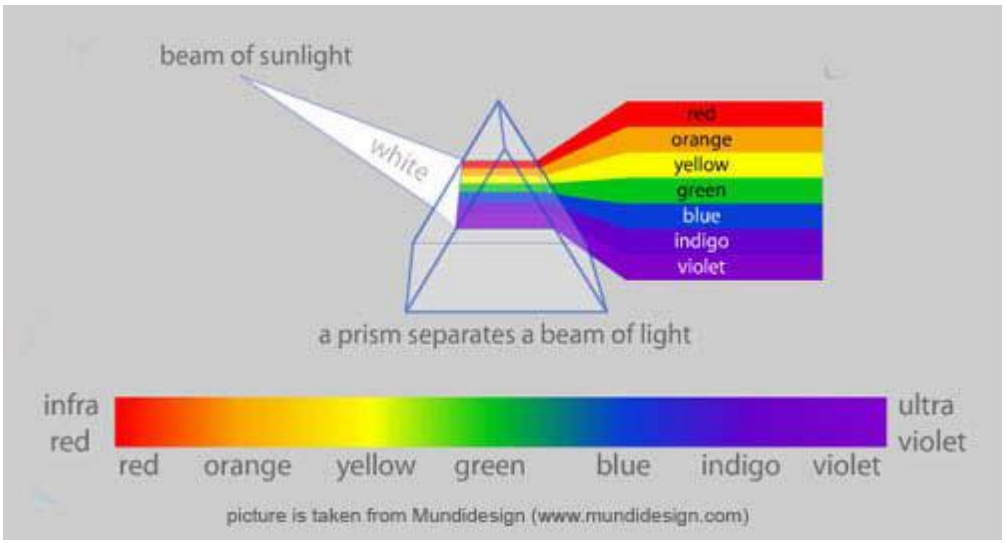
BAB II
TEORI PENUNJANG

2.1 Model Warna (Color Model)

Warna yang terlihat sebenarnya adalah spektrum cahaya yang dipantulkan oleh benda yang kemudian ditangkap oleh indra penglihatan lalu diterjemahkan oleh otak sebagai sebuah warna tertentu. Sebagai contoh ketika melihat warna hijau yang terdapat pada daun karena cahaya datang yang umumnya cahaya matahari dengan spektrum cahaya yang cukup kompleks diserap oleh daun, kemudian cahaya hijau yang terpantul inilah yang kita tangkap sehingga kita dapat melihat bahwa daun berwana hijau. jadi sebenarnya faktor penting bagi kita untuk melihat sebuah warna dengan baik adalah cahaya yang mengenai benda tersebut.

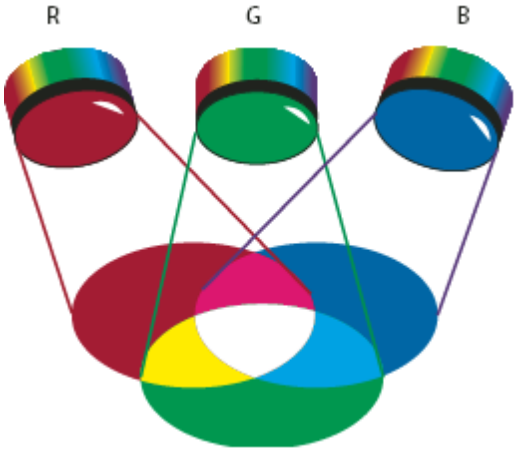


Gambar I.1 Spektrum Warna

Karena terkait dengan cahaya maka kita mengetahui bahwa tidak semua spektrum cahaya dapat ditangkap oleh indra penglihatan kita, karena itu kemudian timbul istilah spektrum cahaya tampak (visible spectrum) yang rangenya cukup besar. range inilah yang menjadi penyebab kita dapat melihat beraneka ragam warna yang secara umum dipisahkan menjadi beberapa spektrum dasar.

2.2 Model Warna RGB

Model warna RGB adalah model warna berdasarkan konsep penambahan kuat cahaya primer yaitu Red, Green dan Blue.

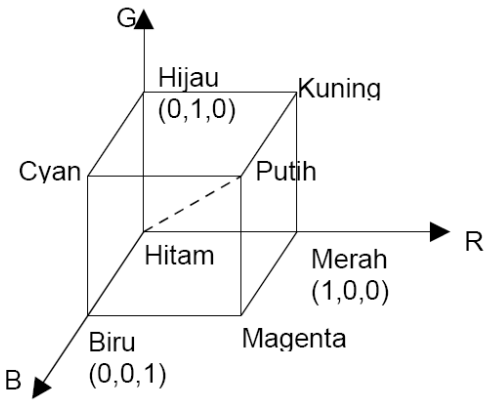


Gambar I.2 Model Warna RGB

Dalam suatu ruang yang sama sekali tidak ada cahaya, maka ruangan tersebut adalah gelap total. Tidak ada signal gelombang cahaya yang diserap oleh mata atau RGB (0,0,0). Apabila menambahkan cahaya merah pada ruangan tersebut, maka ruangan akan berubah warna menjadi merah misalnya RGB (255,0,0), semua benda dalam ruangan tersebut hanya dapat terlihat berwarna merah. Demikian apabila cahaya diganti dengan hijau atau biru.

Berdasar pada *tristimulus theory of vision* yang mengatakan bahwa manusia melihat warna dengan cara membandingkan cahaya yang datang dengan sensor-sensor peka cahaya pada retina. Sensor-sensor tersebut paling peka terhadap cahaya dengan panjang gelombang 630 nm (merah), 530 nm (hijau) dan 450 nm (biru). Model ini dapat digambarkan dengan kubus dengan sumbu-sumbu R, G dan B.

- Warna sudut kubus pada sumbu utama menyatakanwarna primer.
- Warna sudut kubus diluar sumbu utama menyatakan warna komplementer (merah dengan cyan, hijau dengan magenta, biru dengan kuning).
- Warna gray dinyatakan sepanjang diagonal hitam-putih.

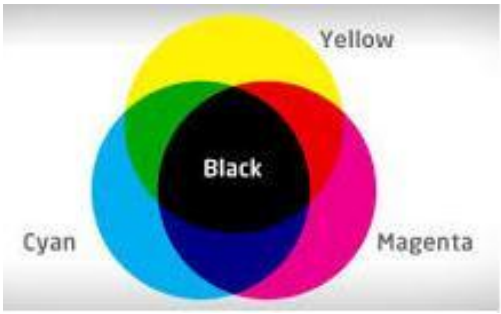


Gambar I.3 Pemodelan Warna RGB

Model warna RGB merupakan model *additive* dimana intensitas warna-warna primer dijumlahkan untuk mendapatkan warna lainnya. RGB model dapat digabung dengan halftoning untuk memperbanyak warna. Dalam sistem *on/off* RGB hanya ada 8 warna, dengan *halftoning* 2x2 didapat 125 macam warna (5 merah x 5 hijau x 5 biru).

2.2.1 Model Warna CMYK

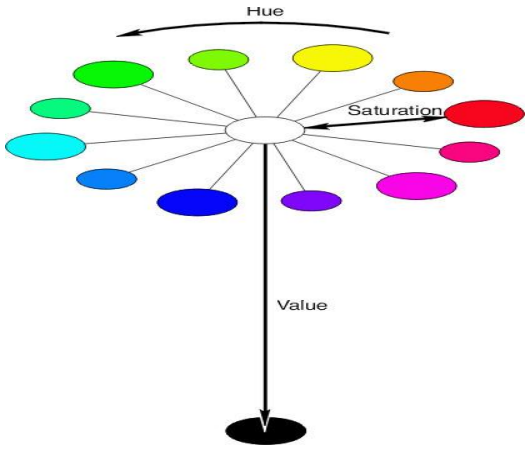
CMYK merupakan model warna yang mengacu pada empat warna pada tinta percetakan yang biasa digunakan, yaitu *cyan, magenta*, *yellow*(kuning) dan *key*(hitam). Warna hitam berfungsi untuk mengatur kontras atau kecerahan suatu warna. Sedangkan warna *cyan, magenta* dan *yellow* merupakan warna-warna *secondary* dari warna-warna dasar RGB. *Cyan* merupakan *secondary* dari warna merah hijau dan biru, sedangkan *magenta* merupakan *secondary* dari warna merah dan biru, kemudian warna *yellow* adalah *secondary* dari warna merah dan hijau.



Gambar I.4 Model Warna CMYK

2.2.2 Model Warna HSV

Model warna HSV mendefinisikan warna dalam terminologi Hue, Saturation dan Value. Hue menyatakan warna sebenarnya, seperti merah, violet, dan kuning. Hue digunakan untuk membedakan warna-warna dan menentukan kemerahan (redness), kehijauan (greeness), dsb, dari cahaya. Hue berasosiasi dengan panjang gelombang cahaya. Saturation menyatakan tingkat kemurnian suatu warna, yaitu mengindikasikan seberapa banyak warna putih diberikan pada warna. Value adalah atribut yang menyatakan banyaknya cahaya yang diterima oleh mata tanpa memperdulikan warna.



Gambar I.5 Model Warna HSV

Karena model warna HSV merupakan model warna yang diturunkan dari model warna RGB maka untuk mendapatkan warna HSV ini, kita harus melakukan proses konversi warna dari RGB ke HSV. HSV merupakan salah satu cara untuk mendefinisikan warna yang didasarkan pada roda warna. Hue mengukur sudut sekitar roda warna (merah pada 0 derajat, 120 derajat di hijau, biru, di 240 derajat). Saturation yang menunjukkan pada radius roda warna sehingga menunjukkan proporsi antara gelap (pusat) untuk warna ke putih murni (di luar).Value Menunjukkan nilai kecerahan. Hue memiliki nilai antara 0 hingga 360 (derajat), Saturation and Value berkisar dari 0 hingga 100%.

2.2.3 Model Warna YUV

Ide dasar dari model warna YUV adalah bertujuan untuk memisahkan informasi warna dari informasi kecerahan (brightness). Komponen-komponen luminan (Y) dan krominan (U,V) adalah:

$Y = 0{,}299R + 0{,}587G + 0{,}114B$

$U = 0{,}492(B - Y) = -0{,}147R - 0{,}289G + 0{,}436B$

$V = 0{,}877(R - Y) = 0{,}615R - 0{,}515G - 0{,}100B$

Y merupakan representasi dari luminan dari suatu image, sedangkan U,V berisi informasi warna, yaitu krominan. Komponen luminan dapat diperlakukan sebagai skala abu-abu dari suatu RGB image. Model warna YUV digunakan pada sistem televisi standar PAL. Kelebihan dari model warna YUV adalah:

- Informasi dari kecerahan (brightness) terpisah dengan informasi warna
- Korelasi antara komponen warna dapat dikurangi atau direduksi
- Banyak informasi yang dapat dikumpulkan melalui komponen Y, karena isi dari informasi U dan V berkurang. Dimana nampak berkebalikan dengan komponen Y adalah jauh lebih besar bila dibandingkan dengan komponen-komponen U dan V.

2.2.4 Model Warna YcbCr

YCbCr merupakan standar internasional bagi pengkodean digital gambar televisi yang didefinisikan di CCIR Recommendation 601. Y merupakan komponen luminance, Cb dan Cr adalah komponen chrominance. Pada monitor monokrom nilai luminance digunakan untuk merepresentasikan warna RGB, secara psikologis ia mewakili intensitas sebuah warna RGB yang diterima oleh mata. *Chrominance* merepresentasikan corak warna dan saturasi. Nilai komponen ini juga mengindikasikan banyaknya komponen warna biru dan merah pada warna (Cuturicu, 1999). YCbCr (256 level) dapat diperoleh dari RGB 8-bit dengan menggunakan rumus berikut:

$Y = 0{,}299\,R + 0{,}587 + 0{,}114\,B$

$Cb = -0{,}1687\,R\ 0.3313\,G + 0{,}5B + 128$

$Cr = 0.5\,R\ 0.4187G\ 0.0813B + 128$

2.3 Perangkat Keras (Hardware)

2.2.1 Mikrokontroler (Basic Stamp 2px)

Basic stamp adalah mikrokontroler yang dikembangkan oleh Parallax Inc yang diprogram menggunakan format bahasa pemrograman basic. Program yang telah dibuat akan diunduh melalui port serial. Kode pemograman basic disimpan didalam EEPROM serial pada board basic stamp. EEPROM menyediakan penyimpanan yang sulit diubah dan menjaga memory saat kehilangan power. EEPROM digunakan dalam basic stamp yang dijamin menyimpan data selama 40 tahun ke depan dan mampu ditulisi ulang 10.000.000 kali per lokasi memori. Mikrokontroler basic stamp memiliki beberapa versi yang berbeda-beda. Basic stamp yang digunakan adalah basic stamp 2px yang mempunyai 16 pin I/O. Berikut ini adalah tampilan basic stamp 2px.

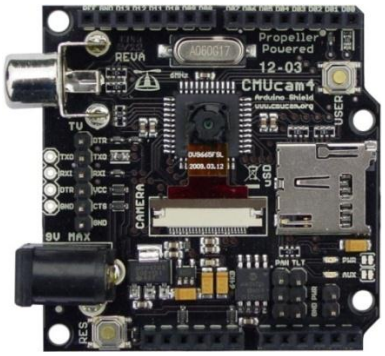


Gambar I.6 BS2px

2.2.2 Sensor Kamera CMUCam4

CMUCam4 adalah modul sensor kamera digital yang telah dilengkapi *image processing* internal. CMUCam4 merupakan versi terbaru dari CMUCam *series*. Fitur yang dimiliki CMUCam4 antara lain:

- memiliki dua port servo(*pan and tilt*).
- video output dengan resolusi 160x120 untuk visualisasi dari proses tracking.
- Penyimpanan data uSD.uSDHC Flash Card dengan format FAT16//32.
- Komunikasi: TTL UART(115200 baud – 19200 baud default).
- Raw image dumps over serial or to flash card.



Gambar I.7 Modul sensor kamera CMUCam4

2.2.3 Driver Motor

Driver motor yang berfungsi untuk mengontrol kecepatan dan arah putaran motor DC. Driver yang digunakan adalah L293 H-Bridge dapat men-*drive* 2 motor serta dapat mengontrol kecepatan dengan memberikan PWM (*pulse witdh modulation*).



Gambar I.8 Driver L293

2.2.4 Motor DC

Motor DC digunakan untuk pergerakan robot sehingga mudah dalam navigasi. Jenis motor DC yang digunakan adalah DC motor speed gearbox memiliki speed 10rpm-300rpm dengan torque 0.5Kg – 3Kg.cm dan tegangan 3Volt-12Volt.



Gambar I.9 Motor DC

2.2.5 LED

Light emitting diode (LED) atau dioda pemancar cahaya merupakan sebuah jenis dioda yang dapat memancarkan cahaya apabila diberikan tegangan 1.8 *volt* dengan arus sebesar 1.5 mA. LED pada robot digunakan sebagai indikator untuk setiap objek yang terdeteksi.



Gambar I.10. LED (Light Emitting Diode)

2.2.6 Catu Daya

Catu daya merupakan bagian penting dalam perancangan robot. Penentuan sistem catu daya yang akan digunakan dipertimbangan kebutuhan tegangan dan arus. Pada perancangan robot menggunakan baterai *lithium polymer* (LiPo). Baterai ini dapat diisi ulang (*rechargeable*). Baterai yang digunakan memiliki tegangan 7,4 Volt dan arus sebesar 2200 mAh dengan 3 *cell* di dalamnya. *Cell* merupakan teknologi konversi energi elektrokimia yang mampu mengubah senyawa hidrogen dan oksigen menjadi air, dan dalam prosesnya menghasilkan listrik. Berikut ini adalah gambar *lithium polymer* yang digunakan.



Gambar I.11 Lithium Polymer

2.2.7 Objek (Lampu RGB)

Objek yang akan dideteksi adalah sebuah lampu yang terdiri dari tujuh LED RGB sehingga dapat diatur keluaran warnanya dengan mengkombinasikan tiga tegangan masukan, yaitu merah, hijau dan biru. Untuk menyalakan lampu ini dibutuhkan tegangan 5 Volt. Contoh untuk mendapatkan keluaran warna kuning maka warna merah dan warna hijau harus diberi tegangan. Berikut ini gambar objek yang digunakan sebagai target.



Gambar I.12 Objek Lampu RGB

2.4 Perangkat Lunak (Software)

2.2.1 Pengenalan Basic Stamp Editor

Mikrokontroler basic stamp BS2px menggunakan bahasa pemrograman basic. Software yang digunakan adalah basic stamp editor. Basic stamp editor adalah sebuah editor yang dibuat oleh Paralax Inc untuk menulis listing program, meng-compile 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.

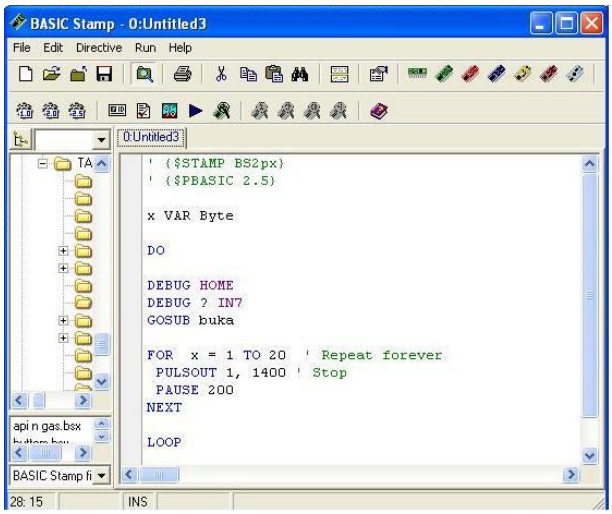
Tabel I.1 Beberapa instruksi dasar Basic stamp

Instruksi	Keterangan
DO...LOOP	Perulangan
GOSUB	Memanggil prosedur
FOR...NEXT	Perulangan
PAUSE	Waktu tunda milidetik
IF...THEN	Perbandingan
PWM	Mengatur kecepatan
I2CIN	Penerima data dari interface I2C
I2COUT	Pengirim data ke interface I2C
GOTO	Loncat ke alamat memori tertentu
HIGH	Menset pin I/O menjadi 1
LOW	Menset pin I/O menjadi 0
SEROUT	Mengirim data secara serial
SERIN	Menerima data secara serial

2.2.2 Memprogram Basic Stamp

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

1. Header
2. Variabel
3. Program utama
4. Prosedur



Gambar I.13 Tampilan Basic Stamp Editor

2.3.1.1 Directive

Directive ditulis paling awal listing program yang dibuat. Bagian ini menentukan tipe prosesor yang digunakan dan versi dari kompiler *PBASIC* yang digunakan untuk meng-kompile bahasa basik menjadi bahasa mesin. Tampilannya adalah seperti gambar berikut :

```
' {$STAMP BS2px}   'pemilihan jenis mikrokontroller
' {$PBASIC 2.5}    'pemilihan kompiler
```

Gambar I.14 Tampilan bagian Directive

2.3.1.2 Menentukan Variabel

Beberapa ketentuan untuk mendeklarasikan variabel dalam mikrokontroler yaitu :

1. PIN : PIN dari mikrokontroler (0-15)
2. VAR : Variabel
3. CON : Konstanta

```
Dout          PIN 4
Cs            PIN 2
clock         PIN 6
'========pin untuk motor kiri==============
en_kiri       PIN 5
in1_kiri      PIN 7
in2_kiri      PIN 9
'========pin untuk morot kanan=============
en_kanan      PIN 11
in1_kanan     PIN 13
in2_kanan     PIN 15
'========pin untuk sensor kompas===========
SDA           PIN 0
SCL           PIN 1
'======== pin pengaktif ================
aktif         PIN 8
'======== pin optocoupler ==============
opto          PIN 3
'======== konstanta ======================
WRITE_DATA    CON     $3C  'Used to perform a Write operation
READ_DATA     CON     $3D  'Used to perform a Read operation
MODE          CON     $02  'Read/Write Register, Selects the operating mode. Default = Single measurement
                           'Send $3C $02 $00 on power up to change to continuous measurement mode.
X_MSB         CON     $03  'Read Register, Output of X MSB 8-bit value.
X_LSB         CON     $04  'Read Register, Output of X LSB 8-bit value.
Z_MSB         CON     $05  'Read Register, Output of Z MSB 8-bit value.
Z_LSB         CON     $06  'Read Register, Output of Z LSB 8-bit value.
Y_MSB         CON     $07  'Read Register, Output of Y MSB 8-bit value.
Y_LSB         CON     $08  'Read Register, Output of Y LSB 8-bit value.
Cont_Mode     CON     $0
SCALEFACTOR   CON     4081 'SCALEFACTOR = 65536 * (255/4095) = 4081
SCALEOFFSET   CON     2048 'The output range of the HMC5883L is -2048 to 2047; therefore to get a 0 (LOWRANGE)to 4095 range
                           'to 4095 (HIGHRANGE) range 2048 must be added to the sensor value.
ATNFACTOR     CON     127  'the ATN function input range is -127 TO 127. In order to get the positive scaled range of
                           '0 to 255 into the range of -127 to 127, 127 must be subtracted.
LOWRANGE      CON     0    'see SCALEOFFSET
```

Gambar I.15 Tampilan bagian deklarasi variable

2.3.1.3 Bagian 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 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.

```
PAUSE 10

  I2COUT SDA, WRITE_DATA, MODE, [Cont_Mode]  ' mode continuous output command
main:
DO
GOSUB baca_kompas
      DEBUG HOME, "RAW X,Y,Z", CR
      DEBUG       "X = ",11, SDEC x, CR
      DEBUG       "Y = ",11, SDEC y, CR
      DEBUG       "Z = ",11, SDEC z, CR         ' Print angle
      DEBUG CR
  GOSUB GetAzimuth
      DEBUG       "SCALED X,Y", CR
      DEBUG       "X = ",11, SDEC x, CR
      DEBUG       "Y = ",11, SDEC y, CR

    azimuth = x ATN -y

      DEBUG "Azimuth ",11, SDEC azimuth, CR ' display scaled azimuth
      DEBUG "COS Azimuth ",11, SDEC COS azimuth, CR ' display scaled cos azimuth

  brads = azimuth     'get angle
  degr = brads */ 361 'convert to degrees
      DEBUG CR
      DEBUG "Heading in NORTH, EAST, SOUTH, and WEST clockwise from NORTH:", CLREOL, CR, CR
      DEBUG "Degree " ,11, SDEC degr, CR 'display degreesGOSUB adc
'==================== program sensor kompas====================
IF ((degr>=189) AND (degr<=191) OR (degr>=269) AND (degr<=271))THEN
GOSUB maju
ELSEIF (degr>=192) OR (degr>=271) THEN
GOSUB kiri_dikit
ELSEIF (degr<=187) OR (degr>=269) THEN
GOSUB kanan_dikit

      DEBUG HOME,
            "cm: ", DEC3 cm, CR
PAUSE 50
```

Gambar I.16. Tampilan bagian program utama

2.3.1.4 Memeriksa Sintaks Program

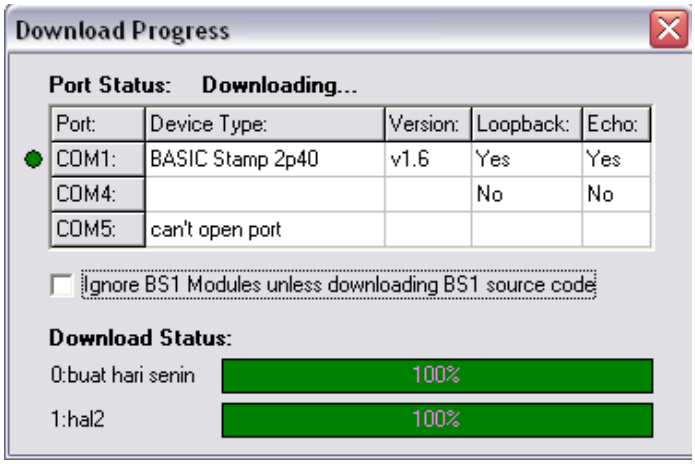
Sangat penting untuk memeriksa sintaks program, hal ini kita lakukan untuk memastikan semua *syntax* sudah benar. Untuk memeriksa sintaks ini bisa pilih menu *RUN*, Cek *Syntax* atau kombinasi tombol CTRL+T. Berikut ini adalah tampilan jika *listing* program yang kita buat sudah benar.



Gambar I.17. Hasil pemeriksaan syntax yang sukses (tokenize successful)

2.3.1.5 Download 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 I.18. Tampilan jika program sukses di-download