BAB II
LANDASAN TEORI

Landasan teori sangat membantu untuk dapat memahami suatu sistem selain itu dapat juga dijadikan sebagai bahan acuan dalam merencanakan suatu sistem yang akan dibuat. Dengan pertimbangan hal-hal tersebut, maka landasan teori merupakan bagian yang harus dipahami untuk pembahasan selanjutnya.

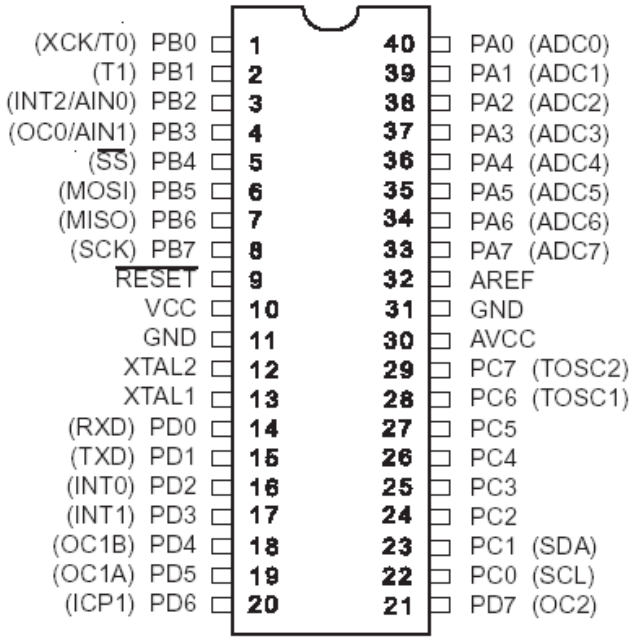
2.1 Mikrokontroler

Mikrokontroler merupakan suatu komponen elektronika yang didalamnya terdapat rangkaian mikroprosesor, memori (RAM/ROM) dan I/O, rangkaian tersebut terdapat dalam *level chip* atau biasa disebut *single chip microcomputer*. Pada Mikrokontroler sudah terdapat komponen-komponen mikroprosesor dengan bus-bus internal yang saling berhubungan. Komponen-komponen tersebut adalah RAM, ROM, *timer*, komponen I/O paralel dan serial, dan *interrupt kontroller*. Adapun keunggulan dari Mikrokontroler adalah adanya sistem *interrupt*. Sebagai perangkat kontrol penyesuaian, mikrokontroler sering disebut juga untuk menaikkan respon eksternal (*interrupt*) pada waktu yang nyata. Perangkat tersebut harus melakukan hubungan *switching* cepat, menunda satu proses ketika adanya respon eksekusi yang lain.[2]

AVR merupakan seri mikrokontroler CMOS 8-bit buatan Atmel, berbasis arsitektur RISC (*Reduced Instruction Set Computer*). Hampir semua instruksi dieksekusi dalam satu siklus clock. AVR mempunyai 32 *register general-purpose*, *timer/counter* fleksibel dengan mode *compare*, *interupt* internal dan eksternal, serial UART, *programmable Watchdog Timer*, dan mode *power saving*. Beberapa diantaranya mempunyai ADC dan PWM internal. AVR juga mempunyai *In- System Programmable Flash on-chip* yang mengijinkan memori program untuk diprogram ulang dalam sistem menggunakan hubungan serial SPI (*Serial Peripheral Inteface*). Chip AVR yang digunakan untuk tugas akhir ini

adalah ATmega 8535. ATmega 8535 adalah mikrokontroler CMOS 8-bit daya-rendah berbasis arsitektur RISC yang ditingkatkan. Kebanyakan instruksi dikerjakan pada satu siklus *clock*, ATmega 8535 mempunyai *throughput* mendekati 1 MIPS per MHz membuat desain sistem untuk mengoptimalkan konsumsi daya dengan kecepatan proses.[2]

AVR ATmega8535 mempunyai 40 kaki, 32 kaki yang digunakan untuk keperluan port paralel setiap port terdiri dari 8 pin, sehingga terdapat 4 port, yaitu port A, port B, port C, port D. seperti gambar dibawah ini :



Gambar 2.1. Kofigurasi Pin AVR ATmega8535

Berikut adalah penjelasan fungsi tiap kaki yang biasa ada pada seri mikrokontroler AVR ATmega8535 yaitu :

1. VCC *(power supply)*
2. GND *(ground)*
3. Port A (PA7..PA0)

Port A berfungsi sebagai *input* analog pada A/D Konverter. Port A juga berfungsi sebagai suatu Port I/O 8-bit dua arah, jika A/D Konverter tidak

digunakan. Pin - pin Port dapat menyediakan resistor *internal pull-up* (yang dipilih untuk masing-masing bit). Port A *output buffer* mempunyai karakteristik gerakan simetris. Ketika pin PA0 ke PA7 digunakan sebagai *input* dan secara eksternal ditarik rendah, pin–pin akan memungkinkan arus sumber jika resistor *internal pull-up* diaktifkan. Pin Port A adalah *tri-stated* manakala suatu kondisi *reset* menjadi aktif, sekalipun waktu habis.

4. Port B (PB7..PB0)

Port B adalah suatu Port I/O 8-bit dua arah dengan resistor *internal pull-up* (yang dipilih untuk beberapa bit). Port B *output buffer* mempunyai karakteristik gerakan simetris. Sebagai *input*, pin port B yang secara eksternal ditarik rendah akan arus sumber jika resistor *pullup* diaktifkan. Pin Port B adalah *tri-stated* pada suatu kondisi tertentu.

5. Port C (PC7..PC0)

Port C adalah suatu Port I/O 8-bit dua arah dengan resistor *internal pull-up* (yang dipilih untuk beberapa bit). Port C *output buffer* mempunyai karakteristik gerakan simetris. Sebagai input, pin port C yang secara eksternal ditarik rendah akan arus sumber jika resistor *pullup* diaktifkan. Pin Port C adalah *tri-stated* pada suatu kondisi tertentu.

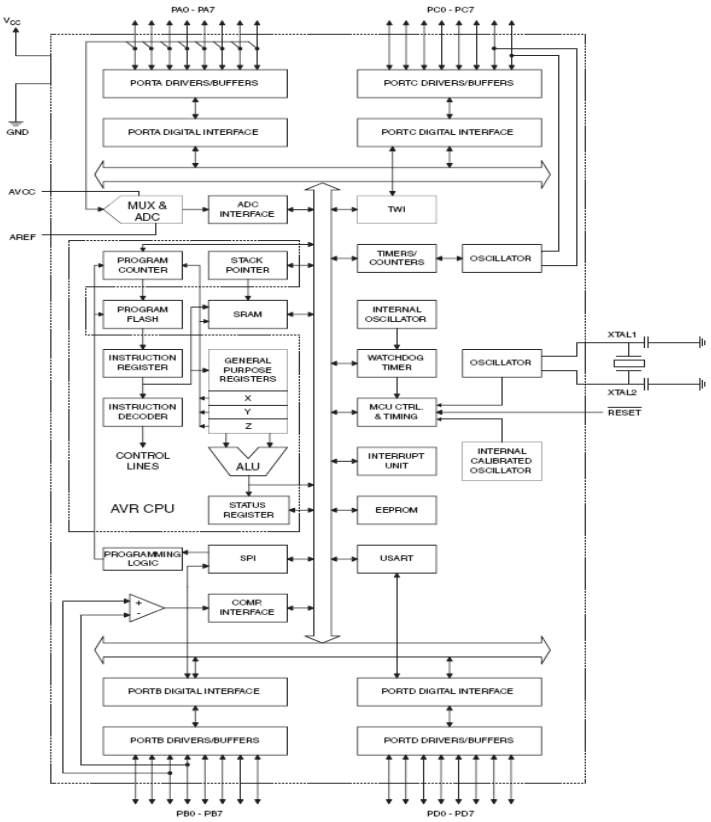
6. Port D (PD7..PD0)

Port D adalah suatu Port I/O 8-bit dua arah dengan resistor *internal pull-up* (yang dipilih untuk beberapa bit). Port D *output buffer* mempunyai karakteristik gerakan simetris. Sebagai input, pin port D yang secara eksternal ditarik rendah akan arus sumber jika resistor *pullup* diaktifkan. Pin Port D adalah *tri-stated* pada suatu kondisi tertentu.

7. *RESET (Reset input)*
8. XTAL1 *(Input Oscillator)*

9. XTAL2 *(Output Oscillator)* AVCC adalah pin penyedia tegangan untuk port A dan A/D Konverter.

10. AREF adalah pin referensi analog untuk A/D konverter.[2]



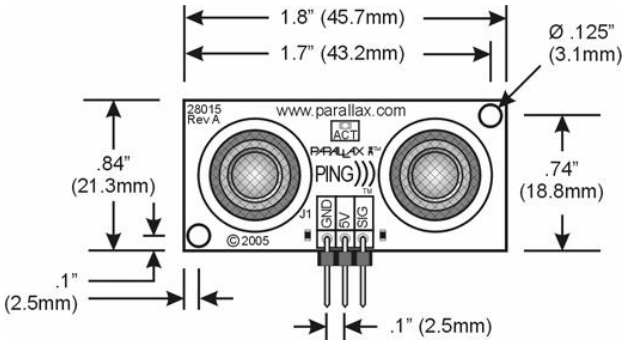
Gambar 2.2. Blok Diagram ATmega 8535

2.2 Sensor Ultrasonik

Sensor ultrasonik adalah suatu modul yang berfungsi untuk mengukur jarak, dengan cara memancarkan sinyal ultrasonik. Dengan ukuran yang cukup kecil yaitu (2.1cm x 4.5cm) sensor ini dapat mengukur jarak antara 3cm sampai 300cm. keluaran dari Ping berupa pulsa yang lebarnya mempresentasikan jarak. Lebar pulsanya bervariasi dari 115µs sampai 18.5ms.

Pada dasarnya, Ping terdiri dari sebuah *chip* pembangkit sinyal 40KHz, sebuah *speker ultrasonic* dan sebuah *microphone ultrasonic*. *Sepeker ultrasonic* mengubah sinyal 40KHz menjadi sinyal, sedangkan *microphone ultrasonic*

berfungsi untuk mendeteksi pantulan sinyalnya. Pada modul Ping terdapat tiga pin yang digunakan untuk jalur *power supply* (+5V), *ground* dan *signal*. Pin *signal* dapat langsung dihubungkan dengan mikrokontroler tanpa tambahan komponen apapun.[3]

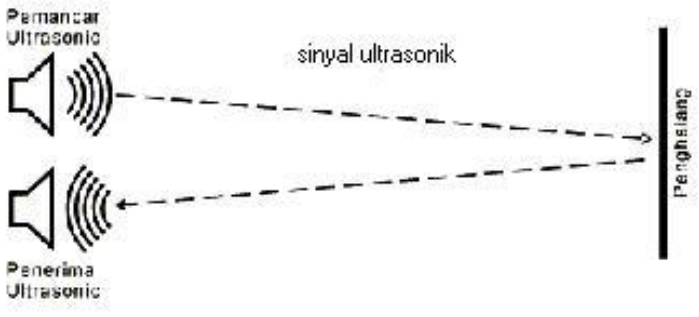


Gambar 2.3. Dimensi Sensor Ultrasonik

Sensor Ping *ultrasonic* ini memiliki spesifikasi sebagai berikut :

1. *Power supply* DC +5 V
2. *Supply Current* 30 mA typ; 35 mA max
3. *Range* 2cm sampai 300cm
4. *Burst Frequency* 40 kHz untuk 200 µs

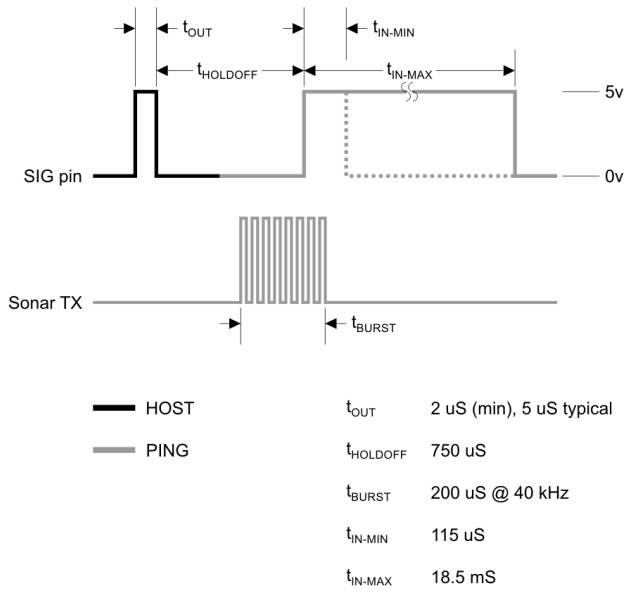
Prinsip kerja dari sensor Ping ultrasonik dapat ditunjukkan dalam gambar dibawah ini :



Gambar 2.4. Prinsip Kerja Sensor Ultrasonik

Ping bekerja dengan cara memancarkan sinyal ultrasonik kemudian menangkap pantulannya, ketika ada pulsa *trigger* dari mikrokontroler (pulsa *high* selama 5µs)

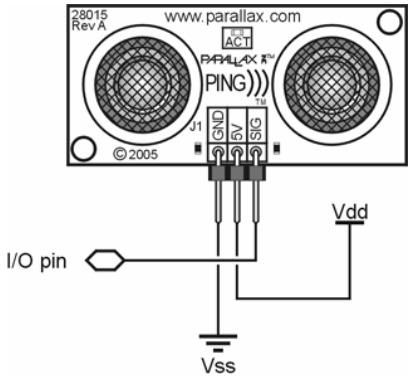
sinyal ultrasonik dengan frekuensi sebesar 40KHz akan dipancarkan selama 200µs. sinyal tersebut akan merambat di udara dengan kecepatan 344.424m/s (1cm setiap 29.034µs). selama menunggu pantulan, sensor ultrasonik akan menghasilkan sebuah pulsa. Pulsa tersebut akan berhenti ketika sinyal pantulan tertangkap oleh Ping. Maka lebar pulsa dapat merepresentasikan jarak sensor ultrasonik dengan objek. Selanjutnya mikrokontroler cukup mengukur lebar pulsa tersebut dan mengkonversinya dalam bentuk jarak.



Gambar 2.5. Cara Kerja Sensor Ultrasonik

Dari gambar di atas, cara kerja sensor Ping ultrasonik adalah:

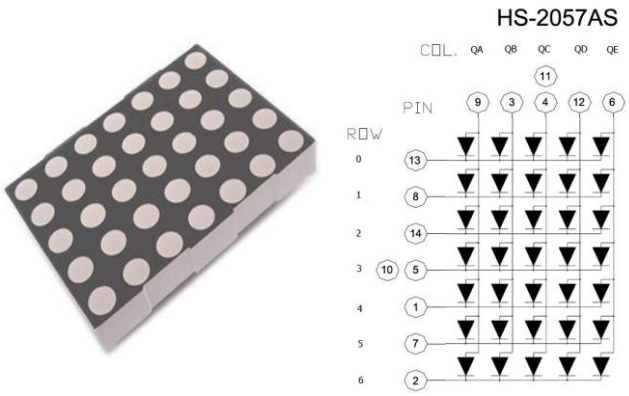
1. Kondisi awal SIG pin adalah logika 0
2. Akan memberikan logika 1 selama 2µs-5µs (*t-out*) untuk mengaktifkan Ping dan mulai pengukuran
3. Tunggu hingga 750µs (*t-holdoff*) yaitu saat Ping mulai memancarkan sinyal ultrasonik
4. Lalu merubah SIG pin sebagai input yang akan membaca jarak dengan cara membuat SIG pin berlogika 1, apabila SIG pin berubah kondisi dari logika 1 menjadi logika 0 maka itulah jarak yang terukur
5. Jika tidak terdapat dinding atau penghalang maka waktu bernilai 18.5ms yang berarti jarak yang terukur adalah 300cm.[3]



Gambar 2.6. Konfigurasi Pin Sensor Ultrasonik

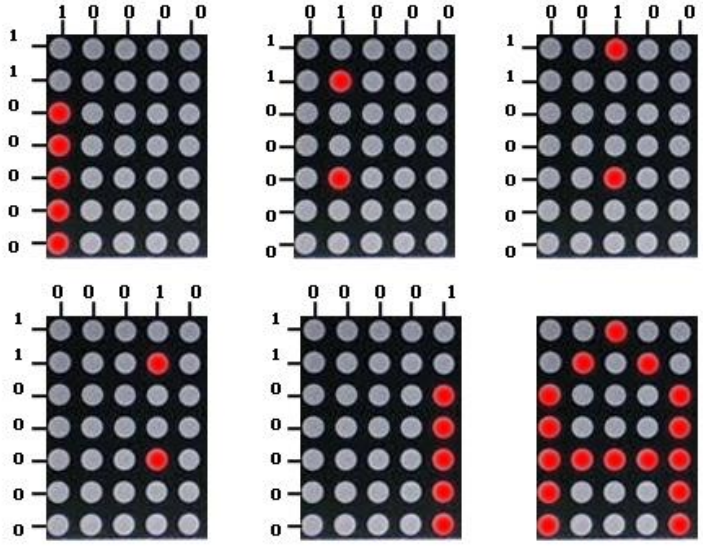
2.3 Dot Matriks

Dot matriks adalah perangkat layar yang digunakan untuk menampilkan informasi pada mesin, jam, indikator kereta api keberangkatan dan perangkat lain yang memerlukan perangkat tampilan sederhana resolusi terbatas. Layar terdiri dari matriks lampu atau indikator mekanik diatur dalam konfigurasi persegi panjang (bentuk lain juga mungkin, meskipun tidak umum) yang dapat beralih atau menonaktifkan lampu yang dipilih, teks atau grafis dapat ditampilkan. Sebuah dot matriks pengontrol mengkonversi instruksi dari prosesor menjadi sinyal-sinyal yang mengubah atau menonaktifkan lampu dalam matriks sehingga muncul pada tampilan dot matriks.



Gambar 2.7. Dot Matriks dan Konfigurasi Pin

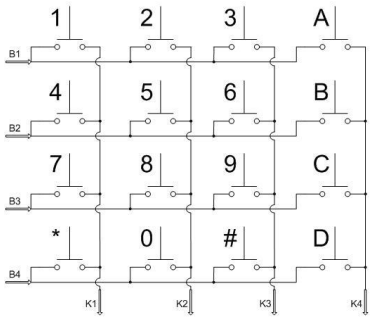
Gambar di bawah ini menunjukkan cara dot matriks menampilkan suatu abjad misalnya huruf “A”, pertama kolom QA diberi logika 1 untuk menyalakan kolom pertama, data 1 tersebut akan bergeser secara serial sampai kolom QE.



Gambar 2.8. Contoh *Scanning* Dot Matriks

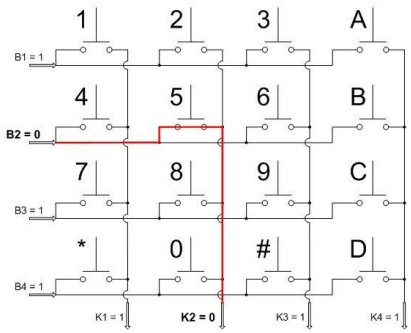
2.4 Keypad

Keypad Matriks adalah tombol-tombol yang disusun secara matriks (baris x kolom) sehingga dapat mengurangi penggunaan pin *input* (masukan). Sebagai contoh, *Keypad* Matriks 4×4 cukup menggunakan 8 pin untuk 16 tombol. Hal tersebut dimungkinkan karena rangkaian tombol disusun secara horizontal membentuk baris dan secara vertikal membentuk kolom:



Gambar 2.9. Keypad

Proses pengecekkan dari tombol yang dirangkai secara matriks adalah dengan teknik *scanning*, yaitu proses pengecekkan yang dilakukan dengan cara memberikan umpan-data pada satu bagian dan mengecek *feedback* (umpan-balik) – nya pada bagian yang lain. Dalam hal ini, pemberian umpan-data dilakukan pada bagian baris dan pengecekkan umpan-balik pada bagian kolom. Pada saat pemberian umpan-data pada satu baris, maka baris yang lain harus dalam kondisi sebaliknya. Tombol yang ditekan dapat diketahui dengan melihat asal data dan di kolom mana data tersebut terdeteksi:

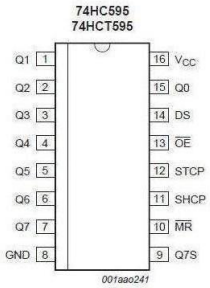


Gambar 2.10. Keypad (1)

Pada contoh di atas, tombol yang ditekan adalah tombol "**5**". Seperti terlihat bahwa B2 bernilai nol, sedangkan B1, B3, dan B4 bernilai satu. Kemudian dengan mengetahui bahwa asal data dari B2, dan umpan-baliknya terdeteksi pada K2, maka dapat disimpulkan bahwa tombol yang ditekan adalah tombol "**5**".[1]

2.5 IC 74HC595

IC 74HC595 berfungsi sebagai *shift register* yaitu untuk menggeser kolom tiap karakter dari dot matriks, pergeseran datanya pun secara berurutan. IC ini melakukan fungsi pergeseran data serial menjadi data faralel setiap pulsa *clock* diberikan.[5]

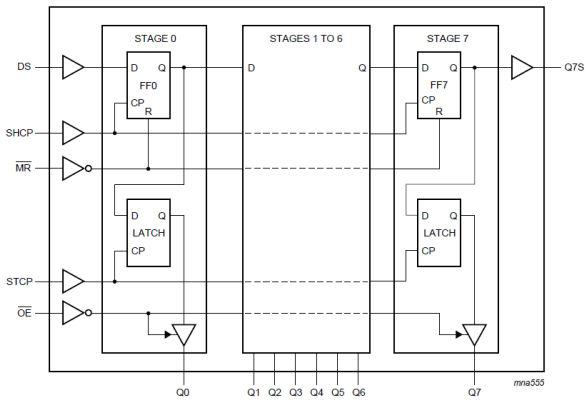


Gambar 2.11. IC 74HC595

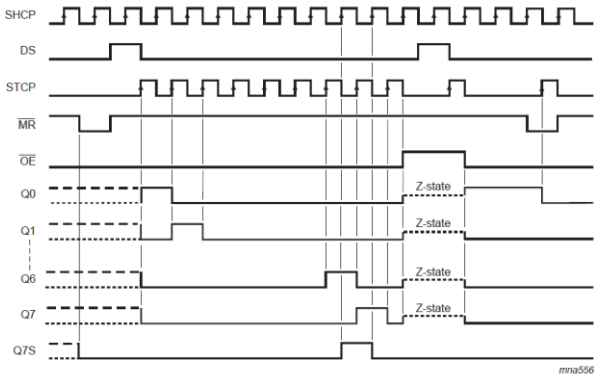
Tabel 2.1. Konfigurasi Pin

Simbol	Pin	Penjelasan
Q0 - Q7	15, 1 - 7	Paralel data *output*
GND	8	*Ground*
Q7'	9	Serial data *output*
MR	10	*Master reset*
SHCP	11	*Shift register clock input*
STCP	12	*Storage register clock input*
OE	13	*Output enable*
DS	14	Serial data *input*
VCC	16	Tegangan 5V

Secara elektronik, register dapat dibangun oleh satu atau beberapa FF. Setiap FF hanya menyimpan data 1 bit. Dalam hal ini FF data merupakan jenis FF yang banyak digunakan dalam menyusun rangkaian register,karena FF data dapat merekam dan menahan data. Berikut ini adalah gambar diagram pada ic 74hc595.



Gambar 2.12. Diagram IC 74HC595

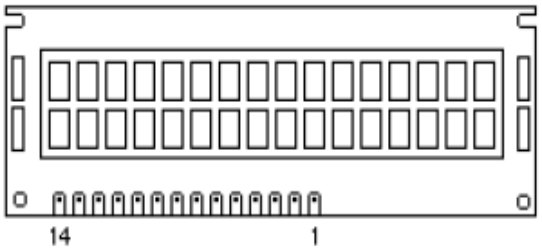


Gambar 2.13. Diagram Waktu

Pada gambar 2.13 terlihat bahwa ketika data tinggi masuk dan *clock* diberi kondisi tinggi data Q0 akan berkondisi 1 kemudian pada *clock* berikutnya data akan digeser secara berkesinambungan sampai Q7 dan untuk menyimpan data 8 bit secara seri dibutuhkan 8 kali perioda *clock*.

2.6 LCD (Liquid Crystal Display)

LCD (*Liquid Crystal Display*) merupakan penampil karakter yang berfungsi untuk menampilkan karakter yang dimasukan baik oleh *keypad* maupun berdasarkan program yang dibuat pada mikrokontroler. Data yang ditampilkan berupa karakter baik angka maupun huruf. Adapun struktur dan deskripsi dari LCD 2 *line* dengan 16 karakter tampak pada gambar dan tabel di bawah ini.



Gambar 2.14. LCD 16x2

Tampilan layar LCD yang digunakan adalah *LCD Dot Matrix Kompatibel Hitachi* yang memiliki *onboard* mikrokontroler HD44870. LCD ini mampu menampilkan 2 baris dengan 16 karakter tiap barisnya.

Tabel 2.2. Deskripsi LCD 16x2

Sinyal	I/O	Koneksi dengan AT8535	Penjelasan
Vss	1	GND	*Ground*
Vdd	1	VCC	Tegangan catu daya +5VDC
VD	1	-1 s/d 5 (dari VR1)	Tegangan untuk mengatur kontras LCD
Rs	1	Port D	Memilih *register* internal dari LCD
R/W	1	Port D	Sinyal *Read/Write*
E	1	Port D	Sinyal untuk *enable*
DB0.DB7	I/O	PD.0 hingga PD.7 (BUS)	8 bit data BUS

Pin data (D0....D7) dapat dihubungkan dengan bus data dari rangkaian mikrokontroler dengan lebar data 8 bit atau 4 bit, sementara pin kontrol terdiri dari Rs, R/W dan E. Pin Rs (*Register select*) berfungsi sebagai indikator untuk menetukan jenis data yang masuk, apakah data atau perintah. Logika low menunjukan yang masuk adalah perintah, sementara logika *high* menunjukan data.

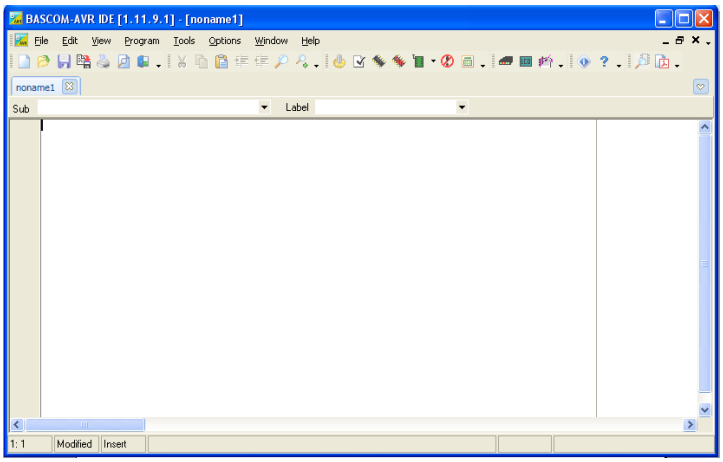
Mikrokontroler akan mengirim suatu karakter atau kata pada input data 8 bit modul LCD, serta mengirim sinyal atau logika *high* pada pin Rs. Kemudian, mikrokontroler akan mengirim perintah penulisan pada baris ke 1 ataupun baris 2 melalui *input* data 8 bit dengan memberikan logika *high* pada pin Rs. Pin R/W berfungsi sebagai instruksi pada LCD, apakah *Read* atau *Write*. Ketika R/W diberi logika *low* berarti penulisan data (*write*), sedangkan logika *high* berarti pembacaan data (*read*). Pin E (*Enable*) digunakan untuk memegang data, baik masuk atau keluar dari LCD.

2.7 Bascom AVR

Bascom-AVR adalah program *basic compiler* berbasis windows untuk mikrokontroler keluarga AVR merupakan pemrograman dengan bahasa tingkat tinggi ” *BASIC* ” yang dikembangkan dan dikeluarkan oleh MCS elektronika sehingga dapat dengan mudah dimengerti atau diterjemahkan.

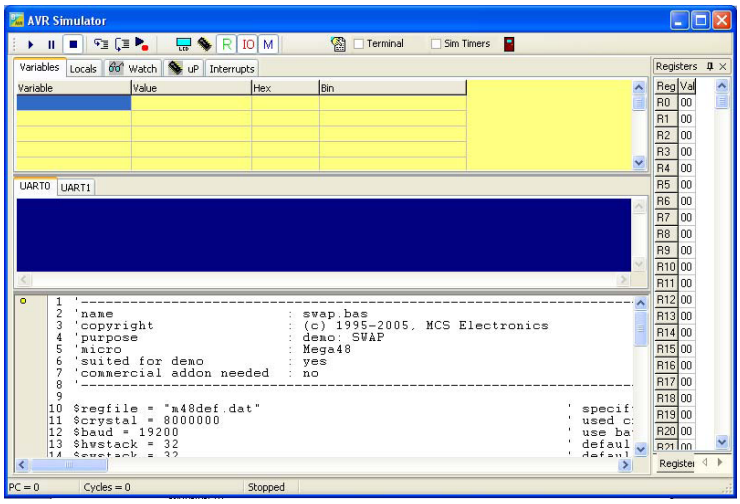
Dalam program bascom-AVR terdapat beberapa kemudahan, untuk membuat program *software* ATmega 8535, seperti program simulasi yang sangat

berguna untuk melihat, simulasi hasil program yang telah kita buat, sebelum program tersebut kita *download* ke IC atau ke mikrokontroler. Ketika program bascom-AVR dijalankan dengan mengklik ikon bascom-AVR, maka jendela berikut akan tampil :



Gambar 2.15. Tampilan Jendela Program Bascom AVR

Bascom-AVR menyediakan pilihan yang dapat mensimulasikan program. Program simulasi ini bertujuan untuk menguji suatu aplikasi yang dibuat dengan pergerakan LED yang ada pada layar simulasi dan dapat juga langsung dilihat pada LCD, jika kita membuat aplikasi yang berhubungan dengan LCD.



Gambar 2.16. Tampilan Simulasi Bascom AVR

Intruksi yang dapat digunakan pada editor Bascom-AVR relatif cukup banyak dan tergantung dari tipe dan jenis AVR yang digunakan. Berikut ini beberapa instruksi-instruksi dasar yang dapat digunakan pada mikrokontroler ATmega 8535.

Tabel 2.3. Beberapa Instruksi Dasar Bascom AVR

Instruksi	Keterangan
DO ..... LOOP	Perulangan
GOSUB	Memanggil Prosedur
IF ..... THEN	Percabangan
FOR ..... NEXT	Perulangan
WAIT	Waktu Tunda Detik
WAITMS	Waktu Tunda MiliDetik
WAITUS	Waktu Tunda MicroDetik
GOTO	Loncat Kealamat Memori
SELECT ..... CASE	Percabangan