari menu di atas, *new process* untuk memulai melakukan pengamatan pertumbuhan tanaman biji kacang hijau, kemudian *cancel process* untuk membatalkan pengamatan pada saat sedang berlangsung. Selanjutnya *view graph* untuk melihat grafik pertumbuhan dan *delete data* untuk menghapus data pengamatan yang tersimpan.

Adapun langkah-langkah kerja sistem alat ini yaitu ditampilkan dalam bentuk *flowchart* di bawah ini:



Gambar 4.16 *Flowchart* Sistem Alat Keseluruhan

Pada gambar 4.16 di atas merupakan langkah-langkah program perintah pada alat pengukur tinggi tanaman ini sampai dengan menampilkan grafik pada LCD Grafik. Program akan membaca nilai variabel dan konstanta dari sensor ultrasonik dan LCD Grafik, kemudian pada alat akan disediakan menu pilihan untuk proses baru, melihat grafik pertumbuhan dan untuk membatalkan proses. Ketika memilih untuk melakukan proses baru maka arduino akan bekerja dan kemudian akan menampilkan nilai jarak dari sensor yang program kalibrasinya sudah ada pada *library* sensor ultrasonik pada *software* arduino. Kemudian akan ditampilkan pada LCD Grafik berupa grafik pertumbuhan.

Sistem bekerja apabila mendapati masukan input dari sensor dan LCD Grafik, dari masukan tersebut diubah oleh ADC (*Analog to Digital Converter*) dan masuk ke modul arduino maka system bekerja. Setelah itu akan ditampilkan pada LCD Grafik nilai ukur dari sensor dimana untuk alat ini merupakan alat ukur tinggi, jadi tinggi aktual akan tampak pada layar LCD beserta grafiknya.

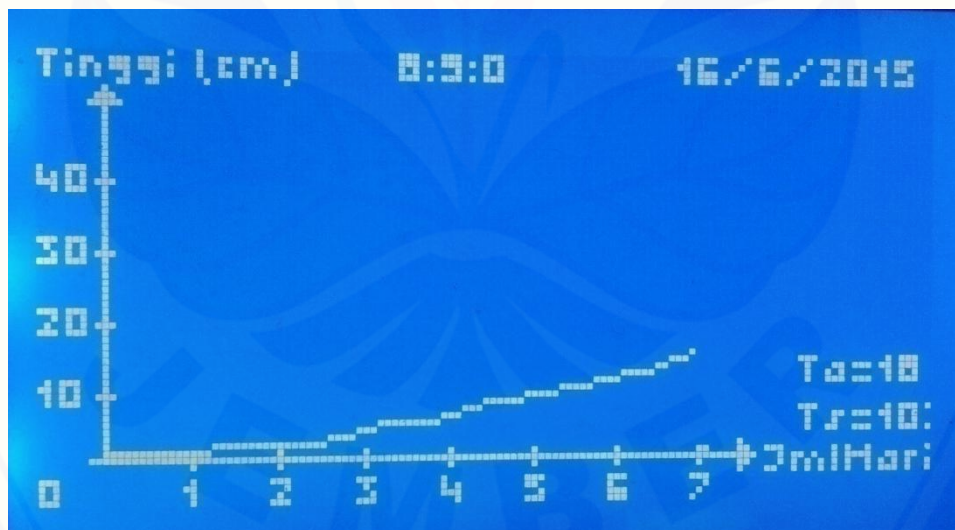
Pengukur tinggi biji tanaman kacang hijau ini dirancang dan dikonstruksikan dalam beberapa bagian utama yang mendukung operasional kerjanya, seperti halnya kerangka. Untuk LCD Grafik difungsikan sebagai penampil tinggi tanaman dari pengamatan pertumbuhan biji kacang hijau yang telah dideteksi oleh sensor ultrasonik/jarak dan dikontrol oleh Arduino Uno.

Pada pembuatan kerangka pengukur tinggi tanaman ini dibuat dari bahan *acrylic* tembus pandang yang dapat kita lihat rangkaian kontrolnya dari luar. Berbentuk seperti *box* yang kemudian di depannya ditempelkan sensor jarak/ultrasonik sebagai pendeteksi obyek yaitu tanaman biji kacang hijau. Di dalam *box* tersebut terdapat modul Arduino Uno, rangkaian *power supply*, dan rangkaian RTC. Serta penutup *box* tersebut ditempelkan LCD grafik sebagai penampil. Rangkaian RTC (*Real Time Clock*) digunakan sebagai penghitung waktu supaya dapat dihasilkan grafik pertumbuhan di LCD Grafik. Berikut ini tabel hasil pengamatan biji kacang hijau selama 7 hari.

Tabel 4.5 Pengamatan Pertumbuhan Biji Kacang Hijau

Hari	Tanggal	Jam	Tinggi (cm)
1	10 Juni 2015	08:08	0
2	11 Juni 2015	08:09	2
3	12 Juni 2015	08:08	5
4	13 Juni 2015	08:10	8
5	14 Juni 2015	08:09	12
6	15 Juni 2015	08:09	15
7	16 Juni 2015	08:09	18

Dari data di atas dapat disimpulkan bahwa pertumbuhan biji kacang hijau akan lebih cepat pada saat biji sudah tumbuh menjadi kecambah. Pada saat masih dalam biji, pertumbuhannya sangat lambat dikarenakan faktor-faktor internal maupun eksternal. Kemudian dari data tersebut juga ditampilkannya grafik pertumbuhan tanaman biji kacang hijau pada LCD Grafik seperti di bawah ini :



Gambar 4.17 Grafik Pertumbuhan Biji Kacang Hijau Selama 7 hari

Pada grafik di atas menunjukkan bahwa pada saat tanaman kacang hijau dalam keadaan biji, pertumbuhannya lambat. Hal itu dapat ditunjukkan pada grafik yang konsisten pada hari pertama, kedua dan ketiga. Pada hari ketika pertumbuhan mulai berjalan cepat dikarenakan tanaman kacang hijau sudah mulai tumbuh menjadi kecambah. Setelah itu dapat dilihat pada gambar 4.17 grafik keseluruhan selama 7 hari pada LCD Grafik.

Tabel 4.6 Data Pengamatan Kedua

Hari	Tanggal	Jam	Tinggi (cm)
1	1 Juli 2015	07:01	0
2	2 Juli 2015	07:01	2
3	3 Juli 2015	07:03	4
4	4 Juli 2015	07:02	7
5	5 Juli 2015	07:03	12
6	6 Juli 2015	07:03	16
7	7 Juli 2015	07:05	20

Tabel 4.7 Data Error % Pertumbuhan Selama 7 hari Pengamatan Pertama

Hari	Pengamatan Pertama		E%
	Pengukuran dengan Alat (A)	Pengukuran dengan Mistar (M)	
1	0 cm	0 cm	0 %
2	2 cm	2,3 cm	13,04 %
3	5 cm	5,2 cm	3,84 %
4	8 cm	8,2 cm	2,43 %
5	12 cm	12,2 cm	1,64 %
6	15 cm	15,1 cm	0,66 %
7	18 cm	18,1 cm	0,55 %

Tabel 4.8 Data Eror % Pertumbuhan Selama 7 hari Pengamatan Kedua

Hari	Pengamatan Kedua		E%
	Pengukuran dengan Alat (A)	Pengukuran dengan Mistar (M)	
1	0 cm	0 cm	0 %
2	2 cm	2,3 cm	13 %
3	4 cm	4,2 cm	4,7 %
4	7 cm	7,3 cm	4,1 %
5	12 cm	12,2 cm	1,64 %
6	16 cm	16,2 cm	1,23 %
7	20 cm	20,2 cm	0,9 %

Terdapat dua tabel, yaitu pengamatan pertama dan pengamatan kedua. Ini dilakukan supaya alat didapatkan e % rata-rata dan tidak terdapat masalah apabila digunakan berulang-ulang.

Pada tabel di atas merupakan pengukuran antara jarak pada alat dengan jarak pengukuran sebenarnya. Pengukuran jarak pada alat menggunakan sensor ultrasonik sedangkan pengukuran jarak sebenarnya menggunakan mistar. Kemudian didapatkan eror % untuk menunjukkan seberapa akurat jarak pada alat dengan cara membandingkan pengukuran alat dengan mistar.

Pada tabel di atas menunjukkan bahwa semakin tinggi nilai jarak pada alat maka semakin kecil eror %. Dikarenakan obyek yang semakin mendekati sensor ultrasonik maka jaraknya akan semakin akurat. Hal itu dapat dibuktikan dengan hasil data eror % yang didapat seperti pada tabel di atas.

Tabel 4.9 Error % Rata-rata

Hari	E % Pengamatan 1	E % Pengamatan 2	E % Rata-rata
1	0 %	0 %	0 %
2	13,04 %	13 %	13,02 %
3	3,84 %	4,7 %	4,27 %
4	2,43 %	4,1 %	3,26 %
5	1,64 %	1,64 %	1,64 %
6	0,66 %	1,23 %	0,94 %
7	0,55 %	0,9 %	0,72 %

Data tabel diatas merupakan e % rata-rata yang didapatkan dengan merata-rata eror % dari pengamatan pertama dan pengamatan kedua. Kemudian dapat dapat disebutkan eror % rata-rata 0%, 13,02%, 4,27%, 3,26%, 1,64%, 0,94%, dan 0,72%.

4.5 Gambar Alat



Gambar 4.18 Tampilan Foto Alat Keseluruhan

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan data yang telah diperoleh dari analisa hasil penelitian, maka dapat diambil kesimpulan sebagai berikut:

1. Biji tanaman kacang hijau akan lebih cepat tumbuh pada saat biji sudah menjadi kecambah. (Gambar 4.15 hal.44).
2. Pengamatan pertumbuhan tanaman biji kacang hijau dilakukan selama 7 hari. (Tabel 4.5 hal.44).
3. Grafik pengamatan pertumbuhan tanaman biji kacang hijau semakin naik. (Gambar 4.15 hal.44).
4. Semakin tinggi pertumbuhan biji kacang hijau maka semakin kecil eror % yang diperoleh, ditunjukkan pada data eror % hari 2-7 yaitu 15 %, 4 %, 2,5 %, 1,67 %, 0,67 %, dan 0,56 %. (Tabel 4.7 hal.45 dan Tabel 4.8 hal.46).
5. Hasil eror % rata-rata dari pengamatan pertama dan kedua hari 2-7 adalah 13,02 %, 4,27 %, 3,26 %, 1,64 %, 0,94 %, dan 0,72 %. (Tabel 4.9 hal.47).

5.2 Saran

Beberapa saran yang dapat diberikan penulis untuk pengembangan proyek akhir ini antara lain :

1. Pengukur tinggi tanaman hanya sebatas pada objek biji kacang hijau, untuk selanjutnya dapat dikembangkan pada objek tanaman lainnya.
2. Alat ini juga belum dapat menyimpan data pertumbuhannya, jadi pada saat alat ini dimatikan data akan hilang.
3. Pengukuran ini masih sebatas langsung tampil pada LCD Grafik, untuk selanjutnya dapat dikembangkan secara jarak jauh atau sebagainya. Mengingat fungsi modul Arduino Uno yang luas.

DAFTAR PUSTAKA

- Candra Kirana dan Idayu Ria Pramudyanti. 2012. *Biologi SMA/MA Kelas XII Semester Gasal*.Klaten: Viva Pakarindo.
- Pratiwi, D. A., Bambang S., Sri Maryati, Srikini, Suharno. 2007. *Biologi untuk SMA Kelas XII*. Jakarta: Erlangga.
- Owen Bishop. 2010. *Dasar-dasar Elektronika*. Jakarta:Erlangga.
- Heryanto Ary dan Wisnu Adi. 2008. *Pemrograman Bahasa C untuk Mikrokontroller ATmega8535*. Yogyakarta: Andi Offset.
- Budiharto, W. dan Rizal, G. 2005. *12 Proyek Mikrokontroler untuk Pemula*. Jakarta Selatan: PT. Elex Media Komputindo.
- <http://biologimediacentre.com/pertumbuhan-dan-perkembangan-1-pertumbuhan-dan-perkembangan-pada-tumbuhan/>
- <http://blog.famosastudio.com/2011/12/bengkel/menggunakan-ultrasonic-range-sensor-hc-sr04-dan-sdm-io/458>
- <http://inkubator-teknologi.com/ultrasonic-range-meter-dengan-srf05/>
- <http://saptaji.com/2014/03/06/tutorial-Arduino-5-menit-bikin-alat-ukur-jarak-dengan-sensor-ultrasonik-dan-Arduino/>
- <http://inkubator-teknologi.com/menampilkan-tulisan-pada-lcd-grafik/>

A. LAMPIRAN FOTO ALAT



B. LAMPIRAN PERHITUNGAN

Perhitungan Error Persen:

$$\text{Error \%} = \left| \frac{A - M}{M} \right| \times 100\%$$

Diketahui: A = Pengukuran dengan alat

M = Pengukuran dengan mistar

PENGAMATAN 1 :

Hari 1:

$$\text{Error \%} = \left| \frac{0 - 0}{0} \right| \times 100\% = 0\%$$

Hari 2:

$$\text{Error \%} = \left| \frac{2 - 2,3}{2,3} \right| \times 100\% = \left| \frac{0,3}{2,3} \right| \times 100\% = 13,04 \%$$

Hari 3:

$$\text{Error \%} = \left| \frac{5 - 5,2}{5,2} \right| \times 100\% = \left| \frac{0,2}{5,2} \right| \times 100\% = 3,84 \%$$

Hari 4:

$$\text{Error \%} = \left| \frac{8 - 8,2}{8,2} \right| \times 100\% = \left| \frac{0,2}{8,2} \right| \times 100\% = 2,43 \%$$

Hari 5:

$$\text{Error \%} = \left| \frac{12 - 12,2}{12,2} \right| \times 100\% = \left| \frac{0,2}{12,2} \right| \times 100\% = 1,64 \%$$

Hari 6:

$$\text{Error \%} = \left| \frac{15 - 15,1}{15,1} \right| \times 100\% = \left| \frac{0,1}{15,1} \right| \times 100\% = 0,66 \%$$

Hari 7:

$$\text{Error \%} = \left| \frac{18 - 18,1}{18,1} \right| \times 100\% = \left| \frac{0,1}{18,1} \right| \times 100\% = 0,55 \%$$

PENGAMATAN 2 :

Hari 1:

$$\text{Error \%} = \left| \frac{0 - 0}{0} \right| \times 100\% = 0\%$$

Hari 2:

$$\text{Error \%} = \left| \frac{2 - 2,3}{2,3} \right| \times 100\% = \left| \frac{0,3}{2,3} \right| \times 100\% = 13 \%$$

Hari 3:

$$\text{Error \%} = \left| \frac{4 - 4,2}{4,2} \right| \times 100\% = \left| \frac{0,2}{4,2} \right| \times 100\% = 4,7 \%$$

Hari 4:

$$\text{Error \%} = \left| \frac{7 - 7,3}{7,3} \right| \times 100\% = \left| \frac{0,3}{7,3} \right| \times 100\% = 4,1 \%$$

Hari 5:

$$\text{Error \%} = \left| \frac{12 - 12,2}{12,2} \right| \times 100\% = \left| \frac{0,2}{12,2} \right| \times 100\% = 1,64 \%$$

Hari 6:

$$\text{Error \%} = \left| \frac{16 - 16,2}{16,2} \right| \times 100\% = \left| \frac{0,2}{16,2} \right| \times 100\% = 1,23 \%$$

Hari 7:

$$\text{Error \%} = \left| \frac{20 - 20,2}{20,2} \right| \times 100\% = \left| \frac{0,2}{20,2} \right| \times 100\% = 0,9 \%$$

PERHITUNGAN RATA-RATA ERROR % :

$$\text{Error \%} = \frac{\text{E\% Pengamatan 1} + \text{E\% Pengamatan 2}}{2}$$

Hari 1:

$$\text{Error \%} = \frac{0 + 0}{2} = 0\%$$

Hari 2:

$$\text{Error \%} = \frac{13,04 + 13}{2} = 13,02 \%$$

Hari 3:

$$\text{Error \%} = \frac{3,84 + 4,7}{2} = 4,27 \%$$

Hari 4:

$$\text{Error \%} = \frac{2,43 + 4,1}{2} = 3,26 \%$$

Hari 5:

$$\text{Error \%} = \frac{1,64 + 1,64}{2} = 1,64\%$$

Hari 6:

$$\text{Error \%} = \frac{0,66 + 1,23}{2} = 0,94 \%$$

Hari 7:

$$\text{Error \%} = \frac{0,55 + 0,9}{2} = 0,72\%$$

C. LAMPIRAN PROGRAM

```
/*-----  
PROGRAM : PEMBACAAN TINGGI KECAMBAH OTOMATIS  
NAMA   : M. ZAHRONI FIRDAUS  
-----*/
```

```
#include <DS1307.h>  
#include <Wire.h>  
#include <NewPing.h>  
#include <U8glib.h>  
#include <EEPROM.h>
```

```
#define KEY_NONE 0  
#define KEY_MENU 1  
#define KEY_UP 2  
#define KEY_DOWN 3  
#define KEY_OK 4  
#define KEY_CANCEL 5
```

```
uint8_t uiKeyMenu = 7;  
uint8_t uiKeyUp = 5;  
uint8_t uiKeyDown = 6;  
uint8_t uiKeyOk = 4;  
uint8_t uiKeyCancel = 3;
```

```
uint8_t uiKeyCodeFirst = KEY_NONE;  
uint8_t uiKeyCodeSecond = KEY_NONE;  
uint8_t uiKeyCode = KEY_NONE;
```

```
#define TRIGGER_PIN 9 // Arduino pin tied to trigger pin on the ultrasonic sensor.
```

```
#define ECHO_PIN 8 // Arduino pin tied to echo pin on the ultrasonic sensor.
#define MAX_DISTANCE 50 // Maximum distance we want to ping for (in centimeters).
    Maximum sensor distance is rated at 400-500cm.

NewPing sonar(TRIGGER_PIN, ECHO_PIN, MAX_DISTANCE); // NewPing setup of pins
    and maximum distance.

U8GLIB_ST7920_128X64 u8g(13, 11, 10, U8G_PIN_NONE);

int xg, yg;
unsigned int uS, h, me, s, d, m, y, menu_current, menu_redraw_required, last_key_code, ta,
    ts;
char buff[10], buff2[33], buff3[10], buff4[33];

unsigned char kondisi=0; // 0=>menu, 1=>data baru, 3=>view graphic, 4=>hapus
unsigned int eh, eme, ehemp;
unsigned char proses=0, datacount=0, count=0;
unsigned char data[85];

void setup() {
uiSetup(); // setup key detection and debounce algorithm
menu_redraw_required = 1; // force initial redraw

// put your setup code here, to run once:
/*RTC.stop();
RTC.set(DS1307_SEC,1); //set the seconds
RTC.set(DS1307_MIN,05); //set the minutes
RTC.set(DS1307_HR,22); //set the hours
RTC.set(DS1307_DOW,4); //set the day of the week
RTC.set(DS1307_DATE,25); //set the date
```

```
RTC.set(DS1307_MTH,9);    //set the month
RTC.set(DS1307_YR,14);   //set the year
RTC.start();*/
}
void uiSetup(void) {
    // configure input keys
    pinMode(uiKeyMenu, INPUT);    // set pin to input
    digitalWrite(uiKeyMenu, HIGH); // turn on pullup resistors
    pinMode(uiKeyUp, INPUT);     // set pin to input
    digitalWrite(uiKeyUp, HIGH); // turn on pullup resistors
    pinMode(uiKeyDown, INPUT);   // set pin to input
    digitalWrite(uiKeyDown, HIGH); // turn on pullup resistors
    pinMode(uiKeyOk, INPUT);     // set pin to input
    digitalWrite(uiKeyOk, HIGH); // turn on pullup resistors
    pinMode(uiKeyCancel, INPUT); // set pin to input
    digitalWrite(uiKeyCancel, HIGH); // turn on pullup resistors
}

void uiStep(void) {
    uiKeyCodeSecond = uiKeyCodeFirst;
    if ( digitalRead(uiKeyMenu) == LOW )
        uiKeyCodeFirst = KEY_MENU;
    else if ( digitalRead(uiKeyUp) == LOW )
        uiKeyCodeFirst = KEY_UP;
    else if ( digitalRead(uiKeyDown) == LOW )
        uiKeyCodeFirst = KEY_DOWN;
    else if ( digitalRead(uiKeyOk) == LOW )
        uiKeyCodeFirst = KEY_OK;
    else if ( digitalRead(uiKeyCancel) == LOW )
        uiKeyCodeFirst = KEY_CANCEL;
```



```
else
    uiKeyCodeFirst = KEY_NONE;

if ( uiKeyCodeSecond == uiKeyCodeFirst )
    uiKeyCode = uiKeyCodeFirst;
else
    uiKeyCode = KEY_NONE;
}

#define MENU_ITEMS 5
char *menu_strings[MENU_ITEMS] = {"New Proses", "Cancel Proses", "View Grafik",
    "Delete Data"};

void draw_menu(void) {
    uint8_t i, h;
    u8g_uint_t w, d;
    u8g.setFont(u8g_font_6x13);
    u8g.setFontRefHeightText();
    u8g.setFontPosTop();
    h = u8g.getFontAscent()-u8g.getFontDescent();
    w = u8g.getWidth();
    for( i = 0; i < MENU_ITEMS; i++ ) { // draw all menu items
        d = (w-u8g.getStrWidth(menu_strings[i]))/2;
        u8g.setDefaultForegroundColor();
        if ( i == menu_current ) { // current selected menu item
            u8g.drawBox(0, i*h+1, w, h); // draw cursor bar
            u8g.setDefaultBackgroundColor();
        }
        u8g.drawStr(d, i*h, menu_strings[i]);
    }
}
```

```
}

void newProccess(void){
    eh=RTC.get(DS1307_HR,true);
    eme=RTC.get(DS1307_MIN,false);
    proses=1;
    kondisi=3;
}

void draw(void) {
    // graphic commands to redraw the complete screen should be placed here
    u8g.setFont(u8g_font_u8glib_4);
    ta=21-uS;
    ts=data[datacount];
    sprintf(buff3,"%d",ts);
    sprintf(buff,"%d",ta);
    u8g.drawStr(108, 54, "Ts=");
    u8g.drawStr(120, 54, buff3);
    u8g.drawStr(108, 47, "Ta=");
    u8g.drawStr(120, 47, buff);
    sprintf(buff2,"%d:%d:%d" , h, me, s);
    sprintf(buff4,"%d/%d/%d", d, m, y);
    u8g.drawStr( 50, 4, buff2);
    u8g.drawStr( 90, 4, buff4);

    //grafik 8 pxel dari bawah => yg=64-7 => yg=56-data[count]
    //grafik 10 pixel dari kiri => xg+10 => xg=10+count
    for(count=0; count<datacount; count++){
        yg=57-(data[count]/2);
        xg=10+count;
```

```
u8g.drawStr(xg,yg,".");  
}  
//u8g.drawLine(64, 5, 5, 0); // panjang-y target- x awal - y awal
```

```
u8g.drawStr( 0, 4, "Tinggi (cm)");  
u8g.drawVLine(9, 5, 52); // x, y , panjang  
u8g.drawHLine(8, 6, 3); // x, y , panjang  
u8g.drawHLine(7, 7, 5); // x, y , panjang  
u8g.drawHLine(8, 18, 3); // x, y , panjang  
u8g.drawHLine(8, 28, 3); // x, y , panjang  
u8g.drawHLine(8, 38, 3); // x, y , panjang  
u8g.drawHLine(8, 48, 3); // x, y , panjang  
u8g.drawStr(0, 64, "0");  
u8g.drawStr(0, 50, "10");  
u8g.drawStr(0, 40, "20");  
u8g.drawStr(0, 30, "30");  
u8g.drawStr(0, 20, "40");
```

```
u8g.drawStr( 103, 60, "JmlHari");  
u8g.drawVLine(100, 56, 3); // x, y , panjang  
u8g.drawHLine(7, 57, 95); // x, y , panjang  
u8g.drawVLine(99, 55, 5); // x, y , panjang  
u8g.drawVLine(21, 56, 3); // x, y , panjang  
u8g.drawVLine(33, 56, 3); // x, y , panjang  
u8g.drawVLine(45, 56, 3); // x, y , panjang  
u8g.drawVLine(57, 56, 3); // x, y , panjang  
u8g.drawVLine(69, 56, 3); // x, y , panjang  
u8g.drawVLine(81, 56, 3); // x, y , panjang  
u8g.drawVLine(93, 56, 3); // x, y , panjang  
u8g.drawStr(20, 64, "1");
```

```
u8g.drawStr(32, 64, "2");  
u8g.drawStr(44, 64, "3");  
u8g.drawStr(56, 64, "4");  
u8g.drawStr(68, 64, "5");  
u8g.drawStr(80, 64, "6");  
u8g.drawStr(92, 64, "7");  
}
```

```
void update_menu(void)  
{  
  if ( uiKeyCode != KEY_NONE && last_key_code == uiKeyCode ) {  
    return;  
  }  
  last_key_code = uiKeyCode;  
  switch ( uiKeyCode ) {  
    case KEY_MENU:  
      if(kondisi == 3){  
        kondisi=0;  
        menu_current=2;  
        menu_redraw_required = 1;  
      }  
      break;  
    case KEY_UP:  
      menu_current++;  
      if(menu_current >= 4)  
        menu_current = 0;  
      menu_redraw_required = 1;  
      break;  
    case KEY_DOWN:  
      if(menu_current == 0)
```

```
    menu_current = 4;
    menu_current--;
    menu_redraw_required = 1;
    break;
case KEY_OK:
    if((menu_current == 0)& kondisi!=1){
        kondisi=1;
    }else if((menu_current == 1)&& kondisi!=2){
        kondisi=2;
    }else if((menu_current == 2)&& kondisi!=3){
        kondisi=3;
    }else if((menu_current == 3)&& kondisi!=4){
        kondisi=4;
    }
    break;
case KEY_CANCEL:
    if(((kondisi==1)||kondisi==2)||kondisi==4){
        kondisi=0;
        menu_redraw_required = 1;
    }
    break;
}
}

void loop(void) {
    uiStep(); // check for key press
    update_menu();

    delay(50); // Wait 50ms between pings (about 20 pings/sec). 29ms should be the shortest
               // delay between pings.
```



```
uS = sonar.ping()/ US_ROUNDTRIP_CM; // Send ping, get ping time in microseconds
    (uS).
h=RTC.get(DS1307_HR,true);
me=RTC.get(DS1307_MIN,false);
s=RTC.get(DS1307_SEC,false);
d=RTC.get(DS1307_DATE,false);
m=RTC.get(DS1307_MTH,false);
y=RTC.get(DS1307_YR,false);

if(proses==1){
    ehtemp=eh+2;
    if(ehtemp>24) ehtemp=ehtemp-24;
    if(ehtemp==h){
        if(eme==me){
            data[datacount]=21-uS;
            eh=eh+2;
            if(eh>24){
                eh=eh-24;
            }
            datacount++;
        }
    }
}

if(kondisi==0){
    if ( menu_redraw_required != 0 ) {
        u8g.firstPage();
        do {
            draw_menu();          // inside picture loop
        } while( u8g.nextPage() );
    }
}
```

```
menu_redraw_required = 0;
}
}
if(kondisi==1){
  if(proses==0){
    u8g.firstPage();
    delay(100);
    do {
      u8g.setFont(u8g_font_courB08);
      u8g.drawStr( 17, 16, "Apakah Anda Akan");
      u8g.drawStr( 17, 32, "Melakukan Proses");
      u8g.drawStr( 17, 48, " Pengukuran ?");
    } while( u8g.nextPage() );
    if( digitalRead(uiKeyOk) == LOW ){
      while(digitalRead(uiKeyOk) == LOW){}
      newProscsess();
      delay(200);
    }
  }else{
    u8g.firstPage();
    delay(100);
    do {
      u8g.setFont(u8g_font_courB08);
      u8g.drawStr( 0, 10, " Proses Sedang");
      u8g.drawStr( 0, 26, " Berlangsung Pilih");
      u8g.drawStr( 0, 42, "'Cancel Proses' Untuk ");
      u8g.drawStr( 3, 58, "Membuat Proses Baru !");
    } while( u8g.nextPage() );
  }
}
```

```
if(kondisi==2){
  if(proses==1){
    u8g.firstPage();
    delay(100);
    do {
      u8g.setFont(u8g_font_courB08);
      u8g.drawStr( 17, 16, "Apakah Anda Akan");
      u8g.drawStr( 10, 32, "Membatalkan Proses");
      u8g.drawStr( 17, 48, " Pengukuran ?");
    } while( u8g.nextPage() );
    if( digitalRead(uiKeyOk) == LOW ){
      while(digitalRead(uiKeyOk) == LOW){}
      proses=0;
      kondisi=0;
      menu_current=1;
      menu_redraw_required = 1;
      delay(200);
    }
  }else{
    u8g.firstPage();
    delay(100);
    do {
      u8g.setFont(u8g_font_courB08);
      u8g.drawStr( 12, 16, " Tidak Ada Proses");
      u8g.drawStr( 12, 32, " Yang Sedang");
      u8g.drawStr( 12, 48, " Berlangsung !");
    } while( u8g.nextPage() );
  }
}
if(kondisi==3){
```

```
u8g.firstPage();
delay(100);
do {
  draw();
} while( u8g.nextPage() );
}
if(kondisi==4){
  if(proses==0){
    if(datacount!=0){
      u8g.firstPage();
      delay(100);
      do {
        u8g.setFont(u8g_font_courB08);
        u8g.drawStr( 17, 16, "Apakah Anda Akan");
        u8g.drawStr( 17, 32, " Menghapus Data");
        u8g.drawStr( 17, 48, " Pengukuran ?");
      } while( u8g.nextPage() );
      if( digitalRead(uiKeyOk) == LOW ){
        while(digitalRead(uiKeyOk) == LOW){}
        for(count=0; count<datacount; count++){
          data[count]=0;
        }
        datacount=0;
        kondisi=0;
        menu_current=3;
        menu_redraw_required = 1;
        delay(200);
      }
    }else{
      u8g.firstPage();
```

```
delay(100);
do {
  u8g.setFont(u8g_font_courB08);
  u8g.drawStr( 17, 16, " Tidak Ada Data");
  u8g.drawStr( 17, 32, " Yang");
  u8g.drawStr( 17, 48, " Tersimpan !");
} while( u8g.nextPage() );
}
}else{
u8g.firstPage();
delay(100);
do {
  u8g.setFont(u8g_font_courB08);
  u8g.drawStr( 0, 10, " Proses Sedang");
  u8g.drawStr( 0, 26, " Berlangsung Pilih");
  u8g.drawStr( 0, 42, "'Cancel Proses' Untuk ");
  u8g.drawStr( 3, 58, "Dapat Menghapus Data!");
} while( u8g.nextPage() );
}
}
// rebuild the picture after some delay
delay(50);
}
```