

BAB II

DASAR TEORI

2.1 Mikropengendali AVR

Kata AVR merupakan singkatan dari *Alf and Vegard RISC (Reduced Instruction Set Computer)* sesuai dengan nama penggagasnya. Mikropengendali AVR yang menggunakan teknologi RISC dan menggunakan arsitektur Harvard ini pertama kali dikembangkan pada tahun 1996 oleh dua orang mahasiswa Norwegian Institute of Technology yaitu Alf Egil Bogen dan Vegard Wollan yang kemudian dikembangkan lebih lanjut oleh perusahaan Atmel. Seri pertama mikropengendali AVR yang dikeluarkan adalah mikropengendali 8 bit dengan nama AT90S8515 dengan konfigurasi pin yang sama dengan mikropengendali 8051, termasuk bus alamat dan bus data yang termultipleks. Mikropengendali AVR mempunyai set instruksi yang lebih sedikit dan mode pengalamatannya yang juga sederhana. Dalam AVR RISC 8 bit, semua instruksi berukuran 16 bit dan sebagian besar dieksekusi dalam satu siklus *clock* kecuali instruksi percabangan yang membutuhkan dua siklus *clock*. Berbeda dengan mikropengendali MCS dengan menggunakan teknologi *Complex Instruction Set Computing (CISC)* yang instruksinya bervariasi antara 8 bit sampai 32 bit dan dieksekusi selama satu sampai empat siklus mesin, dimana satu siklus mesin membutuhkan 12 periode *clock*. Beberapa fitur yang dimiliki mikropengendali dengan arsitektur RISC diantaranya adalah instruksi yang sederhana, set instruksi yang sedikit, panjang instruksi yang sama untuk semua instruksi, register untuk penyimpanan *operand* yang jumlahnya besar, mempunyai arsitektur *load/store*, dan eksekusi instruksi yang lebih cepat sehingga memberikan kecepatan siklus intruksi rata-rata satu *clock* per instruksi. Dalam perkembangannya, AVR dibagi menjadi enam kelas yaitu kelompok ATtiny, AT90Sxx, ATmega, AT90CAN, AT90PWM, dan AT86RFxx. Pada dasarnya yang membedakan masing-masing kelas adalah kapasitas memori, peripheral, fungsinya dan beberapa fitur tambahannya. Sedangkan dari segi arsitektur dan set instruksi yang digunakan hampir sama. Pengelompokan jenis

mikropengendali AVR sesuai varian dan tipe mikropengendalinya ditunjukkan oleh tabel 2.1 [1].

Tabel 2.1 Jenis Varian Mikrokontroler AVR

No	Varian	Tipe Mikropengendali
1	ATtiny	ATTINY10, ATTINY11, ATTINY12, ATTINY13, ATTINY15, ATTINY2313, ATTINY24, ATTINY25, ATTINY261, ATTINY44, ATTINY 45, ATTINY461, ATTINY 48, ATTINY48_32PIN, ATTINY84, ATTINY85, ATTINY861, ATTINY88, ATTINY88_32PIN.
2	AT90S	AT90S1200, AT90S2313, AT90S2323, AT90S2333, AT90S2343, AT90S4433, AT90S4434, AT90S8515, AT90S8535.
3	AT90USB	AT90USB646, AT90USB1286.
4	ATMega	ATMega103, ATMega128, ATMega1280, ATMega1281, ATMega1284P, ATMega16, ATMega162, ATMega164P, ATMega165, ATMega165P, ATMega168, ATMega168_32PIN, ATMega168P, ATMega168P_32PIN, ATMega169, ATMega169P, ATMega2560, ATMega2561, ATMega32, ATMega324P, ATMega325, ATMega325P, ATMega3250, ATMega3250P, ATMega328P, ATMega328P_32PIN, ATMega329, ATMega3290, ATMega3290P, ATMega48, ATMega48P, ATMega48P_32PIN, ATMega64, ATMega640, ATMega644, ATMega644P, ATMega645, ATMega6450, ATMega649, ATMega6490, ATMega8, ATMega8515, ATMega8535, ATMega88, ATMega88_32PIN, ATMega88P, ATMega88P_32PIN.

2.2 Mikropengendali ATMega8535

2.2.1 Arsitektur ATMega8535

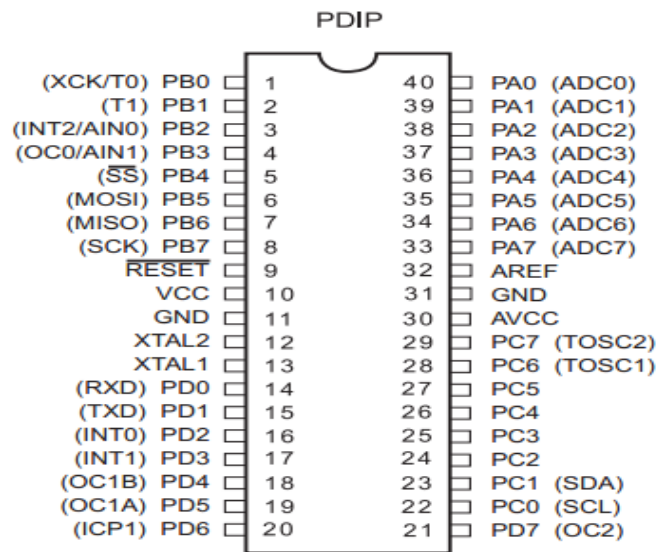
Mikropengendali ATMega8535 merupakan mikropengendali 8-bit teknologi CMOS dengan konsumsi daya rendah yang berbasis arsitektur enhanced RISC AVR. Dengan eksekusi instruksi yang sebagian besar hanya menggunakan satu siklus *clock*, ATMega8535 mencapai *throughput* sekitar 1 MIPS per MHz yang mengizinkan perancang sistem melakukan optimasi konsumsi daya rendah versus kecepatan pemrosesan. Prosesor AVR menggabungkan set instruksi yang kaya dengan 32 *register* umum (*General*

Purpose Register). Dari 32 *register* tersebut dikoneksikan langsung dengan *Arithmetic Logic Unit (ALU)*, mengizinkan dua *register* independen untuk diakses dalam satu instruksi yang dieksekusi dalam satu siklus *clock*. Arsitektur yang dihasilkan adalah arsitektur yang kode operasinya lebih efisien serta pencapaian *throughput*-nya hingga sepuluh kali lebih cepat daripada mikropengendali *Complex Instruction Set Computer (CISC)* konvensional. Beberapa fitur utama yang tersedia pada ATmega8535 diantaranya adalah :

1. *Port I/O* sebanyak 32 bit, yang dikelompokkan dalam *Port A*, *Port B*, *Port C*, dan *Port D*.
2. *Analog to Digital Converter* 10 bit sebanyak 8 *input*.
3. *Timer/Counter* sebanyak 3 buah.
4. CPU 8 bit yang terdiri dari 32 *register*.
5. *Watchdog Timer* dengan *oscillator internal*.
6. SRAM sebesar 512 byte.
7. Memori *Flash* sebesar 8 Kbyte dengan kemampuan *read while write*.
8. *Interrupt* internal maupun eksternal.
9. *Port* komunikasi SPI.
10. EEPROM sebesar 512 byte yang dapat diprogram saat operasi.
11. Analog komparator.
12. Komunikasi serial standar USART dengan kecepatan maksimal 2,5 Mbps.
13. Frekuensi *clock* maksimum 16 MHz [1-6].

2.2.2 Konfigurasi Pin ATmega8535

Mikropengendali ATmega8535 mempunyai pin sebanyak 40 dengan *port I/O* 32 bit yang dikelompokkan dalam empat *port* yaitu *Port A*, *Port B*, *Port C*, dan *Port D*. Masing-masing *port* tersebut harus diatur jika difungsikan sebagai *input* atau *output* karena *port-port* tersebut dapat digunakan untuk jalur *input* maupun *output*. Selain itu beberapa *port* pada ATmega8535 memiliki fungsi khusus. Konfigurasi Pin mikropengendali AVR ATmega8535 untuk 40 pin *Dual Inline Package (DIP)* ditunjukkan pada gambar 2.1 [1-6].



Gambar 2.1 Konfigurasi Pin ATmega8535

Untuk deskripsi kaki-kaki atau pin ATmega8535 ditunjukkan pada tabel 2.2.

Tabel 2.2 Deskripsi Pin ATmega8535

No.Pin	Nama Pin	Keterangan
1-8	Port B : PB0-PB7	<p>Port I/O dua arah dilengkapi internal pull-up resistor. Fungsi khusus :</p> <ul style="list-style-type: none"> • PB0 : To (<i>timer/counter0 external counter input</i>) • PB1 : T1 (<i>timer/counter1 external counter input</i>) • PB2 : AIN0 (<i>analog comparator positive input</i>) • PB3 : AIN1 (<i>analog comparator positive input</i>) • PB4 : SS (<i>SPI slave select input</i>) • PB5 : MOSI (<i>SPI bus master output/slave input</i>) • PB6 : MISO (<i>SPI bus master input/slave output</i>) • PB7 : SCK (<i>SPI bus serial clock</i>)
9	RESET	Memasukkan <i>reset</i> . Sebuah <i>reset</i> terjadi jika sebuah pin ini diberi logika <i>low</i> melebihi periode minimum yang diperlukan.
10	VCC	Catu Daya
11	GND	Ground

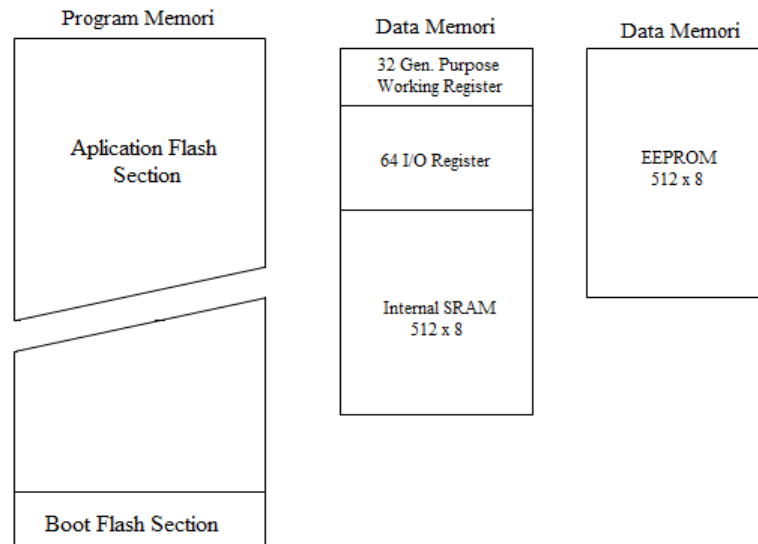
Tabel 2.2 Deskripsi Pin ATmega8535 (Lanjutan)

No.Pin	Nama Pin	Keterangan
12	XTAL1	Keluaran dari <i>inverting oscillator amplifier</i>
13	XTAL2	Masukan ke <i>inverting oscillator amplifier</i> dan masukan ke rangkaian <i>internal clock</i>
14-21	Port D : PD0-PD7	Port I/O dua arah dilengkapi <i>internal pull-up resistor</i> . Fungsi khusus: <ul style="list-style-type: none"> • PD0 : RXD (UART input line) • PD1 : TXD (UART output line) • PD2 : INT0 (<i>External interrupt 0 input</i>) • PD3 : INT1 (<i>External interrupt 1 input</i>) • PD4 : OC1B (<i>timer/counter 1 output compare B match output</i>) • PD5 : OC1A (<i>timer/counter 1 output compare A match output</i>) • PD6 : ICP (<i>timer/counter 1 input capture pin</i>) • PD7 : OC2 (<i>timer/counter 2 output compare match output</i>)
22-29	Port C : PC0-PC7	Port I/O dua arah dengan <i>internal pull-up resistor</i> . PC6 dan PC7 berfungsi sebagai <i>oscillator eksternal</i> untuk <i>timer/counter 2</i> .
30	AVCC	Catu daya untuk port A dan ADC
31	AGND	Analog Ground
32	AREF	Referensi masukan analog untuk ADC
40-33	Port A : PA0-PA7 (ADC0-ADC7)	Port I/O dua arah dilengkapi <i>internall pull-up resistor</i> . Port ini juga dimultipleks dengan masukan analog ke ADC 8 kanal

2.2.3 Peta Memori ATmega8535

Mikropengendali AVR ATmega8535 memiliki dua jenis memori yaitu memori data atau SRAM dan memori program atau memori *flash*. Disamping itu juga dilengkapi memori *Electrically Erasable Programmable Read Only Memory* (EEPROM) untuk penyimpanan data tambahan yang bersifat *non-volatile*. Memori EEPROM ini mempunyai lokasi yang terpisah dengan sistem

register alamat, *register* data dan *register* kontrol yang dibuat khusus untuk EEPROM. EEPROM juga dikenal juga dengan *Electrically Alterable ROM* (EAROM) yang dapat dihapus dan diprogram ulang sebanyak 10.000 kali. Untuk peta memori ATMega8535 ditunjukkan pada gambar 2.2 [1-6].



Gambar 2.2 Peta Memori ATMega8535

2.2.3.1 Memori Program

Mikropengendali ATMega8535 memiliki *On-Chip In-System Reprogrammable Flash Memory* untuk menyimpan program. Dalam penggunaannya memori ini umumnya digunakan untuk penyimpanan program secara internal. Untuk alasan keamanan, memori program dibagi menjadi dua bagian yaitu *Boot Flash Section* dan *Application Flash Section*. *Boot Flash Section* digunakan untuk menyimpan program *Boot Loader*, yaitu program yang harus dijalankan pada saat AVR *reset* atau pertama kali diaktifkan. *Application Flash Section* digunakan untuk menyimpan program aplikasi yang dibuat pengguna. Mikropengendali AVR tidak dapat menjalankan program aplikasi ini sebelum menjalankan program *Boot Loader*. Besarnya memori *Boot Flash Section* dapat diprogram dari 128 *word* sampai 1024 *word* tergantung pengaturan pada konfigurasi bit di-*register* *BOOTSZ*. Jika *Boot Loader*

diproteksi, maka program pada *Application Flash Section* juga sudah aman [1-6].

2.2.3.2 Memori Data

Untuk memori data pada mikropengendali ATmega8535 dibagi menjadi tiga yaitu :

1. Terdapat 32 register keperluan umum yaitu *General Purpose Register* (GPR) atau biasa disebut *register* file di dalam teknologi RISC.
2. Terdapat 64 *register* untuk keperluan *input/output* (*I/O Register*).
3. Terdapat 512 byte SRAM internal. Selain itu terdapat juga EEPROM 512 byte sebagai memori data yang dapat diprogram saat beroperasi [1-6].

2.2.4 EEPROM

Dalam mikropengendali AVR data dapat ditempatkan pada tiga macam memori yaitu memori *flash*, SRAM, dan *Electrically Erasable Programmable Read Only Memory* (EEPROM). EEPROM adalah salah satu memori untuk penyimpanan data internal mikropengendali yang sifatnya *non volatile*. Artinya data tidak akan hilang walaupun catu daya mikropengendali mati. Biasanya memori EEPROM diaplikasikan untuk penyimpanan tabel-tabel data atau konstanta dan penyimpanan *password*. ATmega 8535 mempunyai memori EEPROM 512 byte. EEPROM ini disusun sebagai ruang data yang terpisah dengan yang lain, dimana byte tunggal dapat dibaca dan ditulis. EEPROM diakses melalui *register-register* akses EEPROM yaitu EEPROM *Address Register* (EEAR), EEPROM *Data Register* (EEDR), dan EEPROM *Control Register* (EECR). Untuk divais-divais dengan EEPROM diatas 256 byte, EEAR sebenarnya ada dua *register* yaitu EEARL dan EEARH. EEAR digunakan untuk menentukan alamat EEPROM ke mana data akan ditulisi atau dari mana data akan dibaca. EEAR adalah sebuah *register* baca/tulis yaitu *register* yang dapat dibaca untuk melihat alamat apa saja yang akan digunakan. EEDR adalah *register* data EEPROM dan merupakan *register* baca/tulis. Jika ingin

menuliskan data ke EEPROM akan terjadi *load* data yang diperlukan ke dalam EEDR. Jika ingin membaca data dari EEPROM setelah proses pembacaan berakhir maka akan membaca EEDR untuk data. EECR mempunyai bit-bit kontrol yang diperlukan untuk pembacaan dan penulisan EEPROM. Penulisan ke suatu EEPROM tidak sesederhana seperti menulis ke SRAM. Waktu akses tulis untuk EEPROM pada mikropengendali AVR berkisar 2,5 sampai 4 ms, tergantung pada tegangan suplai [1-6].

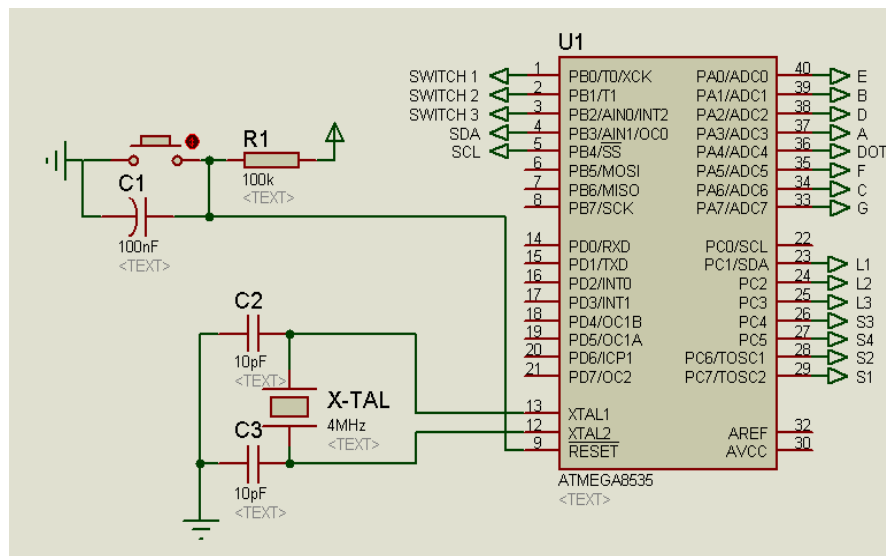
2.2.5 Sistem Minimum ATmega8535

Sistem minimum mikropengendali adalah rangkaian elektronika minimum yang diperlukan untuk dapat mengoperasikan IC mikropengendali. Sistem minimum ini kemudian bisa dihubungkan dengan rangkaian lain untuk menjalankan aplikasi tertentu. Mikropengendali seri ATmega8535 keluarga mikropengendali AVR, adalah salah satu seri yang banyak digunakan. Untuk membuat skematik sistem minimum Atmel AVR ATmega8535 diperlukan beberapa komponen diantaranya adalah :

1. IC mikropengendali ATmega8535.
2. Satu buah XTAL 4 MHz atau 8 MHz atau 12 MHz dan maksimum 16 MHz.
3. Dua buah kapasitor senilai 10 pF.
4. Dua buah kapasitor senilai 100 nF.
5. Satu buah resistor 100 KOhm
6. Satu buah tombol *reset push button*.
7. Header 2x5 yang berfungsi untuk menghubungkan rangkaian pada masing-masing *port* mikropengendali (alternatif).

Pada sistem minimum terdapat rangkaian *oscillator* yang merupakan rangkaian pembangkit frekuensi *clock* pada mikrokontroler. Rangkaian *oscillator* ATmega8535 terdiri dari beberapa komponen yaitu dua buah kapasitor senilai 10 pF dan sebuah kristal. Besar nilai kristal yang digunakan pada rangkaian ATmega8535 dalam tugas akhir ini adalah 4 Mhz. Skematik sistem minimum pada ATmega8535 tersebut juga sudah siap untuk menerima

sinyal analog dengan adanya fasilitas ADC pada *port A*. Untuk skematik sistem minimum ATmega8535 ditunjukkan pada gambar 2.3 [1-6].



Gambar 2.3 Rangkaian Sistem Minimum ATmega8535

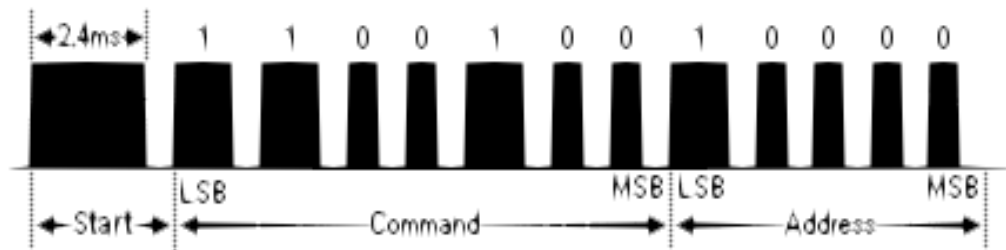
2.3 Sony Infrared Remote Control (SIRC)

2.3.1 Pembacaan SIRC

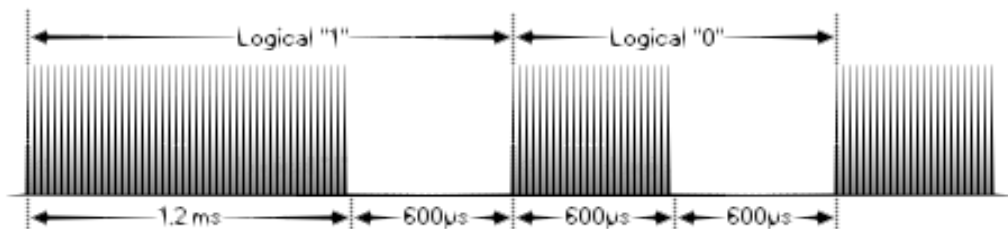
Di zaman sekarang, banyak peralatan elektronika yang dapat dikendalikan dari jarak jauh. Salah satu media yang digunakan adalah inframerah. Komunikasi dengan inframerah sering digunakan dalam aplikasi sehari-hari, misalnya *remote* Televisi (TV), *Air Conditioner* (AC), dan lain sebagainya. Pada pembacaan *remote control* data yang dikirimkan dimodulasi terlebih dahulu. Salah satu modulasi yang sering digunakan adalah *Pulse Width Modulation* (PWM). Data digital dimodulasi dengan frekuensi pembawa (*carrier*) yang umumnya berkisar antara 38-40 Khz. Pada dasarnya semua paket yang dikeluarkan *remote control* telah mengandung frekuensi *carrier* dan dimodulasi dengan menggunakan teknik PWM. Standar *remote control* yang sering dipakai adalah NEC dan *Sony TV Infrared Remote Control* (SIRC). Beberapa hal yang perlu dicatat dalam paket data Sony antara lain adalah [7]:

- 1) Sinyal pembawa berada di dalam bit-bit paket data dengan frekuensi 38-40 Khz.
- 2) Satu *frame* (satu paket data) terdiri atas sebuah starting bit, 12 bit data, sebuah *frame space* yang memisahkannya dengan *frame* berikutnya.

- 3) Satu *frame* membutuhkan 45 ms (mili detik).
- 4) *Starting* bit membutuhkan 2,4 ms. Untuk SIRC protocol ditunjukkan pada gambar 2.4.

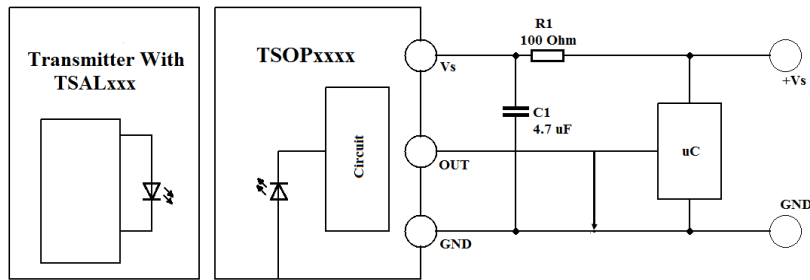
Gambar 2.4. SIRC Protocol^[7]

- 5) Data 1 diwakili oleh 1,2 ms ada pulsa dan 0,6 ms tanpa pulsa. Hal ini seperti yang ditunjukkan pada gambar 2.5.
- 6) Data 0 diwakili oleh 0,6 ms ada pulsa dan 0,6 ms tanpa pulsa. Hal ini seperti yang ditunjukkan pada gambar 2.5.

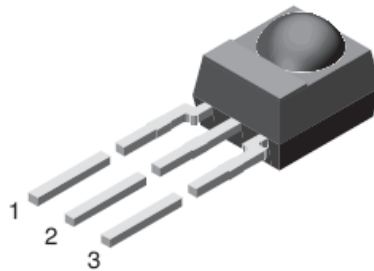
Gambar 2.5. SIRC Modulation^[7]

2.3.2 TSOP4838

. Untuk komponen penerima inframerah yang umumnya digunakan adalah TSOP48xx. Karena *Sony Infrared Remote Control* mempunyai sinyal pembawa antara 38-40 KHz maka sensor infra merah yang digunakan adalah TSOP4838. Sensor penerima infra merah tersebut dapat menerima frekuensi sinyal pembawa sebesar 38 KHz. Melalui sensor penerima infra merah TSOP4838 sinyal dapat langsung dimodulasikan ke mikropengendali sehingga data dapat langsung diproses dan dieksekusi. TSOP4838 memiliki tiga kaki yang berfungsi sebagai keluaran, *ground* dan tegangan *input*. Untuk keluarannya akan bekerja jika mendapat tegangan rendah. Komponen *receiver* dan susunan pinnya ditunjukkan pada gambar 2.6 sedangkan untuk bentuk fisiknya ditunjukkan pada gambar 2.7 [8].



Gambar 2.6 Receiver dan Susunan Pin



Gambar 2.7 Bentuk Fisik TSOP4838

2.4 Seven Segment

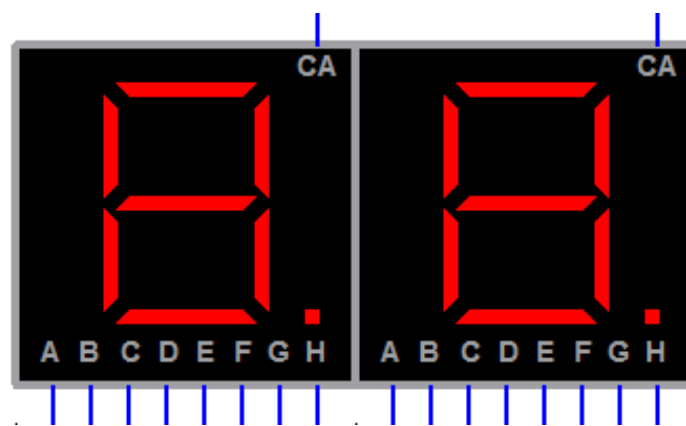
2.4.1 Operasi LED Seven Segment

Tampilan LED *Seven Segment* pada dasarnya dibuat dari tujuh buah dioda pemancar cahaya atau biasa disebut dengan *Light Emitting Diode* (LED) yang saling terpisah yang dikemas dalam satu paket. *Seven Segment* adalah komponen elektronika yang berfungsi untuk menampilkan data dalam bentuk 7 buah LED yang disusun membentuk *display*. Pada penggunaannya *Seven Segment* biasanya digunakan untuk membentuk digit angka dari nol sampai sembilan serta juga bisa merepresentasikan karakter a, b, c, d, e dan f. *Seven segment* terdiri dari dua jenis yaitu *common anode* (CA) dan *common cathode* (CC) [1,9].

2.4.2 Seven Segment Common Anode

Tampilan LED *Seven Segment* mempunyai dua macam konfigurasi yaitu *common anode* dan *common cathode*. Pada konfigurasi *common anode* ketujuh pin anoda dari LED dihubungkan secara bersama (pada simpul yang sama) ke suplai tegangan sebesar 5 volt. Agar LED dapat memberikan cahaya, maka katoda harus ditanahkan (*grounded*) melalui hubungan pembatas tahanan seri. Setiap *segment* pada unit tampilan mendapatkan pencahayaan dengan cara yang

sama. Tampilan *common anode* adalah aktif *low* (*low enable*) sebab LED mendapatkan tegangan *low* untuk pencahayaan setiap segmen. Karena itu IC *decoder* yang digunakan untuk memberikan *drive* pada LED harus mempunyai keluaran yang aktif *low*. Dengan konfigurasi tersebut maka LED *Seven Segment* harus diberi tegangan catu bersama sebesar 5 volt, sehingga segmen akan menyala jika diberi nilai nol (*low*) dan padam jika diberi nilai satu (*high*). Untuk dapat menampilkan sebuah karakter pada LED *Seven Segment*, maka harus diatur agar hanya segmen tertentu saja yang menyala sesuai dengan bentuk tampilan. Tampilan *Seven Segment common anode* dapat ditunjukkan pada gambar 2.8 [1,9].



Gambar 2.8. *Seven Segment Common Anode*

2.4.3 Teknik *Scanning Display* Pada *Seven Segment*

Untuk menampilkan angka yang berderet lebih dari satu digit, misalnya empat digit maka digunakan *port* sebanyak 32 bit. Tetapi cara ini tentunya tidak efisien karena menghabiskan persediaan *port* yang ada pada mikropengendali ATmega8535. Pemborosan penggunaan *port* seperti ini jelas kurang menguntungkan. Untuk mengatasi pemborosan ini maka dapat digunakan teknik *scanning display*, dimana setiap digit angka atau karakter yang akan ditampilkan dihubungkan secara paralel kemudian ditampilkan secara bergantian. Teknik *scanning* ini memanfaatkan kelemahan atau keterbatasan mata manusia yang tidak dapat berakomodasi secara baik ketika dua atau lebih karakter ditampilkan secara bergantian tetapi dengan tundaan waktu (*delay*) yang sangat cepat sehingga seakan-akan karakter-karakter tersebut ditampilkan

secara simultan atau bersamaan. Biasanya teknik *scanning* atau teknik sapuan ini menggunakan *delay* dalam orde milidetik [1,9].

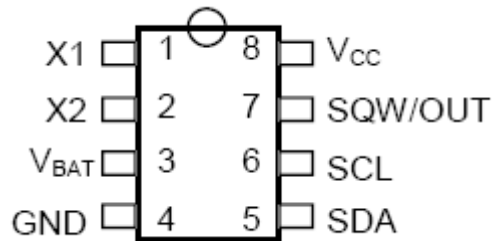
2.5 Real Time Clock (RTC) DS1307

DS1307 merupakan *Real Time Clock* berdaya rendah, dengan kode biner desimal (BCD) jam/kalender ditambah 56 byte NV SRAM. Alamat dan data ditransfer melalui serial bus melalui 2 kawat *bi-directional*. RTC menyediakan data dalam detik, menit, jam, hari, tanggal, bulan, tahun dan informasi mengenai waktu yang bisa diprogram. Penggunaan fungsi pewaktu dapat beroperasi baik dalam format 24 jam atau 12 jam dengan AM/PM indikator. DS1307 memiliki sirkuit *built-in power* dengan pengertian bahwa memiliki kemampuan untuk mendeteksi kegagalan dan otomatisasi beralih ke mode baterai. Fitur-fitur yang dimiliki oleh *Real Time Clock (RTC) DS1307* sesuai dengan yang ada pada *datasheet* diantaranya [10,11] :

- 1) *Real-time clock (RTC)* dapat menghitung detik, menit, jam, tanggal dalam bulan, hari dalam seminggu, dan tahun kabisat dengan tahun kompensasi berlaku hingga 2100.
- 2) 56-byte, baterai *backup, Nonvolatile (NV)*.
- 3) *Random Access Memory (RAM)* untuk penyimpanan data.
- 4) Dua kabel antar muka serial.
- 5) *Programmable* output sinyal *squarewave*.
- 6) Otomatis mendeteksi kegagalan *power* dan *switch circuitry*.
- 7) Mengonsumsi daya kurang dari 500nA pada mode baterai cadangan dengan osilator berjalan.
- 8) Kisaran suhu industri antara -40 ° C sampai +85 ° C.
- 9) Tersedia dalam 8-pin DIP atau SOIC.
- 10) Diakui oleh Underwriters Laboratory (UL).

RTC merupakan sebuah IC yang berhubungan dengan waktu mulai dari detik, menit, jam, hari, bulan dan tahun. Selain itu RTC juga dapat dipergunakan untuk menyimpan data berupa waktu pada internal RAM RTC dimana data dapat tetap disimpan walaupun RTC tidak memperoleh pasokan daya dari catu daya. IC DS1307 dapat menyimpan waktu yang sedang berjalan meskipun IC tersebut tidak mendapatkan pasokan daya dari *power supply*. Hal ini dikarenakan pada IC tersebut

terdapat baterai yang biasa dikenal dengan baterai BIOS yang akan selalu beroperasi. RTC dapat bekerja dengan daya yang rendah. DS1307 memiliki 8 buah pin dan tersedia dalam 8 pin DIP dan 9 pin SOIC. Untuk konfigurasinya ditunjukkan pada gambar 2.9 [11].



Gambar 2.9 Diagram PIN DS1307

RTC memiliki 8 pin diantaranya adalah VCC, X1, X2, V_{BAT}, GND, SDA, SCL, SQW/OUT. VCC merupakan pin untuk sumber tegangan dengan tegangan operasi antara 4,5 sampai 5,5 volt. Pin ground merupakan pin untuk pentanahan. X1 dan X2 akan dihubungkan dengan kristal senilai 32,768 Khz yang merupakan jalur *oscillator internal* yang didesain untuk dapat beroperasi dengan nilai kristal yang telah ditentukan. V_{BAT} merupakan tegangan tambahan yang terhubung pada baterai *backup* yang berfungsi sebagai tegangan pengganti pada saat RTC tidak memperoleh tegangan langsung dari *power supply* sehingga RTC tetap dapat bekerja untuk proses pencacahan waktunya. *Serial Clock Input* (SCL) digunakan sebagai sinkronisasi perpindahan data pada antarmuka serial PIN *Serial Peripheral Interface* (SPI). Untuk *Serial Data Input/Output* (SDA) berfungsi sebagai PIN masukan dan keluaran pada antarmuka serial kabel [11].

2.6 Bahasa Assembler

2.6.1. Penggunaan Assembler

Bahasa *assembly* adalah bahasa pemrograman mendasar yang sangat dekat dengan mesin. *Assembler* adalah program yang mengkonversi kode program sumber ke dalam bahasa mesin. *Assembler* menerjemahkan program yang ditulis dalam bahasa *assembly* ke dalam bahasa mesin, yang dapat dieksekusi oleh komputer maupun mikropengendali. Bahasa *assembly* disebut bahasa level bawah karena dalam struktur dan fungsi dekat dengan bahasa mesin. Pada aplikasi dengan menggunakan bahasa *assembly* biasanya dibuat

subrutin dalam bahasa *assembler* dan memanggilnya dari program bahasa tingkat tinggi. Subrutin bahasa *assembly* menangani operasi-operasi yang tidak tersedia dalam bahasa tingkat tinggi. Salah satu alasan bahasa *assembly* digunakan adalah untuk mempelajari arsitektur sistem operasi seperti pada mikroprosesor dan mikropengendali. Alasan yang lain adalah kegunaan pemrograman tertentu sulit atau tidak mungkin dikerjakan oleh bahasa tingkat tinggi [12].

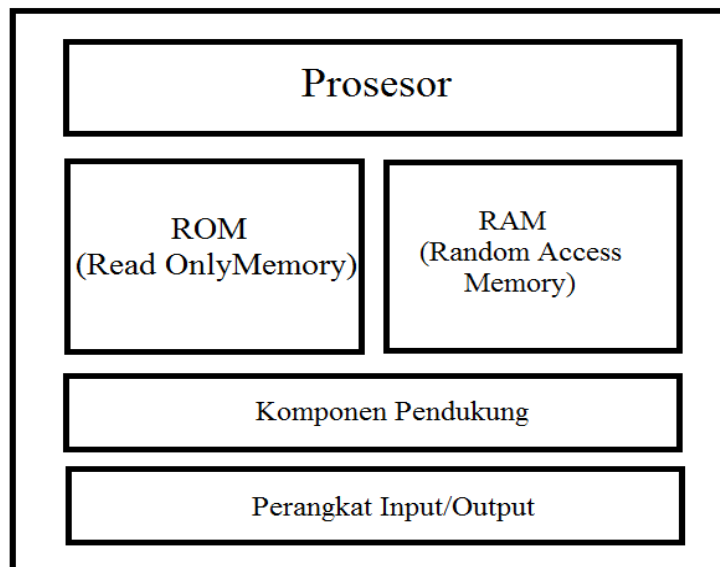
2.6.2. *Assembler Directive*

Assembler Directive (pengarah *assembler*) merupakan kontrol instruksi pada program rakitan, tetapi bukan sebagai bahasa rakitan yang akan dijalankan oleh mikropengendali. *Assembler Directive* tidak diterjemahkan menjadi kode mesin ketika dilakukan kompilasi. Jadi *assembler directive* berfungsi untuk mengubah penunjuk kode rakitan, misalnya mengubah lokasi kode (.asm) pada memori program yang digunakan, memberi label pada SRAM atau mendefinisikan suatu konstanta menggunakan sintaksis *assembler directive*. Pada umumnya *assembler directive* diawali dengan dot (tanda titik) [1].

2.6.3. Implementasi *Assembler* Pada Mikropengendali

Mikropengendali adalah sebuah sistem komputer sederhana yang dimuat dalam satu *chip*, biasa jga disebut *computer-on-chip*. Kata mikro menunjukkan bahwa perangkatnya berukuran kecil dan kata kontroler menunjukkan bahwa Dperangkat kecil tersebut dapat digunakan untuk mengontrol/mengendalikan perangkat lain. Sering juga diistilahkan sebagai *embedded microcontroller*, bersifat sebagai perangkat pendukung dan biasanya ditanamkan ke dalam perangkat yang dikontrolnya. Pada mikropengendali terdapat bagian *hardware* dan *software*. *Programmer* dapat memasukkan program ke dalam mikropengendali sehingga berfungsi sesuai dengan yang diinginkan oleh pengguna. Salah satu kelebihan mikropengendali adalah kesederhanaan dan ukurannya yang relatif kecil. Disamping kelebihan tersebut, terdapat kekurangan yaitu daya muat memori yang relatif rendah sehingga *programmer* dituntut untuk lebih hemat dalam membuat program. Komponen yang terdapat

dalam mikropengendali adalah prosesor, *Read Only Memory* (ROM), *Random Access Memory* (RAM), dan I/O. Prosesor berperan sebagai otak dari sistem. ROM adalah memori hardware yang menyimpan rangkaian program inti, sementara RAM adalah memori untuk menyimpan program yang disisipkan oleh *programmer*. I/O adalah bagian yang menghubungkan mikropengendali dengan dunia luar. RAM adalah bagian yang dapat diubah oleh pengguna sesuai dengan program yang dibuat. Komponen mikropengendali ditunjukkan pada gambar 2.10 [12].



Gambar 2.10. Komponen Mikropengendali

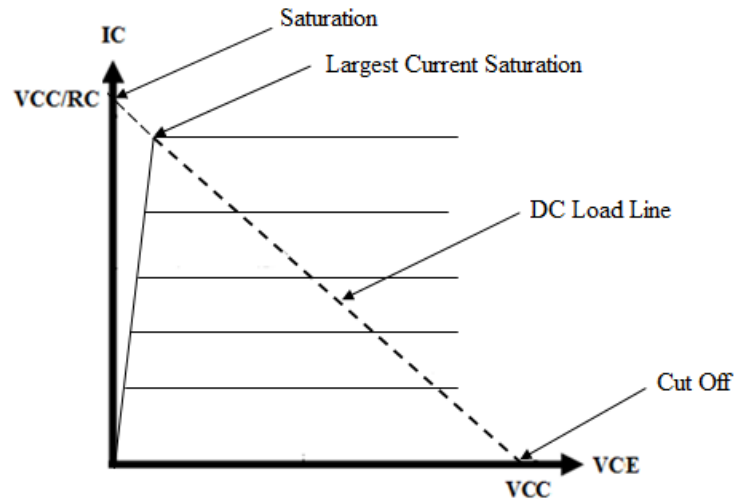
Pemrograman bahasa *assembly* adalah salah satu program dengan ukuran yang relatif kecil. Bahasa *assembly* adalah bahasa yang mendekati bahasa mesin, satu tingkat diatas bahasa mesin, memiliki keuntungan karena dapat mengakses bagian-bagian *hardware* dengan efisien. Pada saat ini telah banyak *chip* mikropengendali yang digunakan untuk berbagai aplikasi baik untuk penelitian maupun keperluan tertentu. Beberapa contoh aplikasi yang banyak menggunakan mikropengendali seperti tampilan *seven segment*, robot *line follower*, *servo controller*, dan jam digital sederhana. Penggunaan mikropengendali pada perangkat sederhana sering dijumpai seperti pada perangkat-perangkat yang memiliki tombol untuk memasukkan data dan tampilan layar digital seperti *handphone*, mesin fax, ataupun jam digital. Perangkat-perangkat sejenis tersebut kemungkinan besar

menggunakan mikropengendali sebagai komponen kendalinya. Pada saat ini dalam pembuatan robot sederhana juga hampir dapat dipastikan menggunakan komponen mikropengendali [13].

2.7 Transistor Sebagai Saklar

Salah satu fungsi transistor adalah sebagai saklar yaitu bila berada pada dua daerah kerjanya yaitu daerah jenuh (saturasi) dan daerah mati (*cut-off*). Transistor akan mengalami perubahan kondisi dari menyumbat ke jenuh dan sebaliknya. Transistor dalam keadaan menyumbat dapat dianalogikan sebagai saklar dalam keadaan terbuka, sedangkan dalam keadaan jenuh seperti saklar yang menutup. Daerah kerja transistor pada keadaan jenuh adalah keadaan dimana transistor mengalirkan arus secara maksimum dari kolektor ke emitor sehingga transistor tersebut seolah-olah *short* pada hubungan kolektor–emitor. Pada daerah ini transistor dikatakan sebagai penghantar maksimum (sambungan CE terhubung maksimum). Untuk daerah aktif transistor merupakan daerah kerja yang biasanya digunakan sebagai penguat sinyal. Transistor dikatakan bekerja pada daerah aktif karena transistor selalu mengalirkan arus dari kolektor ke emitor walaupun tidak dalam proses penguatan sinyal, hal ini ditujukan untuk menghasilkan sinyal keluaran yang tidak cacat. Daerah aktif terletak antara daerah jenuh (saturasi) dan daerah mati (*Cut off*). Daerah mati transistor atau daerah *cut off* merupakan daerah kerja transistor dimana keadaan transistor menyumbat pada hubungan kolektor – emitor. Daerah *cut off* sering dinamakan sebagai daerah mati karena pada daerah kerja ini transistor tidak dapat mengalirkan arus dari kolektor ke emitor. Pada daerah *cut off* transistor dapat di analogikan sebagai saklar terbuka pada hubungan kolektor–emitor. Untuk membuat transistor penghantar, pada masukan basis perlu diberi tegangan. Besarnya tegangan harus lebih besar dari V_{BE} (0,3 untuk *germanium* dan 0,7 untuk *silicon*). Dengan mengatur $I_b > I_c/\beta$ kondisi transistor akan menjadi jenuh seakan kolektor dan emitor *short circuit*. Arus mengalir dari kolektor ke emitor tanpa hambatan dan $V_{ce} \approx 0$. Besar arus yang mengalir dari kolektor ke emitor sama dengan V_{cc}/R_c . Keadaan seperti ini menyerupai saklar dalam kondisi tertutup (ON). Grafik hubungan daerah jenuh, daerah aktif dan daerah mati pada transistor dapat ditunjukkan pada gambar 2.11. Dari grafik tersebut dapat diketahui kondisi daerah transistor pada saat saturasi adalah pada saat transistor mengalirkan arus secara

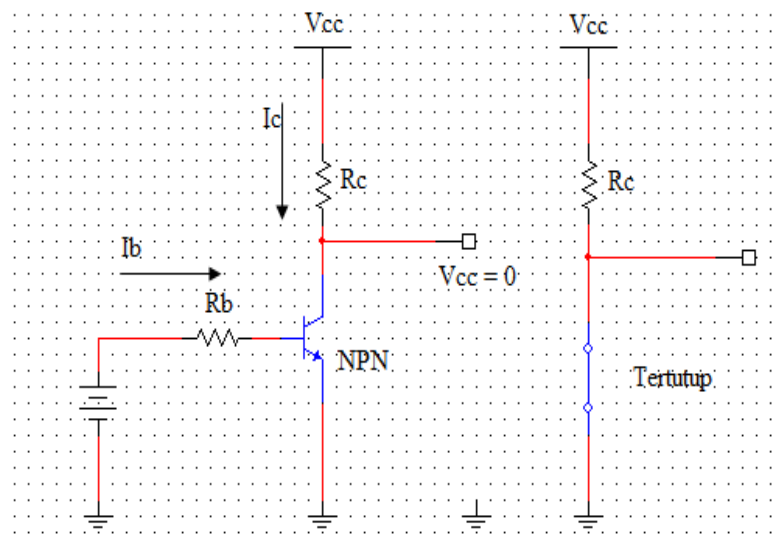
maksimum. Sedangkan untuk kondisi *cut off* merupakan kondisi sebaliknya yaitu karena transistor tidak dapat melewati arus. Untuk daerah aktif berada diantara daerah jenuh dan daerah *cut off* [14-16].



Gambar 2.11 Grafik Garis Beban DC

2.7.1 Transistor Pada Kondisi Jenuh

Untuk kondisi transistor pada kondisi jenuh atau saturasi ditunjukkan pada gambar 2.12.



Gambar 2.12 Transistor Pada Kondisi Jenuh

Untuk kondisi transistor pada keadaan jenuh atau saturasi maka besarnya tegangan kolektor-emitor V_{CE} suatu transistor pada konfigurasi pada gambar 2.12 dapat diketahui sebagai berikut [16]

$$V_{ce} = V_{cc} - I_c \cdot R_c \dots\dots\dots (2.1)$$

Karena kondisi jenuh $V_{ce} = 0$ Volt (transistor ideal) maka besarnya arus kolektor (I_c) adalah

$$I_c = \frac{V_{cc}}{R_c} \dots\dots\dots (2.2)$$

Besarnya arus yang mengalir agar transistor menjadi jenuh (saturasi) adalah

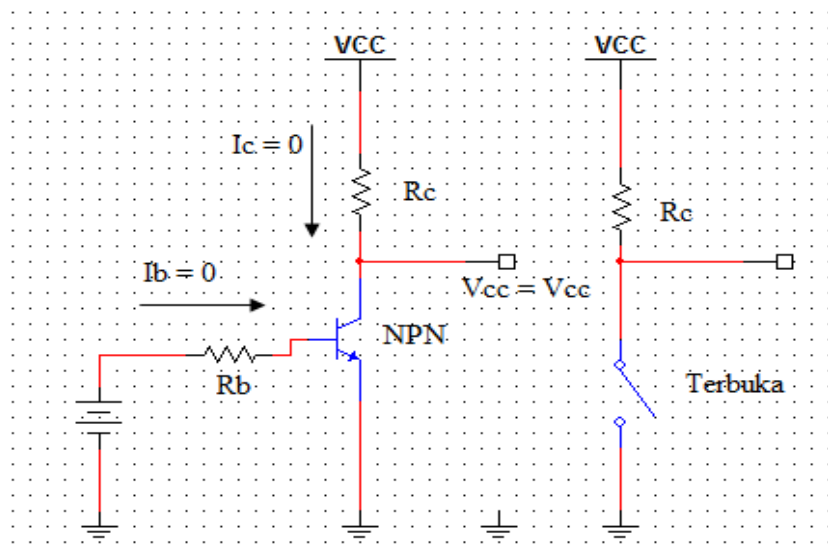
$$R_b = \frac{V_i - V_{be}}{I_b} \dots\dots\dots (2.3)$$

Sehingga besarnya arus basis (I_b) pada kondisi saturasi adalah

$$I_b = \frac{I_c}{\beta} \dots\dots\dots (2.4)$$

2.7.2 Transistor Pada Kondisi Mati

Untuk kondisi transistor pada kondisi mati atau *cut off* ditunjukkan pada gambar 2.13.



Gambar 2.13 Transistor Pada Kondisi *Cut off*

Dengan mengatur $I_b = 0$ atau tidak memberi tegangan pada bias basis atau basis diberi tegangan mundur terhadap emitor maka transistor akan dalam kondisi mati (*cut off*), sehingga tak ada arus mengalir dari

kolektor ke emitor ($I_c \approx 0$) dan $V_{ce} \approx V_{cc}$. Keadaan ini menyerupai saklar pada kondisi terbuka seperti ditunjukkan pada gambar diatas.

Besarnya tegangan antara kolektor dan emitor transistor pada kondisi mati atau *cut off* adalah [16]

$$V_{ce} = V_{cc} - I_c \cdot R_c \dots\dots\dots(2.5)$$

Karena kondisi mati $I_c=0$ (transistor ideal) maka :

$$V_{ce} = V_{cc} \dots\dots\dots(2.6)$$

Dengan demikian maka besar arus basis I_b adalah

$$I_b = \frac{I_c}{\beta} \dots\dots\dots(2.4)$$

$$I_b = 0 \dots\dots\dots(2.7)$$

2.8 Infra Merah

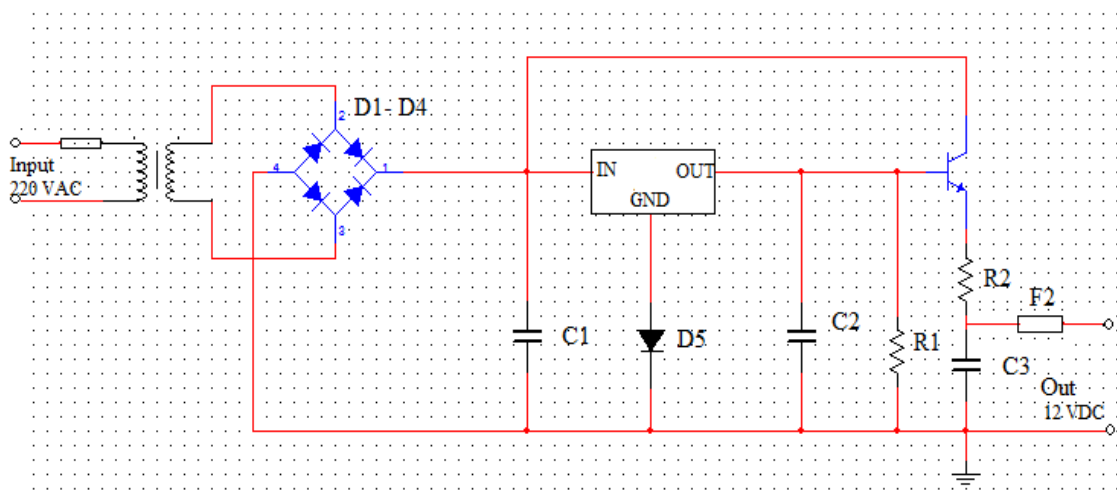
Salah satu contoh pengiriman data digital melalui media nirkabel adalah dengan memanfaatkan infra merah. Prinsip kerja dari spectrum infra merah tidak jauh berbeda dengan spectrum cahaya tampak. Beberapa sifat dari infra merah diantaranya merupakan gelombang tranversal yaitu arah getarannya tegak lurus dengan arah perambatannya, tidak tampak oleh mata, tidak dapat menembus materi yang tidak tembus pandang, dapat ditimbulkan oleh komponen yang menghasilkan panas, dan dapat mengalami pemantulan. Infra merah memiliki panjang gelombang lebih panjang dari cahaya tampak, tetapi lebih pendek dari radiasi gelombang radio yaitu 700 nm sampai 1 mm [13].

2.9 Teori Catu Daya

Pengertian *transformator* atau yang biasa kita kenal dengan *trafo* adalah komponen elektronika yang berfungsi untuk menaikkan atau menurunkan tegangan listrik. *Transformator* berperan dalam menyalurkan tenaga atau daya listrik dari tegangan tinggi ke tegangan yang rendah atau sebaliknya, namun dengan frekuensi yang sama. *Transformator* bekerja berdasarkan prinsip kerja induksi elektromagnetik. Dimana apabila terjadi suatu perubahan fluks magnet pada kumparan primer, maka akan diteruskan ke kumparan sekunder dan menghasilkan suatu gaya gerak listrik (GGL) induksi dan arus induksi. Agar selalu terjadi perubahan fluks magnet, maka

arus yang masuk (*input*) ini harus berupa arus bolak balik (AC). Fungsi *transformator* yang diaplikasikan dalam kehidupan sehari-hari diantaranya :

1. *Trafo step up*, fungsi *transformator* ini digunakan untuk menaikkan tegangan AC, *trafo* jenis ini dipakai dalam rangkaian-rangkaian pembangkit tegangan pada perangkat elektronika seperti *trafo inverter* monitor LCD, *trafo inverter* TV, dll.
2. *Trafo step-down* adalah kebalikannya, fungsi *transformator* ini untuk menurunkan tegangan AC, contoh pemakaiannya pada *adaptor*. Rangkaian *adaptor* ditunjukkan pada gambar 2.14.



Gambar 2.14 Rangkaian *Adaptor*

Adaptor adalah sebuah rangkaian elektronika yang dapat mengubah tegangan AC menjadi DC. Rangkaian ini adalah alternatif pengganti dari sumber tegangan DC, misalnya batu baterai dan *accumulator*. Keuntungan dari *adaptor* dibanding dengan batu baterai atau *accumulator* adalah sangat praktis berhubungan dengan ketersediaan tegangan karena *adaptor* dapat di ambil dari sumber tegangan AC yang ada di rumah, di mana pada jaman sekarang ini setiap rumah sudah menggunakan listrik. Selain itu, *adaptor* mempunyai jangka waktu yang tidak terbatas asal ada tegangan AC, tegangan AC ini sudah merupakan kebutuhan primer dalam kehidupan manusia. *Adaptor* sederhana terdiri dari bagian *input* tegangan yang merupakan bagian yang berfungsi sebagai penghubung sumber tegangan AC dari stop kontak yang ada di dalam rumah. Bagian ini terdiri dari *jack/steker* dan kabel *input*. Stop kontak adalah konektor sumber tegangan AC dari listrik PLN yang digunakan untuk menyalurkan tegangan pada *adaptor* melalui kabel *input* tegangan. Bagian penurun tegangan yang berfungsi untuk menurunkan tegangan AC 220 Volt menjadi tegangan yang lebih kecil, misalnya 3 volt,

4,5 volt, 6 volt, 7,5 volt, 9 volt, atau 12 volt. Untuk memilih *output* tegangan ini digunakan *rotary switch*/saklar putar/saklar 1 induk 6 anak. *Trafo* yang digunakan adalah jenis *step down*, dapat menggunakan *trafo* dengan arus 500 mA (mili Ampere). Bagian penyearah, yaitu mengubah tegangan AC menjadi tegangan DC. Komponen utamanya adalah dioda. Dioda yang digunakan berjumlah 4 dirangkai sedemikian rupa sehingga membentuk jembatan dioda atau *bridge dioda*. Bagian *filter* atau penyaring yang berfungsi untuk menghilangkan tegangan AC yang masih lewat. Efek dari tegangan AC yang lewat ini adalah munculnya suara dengung. Komponen yang dibutuhkan antara lain IC penstabil tegangan dan elco. Bagian *output* tegangan yang berfungsi sebagai keluaran tegangan berupa tegangan DC. Besar keluaran tegangan DC ini sesuai dengan tegangan *output* pada *trafo step down* yang diatur oleh *rotary switch* sesuai yang diinginkan. Fungsi dioda pada rangkaian sangat berhubungan dengan sistem pengendalian arus tegangan. Dioda penyearah (*rectifier*) berfungsi sebagai penyearah tegangan / arus dari arus bolak-balik (AC) ke arus searah (DC) atau mengubah arus AC menjadi DC [17].