BAB II
DASAR TEORI

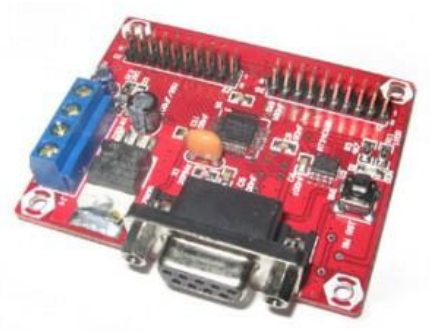
2.1. Perangkat Keras (*hardware*)

2.1.1.Mikrokontroler *Basic Stamp* (BS2P40) [7]

Basic stamp adalah suatu mikrokontroler yang dikembangkan oleh *Parallax Inc* yang diprogram menggunakan bahasa pemrograman *basic* dan populer sekitar pada tahun 1990an. Mikrokontroler *basic stamp* membutuhkan *power supply* saat men*download* dan program di *download* melalui *port serial*.

Mikrokontroler *basic stamp* memiliki beberapa versi yang berbeda – beda, yaitu *basic stamp* 1, *basic stamp* 2, *basic stamp* 1e, *basic stamp* 2P, *basic stamp* 2Pe dan *basic stamp* 2sx. *Basic stamp* bekerja pada tegangan DC 5 sampai 15 volt. *Basic stamp* yang dipakai adalah *basic stamp* BS2P40 yang mempunyai 40 pin I/O. Pemilihan basic stamp ini karena membutuhkan banyak *input* atau *output* yang dipakai dalam pengontrolan pra prototipe satelit.

Kode *basic* (pemograman *basic*) disimpan di dalam EEPROM (*Electrically Erasable Programmable Read-Only Memory*) serial pada *board basic stamp*. EEPROM digunakan dalam *basic stamp* 1 dan 2 yang dijamin menyimpan selama 40 tahun ke depan dan mampu ditulisi ulang 10 juta kali per lokasi memori. Berikut ini adalah tampilan *basic stamp* BS2P40.

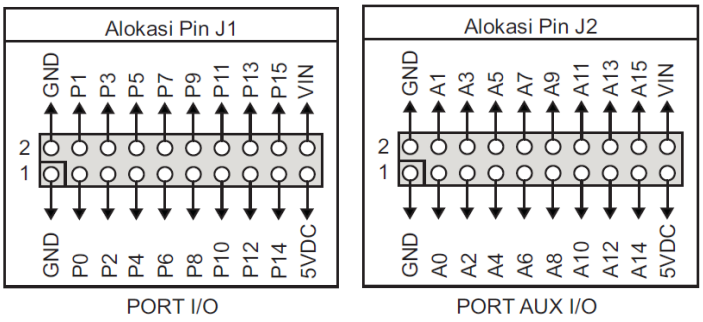


Gambar 2.1 Modul *Basic Stamp* (BS2P40)

Modul b*asic stamp* 2P40 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* atau *output* sebanyak 32 pin.
6. Tersedia jalur komunikasi serial UART RS-232 dengan konektor DB9.
7. Tegangan input 9 – 12 V_{DC} dengan tegangan output 5 V_{DC}.

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



Gambar 2.2 Alokasi Pin *Basic Stamp*

Adapun hubungan antara komputer dengan modul BS2P40 memiliki konfigurasi sebagai berikut :

Tabel 2.1 Hubungan Pin Antara Komputer Dengan BS2P40 (DB9)

COM Port Komputer DB9	Modul BS2p40 DB9
RX (Pin 2)	RX (Pin 2)
TX (pin 3)	TX (pin 3)
DTR (pin 4)	DTR (pin 4)
GND (pin 5)	GND (pin 5)
DSR (Pin 6)	DSR (Pin 6)
RTS (Pin 7)	RTS (Pin 7)

2.1.2. LCD (*Liquid Crystal Display*) [8]

Display LCD sebuah *liquid crystal* atau perangkat elektronik yang dapat digunakan untuk menampilkan angka atau teks. Ada dua jenis utama layar LCD yang dapat menampilkan numerik (digunakan dalam jam tangan, kalkulator dll) dan menampilkan teks *alfanumerik* (sering digunakan pada mesin foto kopi dan telepon genggam).

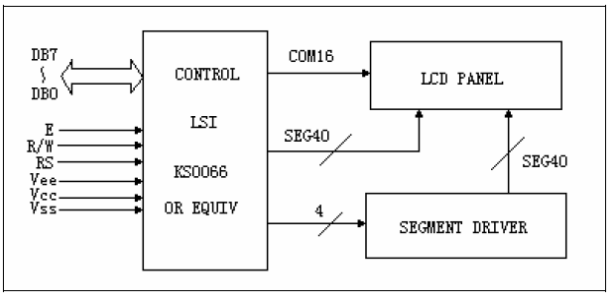
Dalam menampilkan numerik ini kristal yang dibentuk menjadi bar, dan dalam menampilkan *alfanumerik* kristal hanya diatur kedalam pola titik. Setiap kristal memiliki sambungan listrik individu sehingga dapat dikontrol secara independen. Ketika kristal *off* (yakni tidak ada arus yang melalui kristal) cahaya kristal terlihat sama dengan bahan latar belakangnya, sehingga kristal tidak dapat terlihat. Namun ketika arus listrik melewati kristal, itu akan merubah bentuk dan menyerap lebih banyak cahaya. Hal ini membuat kristal terlihat lebih gelap dari penglihatan mata manusia sehingga bentuk titik atau bar dapat dilihat dari perbedaan latar belakang.

Sangat penting untuk menyadari perbedaan antara layar LCD dan layar LED. Sebuah LED *display* (sering digunakan dalam radio jam) terdiri dari sejumlah LED yang benar-benar mengeluarkan cahaya (dan dapat dilihat dalam gelap). Sebuah layar LCD hanya mencerminkan cahaya, sehingga tidak dapat dilihat dalam gelap.

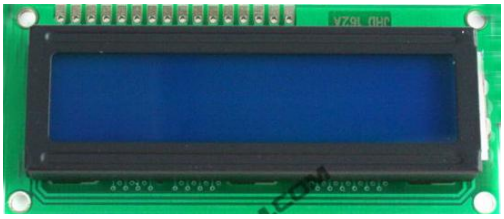
LMB162A adalah modul LCD matrix dengan konfigurasi 16 karakter dan 2 baris dengan setiap karakternya dibentuk oleh 8 baris *pixel* dan 5 kolom *pixel* (1 baris terakhir adalah kursor).

Memori LCD terdiri dari 9.920 bir CGROM, 64 byte CGRAM dan 80x8 bit DDRAM yang diatur pengalamatannya oleh *Address Counter* dan akses datanya (pembacaan maupun penulisan datanya) dilakukan melalui register data.

Pada LMB162A terdapat register data dan register perintah. Proses akses data ke atau dari register data akan mengakses ke CGRAM, DDRAM atau CGROM bergantung pada kondisi *Address Counter*, sedangkan proses akses data ke atau dari Register perintah akan mengakses *Instruction Decoder* (dekoder instruksi) yang akan menentukan perintah–perintah yang akan dilakukan oleh LCD.



(a)



(b)

Gambar 2.3 (a). Block Diagram LCD *(b).* LCD *16x2 Character*

Klasifikasi LED *Display 16x2 Character*

a. 16 karakter x 2 baris
b. 5x7 titik Matrix karakter + kursor
c. HD44780 Equivalent LCD kontroller/driver Built-In
d. 4-bit atau 8-bit MPU Interface
e. Tipe standar
f. Bekerja hampir dengan semua Mikrokontroler.

2.1.3. Karakter LCD

Tabel karakter LCD dibawah ini menunjukkan karakter khas yang tersedia pada layar LCD. Kode karakter diperoleh dengan menambahkan angka di atas kolom dengan nomor di sisi baris.

Perhatikan bahwa karakter 32-127 selalu sama untuk semua LCD, tapi karakter 16-31 & 128-255 dapat bervariasi dengan produsen LCD yang berbeda. Oleh karena itu beberapa LCD akan menampilkan karakter yang berbeda dari yang ditunjukkan dalam tabel.

Karakter 0 sampai 15 dijelaskan *user-defined* sebagai karakter dan harus didefinisikan sebelum digunakan, atau LCD akan berisi perubahan karakter secara acak. Untuk melihat secara rinci bagaimana menggunakan karakter ini dapat dilihat pada data *Character* LCD

Tabel 2.2 *Data CharacterLCD*

High-Order 4 bit / Low-Order 4 bit	0000	0010	0011	0100	0101	0110	0111	1010	1011	1100	1101	1110	1111
××××0000	CG RAM (1)		0	@	P	\	p		ー	タ	ミ	α	p
××××0001	(2)	!	1	A	Q	a	q	。	ア	チ	ム	ä	q
××××0010	(3)	"	2	B	R	b	r	「	イ	ツ	メ	β	θ
××××0011	(4)	#	3	C	S	c	s	」	ウ	テ	モ	ε	∞
××××0100	(5)	$	4	D	T	d	t	、	エ	ト	ヤ	μ	Ω
××××0101	(6)	%	5	E	U	e	u	•	オ	ナ	ユ	σ	ü
××××0110	(7)	&	6	F	V	f	v	ヲ	カ	ニ	ヨ	ρ	Σ
××××0111	(8)	'	7	G	W	g	w	ァ	キ	ヌ	ラ		π
××××1000	(1)	(	8	H	X	h	x	ィ	ク	ネ	リ	√	$\bar{x}$
××××1001	(2)	)	9	I	Y	i	y	ゥ	ケ	ノ	ル	-1	y
××××1010	(3)	*	:	J	Z	j	z	ェ	コ	ハ	レ	j	
××××1011	(4)	+	;	K	[	k	{	ォ	サ	ヒ	ロ	x	
××××1100	(5)	,	<	L	¥	l	|	ャ	シ	フ	ワ	¢	
××××1101	(6)	-	=	M	]	m	}	ュ	ス	ヘ	ン	£	÷
××××1110	(7)	.	>	N	^	n	→	ョ	セ	ホ	゛	$\bar{n}$	
××××1111	(8)	/	?	O	_	o	←	ッ	ソ	マ	゜	ö	■

2.1.4. Deskripsi Pin LCD

Untuk keperluan antar muka suatu komponen elektronika dengan mikrokontroler, perlu diketahui fungsi dari setiap kaki yang ada pada komponen tersebut.

a. Kaki 1 (GND) : Kaki ini berhubungan dengan tegangan +5 Volt yang merupakan tegangan untuk sumber daya.

b. Kaki 2 (VCC) : Kaki ini berhubungan dengan tegangan 0 volt (*Ground*).

c. Kaki 3 (VEE/VLCD) : Tegangan pengatur kontras LCD, kaki ini terhubung pada cermet. Kontras mencapai nilai maksimum pada saat kondisi kaki ini pada tegangan 0 volt.

d. 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.

e. Kaki 5 (R/W) : Logika 1 pada kaki ini menunjukan bahwa modul LCD sedang pada mode pembacaan dan logika 0 menunjukan bahwa modul LCD sedang pada mode penulisan. Untuk aplikasi yang tidak memerlukan pembacaan data pada modul LCD, kaki ini dapat dihubungkan langsung ke *Ground*.

f. Kaki 6 (E) : *Enable Clock LCD*, kaki mengaktifkan *clock* LCD. Logika 1 pada kaki ini diberikan pada saat penulisan atau membacaan data.

g. Kaki 7 – 14 (D0 – D7) : Data bus, kedelapan kaki LCD ini adalah bagian di mana aliran data sebanyak 4 bit ataupun 8 bit mengalir saat proses penulisan maupun pembacaan data.

h. Kaki 15 (Anoda) : Berfungsi untuk tegangan positif dari *backlight* LCD sekitar 4,5 volt (hanya terdapat untuk LCD yang memiliki *backlight*)

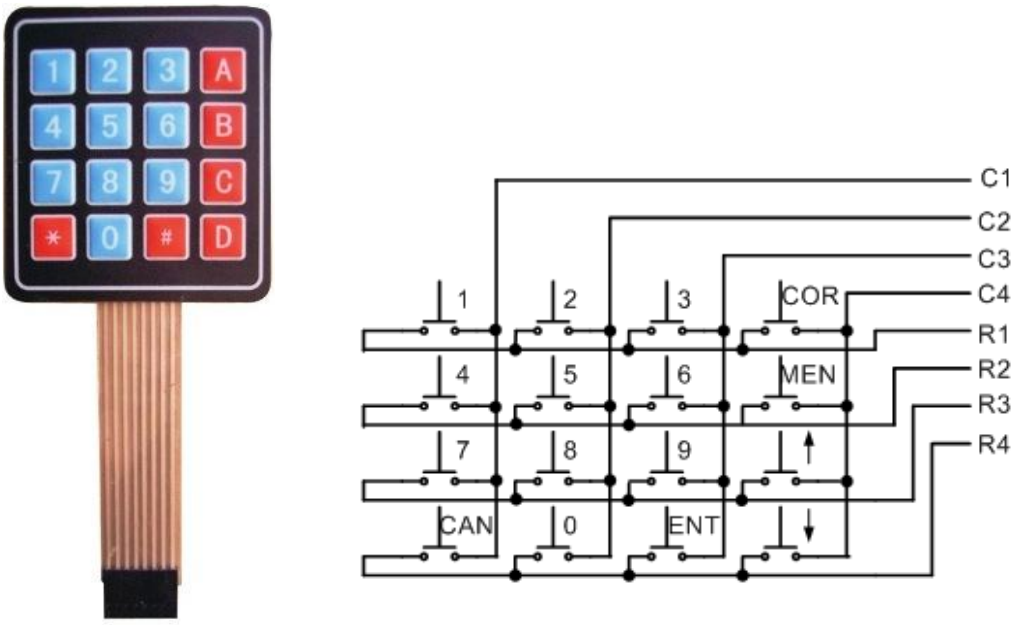
i. Kaki 16 (Katoda) : Tegangan negatif *backlight* LCD sebesar 0 volt (hanya terdapat pada LCD yang memiliki *backlight*).

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
VSS	VCC	VEE	RS	R/W	E	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	LED+	LED-

Gambar 2.4 Blok Pin LCD

2.1.5. *KeyPad* [9]

Keypad sering digunakan sebagi suatu input pada beberapa peralatan yang berbasis mikroprosessor atau mikrokontroller. *Keypad* terdiri dari sejumlah saklar, yang terhubung sebagai baris dan kolom dengan susuan seperti yang ditunjukkan pada gambar 2.5. Agar mikrokontroller dapat melakukan scan *keypad*, maka port mengeluarkan salah satu bit dari 4 bit yang terhubung pada kolom dengan logika low "0" dan selanjutnya membaca 4 bit pada baris untuk menguji jika ada tombol yang ditekan pada kolom tersebut. Sebagai konsekuensi, selama tidak ada tombol yang ditekan, maka mikrokontroller akan melihat sebagai logika *high* "1" pada setiap pin yang terhubung ke baris.



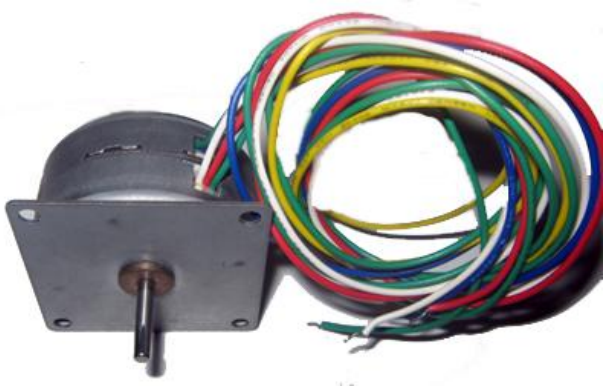
(a) (b)

Gambar 2.5 (a) Bentuk Fisik (b) Rangkaian dasar *keypad 4x4*

2.1.6. Motor *Stepper*

Motor *stepper* merupakan motor DC yang tidak memiliki komutator. Kecepatan motor *stepper* pada dasarnya ditentukan oleh kecepatan pemberian data pada komutatornya. Semakin cepat data yang diberikan maka motor *stepper* akan semakin cepat pula berputarnya. gerakan pada rotornya itu dapat dikendalikan oleh pulsa dari mikroprosessor.

Tidak seperti motor ac dan dc konvensional yang berputar secara kontinyu, perputaran motor *stepper* adalah secara *incremental* atau langkah per langkah (*step by step*). Gerakan motor *stepper* sesuai dengan pulsa-pulsa digital yang diberikan. Seperti halnya motor konvensional dc biasa, motor *stepper* juga dapat berputar dalam dua arah yaitu searah jarum jam (CW, *clockwise*) atau berlawanan arah jarum jam (CCW, *counterclockwise*) yaitu dengan memberikan polaritas yang berbeda. Suatu motor *stepper* mengkonversi sinyal elektrik ke dalam pergerakan (putaran) spesifik. Pergerakan yang diciptakan oleh sinyal masing-masing dapat diulang dengan tepat, itulah sebabnya mengapa motor *stepper* sangat efektif untuk aplikasi pergerakan posisi



Gambar 2.6 Bentuk fisik motor *stepper*

2.1.7. *Infra Red* [3]

Infra merah (*infra red*) ialah sinar elektromagnet yang panjang gelombangnya lebih daripada cahaya nampak yaitu di antara 700 nm dan 1 mm. Sinar infra merah merupakan cahaya yang tidak tampak. Jika dilihat dengan dengan spektroskop cahaya maka radiasi cahaya infra merah akan nampak pada spectrum elektromagnet dengan panjang gelombang di atas panjang gelombang cahaya merah. Dengan panjang gelombang ini maka cahaya *infra red* ini akan tidak tampak oleh mata namun radiasi panas yang ditimbulkannya masih terasa atau terdeteksi.

Infra red dapat dibedakan menjadi tiga daerah yakni :

a. *Infra red* dekat antara 0.75 - 1.5 µm.
b. *Infra red* menengah antara 1.50 - 10 µm.
c. *Infra red* jauh antara 10 - 100 µm.

Sifat-sifat cahaya *infra red* :

1). Tidak tampak secara langsung oleh mata manusia.
2). Tidak dapat menembus materi yang tidak tembus pandang.
3). Dapat ditimbulkan oleh komponen yang menghasilkan panas.

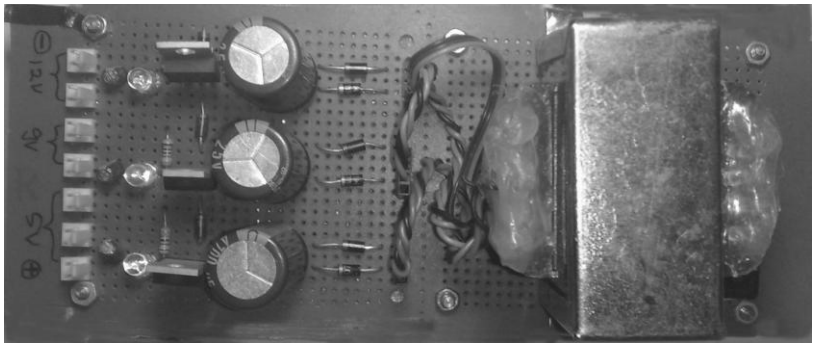


Gambar 2.7 Bentuk Fisik Rangkaian *Infra Red*

2.1.8. Catu Daya

Setiap rangkaian elektronik didesain untuk beroperasi pada tegangan tertentu dalam keadaan konstan. Regulator tegangan menyediakan *output* tegangan dc yang konstan dan secara terus menerus dapat menahan tegangan *output* pada nilai yang diinginkan. Regulator ini hanya dapat bekerja jika tegangan *input* (V_{in}) lebih besar daripada tegangan *output* (V_{out}).

Dalam hal ini sumber tegangan yang diperoleh berasal dari luar yang terhubung dengan mikrokontroler *basic stamp. Perangkat Lunak (Software)*



Gambar 2.8 Bentuk fisik *Regulator*

2.2. Perangkat Lunak *(Software)*

2.2.1. Pengenalan *Basic Stamp Editor* [4]

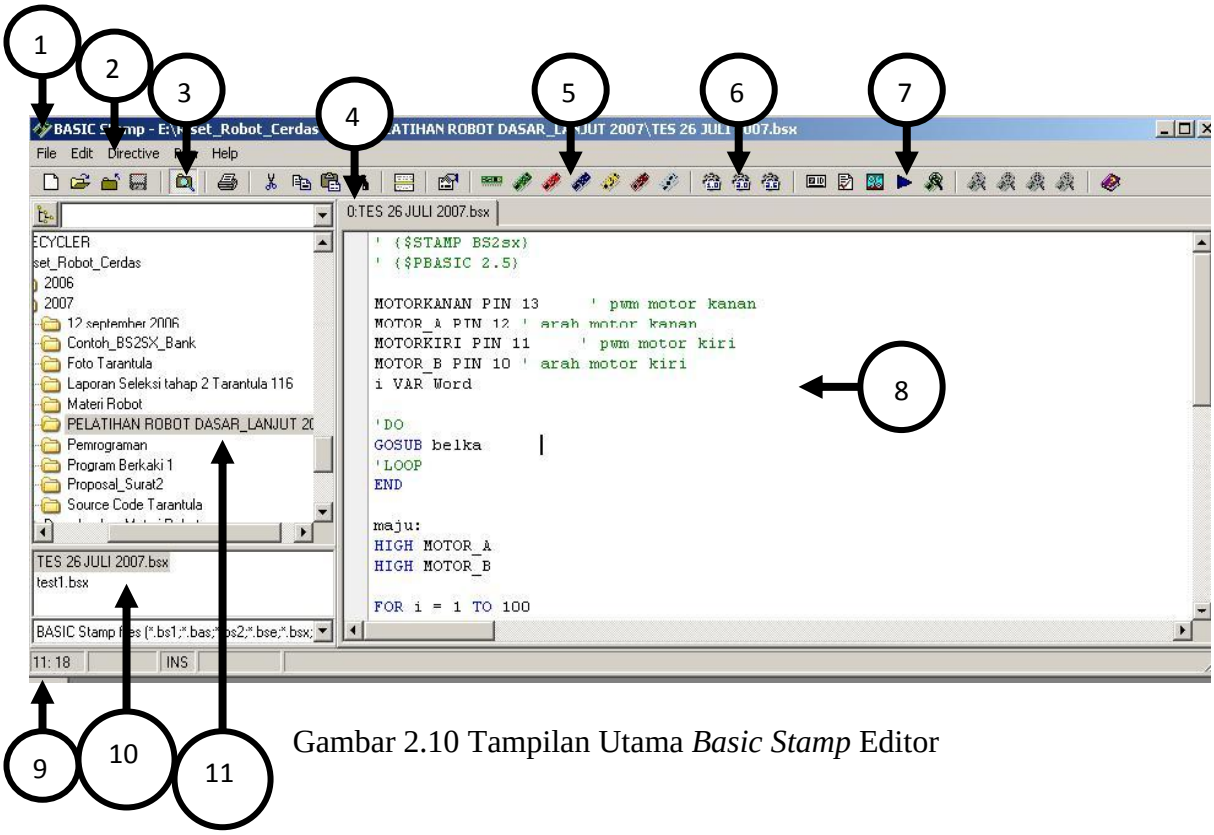
Basic Stamp editor adalah sebuah editor yang di buat oleh *Parallax Inc* untuk menulis program, meng*compile* dan men*download*nya ke mikrokontroler keluarga *Basic Stamp*. Bahasa pemrograman yang digunakan adalah bahasa *Basic*. Langkah-langkah untuk memulai menggunakan editor tersebut adalah sebagai berikut :

- **Klik *Start, Program, Parallax Inc, Basic Stamp Editor V 2.3***
- Selain itu bisa juga meng**klik *icon*** shortcut ***Basic Stamp Editor*** di desktop



Gambar 2.9 Menjalankan *BASIC Stamp Editor*

Setelah memulai untuk menjalankan editor tersebut, maka sekarang editor *BASIC Stamp* sudah jalan dan siap digunakan. Tampilan utamanya adalah sebagai berikut.



Gambar 2.10 Tampilan Utama *Basic Stamp* Editor

Keterangan gambar 2.11 :

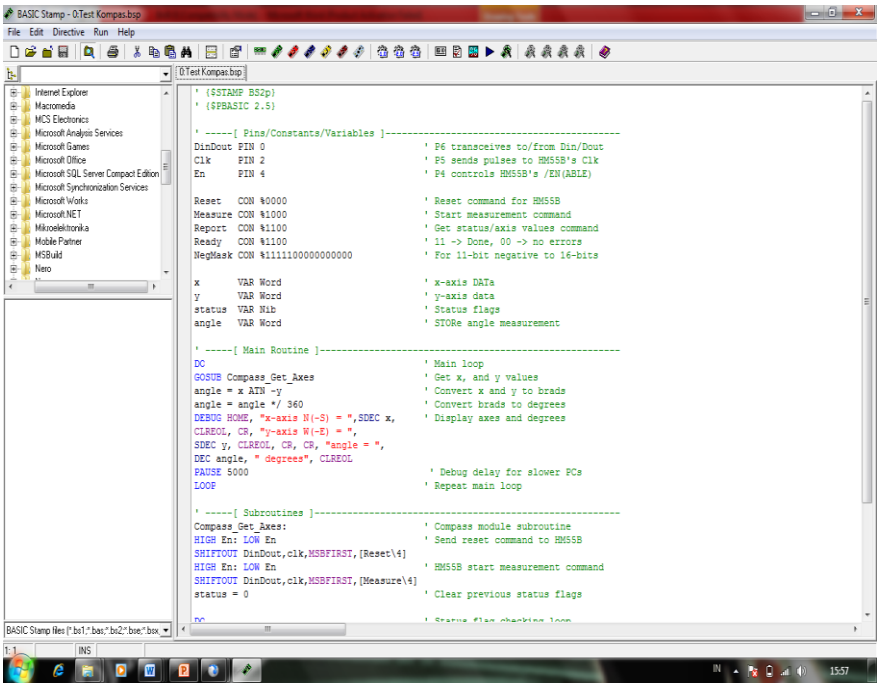
1. Nama editor, nama folder dan nama file yang sedang dibuka atau dikerjakan
2. Menu utama editor
3. *Shortcut* untuk menyimpan, cut, copy, paste, print dan lain-lain
4. Nama file yang sedang dikerjakan
5. Pemilihan jenis mikrokontroler yang digunakan
6. Pemilihan versi compiler PBASIC
7. Menjalankan program (Program RUN)
8. Area utama pengetikan program
9. Status posisi kursor berada (baris-kolom)
10. *List file-file* yang ada di *folder* kerja
11. *Folder* utama yang digunakan untuk menyimpan *file-file* kerja

Perangkat lunak merupakan faktor penting dalam tahap perancangan pra prototipe satelit. Perangkat lunak ini merupakan algoritma atau listing program yang ditanamkan kedalam mikrokontroler. Program dapat bermacam - macam bentuk dan bahasanya sesuai dengan spesifikasi dari mikrokontroler yang digunakan.

Mikrokontroler *basic stamp* (BS2P40) menggunakan bahasa pemrograman *basic*. *Software* yang digunakan adalah *basic stamp editor*. 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*.

Tabel 2.3 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*



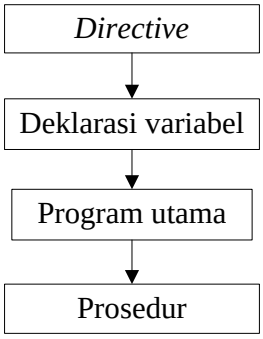
Gambar 2.11 Tampilan *Basic Stamp Editor*

2.2.2. Memprogram Basic Stamp

Dalam membuat sebuah program secara umum, dapat dibagi menjadi empat bagian penting, yaitu :

1. *Directive*
2. Deklarasi variabel
3. Program utama
4. Prosedur

Pemograman dalam *Basic Stamp Editor*, dapat dibagi menjadi empat bagian penting.



Gambar 2.12 Urutan Bagian dari Program Dalam B*asic Stamp*

2.2.2.1.*Directive*

Directive ditulis paling awal dari listing program yang dibuat. Bagian ini menentukan *tipe prosesor* yang digunakan dan *versi dari compiler PBASIC* yang digunakan untuk mengkompile *bahasa basic* menjadi bahasa mesin. Tampilannya adalah seperti gambar berikut :

```
' {$STAMP BS2sx}     ' pemilihan tipe prosesor
' {$PBASIC 2.5}      ' pemilihan versi kompiler PBASIC
```

Gambar 2.13 Tampilan Bagian *Directive*

2.2.2.2.Deklarasi Variabel

Beberapa ketentuan untuk mendeklarasikan variabel dalam mikrokontroler yaitu :

- PIN : PIN dari mikrokontroler (0-15)
- VAR : Variabel
- CON: Konstanta

PIN dalam BS2P40 yang digunakan sudah ditentukan sesuai dengan konfigurasi *hardware* atau *mainboard.*

```
' -----[ Pins/Constants/Variables ]-------
DinDout PIN 0
Clk     PIN 2
En      PIN 4

Reset   CON %0000
Measure CON %1000
Report  CON %1100
Ready   CON %1100
NegMask CON %1111100000000000

x       VAR Word
y       VAR Word
status  VAR Nib
angle   VAR Word
```

Gambar 2.14 Tampilan bagian deklarasi variable

2.2.2.3.Program Utama

Pada bagian program utama bisa melakukan dua mode, yaitu mode pengetikan langsung atau mode pemanggilan prosedur. Mode pengetikan langsung akan efektif jika program tidak terlalu banyak dan kasus yang dikerjakan sederhana. Tetapi jika program sudah mulai banyak atau rumit, maka sebaiknya program utama memanggil prosedur. Pemanggilan prosedur akan mempermudah dalam pemeriksaan dan lebih terkendali. Listing programnya dapat dilihat pada gambar berikut.

```
' -----[ Main Routine ]------------------
DO
GOSUB Compass_Get_Axes
angle = x ATN -y
angle = angle */ 360
DEBUG HOME, "x-axis N(-S) = ",SDEC x,
CLREOL, CR, "y-axis W(-E) = ",
SDEC y, CLREOL, CR, CR, "angle = ",
DEC angle, " degrees", CLREOL
PAUSE 5000
LOOP
```

Gambar 2.15 Tampilan Program Utama yang Memanggil Prosedur

2.2.2.4.Bagian Prosedur

Berikut adalah blok prosedur (*subroutines*) memperoleh data dari sumbu X dan Y yang dipanggil oleh program utama.

```
' -----[ Subroutines/procedure ]---------
Compass_Get_Axes:
HIGH En: LOW En
SHIFTOUT DinDout,clk,MSBFIRST,[Reset\4]
HIGH En: LOW En
SHIFTOUT DinDout,clk,MSBFIRST,[Measure\4]
status = 0

DO
HIGH En: LOW En
SHIFTOUT DinDout,clk,MSBFIRST,[Report\4]
SHIFTIN DinDout,clk,MSBPOST,[Status\4]
LOOP UNTIL status = Ready
SHIFTIN DinDout,clk,MSBPOST,[x\11,y\11]
HIGH En
IF (y.BIT10 = 1) THEN y = y | NegMask
IF (x.BIT10 = 1) THEN x = x | NegMask
RETURN
```

Gambar 2.16 Tampilan Bagian Prosedur

Sebuah prosedur harus mempunyai nama prosedur yang disimpan dibagian paling atas prosedur itu sendiri, serta harus diakhiri dengan *return* agar kembali lagi ke program utama dan melanjutkan kembali urutan program berikutnya.

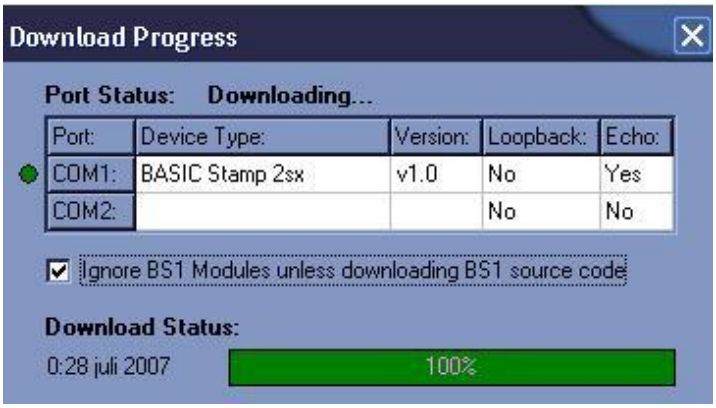
Untuk memeriksa sintaks program, hal ini kita lakukan untuk memastikan semua sintak sudah benar. Untuk memeriksa sintaks ini bisa pilih menu *RUN*, Cek *Syntax* atau kombinasi tombol CTRL+T. Berikut ini adalah tampilan jika listing program yang kita buat sudah benar.



Gambar 2.17 Hasil pemeriksaan sintaks yang sukses (*tokenize successful*)

2.2.3. 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 men*downlod* program sukses.



Gambar 2.18 Tampilan jika program sukses di *download*

2.3 Metode Least Square [10]

Metode Least Square atau Metode Kuadrat Terkecil digunakan untuk mendapatkan penaksir koefisien regresi linier. Model *regresi linier sederhana* dinyatakan dengan persamaan :

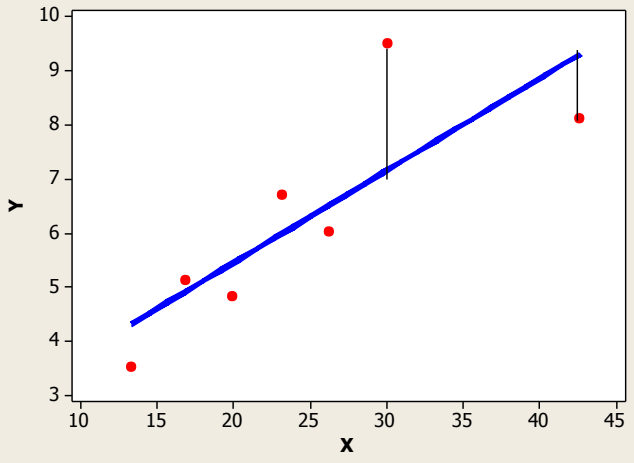
$$Y = \beta_0 + \beta_1 X + \varepsilon$$

Model dugaan dinyatakan oleh :

$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 X$ atau $\hat{Y} = b_0 + b_1$ X

b_1 adalah gradien A, b_0 adalah koefisien perpotongan garis pada sumbu y (B) dimana sumbu y = Ax + B

Secara geometrik, titik-titik hasil eksperimen model dan error digambarkan pada grafik berikut ini :



Gambar 2.19 Hasil eksperimen model dan error

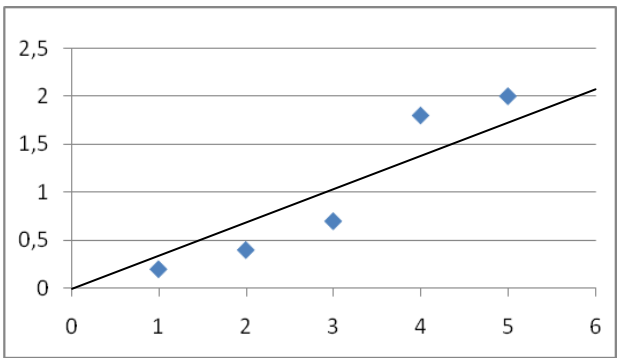
Titik-titik merah adalah nilai hasil eksperimen, dinotasikan Y_i, yang diduga membentuk garis lurus berwarna biru. Garis inilah model yang akan di-taksir, dengan cara menaksir koefisiennya, yaitu b_0 dan b_1, sehingga terbentuk persamaan $\hat{Y}_i = b_0 + b_1 X_i$.

Garis tegak lurus sumbu horisontal yang menghubungkan titik eksperimen dengan garis lurus dugaan dinamai *error*.

Metode least square bertujuan mendapatkan penaksir koefisien regresi, yaitu b_0 dan b_1, yang menjadi-kan jumlah kuadrat error, yaitu $\sum_{i=1}^{n} \varepsilon_i^2$ sekecil mungkin.

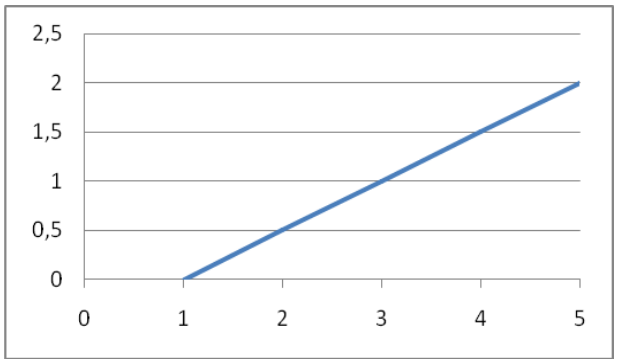
2.3.1. Membuat grafik linier dari percobaan menggunakan metode Least Square

Data yang kita dapat pada suatu percobaan umumnya tidaklah benar-benar linier (sebaran datanya acak)



Gambar 2.20 Sebaran data hasil percobaan

Sedangkan untuk memperoleh informasi Fisis kadang diperlukan data linier :



Gambar 2.21 Grafik garis linier

Ilmu statistika menyediakan sebuah metode untuk keperluan ini yang dikenal dengan metode least square. Hal ini karena sumber informasi fisis yang ingin kita dapatkan dari percobaan ini (misalnya percepatan grafitasi g, titik fokus sebuah lensa, kecepatan suara di udara dll) hanya mungkin didapat dari grafik jika grafik tersebut linier (biasanya diperoleh dari kemiringan dan titik potong grafik terhadap sumbunya).

Misalkan kita mempunyai pasangan data (hasil percobaan) y dengan x, maka kita dapat membuat suatu garis linier dalam sistem koordinat kartesian melalui persamaan linier berikut :

Dengan : $Y=Ax+B$ **(1)**

Ket :

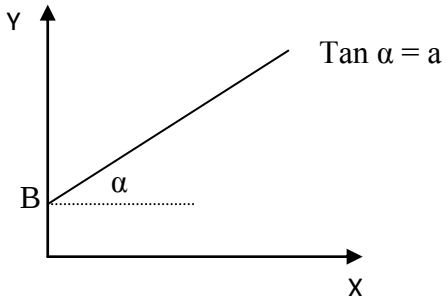
N = Banyak data percobaan
X = panjang tali bandul
Y = T^2

$$\mathbf{A} = \frac{\mathbf{N.}\sum\mathbf{xy}-(\sum\mathbf{x})(\sum\mathbf{y})}{\mathbf{N.}\sum\mathbf{x^2}-(\sum\mathbf{x})^2} \quad \textbf{(2)}$$

A secara grafis ditafsirkan sebagai kemiringan atau gradien atau juga tangen dari sudut suatu garis lurus dari sumbu x Positif.

Dan : $$\mathbf{B} = \frac{\mathbf{1}}{\boldsymbol{N}}\ (\mathbf{\Sigma y} - \mathbf{a\Sigma x}) \quad (3)$$

B tidak lain adalah titik potong (*interaction*) garis linier dengan sumbu y ,N adalah banyaknya percobaan atau banyaknya pasangan data x-y yang dilakukan dan indek i=1,2,3 …. N adalah data ke 1, ke 2, ke3…. data ke N.



Gambar 2.22 Titik potong garis linier pada sumbu Y

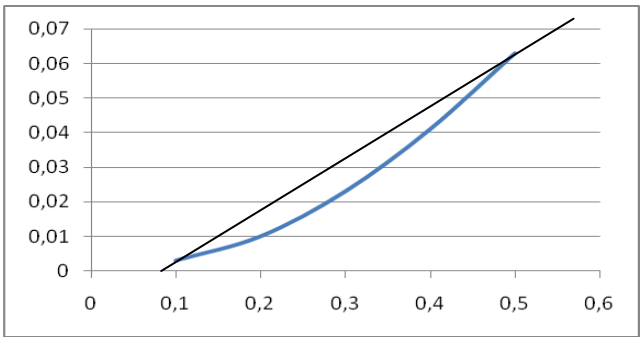
2.3.2 Contoh Data Percobaan Percepatan Gravitasi [5]

Misalkan dalam suatu percobaan kita mengukur lima kali akan didapatkan hasil percobaan sebuah data antara perioda T dengan panjang tali bandul L, untuk menentukan percepatan gravitasi bumi, sebagai berikut :

Tabel 2.4 Pengambilan data Gravitasi [5]

(i)	T (periodea) detik	L (panjang tali) meter
1	0.1	0.003
2	0.2	0.010
3	0.3	0.023
4	0.4	0.041
5	0.5	0.063

Ingin memakai gradien, jika telah pakai least square akan diperoleh 4 nilai gradien padahal yang dihasilkan itu umumnya nilai atau tidak mungkin percepatan gravitasi 4 buah pada satu lokasi



Gambar 2.23 Grafik tabel 2.4

Berapakah percepatan gravitasi pada tabel 2.4 di atas

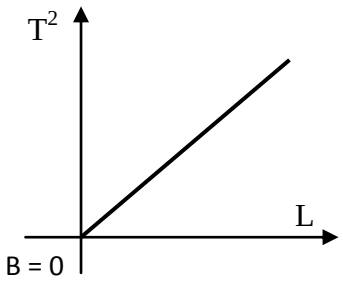
Jawab :

$$T^2 = \frac{4.\pi^2}{g} \text{L} \qquad (4)$$

Dengan metode least square :

1. Langkah pertama

   membandingkan rumusan di atas dengan persamaan y = A x + B. Kemudian variabel T^2 sama dengan sumbu Y dan variable L sama dengan sumbu x, A sebagai gradien sama dengan ($4\pi^2/g$) sehingga kita bias hitung g = ($4\pi^2/g$) sedangkan nilai B nya bernilai nol. Sehingga grafik yang diharapkan adalah seperti :



Gambar 2.24 Grafik yang diharapkan

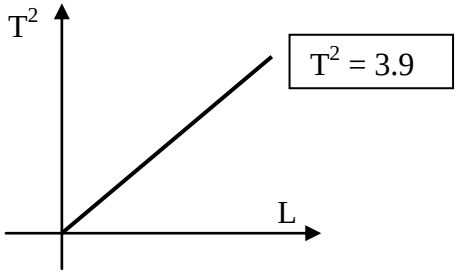
2. Langkah kedua menhitung gradient A kemiringan garis atau gradien yang disesuaikan dengan pesamaan (2) dimana x = L, Y = T^2

$$A = \frac{N.\sum LT^2-(\sum L)(\sum T^2)}{N.\sum L^2-(\sum L)^2}$$

3. Langkah ketiga memasukan data nilai $N.\sum xy-(\sum x)(\sum y) / N.\sum x^2-(\sum x)^2$ akhirnya didapatkan nilai A ≡ 3.9 untuk itu karena a = ($4\pi^2/g$), maka

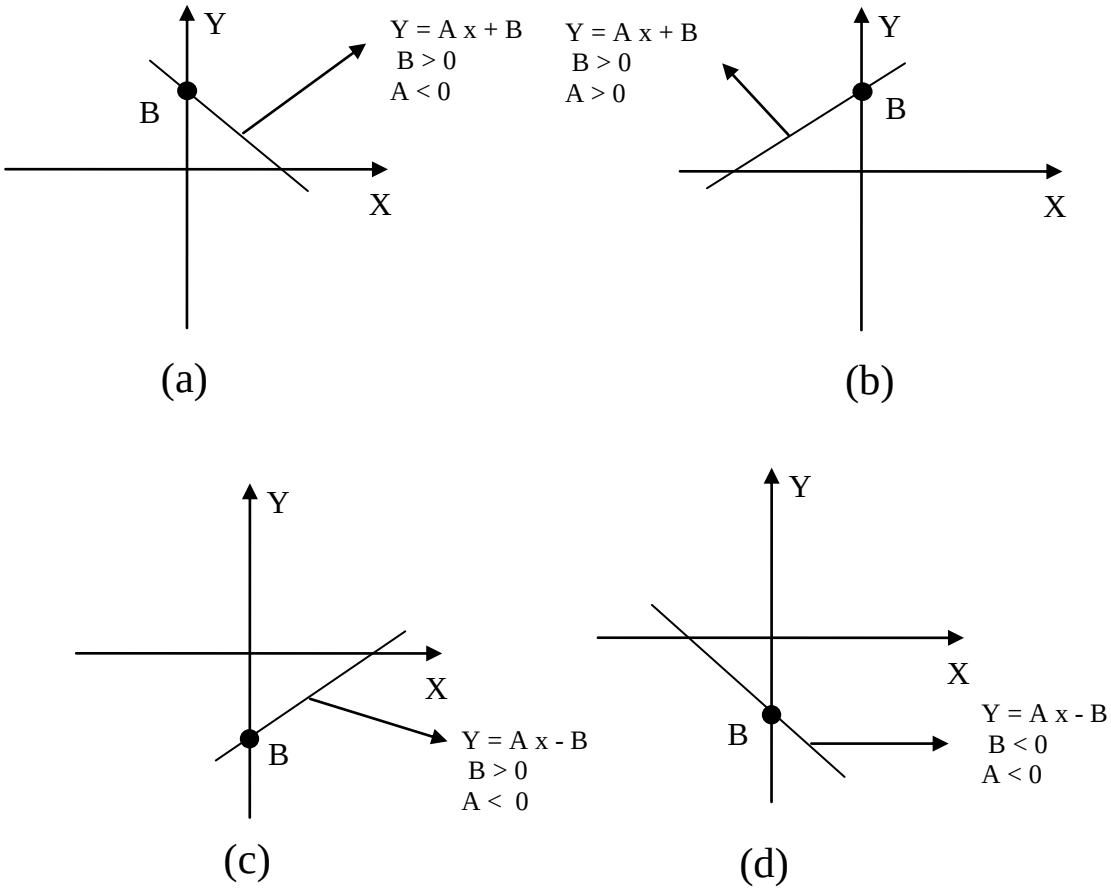
$$g = \frac{4\pi^2}{3.9} = 10{,}11 \text{ m/s}$$

Jadi percepatan gravitasi bumi dengan menggunakan pendekatan metoda Least Square adalah 10.11 m/s^2. dalam seketsa grafik sebagai berikut :



Gambar 2.25 Grafik pendekatan metoda Least Square

2.4. Persamaan Garis Lurus



Gambar 2.26 Persamaan garis lurus

Keterangan Gambar 2.26

a. Adalah gambar dimana titik potong melalui sumbu y positif dengan persamaan Y = A x + B B > 0 A < 0
b. Adalah gambar dimana titik potong melalui sumbu y positif dengan persamaan Y = A x + B B > 0 A > 0
c. Adalah gambar dimana titik potong melalui sumbu y negatif dengan persamaan Y = A x - B B > 0 A < 0
d. Adalah gambar dimana titik potong melalui sumbu y positif dengan persamaan Y = A x - B B < 0 A < 0

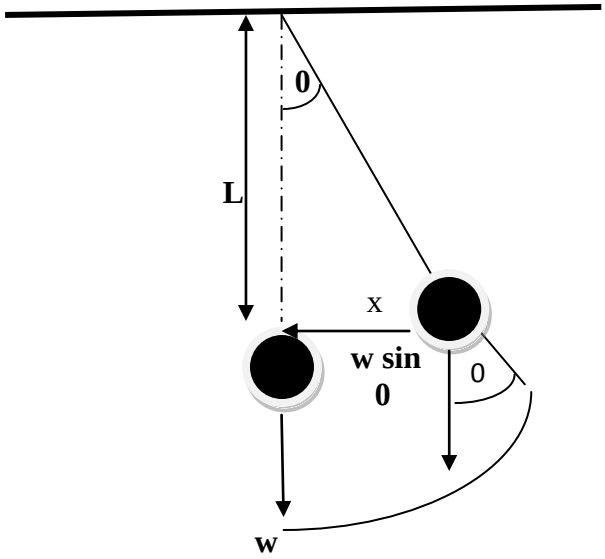
2.5 Gerak Harmonik Sederhana [2]

Bila suatu benda bergerak bolak balik terhadap suatu titik tertentu, maka benda tersebut dinamakan bergetar, atau benda tersebut bergetar. Dalam ilmu fisika dasar, terdapat beberapa kasus bergetar, diantaranya adalah gerak harmonik sederhana. Gerak Harmonik Sederhana adalah gerak bolak – balik benda melalui suatu titik keseimbangan tertentu dengan banyaknya getraran benda dalam setiap *second* selalu konstan.

Gerak Harmonik Sederhana terjadi karena *gaya pemulih (restoring force)*, dinamakan gaya pemulih karena gaya ini selalu melawan perubahan posisi benda agar kembali ke titik setimbang. Karena itulah terjadi gerak harmonik. Pengertian sederhana adalah bahwa kita menganggap tidak ada gaya disipatif, misalnya gaya gesek dengan udara, atau gaya gesesk antara komponen sistem (pegas dengan beban, atau pegas dengan setatipnya.

2.6 Metode Bandul [2]

Jika sebuah bandul diberi simpangan di sekitar titik setimbangnya dengan sudut ayunan Θ (dalam hal ini sudut Θ kecil), maka akan terjadi gerak harmonis, yang timbul karena adanya gaya pemulihan sebesar F = m-g-sinΘ yang arahnya selalu berlawanan dengan arah ayunan bandul.



Gambar 2.27 Sistem Bandul

$\sum F = m\ a$

Dalam arah x:

$- W.\ \sin\Theta = m\ \frac{d^2x}{dt^2}$

$- m\ .\ g\ .\ \sin\Theta = m\ \frac{d^2x}{dt^2}$ dengan menghilangkan m,

$- g\ .\ \sin\Theta = \frac{d^2x}{dt^2}$, untuk sudut Θ yang kecil, maka $\sin\Theta = \tan\Theta$

$- g\ .\ \tan\Theta = \frac{d^2x}{dt^2}$ $\tan\Theta = \frac{x}{L}$, sehingga:

$- g\ .\ \frac{x}{L} = \frac{d^2x}{dt^2}$ atau bisa dituliskan sebagai persamaan diferensial :

$\frac{d^2x}{dt^2} + \left(\frac{g}{L}\right)x = 0$, seperti halnya persamaan $\frac{d^2x}{dt^2} + \frac{k}{m}x = 0$ yang kemudian menghasilkan:

Frekuensi sudut $\omega = \sqrt{\frac{g}{L}}$, dimana $\omega = \frac{2\pi}{T}$, sehingga: $\frac{2\pi}{T} = \sqrt{\frac{g}{L}}$ atau:

$$\mathbf{T^2 = \frac{4\pi^2}{g}\ L}$$

Dari persamaan $T^2 = \frac{4\pi^2}{g}$ L dapat kita lakukan percobaan, dengan mengubah panjang tali L dan dengan mencatat periodenya setiap kali panjang L diubah, maka dengan menggunakan metode least square, dapat dihitung percepatan gravitasi bumi g.