

# REKAYASA PROTOTIPE ROBOT PEMADAM API DAN PELACAK JEJAK BERBASIS MIKROKONTROLER ATMEGA8535

Muhammad Abdurohman<sup>1)</sup>, Aris Kiswanto<sup>2)</sup>, Moh Toni<sup>3)</sup>

<sup>1)</sup> Program Studi Teknik Elektro Fakultas Teknik  
Universitas Muhammadiyah Semarang  
Email: [rohman.wae@gmail.com](mailto:rohman.wae@gmail.com)

## ABSTRAK

*Rancang bangun robot pemadam api yang juga dapat mengikuti jejak menggunakan micro chip ATmega8535, dengan spesifikasi robot menggunakan foto dioda delapan buah sebagai sensor pelacak jejak sehingga robot lebih baik untuk kemungkinan keluar dari garis yang telah ditentukan, sensor jarak sebagai alat navigasi robot agar tidak menabrak, sensor api untuk mendeksi api yang memungkinkan sensor ini bisa diganti-ganti sesuai dengan kebutuhan bisa dengan UVtron, LM35, LDR, atau foto dioda. Sensor mendeteksi adanya sumber api dan mengaktifkan alat pemadam, penulis memasang fan sebagai pemadam apinya.*

*Dilengkapi dengan LCD untuk memonitor progress kerja robot yang sekaligus untuk mengubah program robot secara langsung dengan bantuan empat tombol yang difungsikan sebagai menu atau cancel, ok atau start, up dan down. Program yang dapat dirubah melalui robot adalah kecepatan, pengambilan data sensor terhadap dasar putih sebagai kalibrasi, mencoba fungsi kerja robot dengan demo (maju, mundur, belok kanan, belok kiri). Juga dapat merubah program robot sesuai dengan program pilihan yang sebelumnya telah dimasukkan dalam menu utama robot tanpa menggunakan PC..*

**Kata Kunci:** robot, pemadam api, pelacak jejak, ATmega8535

## 1. PENDAHULUAN

### Latar Belakang

Mikrokontroler dewasa ini merupakan suatu alat untuk otomasi yang paling banyak digunakan, baik peralatan rumah tangga, industri, pertanian dan banyak sektor lainnya. Sehingga pemakaian maupun pengetahuan alat memang sangat penting kita ketahui. Namun pada kenyataannya kita tidak mengetahui maupun memahami meskipun kita sudah sering menggunakan dan tidak bisa lepas dari alat-alat tersebut di era sekarang ini. Contoh saja mesin

cuci, microwave, control pintu pagar, sistem alarm, control mesin mobil, kontrol pintu mobil, inkubator, permainan anak-anak dan masih banyak lagi.

Dalam hal ini mikrokontroler terdiri beberapa komponen yang terpaket menjadi satu dalam sebuah cips IC (*integrated cicuit*). Dalam desain mikrokontroler merupakan suatu desain fungsi kontrol otomasi dengan memanipulasi data yang masuk maupun keluar (*input-output*) yang dapat kita rubah sesuai kebutuhan.

Mikrokontroler ATmega8535 mudah dalam aplikasinya, memiliki 4 buah port input/output 8 bit, yaitu PORTA, PORTB, PORTC, dan PORTD. Selain sebagai *input/output* masing masing port juga memiliki fungsi yang lain. PORTA dapat difungsikan sebagai ADC (*Analog to Digital Converter*), PORTB dapat difungsikan sebagai SPI (Serial Peripheral Interface) communication.

### Pembatasan Masalah

Ada banyak jenis dan macamnya mikrokontroler diantaranya atmel AVR, microchip PIC16C74, motorola 68HC11, 8051 (MCS51), dan hitachi H8. Dari jenis mikrokontroler AVR saja ada tiga kelompok yaitu, ATmega, TinyAVR, dan AT90Sxx. Dari beberapa jenis mikrokontroler yang ada penulis memilih menggunakan AVR tipe ATmega seri 8535 sebagai rancang bangun robot pemadam api dan mampu sebagai pengikut garis (Malik Moh Ibnu, dkk. 1997).

### Tujuan

1. Membuat rancang bangun alat robot pintar yang mampu memadamkan api dan melacak jejak.

Membuat perangkat lunak yang berfungsi sebagai system operasi robot agar dapat menjalankan perintah sesuai fungsinya.

Rekayasa Prototipe Robot.....

### Manfaat

1. Pembuatan prototipe robot pemadam api dapat ditindak lanjuti untuk robot pemadam api sesungguhnya berbasis mikrokontroler.

2. Selain dari manfaat pemrograman juga sebagai bahan pembelajaran dalam pembuatan robotic, sebagai pijakan awal untuk mempelajari teknik digital dan kontrol

## 2. LANDASAN TEORI

### 2.1 Dasar Teori

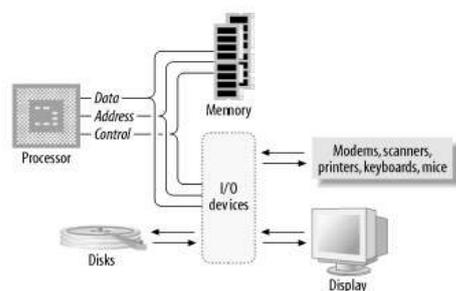
Pada penelitian ini dipilih mikrokontroller jenis ATMEL AVR RISC dengan pertimbangan sebagai berikut:

Atmel avr risc memiliki fasilitas dan kefungisian yang lengkap dengan harga yang relatif murah. Kecepatan maksimum eksekusi instruksi mikrokontroller mencapai 16 MIPS (*Million Instruction per Second*), yang berarti hanya dibutuhkan 1 clock untuk 1 eksekusi instruksi sebesar 16 juta. Konsumsi daya yang rendah jika dibandingkan dengan kecepatan eksekusi instruksi. Ketersediaan kompilator C (*Code Vision AVR*) sehingga memudahkan *user* atau pengguna bisa langsung memprogram menggunakan bahasa C (Andrianto Heri. 2008).

**Tabel 2.1.** Perbandingan kecepatan processor dan efisiensi<sup>1</sup>

Processor	Execution time
AVR	335
8051	9384
PIC16C74	2492
68HC11	5244

Dari tabel diatas dapat dilihat, ketika bekerja dengan kecepatan clock yang sama AVR 7 kali lebih cepat dibandingkan dengan PIC16C74, 15 kali lebih cepat daripada 68HC11, dan 28 kali lebih cepat dibanding 8051. Dari kemampuan dan fasilitas yang dimiliki, AVR RISC cocok dipilih sebagai mikrokontroler untuk membangun bermacam-macam aplikasi *embedded system* (Garland Harry. 1984).



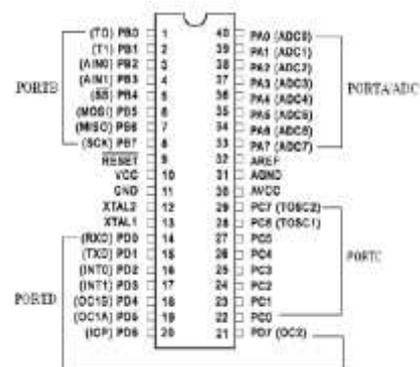
**Gambar 2.1.** Sistem komputer sederhana  
**2.1.1.1. Mikrokontroler ATmega8535**

Oleh karena itu, dalam penelitian ini juga dipilih salah satu jenis AVR RISC sebagai dasar penelitian yaitu ATmega8535. Chip AVR ATmega8535

memiliki 40 pin kaki, berikut skema kaki ATmega8535 (Zaks Rodnay. 1988).

2.1.1.1. Spesifikasi:

Fitur-fitur yang dimiliki oleh mikrokontroler ATmega8535 adalah sebagai berikut:



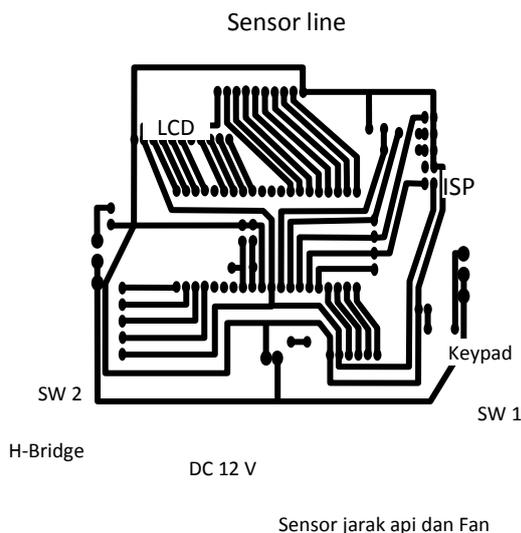
**Gambar 2.2** Skema mikrokontroler avr risc ATmega8535<sup>2</sup>

- Saluran I/O sebanyak 32 buah, yaitu portA, portB, portC, dan portD.
- ADC internal sebanyak 8 saluran.
- Tiga buah *Timer/Counter* dengan kemampuan perbandingan.
- CPU yang terdiri atas 32 buah register.
- SRAM sebesar 512 byte.
- Memori Flash sebesar 8 kb dengan kemampuan *Read While Write*.
- Port antarmuka SPI
- EEPROM sebesar 512 byte yang dapat diprogram saat operasi.
- Antarmuka komparator analog.
- Port USART untuk komunikasi serial.

k. Sistem mikroprosesor 8 bit berbasis RISC dengan kecepatan maksimal 16 MHz.

2.1.1.2. PIN Description

Pada gambar dibawah ini. Sebagai contoh skema pin pada ATmega8535 yang penulis buat yang telah dikelompokkan sesuai dengan kebutuhan dan fungsionalnya.



**Gambar 2.3.** Skema pin mikrokontroler

Dalam pembuatannya pin-pin tersebut tersusun dalam sekema papan rangkaian dan dikelompokkan dalam group sehingga memudahkan dalam aplikasinya

2.1.1.3. Input/Output

Fasilitas *input/output* merupakan fungsi mikrokontroler untuk dapat menerima sinyal masukan (*input*) dan memberikan sinyal keluaran (*output*). Sinyal *input* maupun sinyal *output* adalah berupa data digital 1 (*high*, mewakili Rekeyasa Prototipe Robot.....

tegangan 5 volt) dan 0 (*low*, mewakili tegangan 0 volt). Mikrokontroler ATmega8535 memiliki 4 buah PORT 8 bit bidirectional yang dapat difungsikan sebagai PORT *input* maupun PORT *output* yaitu PORTA, PORTB, PORTC, dan PORT D. Register digunakan untuk mengatur fungsi dari pin-pin pada tiap port. Register dapat dianalogikan sebagai kumpulan *switch on/off* yang digunakan untuk mengaktifkan fungsi apa yang akan dipakai dari port mikrokontroler (Soebhakti Hendrawan, 2007)

*Tri-state* adalah kondisi diantara *high* dan *low*, atau biasa disebut dengan keadaan mengambang (*floating*). Kondisi *tri-state* sangat dihindari dalam dunia digital. Terlepas dari setting DDRxn, PINxn merupakan register yang berfungsi untuk mengetahui keadaan tiap-tiap pin pada mikrokontroler. Register ini sangat dibutuhkan untuk membaca keadaan pin pada saat difungsikan sebagai input.

2.1.1.4. Interrupt

*Interrupt* adalah fasilitas mikrokontroler untuk menyela suatu program yang sedang berjalan, seperti pada tabel. Interrupt dapat dianalogikan sebagai hak untuk menyela pada suatu rapat. Dari sekian banyak peserta rapat hanya 21 orang yang diberi hak untuk menyela. Jika terdapat 2 atau lebih orang yang menyela, maka orang dengan

prioritas paling tinggi yang diperbolehkan bicara. Pada ATmega8535 terdapat 21 fasilitas *interrupt* dengan prioritas

#### 2.1.1.5. Timer/Counter

*Timer* dan *counter* adalah dua fasilitas yang memiliki perangkat yang sama, seperti halnya register penampungnya (TCNTx). Ketika difungsikan sebagai timer, maka register penampung tersebut berisikan jumlah waktu yang terlampaui tiap selang waktu tertentu. Besar selang waktu tersebut dapat disetting sesuai dengan kebutuhan. Jika dipakai sebagai *counter*, maka register penampung tersebut digunakan untuk menyimpan data hasil perhitungan terakhir.

Saat difungsikan sebagai *counter*, maka masuk melewati pin T0 dan T1. Register untuk mengatur kapan *timer* difungsikan sebagai *timer* dan kapan sebagai *counter* adalah TCCRx. ATmega8535 memiliki fasilitas 3 buah *timer/counter* yaitu *timer/counter0* 8 bit, *timer/counter1* 16 bit, dan *timer/counter2* 8 bit. 8 bit dan 16 bit adalah jumlah data yang bisa ditampung pada register penampungnya.

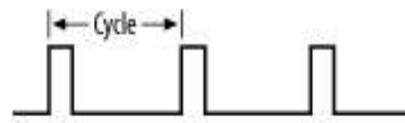
#### 2.1.1.6. PWM (Pulse Width Modulation)

PWM (*Pulse Width Modulation*) adalah teknik mendapatkan efek sinyal

analog dari sebuah sinyal digital yang terputus-putus.

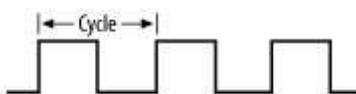
PWM dapat dibangkitkan hanya dengan menggunakan digital i/o yang difungsikan sebagai output.

Pada contoh gelombang diatas, perbandingan waktu antara sinyal *high* (1) dan sinyal *low* (0) adalah sama.



**Gambar 2.4.** PWM dengan *duty cycle* 50%

Gelombang diatas dikatakan memiliki *duty cycle* 50%. *Duty cycle* adalah perbandingan antara lebar sinyal *high* (1) dengan lebar keseluruhan siklus (*cycle*). Jika amplitudo gelombang PWM adalah 5 volt, maka tegangan rata rata (seolah-olah analog) yang kita dapatkan adalah 2,5 volt. Berikut contoh gelombang PWM dengan *duty cycle* 10%, jika amplitudo gelombang 5 volt maka akan didapatkan tegangan rata rata analog 0,5 volt.



**Gambar 2.5.** PWM dengan *duty cycle* 10%

Pada ATmega8535 ada 2 cara membangkitkan PWM, yang pertama PWM dapat dibangkitkan dari port input/outputnya yang difungsikan sebagai

Muhammad Abdurohman, Aris Kiswanto, Moh Toni

output. Yang kedua adalah dengan memanfaatkan fasilitas PWM dari fungsi timer/counter yang telah disediakan. Dengan adanya fasilitas ini proses pengaturan waktu high/low sinyal digital tidak akan mengganggu urutan program lain yang sedang dieksekusi oleh processor. Selain itu, dengan menggunakan fasilitas ini kita tinggal memasukkan berapa porsi periode waktu on dan off gelombang PWM pada sebuah register. OCR1A, OCR1B dan OCR2 adalah register tempat mengatur duty cycle PWM. Pada bab ini akan dipergakan bagaimana cara mendapatkan sinyal analog dari sebuah sinyal digital dengan menggunakan teknik PWM.

#### 2.1.1.7. USART (*Universal Synchronous and Asynchronous Serial Receiver and Transmitter*)

Agar komunikasi serial dapat berjalan dengan baik dibutuhkan suatu protocol/aturan komunikasi. Pada ATmega8535 terdapat beberapa protocol komunikasi serial, yaitu :

USART, SPI, dan I2C. Bab ini akan membahas protocol komunikasi USART, serta mempraktikkan komunikasi antar dua buah mikrocontroller. Dengan menggunakan protocol USART ada 2 jenis mode komunikasi, yaitu :

*Sinkron*, dan *asinkron*. Pada *mode sinkron*, mikrocontroller dan *peripheral* yang berkomunikasi akan menggunakan clock atau detak kerja yang sama, sedangkan pada *mode asinkron* mikrocontroller dan *peripheral* bisa bekerja pada clock-nya masing-masing.

#### 2.1.1.8. ADC (*Analog to Digital Conversion*)

ADC (*Analog to Digital Converter*) adalah salah satu fasilitas mikrocontroller ATmega8535 yang berfungsi untuk mengubah data analog menjadi data digital. ADC memiliki 2 karakter prinsip, yaitu kecepatan *sampling* dan *resolusi*. Kecepatan *sampling* suatu ADC menyatakan seberapa sering sinyal analog dikonversikan ke bentuk sinyal digital pada selang waktu tertentu. Kecepatan *sampling* biasanya dinyatakan dalam *sample per second* (SPS).

Resolusi ADC menentukan ketelitian nilai hasil konversi ADC. Sebagai contoh ADC 8 bit akan memiliki output 8 bit data digital, ini berarti sinyal input dapat dinyatakan dalam 255 ( $2^n - 1$ ) nilai diskrit. ADC 12 bit memiliki 12 bit output data digital, ini berarti sinyal input dapat dinyatakan dalam 4096 nilai diskrit. Dari contoh diatas ADC 12 bit akan memberikan ketelitian nilai hasil konversi yang jauh lebih baik daripada ADC 8 bit.

Prinsip kerja ADC adalah mengkonversi sinyal analog ke dalam bentuk besaran yang merupakan rasio perbandingan sinyal input dan tegangan referensi. Sebagai contoh, bila tegangan referensi 5 volt, tegangan input 3 volt, rasio input terhadap referensi adalah 60%. Jadi, jika menggunakan ADC 8 bit dengan skala maksimum 255, akan didapatkan sinyal digital sebesar  $60\% \times 255 = 153$  (bentuk decimal) atau 10011001 (bentuk biner).

ADC pada ATmega8535 adalah jenis *10 bit successive approximation* dengan tegangan referensi maksimum 5 volt. Pada universal board M.B.3.2 tegangan referensi dibuat fix tidak dapat diubah yaitu 5 volt yang diambil dari tegangan sumber (Vcc). Register-register yang harus di setting adalah ADMUX, ADCSRA, dan SFIOR.

Untuk memilih channel ADC yang digunakan (single ended atau diferensial), dilakukan dengan mengatur nilai MUX4 : 0. Misalnya channel ADC0 sebagai input ADC, maka MUX4 : 0 diberi nilai 00000B. Informasi lebih lengkap dapat dilihat pada datasheet mikrokontroler yang dipakai.

Tegangan referensi ADC dapat ditentukan antara lain dari pin AREF, pin AVCC atau menggunakan tegangan referensi internal mikrokontroler sebesar

2.56V. Agar fitur ADC mikrokontroler dapat digunakan maka ADEN (ADC *Enable*, dalam I/O register ADCSRA) harus diberi nilai 1. Setelah konversi selesai (ADIF *high*), hasil konversi dapat diperoleh pada register hasil (ADCL, ADCH). Untuk konversi single ended, hasilnya ialah :

$$ADC = \frac{V_{IN} \cdot 1024}{V_{REF}} \dots\dots\dots(1)$$

Di mana VIN ialah tegangan pada input yang dipilih dan VREF merupakan tegangan referensi. Jika hasil ADC = 000H, maka menunjukkan tegangan input sebesar 0V, jika hasil ADC = 3FFH menunjukkan tegangan input sebesar tegangan referensi dikurangi 1 LSB. Sebagai contoh, jika diberikan VIN sebesar 0.2V dengan VREF 5V, maka hasil konversi ADC ialah 41. Jika menggunakan *differential channel*, hasilnya ialah 40.96, yang bila dikenakan bisa sekitar 39,40,41 karena ketelitian ADC ATmega 16 sebesar  $\pm 2$ LSB. Jika yang digunakan saluran diferensial, maka hasilnya ialah :

$$ADC = \frac{(V_{POS} - V_{NEG}) \cdot GAIN \cdot 512}{V_{REF}} \dots\dots\dots/(2)$$

Di mana VPOS ialah tegangan pada input pin positif, VNEG ialah tegangan input pada pin negatif, GAIN ialah faktor penguatan dan VREF ialah tegangan

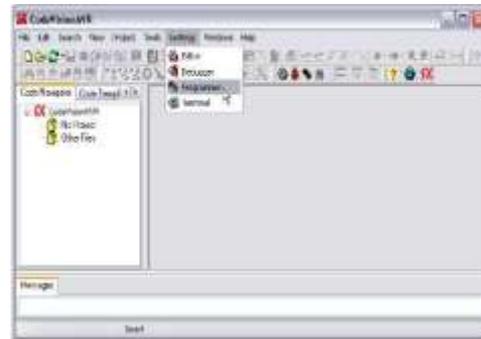
referensi yang digunakan. Dengan mencentang ADC Enabled akan mengaktifkan on-chip ADC. Dan dengan mencentang Use 8 bits, maka hanya 8 bit terpenting yang digunakan. Hasil konversi 10 bit dapat dibaca pada ADC Data Registers ADCH dan ADCL. Misalnya, jika hasil konversi ADC bernilai 54(36H), dalam 10 bit biner ditulis dengan 00 0011 0110B. Jika dalam format right adjusted (ADLAR=0), maka I/O register ADCH berisi 0000 0000B(00H) dan I/O register ADCL berisi 0011 0110B (36H).

## 2.2 Kajian Penelitian

Pemrograman atmel AVR ada 2 cara untuk memprogram mikrokontroller ini, menggunakan software AVR assembler yang berbasis pada bahasa assembly, dan menggunakan software CV AVR (Code Vision AVR) yang berbasis pada bahasa C. Pada penelitian ini akan digunakan cara yang kedua dengan pertimbangan kemudahan pembuatan program dari algoritma yang telah dibangun. Penelitian ini tidak menitikberatkan penggunaan bahasa C pada CV AVR, tapi lebih pada cara dan aplikasi dari mikrokontroller.

Dalam bab ini juga akan dibahas masalah fasilitas eksternal interrupt mikrokontroller INT0 dan INT1. Fasilitas ini sangat penting karena menempati urutan kedua dan ketiga setelah RESET. Rekeyasa Prototipe Robot.....

Register register yang perlu disetting untuk menggunakan fasilitas interrupt adalah MCUCR, MCUSR, GICR, dan GIFR. Berikut tampilan CV AVR



**Gambar 2.6.** Tampilan CVAVR

### 2.2.1 Dioda Emisi Cahaya (LED : *Light Emitting Diode*)

Dioda emisi cahaya atau dikenal dengan singkatan LED merupakan Solid State Lamp yang merupakan piranti elektronik gabungan antara elektronik dengan optik, sehingga dikategorikan pada keluarga “*Optoelectronic*”. Sedangkan elektroda-elektrodanya sama seperti dioda lainnya, yaitu anoda (+) dan Katoda (-). Ada tiga kategori umum penggunaan LED, yaitu :

- a. Sebagai lampu indikator,
- b. Untuk transmisi sinyal cahaya
- c. Sebagai penggandeng rangkaian elektronik yang terisolir secara total.

Bahan dasar yang digunakan dalam pembuatan LED adalah bahan *Galium Arsenida* (GaAs) atau *Galium Arsenida Phospida* (GaAsP) atau juga *Galium Phospida* (GaP), bahan-bahan ini

memancarkan cahaya dengan warna yang berbeda-beda. Bahan GaAs memancarkan cahaya infra-merah, Bahan GaAsP memancarkan cahaya merah atau kuning, sedangkan bahan GaP memancarkan cahaya merah atau hijau. Seperti halnya piranti elektronik lainnya, LED mempunyai nilai besaran terbatas dimana tegangan majunya dibedakan atas jenis warna

**Tabel 2.5** Led dan teganganya

Warna	Tegangan Maju
Merah	1.8 volt
Orange	2.0 volt
Kuning	2.1 volt
Hijau	2.2 volt

Sedangkan besar arus maju suatu LED standard adalah sekitar 20 mA. Karena dapat mengeluarkan cahaya, maka pengujian LED ini mudah, cukup dengan menggabungkan dengan sumber tegangan dc kecil saja atau dengan ohmmeter dengan polaritas yang sesuai dengan elektrodanya

### 2.2.2. Dioda Foto ( *Photo-Diode* )

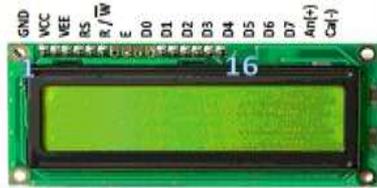
Dioda foto ini bekerja pada daerah reverse, jadi hanya arus bocor saja yang melewatinya. Dalam keadaan gelap, arus yang mengalir sekitar 10 A untuk dioda cahaya dengan bahan dasar germanium dan 1A untuk bahan silikon. Kuat cahaya

dan temperature keliling dapat menaikkan arus bocor tersebut karena dapat mengubah nilai resistansinya dimana semakin kuat cahaya yang menyinari semakin kecil nilai resistansi dioda cahaya tersebut. Penggunaan dioda cahaya diantaranya adalah sebagai sensor dalam pembacaan pita data berlubang (*Punch Tape*), dimana pita berlubang tersebut terletak diantara sumber cahaya dan dioda cahaya. Jika setiap lubang pita itu melewati antara tadi, maka cahaya yang memasuki lubang tersebut akan diterima oleh dioda cahaya dan diubah dalam bentuk signal listrik. Sedangkan penggunaan lainnya adalah dalam alat pengukur kuat cahaya (*Lux-Meter*), dimana dalam keadaan gelap resistansi dioda cahaya ini tinggi sedangkan jika disinari cahaya akan berubah rendah. Selain itu banyak juga dioda cahaya ini digunakan sebagai sensor sistem pengaman (*security*) misal dalam penggunaan alarm.

### 2.2.3. LCD (*Liquid Crystal Display*)

LCD merupakan perangkat keluaran yang digunakan dalam dunia elektrik saat ini, baik itu monitor komputer, *televisi*, ponsel dan lainnya. LCD di buat dalam bentuk berbagai macam dari layar lebar sampai layar yang hanya kecil, dalam praktek dalam robotik biasanya menggunakan LCD 16x2 atau 20x4 yang

maksudnya adalah 16 kolom dan 2 baris dan 20 kolom 4 baris.



**Gambar 2.7** LCD 16x2

#### 2.2.4. Sensor Jarak (*ultrasonic*)

Sensor ultrasonik adalah alat elektronika yang kemampuannya bisa mengubah dari energy listrik menjadi energy mekanik dalam bentuk gelombang suara ultrasonic. Sensor ini terdiri dari rangkaian pemancar ultrasonic yang dinamakan *transmitter* dan penerima ultrasonic yang disebut *receiver*. Alat ini digunakan untuk mengukur gelombang ultrasonic. Gelombang ultrasonic adalah gelombang mekanik yang memiliki ciri-ciri longitudinal dan biasanya memiliki frekuensi di atas 20 Khz. Gelombang Ultrasonic dapat merambat melalui zat padat, cair maupun gas. Gelombang Ultrasonic adalah gelombang rambatan energi dan momentum mekanik sehingga merambat melalui ketiga element tersebut sebagai interaksi dengan molekul dan sifat enersia medium yang dilaluinya.



**Gambar 2.8.** Sensor ultrasonik

Ada beberapa penjelasan mengenai gelombang ultrasonik. Sifat dari gelombang ultrasonik yang melalui medium menyebabkan getaran partikel dengan medium aplitudo sama dengan arah rambat longitudinal sehingga menghasilkan partikel medium yang membentuk suatu rapatan atau biasa disebut Strain dan tegangan yang biasa disebut Strees. Proses lanjut yang menyebabkan terjadinya rapatan dan regangan di dalam medium disebabkan oleh getaran partikel secara periodic selama gelombang ultrasonic lainnya.

#### 2.2.5. FET IRF540 (*Transistor efek medan*)

Transistor efek medan (*field-effect transistor* = FET) mempunyai fungsi yang hampir sama dengan transistor bipolar. Meskipun demikian antara FET dan transistor bipolar terdapat beberapa perbedaan yang mendasar. Perbedaan utama antara kedua jenis transistor tersebut adalah bahwa dalam transistor bipolar arus *output* (IC) dikendalikan oleh arus input (IB). Sedangkan dalam FET arus output (ID) dikendalikan oleh tegangan input (VGS), karena arus input adalah nol. Sehingga *resistansi input* FET sangat besar, dalam orde puluhan megaohm.

Disamping itu, FET lebih stabil terhadap temperatur dan konstruksinya

lebih kecil serta pembuatannya lebih mudah dari *transistor bipolar*, sehingga amat bermanfaat untuk pembuatan keping rangkaian terpadu

### 3. METODOLOGI PENELITIAN

#### Jalan Rancang Bangun Robot

Dalam proses ini sebelum dilakukan pembuatan alat mekanik penulis melakukan proses pembuatan elektroniknya terlebih dahulu, berikut adalah proses yang dilakukan.

- A. Identifikasi blok diagram fungsional robot
- B. Pembuatan blok diagram rangkaian
  1. Rangkaian minimum system atmega 8535 sebagai rangkain kontrol atau rangkaian utama robot.
  2. Rangkaian piranti masukan
    - a) Foto diode sebagai sensor line
    - b) SRF04 sebagai alat nafigasi robot
    - c) LM35 sebagai sensor suhu
    - d) Foto dioda sebagai sensor cahaya api
    - e) *Switch* sebagai *keypad* untuk pemrograman sederhana
  3. Rangkaian piranti keluaran
    - a) LCD 16x2 sebagai alat monitoring progress kerja robot
    - b) H-Bridge sebagai penguat daya Motor dc untuk alat penggerak robot.
    - c) Fan dan drivernya sebagai alat simulasi pemadaman api (lilin)

- C. Perakitan kerangka robot
  1. Pengelompokan fungsi komponen, diantaranya LCD, *switch*, motor dc dan lain sebagainya
  2. Penempatan komponen sehingga tepat guna
  3. Pembuatan PCB
- D. Pembuatan diagram alir pemrograman
- E. Pembuatan code program robot
- F. Memasukkan code program ke chip mikrokontroler
- G. Uji coba robot

### 4. HASIL DAN PEMBAHASAN

#### Hasil

Dari hasil penelitian penulis yang telah lakukan bahwa pembuatan blok diagram rangkaian robot akan mempermudah dalam identifikasi masalah dan perbaikanya, serta memungkinkan untuk pengembangan pada bagian *input* maupun pada bagian *outputnya*.

#### Desain rangkaian

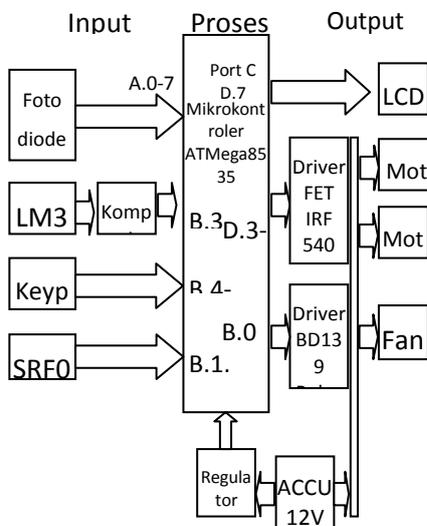
- A. Desain *hardware* robot

Cara operasional robot, dengan cara robot diletakkan diatas jalur hitam dengan warna dasar putih dengan ukuran jalur 2 centimeter, hidupkan saklar 1 ke on pada sisi kiri robot untuk menghidupkan rangkaian mikrokontroler yang kemudian robot dalam kondisi

standby, hidupkan pula saklar 2 pada sisi kanan robot untuk menghidupkan rangkaian H-bridge atau motor dc. Apabila di tekan tombol OK yang berada di atas sisi kanan depan maka program robot akan berjalan mengikuti jalur, dan apabila di tekan tombol cancel yang berada diatas sisi kiri depan maka LCD akan menampilkan menu yang memungkinkan dapat merubah program sederhana pada robot dengan bantuan dua switch yang berada di sebelah tombol cancel untuk pilihan turun (*down*) dan disebelah tombol OK untuk pilihan naik (*up*).

**B. Desain rangkaian robot**

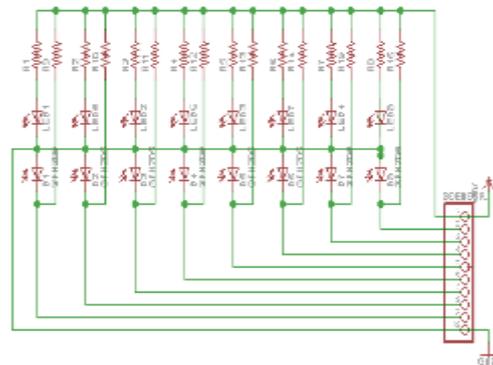
Dalam pembuatan robot penulis membagi dalam tiga kelompok yaitu *input*, proses dan output sesuai dengan blok diagram berikut



**Gambar 4.1** Diagram blok rangkaian

**C. Penjelasan rangkaian**

**1. Sensor foto dioda, SRF04, LM35**



**Gambar 4.2.** Rangkaian sensor foto dioda

Bagian sensor foto dioda, bagian ini terdiri dari foto dioda sebagai sensor cahaya, LED sebagai sumber cahaya untuk sensor foto dioda, cahaya yang dipancarkan mengenai warna hitam yang berarti akan redup, karena warna hitam bersifat menyerap cahaya dan apabila terkena benda warna putih maka cahaya akan dipantulkan yang berarti cahaya sangat kuat dengan adanya dua kondisi tersebut maka akan dikonversi oleh foto dioda dan mikrokontroler menjadi keadaan nilai 0 dan 1. Diperlukannya LED adalah karena dibutuhkan sumber cahaya yang stabil. Rangkaian ini dilengkapi dengan resistor yang digunakan sebagai pembatas maupun pembagi tegangan untuk pengaman LED serta foto dioda.

Bagian Sensor SRF04, sensor memancarkan sinyal ultrasonic dari trigger yang dipicu dari mikrokontroler

kemudian dipancarkan dan bila terkena benda sinyal dipantulkan kembali kemudian diterima SRF04 lagi, diolah oleh mikro berapa lama waktu tempuh sinyal tadi yang kemudian dikonversi menjadi jarak.

Bagian sensor LM35, sensor ini sensitive terhadap perubahan suhu sehingga mampu mendeteksi suhu kamar, dan mengeluarkan tegangan output 0 hingga 5 volt yang kemudian dikonversi menjadi nilai suhu 0-100 derajat celcius ( $^{\circ}\text{C}$ )

## 2. Keypad /Switch

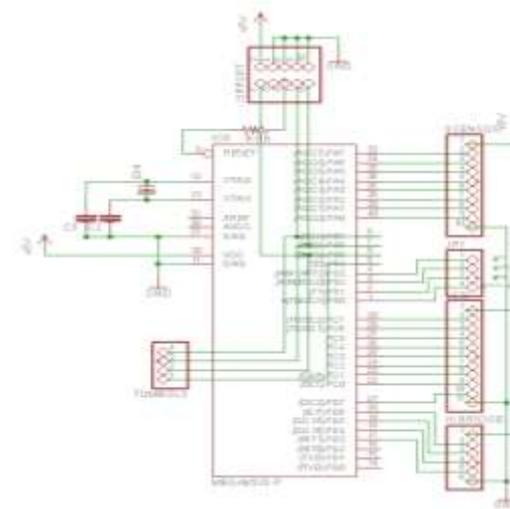
Terdiri dari empat buah *switch* yang dikombinasikan menjadi *keypad*, yang berfungsi sebagai *keypad* yang akan digunakan sebagai tombol satu untuk tombol OK (*start*) menjalankan robot atau menjalankan program, tombol dua sebagai tombol menu (*cancel*) untuk mengubah program robot secara sederhana yang telah disajikan pada menu utama robot, tombol tiga dan empat sebagai tombol (*down*, *up*) sebagai tombol pilihan menggeser ke bawah atau ke atas saat *mensetting* menu program yang telah di simpan pada IC ATmega8535 pada mikrokontroler. Diantaranya adalah kecepatan motor, setting PID, cek motor, cek sensor, melihat nilai ADC pembacaan sensor foto

dioda dan lainnya yang tertera pada program menu.

## 3. Minimum system mikrokontroler ATmega8535

Dalam blok ini, penulis membuat desain rangkaian minimum yang terdiri dari:

- IC ATmega8535 untuk menyimpan program robot
- X-tal 12 M.Hz dan 2 buah kapasitor keramik sebagai pembangkit *frekuensi clock*
- Resistor, sebagai pengaman LED
- LED sebagai indicator catu daya robot



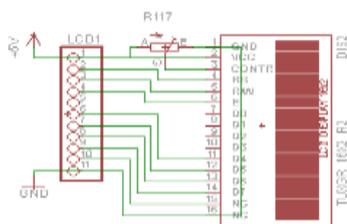
**Gambar 4.3.** Rangkaian *minimum system mikrokontroler*

- IC 7805, dioda, kapasitor elektrolit sebagai penstabil tegangan 5 volt DC sebagai sumber catu daya microprosesor,
- Switch digital* sebagai tombo *reset*

g) 2 buah *switch* untuk *on-off* vcc mikrokontroler dan *switch H-bridge* sebagai *actuator* robot, dan dilengkapi dengan *socket I/O* yang nantinya akan dihubungkan ke bagian yang lain.

#### 4. LCD

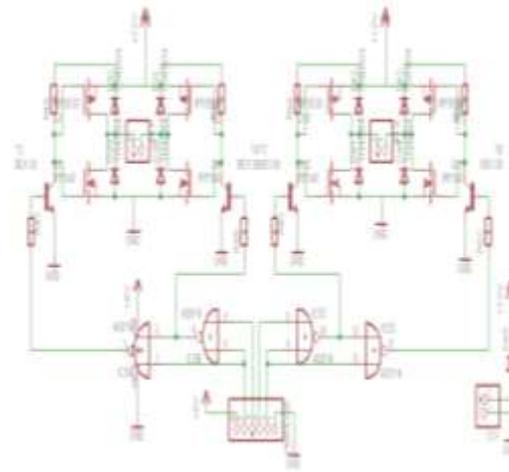
Bagian ini terdiri dari LCD 16x2 yang akan digunakan untuk memonitor program, dan dilengkapi dengan empat (4) buah tombol digital (*OK, Cancel, up, Down*), serta *variable resistor* sebagai pengatur pencahayaan latar pada LCD terang atau gelap



**Gambar 4.4.** Rangkaian LCD

#### 5. Motor (*driver H-bridge*)

Pada bagian ini terdiri dari rangkaian *driver* sebagai penguat daya. Rangkaian ini terdiri dari FET dan transistor sebagai penguat dayanya agar dapat memutar motor dengan daya besar namun menghemat baterai atau arus baterai tidak cepat habis dan tetap bisa menggunakan baterai dibawah 12 volt DC dengan kecepatan tinggi.



**Gambar 4.5.** Rangkaian *H-Bridge*

Rangkaian *H-bridge* berfungsi untuk *driver* motor dc yang terdiri dari dua buah motor dc, gear, dan roda untuk menjalankan robot sebagai mesin penggerak.

#### 4.1.2 Pemrograman

Setelah penulis buat rangka fisik robot maka langkah yang selanjutnya adalah pembuatan program. Dalam pemrograman merupakan bagian inti dari pembuatan tugas akhir ini, yang akan penulis bahas tahap demi tahap

##### A. Langkah-langkah pemrograman

Perangkat lunak yang digunakan pada tugas akhir ini ditulis pada teks editor dalam bahasa C dan *software* yang digunakan adalah *Code Vision Avr*. setelah program selesai dibuat, kemudian program disimpan dengan nama file yang berekstensi \*.c. Selanjutnya program yang telah selesai dibuat tadi dilakukan kompilasi dari \*.c ke dalam kode-kode instruksi mesin yang

sesuai dengan up-code mikrokontroler Intel dengan ekstensi \*.hex, yang kemudian di flash ke chip mikrokontroler.

Dalam pembuatan tugas akhir ini pemrograman merupakan salah satu langkah yang sangat penting sebab bekerja atau tidaknya robot bergantung dari bagaimana memprogram robot, tanpa program robot tidak akan berfungsi sesuai yang direncanakan. Langkah-langkah pembuatan program tersebut yang penulis sajikan adalah sebagai berikut :

#### 1. Pembuatan diagram alir

Dalam menyusun diagram alir diusahakan dapat membagi proses yang kompleks menjadi sub program yang lebih kecil, sehingga pencarian kesalahan akan lebih mudah. Selain itu akan memudahkan orang lain dalam membaca alir program yang dibuat.

#### 2. Pembuatan program

Penulisan program dilaksanakan setelah diagram alir selesai dirancang. Pemilihan editor teks disesuaikan dengan kebiasaan dan kesenangan. Agar teks yang telah dibuat nantinya dapat dimengerti oleh program AVR prog maka hasil penulisan program harus dibuat dengan ekstension \*.c dan dirubah menjadi bahasa mesin dengan ekstensi \*.hex.

#### 3. Kompilasi program

Program yang ditulis menggunakan editor teks kemudian dikompilasi dengan menggunakan program *Code vision AVR*. Bila tidak ada peringatan error atau kesalahan, proses kompilasi telah berhasil. Bila ada pesan kesalahan, dapat dicari kesalahan yang terjadi berdasarkan informasi pesan kesalahan tersebut.

#### 4. Pengisian program

Setelah Program di buat dan tidak ada kesalahan, proses pengisian program ke chip IC bisa dilaksanakan

#### B. *Code Vision AVR*

Setiap akan membuat program baru pada *Code Vision AVR*, kita akan diarahkan untuk membuat project terlebih dahulu. Pada menu File pilih New lalu klik ceklist Project dan klik OK.

Maka akan tampilah jendela konfirmasi setelah dipilih ok, dan apabila kita memilih source maka akan ditampilkan arah untuk membuka program yang sudah pernah dibuat. *Code Vision AVR* akan menanyakan apakah anda akan menggunakan fasilitas *Code Wizard AVR* ataukah tidak untuk suatu program baru dan memudahkan nantinya dalam pembuatan program maupun saat program setelah jadi dan akan digunakan agar tidak terjadi kekeliruan.

Fasilitas ini sangatlah berguna jika anda tidak ingin bersusah payah

melakukan settingan dengan mengetik code-code tertentu yang banyak dan kompleks. Intinya terdapat tab-tab yang tinggal klik dan anda akan langsung disajikan contoh kodenya.

Contohnya jika pembaca ingin membuat program yang memakai eksternal interrupt maka tinggal pilih tab *External IRQ*, pada tampilan *Code Wizard*. Klik pada *INT 0 Enabled* untuk memilih *external interrupt 0* dan pilih mode pembangkitan *interrupt* yang ingin digunakan. Mode pembangkitan *eksternal interrupt* terdiri dari *rising edge* atau *falling edge* atau *any change*. Untuk melihat code yang akan dihasilkan klik pada menu File dan pilih Program *Preview*.

Pada tab chip pilih ATmega8535 dan *clock* sesuai dengan *clock* yang terpasang 12 MHz atau 16 MHz, dan pada gambar 4.13. atau tab *timers* pilih *timer1 Clock source* pilih *System Clock*, kemudian pada *Clock Value* pilih 187.500 kHz, pilihan pada mode adalah *Fast PWM top=00FFh*, sedangkan pada *output* baik *out A* maupun *Out B* pilih *Inverted*.

Kemudian yang berikutnya adalah tab LCD, pada *LCD Port* pilih PORTC sebagai tampilan LCD sesuai dengan gambar 4.14, dan pada *Chars/Line* pilih 16 karena menggunakan LCD 2x16. setelah port LCD dipilih maka akan Reayasa Prototipe Robot.....

langsung ditampilkan pin-pin LCD yang akan digunakan atau terhubung dengan port pada mikrokontroler seperti pada gambar 4.15

Setingan pada tab ADC adalah seperti pada gambar 4.16. *ADC enable* di ceklis dan bila menggunakan 8 bit maka use 8 bit diceklis, pada *volt ref* pilih *AVCC pin*, *clock* pilih 750.000 kHz. Setelah semua telah dilakukan seting sesuai yang diinginkan klik *File < Generate, Save and Exit*. Kemudian lakukan penyimpanan seperti biasa sebanyak tiga kali dengan nama file sama namun berbeda ekstensi. Ketik nama project yang akan dibuat dan klik *save*. File code yang memiliki ekstensi \*.c yang akan dibuat nantinya disave pada suatu tempat dan dimasukkan ke dalam project dengan mengklik Add pada tab File seperti diatas. Pada tab *C Compiler* akan ditemukan tab lagi didalamnya. Tab yang sering digunakan ialah tab *Code Generation*.

Proses coding menggunakan bahasa C belum dapat dimulai setelah mengkonfigurasi project yang dibuat. Setelah selesai membuat code simpan file. Untuk memasukkan file \*.c yang telah disave ke dalam project klik menu *Project* dan pilih *Configure*.

Kita akan dihadapkan kembali pada *windows* seperti pada *Configure Project*

diatas. Klik tab *after make* dan ceklist *Program the chip*, kemudian OK. Yang kemudian klik *compile* dan perhatikan pastikan tidak ada yang *error*, seperti tampilan pada gambar 4.26. Namun jika terjadi *error* untuk mengetahui letak kesalahan pada code, tinggal klik pada tab *navigator* bagian *Errors* atau dapat mengklik pada tab *Messages* untuk mengetahui letak kesalahan pada code. Pada contoh berikut dapat dilihat *sintaks #include <mega8535.h* akan diblok dengan warna biru yang menunjukkan kesalahan terletak pada sintaks tersebut. Setelah klik make maka ikuti prosesnya selain melakukan *compiling* akan *generate* beberapa file termasuk \*.hex yang akan diburn ke mikrokontroler dan file \*.coff yang dapat disimulasikan pada AVR studio.

Setelah proses *compiling* berhasil maka akan muncul proses pengisian ke mikrokontroler, dan apabila gagal berarti komunikasi port nya tidak sesuai, maka sesuaikan dan ulangi untuk memasukkan lagi.

#### 4.2 Pembahasan dan Pengujian Robot

Setelah program dibuat dan di *generate* ke mikrokontroler maka tinggal mencoba pada robot, apakah code program yang dibuat bekerja sesuai yang kita harapkan atau tidak, agar perbaikan dan penulisan ulang program dapat

mudah dilakukan dengan cara pengujian satu persatu.

##### A. Pengujian LCD

Salah satu bagian penting dari robot ini adalah LCD, dengan kegunaan untuk menampilkan progress kerja dari robot, agar LCD dapat berfungsi

##### B. Pengujian 2 (sensor jalur atau garis)

Pada pengujian sensor garis karena menggunakan sensor foto dioda dan menggunakan delapan sensor dengan proses masukan berdasarkan cahaya yang masuk maka dipastikan led sebagai sumber cahaya mandiri dengan menutup semua sisi-sisinya agar cahaya dari luar baik matahari ataupun cahaya penerangan tidak mengganggu.

Bekerja dengan pantulan dari jalur yang ada apabila mengenai warna hitam maka cahaya yang masuk ke sensor kecil dan apabila mengenai warna putih maka akan dipantulkan cahaya tersebut sangat kuat. Dari perbedaan tersebut maka di umpankan ke mikrokontroler dengan nilai digital 0 dan 1.

##### C. Pengujian 3 (sensor jarak / *ultrasonic*)

Pada pengujian ini penulis menggunakan sensor SRF04, yang dipasang pada bagian depan robot sebagai alat navigasi robot yang akan memberitahukan bahwa didepan ada benda, sensor ini dapat mendeteksi jarak

antara 3 cm sampai 400 cm, dan berikut cara membuat code program sensor *ultrasonic* sebagai sensor jarak

#### D. Pengujian 4 (sensor suhu)

Pada pengujian sensor suhu penulis menggunakan IC LM35 sebagai sensor suhunya dan menggunakan rangkaian komparator sebagai penguat tegangan *output ICPengujian 5 (motor dc)*

Dalam pengujian ini penulismenggunakan 2 buah motor dc dengan penguat daya menggunakan komponen FET dengan seting PWM Pengaturan kecepatan dengan H-Bridge

Kecepatan motor dapat dilakukan dengan penambahan komponen pada rangkaian *H-Bridge* dengan memberi komponen *common* FET sehingga dapat menghasilkan daya yang besar meskipun hanya dengan catu sumber 12 volt. Pada rangkaian ini sebagai kunci utama kecepatan motor dengan pengaturan kecepatan motor menggunakan system PWM yang telah dibahas pada BAB. Metodologi Penulisan.

Motor DC sebagai *actuator* robot sehingga robot bisa bergerak maju, mundur, belok kanan, atau belok kiri

#### E. Pengujian 6 (FAN)

Dengan hanya menambakkah definisi fan untuk memudahkan penulisanya yang berarti cukup PORTB.3 sebagai output fan sudah dapat hidup saat Rekayasa Prototipe Robot.....

PORTB.3 di beri nilai 1, nilai 1 ini bisa didapatkan dari masukan sensor cahaya lilin atau api, maka sensor saat bernilai tinggi maka akan akan menhidupkan fan yang terlebih dahulu di beri transistor 2SC828 dan BD139 untuk menggerakkan *relay*.

#### F. Pengujian menyeluruh

Robot yang telah jadi lengkap dengan sensor jalur, jarak dan suhu di program agar dapat bekerja komponen tersebut dengan tujuan robot dapat mematikan api yang di letakkan pada ujung perjalanan robot. Agar robot dapat berjalan dengan sempurna maka perlu penambahan program seperti kontrol PID, dan menu sederhana untuk memudahkan perubahan program pada robot dengan cepat

## 5. PENUTUP

Berdasarkan hasil penelitian dan pengambilan data yang diperoleh, maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Rancangan prototipe perangkat keras robot pemadam api dan pelacak jejak dapat berfungsi secara optimal dalam memadamkan titik api
2. Sistem operasi *software robot* pemadam api dan pelacak jejak dapat berjalan sesuai fungsinya dan dapat mematikan api dalam durasi kurang

dari 10 detik, serta kemampuan jelajah robot berkecepatan 0,2 meter per detik

Zaks Rodnay. 1988. Teknik Perantaraan Mikroprosesor Edisi 3. Jakarta : Erlangga.

## 6. DAFTAR PUSTAKA

- Andrianto Heri. 2008. Pemrograman Mikrokontroler Avr ATMega16 Menggunakan Bahasa C (Codevision Avr). Bandung : Informatika.
- Arifianto. B. Modul Training Microcontroller For Beginner.
- FTI. Unissula. Panduan Praktikum Sistem Mikroprosesor. Semarang.
- Garland Harry. 1984. Pengantar Desain System Mikroprosesor. Jakarta : Erlangga.
- Malik Moh Ibnu, dkk. 1997. Bereksperimen Dengan Mikrokontroler 8031, Jakarta : Elex Media Komputindo.
- Setiawan Iwan. 2006. Tutorial Microcontroller AVR Part 1. Undip : Elektro
- Soebhakti Hendrawan. 2007. Basic Avr Microcontroller Tutorial ATMega 8535. Batam : Politeknik Batam.
- Wardana Lingga. 2006. Belajar Sendiri Mikrokontroler AVR Seri ATMega16, Simulasi Hardware dan Aplikasi. Jogjakarta : Penerbit Andi.