



**PROTOTIPE SISTEM PEMOTONGAN *PLYWOOD* DENGAN
PERANGKAT *ENCODER* DAN *INTERFACE* VISUAL BASIC BERBASIS
ARDUINO UNO**

PROYEK AKHIR

Oleh
Natanael Nugroho
NIM 131903102006

**PROGRAM STUDI DIPLOMA III
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS JEMBER
2016**



**PROTOTIPE SISTEM PEMOTONGAN *PLYWOOD* DENGAN
PERANGKAT *ENCODER* DAN *INTERFACE* VISUAL BASIC BERBASIS
ARDUINO UNO**

PROYEK AKHIR

diajukan guna melengkapi tugas akhir dan memenuhi syarat-syarat
untuk menyelesaikan Program Studi Teknik Elektro (DIII)
dan mencapai gelar Ahli Madya (Amd)

Oleh
Natanael Nugroho
NIM 131903102006

**PROGRAM STUDI DIPLOMA III
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS JEMBER
2016**

PERSEMBAHAN

Tugas akhir ini merupakan sebuah awal, langkah kecil menuju lompatan besar guna menggapai kesuksesan yang lebih baik lagi. Untuk itu saya ucapkan rasa syukur dan terima kasih sebesar-besarnya kepada...

Tuhan yang Maha Esa, dengan segala Keagungan dan Keajaiban-Nya yang senantiasa mendengar do'a ku, menuntunku dari dari kegelapan, serta senantiasa menaungiku dengan kasih dan anugrah dan junjunganku Tuhan Yesus Kristus yang telah menjadi penerang Jalan di dunia dan menjadi teladan bagi kita semua;

Ibunda Maria Sumistun, Ayahanda Andreas S, Kakakku Restu Handayani, Peris Maryati dan Andri Prasetyo, terima kasih atas segala kasih sayang, dukungan, semangat, dan doa selama ini;

Seluruh teman dan sahabat seperjuangan Teknik Elektro angkatan 2013, kalian sebagai tempat berbagi suka dan duka yang tidak akan terlupakan. Aku menjadikan kalian semua bagian dari diriku dan aku sangat menyayangi kalian semua;

Seluruh Dosen Jurusan Teknik Elektro yang terhormat, terima kasih telah banyak memberikan ilmu dan mendidik dengan penuh kesabaran;

Guru-guruku sejak SD, SMP, SMK, sampai PT yang terhormat, terima kasih telah memberikan ilmu dan mendidik dengan penuh keikhlasan dan kesabaran

Almamater Fakultas Teknik Universitas Jember.

MOTTO

"Dengan apakah seorang muda mempertahankan kelakuan bersih? Dengan menjaganya sesuai dengan firman-Mu. Dengan segenap hatiku aku mencari Engkau, janganlah biarkan aku menyimpang dari perintah-perintah MU"

(Bb: Mzm. 119:9-10)

"Bersatu kita teguh, bercerai kita runtuh"

(Bhenika Tunggal Ika)

"Kegagalan adalah kesuksesan yang tertunda, tapi kesuksesan tidak akan datang tanpa kita bisa mengubah kegagalan"

(Natanael Nugroho)

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Natanael Nugroho

NIM : 131903102006

Menyatakan dengan sesungguhnya bahwa proyek akhir yang berjudul: “*Prototipe Sistem Pemotongan Plywood Dengan Perangkat Encoder Dan Interface Visual Basic Berbasis Arduino Uno*” adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya dan belum pernah diajukan pada institusi mana pun 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, 16 Juni 2016

Yang menyatakan,

Natanael Nugroho
NIM 131903102006

PROYEK AKHIR

**PROTOTIPE SISTEM PEMOTONGAN *PLYWOOD* DENGAN
PERANGKAT *ENCODER* DAN *INTERFACE* VISUAL BASIC BERBASIS
ARDUINO UNO**

Oleh

Natanael Nugroho

NIM 131903102006

Pembimbing :

Dosen Pembimbing Utama : Widjonarko, S.T., M.T.

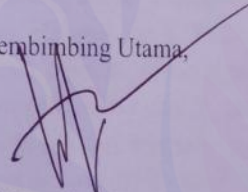
Dosen Pembimbing Anggota : Ir. Widyono Hadi, M.T.

LEMBAR PENGESAHAN

Laporan Proyek Akhir berjudul "*Prototipe Sistem Pemotongan Plywood Dengan Perangkat Encoder Dan Interface Visual Basic Berbasis Arduino Uno*" oleh Natanael Nugroho NIM: 131903102006 telah diuji dan disahkan oleh Fakultas Teknik Elektro Universitas Jember pada;

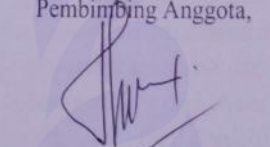
Hari : Selasa
Tanggal : 28 Juni 2016
Tempat : Ruang 22

Pembimbing Utama,




Widjonarko, S.T., M.T.
NIP 19710908 199903 1 001

Pembimbing Anggota,



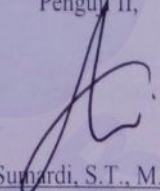
Ir. Widyono Hadi, M.T.
NIP 19610414 198902 1 001

Penguji I,



Dr. Bambang Sri Kaloko, S.T., M.T.
NIP. 19710402 200312 1 001

Penguji II,



Sunardi, S.T., M.T.
NIP 19670113 199802 1 001

Mengesahkan
Dekan,



Dr. H. Hidayat M.U.M
NIP. 19661218 199503 2 001

LEMBAR PENGESAHAN

Laporan Proyek Akhir berjudul “*Prototipe Sistem Pemotongan Plywood Dengan Perangkat Encoder Dan Interface Visual Basic Berbasis Arduino Uno*” oleh Natanael Nugroho NIM: 131903102006 telah diuji dan disahkan oleh Fakultas Teknik Elektro Universitas Jember pada;

Hari : Selasa
Tanggal : 28 Juni 2016
Tempat : Ruang 22

Pembimbing Utama,

Widjonarko, S.T., M.T
NIP 19710908 199903 1 001

Penguji I,

Dr. Bambang Sri Kaloko, S.T., M.T.
NIP. 19710402 200312 1 001

Pembimbing Anggota,

Ir. Widyono Hadi, M.T.
NIP 19610414 198902 1 001

Penguji II,

Sumardi, S.T., M.T.
NIP 19670113 199802 1 001

Mengesahkan
Dekan,

Dr.Ir. Entin Hidayah M.U.M
NIP. 19661215 199503 2 001

PROTOTIPE SISTEM PEMOTONGAN *PLYWOOD* DENGAN PERANGKAT *ENCODER* DAN *INTERFACE VISUAL BASIC* BERBASIS ARDUINO UNO

Natanael Nugroho

Jurusan Teknik Elektro, Fakultas Teknik, Universitas Jember

ABSTRAK

Didalam dunia industri *plywood* pengaturan jarak pemotong tepi masih dilakukan secara manual sehingga dapat memperlambat waktu produksi dan merugikan. Berkembangnya ilmu pengetahuan dan teknologi yang luas dan pesat, telah mendorong untuk menciptakan suatu pergerakan yang dapat meningkatkan kualitas dan efisiensi kerja. Peningkatan efisiensi kerja memerlukan sistem yang bekerja secara otomatis untuk memotong lembaran *plywood*. Sistem otomatis yang dirancang menggunakan sensor *rotary encoder* sebagai pembacaan pulsa dan untuk mengetahui posisi dari besi ulir. *Input* data menggunakan *interface* dari *visual basic* 2010 menuju *arduino* UNO. Motor DC sebagai penggerak tenses (besi drat/ulir), dan motor AC sebagai penggerak *single saw* yang dikontrol oleh *relay*. Sistem *Arduino* berfungsi sebagai pusat kendali dari bagian *output*. Dari hasil pengujian lebar potongan sebanyak 10 kali pemotongan diperoleh rata-rata persentase error sebesar 0,0%. Sedangkan hasil pengujian waktu yang diperlukan untuk perpindahan batas pemotongan rata-rata memerlukan 3:6 detik /cm.

Kata Kunci : *Arduino Uno, Rotary Encoder, Motor DC, Relay, Motor AC, Visual Basic*

***PROTOTYPE OF PLYWOOD CUTTING SYSTEM DEVICE WITH VISUAL
BASIC AND INTERFACE BASED OF ARDUINO UNO***

Natanael Nugroho

Department of Electrical Engineering, Engineering Faculty, University of Jember

ABSTRACT

In the world of industrial plywood cutting edge spacing is still manually so that it can make slow the production time and detrimental. As science and technology are vast and rapidly, it has been pushing to create a movement that can improve the quality and efficiency of work. Improved work efficiency requires a system that works automatically to cut sheets of plywood. Automated systems are designed using a rotary encoder sensor pulse readings and to determine the position of a metal screw. Input data using visual basic interface of 2010 to the arduino UNO. DC motor as a tensen (iron threaded / screw), and the AC motor as the driving single SAW controlled by relays. Arduino system serves as the central control of the output section. From the test results as much as 10 times the width of the pieces obtained by cutting the average percentage error of 0.0%. While the results of the testing time required for the transfer of cutting boundaries need an average of 3: 6 sec / cm.

Keyword : Arduino Uno, Rotary Encoder, Motor DC, Relay, Motor AC, Visual Basic

RINGKASAN

“Prototipe Sistem Pemotongan Plywood Dengan Perangkat Encoder Dan Interface Visual Basic berbasis Arduino Uno”; Natanael Nugroho 131903102006; 2016: 52 halaman; Program Studi Diploma Tiga (DIII) Teknik , Jurusan Teknik Elektro, Fakultas Teknik Universitas Jember.

Kemajuan teknologi dan dunia elektronika pada saat ini sudah memasuki era modernisasi serta perkembangannya memasuki era baru dan sudah serba otomatis. Salah satunya adalah pemotong *plywood*. Mesin pemotong *plywood* dibutuhkan bagi dunia industri *plywood* guna meningkatkan efektifitas waktu kerja bagi para karyawan sehingga hasil produksi semakin banyak dalam waktu yang sedikit. Sehingga dibutuhkan mesin pemotong *plywood* yang dapat bekerja secara otomatis guna mempermudah sistem kerjanya.

Proyek akhir ini mengembangkan sistem mesin pemotong *plywood* yang bekerja secara otomatis. Sistem kerjanya dikendalikan oleh Arduino UNO dengan interface dengan PC sebagai perintah masukan lebar potongan. Dan motor *power window* bergerak bersamaan pembacaan dengan sensor rotary encoder. Yang kemudian ketika lebar potongan telah sesuai maka mesing *saw* akan berputar dengan putaran tinggi untuk memotong *plywood*.

Pengujian dilakukan pada interface data yang terkirim dari visual basic menuju arduino UNO, dari hasil pengujian data yang terkirim telah sama dengan data yang akan di panggil pada arduino UNO. Kemudian pengujian pada lebar potongan dari *plywood*, dimana hasil potongan tidak jauh beda terhadap hasil pengukuran potongan. Pengujian selanjutnya mengukur waktu perpindahan tenses dari ukuran 0 menuju ukuran lebar potongan yang diperintahkan, hasil dari pengujian ini dipengaruhi oleh kecepatan pada motor *power window* yang terkait terhadap gear yang terpasang pada motor *power window* dan pada tenses (besi ulir).

“Prototipe Sistem Pemotongan Plywood Dengan Perangkat Encoder Dan Interface Visual Basicberbasis Arduino Uno”; Natanael Nugroho 131903102006; 2015: 100 pages; *Three Studies Diploma (DIII) Engineering, Department of Electrical Engineering, Faculty of Engineering University of Jember.*

The developing of technology and electronics in the world is now entering this era of modernization and its development into a new era and has been completely automated. One of them is cutting plywood. Cutting machine plywood needed for the industry to increase the effectiveness of working time for employees so that yield more in a little time. So it takes a plywood cutting machine that can work automatically in order to simplify the system works.

The final project is developing systems plywood cutting machine that works automatically. The system works controlled by Arduino UNO to interface with the PC as input commands wide pieces. And the power window motor to move in conjunction with sensor readings rotary encoder. That was when the width of the pieces have been appropriate then mesing saw rotates with high rotation for cutting plywood.

Tests performed on the interface data sent from Visual Basic to the arduino UNO, the test results have been the same data sent with the data to be on call at arduino UNO. Later testing on a wide piece of plywood, the results not much different pieces of the pieces of measurement results. The next test measures the time displacement tensen dariukuran 0 to the width size pieces are ordered, the results of this testing is influenced by the speed at which the power window motor linked to the gear that is attached to the power window motor and the tensen (iron thread).

PRAKATA

Hormat Kami,

Puji syukur ke hadirat Tuhan yang Maha Esa atas segala berkat dan kasih-Nya sehingga laporan proyek akhir yang berjudul “*Prototipe Sistem Pemotongan Plywood Dengan Perangkat Encoder Dan Interface Visual Basic Berbasis Arduino Uno*” 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.

Terselesaikannya laporan proyek akhir ini tidak terlepas dari bantuan berbagai pihak, oleh karena itu disampaikan ucapan terima kasih kepada:

1. Tuhan Yesus Kristus yang telah memberi kelancaran dan kemudahan sehingga terselesaikannya proyek akhir ini.
2. Bapak/Ibu, Keluarga Besar dan saudara Andreas Sukamsi terkasih telah memberikan dorongan semangat, motivasi, dukungan dan doanya demi terselesaikannya proyek akhir ini.
3. Ibu Dr. Ir. Entin Hidayah M.U.M selaku Dekan Fakultas Teknik Universitas Jember.
4. Bapak Dr. Ir. Bambang Sri Kaloko, S.T., M.T. selaku Ketua Jurusan Fakultas Teknik Elektro Universitas Jember.
5. Bapak Widjonarko, A.Md., S.T., M.T. selaku Ketua Prodi D3 Fakultas Teknik Elektro Universitas Jember.
6. Bapak Widjonarko, S.T., M.T. selaku dosen pembimbing utama dan Bapak Ir. Widyono Hadi, M.T. selaku dosen pembimbing anggota yang telah meluangkan waktu dan pikiran guna memberikan bimbingan dan pengarahan dalam penyusunan proyek akhir ini.
7. Bapak Satriyo Budi Utomo, S.T., M.T. yang telah memberikan bimbingan dan pengarahan dalam penyusunan proyek akhir ini.
8. Seluruh Dosen yang ada di Fakultas Teknik khususnya Teknik Elektro beserta karyawan.

9. Keluarga besar Teknik Elektro khususnya angkatan 2013 INTEL UNEJ, terimakasih atas dukungan dan motivasi yang kalian berikan.
10. Teman – teman seperjuangan DEGAN UNEJ 2013 yang selalu mendukung selama menjalani masa kuliah sampai terselesaikannya proyek akhir ini, kenangan dan pengalaman tak akan pernah terlupakan.
11. Semua pihak yang tidak dapat saya sebutkan satu persatu, terima kasih atas dukungan dan motivasi kalian dalam penyusunan proyek akhir ini.

Penulis menyadari bahwa sebagai manusia biasa tidak terlepas dari keterbatasan, yang biasanya akan mewarnai kadar ilmiah dari proposal proyek akhir ini. Oleh karena itu penulis selalu terbuka terhadap masukan dan saran dari semua pihak yang sifatnya membangun untuk mendekati kesempurnaan. Tidak lupa penulis menyampaikan permohonan maaf yang sebesar-besarnya jika terdapat kesalahan dan kekeliruan. Akhir kata penulis berharap laporan ini dapat memberikan manfaat bagi pembaca dan dapat menjadi bahan acuan yang bermanfaat di kemudian hari.

Jember, 16 Juni 2016

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PERSEMBAHAN	ii
HALAMAN MOTTO	iii
HALAMAN PERNYATAAN	iv
HALAMAN PEMBIMBING	v
HALAMAN PENGESAHAN	vi
ABSTRAK	vii
ABSTRACT	viii
RINGKASAN	ix
SUMMARY	x
PRAKATA	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xvi
DAFTAR TABEL	xvii
BAB I. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian	3
1.4 Batasan Masalah	3
1.5 Manfaat	3
BAB II. TINJAUAN PUSTAKA	4
2.1 Definisi <i>plywood</i>	4
2.2 Sensor <i>Rotary Encoder</i>	4
2.3 <i>Arduino Uno</i>	7
2.3.1 <i>Power Supply</i>	9
2.3.2 <i>Memori</i>	10

2.3.3 <i>Input & Output</i>	10
2.3.4 Komunikasi	11
2.3.5 <i>Software Arduino</i>	11
2.3.6 Perangkat Lunak (<i>Arduino IDE</i>)	11
2.4 Motor DC	12
2.5 Motor AC	12
2.6 <i>Lmit Switch</i>	13
2.7 Besi Ulir (<i>Treads/Drat</i>)	14
BAB III. METODOLOGI PENELITIAN	15
3.1 Waktu dan Tempat	15
3.2 Alat dan Bahan	15
3.2.1 Pembuatan Sensor <i>Rotary Encoder</i>	15
3.2.2 Pembuatan Modul Relay 6 <i>Channel</i>	15
3.2.3 Pembuatan Tampilan LCD	15
3.2.4 <i>Software</i>	16
3.2.5 <i>Output</i>	16
3.2.6 Pembuatan <i>Power Supply</i>	16
3.2.7 Alat	16
3.3 Blok Diagram Alat	17
3.3.1 Blok Diagram Kerja Dari Alat	17
3.3.2 Blok Diagram Proses Pengambilan Data	19
3.4 Perancangan Sistem	20
3.4.1 Perancangan <i>Software</i>	20
a. Aduino	20
b. Visual Basic 2010	21
3.4.2 Rangkaian Modul Relay 6 <i>Channel</i>	22
3.4.3 Rangkaian Driver Foward-Reverse Motor Power Window	24
3.4.4 Rangkaian LCD	25
3.4.5 Rangkaian <i>Power Supply</i>	26
3.5 Diagram Alir	27

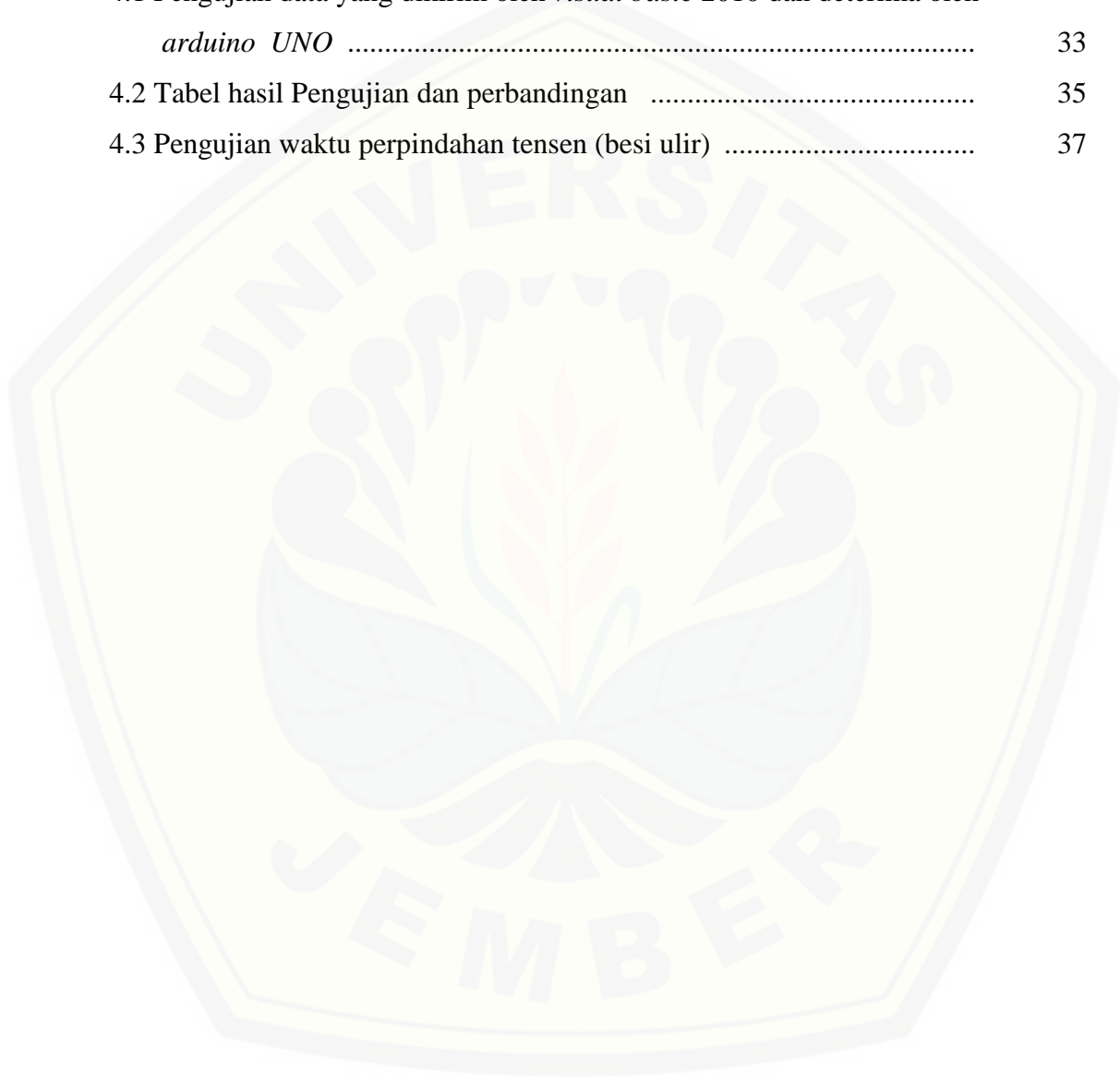
3.6 Perancangan Mekanik Alat	29
3.6.1 Desain Mekanik Alat	29
3.6.2 Bahan Dan Bentuk Alat	30
3.6.3 Hasil Perancangan Mekanik Alat dari Pemotong <i>Plywood</i>	31
BAB IV. ANALISA DATA DAN PEMBAHASAN	33
3.1 Pengujian Tombol Pilihan Dari Software Vibi 2010	33
3.2 Pengujian Potongan Lebar Plywood	35
3.3 Pengujian Waktu Tempuh Perpindahan Tensen Mengikuti Perintah Lebar Potongan	37
BAB V. PENUTUP	39
5.1 Kesimpulan	39
5.2 Saran	40
DAFTAR PUSTAKA	41
LAMPIRAN	

DAFTAR GAMBAR

	Halaman
2.1 Sensor <i>Rotary Encoder</i>	5
2.2 Rangkaian Tipikal Penghasil Pulsa pada <i>Rotary Encoder</i>	6
2.3 Susunan Piringan untuk <i>Incremental Encoder</i>	6
2.4 Contoh Pola Keluaran <i>Incremental Encoder</i>	7
2.5 Output dan Arah Putaran pada Resolusi yang Berbeda-Beda	7
2.6 <i>Board Arduino Uno</i>	8
2.7 Motor DC 12 V	12
2.8 Motor Mesin Jahit 220V	13
2.9 Simbol <i>limit switch</i>	14
2.10 Simbol Besi Ulir (<i>Drat</i>)	15
3.1 Blok Diagram Alat	17
3.2 Proses Pengambilan Data	19
3.3 <i>Listing</i> Program Arduino UNO	21
3.4 <i>Layout</i> Kontrol	22
3.5 Rangkaian Modul Relay 5 <i>Chanel</i>	23
3.6 Rangkaian Modul Relay Pengendali <i>Foward-Reverse</i> motor <i>power window</i>	24
3.7 Rangkaian LCD	25
3.8 Rangkaian <i>Power Supply</i>	26
3.9 Flowchart	27
3.10 Perancangan Mekanik Alat Pemotongan <i>Plywood</i>	29
3.11 Hasil Perancangan Mekanik Alat Pemotongan <i>Plywood</i>	30
3.12 Ukuran Mekanik Pemotong <i>Plywood</i> Tampak Atas	32
3.13 Ukuran Mekanik Pemotong <i>Plywood</i> Tampak Samping	32
4.1 Tampilan pengendali menggunakan <i>visual basic 2010</i>	34

DAFTAR TABEL

	Halaman
2.1 Deskripsi <i>Arduino Uno</i>	9
4.1 Pengujian data yang dikirim oleh <i>visual basic</i> 2010 dan diterima oleh <i>arduino UNO</i>	33
4.2 Tabel hasil Pengujian dan perbandingan	35
4.3 Pengujian waktu perpindahan tenses (besi ulir)	37



BAB I. PENDAHULUAN

1.1 Latar Belakang

Kebutuhan *plywood* dewasa ini terus meningkat, hal ini seiring dengan perkembangan pembangunan. Meningkatnya kebutuhan *plywood* di pasar mendorong pertumbuhan industri *plywood*. Bertumbuhnya industri *plywood* mengakibatkan meningkatnya persaingan baik dalam kuantitas maupun kualitas. Kualitas *plywood* ditentukan oleh bahan baku, serta proses pembuatan *plywood* itu sendiri.

Didalam proses pembuatan *plywood* dimulai dari bahan baku *log* yang diiris menjadi tipis disebut *viner*. *Viner* yang digunakan untuk membuat *plywood* mempunyai beberapa bagian, diantaranya *face viner*, *core viner* dan *face back* atau *back layer*. *Face viner* merupakan *viner* yang ditempatkan pada lapisan paling atas. *Core viner* merupakan bagian lembaran yang diletakan pada bagian tengah. Sedangkan *face back* atau *back layer* merupakan *viner* yang diletakan pada bagian belakang. Bagian-bagian tersebut kemudian diproses kembali dengan berbagai tahapan melalui mesin sehingga menjadi sebuah lembaran *plywood* mentah. Dari bahan tersebut kembali diproses menggunakan mesin sehingga menjadi bahan *plywood*. Untuk mendapatkan hasil *plywood* yang baik maka harus dilakukan pemotongan lebar dan panjang sesuai standart industri *plywood*.

Berdasarkan hasil *survey* diperusahaan PT. Kutai Timber Indonesia Probolinggo, pengaturan jarak pemotong tepi dilakukan secara manual oleh operator. Hal ini menjadi salah satu faktor yang dapat memperlambat waktu produksi yang dapat merugikan dalam dunia industri. Sehingga dengan berkembangnya ilmu pengetahuan dan teknologi yang luas dan pesat, telah mendorong manusia untuk dapat menciptakan suatu pergerakan yang dapat meningkatkan kualitas dan efisiensi kerja. Apabila proses manual yang masih harus dipertahankan terus-menerus tentu terjadinya penghambatan dalam jumlah produksi karena memakan waktu yang cukup lama.

Dari hasil *survey* di lapangan, saya tergerak akan membuat salah satu alat pemotongan kayu yang dikendalikan oleh sistem pengendali elektronika dan tidak

dikendalikan sepenuhnya kembali oleh manusia lagi namun dengan sistem elektronika ini dapat memudahkan dan mempercepat pada saat produksi. Alat yang kami buat ini adalah alat pemotong lebar kayu yang dikendalikan oleh mikrokontroler untuk mengatur putaran posisi motor agar dapat memberikan batasan potongan sesuai jenis lebar pada *plywood* dengan perintah data jenis lebar pemotongan sesuai standar perusahaan.

Alat yang digunakan pada pengatur lebar pemotongan *plywood* ini menggunakan sistem pembatas tepi sebagai batas ukuran yang dijadikan satu dengan *AS DRAT* (batang besi ulir) dan dapat berputar dengan bantuan motor serta dapat dikendalikan dari berapa kali posisi putaran sehingga dapat mengatur lebar potongan *plywood*. Saat lebar potongan terpenuhi maka mulailah *single saw* (roda gergaji) akan melakukan proses pemotongan hingga proses pemotongan selesai. Pengendali posisi lebar potongan menggunakan sensor *encoder*, dimana sistem kerja sensor *encoder* ini berfungsi untuk mengukur posisi poros motor begitu pula kecepatan. *Encoder* merupakan piranti untuk mengukur gerak dengan *output* berupa rangkaian *pulse* digital. Dengan mencacah *bit* tunggal atau melakukan dekoding rangkaian *bit*, *pulse* dapat dikonversikan menjadi posisi absolut atau inkremental. Umumnya yang banyak digunakan adalah *encodermagnetic* dan *encoderoptic*. *Encoder* termasuk sensor pendeteksi posisi. Sensor adalah jenis transduser yang digunakan untuk mengubah besaran mekanis, magnetis, panas, sinar, dan kimia menjadi tegangan dan arus listrik.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, permasalahan yang dapat dirumuskan sebagai berikut:

1. Bagaimana membuat alat pemotongan *plywood* dengan lebar yang dapat dikendalikan oleh sensor *encoder*?
2. Bagaimana merancang sistem kontrol pengendali sensor *encoder* yang dapat mengetahui lebar dari *plywood*?
3. Bagaimana alat pemotong *plywood* yang dikendalikan oleh *interface* dari *visual basic 2010*?

1.3 Tujuan Penelitian

Tujuan Proyek Akhir ini sebagai berikut:

1. Membuat prototipe sistem pemotong *plywood* dengan perangkat *encoder* dan interface visual basic berbasis arduino.
2. Mengendalikan posisi putaran motor menggunakan sensor *rotary encoder*.
3. Proses pemotongan *plywood* dapat dilakukan secara otomatis.
4. Proses pemotongan *plywood* juga dapat dilakukan secara manual.

1.4 Batasan Masalah

Untuk memperjelas arah dari pembahasan penelitian ini, maka diberikan batasan masalah yang meliputi:

1. Pengendalian posisi lebar potongan pada putaran motor menggunakan sensor *rotary encoder*.
2. Sistem *interface* kontrol yang digunakan berbasis Arduino uno.
3. *Prototipe* ini hanya dapat melakukan potongan *plywood*.
4. Proses pemotongan dilakukan hanya lebar *plywood*.
5. Perintah lebar potongan sudah ditetapkan yaitu lebar potongan 5 cm, 10 cm, 20 cm, 25 cm, 30 cm, 40 cm.
6. Cara kerja dari prototipe sistem pemotong *plywood* dengan perangkat *encoder* dan *interface visual basic* 2010 berbasis arduino UNO.

1.5 Manfaat

Manfaat dari pembuatan Proyek Akhir ini sebagai berikut :

1. Mempermudah operator dalam proses pemotongan lebar *plywood*.
2. Dapat memaksimalkan waktu pada saat melakukan proses pemotongan *plywood*.
3. Untuk merancang industri *plywood* yang memiliki teknologi mengikuti perkembangan zaman.

BAB II. TINJAUAN PUSTAKA

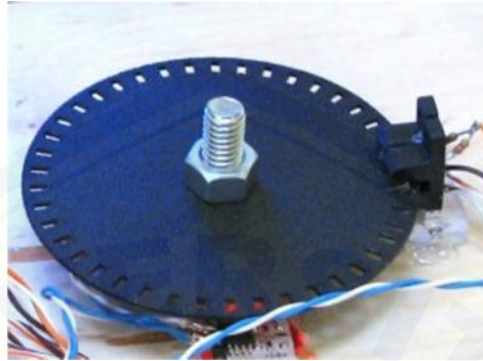
2.1 Definisi *Plywood*

Kayu lapis adalah kayu yang terbuat dari susunan vinir yaitu kayu yang disayat menjadi lembaran-lembaran. Lembaran-lembaran ini disusun dengan arah serat bersilangan dengan jumlah vinir ganjil mulai dari 3 (*triply/triplek*), 5, 7, 9 (*multiply/multiplek*). Bagian-bagian kayu lapis tersebut secara umum memiliki nama disetiap penempatan perlembarannya yakni *face* vinir di tempatkan pada lapisan paling atas dengan kualitas *log* (gulungan kayu) memiliki diameter minimal 45 cm, kualitas bentuk *log* lurus, bulat dan silindris, dan tidak terdapat mata kayu tidak sehat. Selanjutnya pada bagian *core* vinir. Selanjutnya pada bagian *core* viner terletak pada lembaran tengah dengan kualitas *log* (gulungan kayu) memiliki diameter minimal 45 cm, dengan kualitas *log* minimal 85% silindri dan masih segar. Kemudian bagian paling akhir adalah *back layer* atau bisa disebut *face back*, dari bagian sama dengan bagian *face* namun kualitas tidak sebaik dari viner yang diperuntukkan *face layer* contoh *log* yang digunakan antara lain *log* meranti, sengon, mahoni, dll. Lembaran-lembaran tersebut biasanya di peroleh dari proses pengupasan kayu *log* secara rotary. Dari proses ini diperoleh lembaran yang lebar dan panjang pada ketebalan yang kecil (0.3mm – 3mm). Dari konstruksi yang digunakan untuk membuat *plywood*, maka bahan ini sangat tahan terhadap resiko pecah/retak, melengkung atau melintir yang tergantung pula pada ketebalannya. Agar mendapatkan hasil yang maksimal maka dilakukan pemotongan dari panjang dan lebar *plywood* agar mendapatkan hasil yang maksimal dan memiliki kualitas yang lebih dari hasil produksi *plywood* tersebut.

2.2 Sensor *Rotary Encoder*

Rotary encoder adalah divais elektromekanik yang dapat memonitor gerakan dan posisi. Rotary encoder umumnya menggunakan sensor optik untuk menghasilkan serial pulsa yang dapat diartikan menjadi gerakan, posisi, dan arah. Sehingga posisi sudut suatu poros benda berputar dapat diolah menjadi informasi

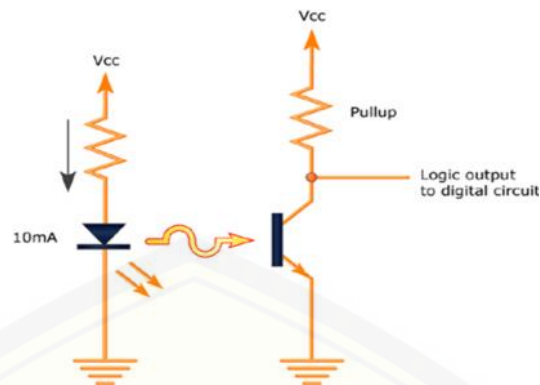
berupa kode digital oleh rotary encoder untuk diteruskan oleh rangkaian kendali. Rotary encoder umumnya digunakan pada pengendalian robot, motor drive, dsb.



Gambar 2.1 Sensor *Rotary Encoder*

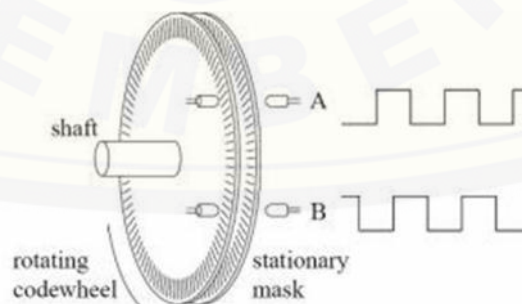
(sumber: <http://all-thewin.blogspot.co.id/2011/02/mengukur-kecepatan-dengan-rotary.html>)

Rotary Encoder tersusun dari suatu piringan tipis yang memiliki lubang-lubang pada bagian lingkaran piringan. LED ditempatkan pada salah satu sisi piringan sehingga cahaya akan menuju ke piringan. Di sisi yang lain suatu photo-transistor diletakkan sehingga photo-transistor ini dapat mendeteksi cahaya dari LED yang berseberangan. Piringan tipis tadi dikopel dengan poros motor, atau divais berputar lainnya yang ingin kita ketahui posisinya, sehingga ketika motor berputar piringan juga akan ikut berputar. Apabila posisi piringan mengakibatkan cahaya dari LED dapat mencapai photo-transistor melalui lubang-lubang yang ada, maka photo-transistor akan mengalami saturasi dan akan menghasilkan suatu pulsa gelombang persegi. Gambar 2.1 menunjukkan bagan skematik sederhana dari rotary encoder. Semakin banyak deretan pulsa yang dihasilkan pada satu putaran menentukan akurasi rotary encoder tersebut, akibatnya semakin banyak jumlah lubang yang dapat dibuat pada piringan menentukan akurasi rotary encoder tersebut.



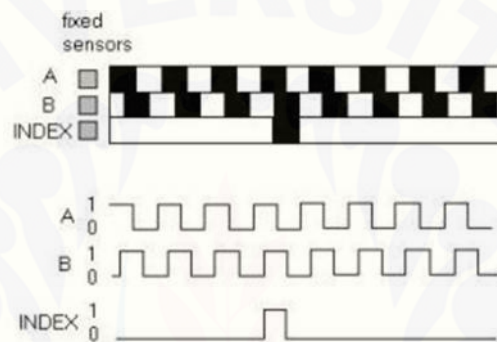
Gambar 2.2 Rangkaian Tipikal Penghasil Pulsa pada *Rotary Encoder*
(sumber: <http://www.embedded.com/design/prototyping-and-development/4024586/Exploring-optical-and-magnetic-sensors>)

Incremental encoder terdiri dari dua track atau single track dan dua sensor yang disebut channel A dan B (Gambar 2.7). Ketika poros berputar, deretan pulsa akan muncul di masing-masing channel pada frekuensi yang proporsional dengan kecepatan putar sedangkan hubungan fasa antara channel A dan B menghasilkan arah putaran. Dengan menghitung jumlah pulsa yang terjadi terhadap resolusi piringan maka putaran dapat diukur. Untuk mengetahui arah putaran, dengan mengetahui channel mana yang leading terhadap channel satunya dapat kita tentukan arah putaran yang terjadi karena kedua channel tersebut akan selalu berbeda fasa seperempat putaran (quadrature signal). Seringkali terdapat output channel ketiga, disebut INDEX, yang menghasilkan satu pulsa per putaran berguna untuk menghitung jumlah putaran yang terjadi.



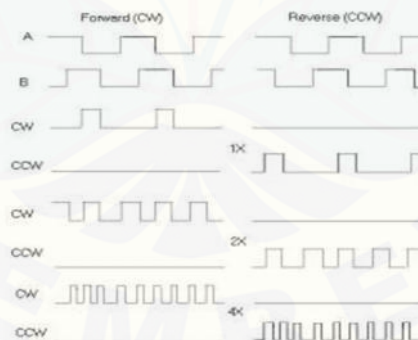
Gambar 2.3 Susunan Piringan untuk *Incremental Encoder*
(sumber: <https://konversi.wordpress.com/2009/06/12/sekilas-rotary-encoder/>)

Contoh pola diagram keluaran dari suatu incremental encoder ditunjukkan pada Gambar 2.8. Resolusi keluaran dari sinyal quadrature A dan B dapat dibuat beberapa macam, yaitu 1X, 2X dan 4X. Resolusi 1X hanya memberikan pulsa tunggal untuk setiap siklus salah satu sinyal A atau B, sedangkan resolusi 4X memberikan pulsa setiap transisi pada kedua sinyal A dan B menjadi empat kali resolusi 1X. Arah putaran dapat ditentukan melalui level salah satu sinyal selama transisi terhadap sinyal yang kedua. Pada contoh resolusi 1X, A = arah bawah dengan B = 1 menunjukkan arah putaran searah jarum jam, sebaliknya B = arah bawah dengan A = 1 menunjukkan arah berlawanan jarum jam.



Gambar 2.4 Contoh Pola Keluaran *Incremental Encoder*

(sumber: <https://konversi.wordpress.com/2009/06/12/sekilas-rotary-encoder/>)



Gambar 2.5 Output dan Arah Putaran pada Resolusi yang Berbeda-beda

(sumber: <https://konversi.wordpress.com/2009/06/12/sekilas-rotary-encoder/>)

2.3 *Arduino Uno*

Arduino Uno adalah sebuah *board mikrokontroler* yang berbasis ATmega328. *Arduino* memiliki 14 *pin input/output* yang mana 6 *pin* dapat digunakan sebagai *output PWM*, 6 *analog input*, *crystal osilator* 16 MHz, koneksi

USB, jack power, kepala ICSP, dan tombol *reset*. *Arduino* mampu men-*support* mikrokontroler, dapat dikoneksikan dengan komputer menggunakan kabel USB.



Gambar 2.6 Board Arduino Uno

(Sumber : <http://dialogsimponi.blogspot.com/2014/11/normal-0-false-false-false-in-x-none-x.html>)

Arduino memiliki kelebihan tersendiri disbanding *board* mikrokontroler yang lain selain bersifat *open source*, *Arduino* juga mempunyai bahasa pemrogramannya sendiri yang berupa bahasa C. Selain itu dalam *board Arduino* sendiri sudah terdapat *loader* yang berupa USB sehingga memudahkan pengguna ketika memprogram mikrokontroler didalam *Arduino*. Sedangkan pada kebanyakan *board* mikrokontroler yang lain yang masih membutuhkan rangkaian *loader* terpisah untuk memasukkan program ketika kita memprogram mikrokontroler. *Port* USB tersebut selain untuk *loader* ketika memprogram, dapat juga difungsikan sebagai *port* komunikasi *serial*.

Arduino menyediakan 20 *pin* I/O, yang terdiri dari 6 *pin* *input analog* dan 14 *pin* *digital input/output*. Untuk 6 *pin* *analog* dapat difungsikan sebagai *output digital* jika diperlukan *output digital* tambahan selain 14 *pin* yang sudah tersedia. Untuk mengubah *pin* *analog* menjadi *digital* cukup mengubah konfigurasi *pin* pada program. Dalam *board* kita bisa lihat *pin* *digital* diberi keterangan 0-13, jadi untuk menggunakan *pin* *analog* menjadi *output digital*, *pin* *analog* yang pada keterangan *board* 0-5 kita ubah menjadi *pin* 14-19. dengan kata lain *pin* *analog* 0-5 berfungsi juga sebagai *pin* *output digital* 14-16.

Tabel 2.1 Deskripsi *Arduino Uno*

Mikrokontroler	ATmega 328
Tegangan Pengoperasian	5 V
Tegangan Input yang disarankan	7 – 12 V
Batas Tegangan Input	6 – 20 V
Jumlah pin I/O digital	14 pin digital (6 diantaranya menyediakan keluaran PWM)
Jumlah pin input Analog	6 pin
Arus DC tiap pin I/O	40mA
Arus DC untuk pin 3,3 V	50mA
Memori Flash	32 KB (ATmega 328) sekitar 0,5 KB digunakan oleh bootloader
SRAM	2 KB (ATmega 328)
EPROM	1 KB (ATmega 328)
Clock Speed	16 MHz

(Sumber: <http://dialogsimponi.blogspot.com/2014/11/normal-0-false-false-false-in-x-none-x.html>)

2.3.1 Power Supply

Arduino dapat diberikan *power* melalui koneksi USB atau *power supply*. *Power*-nya di-select secara otomatis. *Power supply* dapat menggunakan adaptor DC atau baterai. Adaptor dapat dikoneksikan dengan mencolok *jack* adaptor pada koneksi *port input supply*. *Board Arduino* dapat dioperasikan menggunakan *supply* dari luar sebesar 6 - 20 volt. Jika *supply* kurang dari 7V, kadangkala *pin* 5V akan menyuplai kurang dari 5 volt dan *board* bisa menjadi tidak stabil. Jika menggunakan lebih dari 12 V, tegangan di regulator bisa menjadi sangat panas dan menyebabkan kerusakan pada *board*. Rekomendasi tegangan ada pada 7 sampai 12 volt.

Penjelasan pada *pin power* adalah sebagai berikut :

- a. Vin : Tegangan *input* ke *board Arduino* ketika menggunakan tegangan dari luar (seperti yang disebutkan 5 volt dari koneksi USB atau tegangan yang diregulasikan). Pengguna dapat memberikan tegangan melalui *pin* ini, atau jika tegangan suplai menggunakan *power jack*, aksesnya menggunakan *pin* ini.
- b. 5V : Regulasi *power supply* digunakan untuk *power* mikrokontroler dan komponen lainnya pada *board*. 5V dapat melalui Vin menggunakan

regulator pada *board*, atau supply oleh USB atau *supply* regulasi 5V lainnya.

- c. 3V3 : Suplai 3.3 volt didapat oleh FTDI *chip* yang ada di *board*. Arus maksimumnya adalah 50mA
- d. *Pin Ground* : berfungsi sebagai jalur *ground* pada *Arduino*

2.3.2 Memori

ATmega328 memiliki 32 KB *flash* memori untuk menyimpan kode, juga 2 KB yang digunakan untuk *bootloader*. ATmega328 memiliki 2 KB untuk SRAM dan 1 KB untuk EEPROM.

2.3.3 Input & Output

Setiap 14 *pin digital* pada *Arduino* dapat digunakan sebagai *input* atau *output*, menggunakan fungsi `pinMode()`, `digitalWrite()`, dan `digitalRead()`. *Input/output* dioperasikan pada 5 volt. Setiap *pin* dapat menghasilkan atau menerima maksimum 40 mA dan memiliki *internal pull-up* resistor (*disconnected* oleh *default*) 20-50K Ohm.

Beberapa *pin* memiliki fungsi sebagai berikut :

- a. *Serial* : 0 (RX) dan 1 (TX). Digunakan untuk menerima (RX) dan mengirim (TX) TTL data *serial*. *Pin* ini terhubung pada *pin* yang *koresponding* dari USB ke TTL *chip serial*.
- b. *Interrupt eksternal* : 2 dan 3. *Pin* ini dapat dikonfigurasi untuk *trigger* sebuah *interap* pada *low value*, *rising* atau *falling edge*, atau perubahan nilai.
- c. *PWM* : 3, 5, 6, 9, 10, dan 11. Mendukung 8-bit *output* PWM dengan fungsi `analogWrite()`.
- d. *SPI* : 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). *Pin* ini men-*suport* komunikasi SPI, yang mana masih mendukung *hardware*, yang tidak termasuk pada bahasa *Arduino*.
- e. *LED* : 13. Ini adalah dibuat untuk koneksi LED ke *digital pin* 13. Ketika *pin* bernilai *HIGH*, LED hidup, ketika *pin* *LOW*, LED mati.

2.3.4 Komunikasi

Uno Arduino 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). *Firmware Arduino* menggunakan USB *driver standart* COM, dan tidak ada *driver* eksternal yang dibutuhkan. Namun, pada *Windows*, file. Ini diperlukan. Perangkat lunak *Arduino* termasuk *monitor serial* yang memungkinkan data sederhana yang akan dikirim ke *board Arduino*. RX dan TX LED di *board* akan berkedip ketika data sedang dikirim melalui *chip* USB-to-*serial* dan koneksi USB ke komputer.

2.3.5 Software Arduino

Arduino Uno dapat diprogram dengan perangkat lunak *Arduino*. Pada ATmega328 di *Arduino* terdapat *bootloader* yang memungkinkan Anda untuk meng-*upload* kode baru untuk itu tanpa menggunakan *programmer hardware* eksternal.

2.3.6 Perangkat Lunak (Arduino IDE)

IDE *Arduino* adalah *software* yang sangat canggih ditulis dengan menggunakan Java. IDE *Arduino* terdiri dari:

1. Editor program, sebuah *window* yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *Processing*.
2. *Compiler*, sebuah modul yang mengubah kode program (bahasa *Processing*) menjadi kode *biner*. Bagaimanapun sebuah mikrokontroler tidak akan bisa memahami bahasa *Processing*. Yang bisa dipahami oleh mikrokontroler adalah kode biner. Itulah sebabnya *compiler* diperlukan dalam hal ini.
3. *Uploader*, sebuah modul yang memuat kode biner dari komputer ke dalam *memory* didalam papan *Arduino*.

2.4 Motor DC

Motor DC adalah jenis motor listrik yang bekerja menggunakan sumber tegangan DC. Motor DC atau motor arus searah sebagaimana namanya, menggunakan arus langsung dan tidak langsung/direct-unidirectional. *Motor DC* digunakan pada penggunaan khusus dimana diperlukan penyalan torque yang tinggi atau percepatan yang tetap untuk kisaran kecepatan yang luas.

Motor DC memiliki 2 bagian dasar :

1. Bagian yang tetap/stasioner yang disebut stator. Stator ini menghasilkan medan magnet, baik yang dibangkitkan dari sebuah koil (elektro magnet) ataupun magnet permanen.
2. Bagian yang berputar disebut rotor. Rotor ini berupa sebuah koil dimana arus listrik mengalir.



Gambar 2.7 Motor DC 12V

(Sumber : <http://meriwardana.blogspot.co.id/2011/11/prinsip-kerja-motor-arus-searah-dc.html>)

2.5 Motor AC

Motor Ac adalah sebuah motor listrik yang digerakkan oleh alternating current atau arus bolak balik (AC). umumnya, motor AC terdiri dari dua komponen utama yaitu stator dan rotor. seperti yang telah dijelaskan sebelumnya pada motor DC, stator adalah bagian yang diam dan letaknya berada di luar. stator mempunyai coil yang di aliri oleh arus listrik bolak balik dan nantinya akan menghasilkan medan magnet yang berputar. bagian yang kedua yaitu rotor. rotor adalah bagian yang berputar dan letaknya berada di dalam (di sebelah dalam

stator). rotor bisa bergerak karena adanya torsi yang bekerja pada poros dimana torsi tersebut dihasilkan oleh medan magnet yang berputar.

Motor (Dinamo) Control, bagian penting pada mesin jahit bertenaga motor listrik, motor kontrol ini menjadi speed control yang sangat sederhana untuk mengatur kecepatan putaran dari motor dengan cara memindahkan sebagian daya yang seharusnya diberikan ke motor pada sebuah resistor beban yang terbuat dari kepingan-kepingan carbon.

Keping-keping carbon ini selanjutnya berfungsi sebagai hambatan (resistor) yang bersifat variable karena effect tekanan gaya yang diberikan padanya melalui mekanikal pegas yang terhubung pada pedal switch atau saklar injak. Perubahan resistansi sekitar 600 – 250 Ohm sehingga motor secara perlahan berubah kecepatannya sebanding dengan tekanan kaki pada pedal, perubahannya dari lambat menjadi cepat dengan komposisi resistansi mulai dari 600 sampai 250 dan terhubung singkat atau tersambung.



Gambar 2.8 Motor Mesin Jahit 220V

(Sumber : <https://wiwike.wordpress.com/motor-dinamo-control/>)

2.6 *Limit Switch*

Limit switch merupakan jenis saklar yang dilengkapi dengan katup yang berfungsi menggantikan tombol. Prinsip kerja limit switch sama seperti saklar Push ON yaitu hanya akan menghubungkan pada saat katupnya ditekan pada batas penekanan tertentu yang telah ditentukan dan akan memutus saat katup tidak ditekan. Limit switch termasuk dalam kategori sensor mekanis yaitu sensor yang

akan memberikan perubahan elektrik saat terjadi perubahan mekanik pada sensor tersebut. Penerapan dari limit switch adalah sebagai sensor posisi suatu benda (objek) yang bergerak. Dan simbol gambar dari *limit switch* dapat dilihat pada gambar 2.15



Gambar 2.9 Simbol *limit switch*

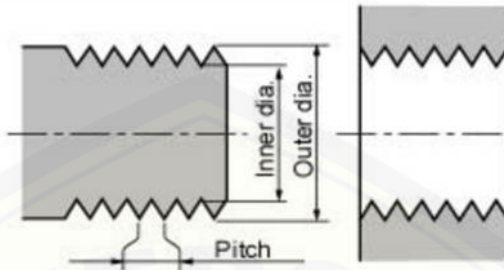
(Sumber <http://elektronika-dasar.web.id/limit-switch-dan-saklar-push-on/svg.png>)

2.7 Besi Ulir (*Treads/Drat*)

Ulir atau drat secara umum digunakan untuk mengikatkan atau mengencangkan beberapa bagian benda. Namun selain itu ulir digunakan untuk menggerakkan.

Dalam hal standar ukuran drat/ulir, ada ulir metrik (M), ulir witworth (W/BSW) ulir paralel untuk pipa (PF), ulir tirus untuk pipa (PT), dan ulir terpadu (UNC, UNF). Banyak digunakan di Jepang dan banyak negara di seluruh dunia. Ukuran Metrik adalah standar internasional, yang digunakan hampir diseluruh dunia, dan Withworth (BSW) banyak dipakai di USA, Inggris atau Eropa. Dari gambar 2.17 menunjukkan gambar Ulir. Salah satu istilah yang paling penting yang digunakan adalah diameter luar. Dalam kasus ulir metrik, baut dinamai sesuai dengan diameter luarnya misalnya baut dengan 5 mm diameter luar dikenal sebagai baut M5. "Pitch" dari penampangnya adalah fitur lain yang penting dari ulir. Pitch atau kisar didefinisikan sebagai interval (jarak) antara sisi ulir satu dan

sebelahnya. Mur & baut harus memiliki pitch yang sama serta diameter yang sama jika mereka akan digunakan bersama-sama.



Gambar 2.10 Simbol Besi Ulir (*Drat*)

(Sumber <http://an-tika.blogspot.co.id/2011/08/pengetahuan-dasar-tentang-ulirdrat.html>)

BAB III. METODOLOGI PENELITIAN

3.1 Waktu dan Tempat

Waktu penelitian dilaksanakan selama enam bulan, dimulai 15 Januari 2016 sampai 4 Juni 2016. Penelitian dilakukan di Laboratorium Elektronika dan Terapan, serta Workshop Jurusan Teknik Elektro Universitas Jember.

3.2 Alat dan Bahan

Alat dan bahan yang digunakan sebagai penunjang dalam penelitian ini adalah sebagai berikut.

3.2.1 Pembuatan Sensor *Rotary Encoder*

1. Piringan/Disk
2. Optocoupler *rotary encoder*
3. LED
4. Resistor 10 K
5. Resistor 330
6. *Limite Switch*
7. Power Supply

3.2.2 Pembuatan Modul Relay 6 Channel

1. Relay G5S-1
2. *Diode 4004*
3. *Transistor BD 139*
4. *Optocoupler PC 817*
5. Resistor 10 K
6. Resistor 330
7. Resistor 2K7

3.2.3 Pembuatan Tampilan LCD

1. LCD 2x16
2. *Variabel resistor 5 K*
3. *Header*

3.2.4 *Software*

1. Proteus 7 Professional
2. Eagle PCB
3. Arduino IDE
4. Visual Basic 2010
5. AutoCAD 2010

3.2.5 *Output*

1. Motor AC dan Motor DC
2. LCD
3. LED
4. *Relay* 24V

3.2.6 *Pembuatan Power Supply*

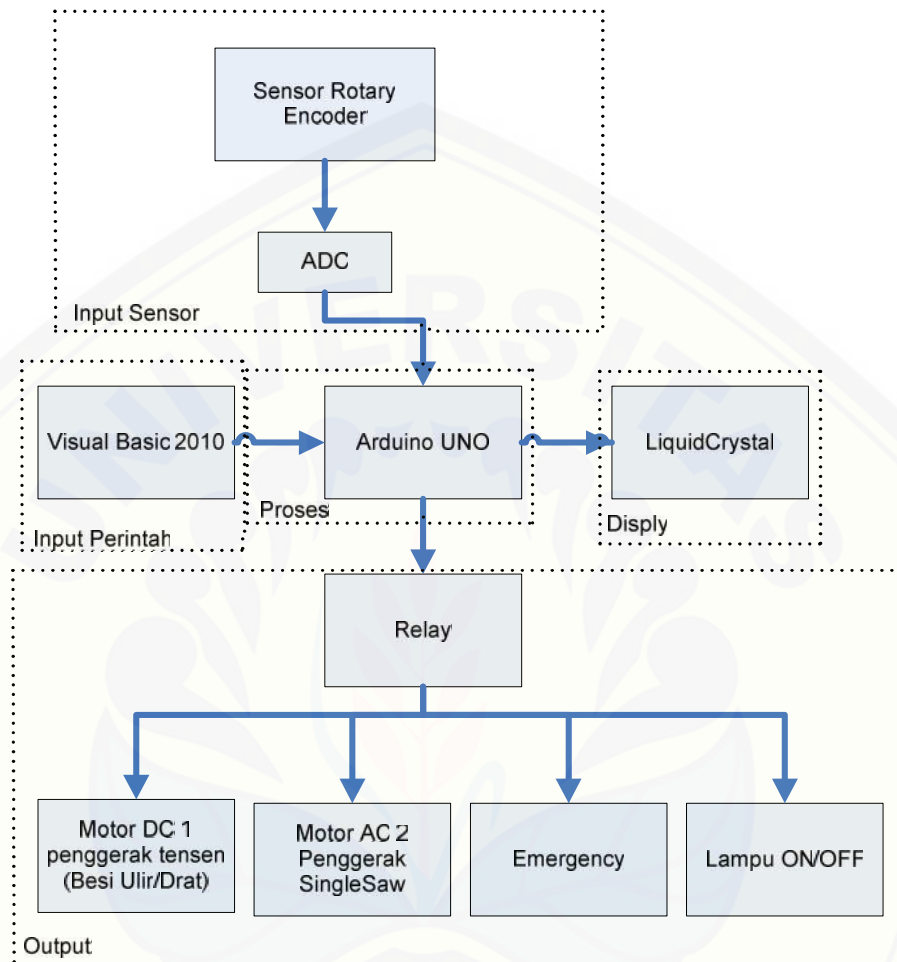
1. Trafo *step down* 3A
2. Fuse 3A
3. *Diode* 4004
4. *Capacitor* 4700 uF
5. Capacitor 0,1 uF
6. IC Regulator 7805
7. IC Regulator 7812
8. IC Regulator 7824
9. Transistor TIP 3055
10. *Header*
11. LED indikator

3.2.7 *Alat*

1. Solder
2. Timah
3. Avometer
4. Tang Kombinasi
5. Penyedot Timah

3.3 Blok Diagram Alat

3.3.1 Blok Diagram Kerja Dari Alat



Gambar 3.1 Blok Diagram Alat

Gambar 3.1 blok diagram alat *prototipe* sistem pemotongan *plywood* dengan perangkat *encoder* dan *interface visual basic* berbasis arduino, menjelaskan tentang bagian-bagian dari rangkaian yang tersusun menjadi satu sistem alat dengan sebuah mikrokontroler arduino yang menjadi pusat pengendali.

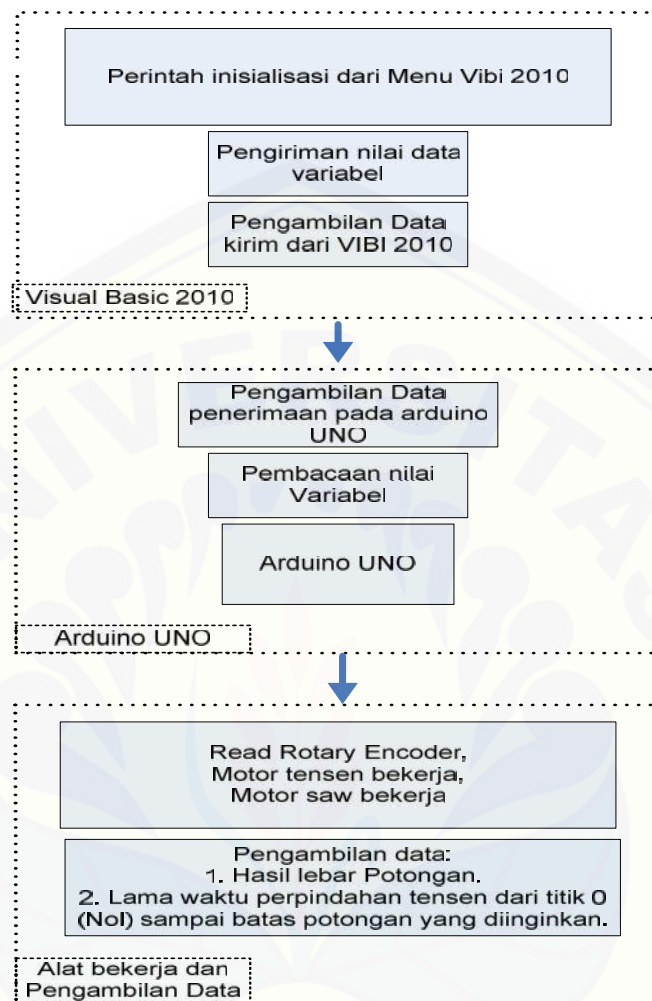
Bagian *input* pada blok diagram terdiri dari sensor *rotary encoder* sebagai pembacaan pulsa dari sensor *rotary encoder* untuk mengetahui posisi dari besi ulir. Bagian *input* data melalui *interface* dari *visual basic* 2010 menuju arduino UNO. Bagian *output* pada blok diagram diatas yaitu motor DC penggerak tensen

(besi drat/ulir), motor AC *singlesaw* yang dikontrol oleh *relay*. Dari diagram blok pada gambar terlihat bahwa alat yang akan dirancang terdiri dari beberapa bagian:

1. Bagian *input* sensor terdapat sensor *rotary encoder* yang berfungsi sebagai pembacaan, sensor *rotary encoder* yang digunakan sebagai pembacaan putaran dan posisi dari *tensen* (besi ulir/drat).
2. Bagian *input* data adalah program interface PC terdapat bagian kontrol menggunakan program *Visual Basic 2010* yang memberikan jenis-jenis menu lebar pemotongan. Interface ini terjadi secara *online* antara PC dan kontrol.
3. Bagian kontrol yaitu sebuah sistem *Arduino* yang berfungsi sebagai pusat kendali dari bagian *output*.
4. Bagian *output* LCD berfungsi untuk menampilkan informasi pembacaan hasil kalibrasi dari pembacaan *rotary encoder*.

Bagian *output* Relay digunakan untuk mengendalikan motor DC 1 penggerak *tensen*(besi ulir/drat), motor DC 2 penggerak motor *singlesaw*.

3.3.2 Blok Diagram Proses Pengambilan Data



Gambar 3.2 Proses Pengambilan Data

Gambar 3.2 merupakan gambar alur dari proses menjalankan alat dari mulai inialisasi hingga pengambilan data yang dilakukan. Pertama-tama yang dilakukan dalam proses menjalankan alat ialah memberikan perintah inialisasi dari PC menggunakan program *visual basic 2010* sehingga dapat mengirim nilai *hexadecimal* perintah data dari *visual basic 2010* menuju *arduino UNO*. Kemudian pengiriman ini menggunakan media *serial port (USB)* yang dikonvert menjadi nilai *decimal* pada bahasa program *arduino UNO*. Fungsi dari *arduino* ini sebagai kontrol yang diperintah terhadap penulisan bahasa program yang kemudian mengontrol relay motor *power window* (tensen) dan pembacaan sensor *rotary encoder* sampai dengan batas lebar potongan yang sesuai terhadap menu

perintah pada *visual basic 2010*. Jika pada saat pembacaan lebar potongan telah usai maka alat akan melakukan eksekusi lebar potongan, selanjutnya pengambilan data hasil lebar potongan dimulai dari lebar potongan 5 cm, 10 cm, 20 cm, 25 cm, 30 cm, 40 cm. Dan lama waktu dari setiap perpindahan dari titik 0 (nol) lebar botongan sampai dengan lebar potongan yang diperintahkan pada setiap pengujian.

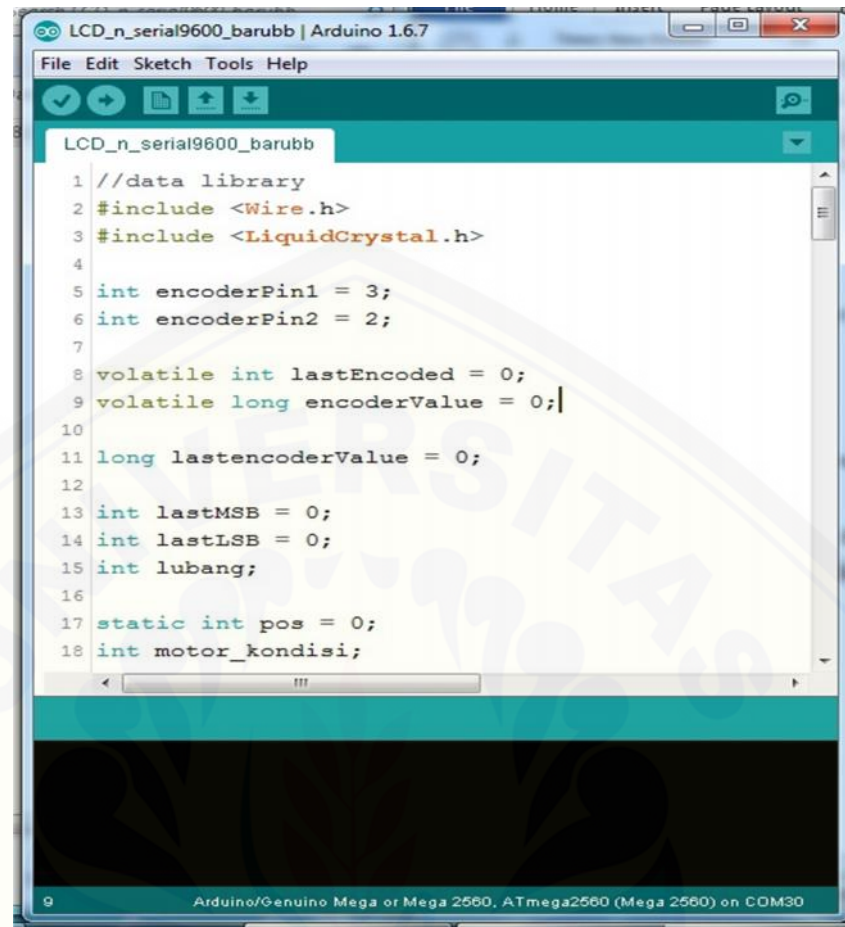
3.4 Perancangan Sistem

3.4.1 Perancangan *Software*

Software atau perangkat lunak yang digunakan untuk proses kerja menyalakan dan mematikan ini terdiri dari beberapa jenis *software*, diantaranya menggunakan *software* yang digunakan pada pengendali Tugas Akhir ini diantara lain Arduino dan *Visual Basic 2010*.

a. Arduino

Dalam sistem ini, *user* yang melakukan atau mengontrol *input* dari aplikasi kendali yang berkomunikasi terhadap perangkat komputer dengan memanfaatkan serial komunikasi. Data *input* yang diterima oleh arduino merupakan data pemanggilan berupa nilai *variable hexadecimal* untuk melakukan perintah kontrol mesin. Kemudian data serial tersebut diterjemahkan kembali oleh mikrokontroler arduino menjadi nilai *decimal* sehingga mengurangi nilai *delay* pada saat pengiriman tersebut. Bentuk komunikasi yang diterima pada rangkaian arduino tersebut berupa indikator LED dari RX dan TX dimana LED akan blink secara sekilas pada saat menerima data dari perangkat komputer. Output dari arduino terhubung terhadap relay sebagai perantara untuk mengaktifkan peralatan lain seperti motor-motor pada mesin.

The image shows a screenshot of the Arduino IDE interface. The title bar reads "LCD_n_serial9600_barubb | Arduino 1.6.7". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". The main text area contains the following code:

```
LCD_n_serial9600_barubb
1 //data library
2 #include <Wire.h>
3 #include <LiquidCrystal.h>
4
5 int encoderPin1 = 3;
6 int encoderPin2 = 2;
7
8 volatile int lastEncoded = 0;
9 volatile long encoderValue = 0;|
10
11 long lastencoderValue = 0;
12
13 int lastMSB = 0;
14 int lastLSB = 0;
15 int lubang;
16
17 static int pos = 0;
18 int motor_kondisi;
```

The status bar at the bottom indicates "Arduino/Genuino Mega or Mega 2560, ATmega2560 (Mega 2560) on COM30".

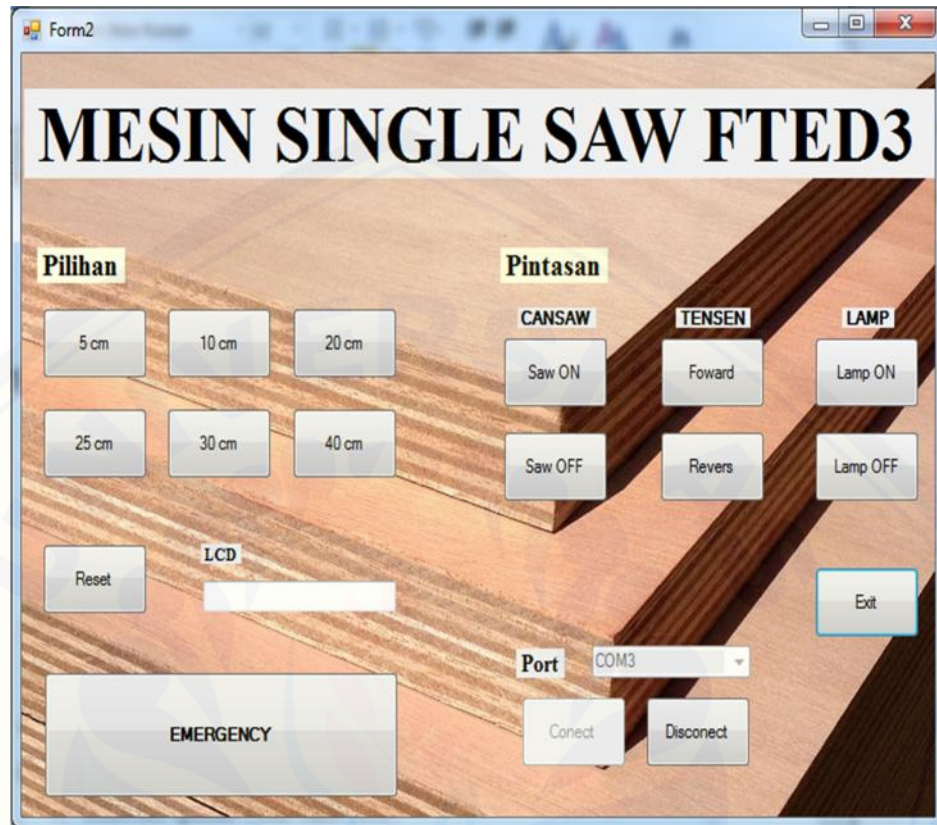
Gambar 3.3 *Listing* Program Arduino UNO

b. *Visual Basic* 2010

Pembuatan *layout* kontrol pada halaman utama dibuat seperti yang telah digambarkan dalam rancangan *interface*. Langkah-langkah pembuatan aplikasi pada *visual basic* 2010 sebagai berikut :

1. Membuka aplikasi *visual basic* 2010.
2. Pilih bahasa program yang akan digunakan.
3. Rancang *layout* peletakan sebagai kendali (botton), serta menamainya.
4. Kemudian melakukan penulisan *listing* program C++ pada halaman program.
5. Melakukan pengujian komunikasi terhadap *interface* komputer menuju arduino.
6. Mengkonvert aplikasi tersebut menjadi aplikasi komputer.

Hasil dari *layout* yang telah dibuat dalam bentuk aplikasi .Exe pada komputer, tampak seperti gambar berikut :



Gambar 3.4 *Layout* Kontrol

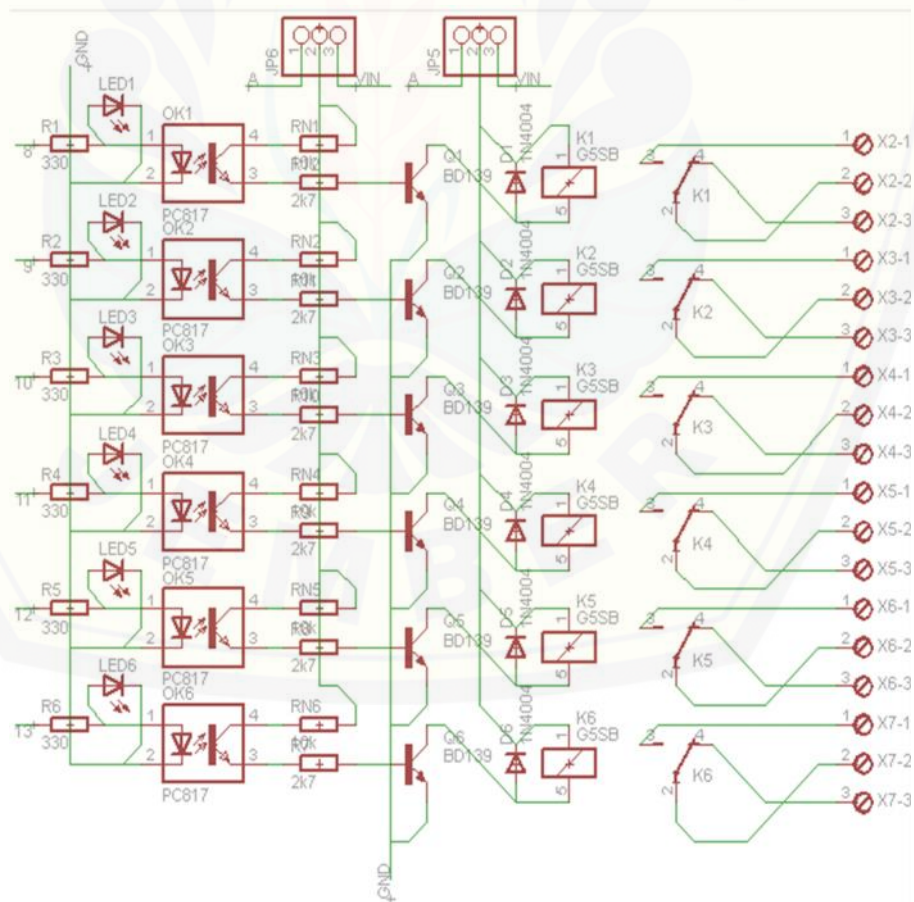
Gambar diatas merupakan gambar hasil rancangan yang telah dibuat pada pada program *visual basic* 2010. Dimana didalam menu aplikasi tersebut terdapat tombol menu pilihan lebar potongan(5 cm, 10 cm, 20 cm, 25 cm, 30 cm, 40 cm), tombol posisi untuk mereset posisi lebar potongan, tombol sistem manual untuk menghidupkan motor *saw*, mengendalikan putaran motor tensen *foward* dan *reverse*, tombol menyalakan lampu, serta tombol pengendali *emergency* pada saat keadaan berbahaya. Dari aplikasi yang dibuat terdapat pula pilihan serial port komunikasi yang dapat dipilih agar dapat di kendalikan tersebut.

3.4.2 Rangkaian Modul Relay 6 Channel

Modul relay 6 *channel* ini merupakan sebuah pengontrol *output* yang menyala serta mati dan pengendali kondisi maju dan mundur, dapat dijabarkan ketika normal relay pada kondisi *normally open*, dan ketika sensor terdeteksi,

arduino memproses kendali untuk membuat *pin output* berkondisi *high*. Pada kondisi ini, relay dipicu untuk *switching* dari *normally open* menjadi *normally close*.

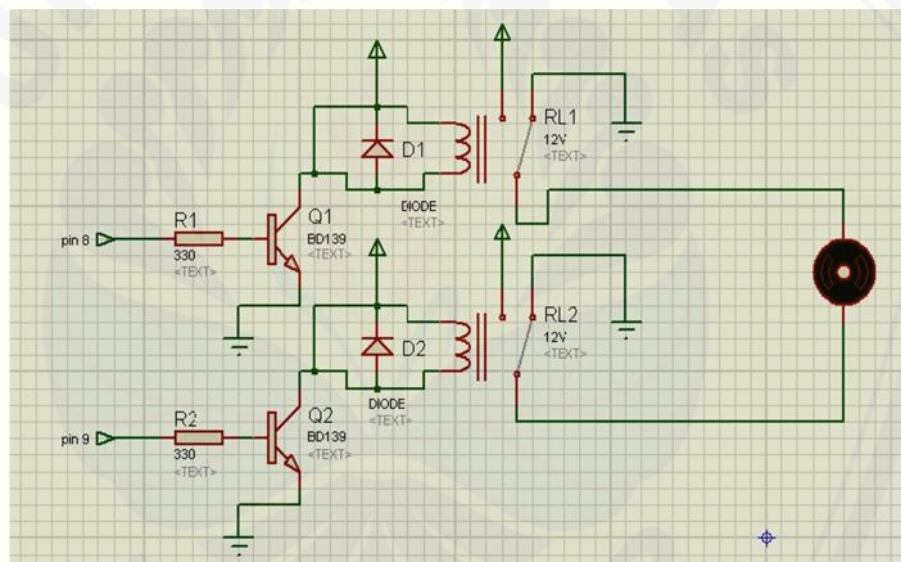
Pada fungsi kerja masing-masing *relay* pada rangkaian yakni pada rangkaian *relay* pertama dan *relay* kedua yaitu mengendalikan putaran tensesen (besi ulir) agar dapat mengendalikan motor *power window* maju dan mundur. Kemudian rangkaian *relay* ketiga mengendalikan motor *saw*. Pada rangkaian *relay* keempat yakni mengendalikan sistem kerja dari pengaman atau *sefty* yang berguna untuk pengaman pada saat motor *saw* serta dan motor *power window* bekerja sehingga dimatikan secara paksa. Rangkaian *relay* kelima digunakan untuk menhidupkan lampu. Rangkaian *relay* keenam digunakan sebagai rangkaian cadangan pada saat *relay* pada keadaan rusak.



Gambar 3.5 Rangkaian Modul Relay 5 Channel

3.4.3 Rangkaian *Driver Foward-Reverse Motor Power Window*

Rangkaian yang digunakan sebagai pengendali motor *power-window* agar dapat berputar maju dan mundur (*foward-reverse*) yaitu menggunakan rangkaian pengendali dua *relay* sebagai *switching* yang bekerja secara bergantian yang dikendalikan oleh perintah *arduino UNO*. Pin *arduino UNO* yang digunakan sebagai pengendali pada rangkaian *foward-reverse* pada rangkaian ini menggunakan pin 8 dan pin 9 kemudian membangkitkan kopling dari aptocopler dan merespon transistor BD139 agar men-*swicthing* relay untuk mengendalikan motor *power window*. Dibawah ini merupakan gambar sederhana pada saat rangkaian memutar motor maju dan memutar motor mundur. Dapat dilihat sebagai berikut :



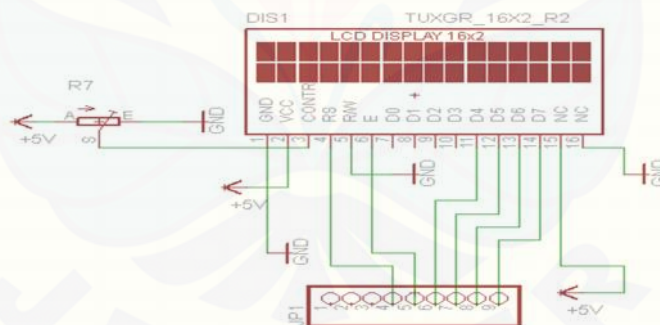
Gambar 3.6 Rangkaian Modul Relay Pengendali *Foward-Reverse* motor *power window*

Gambar 3.6 diatas merupakan rangkaian relay pengendali *foward-reverse* motor *power window* yang dikontrol oleh *arduino UNO*. Pada saat kondisi perintah *foward*, maka *arduino UNO* memerintah pada pin 8 untuk aktif dan pin 9 untuk tidak aktif sehingga penguatan arus yang terjadi pada *relay 1* sehingga pada awalnya NO (*Normali Open*) menjadi NC (*Normali Close*) maka potensial positif motor mendapatkan respon tegangan positif dan sedangkan *relay 2* menahan pada posisi NO (*Normali Open*) dari pengendalian ini maka potensial motor negatif mendapat respon tegangan negatif, sehingga motor berputar ke arah kanan. Begitu

pula sebaliknya pada saat *arduino UNO* memerintah pada pin 8 untuk tidak aktif dan pin 9 untuk aktif sehingga penguatan arus yang terjadi pada relay 2 sehingga pada awalnya NO (*Normali Open*) menjadi NC (*Normali Close*) maka potensial motor negatif mendapat respon tegangan positif dan sedangkan sedangkan relay 1 menahan pada posisi NO (*Normali Open*) maka potensial positif motor mendapatkan respon tegangan negatif, sehingga motor berputar ke arah kiri. Ini adalah proses dari pengendalian motor *power window* yang menggerakkan besi ulir agar tensen pembatas lebar potongan bekerja dari titik 0 sampai pemotongan yang kita inginkan, begitu pula sebaliknya dari potongan yang di inginkan kembali ke titik 0.

3.4.4 Rangkaian LCD

Rangkaian LCD pada sistem ini dipakai sebagai penampil informasi kinerja sistem pengendalian alat. Kerja alat LCD tersebut menampilkan nilai pulsa dari putaran motor, dimana nilai putaran ini dihasilkan oleh *sensor encoder*.

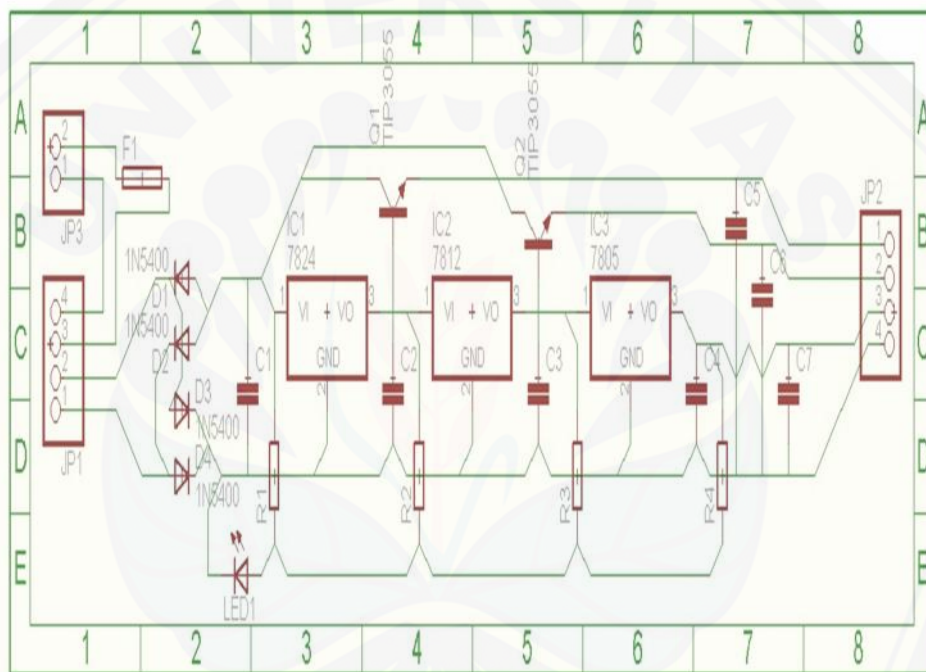


Gambar 3.7 Rangkaian LCD

Didalam rangkaian LCD tersebut terdapat bagian-bagian komponen diantaranya adalah LCD, *Variable Resistor* bernilai 10k , diode 1A, serta pin yang menuju *arduino UNO* yaitu pin yang digunakan adalah pin 7 menuju reset, pin 6 menuju *enable*, untuk pin 5, 4, A3, A4 adalah komunikasi data pada LCD.

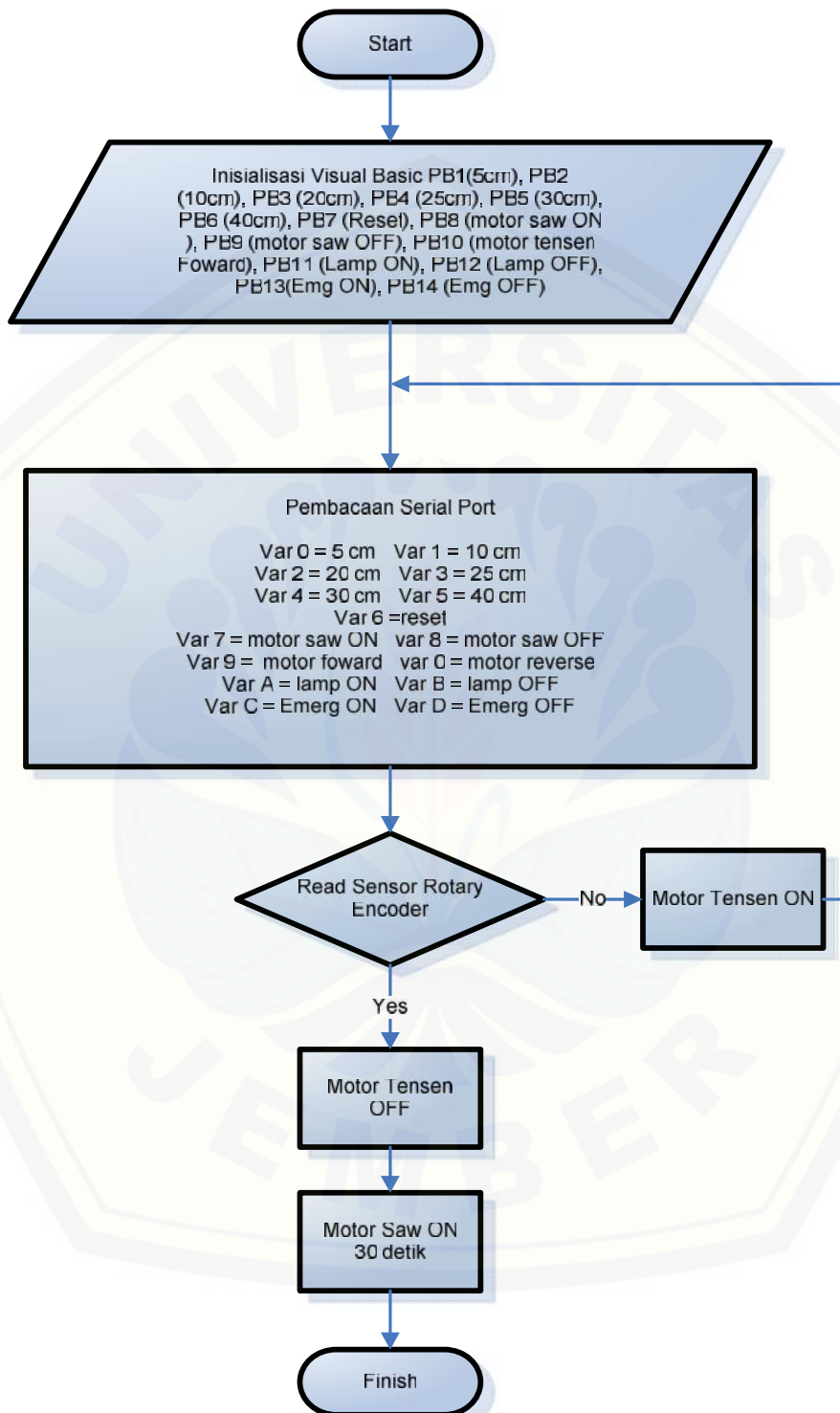
3.4.5 Rangkaian *Power Supply*

Power supply merupakan sumber tegangan atau V_{cc} , yang digunakan untuk memberikan tegangan pada mikrokontroler, sensor maupun LCD agar dapat dioperasikan sesuai fungsinya. Tegangan yang dibutuhkan pada alat ini yaitu sebesar 5V, 12V, dan 24V. IC regulator yang digunakan 7805, 7812, dan 7824 sebagai pembatas tegangan *output* dengan penambahan rangkain penguatan arus dengan transistor TIP3055. Berikut rangkaian *power supply* pada gambar 3.5.



Gambar 3.8 Rangkaian *Power Supply*

3.5 Diagram Alir



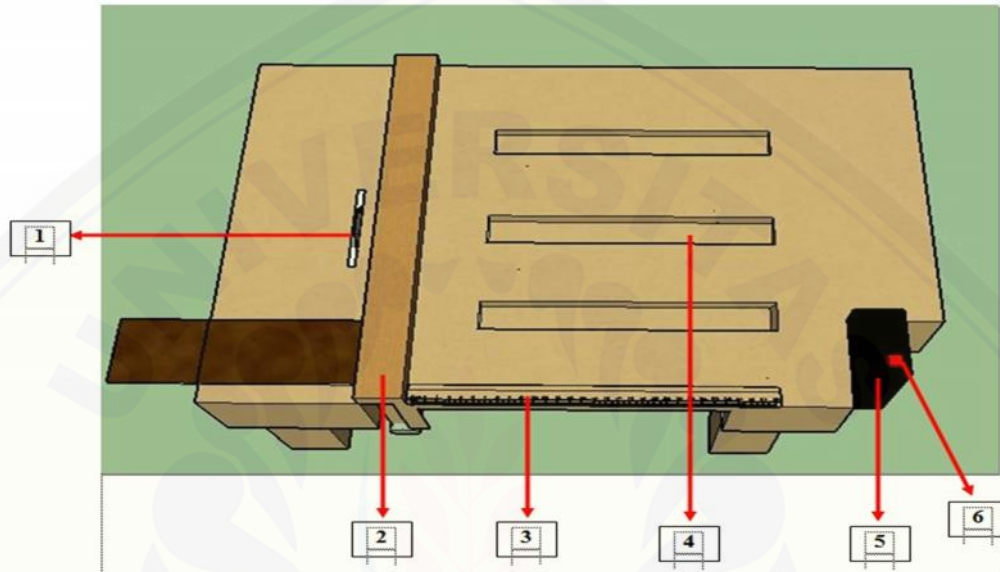
Gambar 3.9 Flowchart

Gambar 3.5 di atas menunjukkan proses mengendalikan alat, pada saat *start* maka langkah awal yang akan dilakukan adalah menginisialisasi perintah *menu* dari *visual basic* 2010 yaitu menu jenis potongan yang dikerjakan secara otomatis, dan dapat pula secara manual untuk mengaktifkan motor saw, motor tensen (motor *power window*) dan lampu. Pada saat memilih jenis lebar potongan yang terdapat pada *visual basic* 2010 maka serial komunikasi arduino membaca *variable* yang masuk sehingga dapat melakukan eksekusi perintah lebar pemotongan dengan melanjutkan pembacaan nilai putaran dari sensor *rotary encoder* dengan dibantu oleh putaran *reverse* pada motor tensen (motor *power window*) sampai dengan nilai putaran sesuai terhadap hasil perintah dari *arduino* sehingga *arduino* memerintah motor tensen (motor *power window*) berhenti, Namun jika pembacaan sensor *rotary encoder* belum sesuai dengan nilai yang diinginkan maka pembacaan sensor terus terulang dan perintah untuk mengaktifkan motor tensen (motor *power window*) terus berjalan. Selanjutnya pembacaan sensor *rotary encoder* sesuai dan motor tensen (motor *power window*), maka secara otomatis motor *saw* berputar untuk melakukan lebar potongan selama waktu 30 detik. Begitu pula pada jenis menu manual yaitu motor *saw* dapat dikendalikan manual antara hidup dan mati, tensen dapat diatur manual *foward* dan *reverse*, lampu dapat di pilih antara hidup dan mati secara manual.

3.6 Perancangan Mekanik Alat

3.6.1 Desain Mekanik Alat

Gambar 3.6 dibawah ini menunjukkan perancangan mekanik dari alat prototipe sistem pemotongan *plywood* dengan perangkat *encoder* dan *interface* visual basic berbasis arduino uno.



Gambar 3.10 Perancangan Mekanik Alat Pemotongan *Plywood*

Pada perancangan mekanik gambar 3.10 di atas, bagian-bagian yang digunakan antara lain :

1. *Singlesaw*, alat pemotong yang berbentuk mata cutting well mesin grenda namun memiliki mata seperti gergaji dan berfungsi sebagai pemotong *plywood*. *Singlesaw* ini digerakan oleh motor AC mempunyai putaran tinggi.
2. Balok kayu, pembatas lebar potongan yang dimulai dari titik 0.
3. Penggaris, batas ukuran dari lebar potongan.
4. Besi ulir (*Drat*), besi ulir yang diputar oleh motor dengan bantuan rantai agar mendorong dan menarik plat sebagai pembatas samping dari pemotongan *plywood*.
5. *Control Panel*, pengendali dari alat yang dikendalikan oleh Arduino UNO dan *relay*.

6. *Emergency switch*, tombol yang digunakan pada saat keadaan berbahaya agar dapat mematikan seluruh kerja mesin secara cepat.

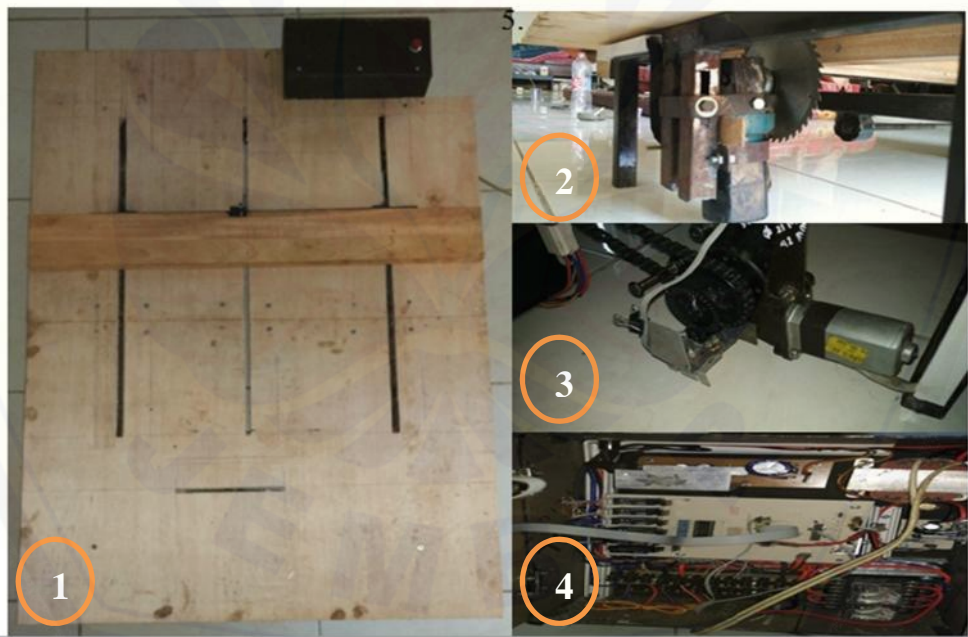
3.6.2 Bahan dan Bentuk Alat

➤ Bahan Mekanik Robot

1. Besi kotak dengan tebal 1,5 mm.
2. Besi Ulir.
3. *Saw Syrcle*.
4. *Plywood* dengan tebal 8 mm.

➤ Bentuk Mekanik Alat

Bentuk mekanik alat berbentuk seperti balok, bentuk tersebut dibuat semirip mungkin dengan bentuk mesin single saw pada umumnya. Bentuk mekanik alat dapat dilihat pada Gambar 3.8.



Gambar 3.11 Hasil Perancangan Mekanik Alat Pemotongan *Plywood*

Pada perancangan mekanik gambar 3.8 di atas, bagian-bagian yang digunakan antara lain :

1. Bentuk mekanik yang terlihat dari atas, dimana berbentuk meja dengan pembatas.

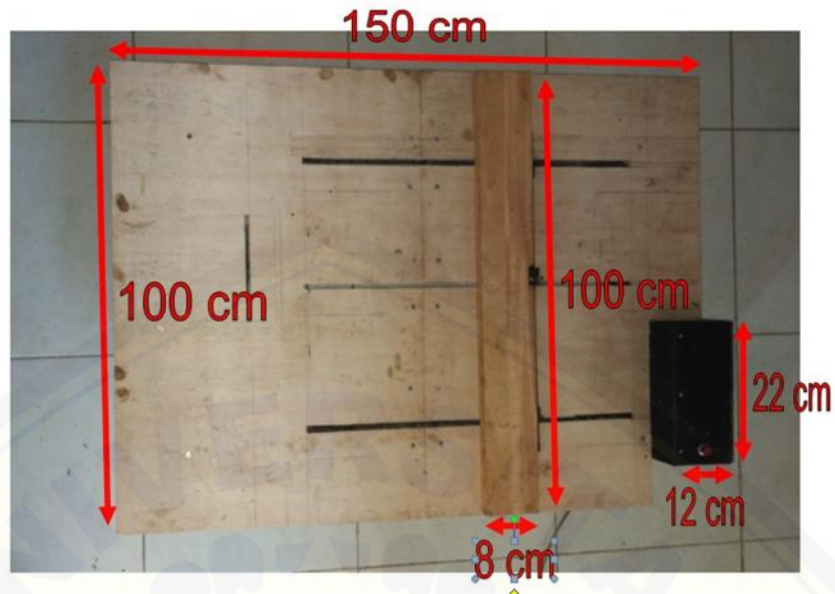
2. Motor *single saw* yang menggunakan motor mesin jahit, dan menggunakan mata graji (*sircle*) yang tajam untuk melakukan pemotongan.
3. Peletakan sensor yang terletak *As* pada putaran motor power *window*, sehingga dapat dihitung putarannya.
4. Rangkaian elektrik pengendali yang terdapat pada panel.

3.6.3 Hasil Perancangan Mekanik Alat dari Pemotong *Plywood*

Didalam desain mekanik yang digambarkan awal, memiliki perbedaan. Yaitu terletak pada motor-motor yang digunakan pada desain awal yang menggunakan sistem belt untuk menghubungkan pada beban namun menggunakan sistem tranfer kopling dan sistem rante kamprat agar mengurangi losis perhitungan pada nilai putaran yang akan dikalibrasikan terhadap bahasa program. Sistem mekanik pada pemotongnya (*sircle/saw*) menggunakan mekanik yang dapat dinaik dan turunkan sesuai tebal dari *plywood*, sehingga dapat menyesuaikan dari hasil potongan yang diinginkan. Selanjutnya pada bagian *sefty* pada pengoprasian sangatlah penting agar mengurangi tingkat kecelakaan yang terjadi pada saat pengerjaan ataupun saat mengujian pada pemotongan, yaitu menggunakan tombol *emergency* yang dapat diaktifkan melalui software begitu pula pada manual. Pada penerapan sistem yang terpasang pada alat yang saya gunakan ini menggunakan rancangan yang berfungsi agar mekanik alat serta elektrik alat dapat dengan mudah digunakan serta dapat memproteksi dalam keadaan berbahaya dan mengurangi losis dari kerugian pembacaan nilai putaran yang terjadi pada putaran tensen agar menentukan titik lebar potongan yang diperintahkan.

➤ Ukuran dari Mekanik Alat Pemotonga *Plywood* Tampak Atas

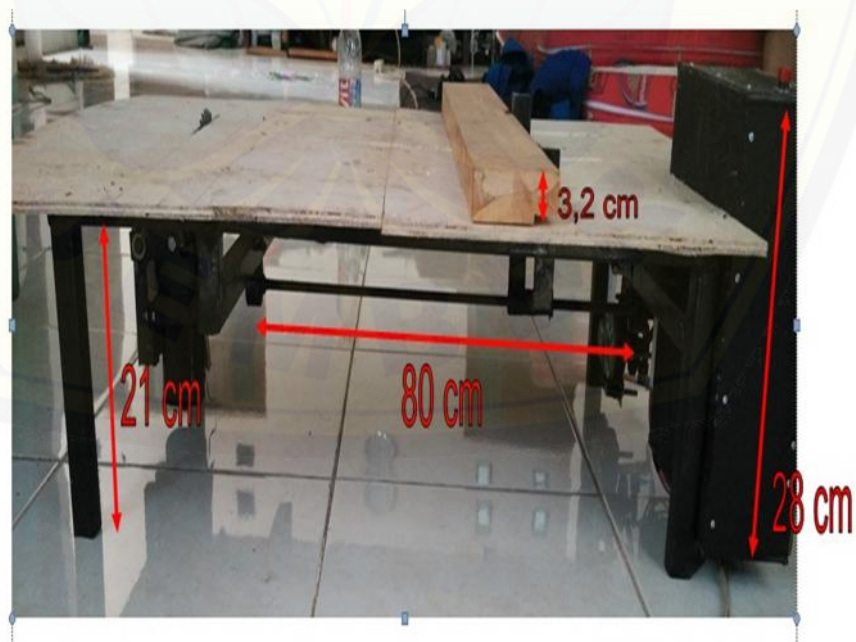
Bentuk mekanik alat pemotong *plywood* secara keseluruhan yang dilihat dari sisi atas serta spesifikasi dari ukuran yang ditunjukkan dapat dilihat pada gambar dibawah ini.



Gambar 3.12 Ukuran Mekanik Pemotong *Plywood* Tampak Atas

➤ Ukuran dari Mekanik Alat Pemotong *Plywood* Tampak Samping

Bentuk mekanik alat pemotong *plywood* secara keseluruhan yang dilihat dari sisi samping serta spesifikasi dari ukuran yang ditunjukkan dapat dilihat pada gambar dibawah ini.



Gambar 3.13 Ukuran Mekanik Pemotong *Plywood* Tampak Samping

BAB IV. HASIL DAN PEMBAHASAN

Di dalam proses produksi *plywood* besar kita tidak asing dengan bentuk mesin pemotong yang bekerja untuk membelah *plywood* menjadi dua bagian atau pun beberapa bagian sesuai standard pemotongan yang ditetapkan oleh perusahaan *plywood*. Mesin yang kami rancang merupakan bagian kecil bentuk mesin pemotong *plywood* dengan bentuk perintah pemotongan 2 bagian yang sederhana. Dimana kerja dari mesin ini ditentukan terhadap perintah potongan yang ditetapkan oleh suatu industri. Pada halaman pembahasan ini menjelaskan mengenai proses melakukan pengujian alat secara keseluruhan dari bentuk hasil tombol pilihan lebar potongan *plywood* dari *software visual basic*, ukuran lebar potongan *plywood*, waktu yang di tempuh pada saat perpindahan jenis potongan lebar dari titik 0 sampai dengan ukuran lebar yang ditentukan. Proses pengujian sistem meliputi bagian-bagian yang merupakan hasil dan pembahasan sebagai berikut.

4.1 Pengujian Tombol Pilihan dari Software Vibi 2010

Pengujian pada bagian ini merupakan salah satu bentuk perintah sebagai menu pilihan untuk memilih bentuk lebar potongan dari *plywood*. Bentuk pilihan yang digunakan menggunakan *interface* program *visual basic* 2010 yang telah diprogram sebagai sarana untuk memilih lebar potongan dari *plywood* yang dapat bekerja secara otomatis dan beberapa menu khusus sebagai sarana pilihan manual. Komunikasi yang digunakan menggunakan komunikasi USB yang terhubung dari PC menuju mikrokontrol *Arduino UNO*, komunikasi ini hanya melakukan tranfer data berupa penetapan nilai *Hexadecimal* yang kemudian diterima oleh serial komunikasi dari *Arduino UNO* dan kemudian *Arduino UNO* memberikan perintah untuk mengaktifkan relay yang berada pada *panel* dan membaca pulsa pada sensor *rotary encoder*. Pengujian yang dilakukan adalah pembacaan dari komunikasi *visual basic* 2010 menuju *arduino UNO*, kemudian perintah yang masuk terhadap *arduino UNO*. Dapat dilihat pada gambar 4.1 tampilan dari bentuk *visual basic* 2010 yang telah dibuat :



Gambar 4.1 Tampilan pengendali menggunakan *visual basic 2010*

Tabel 4.1 Pengujian data yang dikirim oleh *visual basic 2010* dan diterima oleh *arduino UNO*

No.	Menu	Data tranfer Vibi 2010 (Hexadecimal)	Data terima Arduino (Decimal)	Hasil Pengujian	
				Terbaca	Tidak Terbaca
1.	5 cm	"0"	"49"		-
2.	10 cm	"1"	"50"		-
3.	20 cm	"2"	"51"		-
4.	25 cm	"3"	"52"		-
5.	30 cm	"4"	"53"		-
6.	35 cm	"5"	"54"		-
7.	Reset	"6"	"55"		-
8.	Emerg On	"H"	"73"		-
9.	Emerg OFF	"G"	"72"		-
10.	Saw On	"A"	"66"		-
11.	Saw Off	"B"	"67"		-
12.	Foward	"C"	"68"		-
13.	Revers	"D"	"69"		-
14.	Lamp On	"E"	"70"		-
15.	Lamp Off	"F"	"71"		-

BAB V. PENUTUP

5.1 Kesimpulan

Dari hasil pembuatan, pengujian perangkat dan pembahasan dapat diambil kesimpulan sebagai berikut :

1. Dari bentuk komunikasi *interface* yang dilakukan pada pengujian antara *visual basic 2010* menuju *arduino UNO* mengalami pembacaan yang baik yang diterima oleh *arduino UNO*.
2. Dari hasil pengujian lebar pemotongan didapat rata-rata dari keseluruhan hasil percobaan lebar potongan diantaranya lebar potongan 5 cm ialah 4,5 cm, 10 cm ialah 10,0 cm, 20 cm ialah 20,0 cm, 25 cm ialah 24,9 cm 30 cm ialah 29,5 cm, 40 cm ialah 38,5 cm. Nilai lebar potongan yang memiliki nilai jauh sesuai pemotongan terjadi pada lebar pemotongan 40 cm yaitu 38,5 yang berdampak pada rata-rata nilai error persen yaitu 2,7%. Maka nilai untuk mendapatkan batas nilai lebar potongan yang baik dari lebar potongan 5 cm, 10 cm, 20 cm, 25 cm, dan 30 cm.
3. Hasil pengujian lama waktu dari kerja tensen untuk memindahkan batas lebar lebar potongan dari titik 0 sampai dengan jenis lebar potongan dengan 3 kali pengujian maka di dapat pada saat pengujian perpindahan dari titik 0 ke 5 cm didapat rata-rata waktu 15,30 detik, kemudian pada saat 0 ke 10 cm didapat rata-rata waktu 31,70 detik, pada saat 0 ke 20 cm didapat rata-rata waktu 58,62 detik, pada saat 0 ke 25 cm didapat rata-rata waktu 1:07.30 detik, , pada saat 0 ke 30 cm didapat rata-rata waktu 1:19.24 detik, , pada saat 0 ke 40 cm didapat rata-rata waktu 1:58.18 detik. Pada saat proses perpindahan batas lebar potongan bekerja dengan baik.

5.2 Saran

Berdasarkan penelitian yang telah dilakukan tentang “*Prototipe Sistem Pemotongan Plywood Dengan Perangkat Encoder Dan Interface Visual Basic Berbasis Arduino Uno*” penulis memberikan saran berikut dengan harapan untuk penyempurnaan karya ilmiah ini dan lebih memberikan manfaat yang lebih baik dimasa mendatang :

1. Perlu adanya tambahan tensen (besi ulir) yang memiliki kerenggangan gigi yang dapat mempercepat waktu geser pada batas lebar potongan *plywood*.
2. Memiliki meja sebagai alas yang rata serta pembatas lebar potongan yang lebih rata dan sempurna agar dapat memiliki hasil lebar potongan yang lebih presisi.
3. Motor *saw* yang memiliki kecepatan yang sangat tinggi agar mendapat potongan yang lebih rapi.
4. Memiliki *sefty* yang lebih untuk melakukan operasi dari mesin pemotong *plywood* tersebut.
5. Operator pengendali mesin *single saw* memiliki password khusus untuk menjalankan mesin tersebut.

DAFTAR PUSTAKA

- A.Winoto. 2010. *Mikrokontroler AVR ATmega 328/16/32/8535*. Bandung.
- Andi, Wahana. 2000. *Pemrograman Visual Basic 979/533/601/0*. Yogyakarta.
- Sasongko, Hari, Bagus. 2012. *Pemrograman Mikrokontroler dengan Bahasa C++*. ANDI.Yogyakarta
- Dimas Agung Nurcahyo, Tejo Sukmadi, Karnoto. 2014. *Aplikasi PLC pada Mesin Industri Pemotong Kayu dengan Perangkat Konveyor*.
- Djuandi, Feri. 2011. *Pengenalan Arduino*. www.tobuku.com. diakses 8 januari 2014
- Ball, Stuart R. 2003. Exploring optical and magnetic sensors. www.embedded.com/design/prototyping-and-development/4024586/Exploring-optical-and-magnetic-sensors. Diakses pada 23 Juni 2016.
- Muhammad Aldrin. 2011. *Mengukur Kecepatan dengan Rotary Encoder*. <http://all-thewin.blogspot.co.id/2011/02/mengukur-kecepatan-dengan-rotary.html>. Diakses pada 26 Desember 2015.
- Arwindra Rizqiawan. Juni 2009. Sekilas Rotary Encoder. <https://konversi.wordpress.com/2009/06/12/sekilas-rotary-encoder/>. Diakses pada 11 Januari 2016.
- Atom. 2014. Mikrokontroler Arduino UNO. <http://dialogsimponi.blogspot.com/2014/11/normal-0-false-false-false-in-x-none-x.html>. Diakses pada 11 Januari 2016.
- Meri Wardana. 2013 .Prinsip Kerja Motor. <http://meriwardana.blogspot.co.id/2011/11/prinsip-kerja-motor-arus-searah-dc.html>. Diakses pada 3 Maret 2016.
- Wangready. 2012. *Kontrol posisi motor dc menggunakan rotary encoder berbasis mikrokontroler*. <https://wangready.wordpress.com/2012/03/03/kontrol-posisi-motor-dc-menggunakan-rotary-encoder-berbasis-mikrokontroler-avr/>. diakses pada 03 Maret 2012.
- Didit Ardi. 2011. Pengetahuan Dasar Tentang Ulir (Drat). <http://antika.blogspot.co.id/2011/08/pengetahuan-dasar-tentang-ulirdrat.html>. Diakses pada 3 Maret 2016

Meriwardana. 2011. *Prinsip Kerja Motor Arus Searah (DC)*. <http://meriwardana.blogspot.co.id/2011/11/prinsip-kerja-motor-arus-searah-dc.html>. Diakses pada November 2011.

Wiwik. 2010. Motor (Dinamo) Control AC. <https://wiwike.wordpress.com/motor-dinamo-control/>. Diakses pada 5 April 2016.



Lampiran 1. Listing Program

1. Listing Program Arduino

```
//data library
#include <Wire.h>
#include <LiquidCrystal.h>

int encoderPin1 = 3;
int encoderPin2 = 2;

volatile int lastEncoded = 0;
volatile long encoderValue = 0;

long lastencoderValue = 0;

int lastMSB = 0;
int lastLSB = 0;
int lubang;

static int pos = 0;
int motor_kondisi;
boolean benar = false;

//tampil LCD
LiquidCrystal lcd(7, 6, 5, 4, A3, A4);

//kontrol relay
const int m1a = 8; const int m2a = 9; const int m3a = 10;
const int m4a = 11; const int m5a = 12; const int m6a = 13;
const int limitSw = A5;
```

```
int val,sawVal,lampVal,emVal;
```

```
void setup()
```

```
{  
  //setting LCD  
  lcd.begin(16, 2);  
  Serial.begin(9600);  
  pinMode(encoderPin1, INPUT_PULLUP);  
  pinMode(encoderPin2, INPUT_PULLUP);  
  pinMode(limitSw,INPUT);  
  
  //setting pin sebagai output  
  pinMode(m1a,OUTPUT);  
  pinMode(m2a,OUTPUT);  
  pinMode(m3a,OUTPUT);  
  pinMode(m4a,OUTPUT);  
  pinMode(m5a,OUTPUT);  
  pinMode(m6a,OUTPUT);  
  attachInterrupt(0, updateEncoder, CHANGE);  
  attachInterrupt(1, updateEncoder, CHANGE);  
  lcd.setCursor(0, 0);  
  lcd.print("TANGAMES BEH");  
}
```

```
void loop()
```

```
{  
  kontrol_interface();  
}
```

```
void otomatis(int panjang){
```

```
digitalWrite(m3a,LOW);
encoderValue=0;
lubang=0;
while(1){
  motor_kanan();
  lcd.setCursor(0, 1);
  lcd.print(lubang);
  lcd.print(" ");
  if(lubang>=(panjang*11)){motor_stop();break;} //11
}
digitalWrite(m3a,HIGH);
delay(15000);// waktu saw menyala
digitalWrite(m3a,LOW);

}

void updateEncoder(){
  int MSB = digitalRead(encoderPin1); //MSB = most significant bit
  int LSB = digitalRead(encoderPin2); //LSB = least significant bit

  int encoded = (MSB << 1) |LSB; //converting the 2 pin value to single number
  int sum = (lastEncoded << 2) | encoded; //adding it to the previous encoded
value

  if(sum == 0b1101 || sum == 0b0100 || sum == 0b0010) encoderValue ++;
  if(sum == 0b1110 || sum == 0b0111 || sum == 0b0001) encoderValue --;

  lastEncoded = encoded; //store this value for next time
  lubang=encoderValue*100/310;
}
```

```
void motor_stop()                //kendali tensen
{digitalWrite(m1a,HIGH); digitalWrite(m2a,HIGH);}
```

```
void motor_kiri()                //kendali tensen
{digitalWrite(m1a,HIGH); digitalWrite(m2a,LOW);}
```

```
void motor_kanan()              //kendali tensen
{digitalWrite(m1a,LOW); digitalWrite(m2a,HIGH);}
```

```
void kontrol_interface()
{
val=Serial.read()+1;
switch(val){
  case 49 :{otomatis(50);break;}//5 mm
  case 50 :{otomatis(100);break;}//100 mm
  case 51 :{otomatis(200);break;}//206 mm
  case 52 :{otomatis(250);break;}//260 mm
  case 53 :{otomatis(300);break;}//313 mm
  case 54 :{otomatis(366);break;}//366 mm

  case 55 :{rst();break;}

  case 72 :{emVal=0;break;} //kendali emergen off
  case 73 :{emVal=1;break;} //kendali emergen on
  case 66 :{sawVal=1;break;} //kendali secle on
  case 67 :{sawVal=0;break;} //kendali secle off
  case 70 :{lampVal=1;break;} //kendali lampu on
```

```
case 71 :{lampVal=0;break;} //kendali lampu off
case 68 :{motor_kiri();break;} //motor kiri
case 69 :{motor_kanan();break;} //motor kanan
default :{motor_stop();break;} //motor stop
}
```

```
if(sawVal==1){digitalWrite(m3a, HIGH);}
else{digitalWrite(m3a, LOW);}
```

```
if(lampVal==1){digitalWrite(m5a,HIGH);}
else{digitalWrite(m5a, LOW);}
```

```
if(emVal==1){digitalWrite(m4a,HIGH);}
else{digitalWrite(m4a, LOW);}
```

```
//lcd.setCursor(0, 1);
//lcd.print(val);
delay(500);
}
```

```
void rst(){
  int jj = 0;
  while(1){
    jj = digitalRead(limitSw);
    motor_kiri();
    if (jj==HIGH){motor_stop();break;}
  }
}
```

2. Listing Program Microsoft Visual Basic

Imports System.IO

```
Imports System.IO.Ports
Imports System.Threading
```

```
Public Class Form2
```

```
    'setting global
```

```
    Shared _continue As Boolean
```

```
    Shared myport As SerialPort
```

```
    Shared portku As String
```

```
    Dim kirim As Integer = 0
```

```
    Dim data_kirim As Single
```

```
    Private Sub Form2_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
```

```
        Form1.Hide()
```

```
        Button9.Enabled = False
```

```
        Dim konek() As String
```

```
        konek = IO.Ports.SerialPort.GetPortNames
```

```
        Array.Sort(konek)
```

```
        ComboBox1.Items.AddRange(konek)
```

```
        ComboBox1.Text = "Pilih Port"
```

```
        Button1.Enabled = False
```

```
        Button2.Enabled = False
```

```
        Button3.Enabled = False
```

```
        Button4.Enabled = False
```

```
        Button5.Enabled = False
```

```
        Button6.Enabled = False
```

```
        Button7.Enabled = False
```

```
        Button8.Enabled = False
```

```
        Button10.Enabled = False
```

```
        Button9sawon.Enabled = False
```

```
        Button10sawoff.Enabled = False
```

```
        Button11reverse.Enabled = False
```

```
        Button12foward.Enabled = False
```

```
        Button13Lampon.Enabled = False
```

```
        Button14lampoff.Enabled = False
```

```
    End Sub
```

```
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
```

```
        myport.Open()
```

```
        myport.WriteLine("0")
```

```
        TextBox2.Text = "Potongan Lebar 5 Cm"
```

```
        myport.Close()
```

```
    End Sub
```

```
    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
```



```
myport.Open()  
myport.WriteLine("1")  
TextBox2.Text = "Potongan Lebar 10 Cm"  
myport.Close()  
End Sub
```

```
Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button3.Click  
myport.Open()  
myport.Write("2")  
TextBox2.Text = "Potongan Lebar 20 Cm"  
myport.Close()  
End Sub
```

```
Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button4.Click  
myport.Open()  
myport.Write("3")  
TextBox2.Text = "Potongan Lebar 25 Cm"  
myport.Close()  
End Sub
```

```
Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button5.Click  
myport.Open()  
myport.Write("4")  
TextBox2.Text = "Potongan Lebar 30 Cm"  
myport.Close()  
End Sub
```

```
Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button6.Click  
myport.Open()  
myport.Write("5")  
TextBox2.Text = "Potongan Lebar 35 Cm"  
myport.Close()  
End Sub
```

```
Private Sub Button7_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button7.Click  
myport.Open()  
myport.Write("6")  
TextBox2.Text = "Reset Posisi"  
myport.Close()  
End Sub
```

```
Private Sub Button8_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button8.Click
    myport.Open()
    myport.Write("H")
    TextBox2.Text = "STOP!!!"
    Button8.Visible = False
    Button9.Enabled = False
    Button1.Enabled = False
    Button2.Enabled = False
    Button3.Enabled = False
    Button4.Enabled = False
    Button5.Enabled = False
    Button6.Enabled = False
    Button7.Enabled = False
    Button8.Enabled = False
    Button9sawon.Enabled = False
    Button10sawoff.Enabled = False
    Button11reverse.Enabled = False
    Button12foward.Enabled = False
    Button13Lampon.Enabled = False
    Button14lampoff.Enabled = False
    MsgBox("KEADAAN BERBAHAYA..!!")
    myport.Close()
End Sub
```

```
Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button10.Click
    myport.Open()
    myport.Write("G")
    Button8.Visible = True
    Button9.Enabled = True
    Button1.Enabled = True
    Button2.Enabled = True
    Button3.Enabled = True
    Button4.Enabled = True
    Button5.Enabled = True
    Button6.Enabled = True
    Button7.Enabled = True
    Button8.Enabled = True
    Button9sawon.Enabled = True
    Button10sawoff.Enabled = True
    Button11reverse.Enabled = True
    Button12foward.Enabled = True
    Button13Lampon.Enabled = True
    Button14lampoff.Enabled = True
    MsgBox("MELANJUTKAN OPERASI MESIN..!!")
```

```
myport.Close()  
End Sub
```

```
Private Sub Button9sawon_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button9sawon.Click  
myport.Open()  
myport.Write("A")  
TextBox2.Text = "SAW ON"  
myport.Close()  
End Sub
```

```
Private Sub Button10sawoff_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button10sawoff.Click  
myport.Open()  
myport.Write("B")  
TextBox2.Text = "SAW OFF"  
myport.Close()  
End Sub
```

```
Private Sub Button11reverse_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button11reverse.Click  
myport.Open()  
myport.Write("C")  
TextBox2.Text = "REVERSE"  
myport.Close()  
End Sub
```

```
Private Sub Button12foward_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button12foward.Click  
myport.Open()  
myport.Write("D")  
TextBox2.Text = "FOWARD"  
myport.Close()  
End Sub
```

```
Private Sub Button13Lampon_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button13Lampon.Click  
myport.Open()  
myport.Write("E")  
TextBox2.Text = "LAMPU ON"  
myport.Close()  
End Sub
```

```
Private Sub Button14lampoff_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button14lampoff.Click  
myport.Open()
```

```
myport.Write("F")
TextBox2.Text = "LAMPU OFF"
myport.Close()
End Sub
```

```
Private Sub Button15_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button15.Click
    If ComboBox1.SelectedItem = Nothing Then
        MsgBox("Pilih Port")
    Else
        Connect()
        Button15.Enabled = False 'salah
        Button9.Enabled = True 'benar
    End If
End Sub
```

```
Private Sub Connect()
    myport = New SerialPort()
    myport.PortName = portku
    myport.BaudRate = 9600
    myport.DataBits = 8
    myport.Parity = Parity.None
    myport.StopBits = StopBits.One
    myport.Handshake = Handshake.None
    ComboBox1.Enabled = False
    Button1.Enabled = True
    Button2.Enabled = True
    Button3.Enabled = True
    Button4.Enabled = True
    Button5.Enabled = True
    Button6.Enabled = True
    Button7.Enabled = True
    Button8.Enabled = True
    Button10.Enabled = True
    Button9sawon.Enabled = True
    Button10sawoff.Enabled = True
    Button11reverse.Enabled = True
    Button12foward.Enabled = True
    Button13Lampon.Enabled = True
    Button14lampoff.Enabled = True
```

```
End Sub
```

```
Private Sub Button16exit_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button16exit.Click
    Application.ExitThread()
```

End Sub

```
Private Sub Button9_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button9.Click
```

```
    myport.Close()  
    ComboBox1.Enabled = True  
    Button15.Enabled = True  
    Button9.Enabled = False  
    Button1.Enabled = False  
    Button2.Enabled = False  
    Button3.Enabled = False  
    Button4.Enabled = False  
    Button5.Enabled = False  
    Button6.Enabled = False  
    Button7.Enabled = True  
    Button8.Enabled = True  
    Button10.Enabled = True  
    Button9sawon.Enabled = False  
    Button10sawoff.Enabled = False  
    Button11reverse.Enabled = False  
    Button12foward.Enabled = False  
    Button13Lampon.Enabled = False  
    Button14lampoff.Enabled = False  
    TextBox2.Text = ""
```

End Sub

```
Private Sub ComboBox1_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
```

```
ComboBox1.SelectedIndexChanged  
    portku = ComboBox1.Text
```

End Sub

End Class

Lampiran 2. Perhitungan

Rumus E% :

$$Error \% = \frac{|HT-HP|}{HT} \times 100\%$$

Dimana : HT = Hasil Teori (Nilai Berdasarkan Teori)

: HP = Hasil Praktek (Nilai Hasil Pengukuran)

1. Potongan 5 cm

a. $Error \% = \frac{|5-4,8|}{5} \times 100\%$

$$E\% = 4,0\%$$

b. $Error \% = \frac{|5-5,3|}{5} \times 100\%$

$$E\% = 6,0\%$$

c. $Error \% = \frac{|5-4,9|}{5} \times 100\%$

$$E\% = 2,0\%$$

d. $Error \% = \frac{|5-5,1|}{5} \times 100\%$

$$E\% = 2,0\%$$

2. Potongan 10 cm

a. $Error \% = \frac{|10-9,8|}{10} \times 100\%$

$$E\% = 2,0\%$$

b. $Error \% = \frac{|10-9,9|}{10} \times 100\%$

$$E\% = 1,0\%$$

c. $Error \% = \frac{|10-10,2|}{10} \times 100\%$

$$E\% = 2,0\%$$

3. Potongan 20 cm

a. $Error \% = \frac{|20-19,7|}{20} \times 100\%$

$E\% = 3,0\%$

b. $Error \% = \frac{|20-19,9|}{20} \times 100\%$

$E\% = 1,0\%$

c. $Error \% = \frac{|20-19,8|}{20} \times 100\%$

$E\% = 1,0\%$

4. Potongan 25 cm

a. $Error \% = \frac{|25-24,8|}{25} \times 100\%$

$E\% = 0,8\%$

b. $Error \% = \frac{|25-24,7|}{25} \times 100\%$

$E\% = 1,2\%$

c. $Error \% = \frac{|25-24,9|}{25} \times 100\%$

$E\% = 0,4\%$

5. Potongan 30 cm

a. $Error \% = \frac{|30-29,5|}{30} \times 100\%$

$E\% = 1,6\%$

b. $Error \% = \frac{|30-29,9|}{30} \times 100\%$

$E\% = 0,8\%$

c. $Error \% = \frac{|30-28,5|}{30} \times 100\%$

$E\% = 5,0\%$

d. $Error \% = \frac{|30-29|}{30} \times 100\%$

$E\% = 3,3\%$

6. Potongan 40 cm

a. $Error \% = \frac{|40-38,5|}{40} \times 100\%$

$E\% = 3,6\%$

b. $Error \% = \frac{|40-38,7|}{40} \times 100\%$

$E\% = 3,2\%$

c. $Error \% = \frac{|40-38,9|}{40} \times 100\%$

$E\% = 2,7\%$

d. $Error \% = \frac{|40-39,3|}{40} \times 100\%$

$E\% = 1,7\%$

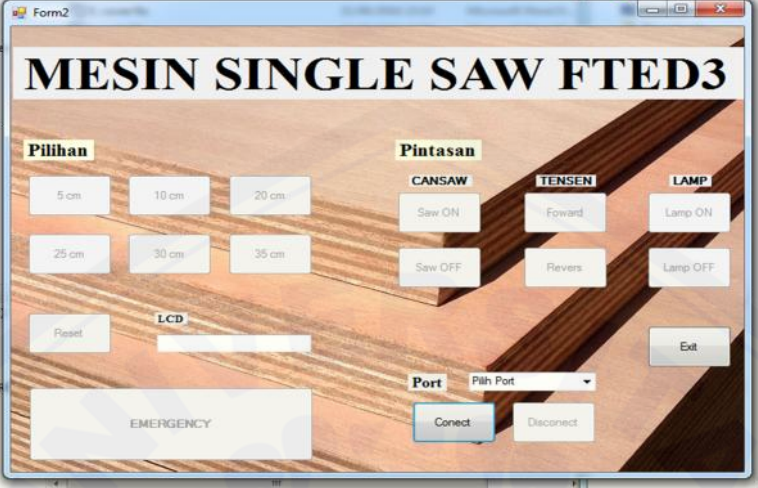

e. $Error \% = \frac{|40-39|}{40} \times 100\%$

$E\% = 2,5\%$


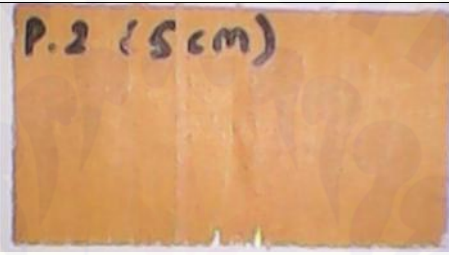
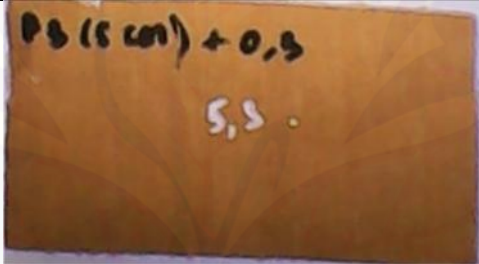
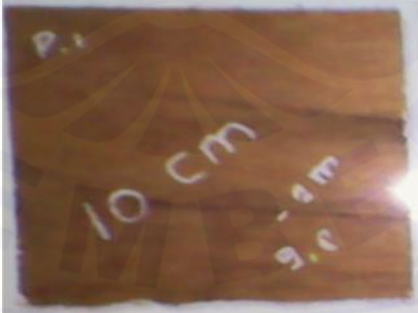
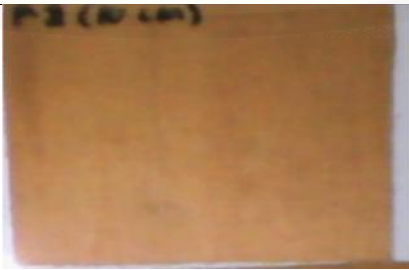
f. $Error \% = \frac{|40-39,1|}{40} \times 100\%$




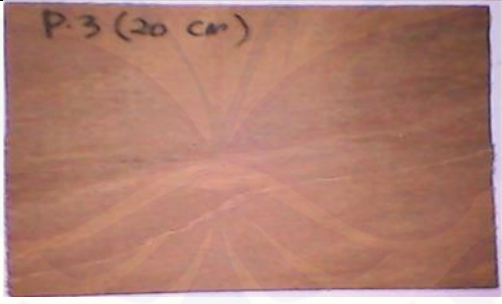

$E\% = 2,2\%$





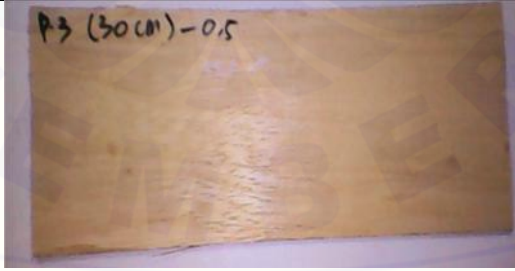

Lampiran 3. Tampilan Visual basic 2010

No.	Tampilan Visual Basic 2010	Kondisi
1.	 <p>The screenshot shows a Windows application window titled 'Form2' with the title bar 'MESIN SINGLE SAW FTED3'. The interface is divided into several sections: 'Pilihan' (Selection) with buttons for 5 cm, 10 cm, 20 cm, 25 cm, 30 cm, and 35 cm; 'Pintasan' (Shortcuts) with sub-sections 'CANSAW' (Saw ON, Saw OFF), 'TENSEN' (Forward, Revers), and 'LAMP' (Lamp ON, Lamp OFF); an 'EMERGENCY' button; an 'LCD' display area; a 'Port' dropdown menu set to 'Pilih Port'; and 'Connect' and 'Disconnect' buttons. The 'Disconnect' button is highlighted in blue, indicating the current state.</p>	<p>saat Disconnect</p>
2.	 <p>The screenshot shows the same application window as above. In this state, the 'Connect' button is highlighted in blue, indicating the current state. The 'Disconnect' button is now disabled and greyed out.</p>	<p>Saat Conect</p>

Lampiran 4. Hasil Contoh Potongan

Potongan	Tampilan Visual Basic 2010	Hasil Potongan (cm)
5 cm		4,8
		5
		5,3
10 cm		9,8
		10

		10
		19,7
20 cm		20
		20
25 cm		25

		25
		25
30 cm		29,9
		29,5
		29,5
40 cm		38,5

		38,7
		38,5

