

**MODUL PEMBELAJARAN APLIKATIF
MIKROKONTROLER ATMEGA8535**

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
Memperoleh gelar Sarjana Teknik



Disusun oleh :

JIMMY HANDOYO WIJAYA
NIM. 0001060334-63

**DEPARTEMEN PENDIDIKAN NASIONAL
UNIVERSITAS BRAWIJAYA
FAKULTAS TEKNIK
MALANG
2007**

PENGANTAR

Puji dan syukur ke hadirat Tuhan Yesus Kristus atas kasih dan karuniaNya sehingga skripsi dengan judul **“Modul Pembelajaran Aplikatif Mikrokontroler ATMEGA8535”** dapat terselesaikan dengan baik.

Penulisan skripsi ini merupakan salah satu bagian yang harus ditempuh mahasiswa Jurusan Elektro Fakultas Teknik Universitas Brawijaya untuk dapat dinyatakan lulus dan memperoleh gelar Sarjana Teknik.

Pada kesempatan ini penulis juga menyampaikan rasa hormat dan terimakasih kepada semua pihak yang turut membantu dalam penyelesaian skripsi ini, secara khusus kepada:

1. Bapa tercinta di Surga yang sudah memberikan Yesus dan Roh Kudus yang selalu memampukan saya.
2. Mama dan Papa yang sudah memberikan segenap dukungannya sehingga saya bisa menyelesaikan kuliah. Titi, Meme, dan Liem adik-adik saya yang setia mendoakan.
3. Diah Kurniasih yang setia mendukung, menemani, menyemangati, memberikan kasih dan mendoakan setiap waktu. *May the best of God always be given to Us.*
4. Bapak M. Rif'an, ST, MT. dan Bapak Ir. Nanang Sulistiyanto selaku dosen pembimbing skripsi
5. Seluruh karyawan dan karyawanati Bagian Administrasi dan Akademik Jurusan Elektro Fakultas Teknik Universitas Brawijaya
6. Seluruh rekan-rekan mahasiswa Teknik Elektro angkatan 2000
7. Teman-teman di PMK Yehezkiel yang turut membantu dan mendoakan saya. Tempat di mana saya boleh melayani.
8. Rekan-rekan kerja di CV. Rekavisitama Inc, terimakasih buat bantuan dan pengertiannya.
9. Tomi dan Lado yang sudah membantu dalam penyelesaian tugas akhir saya. Terimakasih semoga Tuhan Yesus memberkati.

Penulis selalu mengharapkan kritik dan saran yang membangun untuk perbaikan-perbaikan selanjutnya. Semoga tulisan ini dapat bermanfaat bagi semua pihak dan dapat digunakan sebagaimana mestinya.

Malang, Juli 2007

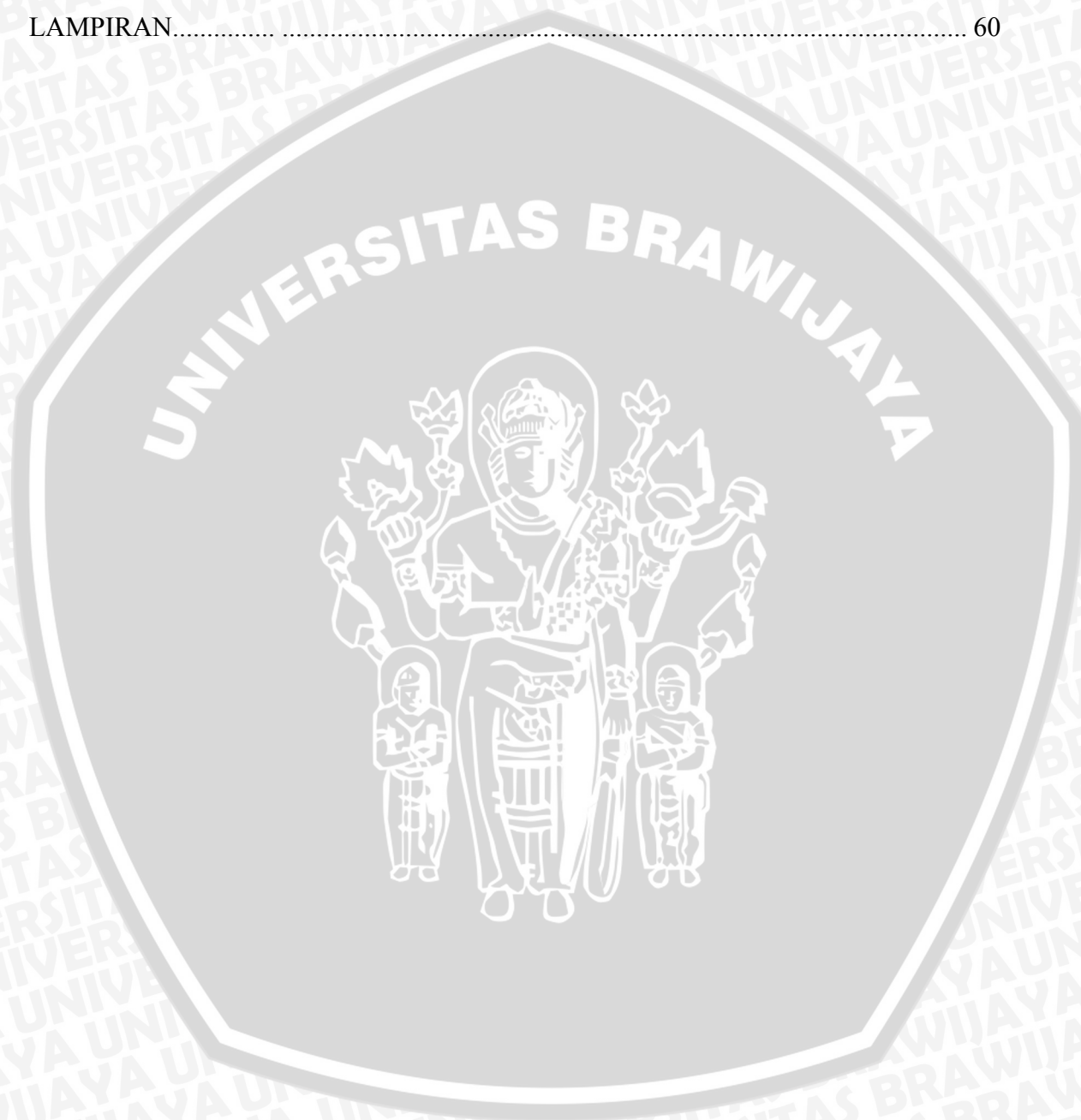
Penulis

DAFTAR ISI

LEMBAR PENGESAHAN	
PENGANTAR	i
DAFTAR ISI	ii
DAFTAR TABEL	v
DAFTAR GAMBAR	vi
DAFTAR LAMPIRAN	vii
ABSTRAK	viii
BAB I PENDAHULUAN	1
1.1 Latar belakang	1
1.2 Rumusan masalah	2
1.3 Ruang lingkup	3
1.4 Tujuan	3
1.5 Sistematika penulisan	3
BAB II KAJIAN TEORI	5
2.1 Mikrokontroler ATMEGA8535	5
2.2 <i>Light Emiting Diode</i>	7
2.3 Motor Steper	8
2.4 <i>Liquid Crystal Display</i>	10
2.5 <i>Keypad</i>	10
2.6 <i>Optoswitch</i>	11
2.7 <i>Schmitt Trigger</i>	12
BAB III METODOLOGI PENELITIAN	15
3.1 Pengumpulan data dan kajian teori	15
3.1.1 Studi literatur	15
3.1.2 Diskusi	15
3.1.3 Survei lapangan	15
3.2 Perencanaan dan pembuatan alat	15
3.2.1 Perencanaan dan realisasi pembuatan perangkat keras	16
3.2.2 Perencanaan dan pembuatan perangkat lunak	16
3.3 Pengujian dan analisis	17
3.3.1 Pengujian perangkat keras	17
3.3.2 Pengujian perangkat lunak	17
3.3.3 Analisis hasil pengujian	18
3.4 Pengambilan kesimpulan dan saran	18
BAB IV PERENCANAAN DAN PEMBUATAN ALAT	18
4.1 Perancangan dan pembuatan perangkat keras	18
4.1.1 Spesifikasi alat	18
4.1.2 Blok diagram	18
4.1.3 Prinsip kerja sistem	20
4.1.4 Perancangan Rangkaian	20
4.1.4.1 Perancangan rangkaian pengisi program	22

4.1.4.2	Perancangan rangkaian reset.....	23
4.1.4.3	Perancangan rangkaian pengendali LED.....	24
4.1.4.4	Perancangan rangkaian pengendali motor stepper.....	25
4.1.4.5	Perancangan rangkaian keypad.....	26
4.1.4.6	Perancangan rangkaian pengendali lampu pijar	27
4.1.4.7	Perancangan rangkaian LCD	27
4.1.4.8	Perancangan rangkaian <i>optoswitch</i>	28
4.1.4.9	Perancangan rangkaian RS232	29
4.2	Perancangan dan pembuatan perangkat lunak	29
4.2.1	Diagram alir	30
4.2.2	Pembuatan perangkat lunak	30
4.2.2.1	Program LED menyala berjalan	30
4.2.2.2	Program menampilkan karakter pada LCD	31
4.2.2.3	Program menggerakkan motor stepper	32
4.2.2.4	Program memasukkan karakter dari keypad dan menampilkan pada LCD	33
4.2.2.5	Program menampilkan nilai tegangan DC pada LCD ..	33
4.2.2.6	Program mengirim/menerima data dari/ke PC	35
4.2.2.7	Program menyalakan LED setiap 5 detik	36
4.2.2.8	Program aplikasi dimmer pada lampu pijar.....	37
4.2.2.9	Program membuat loop tak berhingga sebagai simulasi kondisi <i>hang</i>	37
4.2.2.10	Program membuat LED menyala jika suatu tegangan Melebihi tegangan referensi.....	38
4.2.2.11	Program menyimpan data ke EEPROM dan menampilkan data dari EEPROM pada LCD	39
4.2.2.12	Program simulasi pengaturan suhu ruangan	39
BAB V	PENGUJIAN DAN ANALISIS	43
5.1	Pengujian rangkaian pengendali LED	43
5.2	Pengujian rangkaian motor stepper	44
5.3	Pengujian rangkaian lampu pijar	45
5.4	Pengujian rangkaian LCD.....	45
5.5	Pengujian rangkaian <i>optoswitch</i>	46
5.6	Pengujian program LED menyala berjalan.....	47
5.7	Pengujian program menampilkan karakter pada LCD	47
5.8	Pegujian rangkaian keypad dan program memasukkan karakter dari Keypad serta menampilkannya pada LCD	48
5.9	Pengujian program menggerakkan motor stepper	49
5.10	Pengujian program menampilkan nilai tegangan DC pada LCD	49
5.11	Pengujian program mengirim/menerima data dari/ke PC.....	50
5.12	Pengujian program menyalakan LED setiap 5 detik	51
5.13	Pegujian program aplikasi <i>dimmer</i> pada lampu pijar	52
5.14	Pengujian program membuat loop tak berhingga sebagai simulasi kondisi <i>hang</i>	53
5.15	Pengujian program membuat LED menyala jika suatu tegangan melebihi tegangan referensi.....	54
5.16	Pengujian program menyimpan data ke EEPROM dan menampilkan	

data dari EEPROM pada LCD.....	55
5.17 Pengujian program simulasi pengaturan suhu ruangan	56
BAB VI PENUTUP	58
6.1 Kesimpulan	58
6.2 Saran	58
DAFTAR PUSTAKA	59
LAMPIRAN.....	60



DAFTAR TABEL

Tabel 1.1	Hasil Survey pada 40 Mahasiswa Teknik Elektro	1
Tabel 4.1	Distribusi penggunaan port untuk program simulasi pengendalian suhu ..	21
Tabel 4.2	Jumlah arus yang harus diberikan oleh Port A	21
Tabel 4.3	Jumlah arus yang harus diberikan oleh Port B	21
Tabel 4.4	Jumlah arus yang harus diberikan oleh Port C	22
Tabel 4.5	Jumlah arus yang harus diberikan oleh Port D	22
Tabel 4.6	Contoh program untuk mensimulasikan fitur-fitur ATMEGA8535 dan program aplikatif.....	30
Tabel 4.7	Urutan logika pada pin motor stepper.....	32
Tabel 5.1	Hasil pengujian rangkaian pengendali LED	41
Tabel 5.2	Hasil pengujian motor stepper	45
Tabel 5.3	Hasil pengujian rangkaian kontras LCD.....	46
Tabel 5.4	Hasil pengujian keluaran rangkaian <i>optoswitch</i>	47
Tabel 5.5	Hasil pengujian program memasukkan karakter dari keypad dan menampilkannya pada LCD.....	49
Tabel 5.6	Hasil pengujian program menampilkan nilai tegangan DC pada LCD	50
Tabel 5.7	Hasil pengujian program mengirim/menerima data dari/ke PC	51
Tabel 5.8	Hasil pengujian program menyalakan LED setiap 50 detik	52
Tabel 5.9	Hasil pengujian program membuat loop tak berhingga sebagai simulasi kondisi <i>hang</i>	54
Tabel 5.10	Hasil pengujian program membuat LED menyala jika suatu tegangan melebihi tegangan referensi	55
Tabel 5.11	Hasil pengujian program menyimpan data ke EEPROM dan menampilkan data dari EEPROM pada LCD	56
Tabel 5.12	Hasil pengujian program simulasi pengaturan suhu ruangan	57



DAFTAR GAMBAR

Gambar 2.1	Blok diagram ATMEGA8535	5
Gambar 2.2	Pin Mikrokontroler ATMEGA8535	7
Gambar 2.3	Rangkaian LED.....	8
Gambar 2.4	Bentuk fisik model motor uni polar	9
Gambar 2.5	Bentuk fisik model motor bipolar	9
Gambar 2.6	Blok diagram peraga LCD	10
Gambar 2.7	<i>Keypad</i> matrik 4x4.....	11
Gambar 2.8	<i>Optoswitch</i>	11
Gambar 2.9	Rangkaian <i>optoswitch</i> denganh LED dan <i>phototransistor</i>	12
Gambar 2.10	Rangkaian uji 7404 dan 7414.....	13
Gambar 2.11	Hasil uji 7404.....	13
Gambar 2.12	Hasil uji 7414.....	14
Gambar 2.13	V_{out} terhadap V_{IN} dari <i>Schmitt trigger Inverter</i>	14
Gambar 4.1	Blok diagram sistem	19
Gambar 4.2	Rangkaian pengisi program mikrokontroler ATMEGA8535	23
Gambar 4.3	Rangkaian reset.....	24
Gambar 4.4	Rangkaian pengendali LED	25
Gambar 4.5	Rangkaian pengendali motor stepper.....	26
Gambar 4.6	Rangkaian keypad.....	26
Gambar 4.7	Rangkaian pengendali lampu pijar	27
Gambar 4.8	Rangkaian LCD	27
Gambar 4.9	Rangkaian optoswitch.....	29
Gambar 4.10	Rangkaian RS232	29
Gambar 4.11	Diagram alir program LED menyala berjalan	31
Gambar 4.12	Diagram alir program menampilkan karakter pada LCD	31
Gambar 4.13	Diagram alir program menggerakkan motor stepper	32
Gambar 4.14	Diagram alir scan keypad dan menampilkan karakter keypad pada LCD	34
Gambar 4.15	Diagram alir program menampilkan nilai tegangan DC pada LCD	35
Gambar 4.16	Diagram alir program mengirim/menerima data dari/ke PC	35
Gambar 4.17	Diagram alir program menyalakan LED setiap 5 detik	36
Gambar 4.18	Diagram alir program aplikasi dimmer pada lampu pijar.....	37
Gambar 4.19	Diagram alir Program membuat loop tak berhingga sebagai simulasi kondisi hang.....	38
Gambar 4.20	Diagram alir program membuat LED menyala jika suatu tegangan melebihi tegangan referensi	39
Gambar 4.21	Diagram alir program menyimpan menyimpan data ke EEPROM dan menampilkan data dari EEPROM pada LCD	41
Gambar 4.21	Diagram alir program simulasi pengaturan suhu ruangan	42
Gambar 5.1	Pengujian rangkaian pengendali LED	43
Gambar 5.2	Pengujian rangkaian pengendali motor stepper	45
Gambar 5.3	Pengujian rangkaian pengendali lampu pijar.....	45
Gambar 5.4	Pengujian rangkaian LCD.....	46
Gambar 5.5	Pengujian rangkaian optoswitch	46
Gambar 5.6	Pengujian program menampilkan karakter pada LCD	48
Gambar 5.7	Hasil pengujian program aplikasi dimmer pada lampu pijar.....	53

DAFTAR LAMPIRAN

- Lampiran Layout PCB
- Lampiran Foto Alat
- Lampiran Program
- Lampiran Kuisoner
- Lampiran Buku manual
- Lampiran Data Sheet



ABSTRAK

Jimmy Handoyo Wijaya, Jurusan Elektro, Fakultas Teknik, Universitas Brawijaya, Juli 2007, *Modul Pembelajaran Aplikatif Mikrokontroler ATMEGA8535*, Dosen Pembimbing : M. Rif'an ST, MT. dan Ir. Nanang Sulistiyanto.

Mikrokontroler merupakan topik yang penting dalam bidang elektronika dan untuk mempelajari penggunaannya dibutuhkan metode belajar dengan praktek secara langsung. Oleh karena itu dibutuhkan sebuah perangkat pembantu belajar implementasi mikrokontroler. Pada skripsi ini dipilih mikrokontroler ATMEGA8535 yang cukup populer dan memiliki kemampuan serta fitur yang cukup lengkap.

Modul memiliki sebuah panel kontrol utama (mikrokontroler) yang dapat dihubungkan dengan modul-modul lainnya seperti pemantau I/O (LED), antarmuka dengan pengguna (LCD dan *keypad*), antarmuka dengan komputer (Modul RS232), objek kontrol (motor stepper, *optoswitch*), simulator PWM dan generator sinyal analog. Berdasarkan program yang sudah dibuat, modul LCD harus dihubungkan dengan port C, pin D4 dan D6, modul *keypad* dihubungkan dengan port A dan Modul simulator PWM dihubungkan dengan pin keluaran PWM (Pin OC1A). Modul RS232 dihubungkan dengan antarmuka serial mikrokontroler (Pin RxD dan TxD). Sinyal analog dapat dihubungkan dengan pin A0 (ADC 0) pada percobaan pembacaan level tegangan atau dengan pin B2 dan B3 (Pin AIN0 dan AIN1) untuk percobaan perbandingan dua level tegangan.

Modul ini dapat digunakan untuk mengeksplorasi fitur yang dimiliki oleh mikrokontroler ATMEGA8535 seperti I/O, Timer, Analog signal processing, EEPROM, Programable UART dan Interrupt Eksternal. Modul ini juga dilengkapi dengan 12 percobaan yang bersifat aplikatif.

BAB I PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi elektronika yang semakin maju telah mengarah ke teknologi mikrokontroler, yaitu sebuah komponen elektronik yang dapat bekerja sesuai dengan program yang diisikan ke dalam memorinya seperti layaknya sebuah komputer sederhana. Hal inilah yang membuat belajar menggunakan mikrokontroler menjadi penting bagi mahasiswa Jurusan Elektro maupun penggemar elektronika lainnya, namun sayangnya seringkali kita kesulitan untuk bisa mempelajari cara menggunakan mikrokontroler dikarenakan sulit didapatnya media yang dapat menunjang belajar mikrokontroler yang aplikatif. Sebuah survey yang dilakukan pada 40 mahasiswa yang mengambil kuliah Elektro dan telah menempuh mata kuliah serta praktikum tentang Mikroprosesor, hasil survey tersebut dapat dilihat dalam Tabel 1.1. Adapun contoh kuisioner dilihat dalam Lampiran.

Tabel 1.1. Hasil Survey pada 40 Mahasiswa Teknik Elektro

Kategori	Jumlah (Mahasiswa)	Prosentase (%)
Pernah mengikuti kuliah Mikroprosesor	40	100
Pernah mengikuti praktikum Mikroprosesor	40	100
Memiliki pendapat bahwa materi kuliah mikroprosesor kurang menunjang pengenalan tentang mikroprosesor	19	47,5
Memiliki pendapat bahwa materi praktikum mikroprosesor kurang aplikatif	35	87,5
Tidak dapat menggunakan mikroprosesor untuk hal-hal yang sederhana sekalipun	25	62,5
Setuju pembuatan modul pembelajaran mikrokontroler yang aplikatif	38	95

Sumber: Hasil Survey 2005

Dari hasil survey dapat dilihat bahwa 95% dari mahasiswa tersebut menyetujui dibuatnya perangkat pembelajaran mikrokontroler yang aplikatif. Dengan alasan mata kuliah dan praktikum yang sudah ditempuh kurang bersifat aplikatif sehingga 62,5% dari mahasiswa tersebut belum bisa menggunakan mikrokontroler untuk hal yang sederhana sekalipun.

Berdasarkan hal tersebut di atas maka perlu dibuat suatu perangkat yang dapat digunakan untuk belajar mengenai cara menggunakan mikrokontroler secara aplikatif

untuk membantu mahasiswa Jurusan Elektro maupun penggemar elektronika lainnya lebih mengerti kegunaan dan cara menggunakan mikrokontroler.

Dewasa ini telah banyak beredar berbagai jenis dan merek mikrokontroler di pasaran dan mikrokontroler jenis AVR dari ATMEL Corporation merupakan salah satu jenis mikrokontroler yang cukup handal dengan berbagai fitur yang dimilikinya. ATMEGA8535 adalah salah satu jenis mikrokontroler AVR, mikrokontroler ini memiliki berbagai fitur yang sangat berguna untuk keperluan-keperluan studi maupun industri sebagai contoh fitur ADC Internal, Analog Comparator Internal, EEPROM dan berbagai fitur lainnya. Oleh karena itu dengan mempelajari fungsi dan fitur yang dimiliki oleh mikrokontroler ATMEGA8535 diharapkan dapat membantu mahasiswa Jurusan Elektro maupun penggemar elektronika lainnya lebih memahami tentang mikrokontroler dan berkarya dalam bidang teknologi modern demi kemajuan bangsa dan negara.

Konsep pembelajaran yang digunakan pada pembuatan modul ini mengacu pada sistem yang diterapkan dalam buku *Teknik Kontrol Otomatik* karya Prof. Katsuhiko Ogata yang mana dalam buku tersebut pembaca diberikan sistem-sistem yang sudah jadi untuk dipelajari detailnya. Sedangkan pemilihan fitur-fitur mikrokontroler ATMEGA8535 yang akan dibahas mengacu pada 2 buku yaitu *Panduan Pembuatan Program dan Rangkaian Mikrokontroler MC68705U3* karya Ir. Busono dan *Belajar Mikroprosesor-Mikrokontroler Melalui Komputer PC* karya Arianto Widyatmo. Dari kedua buku tersebut didapatkan bahwa untuk mempelajari implementasi mikrokontroler ada beberapa fitur yang harus dikuasai. Fitur-fitur tersebut antara lain: Programable I/O, Timer/Counter, Interupt dan UART.

1.2 Perumusan Masalah

Berdasarkan latar belakang di atas, maka dirumuskan masalah yang ada antara lain:

1. Bagaimana merancang perangkat keras untuk mengisikan data program ke Flash mikrokontroler ATMEGA8535?
2. Bagaimana merancang perangkat keras yang dapat digunakan sebagai modul pembelajaran aplikatif mikrokontroler ATMEGA8535?
3. Bagaimana merancang perangkat keras yang dapat mensimulasikan implementasi fitur yang dimiliki mikrokontroler ATMEGA8535?

4. Bagaimana merancang perangkat lunak sebagai materi belajar bagi pengguna modul dan mensimulasikan implementasi fitur yang dimiliki mikrokontroler ATMEGA8535?

1.3 Ruang Lingkup

Dengan mengacu pada permasalahan yang telah dirumuskan, maka hal-hal yang berkaitan dengan alat yang akan dibuat, diberi batasan sebagai berikut:

- a. Membahas mengenai rangkaian untuk mengisikan data ke Flash mikrokontroler ATMEGA8535.
- b. Membahas mengenai perancangan dan pembuatan rangkaian yang dapat digunakan sebagai modul pembelajaran aplikatif mikrokontroler ATMEGA8535.
- c. Membahas mengenai perancangan dan pembuatan rangkaian yang dapat mensimulasikan penggunaan fitur yang dimiliki mikrokontroler ATMEGA8535.
- d. Membahas mengenai perancangan dan pembuatan perangkat lunak sebagai materi belajar bagi pengguna modul dan mensimulasikan implementasi fitur yang dimiliki mikrokontroler ATMEGA8535.
- e. Tidak membahas mengenai teknik dan pembuatan perangkat lunak yang akan dipakai untuk mengisikan data ke Flash mikrokontroler ATMEGA8535.
- f. Tidak membahas mengenai catu daya.

1.4 Tujuan

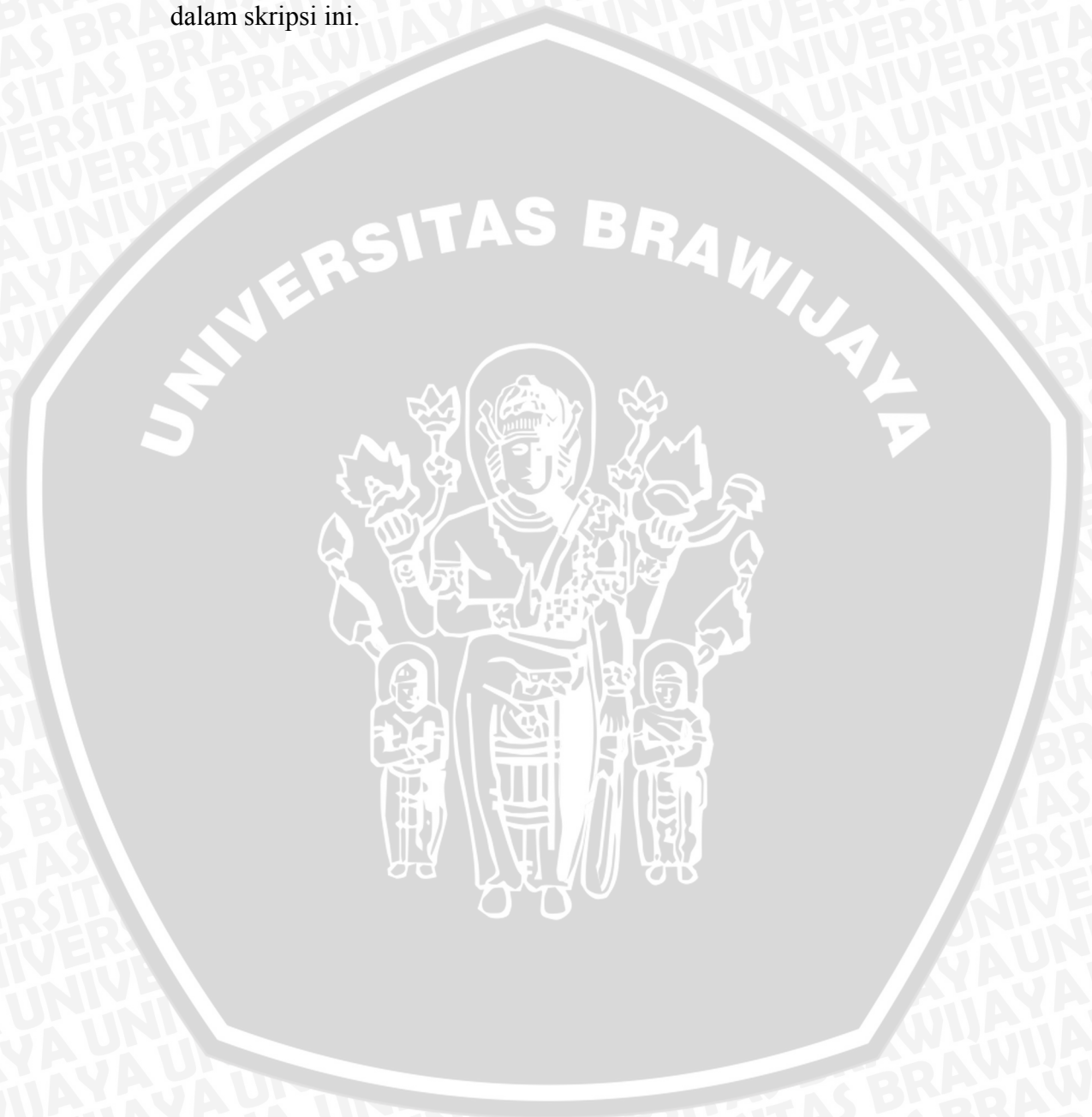
Tujuan penelitian ini adalah merancang dan membuat suatu perangkat yang dapat digunakan untuk belajar menggunakan mikrokontroler secara aplikatif khususnya mikrokontroler ATMEGA8535 sehingga diharapkan setelah mempelajari sifat, karakteristik dan fitur mikrokontroler ATMEGA8535 menggunakan perangkat ini, pengguna akan mampu menggunakan mikrokontroler untuk kebutuhan dan karya-karya mereka masing-masing.

1.5 Sistematika Penulisan

Sistematika penulisan dan gambaran tentang hal-hal yang dibahas dalam setiap bab adalah sebagai berikut:

- | | |
|--------|---|
| BAB I | Pendahuluan memuat latar belakang, tujuan, rumusan masalah, batasan masalah dan sistematika penulisan. |
| BAB II | Dasar teori memuat teori tentang mikrokontroler ATMEGA8535 dan komponen penunjang yang dipergunakan dalam perancangan alat. |

- BAB III Metodologi yang digunakan dalam perancangan dan pembuatan alat ini.
- BAB IV Perencanaan dan pembuatan alat.
- BAB V Pengujian yang dilakukan terhadap alat dan perangkat lunak yang telah dibuat serta menampilkan hasil-hasil pengujian.
- BAB VI Penutup yang memuat kesimpulan dan saran dari tema yang dianalisa dalam skripsi ini.



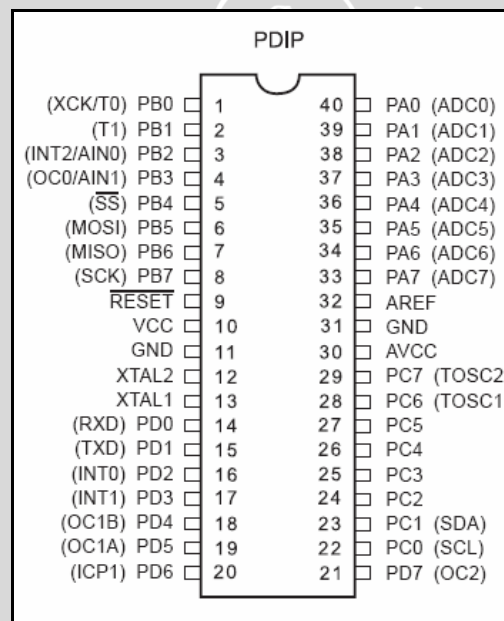
Mikrokontroler ATMEGA8535 dilengkapi dengan berbagai fitur yang penting dan berguna untuk keperluan belajar dan industri. Fitur-fitur tersebut antara lain sebagai berikut:

- 32 x 8 Register yang biasa digunakan secara umum
- Penyelesaian instruksi hingga 16 MIPS pada oscilator 16 MHz
- 8 KByte Programmable Flash
- 512 Byte EEPROM
- 512 Byte Internal SRAM
- Program pengunci untuk keamanan perangkat lunak
- 8-channel, 10 bit ADC
- Programmable UART
- Master/slave SPI Serial Interface
- Dua 8-bit Timer/Counter dengan separate prescaler dan Compare Mode
- Satu 16-bit Timer/Counter dengan separate prescaler, Compare Mode dan Capture Mode dan Dual 8-,9-, atau 10-bit PWM
- Programable Watchdog Timer with On-chip Oscillator
- Analog Comparator
- Power-on Reset Circuit
- Oscillator dan Clock Internal
- Real-time Counter Clock dengan Oscilator terpisah
- External dan Internal Interrupt Source
- Tiga mode tidur, Idle, Power Save dan Power down
- 4 buah port I/O terprogram, yang masing-masing mempunyai 8 jalur I/O

Mikrokontroler ATMEGA8535 memiliki pin sebanyak 40 buah seperti terlihat dalam Gambar 2.3. Setiap pin dapat kita lihat konfigurasi sebagai berikut:

- a. Pin 1 sampai 8 (PortB) merupakan port paralel 8 bit dua arah (*bi-directional*) yang dapat digunakan untuk berbagai keperluan
- b. Pin 9 (Reset) adalah masukan (aktif rendah)
- c. Pin 10 (VCC) merupakan catu tegangan digital
- d. Pin 11 (GND) merupakan *ground* digital
- e. Pin 12 (XTAL 2) adalah pin keluaran *inverting oscillator amplifier*
- f. Pin 13 (XTAL 1) adalah pin masukan *inverting oscillator amplifier* dan masukan untuk internal *clock*

- g. Pin 14 sampai 20 (PortD) adalah port paralel 8 bit dua arah yang memiliki fungsi pengganti yang meliputi *RxD*, *TxD*, *INT0*, *INT1*, *OC1B*, *OC1A*, *ICP*. Bila fungsi pemakai tidak dipakai, pin ini dapat digunakan sebagai port paralel 8 bit serbaguna
- h. Pin 21, sampai 29 (PortC) adalah port paralel 8 bit dua arah (*bi-directional*)
- i. Pin 30 (*AVCC*) adalah pin untuk catu tegangan port A dan *A/D Converter*. Jika ADC tidak digunakan, pin ini harus dihubungkan ke VCC. Jika ADC digunakan, pin ini harus dihubungkan ke VCC melalui *low-pass filter*
- j. Pin 31 *AGND* (*Analog Ground*), jika rangkaian memiliki *ground* analog yang terpisah maka pin ini harus dihubungkan pada *ground* rangkaian tersebut
- k. Pin 32 (*AREF*) merupakan masukan referensi analog untuk ADC. Untuk operasi ADC tegangan harus antara 2 V sampai *AVCC*
- l. Pin 33 sampai 40 (PortA) merupakan port paralel 8 bit dua arah (*bi-directional*). Port A merupakan input analog untuk *A/D Converter*



Gambar 2.2 Pin mikrokontroler ATMEGA8535
Sumber : ATMEL, 2001: 2

4.2 Light Emitting Diode

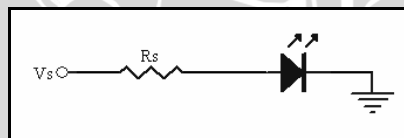
Pada dioda berprategangan maju, elektron bebas melintasi persambungan dan jatuh ke dalam lubang (*hole*). Pada saat elektron ini jatuh dari tingkat energi yang lebih tinggi ke tingkat energi yang lebih rendah, ia memancarkan energi. Pada dioda-dioda biasa, energi ini keluar dalam bentuk panas tetapi pada dioda pemancar cahaya (LED), energi memancar sebagai cahaya. LED telah digunakan secara luas dalam dunia elektronika karena beberapa keunggulannya. Keunggulan itu antara lain:

- Tegangan kerja yang rendah (1 sampai 2 volt)
- Umurnya yang panjang (lebih dari 20 tahun)
- Saklar nyala matinya cepat (dalam orde nanodetik)

Dioda jenis LED ini biasanya menggunakan unsur-unsur seperti galium, arsen dan fosfor, dengan bahan ini pabrik dapat memproduksi LED yang dapat memancarkan cahaya merah, hijau, kuning, biru, jingga dan bahkan infra merah (cahaya tak tampak).

LED mempunyai penurunan tegangan lazimnya dari 1,5 V sampai 2,5 V untuk arus di antara 10 mA sampai 150 mA (Malvino, 1985:97). Penurunan tegangan yang tepat tergantung dari arus LED, warna dan sebagainya. Jika tidak hal lain yang ditentukan maka digunakan tegangan jatuh nominal 2 V pada saat memperbaiki atau menganalisa rangkaian-rangkaian LED. Dalam perancangan disesuaikan dengan *data sheet* LED yang digunakan karena pada umumnya tegangan LED mempunyai kelonggaran yang besar. Kecemerlangan LED bergantung pada arusnya. Idealnya cara terbaik untuk mengendalikan kecemerlangan LED ialah dengan menjalankan LED dengan sumber arus. Cara berikutnya yang terbaik setelah sumber arus adalah dengan catu yang besar dan resistansi seri yang besar. Gambar rangkaian yang digunakan terlihat dalam Gambar 2.3. Dalam hal ini arus diberikan oleh:

$$I = \frac{V_S - V_{LED}}{R_S}$$



Gambar 2.3 Rangkaian LED
Sumber: Malvino, 1991 : 97

Semakin besar tegangan sumber maka semakin kecil pengaruh V_{LED} , dengan kata lain V_S yang besar menghilangkan pengaruh perubahan pada tegangan LED.

4.3 Motor Steper

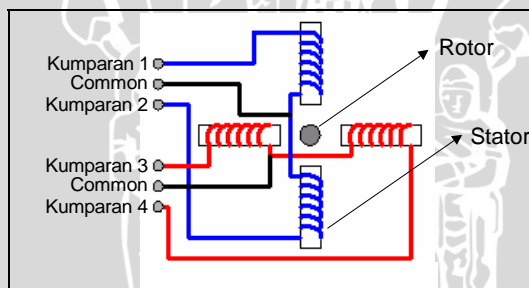
Motor stepper adalah motor yang gerakannya berdasarkan pulsa-pulsa listrik yang diberikan. Setiap kali pengiriman pulsa, maka motor akan bergerak selangkah, yaitu satu putaran sudut kecil. Ukuran langkah tersebut bergantung pada perancangan motor dan dapat sekecil $1,5^\circ$ atau paling besar 30° . Motor akan berputar lebih cepat atau lebih lambat bergantung pada pada frekuensi pengiriman pulsa. Pulsa-pulsa tersebut mampu dikirimkan hingga kecepatan maksimum kurang lebih 2000 pulsa per detik.

Keuntungan menggunakan motor stepper:

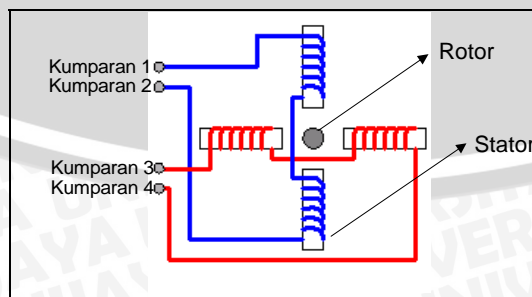
1. Sudut putaran sesuai dengan pulsa masukan
2. Motor memiliki torsi yang penuh dan tidak mudah bergerak (selama lilitan dialiri listrik)
3. Penempatannya presisi dan gerakannya dapat diulangi kembali karena motor stepper yang bagus memiliki akurasi 3 – 5 % untuk masing-masing langkah dan tidak berlaku kumulatif dari langkah yang satu ke langkah berikutnya
4. Respon yang bagus saat mulai bergerak, berhenti dan berputar ke arah sebaliknya
5. Sangat tahan lama karena tidak ada kontak sikat pada motor. Oleh karena itu masa pakai dari motor hanya bergantung pada daya tahan rotornya.
6. Pengaturan yang mudah dan sederhana.
7. Kecepatan putaran dapat diatur dalam range yang lebar sebagaimana kecepatannya bergantung dari frekuensi pulsa yang masuk.

Kerugian menggunakan motor stepper:

1. Resonansi dapat terjadi jika motor tidak dikontrol secara tepat
2. Tidak mudah dioperasikan dalam kecepatan yang sangat tinggi



Gambar 2.4. Bentuk fisik model motor unipolar
Sumber : Wirz Electronics, 1998



Gambar 2.5. Bentuk fisik model motor bipolar
Sumber : Wirz Electronics, 1998

4.4 *Liquid Crystal Display*

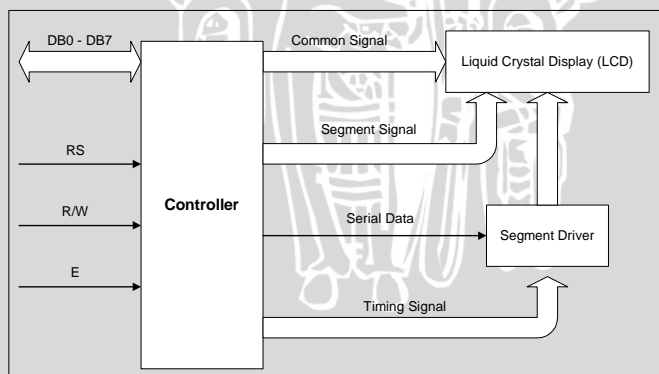
Liquid Crystal Display (LCD) adalah sebuah media penampil yang terdiri atas tumpukan tipis dari dua lembar kaca yang tepinya tertutup rapat. Di antara kedua lembar kaca diberi bahan kristal cair (*Liquid Crystal*) yang tembus cahaya. Permukaan luar dari masing-masing keping kaca mempunyai lapisan penghantar tembus cahaya seperti oksida timah (*tin oxide*) atau oksida indium (*indium oxide*).

Modul peraga ini memiliki kelebihan dibanding dengan peraga lain:

- Register-register terdapat dalam modul.
- Tingkat kesederhanaan dalam rangkaian dan kemudahan dalam pengoperasian.
- Kesederhanaan dalam perangkat lunak.

Modul ini membutuhkan daya yang kecil dan dilengkapi dengan panel LCD dan tingkat kontras yang cukup tinggi serta pengendali LCD CMOS yang terpasang dalam modul tersebut. Pengendali mempunyai pembangkit karakter ROM/RAM dan *display data RAM*. Semua fungsi *display* diatur oleh instruksi-instruksi sehingga modul LCD ini dengan mudah dapat dihubungkan dengan mikrokontroler.

Masukan yang diperlukan untuk mengendalikan modul berupa *bus* data yang masih *ter-multiplex* dengan *bus* alamat serta 3 bit sinyal kontrol. Sementara pengendalian *dot matrix* LCD dilakukan secara internal oleh kontroler yang sudah terpasang pada modul LCD. Gambar 2.6 menunjukkan diagram blok dari LCD.

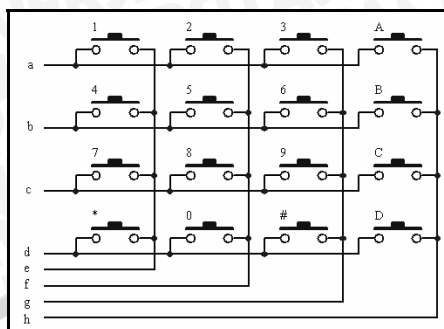


Gambar 2.6. Diagram Blok peraga LCD
Sumber: Seiko Instrument, 1987:3

4.5 Keypad

Tombol tekan atau *keypad* merupakan suatu tombol yang digunakan untuk memasukkan data ke suatu rangkaian sebagai kode-kode. *Keypad* yang digunakan untuk modul pembelajaran mikrokontroler ATMEGA8535 adalah *keypad* matrik 4x4, yang terdiri atas baris dan kolom dan masing-masing berjumlah empat. Bila salah satu tombol

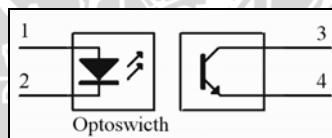
ditekan maka akan terjadi kombinasi keluaran antara baris dan kolom. Gambar rangkaian keypad dapat dilihat dalam Gambar 2.7.



Gambar 2.7. Keypad Matrik 4x4

4.6 Optoswitch

Optoswitch menggabungkan LED (Light Emitting Diode) sebagai pembangkit cahaya dan *Phototransistor* sebagai sensor cahaya dalam satu kemasan. Gambar 2.8 menunjukkan salah satu jenis *Optoswitch*. *Optoswitch* mempunyai LED pada sisi masukan dan *Phototransistor* pada sisi keluaran.



Gambar 2.8. Optoswitch

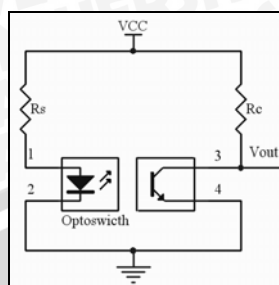
Keterangan : Pin 1 = Anoda Pin 2 = Katoda
Pin 3 = Kolektor Pin 4 = Emitor

LED berfungsi sebagai pembangkit cahaya sedangkan *Phototransistor* sebagai sensor cahaya. Kecemerlangan LED tergantung pada arusnya. Untuk mengatur arus yang mengalir pada LED digunakan sebuah resistor yang terhubung seri dengan LED dan sumber tegangan. LED mempunyai penurunan tegangan lazimnya 1,5 V sampai 2,5 V (Malvino, 1985:97). Untuk menentukan penurunan tegangan yang tepat pada LED, perlu dilakukan pengujian terlebih dahulu.

Ketika tidak ada penghalang antara LED dan *Phototransistor* maka *Phototransistor* akan terkena cahaya dari LED. Pada *Phototransistor*, dioda kolektor sensitif baik terhadap cahaya maupun terhadap panas (Malvino, 1985:135). Hal ini akan mengakibatkan timbul arus kolektor pada *Phototransistor*. Untuk menentukan besarnya arus kolektor yang mengalir pada *Phototransistor*, kita dapat melihat pada grafik "Output Current vs Input Current" pada *data sheet Optoswitch* yang digunakan. Setelah kita mengetahui besarnya arus kolektor yang mengalir pada saat *Phototransistor* terkena

cahaya LED, kita dapat menentukan nilai resistansi kolektor agar *Phototransistor* tersebut dapat berfungsi sebagai *switch*.

Rangkaian *Optoswitch* yang digunakan dapat dilihat dalam Gambar 2.9



Gambar 2.9. Rangkaian *Optoswitch* dengan LED dan *Phototransistor*
Sumber: Malvino, 1987:136

Untuk menentukan nilai R_s , kita gunakan perhitungan sebagai berikut:

$$R_s = \frac{V_{CC} - V_{LED}}{I_{LED}}$$

Untuk menentukan nilai R_c , kita gunakan perhitungan sebagai berikut;

$$R_c = \frac{V_{CC} - V_{CE}}{I_C}$$

Ketika *Phototransistor* terkena cahaya LED, maka *Phototransistor* mengalami keadaan saturasi ($V_{CE} = 0$ V) sehingga nilai dari R_c menjadi:

$$R_c = \frac{V_{CC}}{I_C}$$

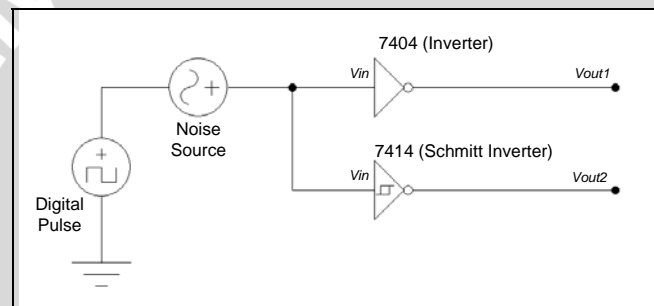
dan $V_{out} = 0$ V. Saat *Phototransistor* tidak terkena cahaya LED (ada penghalang) maka *Phototransistor* akan mengalami keadaan *cut off* sehingga $V_{out} = V_{CC}$.

4.7 Schmitt Trigger

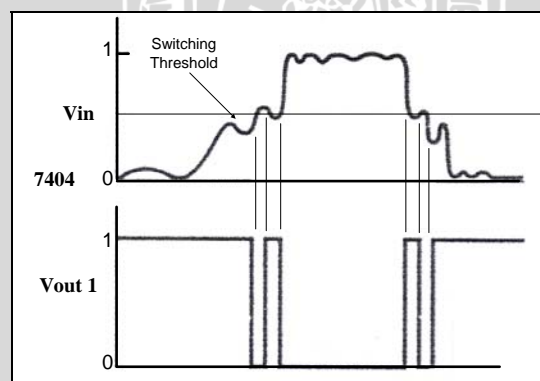
Schmitt Trigger merupakan rangkaian elektronik yang digunakan untuk mengubah bentuk gelombang masukan dengan waktu transisi lambat menjadi gelombang keluaran dengan waktu transisi yang cepat (Kleitz, 1996:367). *Schmitt Trigger* menggunakan sebuah teknik yang dinamakan *positive feedback* secara internal untuk mempercepat waktu transisi (*rise time* dan *fall time*). Selain itu *Schmitt Trigger* juga memperlihatkan *hysteresis effect* karena rangkaian itu berubah dari keadaan pertama ke keadaan kedua pada suatu isyarat masukan dan kemudian kembali dari keadaan kedua ke keadaan pertama pada isyarat masukan yang berbeda. *Schmitt Trigger* sangat berguna untuk diterapkan pada isyarat masukan yang sering dipengaruhi oleh *noise* (gangguan). *Schmitt Trigger* ini dikemas dalam bentuk rangkaian terintegrasi sehingga mempunyai

kelebihan antara lain: mempunyai dimensi yang kecil, sumber daya kecil dan mudah dalam pemakaian.

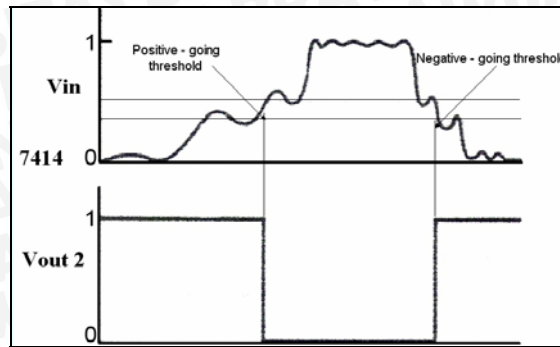
Positive-going threshold adalah nilai isyarat masukan ketika isyarat masukan mulai berubah dari posisi logika rendah ke posisi logika tinggi. Sedangkan *negative-going threshold* adalah nilai isyarat masukan ketika isyarat masukan mulai berubah dari posisi logika tinggi ke posisi logika rendah. Pada *standar inverter (7404)*, *Positive-going threshold* dan *negative-going threshold* memiliki nilai yang sama. Sedangkan pada *Schmitt trigger inverter (7414)*, *Positive-going threshold* memiliki nilai yang lebih tinggi daripada *negative-going threshold*. Perbedaan antara *Positive-going threshold* dan *negative-going threshold* didefinisikan sebagai *hysteresis voltage* (Kleitz, 1996:367). Gambar 2.10 hingga 2.12 mengilustrasikan perbedaan sinyal keluaran antara *standar inverter (7404)* dan *Schmitt trigger inverter (7414)*.



Gambar 2.10. Rangkaian Uji 7404 dan 7414
Sumber: Kleitz, 1996:368

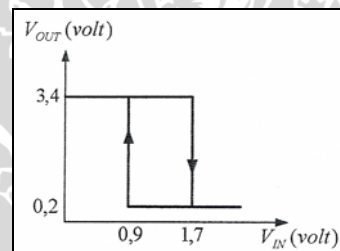


Gambar 2.11. Hasil Uji 7404
Sumber: Kleitz, 1996:368



Gambar 2.12. Hasil Uji 7414
Sumber: Kleitz, 1996:368

Gambar 2.11 menunjukkan bahwa keluaran dari *standar inverter* (7404) mudah sekali dipengaruhi oleh *noise*. Keluaran *standar inverter* (7404) memiliki 3 kali tepi negatif padahal isyarat masukan hanya memiliki 1 kali tepi negatif. Namun hal ini tidak terjadi pada *schmitt trigger inverter* (7414). Gambar 2.12 menunjukkan bahwa keluaran dari *schmitt trigger inverter* (7414) hanya memiliki 1 kali tepi negatif. Pada Gambar 2.13 diperlihatkan grafik hubungan antara V_{OUT} dan V_{IN} dari *schmitt trigger inverter* (7414).



Gambar 2.13. V_{OUT} terhadap V_{IN} dari *Schmitt Trigger Inverter*
Sumber: Kleitz, 1996:368

Pada umumnya *schmitt trigger inverter* (7414) mempunyai *Positive-going threshold* sebesar 1,7 V dan *negative-going threshold* sebesar 0,9 V serta *hyteresis voltage* sebesar 0,8 V (Kleitz, 1996:367). Sedangkan untuk sinyal keluarannya memiliki tegangan 3,4 V untuk level logika tinggi dan 0,2 V untuk level logika rendah (Kleitz, 1996:367).

BAB III

METODOLOGI PENELITIAN

Metode penulisan skripsi ini adalah studi literatur, perancangan dan pembuatan alat, pengujian dan analisis alat serta penarikan kesimpulan dan saran.

3.1 Pengumpulan Data dan Kajian Teori

Pengumpulan data dan kajian teori pada pembuatan skripsi ini melalui studi literatur, diskusi dan survei lapangan.

3.1.1 Studi Literatur

Studi literatur dilakukan untuk mempelajari tentang mikrokontroler ATMEGA8535, motor stepper, LCD, LED, *Optoswitch* dan komponen-komponen lain yang menunjang penelitian ini. Pengumpulan data dan kajian teori dilakukan melalui internet, buku referensi, *data sheet*, *application note*, karya tulis dan lain-lain.

3.1.2 Diskusi

Dalam mengkaji teori juga digunakan metode diskusi, yaitu mencari informasi yang diperoleh dari konsultasi dengan pembimbing, staf pengajar dan rekan-rekan mahasiswa.

3.1.3 Survei Lapangan

Survei lapangan dilakukan untuk mengumpulkan data yang berupa identifikasi perancangan alat. Data yang diperoleh berupa jenis dan spesifikasi komponen elektronik, harga, dan keberadaannya di pasar.

Survei di lapangan diperlukan guna mencocokkan komponen yang digunakan berdasarkan data literatur dan *data sheet* dengan komponen yang ada di pasar. Keberadaan komponen dan harga yang beredar di pasar akan menentukan realisasi perancangan dan pembuatan alat.

3.2 Perancangan dan Pembuatan Alat

Perancangan dan pembuatan alat ini meliputi perangkat keras alat (*hardware*) beserta sistem pendukungnya dan perangkat lunak (*software*) yang digunakan untuk mendukung pembuatan alat ini.

3.2.1 Perancangan dan Realisasi Pembuatan Perangkat Keras

Penentuan spesifikasi adalah bagian pertama yang harus dirumuskan terlebih dahulu. Penentuan spesifikasi alat berdasarkan pada latar belakang, perumusan masalah dan tujuan yang mendasari pembuatan modul pembelajaran aplikatif mikrokontroler ATMEGA8535.

Tahap selanjutnya adalah perencanaan blok diagram. Perencanaan blok diagram didasarkan pada fungsi-fungsi yang diperlukan pada perancangan dan pembuatan alat. Setiap blok akan mewakili fungsi yang dibutuhkan. Penempatan dan hubungan antar blok akan menentukan kinerja dari alat, sehingga diperlukan penempatan yang sesuai agar alat dapat bekerja sesuai dengan yang direncanakan.

Tahap terakhir dari perancangan alat adalah perencanaan rangkaian. Perencanaan rangkaian dikerjakan dengan menentukan komponen-komponen yang diperlukan pada setiap blok rangkaian. Penentuan komponen pada perencanaan rangkaian berdasarkan literatur yang telah didapatkan.

Tahap awal dari pembuatan alat adalah pembuatan skema rangkaian. Pembuatan skema rangkaian menggunakan bantuan program *Protel Design*. Pembuatan skema rangkaian diperlukan untuk mempermudah pembuatan layout *Printed Circuit Board* (PCB) dan pemasangan komponen.

Tahapan selanjutnya adalah pembuatan layout PCB dengan bantuan program *Protel Design*. Pembuatan layout harus memperhatikan penempatan jalur-jalur dan kaki komponen-komponen yang dipergunakan. Penempatan jalur yang baik dapat meminimalkan luas area PCB dan meningkatkan kualitas dari desain tersebut. Tahap pembuatan PCB dimulai dengan penyablonan PCB polos yang dilanjutkan dengan pengecatan serta pengeboran PCB dan diakhiri dengan masking PCB.

Tahap terakhir dari pembuatan alat adalah memasang komponen pada PCB sesuai dengan skema rangkaian yang telah dibuat dan dilanjutkan dengan penyolderan komponen pada PCB.

3.2.2 Perencanaan dan Pembuatan Perangkat Lunak

Perencanaan dan pembuatan perangkat lunak berdasarkan pada fitur-fitur yang dimiliki oleh mikrokontroler ATMEGA8535 serta penggunaan mikrokontroler secara umum dan sederhana oleh mahasiswa Jurusan Elektro maupun penggemar elektronika lainnya. Perencanaan perangkat lunak dimulai dengan pembuatan diagram alir. Pembuatan diagram alir disesuaikan dengan algoritma dan aturan yang telah ditetapkan.

Diagram alir dibuat secara umum untuk semua program yang akan dibuat. Pembuatan diagram alir yang baik akan menghasilkan program yang tidak terdapat kesalahan dan mudah dimengerti oleh pembaca program. Bahasa pemrograman yang digunakan dalam modul pembelajaran ini adalah bahasa pemrograman C++ dengan bantuan *software* Code Vision AVR versi 1.24.0 standar sebagai *editor* dan *compiler* sedangkan untuk pengisian program ke mikrokontroler menggunakan *free software* PoniProg2000.

3.3 Pengujian dan Analisis

Sebelum melakukan pengujian, terlebih dahulu ditetapkan prosedur pengujian. Metode atau prosedur yang berbeda dapat menyebabkan hasil pengujian yang berbeda, walaupun rangkaian yang diuji sama. Dengan demikian pemilihan prosedur yang baik akan mempengaruhi hasil pengujian. Referensi atau literatur yang berhubungan dengan komponen atau alat uji akan membantu dalam penetapan prosedur pengujian.

3.3.1 Pengujian Perangkat Keras

Prosedur pengujian perangkat keras meliputi pemilihan alat uji atau ukur dan cara pengukuran serta urutan pengujian. Alat uji yang digunakan adalah multi-meter digital yang digunakan untuk pengukuran tegangan, arus dan hambatan. Cara pengukuran tegangan dan hambatan adalah paralel dengan komponen yang diukur. Pada pengukuran tegangan, tegangan catu diaktifkan, sedangkan pada pengukuran tahanan tegangan catu dimatikan. Pengukuran arus dengan pemasangan secara seri dengan komponen yang akan diukur. Urutan pengujian pertama kali adalah pengidentifikasian jalur pada PCB dan membandingkan dengan skema yang telah dibuat. Langkah selanjutnya adalah menguji setiap komponen yang akan dipergunakan, dilanjutkan dengan pengujian tiap blok rangkaian. Pengujian perangkat keras secara keseluruhan sistem dilakukan bersamaan dengan pengujian perangkat lunak.

Langkah selanjutnya adalah pengambilan hasil pengujian yang berupa tegangan, arus, hambatan, dan hal lain yang berhubungan dengan alat yang dibuat. Hasil pengujian ditampilkan dalam bentuk tabel, gambar, angka, dan foto.

3.3.2 Pengujian Perangkat Lunak

Prosedur pengujian perangkat lunak dilakukan pada setiap program yang sudah dibuat dengan cara membandingkan hasil kerja perangkat lunak yang dibuat dalam skripsi ini dengan perangkat lunak yang mendukung. Langkah selanjutnya adalah pengujian masing-masing program yang telah digabungkan dengan perangkat keras masing-masing.

3.3.3 Analisis Hasil Pengujian

Analisis meliputi hasil pengukuran tegangan, arus, hambatan dan perangkat lunak pada saat dilakukan pengujian. Apabila analisis hasil pengujian tidak sesuai dengan perumusan masalah dan tujuan, maka dilakukan perancangan ulang dengan melihat kembali atau mencari literatur yang dibutuhkan.

3.4 Pengambilan Kesimpulan dan Saran

Pengambilan kesimpulan dilakukan dengan membandingkan hasil pengujian dan analisis alat dengan perumusan masalah. Kesimpulan berisi tentang hasil analisis mengenai kesesuaian alat dengan rumusan masalah dan tujuan yang ingin dicapai.

Setelah pengambilan kesimpulan langkah selanjutnya adalah pemberian saran berdasarkan kesimpulan yang telah didapat. Saran dapat berupa pengembangan alat atau perubahan sistem atau cara kerja dari alat yang telah dibuat.



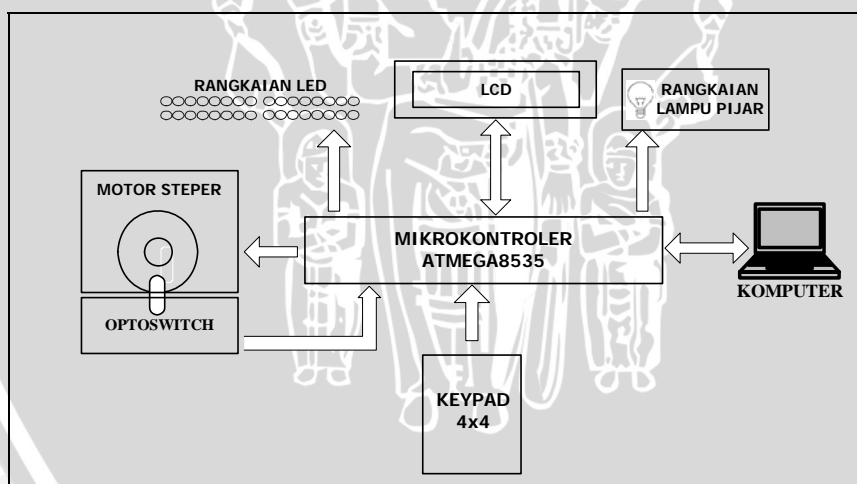
BAB IV PERANCANGAN DAN PEMBUATAN ALAT

4.1 Perancangan dan Pembuatan Perangkat Keras

4.1.1 Spesifikasi Alat

Berdasarkan latar belakang, perumusan masalah dan tujuan yang mendasari pembuatan modul pembelajaran mikrokontroler aplikatif ATMEGA8535 maka dibuatlah spesifikasi alat sebagai berikut:

- a. Modul terdiri dari pengisi program, rangkaian keypad, rangkaian motor stepper, rangkaian LED, rangkaian LCD, rangkaian lampu pijar dan rangkaian *Optoswitch*
- b. Pemrograman mikrokontroler menggunakan bantuan PC
- c. Pemantauan data yang ada pada tiap Port mikrokontroler menggunakan LED
- d. Modul mampu mensimulasikan fitur-fitur yang dimiliki mikrokontroler ATMEGA8535
- e. Modul disertai dengan contoh-contoh program untuk belajar dan program untuk mensimulasikan fitur-fitur yang dimiliki oleh mikrokontroler ATMEGA8535



Gambar 4.1 Blok Diagram Sistem

4.1.2 Blok Diagram

Perencanaan blok diagram dibuat berdasarkan fungsi-fungsi dan komponen yang diperlukan dalam perencanaan dan pembuatan modul. Tiap-tiap blok mewakili fungsi dan komponen yang dibutuhkan. Penempatan blok dan hubungan antar blok akan menentukan kinerja modul, sehingga diperlukan penempatan yang sesuai agar modul dapat bekerja sesuai dengan keinginan. Perencanaan rangkaian dilakukan pada setiap blok diagram.

4.1.3 Prinsip Kerja Sistem

Prinsip kerja modul pembelajaran mikrokontroler ATMEGA8535 adalah sebagai berikut: Setiap contoh program yang diberikan akan diisikan ke mikrokontroler ATMEGA8535 dengan bantuan komputer dan software PonyProg2000 menggunakan rangkaian pengisi program. Pada proses pengisian program, antara komputer dengan rangkaian pengisi program dihubungkan dengan kabel DB25 dengan memanfaatkan Port Parallel komputer.

Program yang sudah diisikan dapat disimulasikan menggunakan perangkat-perangkat yang ada pada modul ini. Pada rangkaian pengisi program terdapat konektor yang terhubung dengan setiap Port mikrokontroler, masing-masing Port dari mikrokontroler dihubungkan dengan peraga LED sebagai pemantau data yang ada di setiap Port dan melalui konektor-konektor yang tersedia setiap Port dihubungkan dengan perangkat lain yang tersedia seperti rangkaian LCD, rangkaian motor stepper, rangkaian Optoswitch, rangkaian lampu pijar dan rangkaian keypad. Penyambungan Port dengan perangkat-perangkat tersebut menggunakan kabel jumper yang dibuat dari kabel flat dan amphenol.

4.1.4 Perancangan Rangkaian

Komponen-komponen yang diperlukan dalam pembuatan modul ini dibagi menjadi beberapa blok. Blok pertama adalah rangkaian pengisi program komponen yang dibutuhkan adalah IC buffer 74HC244, resistor dan kapasitor. Blok kedua adalah rangkaian LED, komponen yang dibutuhkan adalah LED, resistor dan IC buffer 74LS245. Blok yang ketiga adalah driver motor stepper, komponen yang digunakan adalah penguat Darlington ULN 2003 dan satu motor stepper unipolar. Blok keempat adalah rangkaian Optoswitch, komponen yang dibutuhkan adalah *Optoswitch* 860D model "U", IC Schmitt Trigger 74LS14 dan resistor. Blok kelima adalah keypad, komponen yang digunakan adalah switch. Blok keenam adalah rangkaian pengendali lampu pijar komponen yang dibutuhkan adalah penguat Darlington ULN 2003 dan lampu pijar. Blok ketujuh adalah rangkaian LCD, komponen yang dibutuhkan adalah LCD dan resistor variabel. Sedangkan blok kedelapan adalah rangkaian RS232, komponen yang dibutuhkan adalah IC MAX232 dan kapasitor.

Perancangan modul ini harus memperhatikan jumlah arus maksimal yang dapat diberikan oleh mikrokontroler, hal ini dikarenakan begitu banyaknya komponen yang mungkin akan digunakan secara bersamaan baik pada pin atau Port yang sama maupun

pin atau Port yang berbeda. Penghitungan jumlah arus yang dibutuhkan menggunakan asumsi arus maksimal yang dibutuhkan oleh setiap komponen yang digunakan pada saat melakukan simulasi program dan hal ini terjadi pada saat program simulasi pengendalian suhu ruangan dijalankan. Simulasi ini menggunakan hampir semua pin yang dimiliki oleh mikrokontroler ATMEGA8535. Adapun distribusi penggunaan port dan jumlah arus yang dibutuhkan dapat dilihat dalam Tabel 4.1. Jumlah arus maksimal yang dapat diberikan oleh mikrokontroler ATMEGA8535 adalah 20 mA untuk setiap pin, 100 mA untuk total setiap port dan 200 mA untuk total semua port (Datasheet ATMEGA8535).

Tabel 4.1 Distribusi penggunaan Port untuk program simulasi pengendalian suhu ruangan

Komponen yang digunakan	Pin/Port yang digunakan	Jumlah arus yang dibutuhkan dari setiap pin
Driver LED 74LS245	Port A, B, C dan D	0,1 mA
Driver Motor Steper ULN 2003	Pin B4, B5, B6 dan B7	1,35 mA
LCD	Port C, PinD4, D6 dan D7	3 mA

Dari Tabel 4.1 dapat dihitung jumlah arus yang harus diberikan oleh masing-masing pin dan port, hasil perhitungan dapat dilihat dalam tabel 4.2 hingga Tabel 4.5

Tabel 4.2 Jumlah arus yang harus diberikan oleh Port A

Pin	Jumlah arus yang dibutuhkan	Total
A0	0,1 mA	0,1 mA
A1	0,1 mA	0,1 mA
A2	0,1 mA	0,1 mA
A3	0,1 mA	0,1 mA
A4	0,1 mA	0,1 mA
A5	0,1 mA	0,1 mA
A6	0,1 mA	0,1 mA
A7	0,1 mA	0,1 mA
Port A		0,8 mA

Tabel 4.3 Jumlah arus yang harus diberikan oleh Port B

Pin	Jumlah arus yang dibutuhkan	Total
B0	0,1 mA	0,1 mA
B1	0,1 mA	0,1 mA
B2	0,1 mA	0,1 mA
B3	0,1 mA	0,1 mA
B4	0,1 mA + 1,35 mA	1,45 mA
B5	0,1 mA + 1,35 mA	1,45 mA
B6	0,1 mA + 1,35 mA	1,45 mA
B7	0,1 mA + 1,35 mA	1,45 mA
Port B		6,2 mA

Tabel 4.4 Jumlah arus yang harus diberikan oleh Port C

Pin	Jumlah arus yang dibutuhkan	Total
C0	0,1 mA + 3 mA	3,1 mA
C1	0,1 mA + 3 mA	3,1 mA
C2	0,1 mA + 3 mA	3,1 mA
C3	0,1 mA + 3 mA	3,1 mA
C4	0,1 mA + 3 mA	3,1 mA
C5	0,1 mA + 3 mA	3,1 mA
C6	0,1 mA + 3 mA	3,1 mA
C7	0,1 mA + 3 mA	3,1 mA
Port C		24,8 mA

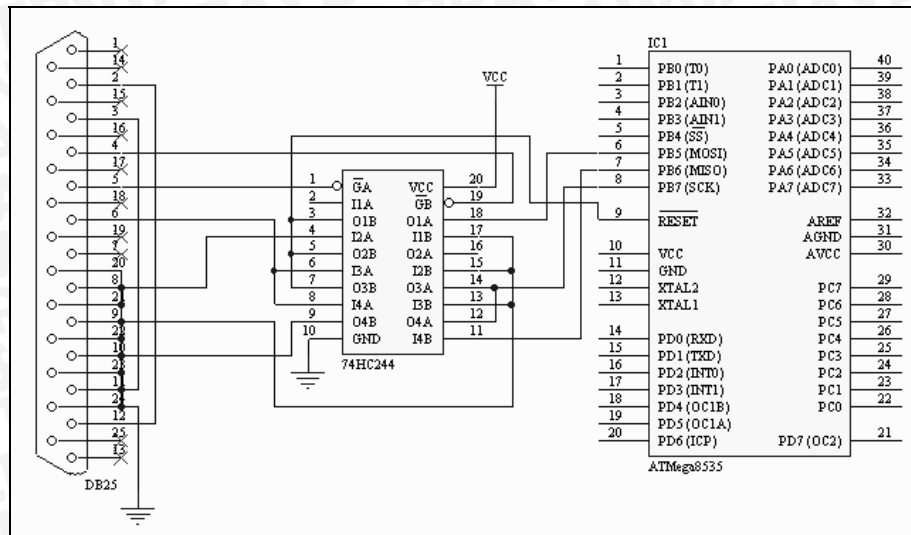
Tabel 4.5 Jumlah arus yang harus diberikan oleh Port D

Pin	Jumlah arus yang dibutuhkan	Total
D0	0,1 mA	0,1 mA
D1	0,1 mA	0,1 mA
D2	0,1 mA	0,1 mA
D3	0,1 mA	0,1 mA
D4	0,1 mA + 3 mA	3,1 mA
D5	0,1 mA	0,1 mA
D6	0,1 mA + 3 mA	3,1 mA
D0	0,1 mA + 3 mA	3,1 mA
Port D		9,8 mA

Dari perhitungan jumlah arus yang harus diberikan oleh masing-masing pin dan port dapat dilihat bahwa jumlah arus yang harus diberikan oleh masing-masing pin dan port masih di bawah kemampuan maksimal mikrokontroler ATMEGA8535 untuk memberikan arus sehingga dapat ditarik kesimpulan bahwa tidak terjadi masalah dalam hal konsumsi arus oleh komponen-komponen yang digunakan dalam simulasi program pengendalian suhu ruangan.

4.1.4.1 Perancangan Rangkaian Pengisi Program

Output Port paralel pada PC memiliki tegangan sebesar 5V dengan arus yang mengalir sebesar 12 mA (Peacock, 2001). Untuk antarmuka pengisian program diperlukan *buffer*, dalam hal ini *buffer* yang digunakan adalah IC 74HC244, yang berfungsi untuk menambah jumlah *fan-out* dari Port paralel PC dan mengamankan Port paralel dari hubungan secara langsung dengan beban karena *buffer* memiliki impedansi masukan yang tinggi. Port DB25 dihubungkan pada pin MISO, MOSI, SCK dan RESET mikrokontroler ATMEGA8535. Gambar rangkaian pengisi program dapat dilihat dalam Gambar 4.2.



Gambar 4.2 Rangkaian pengisi program mikrokontroler ATMEGA8535

4.1.4.2 Perancangan Rangkaian Reset

Perancangan rangkaian reset dapat dilihat dalam Gambar 4.3. Berdasarkan rangkaian tersebut didapat persamaan sebagai berikut:

$$E = V_R + V_C$$

$$E(t) = i(t)R + \frac{1}{C} \int i(t) dt$$

Dengan menggunakan transformasi Laplace diperoleh persamaan:

$$\frac{E(s)}{s} = I(s)R + \frac{1}{C} \frac{1}{s} I(s)$$

$$\frac{E(s)}{s} = I(s) \left(\frac{RC}{C} \frac{s+1}{s} \right)$$

$$I(s) = \left(\frac{C}{RC} \frac{s+1}{s} \right) E(s)$$

$$I(s) = \left(\frac{1}{s + \frac{1}{RC}} \right) \frac{E(s)}{R}$$

Dengan transformasi balik didapatkan persamaan $i(t)$ sebagai berikut

$$i(t) = \frac{E(t)}{R} e^{-\frac{t}{RC}}$$

Dari persamaan arus di atas besarnya V_o dapat dihitung.

$$V_o = E - V_R$$

$$V_o = E - \left(\frac{E(t)}{R} e^{-\frac{t}{RC}} \right) R$$

$$V_o = E \left(1 - e^{-\frac{t}{RC}} \right)$$

Besarnya t dari rangkaian dapat ditentukan dengan menggunakan perhitungan berikut:

$$\begin{aligned}
 V_o &= E(1 - e^{-\frac{t}{RC}}) & \ln\left(\frac{1}{1 - V_o/E}\right) &= \ln e^{\frac{t}{RC}} \\
 V_o/E &= 1 - e^{-\frac{t}{RC}} & \frac{t}{RC} &= \ln\left(\frac{1}{1 - V_o/E}\right) \\
 1 - V_o/E &= e^{-\frac{t}{RC}} & t &= RC \ln\left(\frac{1}{1 - V_o/E}\right) \dots\dots\dots(1) \\
 \frac{1}{(1 - V_o/E)} &= e^{\frac{t}{RC}}
 \end{aligned}$$

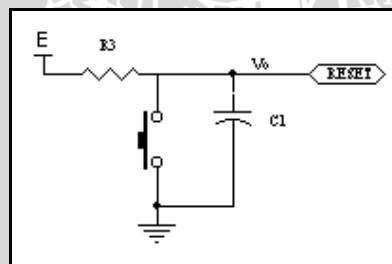
Dengan ketentuan besarnya nilai t harus lebih besar dari 50ns maka dapat dicari nilai R minimum dengan asumsi nilai kapasitor sebesar 1 μF dan tegangan output sebesar 0,85 VCC atau 4,25V maka menurut Persamaan (1) didapat:

$$\begin{aligned}
 50ns &= R \cdot 1\mu F \cdot \ln\left(\frac{1}{1 - 4,25/5}\right) \\
 R_{min} &= \frac{50 \cdot 10^{-9}}{10^{-6} \cdot \ln\left(\frac{1}{0,15}\right)} = \frac{50 \cdot 10^{-3}}{1,897} = 0,026\Omega
 \end{aligned}$$

Nilai R ditetapkan sebesar 10k Ω yang akan menghasikan waktu tunda sebesar :

$$\begin{aligned}
 t &= 10^4 \cdot 10^{-6} \cdot \ln\left(\frac{1}{0,15}\right) \\
 t &= 10^{-2} \cdot 1,897 \\
 t &= 18,97ms
 \end{aligned}$$

Waktu tunda sebesar 18,97 ms maka rangkaian diatas memenuhi syarat sebagai rangkaian reset.



Gambar 4.3 Rangkaian reset

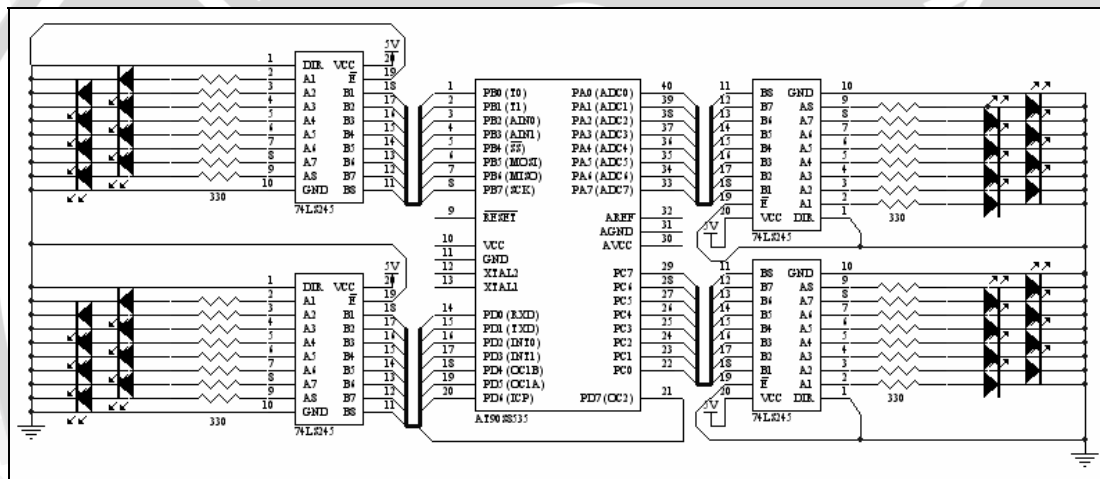
4.1.4.3 Perancangan Rangkaian Pengendali LED

Pada perancangan Modul Pembelajaran ini banyak digunakan komponen-komponen yang membutuhkan arus yang cukup besar termasuk arus yang dibutuhkan oleh LED yang sebesar 1 – 20 mA sehingga untuk pengendali LED digunakan IC buffer 74LS245 untuk memberikan arus yang cukup untuk mengendalikan LED tanpa membebani mikrokontroler. Pada perancangan pengendali LED, tegangan yang

digunakan sebesar 3,4 V sesuai dengan kemampuan maksimal *IC buffer* 74LS245 dengan demikian dapat dihitung besar nilai resistor yang akan digunakan maksimum dan minimum dapat ditentukan melalui perhitungan sebagai berikut :

$$\begin{aligned}
 3,4V &= R.I + V_{LED} & 3,4V &= R.I + V_{LED} \\
 R.I &= 3,4V - 1,7V & R.I &= 3,4V - 2V \\
 R \cdot 1mA &= 1,7 & R \cdot 20mA &= 1,4V \\
 R &= \frac{1,7V}{1mA} & R &= \frac{1,4V}{20mA} \\
 R &= 1,7k\Omega & R &= 70\Omega
 \end{aligned}$$

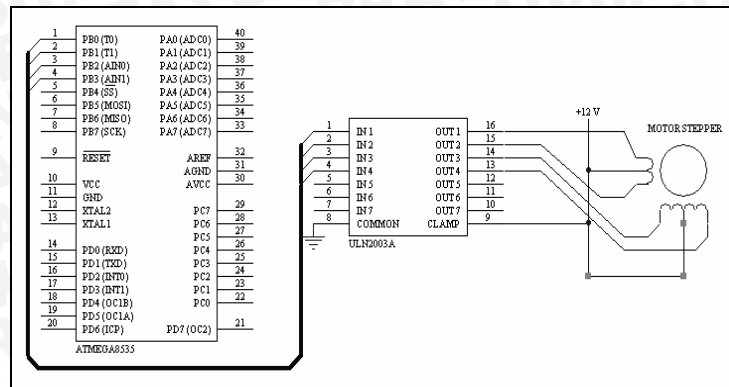
Untuk mendapatkan kecerahan LED yang cukup dapat diamati, berdasarkan perhitungan di atas maka digunakan resistansi sebesar 100 Ω sehingga arus yang melalui LED didapatkan sebesar 14 mA. Rangkaian pengendali LED menggunakan *IC buffer* 74LS245 dapat dilihat dalam Gambar 4.4



Gambar 4.4 Rangkaian Pengendali LED

4.1.4.4 Perancangan Rangkaian Pengendali Motor Steper

Keluaran setiap port dari mikrokontroler memiliki tegangan sebesar 4 V dengan arus sebesar 20 mA, arus sebesar 20 mA tersebut tidak mampu menggerakkan kumparan dari motor steper yang memiliki arus masukan minimum 300 mA. Oleh karena itu digunakan penguat Darlington (IC ULN 2003A) yang mampu memberikan arus hingga sebesar 500 mA dengan arus masukan maksimum 1,35 mA dan tegangan sebesar 3 V untuk logika tinggi. Rangkaian pengendali motor steper dapat dilihat dalam Gambar 4.5.

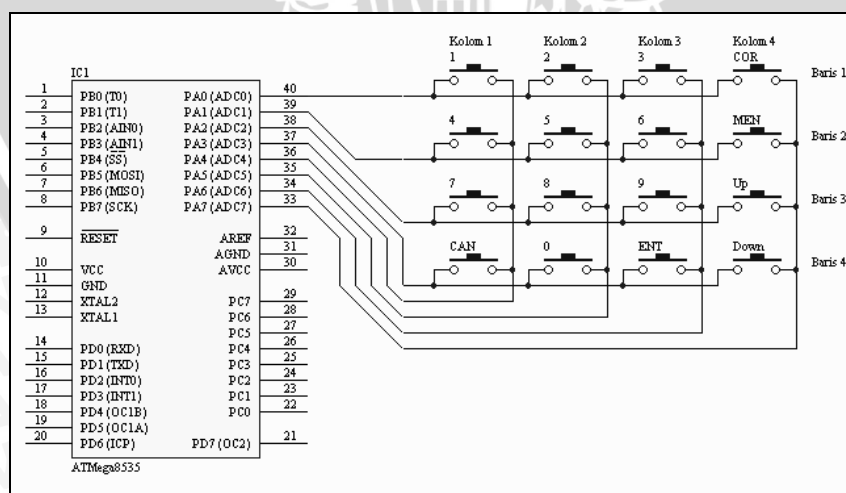


Gambar 4.5 Rangkaian Pengendali Motor Steper

4.1.4.5 Perancangan Rangkaian Keypad

Keypad yang digunakan dalam perancangan modul pembelajaran ini adalah keypad matrik 4x4. Empat kolom dihubungkan dengan port A4 – port A7, sedangkan empat barisnya dari keypad ini dihubungkan dengan port A0 – port A3. Port A4 – port A7 berfungsi sebagai keluaran, sedangkan port A0 – port A3 berfungsi sebagai masukan. Gambar dari perancangan rangkaian keypad dapat dilihat dalam Gambar 4.6

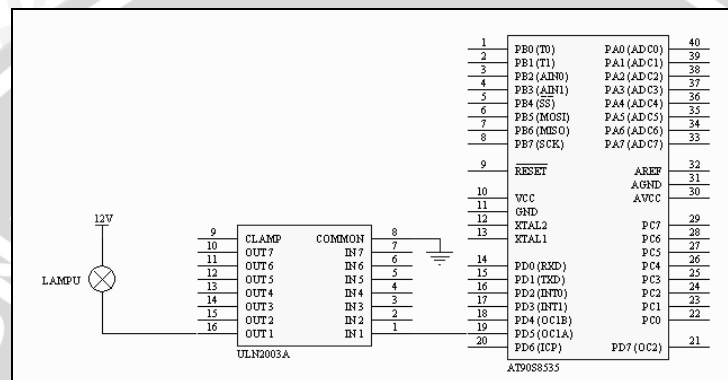
Sebagai masukan, port mikrokontroler memiliki resistor *pull-up* sebesar 20 kΩ yang dapat diaktifkan dengan menuliskan logika satu pada data register port tersebut. Apabila resistor *pull-up* diaktifkan, pin masukan mikrokontroler akan terbaca sebagai logika satu jika diambangkan. Dalam perancangan ini, resistor *pull-up* diaktifkan, agar pin yang tidak mendapat logika 0 dapat dibaca sebagai logika 1 dan rangkaian keypad dapat bekerja dengan baik.



Gambar 4.6 Rangkaian Keypad

4.1.4.6 Perancangan Rangkaian Pengendali Lampu Pijar

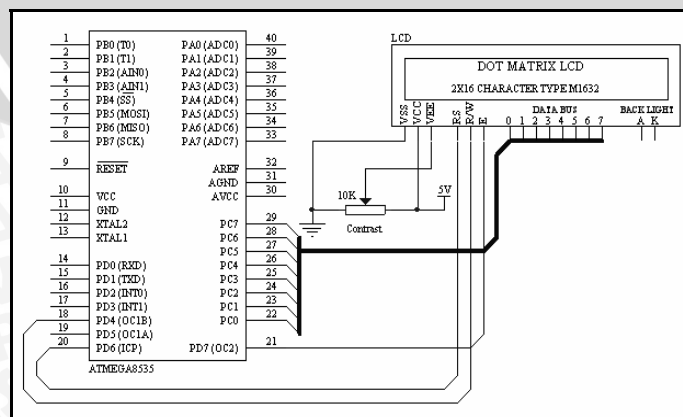
Simulasi penggunaan fitur PWM (*Pulse Width Modulation*) menggunakan lampu pijar sebagai media simulasi. Pada perancangan rangkaian lampu pijar digunakan lampu dengan tegangan sebesar 12 V dengan arus 100 mA. Besar arus yang dapat diberikan oleh mikrokontroler adalah 20 mA oleh karena itu digunakan IC ULN 2003 yaitu penguat darlington yang memiliki kemampuan memberikan arus sebesar 500 mA untuk memenuhi jumlah arus yang dibutuhkan oleh lampu pijar. Rangkaian skematik dari pengendali lampu pijar dapat dilihat dalam Gambar 4.7.



Gambar 4.7 Rangkaian Pengendali Lampu Pijar

4.1.4.7 Perancangan Rangkaian LCD

LCD yang digunakan dalam perancangan modul ini adalah LCD yang sudah dilengkapi dengan rangkaian pengendali sehingga dalam penggunaannya LCD ini dapat langsung di hubungkan dengan port mikrokontroler. Pengaturan contrast LCD dilakukan dengan memberikan tegangan antara 0 – 5 V pada pin VEE. Interpolasi tegangan menggunakan variabel resistor sebesar 10 kΩ yang dihubungkan dengan VCC dan GND. Rangkaian penghubungan LCD dengan mikrokontroler dapat dilihat dalam Gambar 4.8.



Gambar 4.8 Rangkaian LCD

4.1.4.8 Perancangan Rangkaian Optoswitch

Optoswitch yang digunakan adalah *Optoswitch* dengan tipe 860U. *Schmitt Trigger* 74LS14 digunakan untuk menyempurnakan sinyal pulsa dari rangkaian *Optoswitch* (mengubah bentuk gelombang keluaran rangkaian *Optoswitch* yang memiliki waktu transisi lambat menjadi gelombang keluaran dengan waktu transisi cepat) dan menghasilkan tegangan keluaran yang memenuhi level logika “1” dan “0”. Pada saat cahaya yang dipancarkan LED terhalang bagian yang tidak tembus cahaya maka rangkaian *Optoswitch* dan 74LS14 akan menghasilkan tegangan sekitar 0 V. Sebaliknya saat cahaya yang dipancarkan LED diterima oleh *phototransistor* maka rangkaian akan menghasilkan tegangan sekitar 5 V.

Untuk mengatur arus yang mengalir pada LED digunakan resistansi yang terhubung seri dengan LED (R_s) dan sebuah sumber tegangan (V_{cc}). Arus yang melalui LED diinginkan sebesar 20 mA. Besar nilai resistor dapat ditentukan melalui perhitungan sebagai berikut :

$$V_{cc} = R_s \cdot I_{LED} + V_{LED}$$

$$R_s \cdot I = 5 \text{ V} - 1,7 \text{ V}$$

$$R_s \cdot 20 \text{ mA} = 3,3 \text{ V}$$

$$R_s = \frac{3,3 \text{ V}}{20 \text{ mA}}$$

$$R_s = 165 \Omega$$

Untuk mendapatkan kecerahan LED yang cukup dapat diamati, berdasarkan perhitungan di atas maka digunakan resistansi (R_s) sebesar 180 Ω . Pada saat *phototransistor* terkena cahaya LED diinginkan *phototransistor* mengalami saturasi ($V_{ce_{sat}} = 0,4 \text{ V}$) dan arus kolektor saturasi ($I_{c_{sat}}$) 1,8 mA sehingga R_c dapat ditentukan melalui perhitungan berikut:

$$V_{cc} - V_{ce_{sat}} = R_c \cdot I_{c_{sat}}$$

$$R_c \cdot I_{c_{sat}} = 5 \text{ V} - 0,4 \text{ V}$$

$$R_c \cdot 1,8 \text{ mA} = 4,2 \text{ V}$$

$$R_c = \frac{4,2 \text{ V}}{1,8 \text{ mA}}$$

$$R_c = 2,3 \text{ k}\Omega$$

Berdasarkan perhitungan R_c di atas maka digunakan R_c sebesar 2 k Ω yang ada di pasaran, dengan demikian pada saat cahaya dari LED diterima oleh *phototransistor*, diharapkan tegangan keluaran dari 74LS14 adalah sekitar 5 V dan pada saat cahaya dari

Tabel 4.6 Tabel contoh program untuk mensimulasikan fitur-fitur ATMEGA8535 dan bersifat aplikatif

Fitur ATMEGA8535	Contoh program
1. Programmable I/O	<ul style="list-style-type: none"> • Membuat LED Menyala Berjalan • Menampilkan Karakter pada LCD • Membaca Karakter dari Tombol Keypad dan Menampilkannya pada LCD • Menjalankan Motor Steper
2. Timer/Counter	<ul style="list-style-type: none"> • Membuat LED menyala setiap 5 detik • Membuat loop tak berhingga sebagai simulasi kondisi <i>hang</i> • Aplikasi <i>Dimmer</i> pada Lampu Pijar
3. On-chip Analog Comparator	<ul style="list-style-type: none"> • Membuat LED menyala jika suatu tegangan melebihi tegangan referensi
4. 512 Bytes EEPROM	<ul style="list-style-type: none"> • Menyimpan data ke EEPROM dan menampilkan data dari EEPROM ke LCD
5. Programmable UART	<ul style="list-style-type: none"> • Mengirim/menerima data dari/ke PC
6. 8-Channel, 10-bit ADC	<ul style="list-style-type: none"> • Menampilkan Nilai Tegangan DC pada LCD
7. Eksternal Interrupt	<ul style="list-style-type: none"> • Simulasi Pengaturan Suhu Ruangan

4.2.1 Diagram Alir

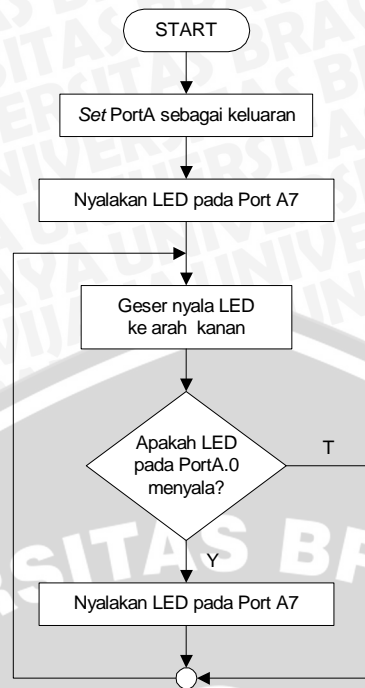
Pembuatan diagram alir dilakukan untuk mempermudah proses perancangan program. Diagram alir meliputi semua contoh program simulasi fitur mikrokontroler ATMEGA8535 dan program aplikatif. Diagram alir setiap program dapat dilihat dalam Gambar 4.11 sampai Gambar 4.22

4.2.2 Pembuatan Perangkat Lunak

Pembuatan perangkat lunak menggunakan bahasa pemrograman C dengan bantuan perangkat lunak Code Vision AVR sebagai *text editor* dan *Compiler*, dan perangkat lunak AVR studio 4 sebagai *simulator*. Pemrograman menggunakan bahasa C dikarenakan bahasa C merupakan bahasa pemrograman tingkat tinggi yang mendekati bahasa manusia sehingga lebih mudah dipahami oleh pembuat program. *Listing* setiap program yang sudah dibuat dapat dilihat dalam lampiran.

4.2.2.1 Program LED Menyala Berjalan

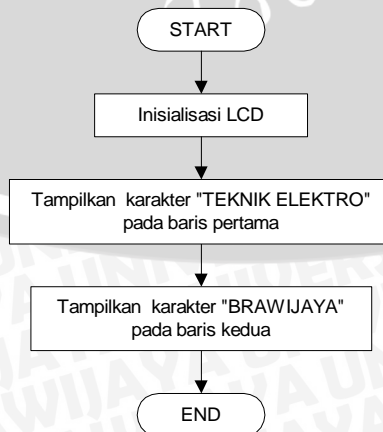
Program ini akan menyalakan LED pada Port A secara bergantian dan berurutan ke arah kanan. Prinsip kerja program adalah sebagai berikut: pertama program akan menentukan fungsi Port A sebagai keluaran kemudian dilanjutkan dengan memberikan logika 1 pada Port A7 kemudian menggeser logika 1 tersebut ke kanan terus menerus dan jika LED yang menyala pada Port A0 maka dilakukan penyalakan kembali LED pada Port A7 sehingga nyala LED akan tampak seolah berjalan dari kiri ke kanan. Diagram alir dari program tersebut dapat dilihat dalam Gambar 4.11.



Gambar 4.11 Diagram alir Program LED menyala berjalan

4.2.2.2 Program Menampilkan Karakter pada LCD

Program ini akan menampilkan karakter yang sudah kita tentukan sebelumnya yaitu TEKNIK ELEKTRO pada baris pertama dan BRAWIJAYA pada baris yang kedua LCD. Mode pengisian data pada LCD yang digunakan adalah 8 bit dan menggunakan Port C dan Port D dalam aplikasinya. Prinsip kerja program adalah sebagai berikut: Pertama kali program akan melakukan inisialisasi LCD meliputi jenis pengiriman data dan tampilan awal LCD. Berikutnya program akan menampilkan karakter pertama yaitu TEKNIK ELEKTRO pada baris pertama LCD dan dilanjutkan dengan menampilkan karakter kedua yaitu BRAWIJAYA pada baris kedua LCD. Diagram alir program dapat dilihat dalam Gambar 4.12.



Gambar 4.12 Diagram Alir Program Menampilkan Karakter Pada LCD



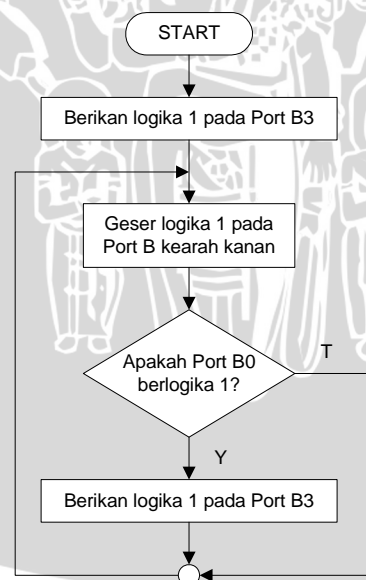
4.2.2.3 Program Menggerakkan Motor Stepper

Pada pembuatan modul ini Motor Steper yang digunakan jenis unipolar dengan urutan data untuk dapat berputar searah jarum jam yang dapat dilihat dalam Tabel 4.7.

Tabel 4.7 Tabel urutan logika pada pin motor stepper

Step	Pin 1	Pin 2	Pin 3	Pin 4
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1

Untuk arah putaran berlawanan dengan arah jarum jam cukup dengan membalik arah perubahan logika pada setiap pin. Pada contoh program menggerakkan motor stepper ini digunakan arah putaran searah dengan jarum jam dan motor akan berputar terus menerus tanpa berhenti. Prinsip kerja program adalah sebagai berikut: program akan memberikan logika 1 pada Port B3 yang terhubung dengan pin pertama motor stepper dan menggeser logika 1 tersebut ke arah kanan secara terus menerus dan jika logika 1 sudah berada pada Port B0 yang terhubung dengan pin keempat motor stepper maka logika satu akan diberikan kembali pada Port B3. Kecepatan putaran motor akan ditentukan oleh waktu tunda yang diberikan pada setiap pergeseran logika 1. Diagram alir program menggerakkan motor stepper dapat dilihat dalam Gambar 4.13.



Gambar 4.13 Diagram alir Program Menggerakkan Motor Stepper

4.2.2.4 Program Memasukkan Karakter dari Keypad dan Menampilkan pada LCD

Program ini akan menampilkan setiap karakter yang tertera pada *keypad* jika tombol pada *keypad* ditekan. Prinsip kerja program ini adalah sebagai berikut: Program akan melakukan *scan keypad* secara terus menerus. Apabila ada tombol pada *keypad* yang ditekan maka program akan menampilkan karakter sesuai dengan yang tertera pada *keypad* dan kembali melakukan *scan keypad*. Apabila tidak ada tombol yang ditekan dalam satu siklus maka program akan melakukan *scan keypad* kembali. Pada proses *scan keypad* program akan memberikan logika 0 pada baris pertama dan logika 1 pada baris lainnya, kemudian program akan memeriksa kondisi logika pada kolom pertama hingga kolom keempat apakah ada yang berlogika 0 untuk mengetahui tombol mana yang ditekan pada baris pertama. Tombol yang ditekan adalah tombol pada kolom yang berlogika 0 pada baris pertama. Susunan baris dan kolom pada *keypad* dapat dilihat dalam Gambar 4.6. Jika pada baris pertama tidak ditemukan tombol yang ditekan maka pemberian logika 0 dilakukan pada baris kedua dan seterusnya dengan proses pemeriksaan yang sama mulai dari kolom pertama hingga kolom keempat. Alur program menampilkan karakter dari *keypad* pada LCD dan *subrutin* program *scan keypad* dapat dilihat dalam Gambar 4.14.

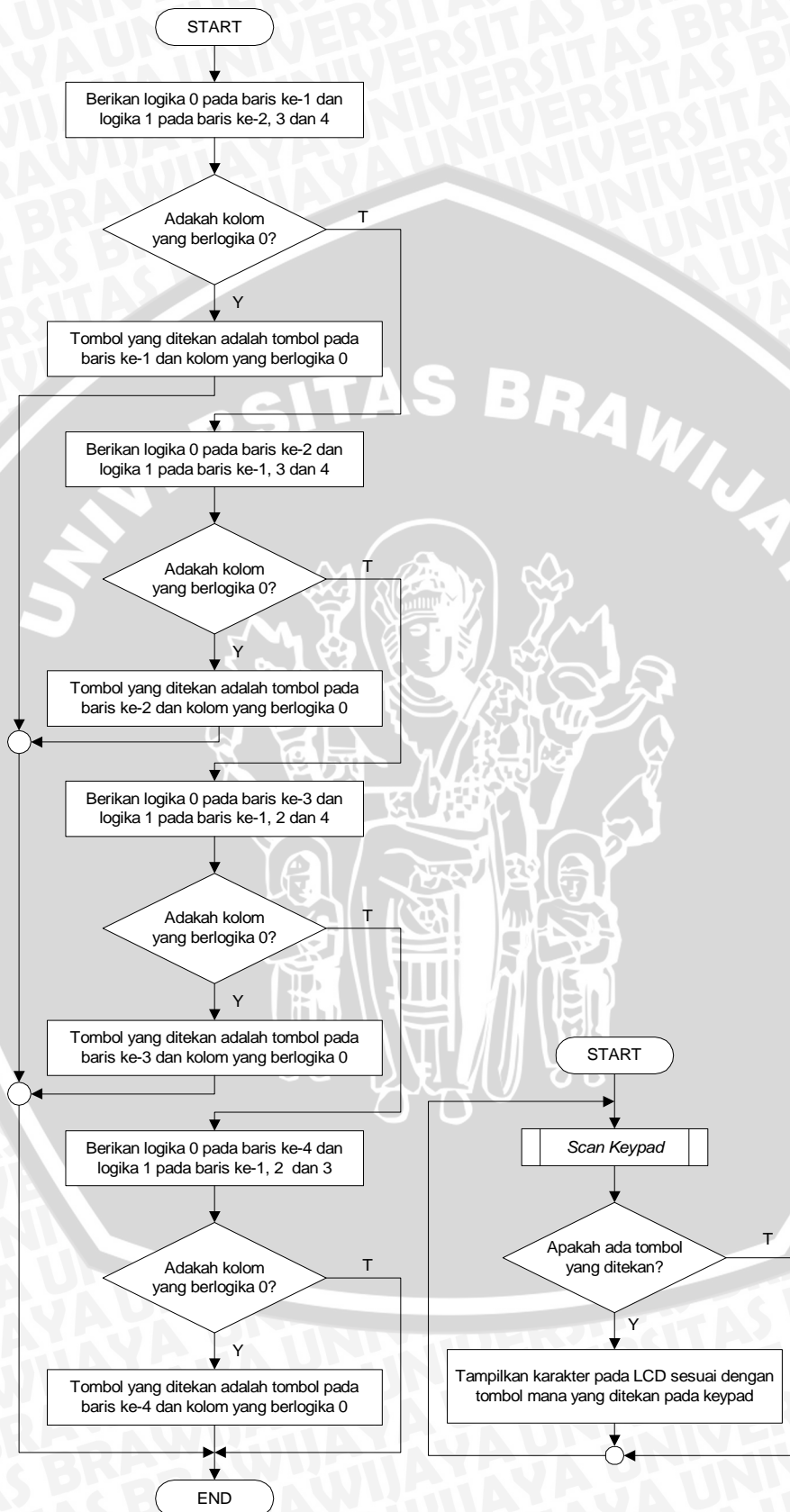
4.2.2.5 Program Menampilkan Nilai Tegangan DC pada LCD

Program ini akan menampilkan nilai tegangan yang diberikan pada ADC internal mikrokontroler ATMEGA8535 pada LCD. Tegangan masukan yang diberikan pada masukan analog adalah 0 V – 5 V, dengan tegangan referensi ADC (V_{reff}) sebesar 5 V dan ADC yang digunakan 8 bit didapatkan resolusi tegangan setiap bitnya menurut persamaan berikut:

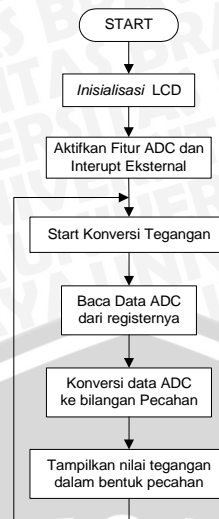
$$\begin{aligned} \text{Resolusi}_{\text{ADC}} &= \frac{V_{\text{reff}}}{2^n - 1} \\ &= \frac{5 \text{ V}}{2^8 - 1} \\ &= 0,02 \text{ V/bit} \end{aligned}$$

Prinsip kerja program adalah sebagai berikut: Program akan melakukan inisialisasi LCD yang digunakan dan mengaktifkan fitur ADC internal mikrokontroler ATMEGA8535. Program selanjutnya akan melakukan konversi tegangan dan data hasil konversi akan diambil dari registernya untuk diolah menjadi data nilai tegangan dalam bentuk pecahan dan ditampilkan pada LCD. Selanjutnya program akan kembali

melakukan konversi tegangan kembali. Demikian secara terus-menerus program akan berjalan, alur program dapat dilihat dalam Gambar 4.15.



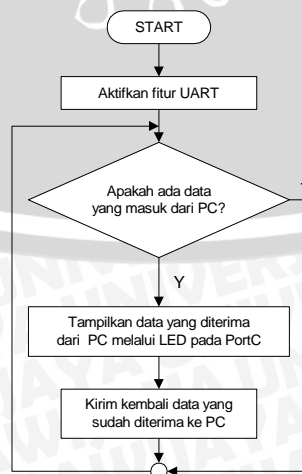
Gambar 4.14 Diagram alir Program *Scan Keypad* dan Menampilkan Karakter *Keypad* pada LCD



Gambar 4.15 Diagram alir Program Menampilkan Nilai Tegangan DC pada LCD

4.2.2.6 Program Mengirim/Menerima Data dari/ke PC

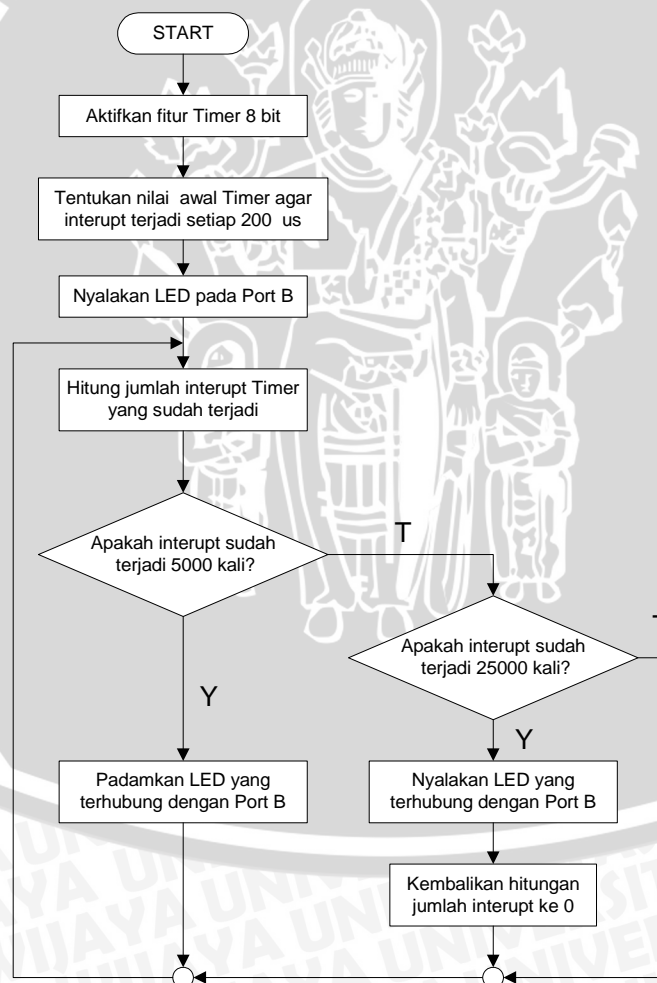
Program ini berfungsi untuk menerima atau mengirim data dari/ke PC menggunakan fasilitas *Terminal* dari Software Code Vision AVR, data yang diterima oleh mikrokontroler dari PC akan ditampilkan melalui LED dan dikirimkan kembali ke PC. Pada program ini data yang diolah adalah data dengan lebar 8 bit saja. Prinsip kerja dari program ini adalah sebagai berikut: Pertama kali program akan mengaktifkan fitur UART mikrokontroler kemudian memeriksa apakah ada data yang diterima dari PC dan jika mikrokontroler mendapatkan data kiriman dari PC maka program akan mengirimkan data pada Port C dan data tersebut akan terlihat melalui nyala LED yang terhubung dengan Port tersebut, setelah data ditampilkan pada LED data akan dikirimkan kembali ke PC. Data yang diterima oleh PC dari mikrokontroler dapat dilihat pada buffer Terminal Software Code Vision AVR. Diagram alir dari program ini dapat dilihat dalam Gambar 4.16.



Gambar 4.16 Diagram Alir Program Menerima/Mengirim Data dari/ke PC

4.2.2.7 Program Menyalakan LED Setiap 5 Detik

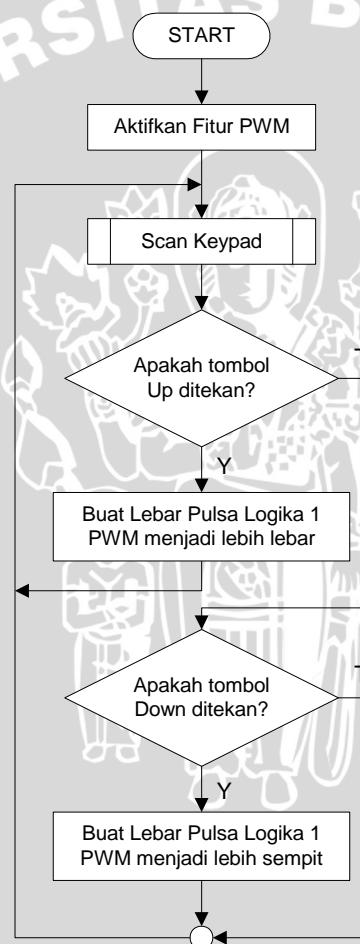
Program akan membuat LED yang terhubung dengan Port B menyala setiap 5 detik sekali selama 1 detik secara terus menerus. Prinsip kerja program adalah sebagai berikut: Pertama program akan mengaktifkan Timer 0 dan menentukan frekuensi Clock yang digunakan yaitu 1 Mhz dan menentukan nilai awal Timer agar interrupt yang terjadi berlangsung setiap 200 μ s dan menyalakan LED pada Port B untuk pertama kali. Selanjutnya program akan menghitung jumlah interrupt yang sudah terjadi dan jika sudah terjadi interrupt sebanyak 5000 kali yang berarti sudah terjadi selang waktu selama 1 detik maka LED akan dipadamkan. Program akan menghitung kembali jumlah interrupt yang sudah terjadi dengan melanjutkan hitungan semula dan jika sudah terjadi 25000 kali interrupt Timer 0 atau selang waktu 4 detik kemudian maka LED akan dinyalakan dan program mulai menghitung kembali jumlah interrupt dari nol. Diagram alir program dapat dilihat dalam Gambar 4.17.



Gambar 4.17 Diagram Alir Program Menyalakan LED Setiap 5 Detik

4.2.2.8 Program Aplikasi *Dimmer* pada Lampu Pijar

Program digunakan untuk melakukan proses *dimmer* pada lampu pijar dengan memanfaatkan fasilitas 16-bit Timer/Counter1 mikrokontroler ATMEGA8535 sebagai Fast PWM dan tombol Up atau Down pada *keypad*. Prinsip kerja program adalah sebagai berikut: Pertama kali program akan mengaktifkan fitur 16-bit Timer/Counter1 sebagai Fast PWM dan melakukan *scan keypad*, apabila tombol Up pada *keypad* ditekan maka program akan memperlebar periode logika 1 PWM sedangkan jika tombol Down yang ditekan maka program akan mempersempit periode logika 1. *Scan keypad* akan dilakukan secara terus menerus selama program masih dijalankan. Diagram alir program dapat dilihat dalam Gambar 4.17

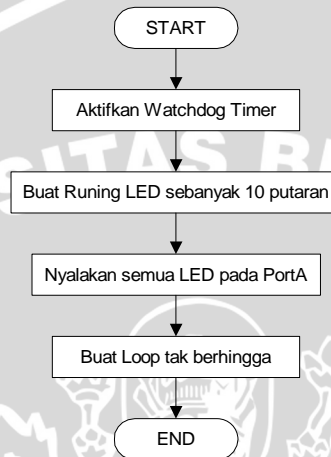


Gambar 4.18 Diagram alir Program Aplikasi *Dimmer* pada Lampu Pijar

4.2.2.9 Program Membuat Loop Tak Berhingga Sebagai Simulasi Kondisi Hang

Program ini digunakan untuk mensimulasikan fasilitas Watchdog yang dimiliki oleh mikrokontroler ATMEGA8535. Prinsip kerja program adalah sebagai berikut: Pertama program akan mengaktifkan fasilitas Watchdog dengan waktu *time out* Watchdog adalah 2,1 s kemudian program akan membuat LED pada PortA menyala

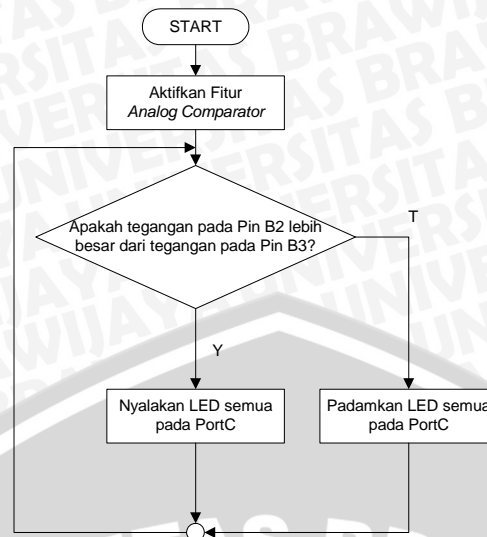
secara bergantian dari Port A7 hingga Port A0 dan berputar lagi sebanyak 10 kali (*running LED*). Setelah *running LED* selesai maka semua LED pada PortA dinyalakan dan program akan mengeksekusi loop tak berhingga yang digunakan sebagai simulasi kondisi hang dimana program mengeksekusi perintah yang sama secara terus-menerus. Setelah 2,1 s fitur *Watchdog* akan bekerja dan mereset mikrokontroler dan program akan dieksekusi kembali mulai dari awal. Diagram alir program dapat dilihat dalam Gambar 4.19.



Gambar 4.19 Diagram alir Program Membuat Loop Tak Berhingga Sebagai Simulasi Kondisi Hang

4.2.2.10 Program Membuat LED Menyala Jika Suatu Tegangan Melebihi Tegangan Referensi

Program ini digunakan untuk mensimulasikan fitur Analog Comparator yang dimiliki oleh mikrokontroler ATMEGA8535, program akan membandingkan tegangan yang diberikan pada kedua pin masukan Analog Comparator dengan range tegangan 0 V – 5 V. Prinsip kerja program adalah sebagai berikut: Pertamakali akan mengaktifkan fitur Analog Comparator mikrokontroler dan secara terus menerus memeriksa kondisi tegangan masukan pada pin masukan Analog Comparator. Jika tegangan pada pin positif Analog Comparataor lebih besar dari pin negatif maka semua LED yang terhubung pada Port C akan dinyalakan oleh program namun jika yang terjadi sebaliknya maka LED tersebut akan dipadamkan. Diagram alir program dapat dilihat dalam Gambar 4.19



Gambar 4.20 Diagram alir program Menyalakan LED Jika Suatu Tegangan Melebihi Tegangan Referensi

4.2.2.11 Program Menyimpan Data ke EEPROM dan Menampilkan Data dari EEPROM pada LCD

Program ini digunakan untuk mensimulasikan fitur EEPROM internal yang dimiliki oleh mikrokontroler ATMEGA8535. Prinsip kerja program adalah: pertama program akan melakukan inisialisasi LCD dan menampilkan karakter pembuka “Program EEPROM”, kemudian program akan menampilkan karakter “Masukkan Data”. Program akan melakukan *scan keypad* secara terus menerus, jika tombol ‘MEN’ ditekan maka muncul karakter “Hapus Masukan” pada baris pertama LCD untuk membersihkan tampilan LCD sebelumnya dan semua data yang disimpan sementara akan ditimpa dengan data yang baru. Apabila tombol “ENT” yang ditekan maka akan tampil karakter “Simpan...” pada baris kedua LCD dan data yang disimpan sementara dalam buffer dipindahkan menuju EEPROM internal. Apabila tombol “COR” ditekan maka muncul karakter “Baca Data” pada baris kedua LCD dan dilakukan pengambilan data dari EEPROM untuk ditampilkan pada LCD. Jika tombol-tombol angka ditekan maka karakter tombol yang ditekan akan ditampilkan pada LCD dan data angka tersebut disimpan sementara di dalam buffer. Diagram alir program dapat dilihat dalam Gambar 4.21.

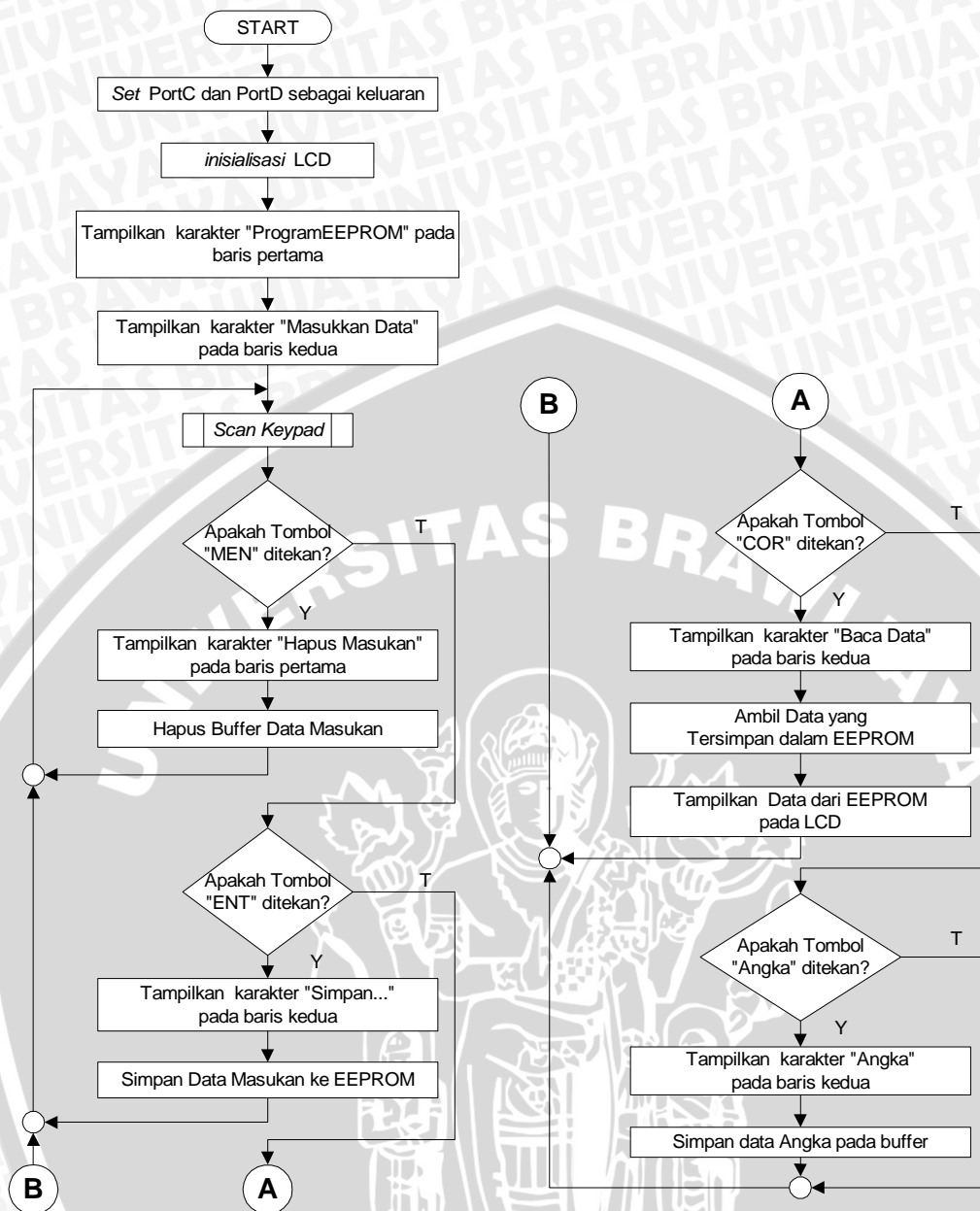
4.2.2.12 Program Simulasi Pengaturan Suhu Ruangan

Program berikut ini ditujukan untuk mewakili suatu kondisi dimana mikrokontroler ATMEGA8535 digunakan untuk mengendalikan sistem yang cukup kompleks. Simulasi ini menggunakan beberapa perangkat yang ada dan sudah dirancang sebelumnya seperti motor stepper, LCD, rangkaian *Optoswitch* dan LED. Prinsip kerja

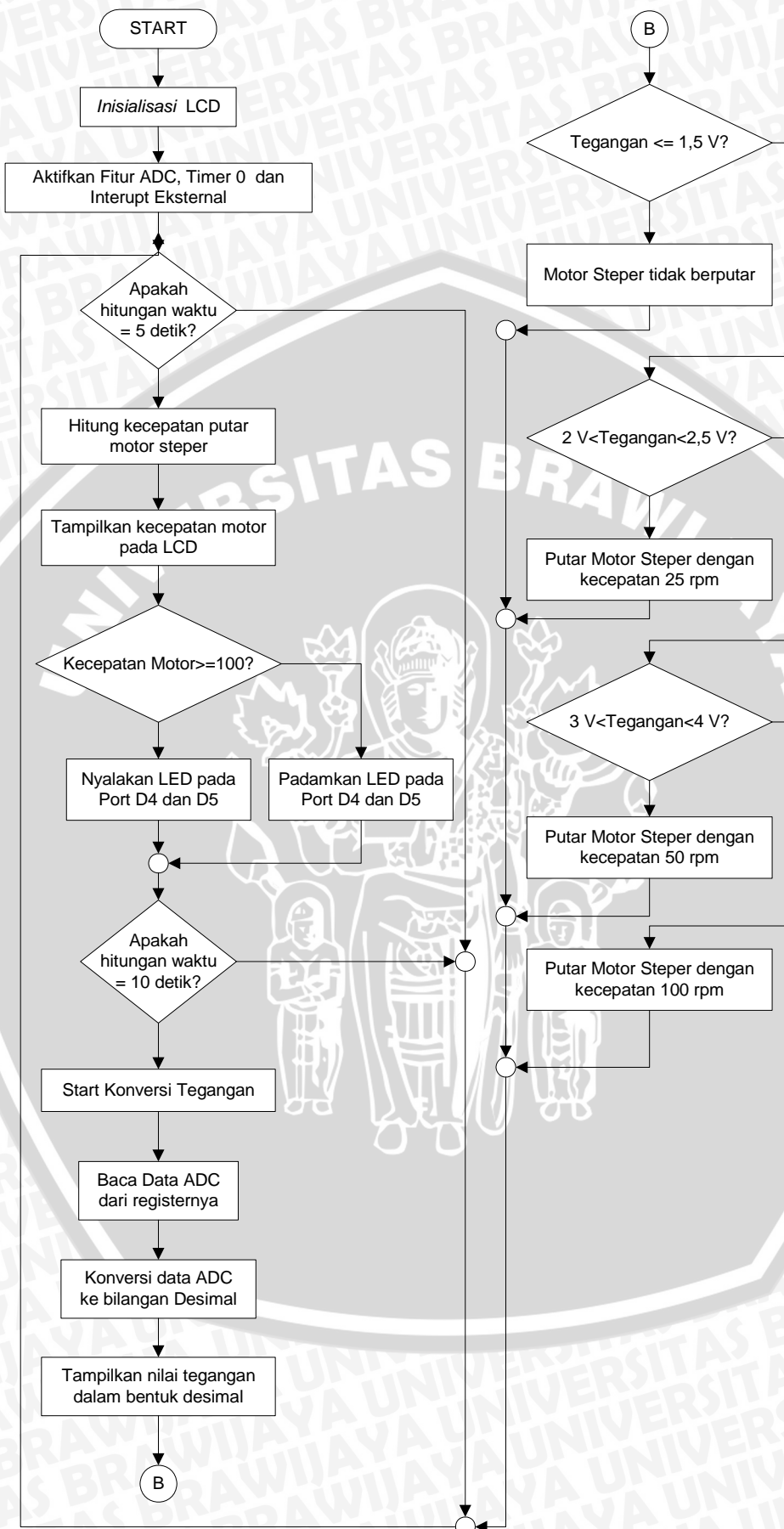
sistem adalah sebagai berikut: Misalkan suhu di suatu ruangan akan diatur secara otomatis, jika suhu ruangan meningkat maka sebuah kipas angin akan berputar dengan kecepatan tertentu dan jika suhu tetap meningkat maka kecepatan kipas angin juga akan bertambah juga, Penambahan kecepatan kipas angin dilakukan pada batas-batas suhu yang sudah ditentukan sebagai contoh jika suhu ruangan mencapai $25\text{ }^{\circ}\text{C}$ maka kipas akan berputar dengan kecepatan 25 rpm sedangkan jika suhu ruangan mencapai $50\text{ }^{\circ}\text{C}$ maka kipas akan berputar dengan kecepatan 50 rpm dan jika kipas berputar pada batas putaran maksimum maka tanda bahaya akan menyala.

Sensor suhu diwakili oleh tegangan yang diberikan pada masukan ADC Internal dan suhu akan ditampilkan pada LCD, jika tegangan yang diberikan mencapai $2\text{ V} - 2,5\text{ V}$ maka kipas yang diwakili oleh motor stepper akan berputar dengan kecepatan 25 rpm. Jika tegangan terus bertambah mencapai $3\text{ V} - 4\text{ V}$ maka motor stepper akan berputar lebih cepat yaitu 50 rpm sedangkan saat tegangan mencapai $4,5\text{ V}$ atau lebih maka motor stepper akan berputar dengan kecepatan 100 rpm. Ketika kecepatan motor stepper mencapai 100 rpm, LED yang pada Port D.4 dan D.5 akan menyala untuk mewakili tanda bahaya. Jika tegangan yang diberikan lebih kecil atau sama dengan $1,5\text{ V}$ maka kipas tidak berputar. Perhitungan kecepatan kipas angin menggunakan pulsa-pulsa yang dihasilkan oleh rangkaian *Optoswitch* yang dipasang di sela-sela kipas, saat cahaya LED *Optoswitch* terhalangi oleh kipas maka rangkaian *Optoswitch* mengirimkan logika 0 dan saat LED *Optoswitch* tidak terhalangi oleh kipas maka *Optoswitch* mengirimkan logika 1 pada Pin Interrupt Eksternal mikrokontroler.

Prinsip kerja program adalah: Program akan melakukan inisialisasi LCD yang digunakan dan mengaktifkan fitur ADC internal, Timer/Counter0 serta fasilitas Interrupt Eksternal. Selanjutnya program akan mengkonversi tegangan yang diberikan pada masukan ADC setiap 10 detik. Data hasil konversi yang dalam bentuk bilangan heksa akan diambil dari registernya dan dikonversi dalam bilangan pecahan untuk ditampilkan pada LCD. Selanjutnya akan dilakukan pemeriksaan nilai tegangan oleh program untuk menentukan kecepatan motor stepper sedangkan kecepatan motor stepper diperiksa setiap 1 detik saat Interrupt Timer terjadi apakah sudah mencapai 100 rpm dan jika kecepatan tersebut sudah dicapai maka LED pada Port D4 dan D5 dinyalakan. Diagram alir program dapat dilihat dalam Gambar 4.22.



Gambar 4.21 Diagram alir program utama Menyimpan Data ke EEPROM dan Menampilkan Data dari EEPROM pada LCD

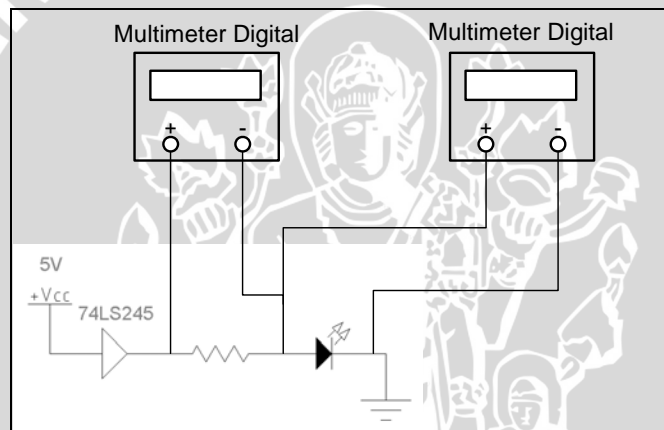


Gambar 4.22 Diagram alir Fungsi Interrupt ADC program Simulasi Pengaturan Suhu Ruangan

BAB V PENGUJIAN DAN ANALISIS

5.1 Pengujian Rangkaian Pengendali LED

Pengujian ini bertujuan untuk mengetahui besar arus yang melewati setiap LED dan mengetahui penyimpangannya terhadap nilai yang diinginkan. Pengujian dilakukan dengan cara memberikan tegangan sebesar 5 V yang setara dengan logika 1 pada setiap masukan *IC buffer* 74LS245. Pengukuran arus yang melewati setiap LED (I_{LED}) dilakukan secara tidak langsung yaitu dengan membagi nilai tegangan pada resistor (V_R) dengan nilai resistor (R). Pengukuran tegangan juga dilakukan pada setiap LED (V_{LED}) untuk mengetahui jatuh tegangan yang terjadi pada LED. Diagram blok pengujian rangkaian pengendali LED dapat dilihat dalam Gambar 5.1.



Gambar 5.1 Pengujian Rangkaian Pegendali LED

Hasil pengujian rangkaian pengendali LED ditunjukkan dalam Tabel 5.1. Dari tabel tersebut dapat dilihat bahwa terjadi penyimpangan nilai I_{LED} yang cukup signifikan pada Pin A4, B0 dan D1 masing-masing sebesar 9,22 mA, 9,50 mA dan 9,02 mA hal ini dikarenakan V_{LED} pada masing-masing LED tersebut cukup besar jika dibandingkan dengan V_{LED} pada Pin lainnya. Dengan mengabaikan I_{LED} pada Pin A4, B0 dan D1 didapatkan rata-rata I_{LED} adalah sebesar 13 mA, I_{LED} rata-rata tersebut sudah mendekati nilai I_{LED} hasil perancangan untuk setiap LED yaitu 14 mA. Dengan arus yang melalui masing-masing LED, nyala LED masih dapat dilihat dengan jelas. Secara keseluruhan dapat dilihat bahwa rangkaian pengendali LED dapat bekerja dengan baik

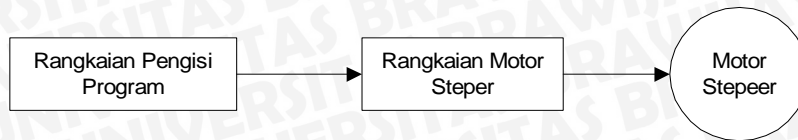
Tabel 5.1 Hasil pengujian rangkaian pegendali LED

Pin	V_R (V)	R (Ω)	V_{LED} (V)	I_{LED} (mA)
A0	1,28	102,0	1,67	12,55
A1	1,32	100,2	1,63	13,17
A2	1,22	103,1	1,89	11,83
A3	1,39	102,0	1,63	13,63
A4	0,96	104,1	2,26	9,22
A5	1,37	100,1	1,66	13,69
A6	1,37	98,8	1,66	13,87
A7	1,40	100,0	1,63	14,00
B0	0,97	102,1	2,13	9,50
B1	1,32	100,0	1,63	13,20
B2	1,15	100,0	1,96	11,50
B3	1,37	99,6	1,63	13,76
B4	1,41	102,2	1,62	13,80
B5	1,39	101,2	1,63	13,74
B6	1,38	99,7	1,65	13,84
B7	1,41	102,5	1,63	13,76
C0	1,28	100,1	1,64	12,79
C1	1,31	99,4	1,62	13,18
C2	1,36	101,3	1,63	13,43
C3	1,36	101,1	1,63	13,45
C4	1,37	102,2	1,64	13,41
C5	1,36	99,2	1,64	13,71
C6	1,36	99,8	1,66	13,63
C7	1,38	102,1	1,63	13,52
D0	1,29	103,0	1,67	12,52
D1	0,92	102,0	2,25	9,02
D2	1,37	102,4	1,66	13,38
D3	1,35	99,7	1,66	13,54
D4	1,39	100,6	1,63	13,82
D5	1,36	98,9	1,66	13,75
D6	1,38	100,8	1,66	13,69
D7	1,39	99,3	1,64	14,00

5.2 Pengujian Rangkaian Motor Steper

Pengujian pada rangkaian motor stepper bertujuan untuk mengetahui apakah rangkaian dapat bekerja dengan baik sesuai dengan yang diharapkan. Pengujian dilakukan dengan menggunakan program uji Motor Steper seperti yang terlampir pada Lampiran hal 77.

Hasil pengujian menggunakan program uji dapat dilihat dalam Tabel 5.2. Dari tabel tersebut dapat diamati sudut putaran hasil pengukuran (Θ_{Ukur}) selalu sama dengan sudut putaran yang diinginkan dalam pengujian (Θ_{Hit}) dengan demikian dapat disimpulkan bahwa motor stepper dan rangkaian pengendalinya dapat bekerja dengan baik.



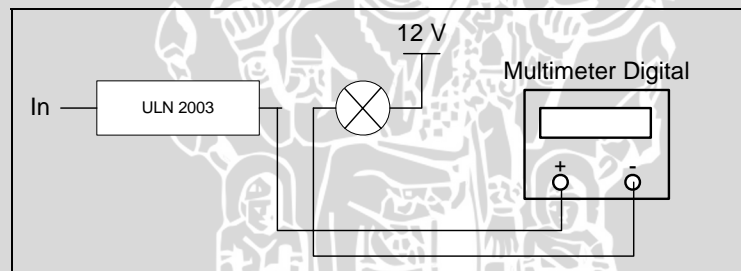
Gambar 5.2 Pengujian Rangkaian Pegendali Motor Steper

Tabel 5.2 Pengujian Motor Steper

Θ_{HIT}	Θ_{Ukur}	Keterangan
90°	90°	50 Langkah
180°	180°	100 Langkah
270°	270°	150 Langkah
360°	360°	200 Langkah

5.3 Pengujian rangkaian pengendali lampu pijar

Pengujian ini bertujuan untuk mengetahui besar arus yang mengalir melalui lampu pijar. Pengujian dilakukan dengan memberikan tegangan catu sebesar 12 V pada IC penguat Darlington ULN 2003 dan tegangan sebesar 5V yang setara dengan logika 1 pada masukannya. Arus yang melalui lampu pijar diukur secara langsung dengan menggunakan multimeter seperti yang terlihat dalam diagram blok pengujian pada Gambar 5.3.



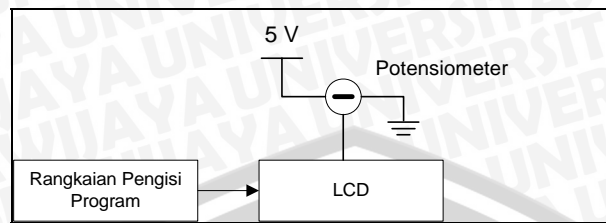
Gambar 5.3 Pengujian Rangkaian Pegendali Lampu Pijar

Hasil pengukuran menunjukkan bahwa arus yang melalui lampu pijar sebesar 158 mA dan lampu menyala dengan terang. Pada perancangan digunakan lampu pijar dengan tegangan catu sebesar 12 V dengan ketahanan arus 0,1 A (label lampu pijar) namun dari hasil pengujian menunjukkan bahwa arus yang melalui lampu pijar adalah 158 mA dengan nyala lampu pijar cukup terang. Dari hasil pengujian didapatkan bahwa perbedaan besar arus hasil pengujian dengan perancangan tidak mempengaruhi kinerja rangkaian karena lampu pijar dapat menyala dengan baik tanpa mengalami masalah.

5.4 Pengujian Rangkaian LCD

Pengujian ini bertujuan untuk mengetahui apakah rangkaian LCD dapat bekerja dengan baik. Pengujian dilakukan dengan menggunakan program Menampilkan

Karakter pada LCD yang dibuat sebagai salah satu contoh program dan menginterpolasi potensiometer yang digunakan sebagai pengatur kontras LCD. Diagram blok pengujian rangkaian LCD dapat dilihat dalam Gambar 5.4.



Gambar 5.4 Pengujian Rangkaian LCD

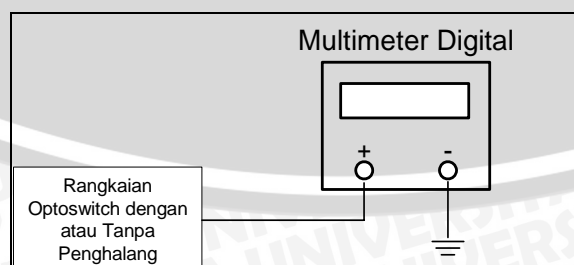
Hasil pengujian rangkaian LCD menunjukkan bahwa LCD dapat menampilkan karakter yang diberikan pada LCD seperti terlihat dalam Gambar 5.6 Sedangkan hasil pengujian rangkaian kontras LCD dapat dilihat dalam Tabel 5.3. Dari tabel tersebut didapatkan bahwa tampilan LCD dengan tingkat kejelasan karakter yang baik saat resistansi potensiometer bernilai 3,88 k Ω sedangkan dari Gambar 5.6 dapat dilihat bahwa rangkaian LCD dapat bekerja dengan baik.

Tabel 5.3 Pengujian Rangkaian Kontras LCD

Resistansi (k Ω)	Tampilan LCD
0,027	Tidak ada karakter yang terlihat
3,88	Karakter terlihat jelas
9,8	Karakter tertutup oleh semua pixel LCD

5.5 Pengujian Rangkaian *Optoswitch*

Pengujian rangkaian *Optoswitch* bertujuan untuk mengetahui tegangan keluaran IC 74LS14 ketika cahaya LED *Optoswitch* terhalangi dan tidak terhalangi oleh kipas. Pengujian dilakukan dengan memberikan catu daya pada rangkaian *Optoswitch* dan mengukur tegangan pada pin keluaran IC 74LS14 menggunakan multimeter selanjutnya menghalangi cahaya LED agar tidak menuju phototransistor dan mengukur tegangan pada pin keluaran IC74LS14. Diagram blok pengujian dapat dilihat dalam Gambar 5.5.



Gambar 5.5 Pengujian Rangkaian *Optoswitch*

Hasil pengukuran tegangan dapat dilihat dalam Tabel 5.4. Dari hasil pengukuran didapatkan bahwa ketika cahaya LED tidak dihalangi maka tegangan pada pin keluaran

IC 74LS14 berlogika 1 dengan tegangan sebesar 4,40 V dan ketika cahaya LED tegangan pada pin keluaran IC 74LS14 berlogika 0 dengan tegangan sebesar 0,12 V dengan demikian keluaran dari rangkaian *Optoswitch* dapat digunakan sebagai masukan oleh mikrokontroler ATMEGA8535 yang memiliki batas minimum tegangan sebesar $0,6 \times V_{CC}$ (3,168V untuk $V_{CC}=5,28V$) untuk logika 1 dan batas maksimum tegangan sebesar $0,2 \times V_{CC}$ (1,056V untuk $V_{CC}=5,28V$) untuk logika 0.

Tabel 5.4 Hasil pengujian keluaran rangkaian *Optoswitch*

Cahaya LED	Tegangan Keluaran (V)	Logika
Tidak dihalangi	4,40	1
Terhalangi	0,12	0

5.6 Pengujian Program LED Menyala Berjalan

Pengujian bertujuan untuk mengetahui kinerja program LED Menyala Berjalan apakah sudah berjalan sesuai dengan perancangan program. Pengujian dilakukan dengan cara mensimulasikan program menggunakan software AVR Studio 4 dan mengimplementasikan program pada perangkat keras yang sudah dibuat dan diuji sebelumnya. Pada pengujian dengan menggunakan perangkat keras, Port A mikrokontroler dihubungkan rangkaian LED dan nyala LED diamati pergerakannya.

Pada proses simulasi menggunakan software AVR Studio 4 menunjukkan bahwa program bekerja sesuai dengan yang diharapkan yaitu pada Port A logika 1 bergerak dari Port A7 menuju Port A0 dan kembali ke Port A7 menuju Port A0 seterusnya. Hasil pengujian menggunakan perangkat keras didapati LED pada Port A menyala bergantian secara berurutan dari kiri ke kanan dan berulang secara terus menerus, hal ini menunjukkan bahwa program sudah berjalan sesuai dengan yang diharapkan.

5.7 Pengujian Program Menampilkan Karakter pada LCD

Pengujian ini bertujuan untuk mengetahui kinerja program Menampilkan Karakter pada LCD apakah sudah berjalan sesuai dengan perancangan program. Pengujian dilakukan dengan cara mensimulasikan program menggunakan software AVR Studio 4 dan mengimplementasikan program pada perangkat keras yang sudah dibuat dan diuji sebelumnya. Pada pengujian dengan menggunakan perangkat keras, Port C dan D dihubungkan dengan rangkaian LCD dan hasil yang terjadi diambil gambarnya dengan menggunakan kamera.

Pada proses simulasi menggunakan software AVR Studio 4, program berjalan secara berurutan sesuai dengan algoritma yang dibuat dimana setiap data yang

dikirimkan pada LCD dikirim melalui Port C sedangkan sinyal Enable dan RS LCD diatur oleh Port D0 dan D1. Hasil pengujian menggunakan perangkat keras dapat dilihat bahwa LCD dapat menampilkan karakter sesuai dengan yang diharapkan seperti terlihat dalam Gambar 5.6 dari hasil pengujian dapat disimpulkan bahwa program Menampilkan Karakter pada LCD dapat bekerja dengan baik.



Gambar 5.6 Pengujian Program Menampilkan Karakter pada LCD

5.8 Pengujian Rangkaian Keypad dan Program Memasukkan Karakter dari Keypad dan Menampilkannya pada LCD

Pengujian ini bertujuan untuk mengetahui apakah rangkaian keypad dan program Memasukkan Karakter dari Keypad dan Menampilkannya pada LCD sudah bekerja sesuai dengan perancangan. Pengujian program dilakukan dengan cara mensimulasikan program menggunakan software AVR Studio 4 dan mengimplementasikan program pada perangkat keras. Pada pengujian dengan menggunakan perangkat keras, Port C dan D dihubungkan dengan rangkaian LCD dan keypad dihubungkan dengan Port A mikrokontroler. Setiap tombol yang ada pada keypad ditekan secara bergantian dan setiap karakter yang tampil pada LCD ditabelkan.

Pada proses simulasi software AVR Studio 4 didapatkan program bekerja sesuai dengan algoritma yang dibuat dimana setiap data keypad yang masuk pada Port A digunakan untuk memilih karakter yang ditampilkan pada LCD dan setiap data yang diberikan pada LCD dikirim melalui Port C dan Port D. Hasil pengujian menggunakan perangkat keras dapat dilihat dalam Tabel 5.5. Dari tabel tersebut dapat dilihat bahwa karakter yang ditampilkan pada LCD selalu sesuai dengan yang ditekan pada keypad. Hasil pengujian menunjukkan bahwa program berjalan sesuai dengan yang diharapkan dan rangkaian keypad dapat bekerja dengan baik.

Tabel 5.5 Hasil pengujian program memasukkan karakter dari keypad dan menampilkannya pada LCD

Tombol yang ditekan	Tampilan Karakter pada LCD
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
0	0
COR	COR
MEN	MEN
▲	Up
▼	Down
CAN	CAN
ENT	ENT

5.9 Pengujian Program Menggerakkan Motor Steper

Pengujian ini bertujuan untuk mengetahui kinerja program Menggerakkan Motor Steper apakah sudah berjalan sesuai dengan perancangan program. Pengujian dilakukan dengan cara mensimulasikan program menggunakan software AVR Studio 4 dan mengimplementasikan program pada perangkat keras yang sudah dibuat dan diuji sebelumnya. Pada pengujian dengan menggunakan perangkat keras, Port B mikrokontroler dihubungkan rangkaian pengendali motor stepper, nyala LED pada Port B diamati pergerakannya dan Motor Steper juga diamati pergerakannya.

Pada proses simulasi menggunakan software AVR Studio 4 dapat dilihat bahwa logika 1 pada Port B bergerak dari Port B.3 menuju Port B0 dan kembali ke Port B3 menuju Port B0 seterusnya dengan jeda yang konstan pada setiap pergerakan logika. Pada pengujian menggunakan perangkat keras pergerakkan logika 1 pada Port B dapat diamati melalui LED yang terhubung pada Port B, terlihat bahwa nyal LED bergerak dari Port B3 menuju Port B0 dan kembali ke Port B3 secara terus menerus. Motor Steper berputar secara terus-menerus searah jarum jam dengan kecepatan yang konstan. Dari hasil pengujian dapat disimpulkan bahwa program sudah berjalan sesuai dengan yang diharapkan.

5.10 Pengujian Program Menampilkan Nilai Tegangan DC pada LCD

Pengujian ini bertujuan untuk mengetahui kinerja program Menampilkan Nilai Tegangan DC pada LCD apakah sudah berjalan sesuai dengan perancangan program. Pengujian dilakukan dengan cara mensimulasikan program menggunakan software

AVR Studio 4 dan mengimplementasikan program pada perangkat keras yang sudah dibuat dan diuji sebelumnya. Pada pengujian dengan menggunakan perangkat keras, Port C dan Port D mikrokontroler dihubungkan rangkaian LCD sedangkan masukan tegangan analog yang akan ditampilkan pada LCD diberikan pada Port A1. Data hasil konversi juga dapat diamati melalui Port B.

Pada proses simulasi menggunakan software AVR Studio 4 data hasil konversi diberikan secara langsung pada registernya dan interrupt ADC diaktifkan secara manual, dapat dilihat bahwa proses pengolahan data berjalan sesuai dengan algoritma yang dibuat dan data yang dikirimkan pada LCD sudah benar. Hasil pengujian menggunakan perangkat keras dapat dilihat dalam Tabel 5.6. Dari tabel tersebut dapat dilihat bahwa terjadi perbedaan antara tegangan pada masukan ADC (V_{IN}) dengan tegangan yang ditampilkan pada LCD (V_{LCD}), perbedaan terbesar adalah 0,03 V hal ini dikarenakan akurasi dari ADC internal mikrokontroler ATMEGA8535 memiliki tingkat kesalahan sebesar ± 2 LSB yang mana dengan Resolusi ADC sebesar 0,02 V maka 2 LSB senilai dengan 0,04 V. Pada Port B dapat dilihat nilai biner dari tegangan analog yang diberikan. Dari hasil simulasi dan pengujian dengan perangkat keras dapat disimpulkan bahwa program dapat bekerja dengan baik.

Tabel 5.6 Hasil pengujian program menampilkan nilai tegangan DC pada LCD

V_{in} (V)	Port B	V_{LCD} (V)	ΔV (V)
1,14	00111001	1,11	0.03
2,34	01110111	2,32	0.02
3,05	10011011	3.02	0.02
4,25	11011010	4,25	0.02
5,00	11111111	4,98	0.02

5.11 Pengujian Program Mengirim/Menerima Data dari/ke PC

Pengujian ini bertujuan untuk mengetahui kinerja program Mengirim/Menerima Data dari/ke PC apakah sudah berjalan sesuai dengan perancangan program dan menguji apakah rangkaian RS232 dapat bekerja sebagaimana mestinya. Pengujian dilakukan dengan cara mensimulasikan program menggunakan software AVR Studio 4 dan mengimplementasikan program pada perangkat keras yang sudah dibuat dan diuji sebelumnya. Pada pengujian dengan menggunakan perangkat keras, Port UART mikrokontroler dihubungkan dengan Port Serial PC dan menggunakan bantuan software Code Vision AVR untuk melakukan proses pengiriman dan penerimaan data. Data yang dikirim melalui PC merupakan data acak dan data yang diterima oleh mikrokontroler

dapat dilihat melalui LED pada Port C sedangkan data yang diterima oleh PC dapat dilihat melalui tampilan pada Software Code Vision AVR.

Pada proses simulasi menggunakan software AVR Studio 4, Interrupt Receive mikrokontroler diaktifkan secara manual dan UDR(UART Data register) diisi dengan data terlihat bahwa fungsi interrupt receive dieksekusi dan data ditampilkan pada Port C selanjutnya dikirim kembali menggunakan fungsi Send_Data ke PC melalui Pin Tx (Pin D1). Hasil pengujian menggunakan perangkat keras dapat dilihat dalam Tabel 5.7. Dari tabel tersebut dapat dilihat bahwa setiap data yang dikirim dari PC diterima dengan benar oleh mikrokontroler dan ditampilkan melalui LED yang terhubung pada Port C dan setiap data yang sudah diterima oleh mikrokontroler dikirim kembali ke PC dan terlihat bahwa data yang diterima oleh PC sama dengan data yang dikirim sebelumnya. Dari hasil pengujian yang dilakukan baik menggunakan simulasi maupun implementasi pada perangkat keras dapat disimpulkan bahwa program sudah berjalan sesuai dengan yang diharapkan tanpa ada kesalahan data dan rangkaian RS232 dapat bekerja dengan baik.

Tabel 5.7 Hasil Pengujian Program Mengirim/Menerima Data dari/ke PC

Data dari PC (Heksa)	Logika pada Port C	Data diterima PC (Heksa)
01	00000001	01
30	00110000	30
4A	01001010	4A
FA	11111010	FA
5C	01011100	5C
FF	11111111	FF
7E	01111110	7E
45	01000101	45
20	00100000	20
81	10000001	81

5.12 Pengujian Program Menyalakan LED Setiap 5 Detik

Pengujian ini bertujuan untuk mengetahui kinerja Program Menyalakan LED Setiap 5 Detik. Pengujian dilakukan dengan menggunakan program uji seperti pada Lampiran yang memiliki siklus penyalan LED 10 kali lebih lama dari program sebenarnya, hal ini bertujuan untuk memudahkan pengukuran waktu dan mendapatkan ketelitian pengukuran yang mendekati kondisi sebenarnya. Pengujian dilakukan dengan cara mensimulasikan program menggunakan software AVR Studio 4 dan mengimplementasikan program pada perangkat keras yang sudah dibuat dan diuji sebelumnya.

Hasil pengujian menggunakan software AVR Studio 4, interrupt timer terjadi setiap 208,13 μ s. Hal ini menunjukkan bahwa terjadi penyimpangan sebesar 8,13 μ s dari perhitungan yang diinginkan yaitu 200 μ s. Hal ini dikarenakan pada proses pengeksekusian fungsi Interrupt Timer menimbulkan jeda waktu sebanyak perintah-perintah yang ada pada fungsi Interrupt tersebut. Hasil pengujian menggunakan perangkat keras dapat dilihat dalam Tabel 5.8. Dari tabel tersebut dapat dilihat bahwa selang waktu menyalanya LED sebesar 51 detik terjadi selisih sebesar 1 detik dari kondisi yang diinginkan, hal ini terjadi berkaitan dengan hasil simulasi menggunakan software AVR Studio 4 yang menunjukkan adanya selisih waktu sebesar 208,13 μ s dari nilai yang diinginkan sehingga menimbulkan perbedaan perhitungan waktu sebesar 1 detik. Secara keseluruhan dari hasil simulasi dan pengujian menggunakan perangkat keras program dapat bekerja dengan baik.

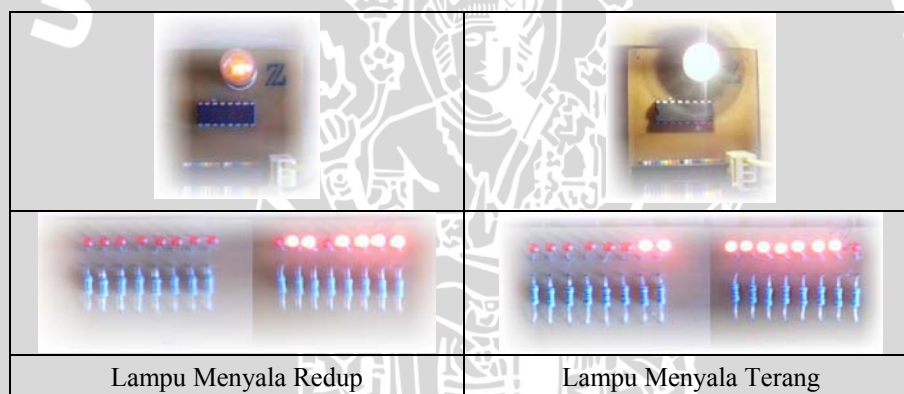
Tabel 5.8 Hasil Pengujian Program Menyalakan LED Setiap 50 Detik

Waktu LED Menyala	$T_n - T_{n-1}$ (s)
0:51	-
1:42	51
2:33	51
3:24	51
4:15	51
5:06	51
5:57	51
6:48	51
7:39	51
8:30	51

5.13 Pengujian Program Aplikasi *Dimmer* pada Lampu Pijar

Pengujian ini bertujuan untuk mengetahui kinerja program Aplikasi *Dimmer* pada Lampu Pijar apakah sudah berjalan sesuai dengan perancangan program. Pengujian dilakukan dengan cara mensimulasikan program menggunakan software AVR Studio 4 dan mengimplementasikan program pada perangkat keras yang sudah dibuat dan diuji sebelumnya. Pada pengujian dengan menggunakan perangkat keras Port D mikrokontroler dihubungkan rangkaian pengendali lampu pijar sedangkan keypad dihubungkan dengan Port A mikrokontroler selanjutnya tombol Down ditekan secara terus menerus untuk menurunkan tegangan efektif PWM dan diikuti dengan menekan tombol Up untuk menaikkan tegangan efektif PWM. Perubahan nyala lampu pijar akan diamati.

Pada proses simulasi menggunakan software AVR Studio 4 Port D.5 selalu berlogika 1 saat nilai dari register TCNT1 sama dengan OCR1A dan setelah TCNT1 mencapai nilai puncak Port D.5 berlogika 0 register TCNT1 mengulang pencacahan kembali dari 0 ke nilai maksimum yang sudah ditetapkan. Ketika data Keypad yang diberikan 0x0E (Tombol Up) nilai OCR1A dinaikkan pula sedangkan ketika Data Keypad yang diberikan 0x0F diamati secara langsung dan terlihat ketika tombol Up pada keypad ditekan lampu pijar menyala semakin terang sedangkan ketika tombol Down pada keypad ditekan lampu pijar menyala semakin redup seperti yang dapat dilihat dalam Gambar 5.7. Pada gambar tersebut dapat dilihat bahwa saat lampu redup nilai OCR1A dalam kisaran rendah yaitu 0x006F sedangkan saat lampu terang nilai OCR1A sangat tinggi yaitu 0x03FE. Dari hasil pengujian menggunakan simulasi maupun dengan menggunakan perangkat keras dapat disimpulkan bahwa program dapat bekerja dengan baik sesuai dengan perancangan.



Gambar 5. 7 Hasil Pengujian Program Aplikasi Dimmer Pada Lampu Pijar

5.14 Pengujian Program Membuat Loop Tak Berhingga Sebagai Simulasi Kondisi

Hang

Pengujian ini bertujuan untuk mengetahui kinerja program Membuat Loop Tak Berhingga Sebagai Simulasi Kondisi *Hang* apakah sudah berjalan sesuai dengan perancangan program. Pengujian dilakukan dengan cara mensimulasikan program menggunakan software AVR Studio 4 dan mengimplementasikan program pada perangkat keras yang sudah dibuat dan diuji sebelumnya. Pada pengujian dengan menggunakan perangkat keras pengamatan dilakukan dengan melihat proses kejadian pada LED yang terhubung pada Port tersebut.

Pada proses simulasi menggunakan software AVR Studio 4 dapat dilihat bahwa logika 1 pada Port A bergerak secara berulang sebanyak 10 kali dari Port D7 menuju Port D0 dan dilanjutkan dengan Port A yang bernilai 0xFF. Program mengeksekusi loop abadi dan setelah 2,1 detik program dieksekusi kembali dari awal. Proses pengujian menggunakan perangkat keras dapat dilihat dalam Tabel 5.9. Dari tabel tersebut didapati bahwa urutan proses berjalannya program menunjukkan bahwa fasilitas Watchdog dieksekusi setelah LED menyala semua sehingga setelah LED menyala semua LED kembali padam seperti kondisi awal program dijalankan. Dari hasil pengujian menggunakan simulasi dan implementasi pada perangkat keras dapat disimpulkan bahwa program dapat bekerja dengan baik.

Tabel 5.9 Hasil Pengujian Program Membuat Loop Tak Berhingga Sebagai Simulasi Kondisi Hang

No	Tampilan LED pada Port A	Keterangan
1	LED padam sekitar 1 detik	Program mulai berjalan
2	Menyala bergerak dari kiri ke kanan	Runing LED dijalankan
3	Menyala Semua	Runing LED selesai dijalankan
4	LED padam sekitar 1 detik	Program kembali dieksekusi dari awal

5.15 Pengujian Program Membuat LED Menyala Jika Suatu Tegangan Melebihi Tegangan Referensi

Pengujian ini bertujuan untuk mengetahui kinerja program Membuat LED Menyala Jika Suatu Tegangan Melebihi Tegangan Referensi apakah sudah berjalan sesuai dengan perancangan program. Pengujian dilakukan dengan cara mensimulasikan program menggunakan software AVR Studio 4 dan mengimplementasikan program pada perangkat keras yang sudah dibuat dan diuji sebelumnya. Pada saat simulasi perbedaan tegangan antara Pin AIN0 dan AIN1 diberikan dengan cara mengaktifkan Interrupt Analog Comparator secara langsung. Pada pengujian dengan menggunakan perangkat keras dilakukan pengambilan beberapa sampel perbedaan tegangan yang diberikan pada Pin AIN0 dan AIN1 dan pengamatan dilakukan pada nyala LED yang terhubung dengan Port C.

Pada proses simulasi menggunakan software AVR Studio 4 saat Interrupt Analog Comparator flag diberi logika 1 maka fungsi interrupt Analog Comparator dieksekusi dan Port C bernilai 0xFF dan saat Interrupt Analog Comparator flag kembali diberi logika 0 maka port C bernilai 0x00. Hasil pengujian menggunakan perangkat keras dapat dilihat dalam Tabel 5.10. Dari tabel tersebut dapat dilihat bahwa ketika tegangan masukan pada Pin positif Analog Comparator (V_{AIN0}) lebih besar dari tegangan

masukan pada Pin negatif (V_{AIN1}) maka LED pada Port C akan menyala dan jika kondisi sebaliknya yang terjadi maka LED pada Port C akan padam. Hasil simulasi dan pengujian ini menunjukkan bahwa program dapat bekerja sesuai dengan perancangan.

Tabel 5.10 Hasil Pengujian Program Membuat LED Menyala Jika Suatu Tegangan Melebihi Tegangan Referensi

V_{AIN0} (V)	V_{AIN1} (V)	LED pada Port C
1,14	2,52	Padam
2,24	1,64	Menyala
2,98	3,28	Padam
5,00	3,18	Menyala

5.16 Pengujian Program Menyimpan Data ke EEPROM dan Menampilkan Data dari EEPROM pada LCD

Pengujian ini bertujuan untuk mengetahui kinerja program Menyimpan Data ke EEPROM dan Menampilkan Data dari EEPROM pada LCD apakah sudah berjalan sesuai dengan perancangan program. Pengujian dilakukan dengan cara mensimulasikan program menggunakan software AVR Studio 4 dan mengimplementasikan program pada perangkat keras yang sudah dibuat dan diuji sebelumnya. Pada pengujian dengan menggunakan perangkat keras, data yang dimasukkan berupa angka-angka secara acak sebanyak 10 data kemudian data disimpan untuk dibaca kembali setelah catu daya dimatikan.

Pada proses simulasi menggunakan software AVR Studio 4 data dari keypad diberikan dengan memberikan logika 0 pada pin-pin yang ingin digunakan sebagai masukan dari keypad. Data yang diperoleh dari keypad disimpan sementara dalam larik-larik dan ketika data untuk menyimpan data pada EEPROM diberikan semua data dari larik tersebut disimpan pada EEPROM dengan alamat yang berurutan sedangkan ketika data untuk membaca EEPROM diberikan program mengakses EEPROM untuk mendapatkan data yang sudah disimpan. Hasil pengujian dengan menggunakan perangkat keras dapat dilihat dalam Tabel 5.11 Dari tabel tersebut dapat dilihat bahwa semua data yang dimasukkan dapat dibaca kembali dari EEPROM dengan tepat tanpa ada kesalahan. Dari hasil simulasi dan pengujian menggunakan perangkat keras dapat disimpulkan bahwa program ini dapat bekerja dengan baik.

Tabel 5.11 Hasil Pengujian Program Menyimpan Data ke EEPROM dan Menampilkan Data yang tersimpan di EEPROM pada LCD

Masukan Angka	Data hasil pembacaan EEPROM
123456	123456
777	777
051201	051201
241466	241466
888	888
170845	170845
9454	9454
7805	7805
100	100
730	730

5.17 Pengujian Program Simulasi Pengaturan Suhu Ruangan

Pengujian ini bertujuan untuk mengetahui kinerja program Simulasi Pengaturan Suhu Ruangan apakah sudah berjalan sesuai dengan perancangan program. Pengujian dilakukan dengan cara mensimulasikan program menggunakan software AVR Studio 4 dan mengimplementasikan program pada perangkat keras yang sudah dibuat dan diuji sebelumnya. Pada proses simulasi data ADC diberikan secara langsung melalui registernya dan interrupt ADC maupun eksternal diberikan secara manual dengan memberikan logika 1 pada flagnya. Pada pengujian dengan menggunakan perangkat keras, tegangan yang diberikan pada ADC merupakan tegangan yang ada pada range-range yang menghasilkan kecepatan putar motor stepper yang berbeda-beda.

Hasil proses simulasi menggunakan software AVR Studio 4 menunjukkan data yang diberikan pada register ADC diolah sesuai dengan algoritma yang dibuat. Hasil pengujian menggunakan perangkat keras dapat dilihat dalam Tabel 5.12. Dari tabel tersebut dapat dilihat bahwa tegangan yang tampil pada LCD (V_{LCD}) memiliki selisih 0,02 V dengan tegangan masukan (V_{IN}) yang disebabkan oleh akurasi dari ADC internal mikrokontroler ATMEGA8535 ± 2 LSB. Putaran dari motor stepper sesuai dengan batas tegangan yang ditentukan sebelumnya, untuk tegangan 0 – 1,5 V motor stepper tidak berputar, 2 V – 2,5 V putaran motor stepper 25 rpm, untuk tegangan 3 V – 4 V putaran motor stepper 50 rpm dan untuk tegangan 4,5 V atau putaran motor stepper 100 rpm. Saat putaran motor stepper mencapai 100 rpm LED pada Port D4 dan D5 menyala. Dari hasil simulasi dan pengujian dapat disimpulkan bahwa program dapat bekerja dengan baik.

Tabel 5.12 Hasil Pengujian Program Simulasi Pengaturan Suhu Ruangan

V_{in} (V)	V_{LCD} (V)	RPM Kipas	LED Port D4 dan D5
1,12	1,14	0	Padam
2,20	2,18	25	Padam
3,30	3,28	50	Padam
5,00	4,98	100	Menyala



BAB VI PENUTUP

4.3 Kesimpulan

Berdasarkan perancangan, pembuatan, pengujian dan analisis hasil pengujian pada setiap blok rangkaian dan perangkat lunak, secara keseluruhan dapat disimpulkan sebagai berikut:

- a. Setiap rangkaian yang telah dibuat dan diuji dapat berjalan dengan baik dan sesuai dengan yang diharapkan melalui perancangan perangkat keras.
- b. Perangkat lunak yang dibuat dapat bekerja dengan baik sesuai dengan yang diharapkan dalam perancangan perangkat lunak.
- c. Perangkat keras dan perangkat lunak yang sudah dibuat dapat digunakan untuk belajar mikrokontroler ATMEGA8535 karena dapat menampilkan fitur-fitur penting yang dimiliki oleh mikrokontroler ini.
- d. Terjadi perbedaan antara tegangan pada masukan ADC dengan tegangan yang ditampilkan pada LCD, perbedaan terbesar adalah 0,03 V hal ini dikarenakan akurasi dari ADC internal mikrokontroler ATMEGA8535 memiliki tingkat kesalahan sebesar $\pm 0,04 \text{ V } (\pm 2\text{LSB})$
- e. Terjadi selisih waktu penyalan LED dengan hasil perancangan pada proses pengujian Program Menyalakan LED Setiap 5 Detik sebesar 0,1 detik. Hal ini dikarenakan pada proses pengeksekusian fungsi Interrupt Timer menimbulkan jeda waktu sebanyak perintah-perintah yang dieksekusi pada fungsi Interrupt tersebut

4.4 Saran

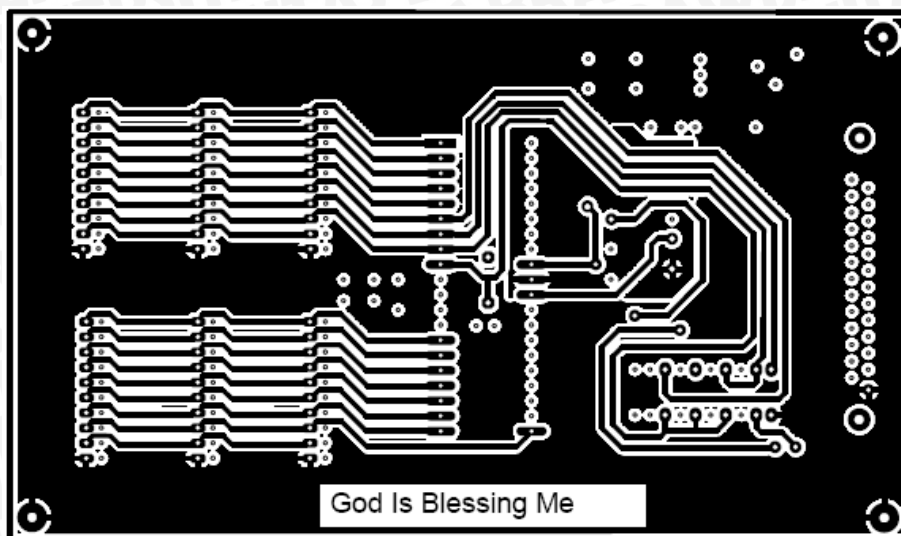
Terjadinya selisih waktu penyalan LED dengan hasil perancangan dapat diminimalkan dengan mengatur kembali nilai awal Timer (TCNT0) yang diberikan sesuai dengan besar selisih waktu tersebut. Penggunaan modul ini akan lebih efektif jika didampingi oleh instruktur.

DAFTAR PUSTAKA

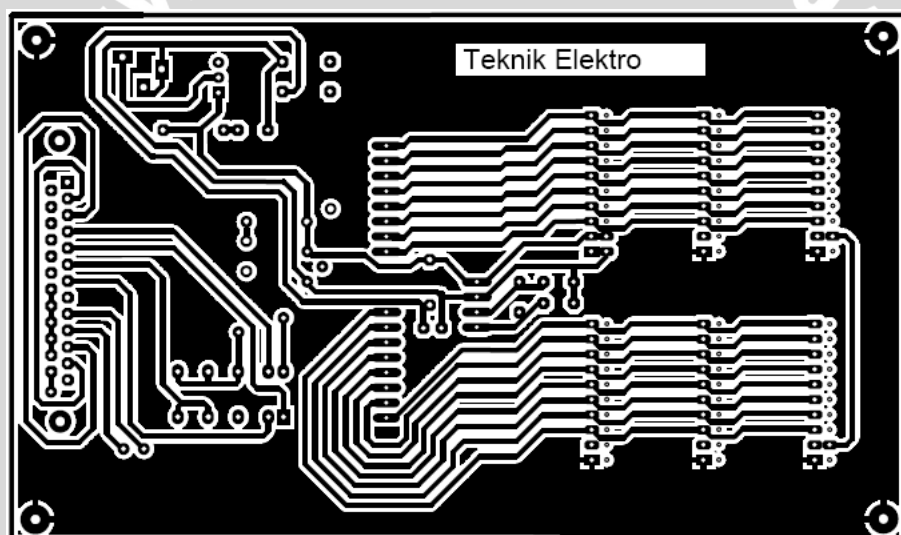
- Malvino, Albert Paul. 1991. *Prinsip-Prinsip Elektronika*. Edisi Ketiga. Jakarta: Erlangga.
- Malvino, Albert Paul. 1987. *Prinsip-Prinsip dan Penerapan Digita*., Edisi Ketiga. Jakarta: Erlangga.
- Atmel, 2001. *8-bit AVR Microcontroller with 8 K Byte In-System Programmable Flash ATMEGA8535*. <http://www.atmel.com>. Diakses tahun 2005
- Kleitz. 1996. *Digital Electronics (A Practical Approach)*. New Jersey: Prentice Hall
- Wirtz Electronic. 1998, *The Stepper Archive - Identifying Common Stepper Motor Types*. www.Wirtz-Electronic.com. Diakses Tahun 2003
- Seiko Instrument Inc. 1987. *Liquid Crystal Display Module M1632 User Manua*
- Peacock, Craig. 2001. *Interfacing the Standard Parallel Port*. www.beyondlogic.org. Diakses Tahun 2003
- Busono. 1992. *Panduan Pembuatan Program dan Rangkuman Mikrokontroler MC68705U3*. Jakarta: Dinastindo
- Widyatmo, Arianto. 1994. *Belajar MIKROPROSESOR – MIKOKONTROLER melalui komputer PC*. Jakarta: Elexmedia Komputindo

LAMPIRAN LAYOUT PCB

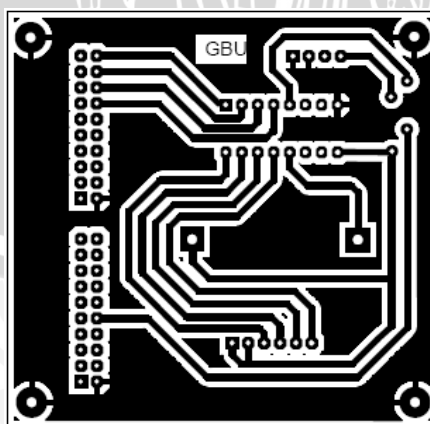




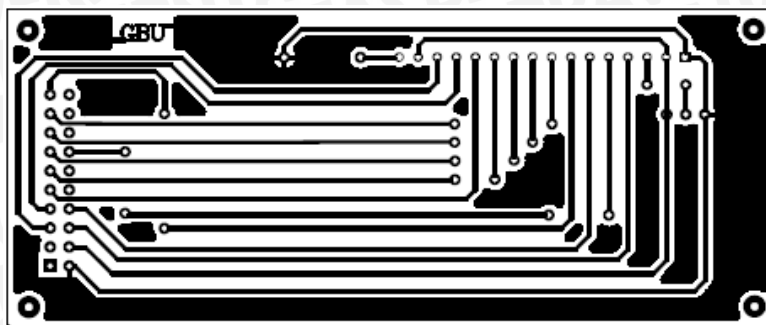
Layout PCB rangkaian pengisi program bagian atas



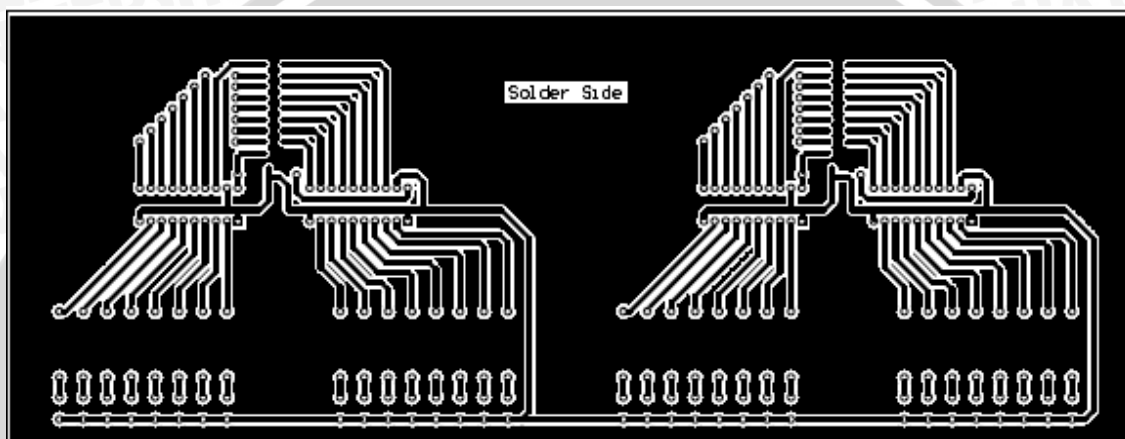
Layout PCB rangkaian pengisi program bagian bawah



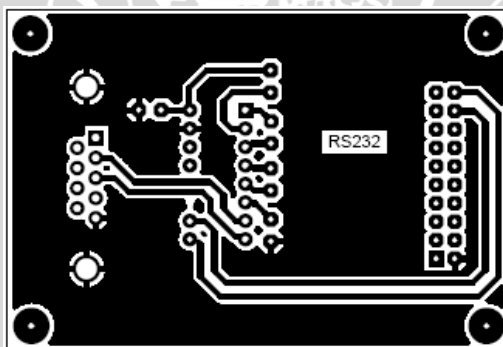
Layout PCB pengendali lampu pijar dan motor stepper



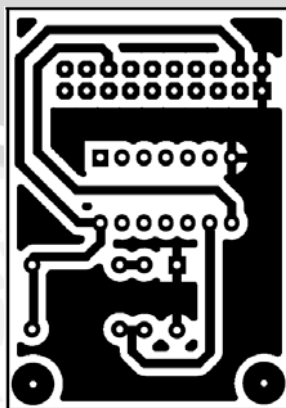
Layout PCB rangkaian LCD



Layout PCB rangkaian pengendali LED



Layout PCB rangkaian RS232



Layout PCB rangkaian *Optoswitch*

LAMPIRAN FOTO ALAT





LAMPIRAN KUISONER



KUISONER

Angkatan :

Konsentrasi :

1. Apakah saudara pernah mengikuti kuliah Mikroprosesor?
 - a. Ya
 - b. Tidak
2. Apakah anda sudah mengikuti praktikum Mikroprosesor?
 - a. Ya
 - b. Tidak
3. Apakah materi kuliah Mikroprosesor sudah menunjang pengenalan anda tentang Mikroprosesor?
 - a. Ya
 - b. Tidak
4. Apakah dalam materi praktikum diadakan percobaan yang benar-benar aplikatif?
 - a. Ya
 - b. Tidak
5. Apakah setelah mengikuti praktikum Mikroprosesor anda dapat menggunakan Mikroprosesor paling tidak untuk hal-hal yang sederhana (contoh: membuat running LED)?
 - a. Ya
 - b. Tidak
6. Apakah anda setuju dengan dibuatnya suatu modul pembelajaran mikrokontroler yang bersifat aplikatif (misal: belajar membuat program untuk menampilkan karakter di LCD, menampilkan suhu ruangan pada LCD, menggerakkan motor dll) beserta peralatannya yang menunjang?
 - a. Ya
 - b. Tidak

REKAPITULASI HASIL SURVEY

Kategori	Jumlah (Mahasiswa)	Prosentase (%)
Pernah mengikuti kuliah Mikroprosesor	40	100
Pernah mengikuti praktikum Mikroprosesor	40	100
Memiliki pendapat bahwa materi kuliah mikroprosesor kurang menunjang pengenalan tentang mikroprosesor	19	47,5
Memiliki pendapat bahwa materi praktikum mikroprosesor kurang aplikatif	35	87,5
Tidak dapat menggunakan mikroprosesor untuk hal-hal yang sederhana sekalipun	25	62,5
Setuju pembuatan modul pembelajaran mikrokontroler yang aplikatif	38	95

LAMPIRAN BUKU MANUAL



**BUKU MANUAL
MODUL PEMBELAJARAN APLIKATIF
MIKROKONTROLER ATMEGA8535**



**TEKNIK ELEKTRO
UNIVERSITAS BRAWIJAYA
2007**



BAB I PENGENALAN DAN OPERASI PANEL MODUL

Modul Pembelajaran Aplikatif ATMEGA8535 digunakan untuk belajar menggunakan mikrokontroler tersebut dengan memanfaatkan perangkat-perangkat keras yang sudah disediakan beserta perangkat lunaknya. Hal-hal yang dapat dipelajari disini adalah fungsi-fungsi dan fitur-fitur dasar yang dimiliki oleh mikrokontroler ATMEGA8535 serta penggunaan mikrokontroler untuk aplikasi yang bersifat kompleks.

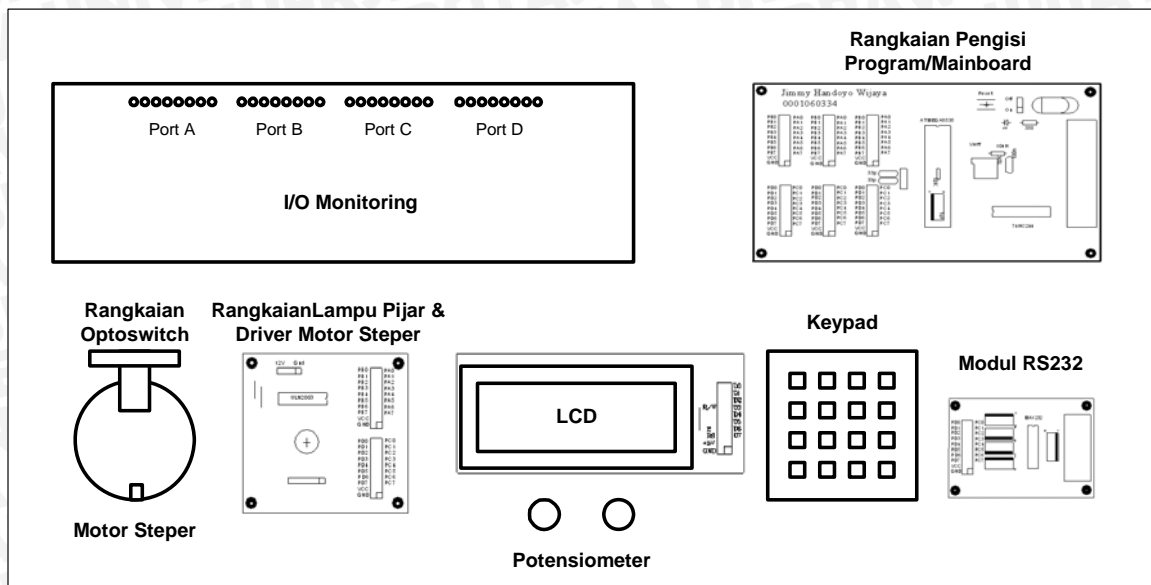
Pengguna modul diharuskan untuk mempelajari teori-teori dasar tentang mikroprosesor maupun mikrokontroler dan bahasa pemrograman C++ untuk lebih mudah memahami penggunaan mikrokontroler dan memahami contoh-contoh perangkat lunak yang diberikan. Pada setiap percobaan, pengguna modul juga diharuskan untuk mempelajari alur contoh program yang diberikan.

Kelengkapan Modul

1. Modul Pengisi Program
2. Modul I/O Monitoring
3. Modul Optoswitch
4. Modul PWM dan Motor Steper
5. Modul LCD
6. Modul Keypad
7. Modul RS232
8. Kabel DB25
9. Kabel Serial DB9
10. Kabel Jumper
11. Power Supply
12. Buku Manual Penggunaan
13. CD Software



Keterangan Panel Modul

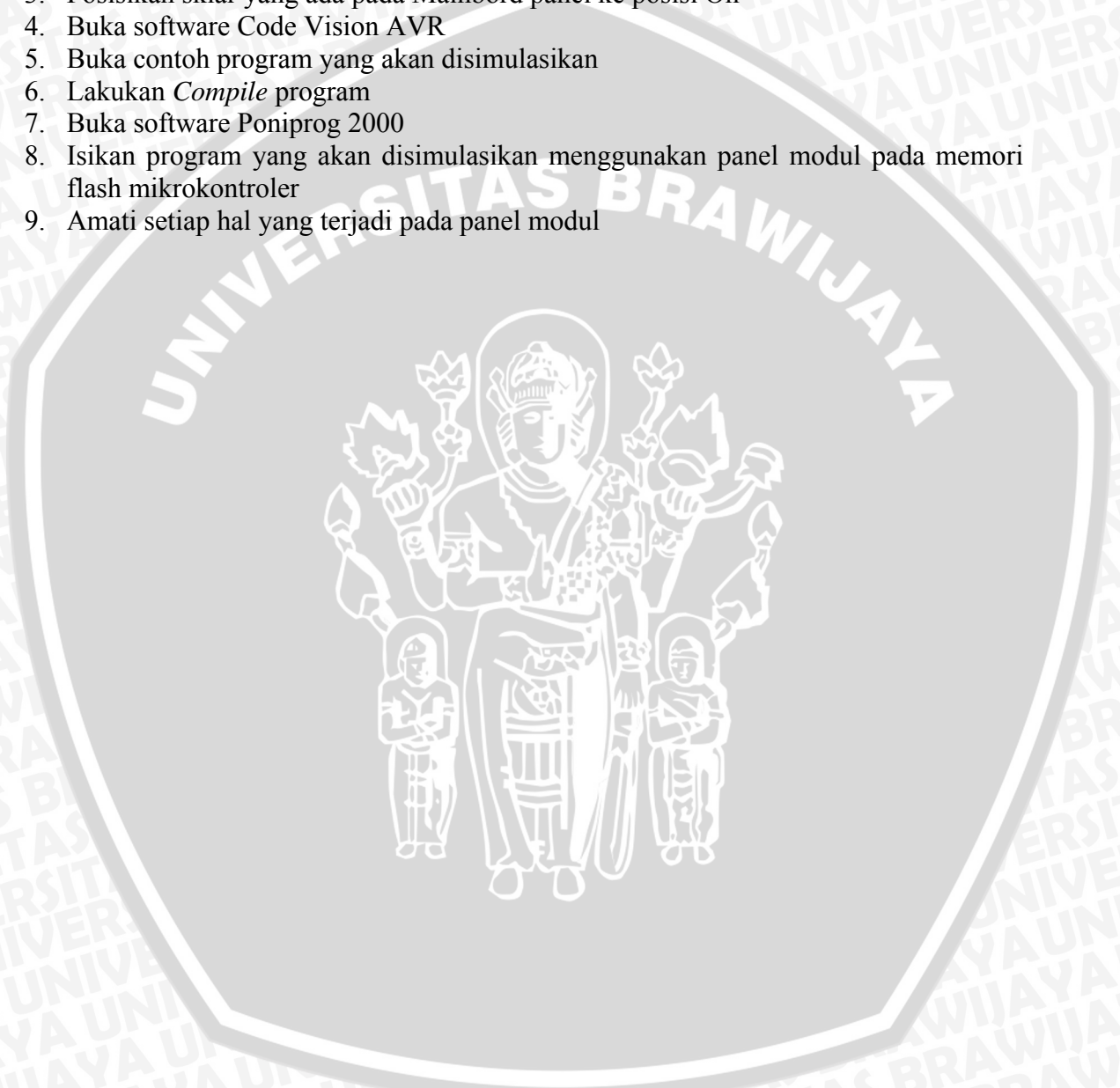


1. Rangkaian Pengisi Program (Mainboard)
Rangkaian Pengisi Program (Mainboard) digunakan untuk mengisikan program yang sudah dibuat ke dalam memory flash mikrokontroler dan digunakan sebagai pusat operasi Modul. Pada rangkaian ini terdapat konektor dari tiap port yang dihubungkan ke seluruh perangkat keras lainnya dalam modul.
2. I/O Monitoring
I/O Monitoring digunakan untuk memantau data yang ada pada setiap port mikrokontroler dan digunakan untuk belajar menggunakan I/O mikrokontroler serta aplikasi dasar mikrokontroler.
3. Modul RS232
Modul RS232 berisikan IC RS232 yang digunakan untuk melakukan komunikasi mikrokontroler dengan komputer.
4. Keypad
Keypad digunakan sebagai masukan pada mikrokontroler
5. LCD
LCD digunakan sebagai media peraga
6. Rangkaian Lampu Pijar dan Driver Motor Steper
Rangkaian Lampu Pijar digunakan sebagai media simulasi fitur PWM yang dimiliki oleh mikrokontroler ATmega8535. Sedangkan rangkaian driver motor stepper digunakan untuk mengendalikan/menggerakkan motor stepper.
7. Rangkaian Optoswitch
Rangkaian Optoswitch digunakan sebagai masukan pada pin Eksternal Interrupt mikrokontroler ATMEGA8535.
8. Potensiometer
Potensiometer yang diberikan sebanyak 2 buah digunakan sebagai interpolator tegangan masukan Internal Analog Comparator dan Internal ADC mikrokontroler ATMEGA8535.

Operasi Panel Modul

Sebelum menggunakan modul semua software yang digunakan harus diinstal pada komputer dan semua contoh program dicopy pada *directory* tertentu yang dikehendaki dalam komputer yang akan digunakan. Untuk mengoperasikan modul langkah-langkah yang dapat diikuti adalah sebagai berikut

1. Masukkan AC Power Cord ke jala-jala listrik 220 V
2. Hubungkan kabel DB25 dari Mainboard panel ke port paralel komputer
3. Posisikan sklar yang ada pada Mainbord panel ke posisi On
4. Buka software Code Vision AVR
5. Buka contoh program yang akan disimulasikan
6. Lakukan *Compile* program
7. Buka software Poniprogram 2000
8. Isikan program yang akan disimulasikan menggunakan panel modul pada memori flash mikrokontroler
9. Amati setiap hal yang terjadi pada panel modul



BAB II PENGENALAN DAN OPERASI SOFTWARE

Software yang digunakan dalam penggunaan modul pembelajaran ini ada 2 buah, masing-masing memiliki fungsi yang berbeda. Adapun software tersebut adalah sebagai berikut:

1. Code Vision AVR
Berfungsi sebagai teks editor program dalam bahasa C dan *compiler* program.
2. Poniproq 2000
Berfungsi sebagai pengisi program ke memori flash mikrokontroler.

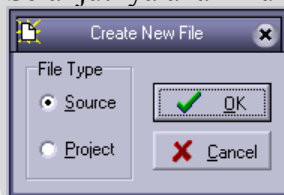
Semua software adalah bersifat free atau gratis dan terdapat dalam CD software yang sudah disertakan. Setiap software harus diinstal terlebih dahulu sebelum digunakan. Khusus untuk software Code Vision AVR, aplikasi yang lebih besar dan kompleks diharuskan untuk membeli *Full Version* software tersebut agar program dapat di-*Compile*.

Code Vision AVR

Untuk membuat program dalam bahasa C menggunakan software Code Vision AVR, langkah-langkah yang harus dilakukan adalah sebagai berikut:

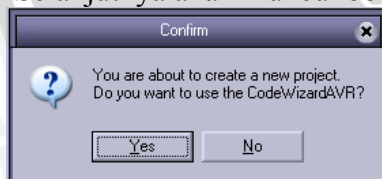
1. Klik ikon software Code Vision AVR  untuk membuka software.
2. Klik **File >> New**

Selanjutnya akan muncul box dialog sebagai berikut

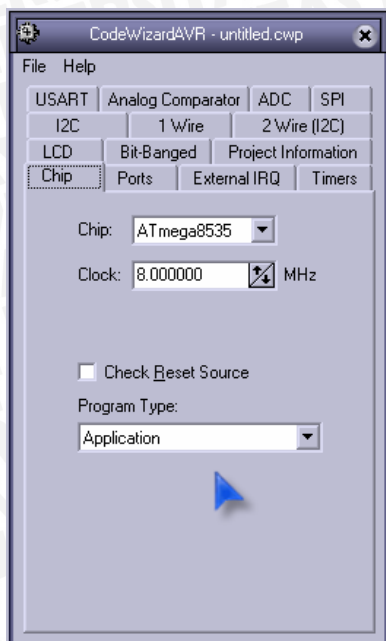


Pilih **Project** dan klik **OK** untuk membuat *project* baru dalam bahasa C, sedangkan pilhan **Source** digunakan untuk membuat teks program dalam bahasa C.

3. Selanjutnya akan muncul box dialog sebagai berikut

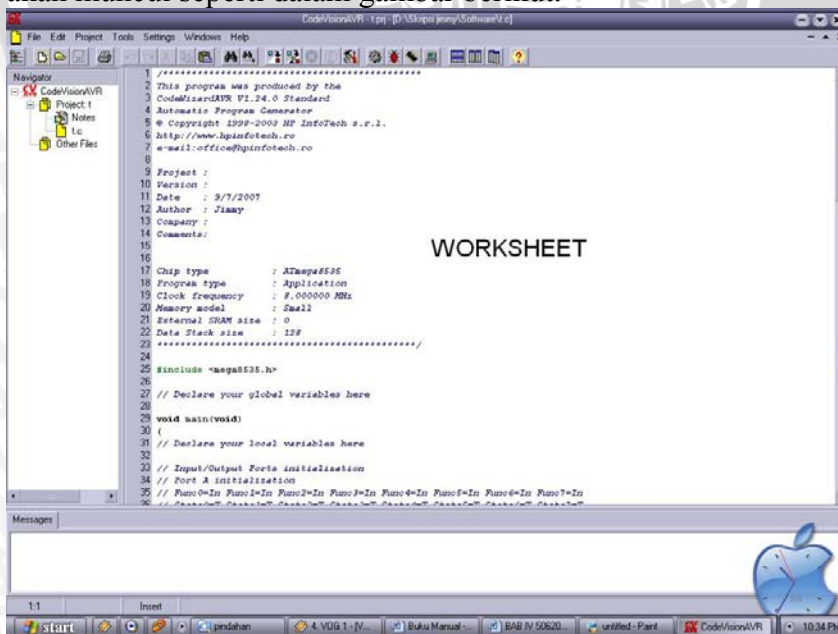


Klik **Yes** jika ingin menggunakan CodeWizardAVR, CodeWizardAVR berguna untuk memunculkan komponen-komponen program mengenai fitur-fitur mikrokontroler secara otomatis dalam program yang akan dibuat. Jika dipilih menggunakan CodeWizardAVR maka akan muncul box dialog sebagai berikut

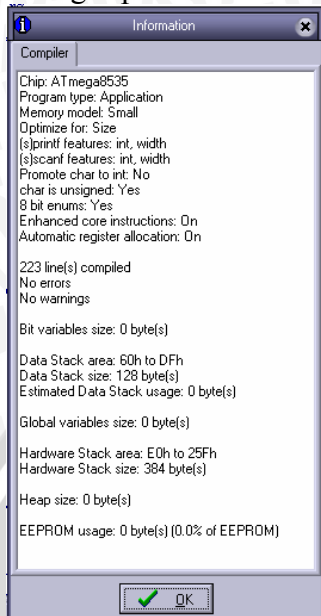


Dapat dilihat dalam box dialog tersebut terdapat panel-panel pilihan yang akan digunakan dalam pembuatan program. Mulai dari jenis mikrokontroler (Chip), nilai Kristal yang dipakai hingga fitur-fitur yang akan digunakan. Untuk memilih jenis mikrokontroler dan fitur-fitur yang digunakan dapat dilakukan dengan cara *sorting* dan memberikan tanda centang (√) pada komponen yang dikehendaki.

4. Setelah semua komponen fitur-fitur yang dikehendaki sudah dipilih maka klik **File>> Generate, Save and Exit**. selanjutnya akan muncul box dialog pengisian nama File sebanyak 3 buah dengan ekstensi sebanyak 3 buah pula mulai dari .c, .prj dan .cwp. Masing-masing file akan dibutuhkan oleh software Code Vision AVR untuk proses *Compile*.
5. Setelah proses *saving* nama file maka program siap diketikan dalam worksheet yang akan muncul seperti dalam gambar berikut.



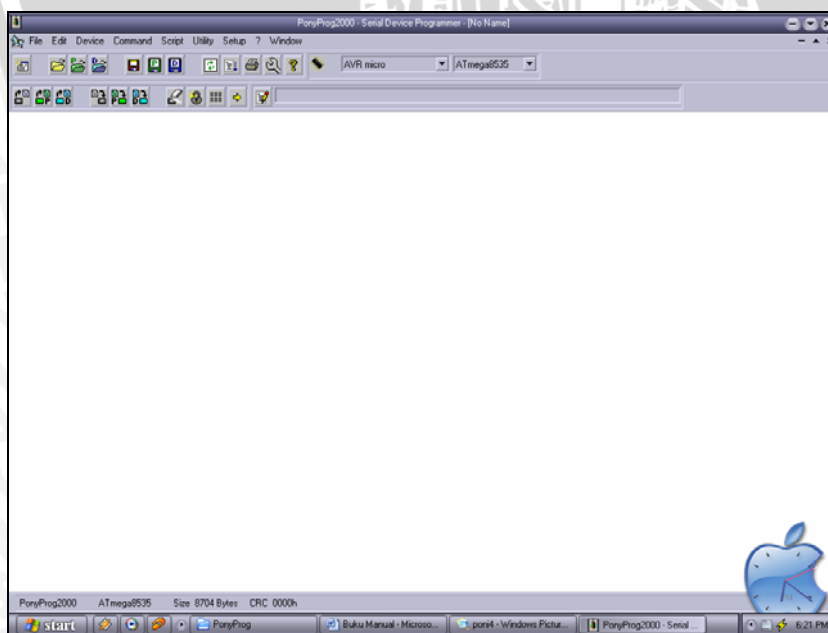
- Setelah program selesai diketikkan maka untuk proses *Compiling* dilakukan dengan melakukan klik pada **Project>> Compile** atau **F9**, tetapi untuk menghasilkan file .hex yang digunakan oleh software Poniprogram 2000 untuk mengisikan program tersebut dalam flash mikrokontroler maka *Compile* dilakukan dengan klik pada **Project>> Make** atau **F10**. Setelah Proses *Compile* atau *Make* maka muncul box dialog seperti di bawah ini



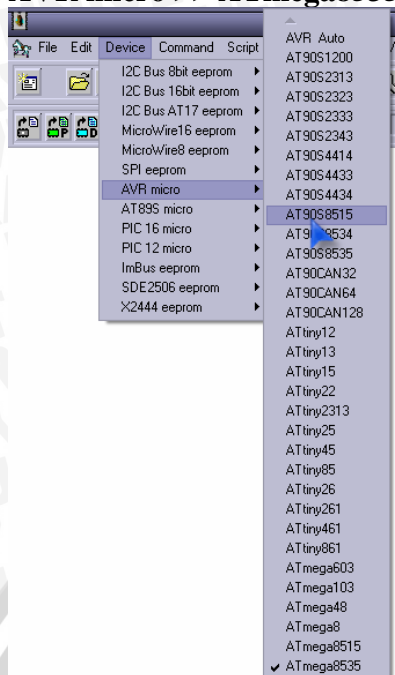
Jika terjadi kesalahan dalam pembuatan program maka pada box dialog tersebut akan diinformasikan.

Poniprogram 2000

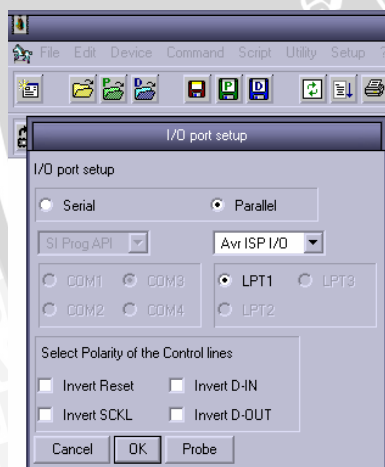
- Klik ikon software Poniprogram 2000  untuk membuka software.



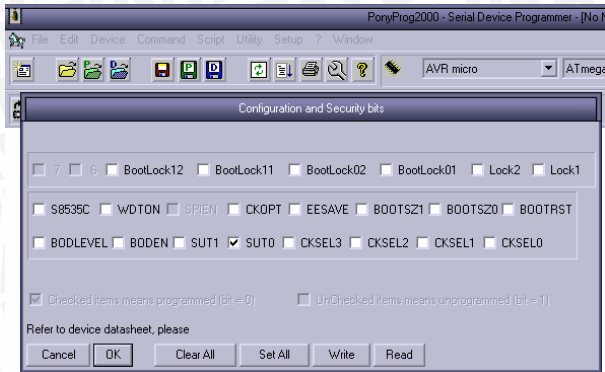
2. Pilih jenis mikrokontroler yang akan digunakan dengan melakukan klik **Device >> AVR micro >> ATmega8535**



3. Untuk memilih Port Paralel sebagai media pemrograman klik **Setup >> Interface Setup**, kemudian pilih Port Paralel dengan jenis Avr ISP I/O seperti pada gambar berikut

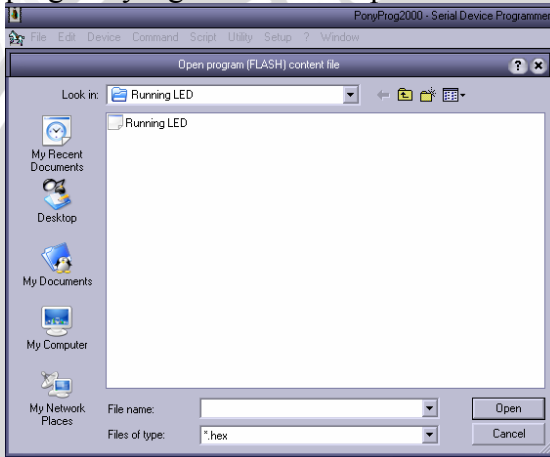


4. Pilih jenis *fuse* mikrokontroler dengan cara klik **Command >> Security and Configuration Bits** dan akan muncul dialog box seperti berikut

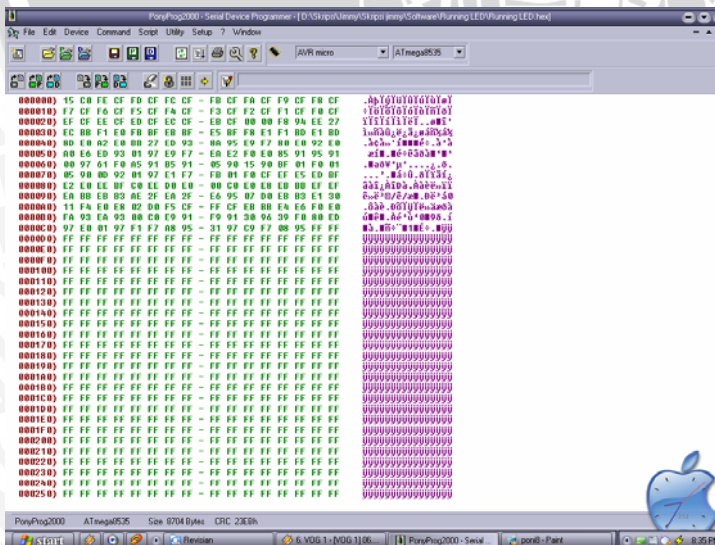


Pilih **SUT0** dengan cara memberikan tanda centang pada boxnya

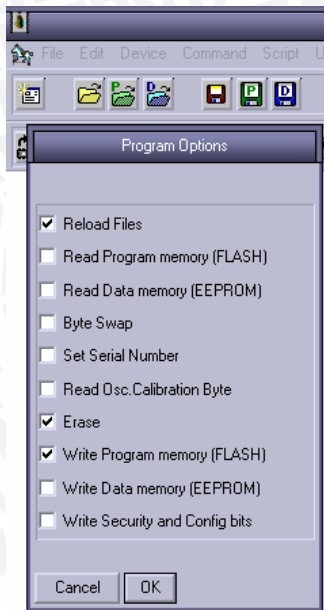
- Untuk membuka file .hex program yang akan diisikan pada flash mikrokontroler klik **File >> Open Program (FLASH) File** akan muncul dialog box Open Program (FLASH) Content File seperti di bawah ini, pilih tipe file .hex kemudian cari file program yang akan diisikan pada flash mikrokontroler.



Program yang dibuka akan muncul pada layar seperti gambar berikut



- Sebelum melakukan pengisian program harus dilakukan pengaturan pemrograman dengan cara klik **Command >> Program Option** maka akan muncul dialog box berikut



Pilih konfigurasi standar seperti pada gambar. **Reload Files** digunakan untuk membuka kembali file .hex program yang sudah diisikan, **Erase** digunakan untuk menghapus isi memori Flash mikrokontroler sebelum diisikan program yang baru sedangkan **Write Data memory (FLASH)** digunakan untuk mengisi memori Flash mikrokontroler dengan data program yang sudah dibuka sebelumnya.

7. Untuk mengisikan program pada memori Flash mikrokontroler klik **Command >> Program** atau **Ctrl P**

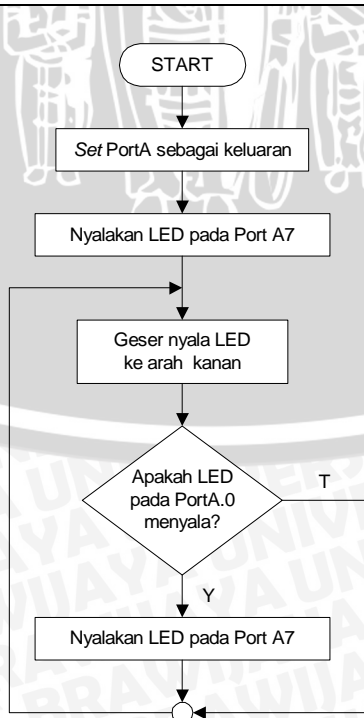
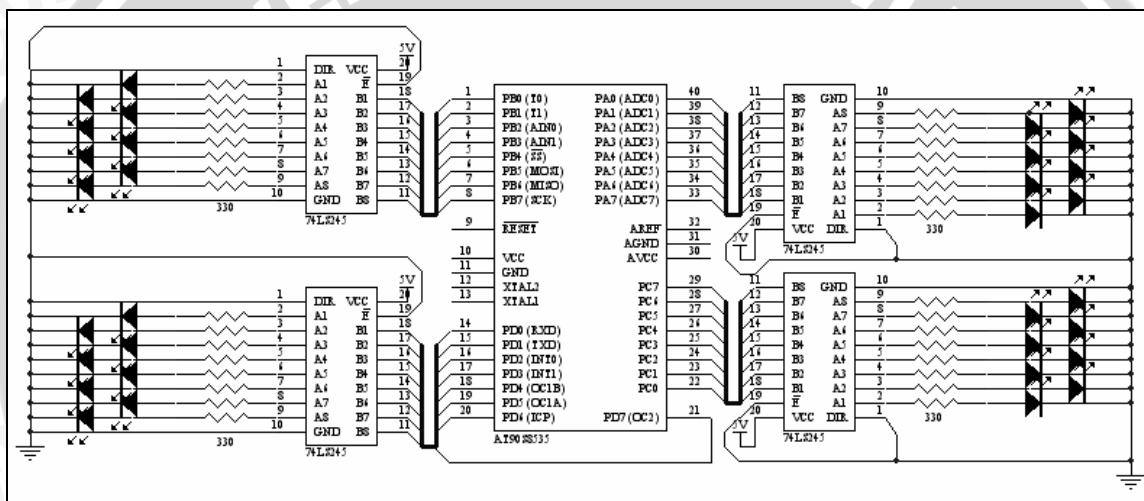


BAB III PROGRAMABLE I/O

Mikrokontroler ATMEGA8535 memiliki 32 I/O yang terbagi menjadi 4 Port, masing-masing I/O tersebut secara individu dapat dikontrol sebagai Input atau Output melalui program yang dibuat. Pada bab ini percobaan menggunakan Programable I/O memanfaatkan fasilitas Port monitoring panel yang berupa LED pada setiap Port, LCD, Keypad dan Motor stepper.

PERCOBAAN LED MENYALA BERJALAN

Pada percobaan ini kita akan membuat LED pada Port A menyala berjalan dari arah kiri ke kanan. Diagram alir program dan Rangkaian antarmuka LED dengan mikrokontroler dapat dilihat dalam gambar berikut

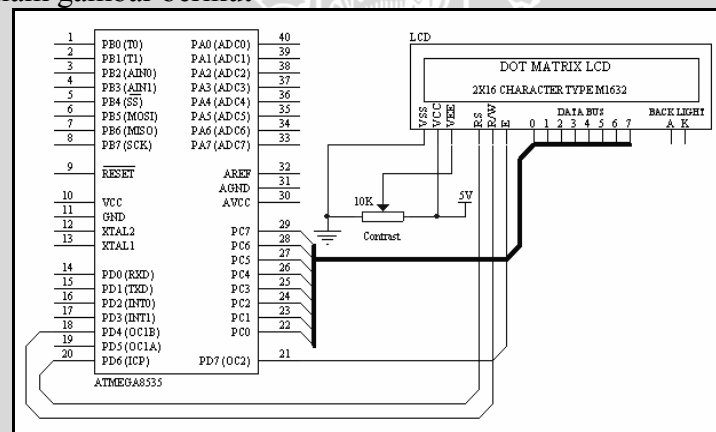


Prosedur Percobaan

1. Buka software Code Vision AVR
2. Buka Program Running LED dari *directory*-nya
3. Lakukan *Make*
4. Masukkan program tersebut pada flash mikrokontroler
5. Perhatikan nyala LED pada Port A
6. Ubah perintah DDRA = 0xFF menjadi 0x00 pada listing program dan lakukan *Make*
7. Masukkan program yang sudah diubah pada flash mikrokontroler
8. Perhatikan nyala LED pada Port A dan catat hal yang terjadi
9. Ubah kembali perintah dari DDRA = 0x00 menjadi 0xFF, *Make* dan masukkan program pada flash mikrokontroler
10. Perhatikan nyala LED pada Port A dan buat kesimpulan dari hasil pengamatan

MENAMPILKAN KARAKTER PADA LCD

Pada percobaan ini kita akan menampilkan karakter yang sudah ditentukan yaitu TEKNIK ELEKTRO dan BRAWIJAYA pada 2 baris yang tersedia pada LCD. Program menggunakan teknik penggabungan file yang dalam hal ini adalah file Rutin_LCD.c. File yang akan digabungkan harus disimpan dalam folder yang sama dengan file induk (main). Diagram alir program dan Rangkaian antarmuka LCD dengan mikrokontroler dapat dilihat dalam gambar berikut



START

Inisialisasi LCD

Tampilkan karakter "TEKNIK ELEKTRO"
pada baris pertama

Tampilkan karakter "BRAWIJAYA"
pada baris kedua

END

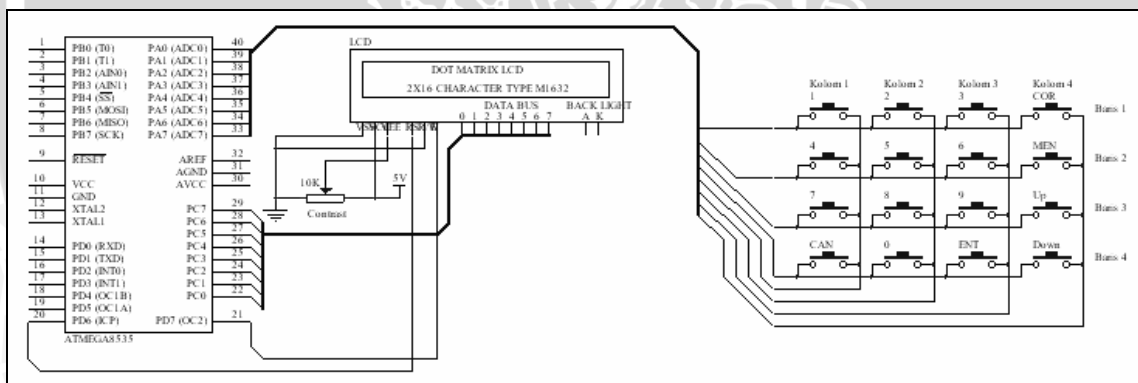
Prosedur Percobaan

1. Buka software Code Vision AVR
2. Buka Program Tampilkan Karakter dari *directory*-nya
3. Lakukan *Make*

4. Masukkan program tersebut pada flash mikrokontroler
5. Perhatikan karakter yang muncul pada LCD
6. Ubah perintah DDRC = 0xFF pada fungsi Inisialisasi_LCD menjadi 0x00 dan lakukan *Make*
7. Masukkan program yang sudah diubah pada flash mikrokontroler
8. Perhatikan karakter yang muncul pada LCD dan catat hal yang terjadi
9. Ubah kembali perintah dari DDRA = 0x00 menjadi 0xFF, *Make* dan masukkan program pada flash mikrokontroler
10. Perhatikan karakter yang muncul pada LCD dan buat kesimpulan dari hasil pengamatan

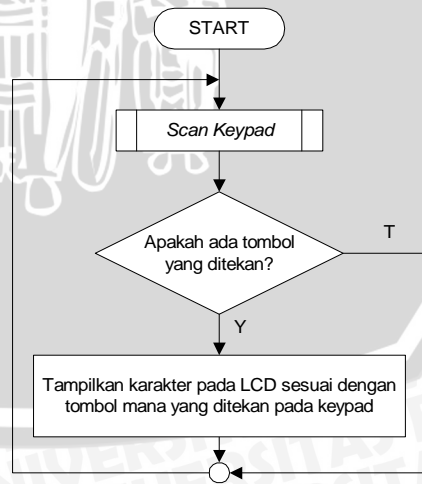
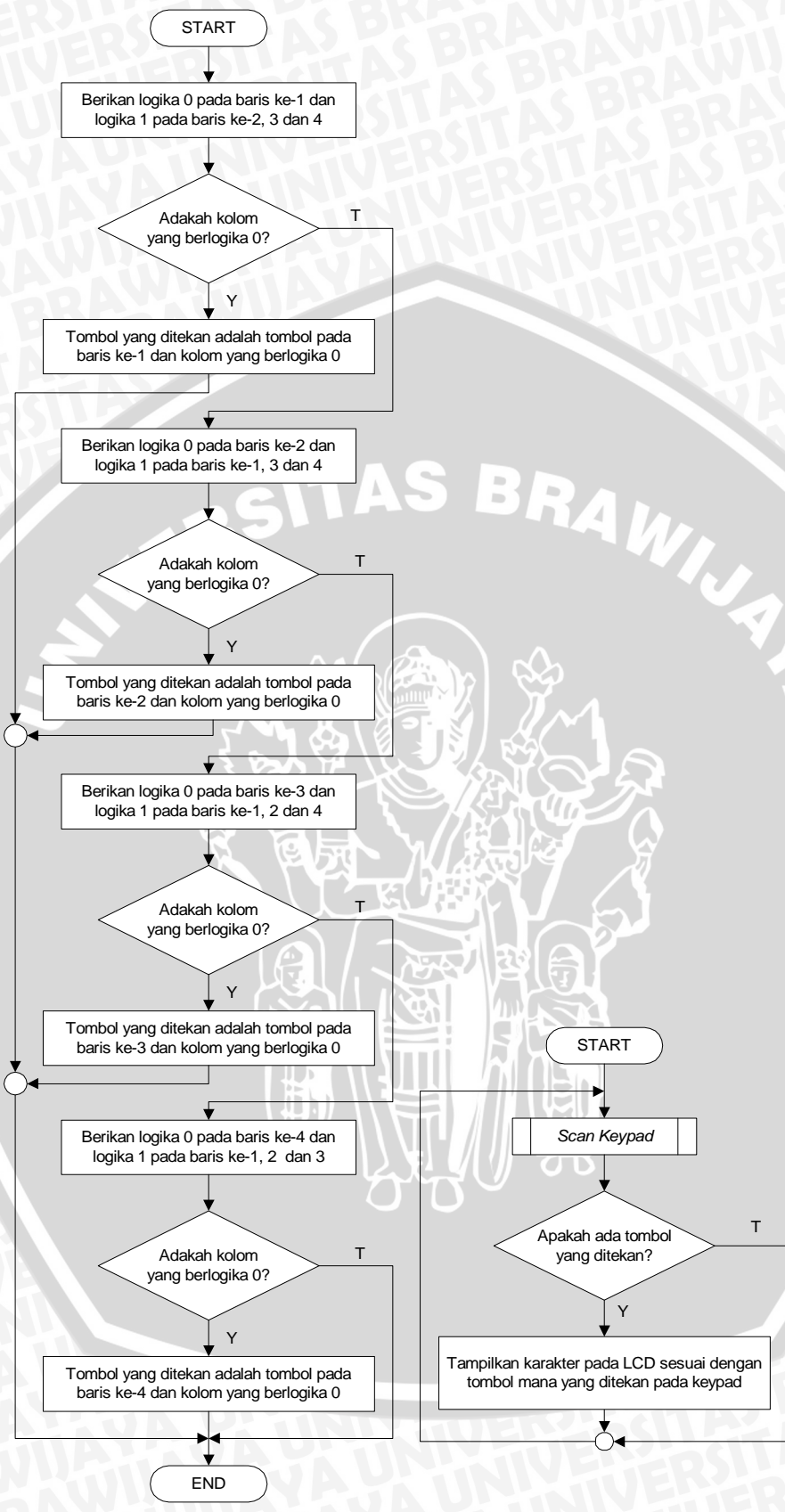
MEMASUKKAN KARAKTER DARI KEYPAD DAN MENAMPILKAN PADA LCD

Pada percobaan ini kita akan memberikan masukan pada mikrokontroler menggunakan perangkat Keypad yang sudah tersedia. Ketika keypad ditekan maka karakter yang sesuai dengan yang tertera pada tombol keypad harus muncul pada LCD. Program akan memanfaatkan file Rutin LCD yang sudah dibuat dalam percobaan sebelumnya. Diagram alir program dan Rangkaian antarmuka Keypad dengan mikrokontroler dan LCD dapat dilihat dalam gambar berikut



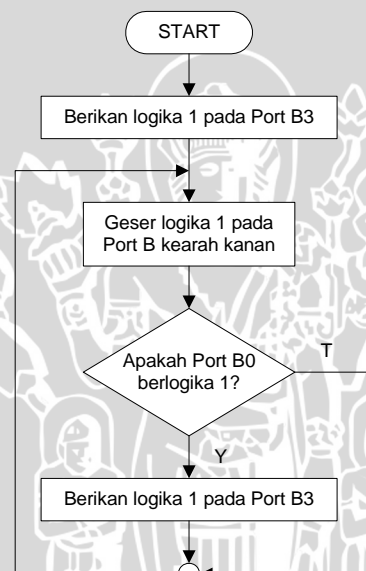
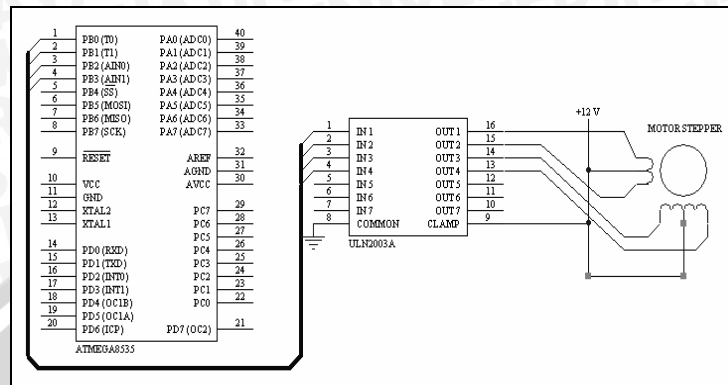
Prosedur Percobaan

1. Buka software Code Vision AVR
2. Buka Program Tampilkan Karakter Keypad dari *directory*-nya
3. Lakukan *Make*
4. Masukkan program tersebut pada flash mikrokontroler
5. Tekan setiap tombol yang ada pada Keypad dan perhatikan karakter yang muncul pada LCD
6. Tambahkan bagian program agar saat tombol ditekan berkali-kali karakter yang muncul ditampilkan tanpa menghapus karakter sebelumnya, jika sudah selesai lakukan *Make* dan masukkan program tersebut pada flash mikrokontroler untuk diuji.



MENGERAKKAN MOTOR STEPPER

Pada percobaan ini kita akan menggerakkan motor stepper dengan arah dan kecepatan yang sudah ditentukan terlebih dahulu. Diagram alir program Rangkaian antarmuka motor stepper dengan mikrokontroler dapat dilihat dalam gambar berikut



Prosedur Percobaan

1. Buka software Code Vision AVR
2. Buka Program Motor Steper dari *directory*-nya
3. Lakukan *Make*
4. Masukkan program tersebut pada flash mikrokontroler
5. Perhatikan arah putaran motor stepper dan kecepatannya
6. Ubah bagian dari listing program berikut

```
while (1)
{
    PORTB>>=1
    delay_ms(10);
    if (PORTB==0x01)
    {
        PORTB=0x08;
        delay_ms(10);
    }
};
```

Menjadi

```
while (1)
{
    PORTB<<=1
    delay_ms(10);
    if (PORTB==0x08)
    {
        PORTB=0x01;
        delay_ms(10);
    }
};
```

7. Lakukan *Make* dan masukkan program tersebut pada flash mikrokontroler
8. Catat perubahan yang terjadi
9. Ubah perintah `delay_ms(10)` menjadi `delay_ms(500)`
10. Lakukan *Make* dan masukkan program tersebut pada flash mikrokontroler
11. Catat perubahan yang terjadi dan buat kesimpulan

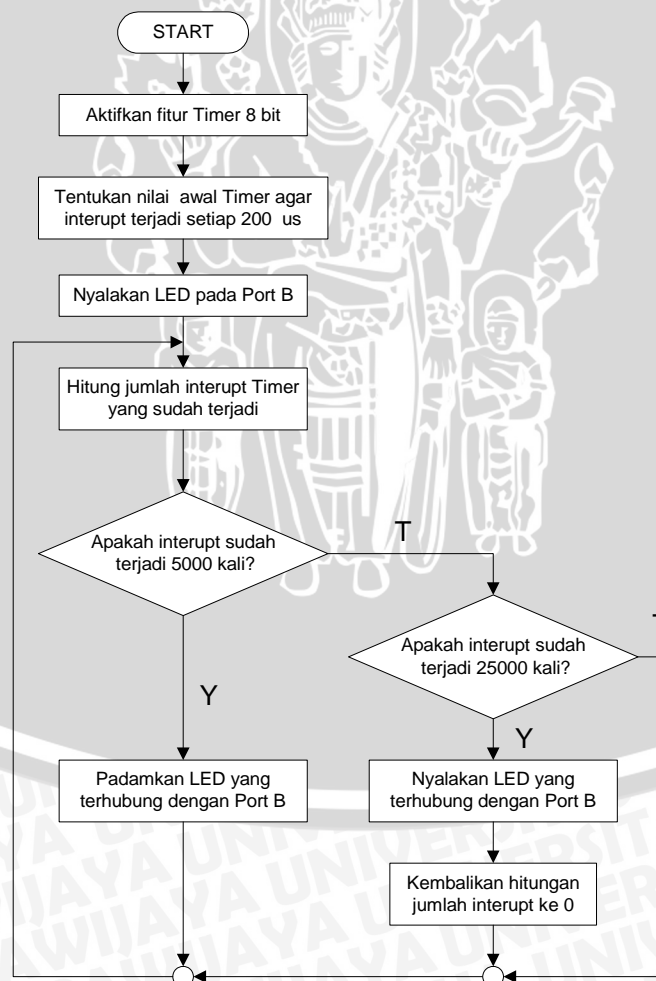


BAB IV TIMER/COUNTER

Mikrokontroler ATMEGA8535 memiliki 3 buah Timer/Counter yaitu Timer/Counter 0, Timer/Counter 1 dan Timer/Counter 2. Timer/Counter tersebut dapat dimanfaatkan sebagai pembangkit sinyal PWM (Pulse Width Modulation). ATMEGA8535 juga memiliki fasilitas Reset Watchdog yang dapat diaktifkan setiap saat. Pada bab ini kita akan mempelajari cara menggunakan Timer/Counter 0 dan 1 serta mengaktifkan fitur PWM dan Watchdog dengan memanfaatkan fasilitas lampu pijar, keypad dan Port monitoring panel.

PERCOBAAN MENYALAKAN LED SETIAP 5 DETIK

Pada percobaan ini kita akan menyalakan LED yang terhubung pada Port B setiap ± 5 detik memanfaatkan fasilitas Timer/Counter 0 mikrokontroler ATMEGA8535. Interrupt Timer dikondisikan terjadi setiap $200 \mu\text{s}$ dengan clock yang digunakan 1 MHz. Diagram alir program dapat dilihat dalam gambar berikut

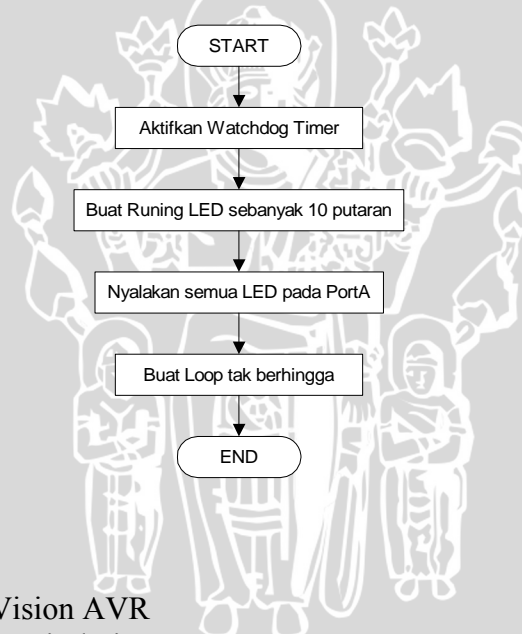


Prosedur Percobaan

1. Buka software Code Vision AVR
2. Buka Program Timer 8 Bit dari *directory*-nya
3. Lakukan *Make*
4. Masukkan program tersebut pada flash mikrokontroler
5. Perhatikan nyala LED pada Port B dan catat waktu LED menyala
6. Ubah nilai TCNT0 = 56 menjadi 156. Lakukan *Make* dan masukkan program tersebut pada flash mikrokontroler
7. Perhatikan nyala LED dan bandingkan dengan nyala LED semula.
8. Buat kesimpulan

MEMBUAT LOOP TAK BERHINGGA SEBAGAI SIMULASI KONDISI HANG

Pada percobaan ini kita akan membuat program mengalami kondisi hang melalui loop pada program yang dieksekusi secara terus menerus dan saat program mengalami kondisi demikian maka reset watchdog mikrokontroler akan aktif dan program dieksekusi dari awal. Program berupa Runing LED yang diakhiri dengan nyala LED pada Port A. Diagram alir program dapat dilihat dalam gambar berikut

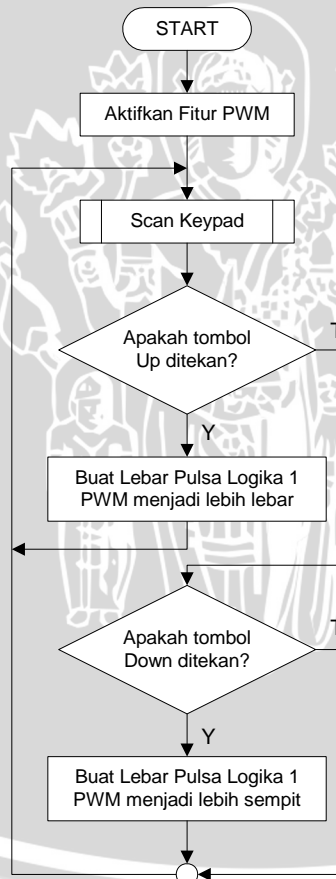
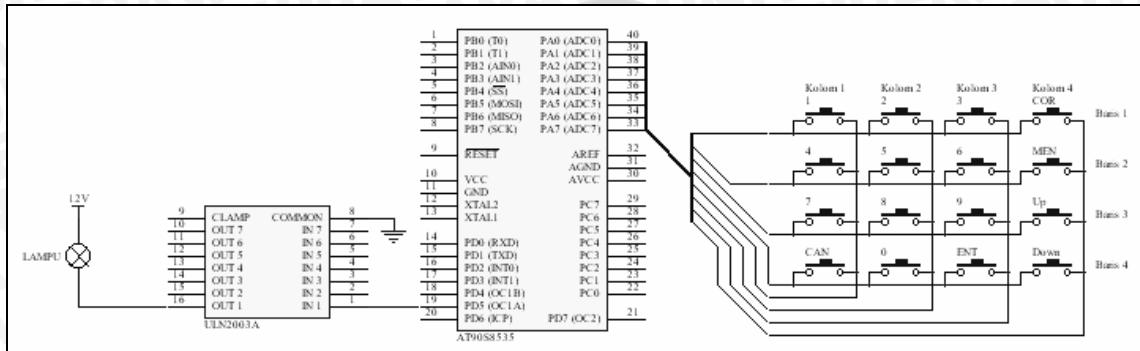


Prosedur Percobaan

1. Buka software Code Vision AVR
2. Buka Program Timer 8 Bit dari *directory*-nya
3. Lakukan *Make*
4. Masukkan program tersebut pada flash mikrokontroler
5. Perhatikan proses nyala LED yang terjadi pada Port A
6. Ubah perintah WDTCR = 0x1F menjadi 0x00. Lakukan *Make* dan masukkan program tersebut pada flash mikrokontroler
7. Perhatikan perubahan yang terjadi dan buat kesimpulan

APLIKASI DIMMER PADA LAMPU PIJAR

Pada percobaan ini kita akan memanfaatkan fasilitas PWM yang ada pada Port D5 dengan menggunakan Timer 1 16 bit mikrokontroler ATMEGA8535. Sinyal PWM akan digunakan untuk menyalakan lampu pijar yang ada pada panel. Diagram alir program dan Rangkaian antarmuka lampu pijar, keypad dan mikrokontroler dapat dilihat dalam gambar berikut.



Prosedur Percobaan

1. Buka software Code Vision AVR
2. Buka Program PWM dari *directory*-nya
3. Lakukan *Make*
4. Masukkan program tersebut pada flash mikrokontroler
5. Tekan tombol Down pada keypad untuk menurunkan tegangan efektif PWM dan tombol Up untuk menaikkan tegangan efektif PWM

6. Ubah perintah `TCCR1A=0x83` menjadi `0x00`. Lakukan *Make* dan masukkan program tersebut pada flash mikrokontroler
7. Perhatikan hal yang terjadi saat tombol Up dan Down ditekan
8. Kembalikan perintah tersebut ke posisi semula dan simulasikan kembali menggunakan perangkat keras
9. Buat kesimpulan dari hasil pengamatan

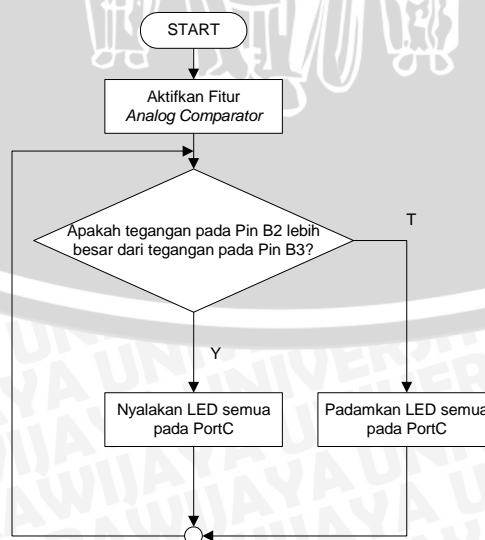
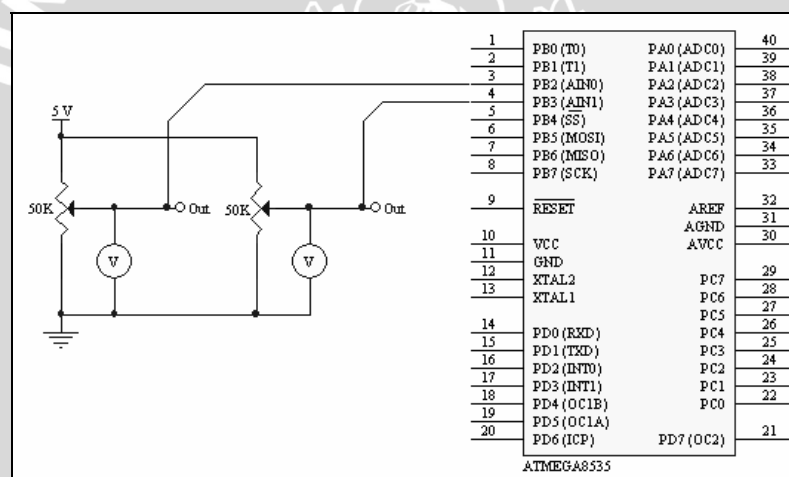


BAB V ANALOG COMPARATOR

Mikrokontroler ATMEGA8535 juga dilengkapi dengan fitur Analog *Comparator* internal dengan input negatif sebanyak 9 buah yang dapat digunakan secara bergantian. Pada bab ini kita akan mempelajari cara menggunakan fitur Analog Comparator Internal mikrokontroler ATMEGA8535 dengan memanfaatkan sumber tegangan yang dapat diinterpolasi dan Port monitoring panel.

PERCOBAAN MEMBUAT LED MENYALA JIKA SUATU TEGANGAN MELEBIHI TEGANGAN REFERENSI

Pada percobaan kali ini kita akan membandingkan besar tegangan yang diberikan pada Pin AIN0 dan AIN1, apabila tegangan pada AIN0 lebih besar dari AIN1 maka LED pada Port C akan menyala. Jika yang terjadi adalah yang sebaliknya maka LED pada Port C akan padam. Diagram alir program dan Rangkaian antarmuka sumber tegangan dengan mikrokontroler dapat dilihat dalam gambar berikut



Prosedur Percobaan

1. Pasang kabel dari sumber tegangan ke Pin B2 dan B3
2. Buka software Code Vision AVR
3. Buka Program PWM dari *directory*-nya
4. Lakukan *Make*
5. Masukkan program tersebut pada flash mikrokontroler
6. Berikan tegangan pada Pin AIN0 dan AIN1
7. Perhatikan nyala LED pada Port C dan ukur tegangan pada pin-pin tersebut
8. Lakukan interpolasi tegangan pada pin-pin tersebut
9. Catat nilai tegangan pada setiap perubahan dan nyala LED
10. Ubah perintah ACSR = 0x00 menjadi 0x80. Lakukan *Make* dan masukkan program tersebut pada flash mikrokontroler
11. Lakukan interpolasi tegangan kembali pada pin-pin tersebut dan perhatikan nyala LED pada Port C
12. Buat kesimpulan dari hasil pengamatan

UNIVERSITAS BRAWIJAYA



BAB VI EEPROM

Mikrokontroler ATMEGA8535 juga dilengkapi dengan fitur EEPROM 512 Bytes. EEPROM tersebut dapat digunakan setiap saat hanya dengan mengatur register kontrrolnya. Pada bab ini kita akan mempelajari cara menggunakan fitur EEPROM dari mikrokontroler ATMEGA8535 dengan memanfaatkan rangkaian LCD dan Keypad pada panel.

PERCOBAAN MENYIMPAN DATA KE EEPROM DAN MENAMPILKAN DATA DARI EEPROM PADA LCD

Pada percobaan ini kita akan menyimpan data berupa angka maksimal 16 kombinasi angka pada EEPROM internal mikrokontroler sesuai dengan jumlah karakter yang dapat ditampilkan secara bersamaan pada baris pertama LCD. Angka yang disimpan dimasukkan sebagai data melalui keypad dan ditampilkan pada LCD saat proses menyimpan dan membaca kembali data tersebut. Untuk menyimpan angka digunakan tombol ENT sedangkan untuk menampilkan data yang sudah tersimpan digunakan tombol CAN. Sebagai proses uji apakah penyimpanan data berhasil maka setelah data disimpan maka catu daya akan dimatikan terlebih dahulu sebelum data ditampilkan kembali. Diagram alir program dapat dilihat dalam gambar berikut

Prosedur Percobaan

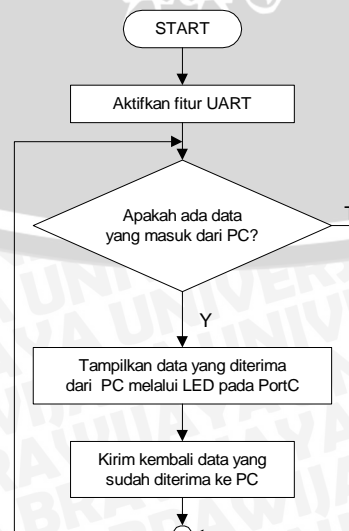
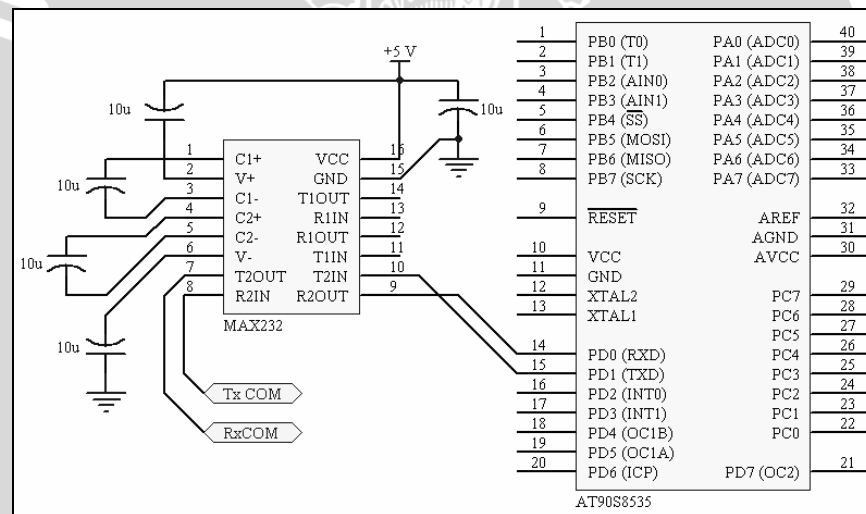
1. Buka software Code Vision AVR
2. Buka Program EEPROM dari *directory*-nya
3. Lakukan *Make*
4. Masukkan program tersebut pada flash mikrokontroler
5. Masukkan angka-angka yang akan disimpan melalui keypad dan catat angka-angka tersebut.
6. Simpan angka-angka tersebut dan matikan catu daya
7. Nyalakan kembali catu daya kemudian tampilkan data-data yang sudah tersimpan sebelumnya. Lakukan proses tersebut sebanyak lima kali sambil dicatat data-data yang dimasukkan dan yang ditampilkan.

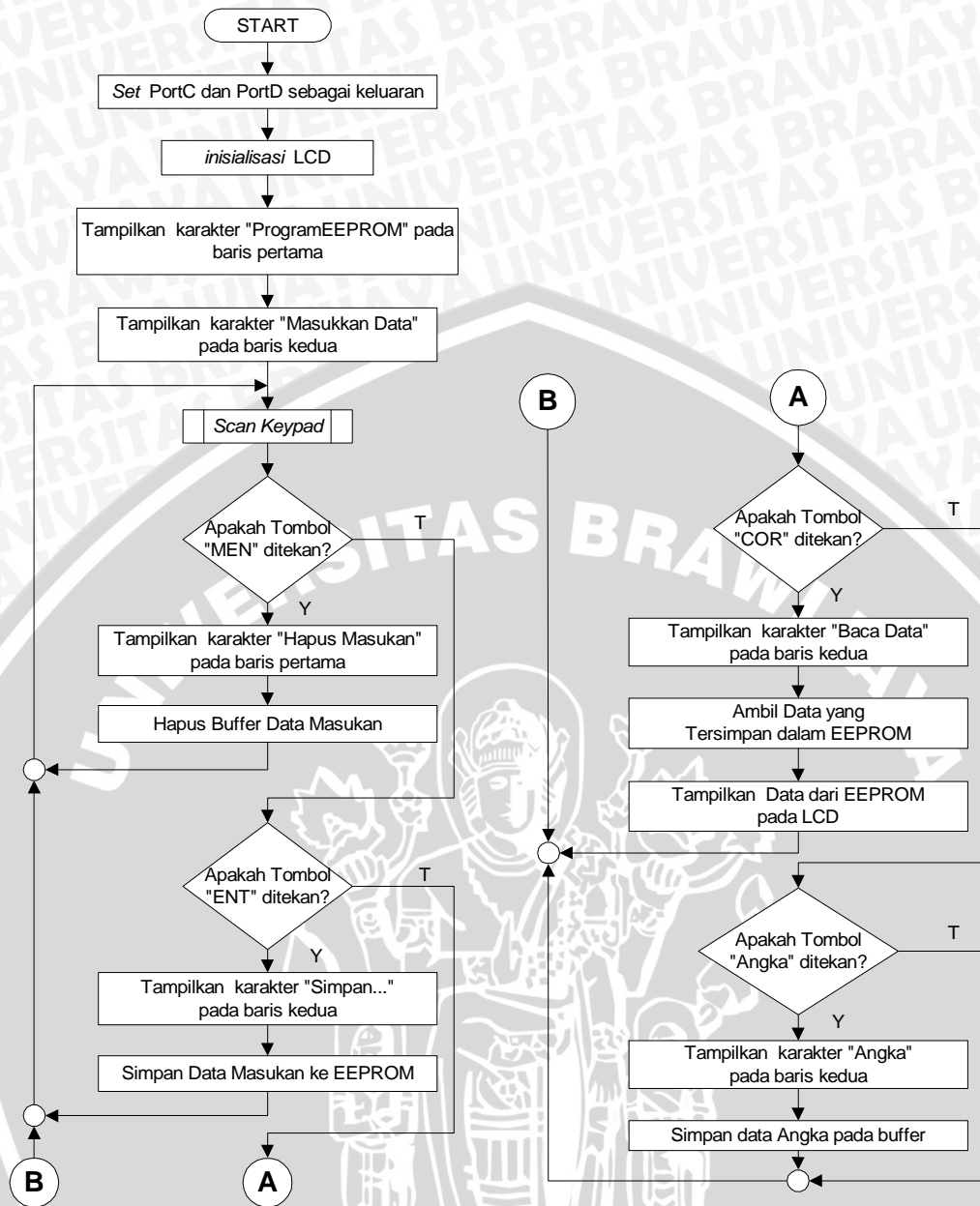
BAB VII PROGRAMABLE UART

Programable UART merupakan salah satu fitur yang dimiliki oleh mikrokontroler ATMEGA8535 yang dapat digunakan untuk melakukan komunikasi data dengan mikrokontroler lain atau komputer. Pada bab ini kita akan mempelajari bagaimana cara menggunakan fitur tersebut dengan memanfaatkan rangkaian RS232 dan Port monitoring panel. pada panel serta komputer.

PERCOBAAN MENGIRIM/MENERIMA DATA DARI/KE PC

Pada percobaan ini kita akan mengirimkan data dari komputer melalui fasilitas Terminal software Code Vision AVR. Data yang dikirim melalui komputer akan ditampilkan melalui LED pada Port C dan dikirim kembali ke komputer. Data yang diterima oleh komputer dapat dilihat pada layar monitor. Diagram alir program dan Rangkaian antarmuka RS232 dengan mikrokontroler dapat dilihat dalam gambar berikut.

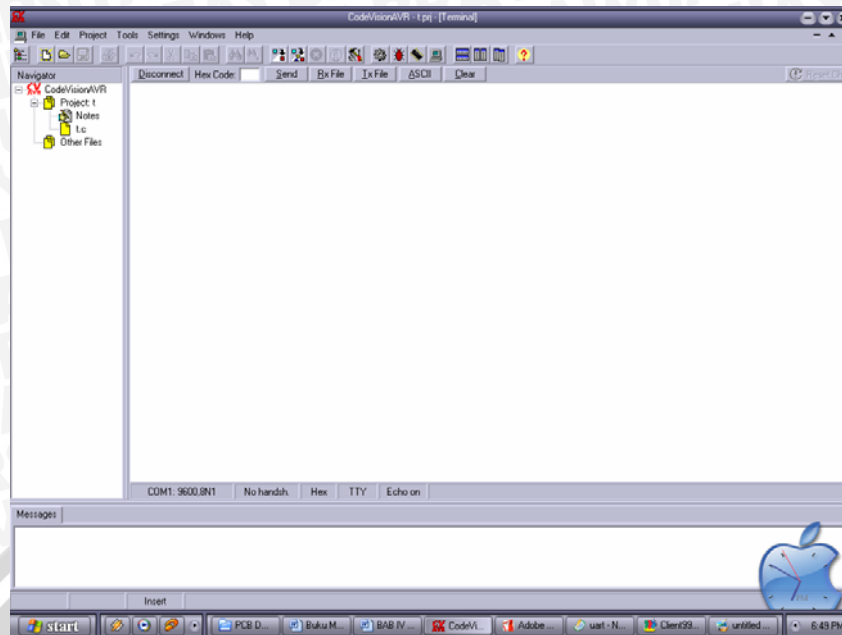




Prosedur Percobaan

1. Pasang kabel serial dari rangkaian RS232 ke port serial dari komputer
2. Buka software Code Vision AVR
3. Buka Program UART dari *directory*-nya
4. Lakukan *Make*
5. Masukkan program tersebut pada flash mikrokontroler
6. Kirim data dari komputer dengan cara sebagai berikut:
 - a. Klik **Tools >> Terminal** dan akan muncul window berikut





- b. Klik **Hex** untuk mengirim data dalam format data Heksa
- c. Ketikkan data yang akan dikirim dalam format heksa pada kolom Hex. Code kemudian klik **Send**
7. Perhatikan data yang tampil pada Port C dan data yang muncul pada layar monitor komputer
8. Ubah Perintah UCSRB = 0x48 pada fungsi Send_Data menjadi 0x40. Lakukan *Make* dan masukkan program tersebut pada flash mikrokontroler
9. Kirim kembali data dari komputer dan perhatikan hal yang terjadi. Lakukan pengiriman data beberapa kali
10. Buat kesimpulan dari hasil pengamatan

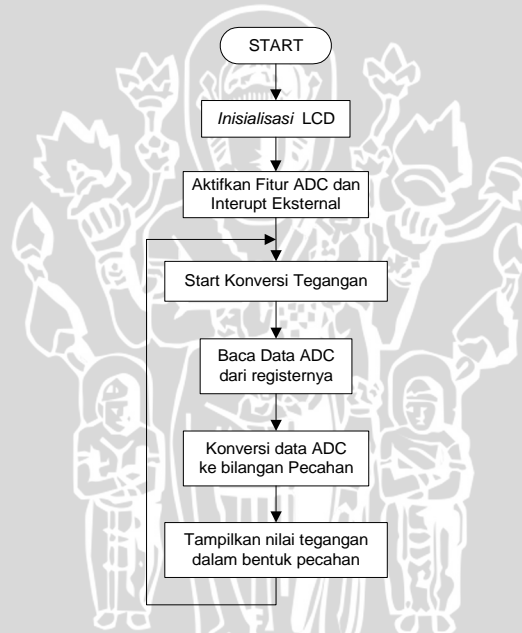


BAB VIII INTERNAL ADC

Mikrokontroler ATMEGA8535 juga dilengkapi dengan ADC internal 8 masukan yang dapat digunakan secara bergantian. ADC tersebut memiliki tingkat kesalahan maksimal ± 2 LSB dengan resolusi tertinggi 10 bit. Pada bab ini kita akan mempelajari cara menggunakan fitur ADC dengan memanfaatkan sumber tegangan yang dapat diinterpolasi dan rangkaian LCD pada Panel.

PERCOBAAN MENAMPILKAN NILAI TEGANGAN DC PADA LCD

Pada percobaan ini kita akan menampilkan nilai tegangan yang diberikan pada pin A0 yaitu channel ADC0 pada layar LCD. Tegangan yang diberikan antara 0 – 5 V dengan tegangan referensi 5 V dan resolusi ADC 8 bit. Diagram alir program dapat dilihat dalam gambar berikut



Prosedur Percobaan

1. Pasang kabel dari sumber tegangan pada pin A0
2. Buka software Code Vision AVR
3. Buka Program Tampilkan Nilai Tegangan dari *directory*-nya
4. Lakukan *Make*
5. Masukkan program tersebut pada flash mikrokontroler
6. Lakukan interpolasi tegangan yang diberikan pada pin A0 dan lakukan pengukuran
7. Perhatikan nilai tegangan yang tampil pada LCD
8. Ubah perintah `ADCSRA|=0x40` menjadi `0x00`. Lakukan *Make* dan masukkan program tersebut pada flash mikrokontroler.
9. Lakukan interpolasi tegangan kembali dan perhatikan nilai tegangan yang muncul pada LCD
10. Buat kesimpulan dari hasil pengamatan

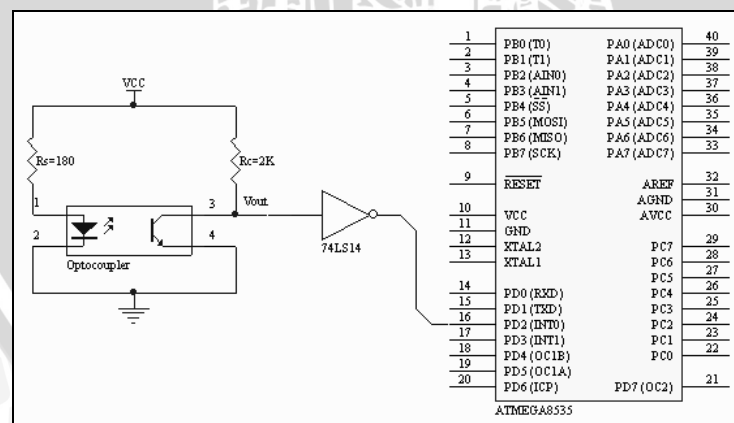
BAB IX INTERUPT EKSTERNAL

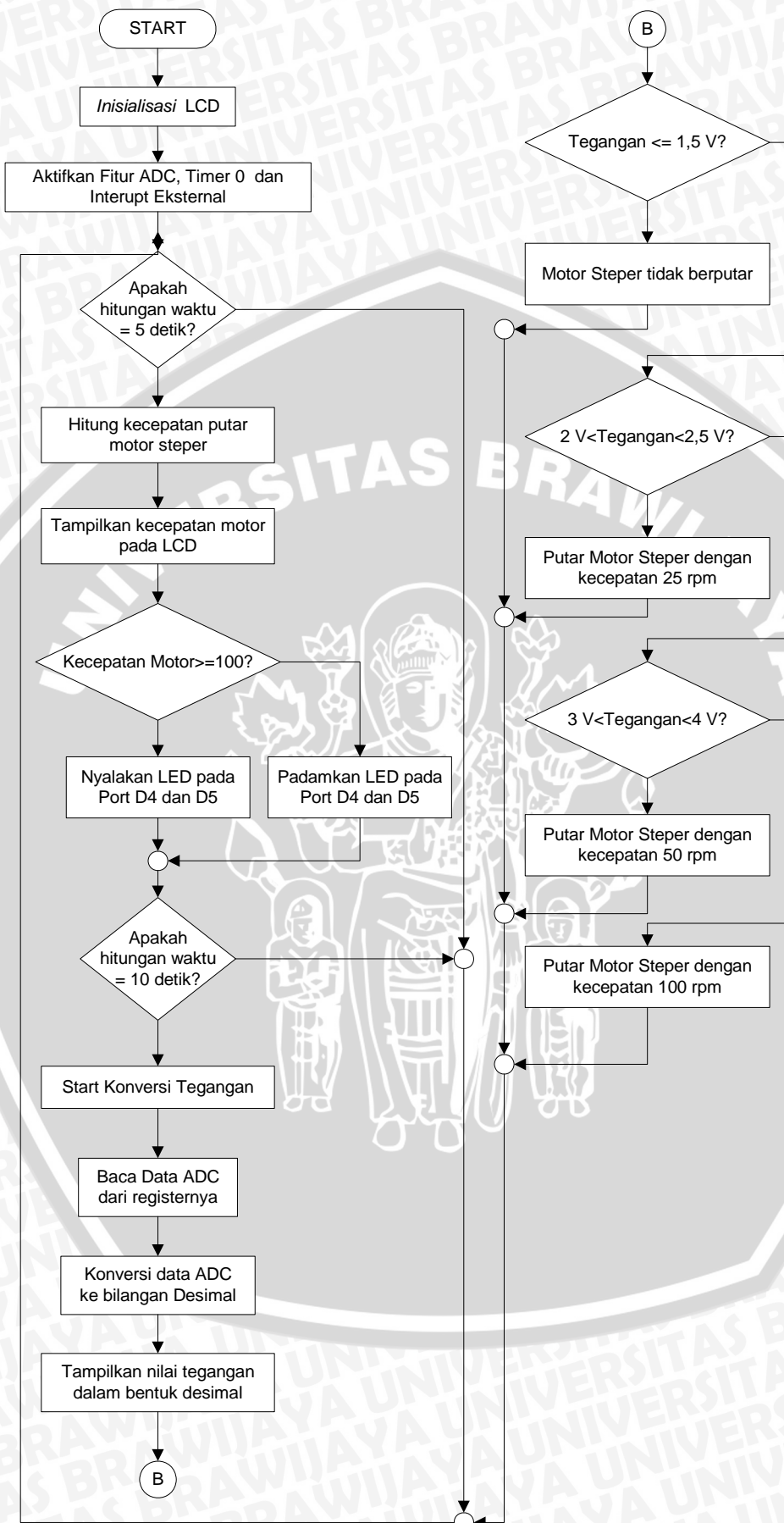
Fitur terakhir yang akan kita pelajari menggunakan modul ini adalah Eksternal Interrupt. Mikrokontroler ATMEGA8535 memiliki 3 fasilitas Eksternal Interrupt pada pin B2, D2 dan D3. Pada bab ini kita akan mempelajari penggunaan Eksternal Interrupt tersebut dan mempelajari contoh program yang bersifat kompleks dengan memanfaatkan Rangkaian Optoswitch, motor stepper, LCD, LED dan sumber tegangan yang dapat diinterpolasi.

PERCOBAAN SIMULASI PENGATURAN SUHU RUANGAN

Pada percobaan ini kita akan mensimulasikan pengaturan suhu ruangan. Sensor suhu diwakili oleh sumber tegangan yang dapat diinterpolasi, kipas angin diwakili oleh motorsteper dan alarm diwakili oleh LED pada Port D4 dan D5. Untuk mengukur kecepatan kipas angin digunakan rangkaian optoswitch yang dihubungkan pada Pin D2. besar nilai Suhu (Tegangan) dan kecepatan kipas angin (Motor Steper) ditampilkan pada LCD.

Jika tegangan yang diberikan mencapai $2\text{ V} - 2,5\text{ V}$ maka kipas yang diwakili oleh motor stepper akan berputar dengan kecepatan 25 rpm. Jika tegangan terus bertambah mencapai $3\text{ V} - 4\text{ V}$ maka motor stepper akan berputar lebih cepat yaitu 50 rpm sedangkan saat tegangan mencapai $4,5\text{ V}$ atau lebih maka motor stepper akan berputar dengan kecepatan 100 rpm. Ketika kecepatan motor stepper mencapai 100 rpm, LED yang pada Port D.4 dan D.5 akan menyala untuk mewakili tanda bahaya. Jika tegangan yang diberikan lebih kecil atau sama dengan $1,5\text{ V}$ maka kipas tidak berputar. Rangkaian antarmuka Optoswitch dengan mikrokontroler dapat dilihat dalam gambar berikut





Prosedur Percobaan

1. Pasang kabel dari sumber tegangan pada pin A0
2. Buka software Code Vision AVR
3. Buka Program Simulasi Pengaturan Suhu dari *directory*-nya
4. Lakukan *Make*
5. Masukkan program tersebut pada flash mikrokontroler
6. Lakukan interpolasi pada sumber tegangan yang diberikan pada pin A0 dari 0 V hingga 5 V secara bertahap
7. Perhatikan setiap hal yang terjadi pada setiap bagian panel yang digunakan

