BAB II

LANDASAN TEORI

2.1 Barcode Batang dan Barcode Dua Dimensi

Penerapan informasi dalam kehidupan sehari-hari, misalnya pada produk makanan, obat-obatan, tiket pesawat, kartu tanda penduduk bahkan kartu tanda mahasiswa pun dapat ditemukan barcode, selaian itu juga barcode juga sering di manfatkan untuk proses pemeriksaan barang-barang secara otomatis pada pasar-pasar swalayan.[10]

Pada dasarnya barcode terdiri atas susunan garis vertical hitam (bar) dan putih (spasi) dengan ketebalan yang berbeda-beda. Kode batang juga dapat digambarkan dengan angka 1 untuk melambangkan garis hitam dan 0 untuk garis putih. Misalnya 0011001 mepresentasikan spasi-spasi-garis-garis-spasi-spasi-garis. Garis-garis ini digambarkan berderet secara horizontal dan merupakan representasi karakter-karakter *alpha-numerik* (*alphabet* dan numerik), untuk membantu pembacaan barcode secara manual, biasanya dicantumkan juga angka-angka atau huruf-huruf di bawah kode baris tersebut.

Barcode pertama kali diperkenalkan oleh dua orang mahasiswa Drexel Institute of Technology Bernard Silver dan Norman Joseph Woodland di tahun 1948. Mereka mempatenkan inovasi tersebut pada tahun 1949 dan permohonan tersebut dikabulkan pada tahun 1952, tapi baru pada tahun 1996, penemuan mereka digunakan dalam dunia komersial, pada kenyataannya penggunaannya tidak begitu sukses hingga pasca 1980an. Kode batang dapat dibaca oleh pemindai optik yang disebut pembaca kode batang atau barcode reader. Beberapa contoh dari barcode satu dimensi adalah sebagai berikut:

- Code 39 / 3 of 9
- Code 128
- Dan lain-lain

Bersamaan dengan pesatnya penggunaan barcode, kini barcode tidak hanya bisa mewakili karakter angka saja tapi sudah meliputi seluruh kode ASCII.

Kebutuhan akan kombinasi kode yang lebih rumit itulah yang kemudian melahirkan inovasi baru berupa kode matriks dua dimensi (2D barcodes) yang berupa kombinasi kode matriks bujur sangkar. Menunjukkan perbedaan utama antara barcode satu dimensi dan barcode dua dimensi.



Gambar 2.1. Barcode Dua Dimensi dan Barcode Batang

Barcode dua dimensi adalah lebih baik daripada barcode satu dimensi dari segi kapasitas data. Barcode dua dimensi tidak seperti barcode satu dimensi yang perlu pemindai khusus atau scanner untuk menterjemahkan kode tersebut sedangkan barcode dua dimensi hanya membutuhkan kamera baik kamera ponsel maupun wabcam untuk memecahkan kode.

Barcode dua dimensi telah memberikan kemudahan kepada para pengguna baik dengan mentranfer informasi dengan kecepatan tinggi maupun dengan banyaknya kapasitas data. Maka dari itu banyak perusahaan – perusahaan yang beralaih menggunakan barcode dua dimensi, ada sekitar 40 lebih barcode dua dimensi yang telah ada. Berikut adalah beberapa contoh barcode dua dimensi yang sering digunakan.*[8]*

- PDF417
- Data Matrix
- QR Code
- Dan lain-lain

2.2 QR Code

QR Code adalah suatu jenis kode matriks atau kode batang dua dimensi yang dikembangkan oleh Denso Wave, sebuah divisi Denso Corporation yang merupakan sebuah perusahaan Jepang dan dipublikasikan pada tahun 1994 dengan fungsionalitas utama yaitu dapat dengan mudah dibaca oleh pemindai. QR Code

merupakan singkatan dari *quick response* atau respons cepat, yang sesuai dengan tujuannya adalah untuk menyampaikan informasi dengan cepat dan mendapatkan respons yang cepat pula, berbeda dengan kode batang, yang hanya menyimpan informasi secara horizontal, QR Code mampu menyimpan informasi secara horizontal dan vertikal, oleh karena itu secara otomatis QR Code dapat menampung informasi yang lebih banyak daripada kode batang.[9]

Tabel 2.1. Kapasitas Data QR Code

Tipe Data	Maximun Karakter
Numerik	7.089
Alphanumerik	4.296
Biner	2.953
Kanji	1.817



Gambar 2.2. Contoh QR Code

Awalnya QR Code digunakan untuk pelacakan kendaraan bagian di manufaktur, namun kini QR Code digunakan dalam konteks yang lebih luas, termasuk aplikasi komersial dan kemudahan pelacakan aplikasi berorientasi yang ditujukan untuk pengguna telepon selular maupun PC. Di Jepang, penggunaan QR Code sangat populer, hampir semua jenis ponsel di Jepang bisa membaca QR Code sebab sebagian besar pengusaha disana telah memilih QR Code sebagai alat tambahan dalam program promosi produknya, baik yang bergerak dalam perdagangan maupun dalam bidang jasa. Pada umumnya QR Code digunakan untuk menanamkan informasi alamat situs suatu perusahaan. Di Indonesia, QR Code pertama kali diperkenalkan oleh KOMPAS. Dengan adanya QR Code pada koran harian di Indonesia ini, pembaca mampu mengakses berita melalui

ponselnya bahkan bisa memberi masukan atau opini ke reporter atau editor surat kabar tersebut.

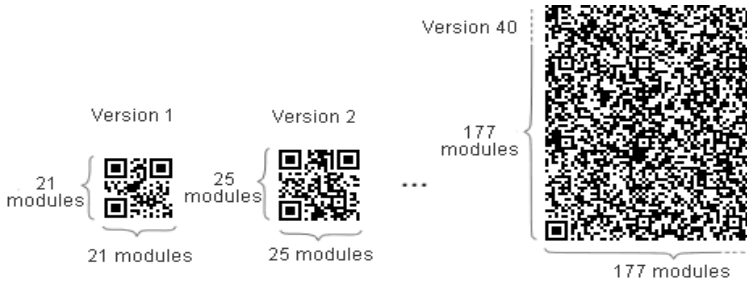
Ada beberapa dokumen standar yang meliputi pengkodean fisik dari QR Code, yaitu :

- Oktober 1997 : Disetujui sebagai AIM Internasional (*Automatic Identification Manufacturers International*) standar (ISS - QR Code)
- Maret 1998 : Disetujui sebagai JEIDA (*Japanese Electronic Industry Development Association*) standar (JEIDA - 55)
- Januari 1999 : Disetujui sebagai JIS JIS (*Japanese Industrial Standards*) standar (JIS X 0510)
- June 2000 : Disetujui sebagai standar international (ISO / IEC18004).
- November 2004 : Micro QR Code Disetujui sebagai JIS (Japanese Industrial Standards) standar (JIS X 0510)

QR Code memiliki kapasitas tinggi dalam data pengkodean, yaitu mampu menyimpan semua jenis data, seperti data numerik, data alphanumerik, kode biner dan kanji. Secara spesifik QR Code mampu menyimpan data jenis numerik sampai dengan 7.089 karakter, data alphanumerik sampai dengan 4.296 karakter, kode binari sampai dengan 2.844 byte, dan huruf kanji sampai dengan 1.817 karakter. Selain itu QR Code memiliki tampilan yang lebih kecil daripada kode batang. Hal ini dikarenakan QR Code mampu menampung data secara horizontal dan vertikal, oleh karena itu secara otomatis ukuran dari tampilannya gambar QR Code bisa hanya seperspuluh dari ukuran sebuah kode batang. Tidak hanya itu, QR Code juga tahan terhadap kerusakan, sebab QR Code mampu memperbaiki kesalahan sampai dengan 30% untuk *Level H*, lalu untuk *Level Q* sampai 25%, *Level M* sampai 15%, dan *Level L* sampai 7%. Oleh karena itu, walaupun sebagian simbol QR Code kotor ataupun rusak, data tetap dapat disimpan dan dibaca. Tiga tanda berbentuk persegi di tiga sudut memiliki fungsi agar simbol dapat dibaca dengan hasil yang sama dari sudut manapun sepanjang 360 derajat.

Pembuatan dan pembacaan QR Code selain dapat mengunakan perangkat lunak juga dapat ditelaah secara manual, pada bagian ini akan dipapar mengenain

pembacaan dan pembuat QR Code secara manual dengan tujuan agar dapat dimengerti dan dapar memahami teori QR Code tersebut, sebelum masuk ke proses decoder dan encoder akan lebih baik memahami versi dalam QR Code, yang dapat dilihat di gambar 2.3



Gambar 2.3. Versi QR Code

Versi dalam QR Code berkisar antara versi 1 (21 x 21 modul) sampai versi 40 (177 x 177 modul). Setiap versi dlam QR Code memiliki kapasitas data maksimum dengan jumlah data, jenis karakter, dan koreksi error masing-masing. Untuk lengkapnya dapat dilihat di tabel 2.2.

Tabel 2.2. Versi QR Code

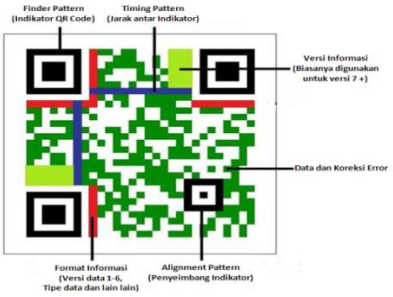
Versi	Modul	Lavel Koreksi Error	Data bit	Numerik	Alfanumerik	Biner	Kanji
1	21 X 21	L	125	41	25	17	10
		M	128	34	20	14	8
		Q	104	27	16	11	7
		H	72	17	10	7	4
2	25 X 25	L	272	77	47	32	20
		M	224	63	38	26	16
		Q	176	48	29	20	12
		H	128	34	20	14	8
:	:	:	:	:	:	:	:

Dari tabel kita dapat dilihat batas kapasitas data, dan dapat membandingkan nilai kapasitas tersebut dengan data yang akan dikodekan. Kebanyakan aplikasi

decoding dan *encoding* secara otomatis akan memodifikasi versi dengan melihat dalam batas jumlah data yang akan dikodekan.

2.2.1 Decoder

Decoder adalah proses pembacaaan QR Code, ada beberapa bagian yang dapat dilihat secara manual yang dapat dilihat di gambar 2.4



Gambar 2.4. Lokasi Area QR Code

- Finder pattern

  Finder pattern untuk membantu mendeteksi area simbol QR Code dalam aplikasi decoder. Simbol ini memiliki beberapa sudut menurun yang mampu rotasi 360 derajat melalui pola finder di sudut-sudut.

- Alignment pattern

  Alignment pattern sebagai pembantu atau penyeimbang pola finder. Finder pattern dan Alignment pattern adalah bagian paling penting untuk menemukan simbol. Satu-satunya perbedaan antara versi QR Code adalah bahwa Alignment pattern selalu ada di versi 1 tidak mengunakan alignment pattern, untuk versi 2 sampai versi 6 hanya satu alignment pattern, dan untuk versi 7 sampai versi 40 menggunakan lebih dari satu alignment pattern.

- Timing Pattern

  Pola ini digunakan untuk membantu menentukan koordinat modul simbol dalam aplikasi QR Code decoder.

- Quiet Zone

  Bagian ini adalah ruang kosong ditempatkan di sekitar simbol QR Code dan digunakan untuk meningkatkan pola finder untuk terdeteksi dengan cepat.
- Format Area

  Bagian ini menyimpan beberapa informasi tentang simbol QR Code, seperti versi, tipe data, dan sebagainya.
- Data Area

  Daerah ini adalah bagian paling penting dari QR Code karena bagian ini lokasi data yang dikodekan.

2.2.2 Encoder

Proses encoder adalah proses dimana input data text sampai menjadi QR Code dengan tahapan-tahapannya, sebelum masuk ke tahapan-tahapan *user* harus menentukan tipe data, versi data dan tingkat koreksi error yang akan dibuat menjadi QR Code.

2.2.2.1 Menentukan Kapasitas Data

Menentukan kapasitas data adalah proses awal sebelum proses pembuatan QR Code, untuk lebih memahaminya penulis memasukan contoh data yang di buat menjadi QR Code. **Contoh: ABCD123**

a. Tipe data.

   Tipe data dalam QR Code sebagai berikut:

   - Numerik
   - Alfanumerik
   - Biner
   - Kanji

b. Versi data.

   Versi data dalam QR Code telah dipaparkan dalam pembahasan dan tabel sebelumnya.

c. Tingkat koreksi error.

Kapasitas data dalam QR Code cukup banyak, tetapi terkadang hasil cetakan dari QR Code mengalami kerusakan yang di akibatkan atau kotor. Data di dalamnya dapat dipulihkan dalam kisaran tertentu dengan kemampuan koreksi kesalahan yang bahkan jika simbol sebagian hilang. Pada tabel 2.3 dibawah ini, terdapat empat tingkat koreksi kesalahan dan QR Code dapat memulihkan QR Code.

Tabel 2.3. Pemulihan Koreksi Error

Level L	7% dari kode yang hilang dapat dikembalikan
Level M	15% dari kode yang hilang dapat dikembalikan
Level Q	25% dari kode yang hilang dapat dikembalikan
Level H	30% dari kode yang hilang dapat dikembalikan

2.2.2.2 Pengkodean Data

Dalam proses pengkodean QR Code, Dari contoh di atas mendapatkan informasi data sebagai berikut:

Tabel 2.4. Contoh QR Code

Informasi Data	Isi
Data	ABCDE123
Tipe data	Alfanumerik
Version	1
Tingkat Kesalahan	H (30%)

a. Mode indicator.

Pertama mode indikator dibuat dengan menambahkan 4 bit bilangan biner dengan ketentuan sebagai berikut:

- Numerik : 0001
- *Alfanumerik: 0010*
- Biner : 0100
- Kanji : 1000

Dari contoh kasus di atas memperoleh **0010**

b. Menghitung karakter indicator.

Pertama harus mengihung jumlah karakter yang akan di kodekan lalu karakter indicator dengan ketentuan sebagai berikut

- Numerik : 10 bit
- *Alphanumerik* : *9 bit*
- Biner : 8 bit
- Kanji : 8 bit

0010 **000001000**

c. Mengkodekan data ke dalam biner

Dalam mengkodekan data teks ke biner, pertama data tersebut di bagi dalam kelompok-kelompok per-2 karakter, setiap karakter masukan dicocokan pada tabel 2.5, untuk mendapatkan reperesentasi biner dari teks masukan, nilai karekter pertama dikalikan dengan "45" dan di tambah dengan nilai karakter ke-2. Hasil perhitungan tersebut kemudian dikodekan ke dalam biner dengan 11 digit. Apabila jumlah karakter yang akan dikodekan berjumlah ganjil, maka pada kelompok terkhir hanya ada 1 karakter yang harus dikodekan dalam reperentasi ke biner.

Tabel 2.5. Konversi Alfanumerik ke Biner

Data	Biner
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
A	10

Data	Biner
B	11
C	12
D	13
E	14
F	15
G	16
H	17
I	18
J	19
K	20
L	21

Data	Biner
M	22
N	23
O	24
P	25
Q	26
R	27
S	28
T	29
U	30
V	31
W	32

Data	Biner
X	33
Y	34
Z	35
Spasi	36
$	37
%	38
*	39
+	40
-	41
.	42
/	43
:	44

“ AB ”	“ CD “	“ E1 “	“23”
(45x10)+11=461	(45 x12)+13 =553	(45x14)+1=631	(45x2)+3=93
00111001101	01000101001	01001110111	00001011101

0010 000001000 **00111001101 01000101001 01001110111 00001011101**

d. Terminator koreksi error

Langkah ini tidak diperlukan untuk koreksi error dengan menambahkan “0000”.

0010 000001000 00111001101 01000101001 01001110111 00001011101 **0000**

e. Mengkodekan biner ke dalam data

Pada bagian ini semua data yang telah di koreksi error di satukan, lalu di kelompok-kelompokan per 8 bit, apabila kurang maka di bagian belakang dapat di tambahkan “0” sampai kelompok terkhir berjumlah 8 bit

00100000 01000001 11001101 01000101 00101001 11011100 00101110 10000**000**

Jika jumlah karakter dari contoh kurang dari jumlah karakter dalam tabel 2.6, maka tambahkan di bagian belakang 11101100 dan 00010001 sampai jumlah karakter encoder penuh, dikarenakan jumlah karakter dari contoh adalah 8 dan jumlah karakter encoder 9, maka penambahan data hanya 11101100, lalu data tersebut konversi menjadi desimal.

Tabel 2.6. Kapasitas Data

Versi	**Level Koreksi Error**	**Jumlah karakter Encoder**	**Jumlah Koreksi Error**
1	L	19	7
	M	16	10
	Q	13	13
	H	9	17
:	:	:	:

00100000 01000001 11001101 01000101 00101001 11011100 00101110 10000000 **11101100**

32 65 205 69 41 220 46 128 236

2.2.2.3 Menghitung Koreksi Error

Dalam proses koreksi error menggunakan rumus reed-solomon. Pertama, hasil pengolagan data sebelumnya masukan ke tabel 2.7, dalam contoh yang di buat, blok RS adalah 1 dan Hitungan Kata Kode EC adalah 17 maka data tersebut akan masuk ke dalam rumus dalam tabel 2.7.

Tabel 2.7. Reed Solomon

Versi	Lavel Koreksi Error	Hitungan Kata Kode Data	Hitungan Kata Kode EC	Hitungan Blok RS (Reed-Solomon)
1	L	19	7	1
	M	16	10	1
	Q	13	13	1
	H	9	17	1
2	L	34	10	1
	M	28	16	1
	Q	22	22	1
	H	16	28	1
3	L	55	15	1
	M	44	26	1
	Q	34	36	2
	H	26	44	2
4	L	82	20	1
	M	64	36	2
	Q	48	52	2
	H	36	64	4
:	:	:	:	:

Tabel 2.8. Generator Koreksi Error

Hitungan data EC	Generator Polinomial g(h)
7	$x^7 + \alpha^{87} x^6 + \alpha^{229} x^5 + \alpha^{146} x^4 + \alpha^{149} x^3 + \alpha^{238} x^2 + \alpha^{102} x + \alpha^{21}$
10	$x^{10} + \alpha^{251} x^9 + \alpha^{67} x^8 + \alpha^{46} x^7 + \alpha^{61} x^6 + \alpha^{118} x^5 + \alpha^{70} x^4 + \alpha^{64}$ $x^3 + \alpha^{94} x^2 + \alpha^{32} x + \alpha^{45}$
13	$x^{13} + \alpha^{74} x^{12} + \alpha^{152} x^{11} + \alpha^{176} x^{10} + \alpha^{100} x^9 + \alpha^{86} x^8 + \alpha^{100}$ $x^7 + \alpha^{106} x^6 + \alpha^{104} x^5 + \alpha^{130} x^4 + \alpha^{218} x^3 + \alpha^{206} x^2 + \alpha^{140} x + \alpha^{78}$
16	$x^{16} + \alpha^{120} x^{15} + \alpha^{104} x^{14} + \alpha^{107} x^{13} + \alpha^{109} x^{12} + \alpha^{102} x^{11} + \alpha^{161}$ $x^{10} + \alpha^{76} x^9 + \alpha^3 x^8 + \alpha^{91} x^7 + \alpha^{191} x^6 + \alpha^{147} x^5 + \alpha^{169} x^4 + \alpha^{182}$ $x^3 + \alpha^{194} x^2 + \alpha^{225} x + \alpha^{120}$
17	$x^{17} + \alpha^{43} x^{16} + \alpha^{139} x^{15} + \alpha^{206} x^{14} + \alpha^{78} x^{13} + \alpha^{43} x^{12} + \alpha^{239}$ $x^{11} + \alpha^{123} x^{10} + \alpha^{206} x^9 + \alpha^{214} x^8 + \alpha^{147} x^7 + \alpha^{24} x^6 + \alpha^{99}$ $x^5 + \alpha^{150} x^4 + \alpha^{39} x^3 + \alpha^{243} x^2 + \alpha^{163} \alpha + x^{136}$
18	$x^{18} + \alpha\, x^{215} x^{17} + \alpha^{234} x^{16} + \alpha^{158} x^{15} + \alpha^{94} x^{14} + \alpha^{184} x^{13} + \alpha^{97}$ $x^{12} + \alpha^{118} x^{11} + \alpha^{170} x^{10} + \alpha^{79} x^9 + \alpha^{187} x^8 + \alpha^{152} x^7 + \alpha^{148}$ $x^6 + \alpha^{252} x^5 + \alpha^{179} x^4 + \alpha^5 x^3 + \alpha^{98} x^2 + \alpha^{96} x + \alpha^{153}$
22	$x^{22} + \alpha^{210} x^{21} + \alpha^{171} x^{20} + \alpha^{247} x^{19} + \alpha^{242} x^{18} + \alpha^{93} x^{17} + \alpha^{230}$ $x^{16} + \alpha^{14} x^{15} + \alpha^{109} x^{14} + \alpha^{221} x^{13} + \alpha^{53} x^{12} + \alpha^{200} x^{11} + \alpha^{74}$ $x^{10} + \alpha^8 x^9 + \alpha^{172} x^8 + \alpha^{98} x^7 + \alpha^{80} x^6 + \alpha^{219} x^5 + \alpha^{134} x^4 + \alpha^{160}$ $x^3 + \alpha^{105} x^2 + \alpha^{165} x + \alpha^{231}$
28	$x^{28} + \alpha^{168} x^{27} + \alpha^{223} x^{26} + \alpha^{200} x^{25} + \alpha^{104} x^{24} + \alpha^{224} x^{23} + \alpha^{234}$ $x^{22} + \alpha^{108} x^{21} + \alpha^{180} x^{20} + \alpha^{110} x^{19} + \alpha^{190} x^{18} + \alpha^{195} x^{17} + \alpha^{147}$ $x^{16} + \alpha^{205} x^{15} + \alpha^{27} x^{14} + \alpha^{232} x^{13} + \alpha^{201} x^{12} + \alpha^{21} x^{11} + \alpha^{43}$ $x^{10} + \alpha^{245} x^9 + \alpha^{87} x^8 + \alpha^{42} x^7 + \alpha^{195} x^6 + \alpha^{212} x^5 + \alpha^{119}$ $x^4 + \alpha^{242} x^3 + \alpha^{37} x^2 + \alpha^9 x + \alpha^{123}$
:	:

g (x) = $x^{17} + \alpha^{43} x^{16} + \alpha^{139} x^{15} + \alpha^{206} x^{14} + \alpha^{78} x^{13} + \alpha^{43} x^{12} + \alpha^{239} x^{11} + \alpha^{123}$
$x^{10} + \alpha^{206} x^9 + \alpha^{214} x^8 + \alpha^{147} x^7 + \alpha^{24} x^6 + \alpha^{99} x^5 + \alpha^{150} x^4 + \alpha^{39} x^3 + \alpha^{243}$
$x^2 + \alpha^{163} \alpha + x^{136}$

Dalam menjumlahkan dua polinomial, terdapat pada polinomial g(x) dengan tabel 2.8 yang akan mendapatkan konversi α dalam berikutnya, selain itu polinomial memiliki beberapa sifat dalam transformasi sebagai berikut.

- Empat operasi aritmatika yang didukung.
- $\alpha^{255} = 1$.
- Koefisien α dapat disederhanakan dengan tabel 2.9.

Ketiga sifat diatas digunakan sebagai aturan perhitungan data koreksi kesalahan yang akan digunakan terus menerus dalam operasi penjumlahan logika.

$f(x) = 32x^{25} + 65x^{24} + 205x^{23} + 69x^{22} + 41x^{21} + 220x^{20} + 46x^{19} + 128x^{18} + 236x^{17}$ ..... (4.1)

f(x) di bagi dengan g(x)

Koefisien f (x) adalah 32, untuk 32 adalah α^5 dari tabel 2.9.

Tabel 2.9 GF

Exponent of α	**Integer**	**Integer**	**Exponent of α**
:	:	:	:
31	192	31	113
32	157	32	5
33	39	33	138
:	:	:	:

$g(x) * (\alpha^5 * x^8) = (\alpha^5 * x^8 * x^{17}) + (\alpha^5 * \alpha^{43} * x^8 * x^{16}) + (\alpha^5 * \alpha^{139} * x^8 * x^{15}) +$

$(\alpha^5 * \alpha^{*\,206} * x^8 * x^{14}) + (\alpha^5 * \alpha^{78} * x^8 * x^{13})...$

$= 32x^{25} + 70x^{24} + 168x^{23} + 178x^{22} + 187x^{21}$..... (4.2)

Untuk menghitung f '(x) adalah f(x) XOR g(x), dengan demikaian di peroleh hasil sebagai berikut,

f '(x) = (4.1) XOR (4.2)

$f'(x) = 0 + 7x^{24} + 101x^{23} + 247x^{22} + 146x^{21} + \ldots$

Ulangi langkah yang sama hingga perhitungan ini membagi selesai. Jika eksponen dari α yang lebih dari 255, maka kita turun dengan menggunakan $\alpha = 1^{255}$

Akhirnya kita bisa mendapatkan di bawah sisa R (x).

Ulangi langkah logika yang sama sampai nilai f (x) terbaru dan tertinggi harus lebih kecil dari eksponen g (x). Jika eksponen dari α yang lebih dari 255, maka kita turun dengan menggunakan $\alpha = 1^{255}$. Dalam contoh setelah jumlah logika berulang, kita dapat memperoleh polinomial sisa terakhir sebagai berikut.

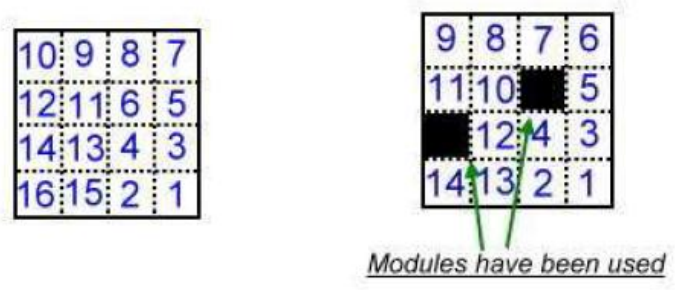
$$R(x) = 42x^{16} + 159x^{15} + 74x^{14} + 221x^{13} + 244x^{12} + 169x^{11} + 239x^{10} + 150x^{9} + 138x^{8} + 70x^{7} + 237x^{6} + 85x^{5} + 224x^{4} + 96x^{3} + 74x^{2} + 219x + 61$$

Koreksi error dalam contoh ini adalah "42 159 74 221 244 169 239 150 138 70 237 85 224 96 74 219 61", hasil ini dapat di tambahkan ke data encoding sebelumnya.

32 65 205 69 41 220 46 128 236 **42 159 74 221 244 169 239 150 138 70 237 85 224 96 74 219 61**

2.2.2.4 Alokasi Data

Pengalokasian data adalah tahapan untuk menetapkan data dari bagian sebelumnya ke dalam kerangka dengan aturan alokasi data, setiap modul dalam simbol QR Code memiliki koordinat yang memiliki baris dan kolom, misalnya versi 1 adalah 21 x 21 modul, dikarenakan hasil dari perhitungan Blok RS yang berada pada tabel RS (*Reed-solomon*) versi 1 menjadi 20 x 20 modul dan modul yang berasal dari (0, 0) ke (20, 20), untuk lebih jelas dapat dilihat pada gambar 2.5



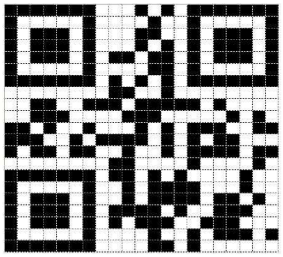
(a) (b)

Gambar 2.5. QR Code Arah Alokasi.

Ilustrasi gambar (a) angka 1 menjukan langkah pertama dalam penempatan, dan angka 2 menjukan langkah berikutnya secara terus menerus sampai pada angka terkhir, sedangkan untuk ilustrasi gambar (b) langkah-langkahnya sama

seperti di ilustrasi gambar (a) tetapi apabila ada beberapa modul tersebut telah terisi data (format informasi) maka penampatan data dilanjutkan ke modul selanjutnya gambar (b) pengalokasian data pada contoh data sebelumnya.

Dari langkah Menentukan kapasitas data, Pengkodean data, Menghitung koreksi error dan Alokasi data dengan aturan-aturan yang telah dijelaskan di atas, maka dapat menggambarkan secara manual QR Code tanpa pola *mask* sepeti pada gambar 2.6.



Gambar 2.6. QR Code Tanpa Pola Mask

2.2.2.5 Pola Mask

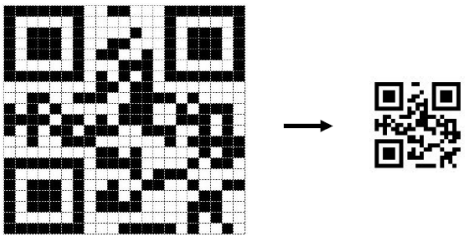
Pola mask berfungsi untuk menghilangkan bagian yang bukan merupakan modul data yang diinginkan, terdapat delapan jenis indikator pola mask dalam QR Code, yang lihat pada tabel 2.10.

Tabel 2.10. Indikator Pola Mask

Pola *Mask*	**Aturan**
000	Jika (x+y) mod = 0, maka isi matrik mask pada posis tersebut dengan 1 (true)
001	Jika y mod 2 = 0, maka isi matrik mask pada posisi tersebut dengan 1 (true)
010	Jika x mod 3 = 0, maka isi matrik mask pada posisi tersebut dengan 1 (true)
011	Jika (x+y) mod 3 = 0, maka isi matrik mask pada posisi tersebut dengan 1 (true)

100	Jika ((x/3)+(y/2 mod 2 = 0, maka isi matrik mask pada posisi tersebut dengan 1 (true)
101	Jika ((x*y) mod 2 + (x*y) mod 3=0), maka isi matrik mask pada posisi tersebut dengan 1 (true)
110	Jika (((x*y) mod 3 + (x+y) mod 2) mod 2 = 0), maka isi matrik mask pada posisi tersebut dengan 1 (true)
111	Jika (((x*y) mod 3 + (x+y) mod 2) mod 2 = 0), maka isi matrik mask pada posisi tersebut dengan 1 (true)

QR Code yang di contoh dengan masukan data "ABCD123" dan menggunakan pola mask "011" , telah dapat di cetah seperti pada gambar 2.7.



Gambar 2.7. QR Code.

Meskipun membuat QR Code secara manual, pengguna mungkin merasa bingung pada pola mask, dengan delapan jenis mask pada QR Code. Sebenarnya tidak ada khusus untuk menentukan pola mask. Setiap aplikasi encoder memiliki pola mask tetap atau dengan acak, hal ini tergantung pada pengembangan coding perangkat lunak yang di buat.

2.3 Webcam

Webcam (singkatan dari *web camera*) adalah sebutan bagi kamera *real-time* (bermakna keadaan pada saat ini juga) yang gambarnya bisa diakses atau dilihat melalui *World Wide Web*, program *instant messaging*, atau aplikasi *video call*. Istilah webcam merujuk pada teknologi secara umumnya, sehingga kata web kadang-kadang diganti dengan kata lain yang mendeskripsikan pemandangan yang ditampilkan di kamera, misalnya StreetCam yang memperlihatkan

pemandangan jalan. Ada juga Metrocam yang memperlihatkan pemandangan panorama kota dan pedesaan, TraffiCam yang digunakan untuk memonitor keadaan jalan raya, cuaca dengan Weather Cam, bahkan keadaan gunung berapi dengan VolcanoCam. Webcam atau web camera adalah sebuah kamera video digital kecil yang dihubungkan ke komputer melalui (biasanya) port USB ataupun port COM.[7]



Gambar 2.8. Webcam

2.3.1 Cara Kerja Webcam

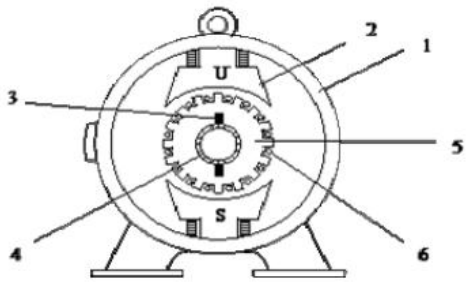
Sebuah web camera yang sederhana terdiri dari sebuah lensa standar, dipasang di sebuah papan sirkuit untuk menangkap sinyal gambar, casing (cover), termasuk casing depan dan casing samping untuk menutupi lensa standar dan memiliki sebuah lubang lensa di casing depan yang berguna untuk memasukkan gambar, kabel *support*, yang dibuat dari bahan yang *fleksibel*, salah satu ujungnya dihubungkan dengan papan sirkuit dan ujung satu lagi memiliki connector, kabel ini dikontrol untuk menyesuaikan ketinggian, arah dan sudut pandang web camera. Sebuah web camera biasanya dilengkapi dengan software, software ini mengambil gambar-gambar dari kamera digital secara terus menerus ataupun dalam interval waktu tertentu dan menyiarkannya melalui koneksi internet. Ada beberapa metode penyiaran, metode yang paling umum adalah software mengubah gambar ke dalam bentuk file JPEG dan menguploadnya ke web server menggunakan *File Transfer Protocol* (FTP).

Frame rate mengindikasikan jumlah gambar sebuah software dapat ambil dan transfer dalam satu detik. Untuk streaming video, dibutuhkan minimal 15 *frame per second* (fps) atau idealnya 30 fps. Untuk mendapatkan *frame rate* yang tinggi, dibutuhkan koneksi internet yang tinggi kecepatannya. Sebuah web camera

tidak harus selalu terhubung dengan komputer, ada web camera yang memiliki software webcam dan web *server bulit-in*, sehingga yang diperlukan hanyalah koneksi internet. Web camera seperti ini dinamakan "*network camera*". Kita juga bisa menghindari penggunaan kabel dengan menggunakan hubungan radio, koneksi Ethernet ataupun WiFi.

2.4 Motor DC

Motor arus searah (motor DC) adalah salah satu jenis motor yang telah ada selama lebih dari seabad. Keberadaan motor DC telah membawa perubahan besar sejak dikenalkan motor induksi yang nama lain dari motor listrik arus bolak balik (AC) karena motor DC mempunyai keunggulan dalam kemudahan untuk mengatur dan mengontrol kecepatan dibandingkan motor AC (motor bolak-balik yang bekerja memerlukan suplay tegangan bolak balik). Motor DC dapat berfungsi sebagai motor apabila didalam motor listrik tersebut terjadi proses konversi dari energi listrik menjadi energi mekanik. Motor DC itu sendiri memerlukan suplai tegangan yang searah pada kumparan jangkar dan kumparan medan untuk diubah menjadi energi mekanik. Pada motor DC kumparan medan disebut stator (bagian yang tidak berputar) dan kumparan jangkar disebut rotor (bagian yang berputar).[5]



Gambar 2.9. Bagian – bagian Motor DC

Bagian – bagian motor DC secara umum, yaitu :

1. Badan Mesin

   Badan mesin ini berfungsi sebagai tempat mengalirnya fluks magnet yang dihasilkan kutub magnet, sehingga harus terbuat dari bahan ferromagnetik. Fungsi lainnnya adalah untuk meletakkan alat-alat tertentu dan

mengelilingi bagian-bagian dari mesin, sehingga harus terbuat dari bahan yang benar-benar kuat, seperti dari besi tuang dan plat campuran baja.

2. Inti kutub magnet dan belitan penguat magnet

   Inti kutub magnet dan belitan penguat magnet ini berfungsi untuk mengalirkan arus listrik agar dapat terjadi proses elektromagnetik. Adapun aliran fluks magnet dari kutub utara melalui celah udara yang melewati badan mesin.

3. Sikat-sikat

   Sikat-sikat ini berfungsi sebagai jembatan bagi aliran arus jangkar dengan bebas, dan juga memegang peranan penting untuk terjadinya proses komutasi.

4. Komutator

   Komutator ini berfungsi sebagai penyearah mekanik yang akan dipakai bersama-sama dengan sikat. Sikat-sikat ditempatkan sedemikian rupa sehingga komutasi terjadi pada saat sisi kumparan berbeda.

5. Jangkar

   Jangkar dibuat dari bahan ferromagnetik dengan maksud agar kumparan jangkar terletak dalam daerah yang induksi magnetiknya besar, agar ggl induksi yang dihasilkan dapat bertambah besar.

6. Belitan jangkar :

   Belitan jangkar merupakan bagian yang terpenting pada mesin arus searah, berfungsi untuk tempat timbulnya tenaga putar motor.

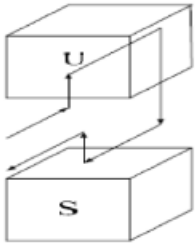
2.4.1 Jenis – jenis Motor DC

Secara umum motor DC dibagi atas 2 (dua) macam, yaitu ;

1. Motor DC dengan sikat yang berfungsi sebagai pengubah arus pada kumparan sedemikian rupa sehingga arah putaran motor akan selalu sama.
2. Motor DC tanpa sikat menggunakan semi konduktor untuk merubah maupun membalik putarannya untuk menggerakkan motor, tingkat kebisingan motor jenis ini rendah karena putarannya halus.

2.4.2 Prinsip Kerja Motor DC

Daerah kumparan medan yang yang dialiri arus listrik akan menghasilkan medan magnet yang melingkupi kumparan jangkar dengan arah tertentu. Konversi dari energi listrik menjadi energi mekanik (motor) maupun sebaliknya berlangsung melalui medan magnet, dengan demikian medan magnet disini selain berfungsi sebagai tempat untuk menyimpan energi, sekaligus berfungsi sebagai tempat berlangsungnya proses perubahan energi dan daerah tersebut dapat dilihat pada gambar dibawah ini :



Gambar 2.10. Prinsip Kerja Motor DC

Dengan mengacu pada hukum kekekalan energi :

Proses energi listrik = energi mekanik + energi panas + energi didalam medan magnet. Maka dalam medan magnet akan dihasilkan kumparan medan dengan kerapatan fluks sebesar B dengan arus adalah I serta panjang konduktor sama dengan L maka diperoleh gaya sebesar F, dengan persamaan sebagai berikut:

F = B I L (2.1)

dengan :

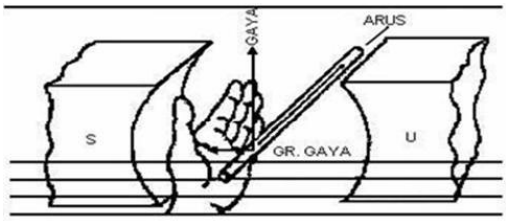
F = Gaya magnet pada sebuah arus (Newton)

B = Medan magnet (Tesla)

I = Arus yang mengalir (Ampere)

L = Panjang konduktor (meter)

Arah dari gaya ini ditentukan oleh aturan kaidah tangan kiri, adapun kaidah tangan kiri tersebut adalah sebagai berikut :



Gambar 2.11. Kaidah Tangan Kiri

Ibu jari sebagai arah gaya (F), telunjuk jari sebagai fluks (B), dan jari tengah sebagai arus (I). Saat gaya (F) tersebut dibandingkan, konduktor akan bergerak didalam kumparan medan magnet dan menimbulkan gaya gerak listrik yang merupakan reaksi lawan terhadap tegangan sumber. Agar proses perubahan energi mekanik tersebut dapat berlangsung secara sempurna, maka tegangan sumber harus lebih besar dari pada tegangan gerak yang disebabkan reaksi lawan. Dengan memberi arus pada kumparan jangkar yang dilindungi oleh medan maka menimbulkan perputaran pada motor.

2.5 Infrared Transmitter

Infrared transmitter merupakan suatu modul pengirim data melalui gelombang infra merah dengan frekuensi sebesar 38 KHz. Modul ini dapat difungsikan sebagai output dalam aplikasi transmisi data nirkabel seperti robotik, sistem pengaman, data logger, absensi, dan sebagainya. Cahaya inframerah (IR) adalah radiasi elektromagnetik dengan panjang gelombang antara 0,7 mikrometer dan 300 mikrometer, yang setara dengan rentang frekuensi antara 1 dan sekitar 430 THz.

IR panjang gelombang lebih panjang dari pada cahaya tampak, tetapi lebih pendek dari radiasi gelombang mikro Terahertz. sinar matahari Bright memberikan radiasi hanya lebih dari 1 kilowatt per meter persegi di permukaan laut. Energi ini, 527 watt adalah radiasi infra merah, 445 watt adalah cahaya, dan 32 watt adalah radiasi *ultraviolet*. Inframerah ditemukan secara tidak sengaja oleh Sir William Herschell, astronom kerajaan Inggris ketika ia sedang mengadakan penelitian mencari bahan penyaring optik yang akan digunakan untuk mengurangi kecerahan gambar matahari dalam tata surya teleskop

Spesifikasi hardware infra red transmitter :

1. Tegangan kerja +5 VDC.
2. Frekuensi carrier penerima infra merah 38 kHz.
3. Panjang gelombang puncak 940 nm.
4. Sudut pancaran ±17º.
5. Jarak maksimum yang teruji pada sudut 0^o: 16 m. Jarak maksimum sesuai datasheet: 35 m
6. Memiliki input yang kompatibel dengan level tegangan TTL,CMOS, dan RS-232.
7. Terdapat 2 mode output: non-inverting dan inverting.
   Kompatibel penuh dengan DT-51™ Minimum System (MinSys)ver 3.0, DT-51™ PetraFuz, DT-BASIC Series, DT-51™ Low CostSeries, DT-AVR Low Cost Series, dan lain-lain [4]

2.6 Phototransistor

Phototansistor merupakan suatu jenis transistor yang sangat peka trahadap cahaya yang ada disekitarnya. Pada gambar 2.12. Ketika basis menangkap cahaya maka *collector*akan terhubung dengan *emitter* dalam hal ini transistor bekerja. Prinsip kerja *phototransistor* sama seperti transistor pada umumnya dengan kata lain *phototransistor* akan bekerja seperti saklar dengan parameter cahaya untuk mendapatkan kondisi *on* dan *off*. Berikut ini adalah simbol *phototransistor*.[6]



Gambar 2.12. Bentuk Fisik dan Simbol Phototransistor

2.7 IC LM 393

IC Komparator atau IC pembanding adalah sebuah IC yang berfungsi untuk membandingkan dua macam tegangan yang terdapat pada kedua inputnya.

Komparator memiliki 2 buah input dan sebuah output. Inputnya yaitu input(+) dan input (-). Dapat di lihat pada gambar 2.13 (b) [3]

LM 393 dalam satu kemasannya mempunyai dua buah komparator didalamnya. IC ini memiliki fitur sebagai berikut:

IC komparator LM 393 memiliki fitur-fitur sebagai berikut:

a. Dapat bekerja dengan single supply 2V sampai 36V
b. Dapat bekerja dengan tegangan input -3V sampai +36V
c. Dapat bekerja dengan segala macam bentuk gelombang *logic*
d. Dapat membandingkan tegangan yang mendekati ground.

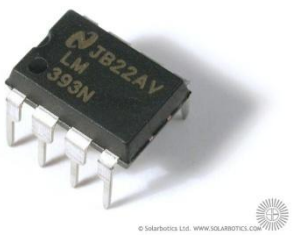
Dalam aplikasinya output dari komparator LM 393, membutuhkan resistor pullup dengan tegangan V+ yaitu untuk menjaga tegangan output supaya memiliki logika satu ketika kondisi diam.

Cara kerja komparator :

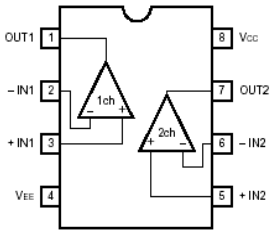
Komparator bekerja berdasarkan tegangan yang masuk pada kedua pin inputnya

1. Jika tegangan pada pin(+) lebih besar pada tegangan pin(-) maka output komparator akan bergerak kearah V+
2. Jika tegangan pada pin(+) lebih kecil pada tegangan pin(-) maka output komparator akan bergerak kearah V-

Dalam aplikasinya biasanya salah satu pin input dari komparator sebagai tegangan reverensi sedangkan pin input lainya sebagai tegangan yang akan dibandingkan.



(a)



(b)

Gambar 2.13. (a) Kemasan IC LM393N (b) Diagram rangkaian IC LM393N

2.8 Mikrokontroler AVR ATMega16

AVR merupakan seri mikrokontroler *Complementary Metal Oxide Semiconductor* (CMOS) 8-*bit* buatan Atmel berbasis arsitektur RISC *(Reduced Instruction Set Computer*). Hampir semua instruksi pada program dieksekusi dalam satu siklus *clock*. AVR mempunyai 32 register *general-purpose*, *timer/counter* fleksibel dengan mode *compare*, interupsi *internal* dan *eksternal*, serial UART, *programmable Watchdog Timer*, *power saving mode*, ADC dan PWM. AVR pun mempunyai In-*System Programmable* (ISP) *Flash on-chip* yang mengijinkan memori program untuk diprogram ulang (*read/write*) dengan koneksi secara serial yang disebut *Serial Peripheral Inteface* (SPI). [2]

AVR memilki keunggulan dibandingkan dengan mikrokontroler lain, keunggulan mikrokontroler AVR yaitu memiliki kecepatan dalam mengeksekusi program yang lebih cepat, karena sebagian besar instruksi dieksekusi dalam 1 siklus *clock* (lebih cepat dibandingkan mikrokontroler keluarga MCS 51 yang memiliki arsitektur *Complex Intrukstion Set Compute*).

ATMega16 mempunyai *throughput* mendekati 1 *Millions Instruction Per Second* (MIPS) per MHz, sehingga membuat konsumsi daya menjadi rendah terhadap kecepatan proses eksekusi perintah.

Beberapa keistimewaan dari AVR ATMega16 antara lain:

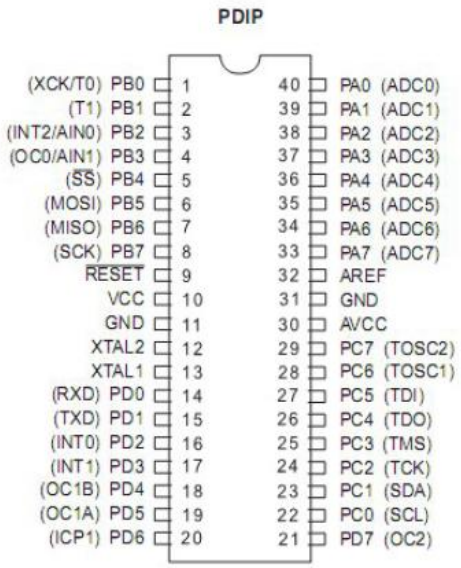
1. Mikrokontroler AVR 8 *bit* yang memilliki kemampuan tinggi dengan konsumsi daya rendah
2. Arsitektur RISC dengan *throughput* mencapai 16 MIPS pada frekuensi 16MHz
3. Memiliki kapasitas *Flash* memori 16 Kbyte, EEPROM 512 Byte dan SRAM 1 Kbyte
4. Saluran I/O sebanyak 32 buah, yaitu *Port* A, *Port* B, *Port* C dan *Port* D
5. CPU yang terdiri dari 32 buah *register*
6. nit interupsi dan eksternal
7. *Port* USART untuk komunikasi serial
8. Fitur *peripheral*
    - Tiga buah *Timer/Counter* dengan kemampuan perbandingan (*compare*)

- ✓ Dua buah *Timer/Counter* 8 *bit* dengan *Prescaler* terpisah dan *Mode Compare*
- ✓ Satu buah *Timer/Counter* 16 *bit* dengan *Prescaler* terpisah, *Mode Compare* dan *Mode Capture*

- *Real Time Counter* dengan *Oscillator* tersendiri
- *Real Time Counter* dengan *Oscillator* tersendiri
- Empat kanal PWM
- 8 kanal ADC
  - ✓ 8 *Single-ended Channel* dengan keluaran hasil konversi 8 dan 10 resolusi (register ADCH dan ADCL)
  - ✓ 7 *Diferrential Channel* hanya pada kemasan *Thin Quad Flat Pack* (TQFP)
  - ✓ 2 *Differential Channel* dengan *Programmable Gain*
- Antarmuka *Serial Peripheral Interface* (SPI) *Bus*
- *Watchdog Timer* dengan *Oscillator Internal*
- *On-chip Analog Comparator*

9. *Non-volatile program memory*

2.8.1 Konfigurasi Pin



Gambar 2.14. Konfigurasi Kaki (*pin*) ATMega16

Konfigurasi *pin* ATMega16 dengan kemasan 40 *pin Dual In-line Package* (DIP) dapat dilihat pada Gambar 2.14. dari gambar diatas dapat dijelaskan fungsi dari masing-masing *pin* ATMega16 sebagai berikut.

1. VCC merupakan *pin* yang brfungsi sebagai masukan catu daya
2. GND merupakan *pin Ground*
3. *Port* A (PA0 – PA7) merupakan *pin input/output* dua arah (*full duplex*) dan selain itu merupakan *pin* masukan ADC
4. *Port* B (PB0 – PB7) merupakan *pin input/output* dua arah (*full duplex*) dan selain itu merupakan *pin* khusus, seperti dapat dilihat pada tabel dibawah ini.

Tabel 2.11. Fungsi Khusus Port B

PIN	Fungsi Khusus
PB0	XCK (USART *External Clock Input/Output*) T0 (*Timer/Counter0 External Counter Input*)
PB1	T1 (*Timer/Counter1 External Counter Input*)
PB2	OC0 (*Timer/Counter0 Output Compare Macth Output*) AIN1 (*Analaog Comparator Negative Input*)
PB3	OC0 (*Timer/Counter0 Output Compare Macth Output*) AIN1 (*Analaog Comparator Negative Input*)
PB4	(SPI *Slave Select Input*)
PB5	MISO (SPI *Bus Master Input/Slave Output*)
PB6	MISO (SPI *Bus Master Input/Slave Output*)
PB7	SCK (SPI Bus Serial Clock)

5. *Port* A (PC0 – PC7) merupakan *pin input/output* dua arah (*full duplex*) dan selain itu merupakan *pin* khusus, seperti dapat dilihat pada tabel dibawah ini.

Tabel 2.12. Fungsi Khusus Port C

PIN	Fungsi Khusus
PC0	SCL (*Two-wire Serial Bus Clock Line*)
PC1	SDA (*Two-wire Serial BusData Input/Output Line*)
PC2	TCK (*Joint Test Action Group Test Clock*)

PC3	TMS (*JTAG Test Mode Select*)
PC4	TDO (*JTAG Data Out*)
PC5	TDI (*JTAG Test Data In*)
PC6	TOSC1 (*Timer Oscillator pin 1*)
PC7	TOSC2 (*Timer Oscillator pin 2*)

6. *Port* D (PD0 – PD7) merupakan *pin input/output* dua arah (*full duplex*) dan selain itu merupakan *pin* khusus, seperti dapat dilihat pada tabel dibawah ini.

Tabel 2.13. Fungsi Khusus Port D

Pin	Fungsi Khusus
PD0	RXD (USART *Input Pin*)
PD1	TXD (USART *Output Pin*)
PD2	INT0 (*External Interupt 0 Input*)
PD3	INT1 (*External Interupt 1 Input*)
PD4	OC1B (*Timer/Counter1 Output Compare B Macth Output*)
PD5	OC1A (*Timer/Counter1 Output Compare A Macth Outpu*
PD6	ICP (*Timer/Counter1 Input Capture Pin*)
PD7	OC2 (*Timer/Counter2 Output Compare Macth Output*)

7. RESET merupakan *pin* yang digunakan untuk me-*reset* mikrokontroler
8. XTAL1 dan XTAL2, merupakan *pin masukan external clock*
9. AVCC merupakan *pin* masukan tegangan untuk ADC
10. AREF merupakan *pin* masukan tegangan referensi untuk ADC.

2.8.2 Timer/Counter

Mikrokontroler AVR ATMega16 memiliki tiga buah *Timer/Counter*, yaitu: *Timer* 0 (8 *bit*), *Timer* 1 (16 *bit*) dan *Timer* 2 (8 *bit*). *Timer/Counter* 1 mempunyai keunggulan dibanding *Timer/Counter* 0 atau 2, namun cara mengatur *Timer* 0, 1, 2 sama saja, yaitu pada masing-masing registernya. *Timer/Counter 1* dapat menghitung sampai dengan 65536 *Timer/Counter* 0 atau 2 hanya sampai dengan 256. Selain itu, *Timer* 0 ini memiliki mode operasi sebanyak 8 mode.

2.8.3 Interupsi *eksternal*

Interupsi *eksternal* merupakan fitur tambahan dari mikrokontroler AVR ATMega16 yang khusus difungsikan untuk interupsi. Interupsi *eksternal* adalah jenis interupsi asinkron yang pengaktifannya bukan dipicu dari fitur: timer/counter, ADC, komparator analog ataupun dari komunikasi antarmuka, tetapi dipicu secara logika dari luar mikrokontroler (eksternal). Mikrokontroler AVR ATMega16 mempunyai 3 buah pemicu interupsi eksternal, yaitu pada *pin* INT0 (PORTD2), *pin* INT1 (PORTD3), *pin* INT2 (PORTB2).

2.8.4 Prescaler

Pada dasarnya *Timer* hanya menghitung pulsa *clock*. Frekuensi pulsa *clock* yang dihitung tersebut sama dengan frekuensi Kristal yang diginakan atau dapat diperlambat menggunakan *prescaler* dengan faktor 1, 8, 64, 256 atau 1024.

2.9 BASCOM AVR

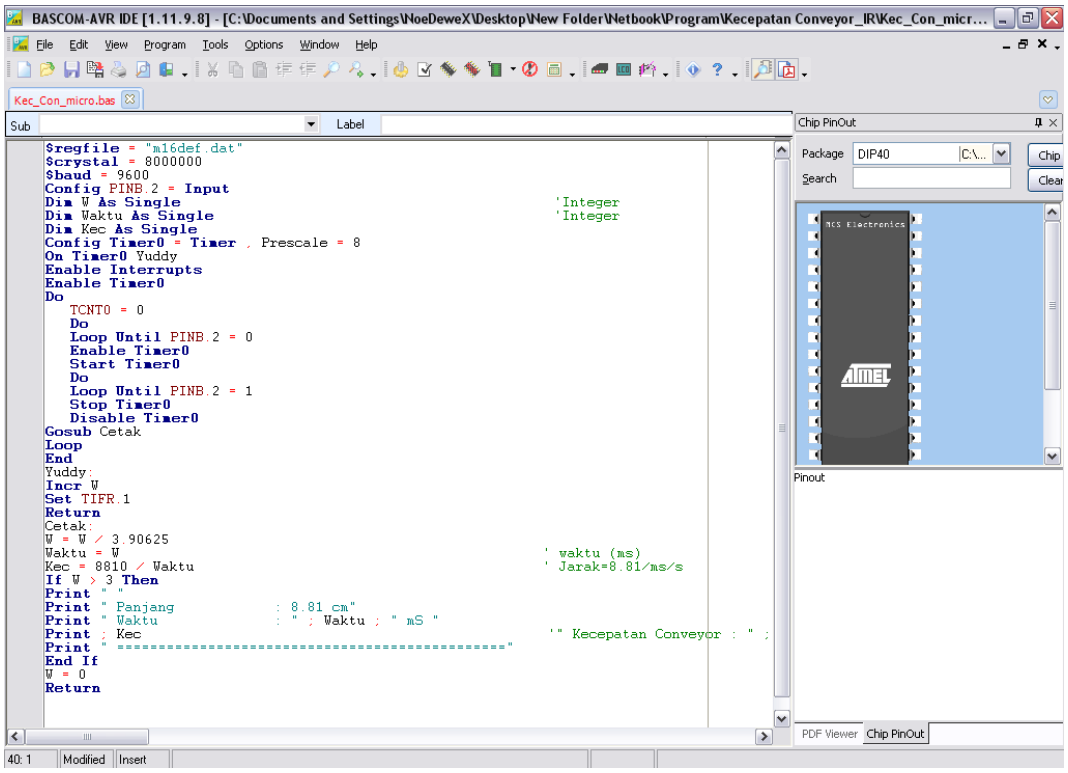
BASCOM AVR merupan singkatan dari Basic Compiler AVR. BASCOM AVR termasuk bahasa pemograman *microcontroller* buat MSC *Electronics* yang mengadaptasi bahasa tingkat tinggi yang sering digunakan oleh awam (Bahasa Basic). Dengan menggunakan bahasa tingkat tinggi, maka pemograman mendapatkan banyak kemudahan dalam mengatur sistem kerja dari *microcontroller* yang digunakan masih baru dan tidak terlalu dikenal.

BASCOM AVR memiliki program sendiri untuk memasukkan program yang telah dikompilasi ke dalam AVR. Untuk ATMega16 yang menggunakan ISP, konfigurasi dari *programmer* harus diubah menjadi STK300/STK200. *Connection* yang digunakan pada PC untuk men-download program ke IC *micro* ATMega16 haruslah menggunakan LPT port atau LPT1

Keuntungan yang dimiliki oleh BASCOM AVR adalah sebagai berikut:

- Bahasa Basic yang tersetruktur dan mempunyai label
- Mempunyai kecepatan dalam pengkodean data yang tinggi
- *Terminal emulator* dengan pemilihan *download* yang telah terintegrasi
- Terdapat *library*, *local variable* yang akan memudahkan *user*

- Terdapat *simulasi* untuk testing program



Gambar 2.15. Tampilan Utama Software BASCOM AVR

2.10 QuickMark

QuickMark adalah software barcode untuk membuat dan scanner barcode, QuickMark membuat aplikasi ini yang dapat digunakan pada mobile (telpon gengam) dan PC (*personal computer)*, yang dapat membuat dan menye-scan barcode batang (1D) dan barcode 2D.

Software QuickMark diluncurkan pada 2005, sekarang dapat software QuickMark dapat digunakan di berbagai sistem operasi di mobile (Java, Symbian, iPhone, Andorid, Windows Phone dan Mac OS X) maupun di PC (Mac dan Windows). Sekarang software QuickMark telah mencapai versi V3.8.5 untuk Windows. Pengembangan software QuickMark telah banyak baik untuk di sistem operasi mobile maupun sistem operasi PC.

2.10.1. Barcode SDK

Barcode SDK adalah fasilitas yang dapat dikembangkan pengembang dengan fleksibilitas dalam pengembangan aplikasi pada mobile dan PC, dengan bertujuan untuk member kenyamanan bagi penggunanya.

Pengembang dapat mendownload Barcode SDK dengan gratis dengan waktu atau penggunaan terbatas yaitu 99 scan atau 2 minggu pengunaan. Untuk melanjutkan pengmbangan atau pengunaan fasilitas ini user harus melakukan berbayar (*Lisensi distribusi*) agar pengguna dapat menggunakan dan mempublikasikan aplikasi. Berikut adakah beberapa contoh pengembangan Barcode SDK, Q+ adalah kamera eksternal untuk meningkatkan lensa kamera iPhone agar dapat membaca barcode yang berkururan kecil, Designer QR Code Sticker adalah pembuat barcode dengan desain yang menarik dan unik, QuickMark for PC adalah pembuat dan scanner barcode untuk PC, Send2 adalah aplikasi mobile yang dapat digunakan sebagai mengirim data barcode dengan cepat melalui berbagai metode dan lain-lain

2.10.2. Barcode API

Barcode API adalah *opensource* pembuatan *decoder* atau scanner QRCode dari QuickMark baik scanner QRCode untuk mobile atau scanner QRCode untuk PC dan file-file pendukung untuk pengembangan untuk teknologi di masa depan QRCode salah satunya adalah file msgapi.exe yang yang dapat membuat *output* hasil pembacaan data QRCode menjadi file txt.



Gambar 2.16. Scaner QuickMark [11]