BAB II
LANDASAN TEORI

2.1. Mikrokontroler BS2p40

Basic stamp adalah mikrokontroler yang dikembangkan oleh *Parallax Inc* yang diprogram menggunakan *format* bahasa pemrograman *basic*. Program yang dibuat di-*downloa*d melalui *port* serial dan dapat menggunakan konverter USB to Serial untuk komputer yang tidak memiliki *port* serial. Mikrokontroler *basic stamp* membutuhkan *power supply* saat men*download* pemrograman dan tidak kehilangan program yang sudah di *download* saat baterai atau *power supply* dicabut.

Kode Basic disimpan di dalam EEPROM serial pada *board basic stamp*. EEPROM menyediakan penyimpanan yang sulit diubah, yaitu menjaga memory saat kehilangan *power*. EEPROM digunakan dalam *basic stamp* 1 dan 2 yang dijamin menyimpan selama 40 tahun ke depan dan mampu ditulisi ulang 10.000.000 kali per lokasi memori.

Mikrokontroler *basic stamp* memiliki versi yang berbeda - beda. Basic stamp memiliki tujuh versi, yaitu *basic stamp* 1, *basic stamp* 2, *basic stamp* 1e, *basic stamp* 2P, *basic stamp* 2Pe dan *basic stamp* 2sx. *Basic* 5 *stamp* jalan pada tegangan DC 5 sampai 15 volt. Pada modul *basic stamp* terdapat IC *regulator* LM7805 dengan *output* 5 volt yang mengubah input 6 hingga 15 volt (pada pin VIN) turun menjadi 5 volt yang dibutuhkan komponen. *Basic stamp* yang di pakai adalah *basic stamp* BS2P40 yang mempunyai 40 pin I/O. Pemilihan basic stamp ini karena membutuhkan banyak *input* atau *output* yang dipakai. Berikut ini adalah tampilan *basic stamp* BS2P40.

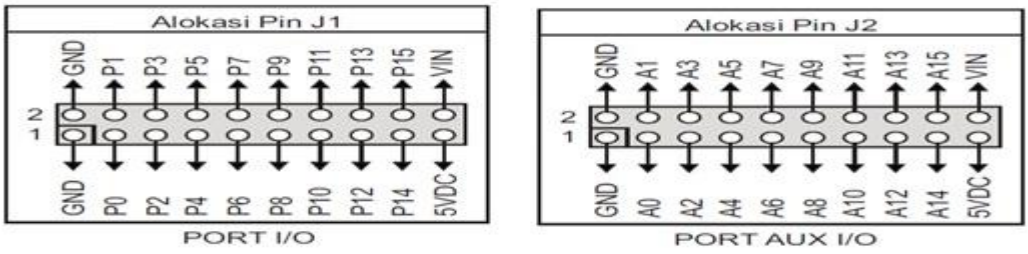


Gambar 2.1-1 Mikrokontroler

Basic stamp ini mempunyai spesifikasi *hardware* sebagai berikut:

1. Mikrokontroler *basic stamp* 2P40 *Interpreter Chip* (PBASIC48W/P40)
2. 8 x 2Kbyte EEPROM yang mampu menampung hingga 4000 instruksi.
3. Kecepatan prosesor 20MHz Turbo dengan kecepatan eksekusi program hingga 12000 instruksi per detik.
4. RAM sebesar 38*byte* (12 I/O, 26 variabel) dengan *Scratch Pad* sebesar 128 *byte*.
5. Jalur *input* / *output* sebanyak 32 pin.
6. Tersedia jalur komunikasi serial UART RS-232 dengan konektor DB9.
7. Tegangan input 9 – 12 VDC dengan tegangan output 5 VDC.

Berikut ini adalah alokasi pin yang terdapat pada mikrokontroler *basic stamp* BS2P40.



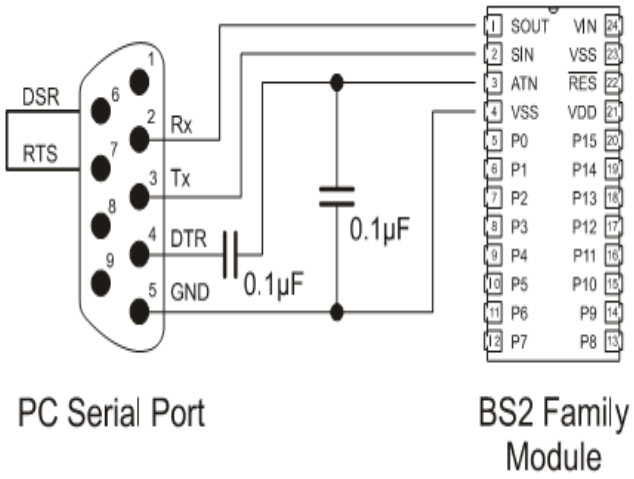
Gambar 2.1-2 Alokasi pin basic stamp

2.1.1. Koneksi Pemrograman Basic Stamp

Dianjurkan menggunakan papan buatan *Parallax* dan kabelnya dalam pemrograman modul *Basic Stamp*. Jika item ini tidak tersedia,

anda dapat membuat papan sendiri dengan mencontoh diagram berikut pada circuit dan kabel buatan sendiri. Hati-hati mengikuti diagram ini, yang biasa muncul dalam pemrograman *Basic Stamp* adalah kabel modif buatan yang jelek atau koneksi pemrograman pada papan aplikasinya.

Dengan koneksi pemrograman untuk seluruh modul BS2, memungkinkan membalikkan pasangan kabel dan masih mendapat hasil yang positif dalam penggunaan penguji "koneksi" teknologi *parallax*. Tim pendukung mencoba dan belum tetap dapat mengkomunikasikan *Basic Stamp*. Sangat vital untuk memeriksa koneksi menggunakan meteran dan memastikan nomor pin untuk menghindari masalah seperti ini, lebih jelasnya lihat gambar 2.1.3. dibawah ini.



Gambar 2.1-3 PC Serial Port

Catatan : meskipun tidak tampak (lihat gambar 2.1-3), power harus dihubungkan dengan Basic Stamp saat memprogramnya. Juga koneksi pemrograman adalah sama untuk BS2p40. Model 40 pin BS2p memiliki ekstra set register *INS, OUTS,* dan *DIRS* untuk total 38 bytes RAM variabel. Terdapat register *"shadow"* dengan *switched in* dan *out* peta memori dengan perintah *AUXIO, MAINIO*, dan *IOTERM*. Kata variabel *INS* adalah unik yang hanya bersifat read-only. 16 bit INS mencerminkan bagian pin I/O p0 melalui p15. Ini hanya read-only, tidak dapat dilakukan proses write. *OUTS* mengandung bagian 16 kancing output. Kontrol DIRS langsung (input atau

output) tiap 16 pin I/O. Sebuah nilai 0 pada bit DIRS lain membentuk pin korespondensi sebagai input dan nilai 1 membentuk pin korespondensi sebagai output. Jadi jika bit 5 DIRS bernilai 0 dan bit 6 DIRS adalah 1, kemudian pin 5 I/O (P5) adalah sebagai input dan pin 6 I/O (P6) adalah sebagai output. Sebuah pin sebagai input di dalam circuit di luar Basic Stamp dan Basic Stamp tidak mengubah bagian ini. Sebuah pin output ditetapkan sebagai bagian yang diindikasikan oleh korespondensi bit register OUTS.Ketika Basic Stamp di beri power atau reset, seluruh lokasi memori dikosongkan menjadi 0. Jadi seluruh pin adalah sebagai input (DIRS=%0000000000000000). Juga jika program PBasic mengatur seluruh pin I/O menjadi output (DIRS=%1111111111111111), kemudian mereka berinisial output low, mulai kancing output (OUTS) dikosongkan menjadi kosong seluruhnya pada saat power-up atau reset.

2.2. RTC DS1307 (*Real-Time Clock*)

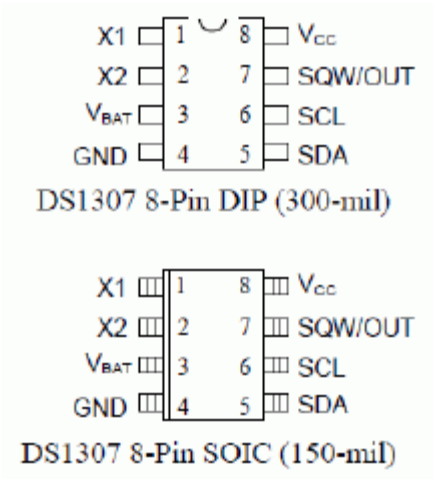
Real-time clock DS1307 adalah IC yang dibuat oleh perusahaan Dallas *Semiconductor*. DS1307 merupakan sebuah IC yang dapat digunakan sebagai pengaturan waktu yang meliputi detik, menit, jam, hari, tanggal, bulan dan tahun. Pengaksesan data dilakukan dengan sistem serial sehingga hanya membutuhkan dua jalur untuk komunikasi yaitu jalur *clock* untuk membawa informasi data *clock* dan jalur data yang membawa data. *Real-time clock* DS1307 memiliki fitur sebagai berikut:

1. *Real-time clock* (RTC) meyimpan data-data detik, menit, jam, tanggal dan bulan, hari dalam seminggu, dan tahun valid hingga 2100.
2. 56-byte, *battery-backed*, RAM *nonvolatile* (NV) RAM untuk penyimpanan.
3. Antarmuka serial *Two-wire* (I2C).
4. Sinyal keluaran gelombang kotak terprogram (*Programmable squarewave*).
5. Deteksi otomatis kegagalan-daya (*power-fail*) dan rangkaian *switch.*
6. Konsumsi daya kurang dari 500nA menggunakan mode baterei cadangan dengan operasional osilator.

7. Tersedia fitur industri dengan ketahanan suhu: -40°C hingga +85°C.
8. Tersedia dalam kemasan 8-pin DIP atau SOIC.

Sedangkan daftar pin RTC DS1307 adalah sebagai berikut:

1. VCC – *Primary Power Supply*.
2. X1, X2 – 32.768kHz *Crystal Connection*.
3. VBAT – +3V *Battery Input*.
4. GND – *Ground*.
5. SDA – *Serial Data*.
6. SCL – *Serial Clock.*
7. SQW/OUT – *Square Wave/Output Driver*.



Gambar 2.2-1 RTC DS1307 (Data Sheet IC Real-Time Clock DS1307)

2.2.1. Definisi-definisi Kondisi Bus

Berikut ini adalah defenisi kondisi bus pada sistem komunikasi serial I2C/ TWI:

1. **Bus tidak sibuk (*bus not busy*):** menyatakan pada saat ini bus tidak sibuk yaitu pada saat jalur *clock* (SCL) dan jalur data (SDA) dua-duanya dalam keadaan HIGH.
2. **Mulai transfer data (*start data transfer*):** ditandai dengan perubahan kondisi SDA dari HIGH ke LOW ketika SCL dalam kondisi HIGH.

3. **Stop transfer data (*stop data transfer*):** ditandai dengan perubahan kondisi SDA dari LOW ke HIGH ketika SCL dalam kondisi HIGH.
4. **Data valid:** data yang dikirim bit demi bit dianggap valid jika setelah START, kondisi SDA tidak berubah selama SCL HIGH, baik SDA HIGH maupun SDA LOW tergantung dari bit yang ingin ditransfer. Setiap siklus HIGH SCL baru menandakan pengiriman bit baru. *Duty cycle* untuk SCL tidak mesti 50%, tetapi frekuensi kemunculannya hanya ada dua macam, yaitu mode standar 100 kHz dan *fast mode* atau mode cepat 400 kHz. Setelah SCL mengirimkan sinyal HIGH yang kedelapan, arah transfer SDA berubah, sinyal kesembilan pada SDA ini dianggap sebagai *acknowledge* dari *receiver* ke *transmitter*. DS1307 hanya bisa melakukan transfer pada mode standar 100 kHz.
5. **Pemberitahuan (*Acknowledge*):** setiap receiver wajib mengirimkan sinyal *acknowledge* atau sinyal balasan setiap selesai pengiriman 1-byte (8-bit data). *Master* harus memberikan *eksta clock* atau *clock* tambahan pada SCL, yaitu *clock* kesembilan untuk memberikan kesempatan *receiver* mengirimkan sinyal *acknowledge* ke *transmitter* berupa keadaan LOW pada SDA selama SCL HIGH. Meskipun *master* berperan sebagai *receiver*, ia tetap sebagai penentu sinyal STOP. Pada bit akhir penerimaan *byte* terakhir, *master* tidak mengirimkan sinyal *acknowledge*, SDA dibiarkan HIGH oleh *receiver* dalam hal ini *master*, kemudian master mengubah SDA dari LOW menjadi HIGH yang berarti sinyal STOP.

2.2.2. Mode Operasi RTC DS1307

IC RTC DS1307 beroperasi dalam dua mode, yaitu *Slave Receiver Mode (Write Mode)* dan *Slave Transmitter Mode (Read Mode).*

1. ***Slave Receiver Mode (Write Mode):***

   Mode penerima *slave* (*write mode*) dalam pengiriman sinyal memiliki urutan:

1) Setelah sinyal START, master mengirim *byte* pertama yang terdiri dari 7-*bit address* IC DS1307, yaitu **1101000** dan 1-*bit* R/W, yaitu LOW, karena ini adalah opersai WRITE.
2) *Hardware* pada DS1307 akan membaca *address* yang dikirimkan oleh *master* tersebut, kemudian *slave*, dalam hal ini IC DS1307 akan *bit-acknowledge* pada SDA.
3) Setelah itu *master* akan mengirimkan *address* tempat data pertama akan diakses. *Address* ini berbeda dengan 7-*bit address* tadi, ini adalah *address* "isi" IC DS1307, bukan *address* dari IC DS1307. *Address* ini akan diimpan dalam *register* pointer oleh DS1307 yang juga mengirim sinyal *acknowledge* ke *master*.
4) Setelah itu *master* dapat mengirimkan sejumlah *byte* ke *slave*, dimana setiap *byte* dibalas dengan *acknowledge* oleh *slave*. Setiap menerima *byte* baru isi *register pointer* ditambah satu sehingga *register* ini menunjuk ke alamat berikutnya dari lokasi data pada DS1307. Setelah menerima *acknowledge* terakhir, *master* akan mengirim sinyal STOP untuk mengakhiri transfer data.

2. ***Slave Transmitter Mode (Read Mode):***

Sama seperti *mode write*, setelah *master* memberikan sinyal START, ia mengirimkan *byte* pertama yang terdiri dari 7-bit dalam IC DS1307, yaitu **1101000**, diikuti 1-*bit* R/W, yaitu HIGH. Setelah menerima *byte* pertama ini, *slave*, dalam hal ini DS1307 akan mengirimkan bit *acknowledge* pada SDA. Setelah itu slave mulai mengirimkan sejumlah *byte* ke *master*. Setiap *byte* pengiriman dibalas dengan 1-*bit acknowledge* oleh *master*. *Byte* pertama yang dikirimkan oleh *slave* atau DS1307 adalah data yang alamatnya ditunjuk oleh *register pointer* pada DS1307. Setiap kali pengiriman *byte* ke *master*, secara otomatis isi *register pointer* ditambah satu. DS1307 akan terus menerus mengirimkan *byte* ke *master* sampai *master* mengirimkan bit *'not acknowledge'* diikuti dengan sinyal STOP. (*Data sheet* RTC DS1307).

2.2.3. Peta Alamat DS1307

Pemetaan alamat (*address map*) pada RTC dimana register-register RTC di tempatkan pada lokasi pengalamatan 00h sampai 07h. sedangkan register-register RAM (*Random Access Memory*) ditempatkan pada lokasi pengalamatan 08h sampai 3Fh.

Khusus alamat **02H**, *bit*-6 LOW untuk siklus jam 00 – 24 dan HIGH untuk siklus jam 00 – 12. *Bit*-5 HIGH pada saat PM dan LOW pada saat AM atau angka puluhan jika bit-6 LOW.

2.2.4. Register Kontrol RTC DS1307

Register kontrol pada RTC DS1307 digunakan untuk mengontrol operasi pada pin SQW/OUT.

Table 2.2-1 kontrol (control register) (Data sheet RTC DS1307)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
OUT	0	0	SQWE	0	0	RS1	RS0

Keterangan bit-bit pada register kontrol:

1. ***Bit*-7: *Output Control (OUT)*** yaitu jika pin SQW/OUT *di-disable* sehingga tidak mengeluarkan *clock*, bit-7 ini menentukan level sinyal yang keluar dari pin SQW/OUT. Jika *bit*-7 ini LOW, maka level pin SQW/OUT ikut LOW dan jika *bit*-7 ini HIGH, maka level pin SQW/OUT ikut HIGH.
2. ***Bit*-4: *Square-wave Enable*** digunakan untuk *enable/ disable* keluarnya clock dari pin SQW/OUT. HIGH berarti *enable* dan LOW berarti *disable*. Frekuensi sinyal *clock* yang keluar dari pin SQW/OUT ditentukan oleh kondisi bit-1 dan bit-0.
3. ***Bits* 1, 0: *Rate Select* (RS1, RS0)** untuk menentukan frekuensi yang keluar dari pin SQW/OUT. Kombinasi nilai RS0, dan RS1 menghasilkan *output* gelombang kotak dengan nilai frekuensi masing-masing yang ditunjukkan oleh tabel 2.2-2.

Table 2.2-2 Rate select (RS1, RS0) (Data sheet RTC DS1307)

RS1	RS0	SQUARE-WAVE OUTPUT FREQUENCY
0	0	1Hz
0	1	4.096kHz
1	0	8.192kHz
1	1	32.768kHz

2.3. EEPROM *INTERNAL*

Basic stamp memiliki dua jenis memory yaitu *RAM* (untuk variable yang digunakan oleh program) dan EEPROM (untuk menyimpan program). EEPROM adalah tipe khusus dari *PROM* (*Programmable Read-Only Memory*) yang bisa dihapus dengan memakai perintah elektris. Seperti juga tipe *PROM* lainnya, EEPROM dapat menyimpan isi datanya, bahkan saat listrik sudah dimatikan.

EEPROM sangat mirip dengan *flash memory* yang disebut juga *flash* EEPROM. Perbedaan mendasar antara *flash memory* dan EEPROM adalah penulisan dan penghapusan EEPROM dilakukan pada data sebesar satu byte, sedangkan pada flash memory penghapusan dan penulisan data ini dilakukan pada data sebesar satu block. Oleh karena itu *flash memory* lebih cepat.

Dengan ROM biasa, penggantian BIOS hanya dapat dilakukan dengan mengganti chip. Sedangkan pada EEPROM program akan memberikan instruksi kepada pengendali chip supaya memberikan perintah elektronis untuk kemudian mendownload kode BIOS baru untuk diisikan kepada chip. Hal ini berarti perusahaan dapat dengan mudah mendistribusikan BIOS baru atau update, misalnya dengan menggunakan disket. Hal ini disebut juga flash BIOS.

EEPROM juga digunakan untuk menyimpan sejumlah data seperti halnya pada komputer dekstop yang menggunakan Harddisk untuk menyimpan program dan file. Perbedaan penting antara RAM dan EEPROM adalah :

1. RAM kehilangan isinya ketika *Basic Stamp* kehilangan power, ketika power diberikan kembali, semua lokasi RAM berisi 0.
2. EEPROM mempertahankan isi memory-nya dengan atau tanpa power, sampai memory EEPROM diisi ulang.

Ketika program di *download* ke dalam *Basic Stamp*, semua data dan instruksi disimpan di EEPROM dimulai pada alamat tertinggi (2047) dan diteruskan ke alamat terendah, Arah data mengikuti alur yang diberikan kepada sekumpulan data untuk disimpan di lokasi EEPROM yang tersedia. Hasilnya, 10 byte pertama dari EEPROM akan terlihat seperti gambar 2.3.1 berikut :

Table 2.3-1 Peta Memory EEPROM

	EEPROM Location (address)									
	0	1	2	3	4	5	6	7	8	9
Contents	72	69	76	76	79	104	101	108	108	111

Hal ini penting untuk direalisasikan bahwa EEPROM tidak ditulis ulang selama program masih membutuhkan tempat penyimpanan data, atau selama EEPROM masih dalam proses pengisian data tertentu. Selama proses *download*, EEPROM selalu menulis 16 byte data jika dan hanya jika beberapa lokasi sedang diisi / ditulis.

2.4. LCD (*Liquid Crystal Display*)

LCD yang digunakan adalah jenis LCD M1632, yang merupakan modul LCD dengan tampilan 16 x 2 baris dengan konsumsi daya rendah. Modul tersebut dilengkapi dengan desain mikrokontroler yang didesain khusus untuk mengendalikan LCD. Mikrokontroler HD44780 buatan Hitachi yang berfungsi sebagai pengendali LCD memiliki CGROM, CGRAM, dan DDRAM. Berikut bagian-bagian dari LCD M1632:

1. DDRAM (*Display data Random Accsee Memory*) merupakan memori tempat karakter yang ditampilkan berada. Contoh untuk karakter ‘L’ atau 4CH yang ditulis pada alamat 00, karakter tersebut akan tampil pada baris pertama dan kolom pertama dari LCD. Apabila karakter tersebut ditulis pada alamat 40, maka karakter tersebut akan tampil pada baris kedua kolom pertama dari LCD.

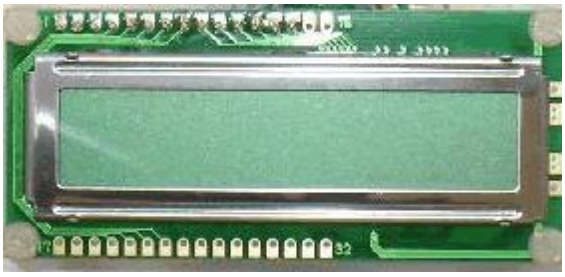
2. CDRAM (*Character Generator Random Acces Memory*) merupakan memori untuk menggambarkan pola sebuah karakter dimana bentuk dari

karakter dapat diubah-ubah sesuai keinginan. Namun, memori akan hilang saat power supply tidak aktif sehingga pola karakter akan hilang.

3. CGROM (*Character Generator Read Only Memory*) merupakan memori untuk menggambarkan pola sebuah karakter dimana pola tersebut sudah ditentukan secara permanent dari HD44780 sehingga pengguna tidak dapat mengubahnya lagi. Namun, oleh karena ROM bersifat permanent, pola karakter tersebut tidak akan hilang walaupun power supply tidak aktif.

2.4.1. Konfigurasi PIN LCD

Untuk keperluan antarmuka suatu komponen elektronik dengan mikrokontroler, perlu diketahui fungsi dari setiap kaki pada LCD.



Gambar 2.4-1 Modul LCD karakter 2x16

Dari gambar diatas dapat dijelasakan fungsi dari setiap kaki pada LCD, sebagai berikut:

1. Kaki 1 (GND)

   Kaki ini dihubungkan dengan tegangan +5 Volt yang merupakan tegangan untuk sumber daya dari HD44780 (khusus untuk modul M1632 keluaran hitachi, kaki ini adalah VCC).
2. Kaki 2 (VCC)

   Kaki ini dihubungkan dengan tegangan 0 volt (*ground*) dan modul LCD (khusus untuk modul M1632 keluaran hitachi, kaki ini adalah GND).
3. Kaki 3 (VEE/VLCD)

   Tegangan pengatur kontras LCD, kaki ini terhubung pada V5. Kontras mencapai nilai maksimum pada saat kondisi kaki ini pada tegangan 0 volt.
4. Kaki 4 (RS)

Register Select, kaki pemilih register yang akan diakses. Untuk akses ke register data, logika dari kaki ini adalah 1 dan untuk akses ke register perintah, logika dari kaki ini adalah 0.

5. Kaki 5 (R/W)

   Logika 1 pada kaki ini menunjukkan bahwa modul LCD sedang pada mode pembacaan dan logika 0 menunjukkan bahwa modul LCD sedang pada mode penulisan. Untuk aplikasi yang tidak memerlukan pembacaan data pada modul LCD, kaki ini dapat dihubungkan langsung ke *ground*.

6. Kaki 6 (E)

   Enable Clock LCD, kaki ini mengaktifkan clock LCD. Logika 1 pada kaki ini diberikan pada saat penulisan atau pembacaan data.

7. Kaki 7-14 (D0-D7)

   Data bus, kedelapan kaki modul LCD ini adalah bagian dimana aliran data sebanyak 4 bit atau 8 bit mengalir saat proses penulisan maupun pembacaan data.

8. Kaki 15 (Anoda)

   Berfungsi untuk tegangan positif dari *backlight* modul LCD sekitar 4,5 volt (hanya terdapat untuk M1632 yang memiliki *backlight*).

9. Kaki 16 (Katoda)

   Tegangan negatif *backlight* modul LCD sebesar 0 volt (hanya untuk M1632 yang memiliki *backlight*).

Display karakter pada LCD diatur oleh pin EN, RS dan R/W, dimana: Jalur EN dinamakan Enable. Jalur ini digunakan untuk memberitahu LCD bahwa data sedang dikirimkan. Untuk mengirimkan data ke LCD, maka melalui program EN harus dibuat logika low “0” dan set pada dua jalur kontrol yang lain RS dan RW. Ketika dua jalur yang lain telah siap, set EN dengan logika “1” dan tunggu untuk sejumlah waktu tertentu (sesuai dengan datasheet dari LCD tersebut) dan berikutnya set EN ke logika low “0” lagi.

Jalur RS adalah jalur Register Select. Ketika RS berlogika low “0”, data akan dianggap sebagai sebuah perintah atau instruksi khusus (seperti clear screen, posisikursor, dll). Ketika RS berlogika high “1”, data yang dikirim adalah data text yang akan ditampilkan pada display LCD. Sebagai

contoh, untuk menampilkan huruf “1” pada layer LCD maka RS harus diset logika high “1”, jalur R/W adalah jalur control Read/Write. Ketika RW berlogika low (0), maka informasi pada bus data akan dituliskan pada layar LCD. Ketika RW berlogika high ”1”, maka program akan melakukan pembacaan memori dari LCD. Sedangkan pada aplikasi umum pin R/W selalu diberi logika low “0”. Pada akhirnya, bus data terdiri dari 4 atau 8 jalur (bergantung pada mode operasi yang dipilih oleh user). Pada kasus bus data 8 bit, jalur diacukan sebagai DB0 s/d DB7.

Beberapa perintah dasar yang harus dipahami adalah inisialisasi LCD Character:

1. Function Set berfungsi untuk mengatur interface lebar data, jumlah dari baris dan ukuran font karakter.

RS	**R/W**	**DB7**	**DB6**	**DB5**	**DB4**	**DB3**	**DB2**	**DB1**	**DB0**
0	0	0	0	1	DL	N	F	X	X

Dimana:

X : Don’t care.

DL : Mengatur lebar data.

DL=1, Lebar data interface 8 bit (DB7 s/d DB0).

DL=0, Lebar data interface 4 bit (DB7 s/d DB4) Ketika menggunakan lebar data 4 bit, data harus dikirimkan dua kali.

N: Pengaktivan baris.

N=0, 1 baris.

N=1, 2 baris.

F: Penentuan ukuran font karakter.

F=0, 5x7.

F=1, 5x8.

2. Entry Mode Set berfungsi untuk mengatur increment/ decrement dan mode geser.

RS	**R/W**	**DB7**	**DB6**	**DB5**	**DB4**	**DB3**	**DB2**	**DB1**	**DB0**
0	0	0	0	0	0	0	1	I/D	S

Dimana:

I/D: Increment/ decrement dari alamat DDRAM dengan 1 ketika kode karakter dituliskan ke DDRAM.

I/D = “0”, decrement.

I/D= “1”, increment .

S: Geser keseluruhan display kekanan dan kekiri.

S=1, geser kekiri atau kekanan bergantung pada I/D.

S=0, display tidak bergeser.

3. Display On/ Off Cursor berfungsi untuk mengatur status display ON atau OFF, cursor ON/ OFF dan fungsi Cursor Blink.

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	1	D	C	B

Dimana:

D : Mengatur display

D = 1, Display is ON.

D = 0, Display is OFF.

Pada kasus ini data display masih tetap berada di DDRAM, dan dapat ditampilkan kembali secara langsung dengan mengatur D=1.

C : Menampilkan kursor.

C = 1, kursor ditampilkan.

C = 0, kursor tidak ditampilkan.

B : Karakter ditunjukkan dengan kursor yang berkedip.

B=1, kursor blink.

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	0	0	1

4. Clear Display berfungsi sebagi perintah untuk menghapus layar.
5. Geser Kursor dan Display berfungsi untuk menggeser posisi kursor atau display ke kanan atau kekiri tanpa menulis atau baca data display. Fungsi ini digunakan untuk koreksi atau pencarian display.

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0

0	0	0	0	0	1	S/C	R/L	X	X

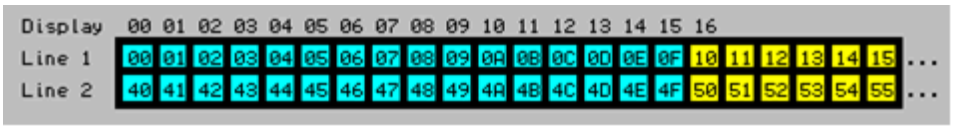
Catatan : x = Dont care

S/C	**R/L**	**Note**
0	0	Shift cursor position to the left
0	1	Shift cursor position to the right
1	0	Shift the entire display to the left
1	1	Shift the entire display to the right

2.4.2. Posisi Kursor

Modul LCD terdiri dari sejumlah memori yang digunakan untuk display. Semua text yang kita tuliskan ke modul LCd adalah disimpan didalam memori ini, dan modul LCD secara berurutan membaca memori ini untuk menampilkan text ke modul LCD itu sendiri.

Table 2.4-1 Peta Memori LCD



Pada peta memori tersebut, daerah yang berwarna biru (00 s/d 0F dan 40 s/d 4F) adalah display yang tampak. Sebagaimana yang dilihat, jumlahnya sebanyak 16 karakter per baris dengan dua baris. Angka pada setiap kotak adalah alamat memori yang bersesuaian dengan posisi dari layar. Demikianlah karakter pertama di sudut kiri atas adalah menempati alamah 00h. Posisi karakter berikutnya adalah alamat 01h dan seterusnya.

Akan tetapi, karakter pertama dari baris 2 sebagaimana yang ditunjukkan pada peta memori adalah pada alamat 40h. Demikianlah kita perlu untuk mengirim sebuah perintah ke LCD untuk mangatur letak posisi kursor pada baris dan kolom tertentu. Instruksi Set Posisi Kursor adalah 80h.

Untuk ini kita perlu menambahkan alamat lokasi dimana kita berharap untuk menempatkan kursor. Sebagai contoh, kita ingin menampilkan kata ”World” pada baris ke dua pada posisi kolom ke sepuluh. Sesuai peta memori, posisi karakter pada kolom 11 dari baris ke dua, mempunyai alamat 4Ah, sehingga sebelum kita tulis kata ”World” pada LCD, kita harus mengirim instruksi set posisi kursor, dan perintah untuk instruksi ini adalah 80h ditambah dengan alamat 80h+4Ah =0Cah. Sehingga dengan mengirim perintah Cah ke LCD, akan menempatkan kursor pada baris kedua dan kolom ke 11 dari DDRAM.

Set alamat memori DDRAM:

RS	**R/W**	**DB7**	**DB6**	**DB5**	**DB4**	**DB3**	**DB2**	**DB1**	**DB0**
0	0	1	A	A	A	A	A	A	A

Dimana:

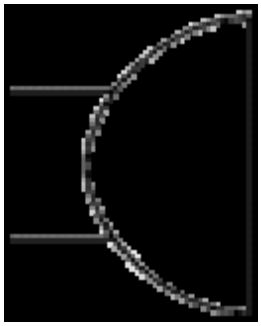
A : Alamat RAM yang akan dipilih, sehingga alamat RAM LCD adalah 000 0000 s/d 111 1111b atau oo s/d 7Fh.

2.5. BUZZER

Buzzer merupakan suatu komponen yang dapat menghasilkan suara yang mana apabila diberi tegangan pada input komponen, maka akan bekerja sesuai dengan karakteristik dari alarm yang digunakan. Dalam pembuatan proyek tugas akhir ini, penulis menggunakan “Buzzer” sebagai informasi suara. Hal ini dikarenakan karakteristik dari komponen yang mudah untuk diaplikasikan dan suara yang dihasilkan relatif kuat.

Buzzer merupakan sebuah komponen elektronik yang dapat mengkonversikan energi listrik menjadi suara yang di dalamnya terkandung sebuah osilator internal untuk menghasilkan suara dan pada buzzer osilator yang digunakan biasanya diset pada frekuensi kerja sebesar 400 Hz.

Dalam penggunaannya dalam rangkaian, buzzer dapat digunakan pada tegangan sebesar antara 6V sampai 12V dan dengan tipical arus sebesar 25 mA. Pada gambar 2.5-1 dapat dilihat simbol dari komponen buzzer.

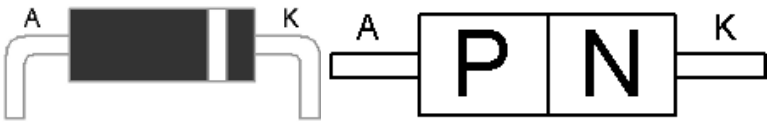


Gambar 2.5-1 Simbol Buzzer

2.6. Dioda

Dioda adalah komponen elektronika yang hanya memperbolehkan arus listrik mengalir dalam satu arah sehingga dioda biasa disebut juga sebagai "Penyearah"

Dioda terbuat dari bahan semikonduktor jenis *silicon* dan *germanium*. Dioda terbuat dari penggabungan dua tipe semikonduktor yaitu tipe P (*Positive*) dan tipe N (*Negative*), kaki dioda yang terhubung pada semikonduktor tipe P dinamakan "*Anode*" sedangkan yang terhubung pada semikonduktor tipe N disebut "*Katode*". Pada bentuk aslinya pada dioda terdapat tanda cincin yang melingkar pada salah satu sisinya, ini digunakan untuk menandakan bahwa pada sisi yang terdapat cincin tersebut merupakan kaki Katode. Arus listrik akan sangat mudah mengalir dari anoda ke katoda hal ini disebut sebagai "*Forward-Bias*" tetapi jika sebaliknya yakni dari katoda ke anoda, arus listrik akan tertahan atau tersumbat hal ini dinamakan sebagai "*Reverse-Bias*".



Gambar 2.6-1 Dioda

Jenis-Jenis dioda:

1. Diode Zener

   Ketika tegangan *reserve-bias* maksimum diberikan kepada dioda, maka arus listrik akan mengalir seperti layaknya pada keadaan *forward-bias*. Arus listrik ini tidak akan merusak dioda jika tidak melebihi dari apa yang telah ditentukan. Ketika tegangan

reserve-bias ini dapat dikendalikan pada level tertentu maka dioda ini disebut sebagai Dioda Zener.

2. LED (*Light Emitting Diodes*)

   LED merupakan jenis dioda yang jika diberikan tegangan forward-bias akan menimbulkan cahaya dengan warna-warna tertentu seperti merah, hijau, dan kuning.

3. Photodioda

   Photodioda adalah dioda yang bekerja berdasarkan intensitas cahaya, dimana jika photodioda terkena cahaya maka photodioda bekerja seperti dioda pada umumnya, tetapi jika tidak mendapat cahaya maka photodioda akan berperan seperti resistor dengan nilai tahanan yang besar sehingga arus listrik tidak dapat mengalir.

2.7. Kristal

Kristal lazimnya digunakan untuk rangkaian osilator yang menuntut stabilitas frekuensi yang tinggi dalam jangka waktu yang panjang. Alasan utamanya adalah karena perubahan nilai frekuensi kristal seiring dengan waktu, atau disebut juga dengan istilah faktor penuaan frekuensi (*frequency aging*), jauh lebih kecil dari pada osilator-osilator lain. Faktor penuaan frekuensi untuk kristal berkisar pada angka ±5*ppm*/tahun, jauh lebih baik dari pada faktor penuaan frekuensi osilator *RC* ataupun osilator *LC* yang biasanya berada diatas ±1%/tahun.

Kristal juga mempunyai stabilitas suhu yang sangat bagus. Lazimnya, nilai koefisien suhu kristal berada dikisaran ±50*ppm* direntangan suhu operasi normal dari -20°C sampai dengan +70°C. Bandingkan dengan koefisien suhu kapasitor yang bisa mencapai beberapa persen. Untuk aplikasi yang menuntut stabilitas suhu yang lebih tinggi, kristal dapat dioperasikan didalam sebuah oven kecil yang dijaga agar suhunya selalu konstan.



Gambar 2.7-1 Kristal

2.8. Kapasitor

Kapasitor (Kondensator) yang dalam rangkaian elektronika dilambangkan dengan huruf “C” adalah suatu alat yang dapat menyimpan energi/muatan listrik di dalam medan listrik, dengan cara mengumpulkan ketidakseimbangan internal dari muatan listrik.

Struktur sebuah kapasitor terbuat dari 2 buah plat metal yang dipisahkan oleh suatu bahan dielektrik. Bahan-bahan dielektrik yang umum dikenal misalnya udara vakum, keramik, gelas dan lain-lain. Jika kedua ujung plat metal diberi tegangan listrik, maka muatan-muatan positif akan mengumpul pada salah satu kaki (elektroda) metalnya dan pada saat yang sama muatan-muatan negatif terkumpul pada ujung metal yang satu lagi. Muatan positif tidak dapat mengalir menuju ujung kutub negatif dan sebaliknya muatan negatif tidak bisa menuju ke ujung kutub positif, karena terpisah oleh bahan dielektrik yang non-konduktif. Muatan elektrik ini tersimpan selama tidak ada konduksi pada ujung-ujung kakinya. Di alam bebas, phenomena kapasitor ini terjadi pada saat terkumpulnya muatan-muatan positif dan negatif di awan.berikut adalah jenis-jenis kapasitor:

1. Kapasitor Polar

   Sesuai dengan namanya kapasitor ini memiliki polaritas pada kedua kakinya yaitu polaritas positif (+) dan polaritas negatif (-). Kapasitor ini termasuk dalam kelompok kapasitor yang memiliki nilai kapasitas yang tetap dan memiliki nilai kapasitas yang besar.



Gambar 2.8-1 Kapasitor Polar

2. Kapasitor Variabel

   Kapasitor variabel adalah kapasitor yang nilai kapasitas-nya dapat diubah-ubah sesuai keinginan. Oleh karena itu kapasitor ini di kelompokan ke dalam kapasitor yang memiliki nilai kapasitas yang tidak tetap.



Gambar 2.8-2 Kapasitor Variabel

3. Kapasitor Nonpolar

   Kapasitor nonpolar merupakan jenis kapasitor yang memiliki kapasitas yang tetap, kapasitor ini memiliki kapasitas yang tidak terlalu besar serta tidak dibedakan antara kaki positif dan negatifnya.



Gambar 2.8-3 Kapasitor Nonpolar

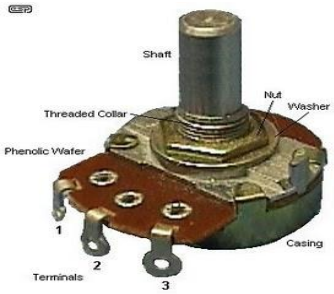
2.9. Resistor

Resistor adalah komponen elektronik yang didesain untuk menahan arus listrik dengan memproduksi penurunan tegangan diantara kedua salurannya sesuai

dengan arus yang mengalirinya.Resistor digunakan sebagai bagian elektronik dan sirkuit elektronik, dan merupakan salah satu komponen yang paling sering digunakan. Karakteristik utama dari resisitor adalah resistansinya dan daya listrik yang dapat diboroskan. Berikut adalah jenis-jenis dari resistor.

1. Variabel Resistor

   Resistor yang nilainya bisa diubah-ubah atau berubah. Trimpot yaitu variabel resistor yang nilai hambatannya dapat diubah dengan mengunakan obeng. Potensio yaitu variabel resistor yang nilai hambatannya dapat diubah langsung mengunakan tangan dengan cara memutar poros engkol atau mengeser kenop untuk potensio geser.



Gambar 2.9-1 Resistor Variabel

2. Non Linear Resistor

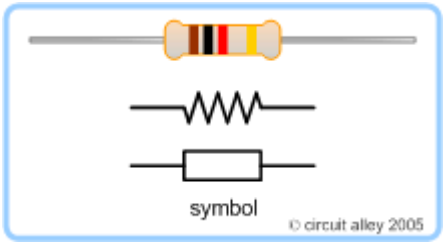
   Resistor yang nilai hambatannya tidak linier karena pengaruh faktor lingkungan misalnya suhu dan cahaya. Positive Temperatur Coefisien adalah jenis resistor non linier yang nilai hambatannya terpengaruh oleh perubahan suhu. Makin tinggi suhu yang mempengaruhi makin besar nilai hambatannya. Negative Temperatur Coefisien adalah jenis resistor non linier yang nilai hambatannya terpengaruh oleh perubahan suhu. Makin tinggi suhu yang mempengaruhi makin kecil nilai hambatannya. Light Dependent Resistor adalah jenis resistor non linier yang nilai hambatannya terpengaruh oleh perubahan intensitas cahaya yang mengenainya. Makin besar intensitas cahaya yang mengenainya makin kecil nilai hambatannya.



Gambar 2.9-2 Non Linier Resistor

3. Fixed Resistor

   Resistor yang nilainya tetap.



Gambar 2.9-3 Resistor Fixed

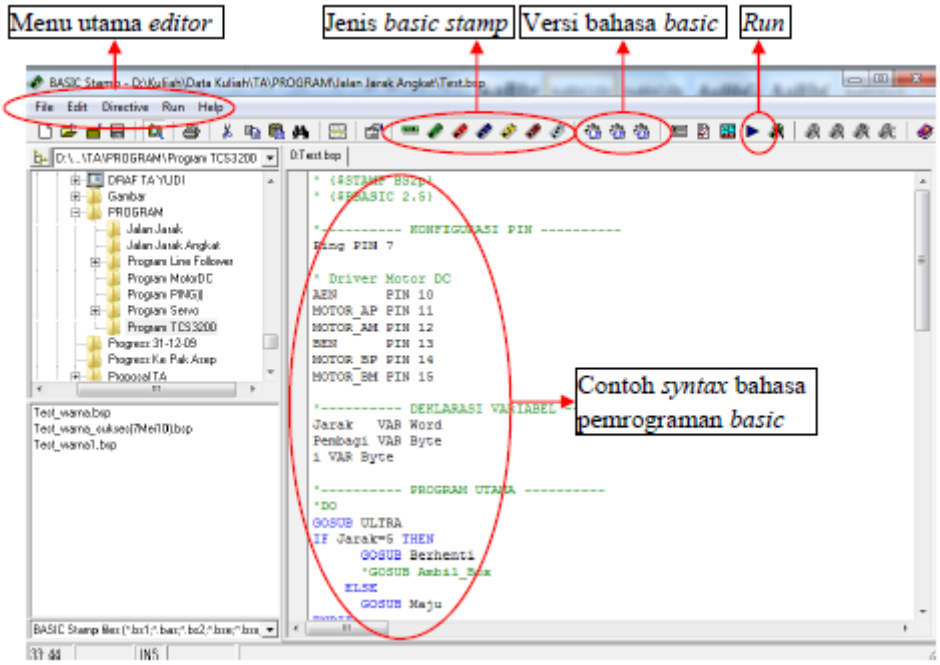
2.10. Pengenalan Basic Stamp Editor

Program dapat bermacam-macam bentuk versi dan bahasa pemrogramannya, sesuai dengan spesifikasi dari mikrokontroler yang digunakan. Mikrokontroler *basic stamp* (BS2P40) menggunakan bahasa pemrograman *basic*. *Software* yang digunakan adalah *basic stamp editor*. *Basic stamp editor* adalah sebuah editor yang dibuat oleh *Paralax Inc* untuk menulis program, meng*kompile* dan men*download*nya ke mikrokontroler keluarga *basic stamp*. Program ini memungkinkan penggunanya memprogram *basic stamp* dengan bahasa *basic* yang relatif ringan dibandingkan bahasa pemrograman lainnya. Berikut ini beberapa instruksi-instruksi dasar yang dapat digunakan pada mikrokontroler *basic stamp*.

Table 2.10-1 Beberapa Instruksi dasar Basic Stamp

Instruksi	Keterangan
DO...LOOP	Perulangan
GOSUB	Memanggil prosedur

IF..THEN	Percabangan
FOR...NEXT	Perulangan
PAUSE	Waktu tunda milidetik
IF...THEN	Perbandingan
PULSOUT	Pembangkit pulsa
PULSIN	Menerima pulsa
GOTO	Loncat ke alamat memori tertentu
HIGH	Menset pin I/O menjadi 1
LOW	Menset pin I/O menjadi 0
PWM	Konversi suatu nilai digital ke keluaran analog lewat pulse width modulasi



Gambar 2.10-1 Tampilan basic stamp editor

2.10.1. Memeriksa Sintaks Program

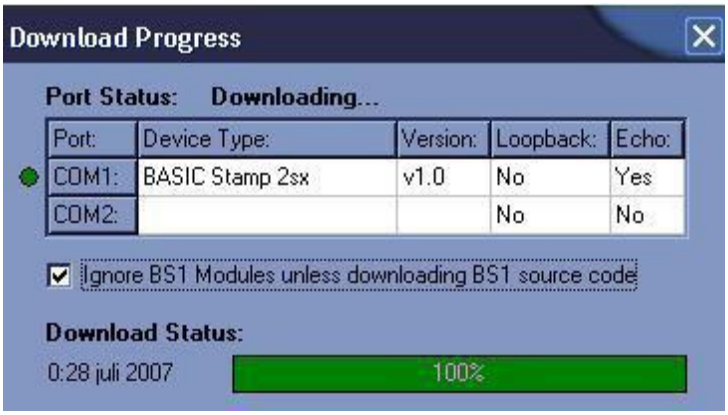
Sangat penting untuk memeriksa sintaks program, hal ini kita lakukan untuk memastikan semua sintaks sudah benar. Untuk memeriksa sintaks ini bisa pilih menu RUN, Cek Syntax atau kombinasi tombol CTRL+T. Berikut ini adalah tampilan jika listing program yang kita buat sudah benar.



Gambar 2.10-2 pemeriksaan sintaks yang sukses (tokenize successful)

2.10.2. Menjalankan Program

Setelah program selesai, program siap di download ke modul basic stamp. Cara untuk menjalankan program dapat memilih menu RUN atau kombinasi tombol CTR+R. Berikut adalah tampilan jika pen*downlod*an program sukses.



Gambar 2.10-3 Tampilan jika pendownloadan program sukses