# BAB II

# LANDASAN TEORI

## 2.1. Pengertian Sistem Informasi

### 2.1.1. Konsep Dasar Sistem

Terdapat dua kelompok pendekatan dalam mendefinisikan suatu sistem, yaitu:

a) Pendekatan sistem yang menekankan pada prosedur :

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu.

b) Pendekatan sistemyang menekankan pada elemen atau komponennya :

Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu.

Sistem memiliki karakteristik, yaitu :

a) Mempunyai komponen
b) Batas sistem
c) Penghubung
d) Masukan
e) keluaran
f) Pengolah / proses
g) Lingkungan luar system, dan
h) Sasaran atau tujuan.

Sistem juga diklasifikasikan menjadi :

a) Sistem abstrak (*abstract system*) dan sistem fisik (*physical system*)
   1. Sistem abstrak adalah system yang berupa pemikiran atau ide yang tidak tampak secara fisik, contoh : Norma Kehidupan
   2. Sistem fisik adalah sistem yang ada secara fisik

b) Sistem alamiah (*natural system*) dan sistem buatan (*human made system*)
   1. Sistem alamiah adalah sistem yang terjadi melalui proses alam
   2. Sistem buatan adalah sistem yang dirancang oleh Manusia

c) Sistem tertentu (*deterministic system*) dan sistem tak tentu (*probabilistic system*)
   1. Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi
   2. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi.

d) Sistem tertutup (*closed system*) dan sistem terbuka (*open system*)
   1. Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya.
   2. Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya

### 2.1.2. Konsep Dasar Informasi

Informasi ibarat darah yang mengalir di dalam tubuh suatu organisasi. Sehingga informasi merupakan salah satu bentuk sumber daya utama dalam suatu organisasi yang digunakan oleh manager untuk mengendalikan perusahaan dalam mencapai tujuan.

Informasi merupakan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. Sedangkan pengertian data itu sendiri merupakan kenyataan yang menggambarkan suatu kejadian–kejadian dan kesatuan yang nyata, atau data adalah representasi dunia nyata yang mewakili suatu objek seperti manusia (pegawai, mahasiswa, pelanggan), hewan, peristiwa, konsep, keadaan dll, yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi atau kombinasinya.

Kualitas suatu informasi tergantung pada 4 hal, yaitu :

a) Akurat

Akurat disini berarti informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan bagi orang yang menerima informasi tersebut. Akurat juga berarti informasi harus jelas mencerminkan maksudnya. Dalam prakteknya, mungkin dalam penyampaian suatu informasi banyak terjadi gangguan (*noise*) yang dapat merubah atau merusak isi dari informasi tersebut.

Komponen akurat meliputi :

1. *Completeness*, berarti informasi yang dihasilkan atau dibutuhkan harus memiliki kelengkapan yang baik, karena bila informasi yang dihasilkan sebagian-sebagian akan mempengaruhi dalam pengambilan keputusan.
2. *Correctness*, berati informasi yang dihasilkan atau dibutuhkan harus memiliki kebenaran.
3. *Security*, berati informasi yang dihasilkan atau dibutuhkan harus memiliki keamanan.

b) Tepat pada waktunya

Informasi yang diterima harus tepat pada waktunya, sebab informasi yang usang (terlambat) tidak mempunyai nilai yang baik, sehingga bila digunakan sebagai dasar dalam pengambilan keputusan akan dapat berakibat fatal. Saat ini mahalnya nilai informasi disebabkan harus cepatnya informasi tersebut didapat, sehingga diperlukan teknologi-teknologi mutakhir untuk mendapatkan, mengolah dan mengirimkannya.

c) Relevan

Informasi harus mempunyai manfaat bagi si penerima. Relevansi informasi akan berbeda untuk setiap orang. Misalnya informasi mengenai sebab-sebab kerusakan mesin produksi kepada akuntan perusahaan adalah kurang relevan dan akan lebih relevan bila ditujukan kepada ahli teknik perusahaan.

d) Ekonomis

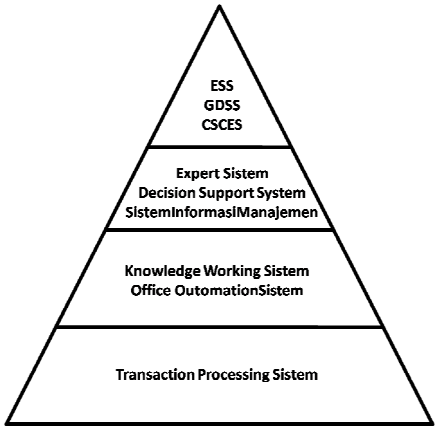
Informasi yang dihasilkan mempunyai manfaat yang lebih besar dibandingkan dengan biaya mendapatkannya dan sebagian besar informasi tidak dapat tepat ditaksir keuntungannya dengan satuan nilai uang tetapi dapat ditaksir nilai efektivitasnya

### 2.1.3. Sistem Informasi

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

Definisi lain sistem informasi adalah kerangka kerja yang mengkoordinasikan sumber daya (manusia dan komputer) untuk mengubah masukan (input) menjadi keluaran (informasi) guna mencapai sasaran-sasaran perusahaan.

Sistem informasi dikembangkan untuk tujuan-tujuan yang bebeda-beda, tergantung pada kebutuhan bisnis. Keanekaragaman sistem informasi dapat dilihat dari gambar 2.1 dibawah ini.




Gambar 2.1. Keanekaragaman Sistem Informasi

## 2.2. Model Analisis

### 2.2.1. Flow Chart Diagram

Flow Chart adalah penyajian yang sistematis tentang proses dan logika dari kegiatan penanganan informasi.

Program Flow Chart adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (instruksi) dengan proses lainnya dalam suatu program. Berikut adalah simbol-simbol yang terdapat pada Flow Chart.

### 2.2.2. Entity Relationship Diagram (ERD)

ERD adalah suatu penyajian data dengan menggunakan Entity dan Relationship.

a). Entity

1. Entity adalah obyek yang dapat dibedakan dalam dunia nyata
2. Entity set adalah kumpulan dari entity yang sejenis

3. Entity set dapat berupa :

   3.a. Obyek secara fisik : Rumah, Kendaraan, Peralatan

   3.b. Obyek secara konsep : Pekerjaan, Perusahaan, Rencana

b). Relationship

1. Relationship adalah hubungan yang terjadi antara satu atau lebih entity.
2. Relationship set adalah kumpulan relationship yang sejenis.




Gambar 2.2. Relationship

c). Atribut

1. Atribut adalah karakteristik dari entity atau relationship, yang menyediakan penjelasan detail tentang entity atau relationship tersebut.
2. Nilai Atribut merupakan suatu data aktual atau informasi yang disimpan pada suatu atribut di dalam suatu entity atau relationship.

d). Jenis-jenis Atribut

1. *Key* : Atribut yang digunakan untuk menentukan suatu entity secara unik.
2. Atribut *Simple* : Atribut yang benilai tunggal.
3. Atribut *Multivalue* : Atribut yang memiliki sekelompok nilai untuk setiap instan entity.
4. Atribut *Composite* : Suatu atribut yang terdiri dari beberapa atribut yang lebih kecil yang mempunyai arti tertentu.
5. Atribut *Derivatif* : Suatu atribut yang dihasilkan dari atribut yang lain.

### 2.2.3. Data Flow Diagram (DFD)

DFD merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh profesional sistem kepada pemakai maupun pembuat program.

Tidak ada aturan baku untuk menggambarkan DFD, tapi dari berbagai referensi yang ada, secara garis besar :

a) Diagram Context

Diagram ini adalah diagram level tertinggi dari DFD yang menggambarkan hubungan sistem dengan lingkungan luarnya. Cara :

1. Tentukan nama sistemnya.
2. Tentukan batasan sistemnya.
3. Tentukan terminator apa saja yang ada dalam sistem.
4. Tentukan apa yang diterima/diberikan terminator dari/pada sistem.
5. Gambarkan diagram context.

b) Diagram Level Zero

Diagram ini adalah dekomposisi dari diagram Context. Cara :

1. Tentukan proses utama yang ada pada sistem.
2. Tentukan apa yang diberikan/diterima masing-masing proses pada/dari sistem sambil memperhatikan konsep keseimbangan (alur data yang keluar/masuk dari suatu level harus sama dengan alur data yang masuk/keluar pada level berikutnya)

3. Apabila diperlukan, munculkan data store (master) sebagai sumber maupun tujuan alur data.
4. Gambarkan diagram level zero.
   4.a. Hindari perpotongan arus data
   4.b. Beri nomor pada proses utama (nomor tidak menunjukkan urutan proses).

c) Diagram Level Satu

Diagram ini merupakan dekomposisi dari diagram level zero. Cara :

1. Tentukan proses yang lebih kecil (sub-proses) dari proses utama yang ada di level zero.
2. Tentukan apa yang diberikan/diterima masing-masing sub-proses pada/dari sistem dan perhatikan konsep keseimbangan.
3. Apabila diperlukan, munculkan data store (transaksi) sbg sumber maupun tujuan alur data.
4. Gambarkan DFD level Satu
   4.a. Hindari perpotongan arus data.
   4.b. Beri nomor pada masing-masing sub-proses yang menunjukkan dekomposisi dari proses sebelumnya. Contoh : 1.1, 1.2, 2.1

d) DFD Level Dua, Tiga, ...

Diagram ini merupakan dekomposisi dari level sebelumnya. Proses dekomposisi dilakukan sampai dengan proses siap dituangkan ke dalam program. Aturan yang digunakan sama dengan level satu.

## 2.3. Stuktured Query Language (SQL)

SQL merupakan bahasa *query* standar yang digunakan untuk mengakses database relasional. *Standard* internasional terhadap SQL pertama kali dilakukan oleh ANSI (*American National Standard Institution*) melalui publikasi *Database Language* SQL. Meskipun sebenarnya SQL tidak ditemukan oleh ANSI. SQL sendiri, merupakan hasil riset IBM, akan tetapi perusahaan lain mengambil dan mengembangkannya menjadi lebih baik. Salah satu contohnya adalah Oracle yang mengalahkan IBM dalam persaingan SQL.

Dalam SQL, terdapat dua pengertian atau definisi, yaitu *interactive* dan *Embedded* SQL. *Interactive* SQL digunakan untuk dioprasikan secara langsung pada suatu database agar menghasilkan *output* untuk langsung dilihat hasilnya. Pada SQL bentuk ini, pengetikan perintah eksekusi dilakukan pada saat itu juga, dan segera dapat dilihat hasil *output*-nya. Sementara embedded SQL terdiri atas perintah-perintah SQL yang diletakan didalam program, umumnya menggunakan bahasa pemrograman lain, seperti Delphi atau Visual Basic. Cara seperti ini membutuhkan penyesuaian yang baik antara bahasa pemrograman dengan struktur SQL.

SQL sendiri terbagi menjadi 3 bagian, yaitu :

1. Data Definition Language (DDL)

   DDL dapat disebut juga dengan schema definition language dalam versi ANSI. DDL terdiri atas perintah-perintah SQL untuk membuat suatu objek seperti *table*, *index*, *view*, dan lain-lain dalam suatu database.

2. Data Manipulation Language (DML)

   Merupakan kumpulan perintah-perintah SQL untuk menentukan atau memanipulasi isi suatu table

3. Data Control Language (DCL)

   DCL terdiri atas perintah untuk menentukan izin akses dari *user* dam suatu database

## 2.4. Teori Barcode

Mungkin tanpa disadari setiap hari kita akan menemui barcode, misalnya pada produk makanan, obat, barang konsumer yang kita miliki, tiket pesawat, bahkan sampai disampul surat yang kita terima (biasanya dari luar negeri) pun kita temui barcode. Barcode pada dasarnya adalah susunan garis vertikal hitam dan putih dengan ketebalan yang berbeda, sangat sederhana tetapi sangat berguna, dengan kegunaan untuk menyimpan data-data spesifik misalnya kode produksi, tanggal kadaluwarsa, nomor identitas dengan mudah dan murah, dan lain sebagainya.

Jenis barcode sangatlah banyak akan tetapi penulis akan mencoba membatasi pada jenis –jenis yang populer digunakan, antara lain :

### 2.4.1. *Code 39* (*3 of 9*).

*Code 39* dapat mengkodekan karakter alphanumeric yaitu angka desimal dan huruf besar serta tambahan karakter spesial (-, *, $, /, %,+, dan space). Satu karakter dalam *code 39* terdiri dari sembilan elemen, yaitu : lima bar (garis vertikal hitam) dan empat spasi (garis vertikal putih) yang di susun

bergantian antara bar dan spasi. Tiga dari sembilan elemen tersebut memiliki ketebalan lebih tebal dari yang lainnya oleh karenanya kode ini bisa disebut juga *code 3 of 9*, 3 elemen yang lebih besar tersebut terdiri dari dua bar dan satu spasi. Elemen yang lebar mewakili digit biner 1 dan elemen yang sempit mewakili digit biner 0. Tabel karakter *code 39* beserta nilai karakternya tercantum pada tabel dibawah ini.

| **Karakter ASCII** | **Digit Biner** | | | | | | | | | **Nilai Karakter** |
|---|---|---|---|---|---|---|---|---|---|---|
| | **B** | **S** | **B** | **S** | **B** | **S** | **B** | **S** | **B** | |
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| 2 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 2 |
| 3 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 3 |
| 4 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 4 |
| 5 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 5 |
| 6 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 6 |
| 7 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 7 |
| 8 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 8 |
| 9 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 9 |
| A | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 10 |
| B | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 11 |
| C | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 12 |
| D | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 13 |
| E | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 14 |
| F | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 15 |
| G | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 16 |
| H | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 17 |
| I | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 18 |
| J | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 19 |
| K | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 20 |
| L | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 21 |
| M | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 22 |
| N | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 23 |
| O | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 24 |
| P | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 25 |
| Q | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 26 |
| R | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 27 |
| S | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 28 |
| T | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 29 |
| U | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 30 |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| V | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 31 |
| W | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 32 |
| X | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 33 |
| Y | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 34 |
| Z | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 35 |
| - | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 36 |
| . | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 37 |
| SPACE | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 38 |
| * | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | - |
| $ | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 39 |
| / | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 40 |
| + | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 41 |
| % | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 42 |

Stuktur barcode code 39 adalah sebagai berikut :

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| QZ | SC | ICG | C1 | ICG | C2 | ICG | … | CN | ICG | CC | ICG | PC | QZ |

Dimana :

**X** : Ketebalan element yang sempit (minimal 0,99 mm)

**QZ** : Quiet Zone atau Start-Stop Margin dengan ketebalan minimum 6 mm atau 10 kali X

**SC** : Start Character (karakter*)

**ICG** : InterCharacterGap dengan ketebalan 1 kali X

**C1 .. CN** : Character ke-1 sampai dengan character ke-N.

**CC** : Check Character

**PC** : Stop Character (karakter*)

Untuk dapat membedakan garis vertikal lebar dan sempit maka perbandingan ketebalan antara garis vertikal lebar dan sempit minimum 2:1, dimana perbandingan 3:1 akan lebih baik.

Lebar keseluruhan barcode dapat dirumuskan sebagai berikut:

$$L = \underbrace{N(3RX+7X)}_{I} + \underbrace{(6RX+13X)}_{II} + \underbrace{(3RX+7X)}_{III} + \underbrace{(M1 + M2)}_{IV}$$

Dimana :

L : Lebar keseluruhan barcode

N : Jumlah karakter

R : Perbandingan garis vertikal lebar dan sempit

X : Ketebalan garis vertikal sempit

I : Lebar N karakter plus N intercharactergap

II : Lebar start dan stop character plus 1 intercharactergap antara start character dan character pertama.

III : Lebar Check Character plus 1 intercharcatergap

IV : Lebar 2 kali quiet zone (M1(startmargin)+M2(stopmargin)).

Check character adalah sisa dari jumlah seluruh nilai karakter dibagi dengan 43, sebagaicontoh :

Message : CODE39

Karakter : C O D E SPACE 3 9

Nilai karakter : 12 24 13 14 38 3 9

JUMLAH : 12 + 24 + 13 + 14 + 38 + 3 + 9 = 113

113 / 43 = 2 sisa 27

27 adalah nilai dari karakter R, maka

Message + checkcharacter : CODE39R

Contoh Barcode Code 39



Gambar 2.3. Contoh Barcode Code 39

### 2.4.2. *Extended* Code 39

Extended code 39 adalah pengembangan dari code39 yang dapat mengkodekan seluruh karakter ASCII dengan cara mengkodekannya dengan pasangan code 39, yang dapat dilihat pada tabel dibawah ini.

| ASCII | Exc. Code 39 | ASCII | Exc. Code 39 | ASCII | Exc. Code 39 | ASCII | Exc. Code 39 |
|---|---|---|---|---|---|---|---|
| NU | %U | SP | SPACE | @ | %V | ` | %W |
| SH | $A | ! | /A | A | A | a | +A |
| SX | $B | “ | /B | B | B | b | +B |
| EX | $C | # | /C | C | C | c | +C |
| ET | $D | $ | /D | D | D | d | +D |
| EQ | $E | % | /E | E | E | e | +E |
| AK | $F | & | /F | F | F | f | +F |
| BL | $G | ' | /G | G | G | g | +G |
| BS | $H | ( | /H | H | H | h | +H |
| HT | $I | ) | /I | I | I | I | +I |
| LF | $J | * | /J | J | J | j | +J |
| VT | $K | + | /K | K | K | k | +K |
| FF | $L | , | /L | L | L | l | +L |
| CR | $M | - | /M | M | M | m | +M |
| SO | $N | . | /N | N | N | n | +N |
| SI | $O | / | /O | O | O | o | +O |
| DL | $P | 0 | /Patau0 | P | P | p | +P |
| D1 | $Q | 1 | /Qatau1 | Q | Q | q | +Q |
| D2 | $R | 2 | /Ratau2 | R | R | r | +R |
| D3 | $S | 3 | /Satau3 | S | S | s | +S |
| D4 | $T | 4 | /Tatau4 | T | T | t | +T |
| NK | $U | 5 | /Uatau5 | U | U | u | +U |
| SY | $V | 6 | /Vatau6 | V | V | v | +V |
| EB | $W | 7 | /Watau7 | W | W | w | +W |
| CN | $X | 8 | /Xatau8 | X | X | x | +X |
| EM | $Y | 9 | /Yatau9 | Y | Y | y | +Y |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| SB | $Z | : | /Z | Z | Z | z | +Z |
| EC | %A | ; | %F | [ | %K | { | %P |
| FS | %B | < | %G | \ | %L | \| | %Q |
| GS | %C | = | %H | ] | %M | } | %R |
| RS | %D | > | %I | ^ | %N | ~ | %S |
| US | %E | ? | %J | _ | %O | DEL | %T |

Keterangan :

- Karakter $ / + % mengkodekan karakter ASCII-nya sendiri jika diikuti oleh digit, atau spasi, atau dirinya sendiri ($ / + %), atau salah satu dari tiga spesial karakter yang lain (-.*), atau stop character.
- %X, %Y, %Z mengkodekan karakter ASCII DEL.

### 2.4.3. *Interleaved* 2 of 5 (ITF)

ITF barcode hanya dapat mengkodekan angka saja dan sering digunakan pada produk-produk yang memiliki kemasan dengan permukaan yang tidak rata (misalnya corugatedbox), hal ini disebabkan struktur dan cara pengkodean ITF yang unik.

Setiap karakter pada ITF barcode dikodekan dengan 5 elemen yaitu 2 elemen tebal dan 3 elemen sempit, dimana elemen tebal mewakili digit biner 1 sedangkan elemen tipis mewakili digit biner 0 dengan perbandingan ketebalan antara elemen tebal dengan elemen tipis 2:1 sampai 3:1.

Keunikan dari ITF adalah pengkodean karakternya apakah menggunakan bar ataukah menggunakan spasi tergantung pada posisi sesuai dengan namanya interleaved, atau lebih jelasnya sebagai berikut : Karakter

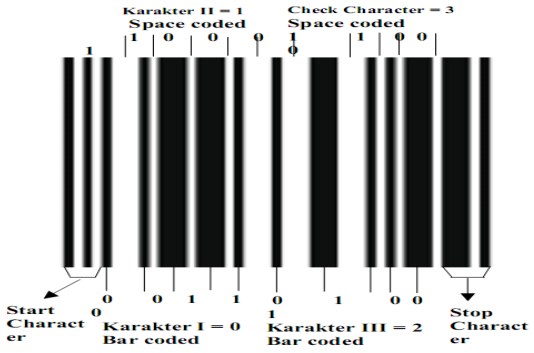
pertama dikodekan menggunakan bar setelah start character, sedangkan karakter kedua dikodekan menggunakan spasi secara interleaved pada karakter pertama, karena sifat berpasang-pasangan itulah panjang message termasuk check character haruslah genap jika jumlahnya ganjil maka harus ditambahkan karakter 0 pada awal message, sebelum diillustrasikan contoh ITF barcode, akan lebih jelas jika melihat tabel karaterset ITF dibawah ini.

| Karakter ASCII | Kode Biner | Nilai Karakter |
|---|---|---|
| 1 | 10001 | 1 |
| 2 | 01001 | 2 |
| 3 | 11000 | 3 |
| 4 | 00101 | 4 |
| 5 | 10100 | 5 |
| 6 | 01100 | 6 |
| 7 | 00011 | 7 |
| 8 | 10010 | 8 |
| 9 | 01010 | 9 |
| 0 | 00110 | 0 |
| Start | 0000* | |
| Stop | 100* | |

Keterangan :

Start dan Stop character tidak dikodekan secara interleaved tetapi bergantian bar dan spasi.

Contoh ITF Barcode




Gambar 2.4. Contoh Barcode ITF

Contoh perhitungan check character ITF barcode adalah sebagai berikut:

- Message : 2632534
- Karakter : 2632534
- Posisi : EOEOEOE (karakter paling kanan dimulai sebagai E)
- Jumlah dari posisi odd : 6+2+3=11
- Jumlah dari posisi even : 2+3+5+4=14
- Jumlah odd x3 : 14x3=42
- Step 6 + step 7 : 11+42=53
- Angka terkecil untuk menjadi kelipatan 10 : 7
- Check character : 7
- Tag to be encoded : 26325347

Jumlah keseluruhan karakter plus check character genap sehingga tidak diperlukan leading zero.

Struktur ITF barcode dapat dilihat pada gambar dibawah ini :

| Quiet Zone | Start Character | Message + Check Character | EnterStop Character | Quiet Zone |
|---|---|---|---|---|

Lebar dan tinggi ITF barcode mengikuti aturan sbb :

$$L = (C\,(2R + 3) + 6 + R)\,X$$

Dimana :

L : Lebar barcode (tidak termasuk quiet zone)

C : Jumlah karakter

X : Lebar elemen tersempit (minimum 0.19mm)

R : Perbandingan elemen lebar dengan elemen sempit (2:1 sampai 3:1)

Quiet zone minimum 10 kali X

Tinggi barcode minimum 0.15 kali lebar barcode.

Untuk menghindari terjadi partial scan atau pembacaan barcode yang terpotong yang dapat menyebabkan kesalahan informasi, maka sering kali ITF barcode diberi tambahan bearer bar yaitu bar sepanjang barcode yang ditempatkan diatas dan dibawah barcode seperti tampak dibawah ini.




Gambar 2.5. Contoh Barcode ITF dengan bearer bar

### 2.4.4. *Code* 128

Code 128 adalah barcode dengan kerapatan tinggi, dapat mengkodekan keseluruhan simbol ASCII (128karakter) dalam luasan yang paling minim dibandingkan dengan barcode jenis lain, hal ini disebabkan karena code 128 menggunakan 4 ketebalan elemen (bar atau spasi) yang berbeda (jenis

yang lain kebanyakan menggunakan 2 ketebalan elemen yang berbeda). Setiap karakter pada code 128 dikodekan oleh 3 bar dan 3 spasi (atau 6 elemen) dengan ketebalan masing-masing elemen 1 sampai 4 kali ketebalan minimum (module), jika dihitung dengan satuan module maka tiap karakter code 128 terdiri dari 11 module kecuali untuk stop character yang terdiri dari 4 bar 3 spasi (13module). Jumlah total module untuk bar selalu genap sedangkan untuk spasi selalu ganjil, selain itu code 128 memiliki 3 start character yang berbeda sehingga code 128 memiliki 3 subset karakter yang bersesuaian dengan start characternya seperti tampak pada tabel karakter set dibawah ini.

| Value | Code Set A | Code Set B | Code Set C | Bar / Space Pattern B S B S B S |
|---|---|---|---|---|
| 0 | SP | SP | 00 | 2 1 2 2 2 2 |
| 1 | ! | ! | 01 | 2 2 2 1 2 2 |
| 2 | " | " | 02 | 2 2 2 2 2 1 |
| 3 | # | # | 03 | 1 2 1 2 2 3 |
| 4 | $ | $ | 04 | 1 2 1 3 2 2 |
| 5 | % | % | 05 | 1 3 1 2 2 2 |
| 6 | & | & | 06 | 1 2 2 2 1 3 |
| 7 | ' | ' | 07 | 1 2 2 3 1 2 |
| 8 | ( | ( | 08 | 1 3 2 2 1 2 |
| 9 | ) | ) | 09 | 2 2 1 2 1 3 |
| 10 | * | * | 10 | 2 2 1 3 1 2 |
| 11 | + | + | 11 | 2 3 1 2 1 2 |
| 12 | , | , | 12 | 1 1 2 2 3 2 |
| 13 | - | - | 13 | 1 2 2 1 3 2 |
| 14 | . | . | 14 | 1 2 2 2 3 1 |
| 15 | / | / | 15 | 1 1 3 2 2 2 |
| 16 | 0 | 0 | 16 | 1 2 3 1 2 2 |
| 17 | 1 | 1 | 17 | 1 2 3 2 2 1 |
| 18 | 2 | 2 | 18 | 2 2 3 2 1 1 |
| 19 | 3 | 3 | 19 | 2 2 1 1 3 2 |
| 20 | 4 | 4 | 20 | 2 2 1 2 3 1 |
| 21 | 5 | 5 | 21 | 2 1 3 2 1 2 |
| 22 | 6 | 6 | 22 | 2 2 3 1 1 2 |
| 23 | 7 | 7 | 23 | 3 1 2 1 3 1 |

| | | | | |
|---|---|---|---|---|
| 24 | 8 | 8 | 24 | 3 1 1 2 2 2 |
| 25 | 9 | 9 | 25 | 3 2 1 1 2 2 |
| 26 | : | : | 26 | 3 2 1 2 2 1 |
| 27 | ; | ; | 27 | 3 1 2 2 1 2 |
| 28 | < | < | 28 | 3 2 2 1 1 2 |
| 29 | = | = | 29 | 3 2 2 2 1 1 |
| 30 | > | > | 30 | 2 1 2 1 2 3 |
| 31 | ? | ? | 31 | 2 1 2 3 2 1 |
| 32 | @ | @ | 32 | 2 3 2 1 2 1 |
| 33 | A | A | 33 | 1 1 1 3 2 3 |
| 34 | B | B | 34 | 1 3 1 1 2 3 |
| 35 | C | C | 35 | 1 3 1 3 2 1 |
| 36 | D | D | 36 | 1 1 2 3 1 3 |
| 37 | E | E | 37 | 1 3 2 1 1 3 |
| 38 | F | F | 38 | 1 3 2 3 1 1 |
| 39 | G | G | 39 | 2 1 1 3 1 3 |
| 40 | H | H | 40 | 2 3 1 1 1 3 |
| 41 | I | I | 41 | 2 3 1 3 1 1 |
| 42 | J | J | 42 | 1 1 2 1 3 3 |
| 43 | K | K | 43 | 1 1 2 3 3 1 |
| 44 | L | L | 44 | 1 3 2 1 3 1 |
| 45 | M | M | 45 | 1 1 3 1 2 3 |
| 46 | N | N | 46 | 1 1 3 3 2 1 |
| 47 | O | O | 47 | 1 3 3 1 2 1 |
| 48 | P | P | 48 | 3 1 3 1 2 1 |
| 49 | Q | Q | 49 | 2 1 1 3 3 1 |
| 50 | R | R | 50 | 2 3 1 1 3 1 |
| 51 | S | S | 51 | 2 1 3 1 1 3 |
| 52 | T | T | 52 | 2 1 3 3 1 1 |
| 53 | U | U | 53 | 2 1 3 1 3 1 |
| 54 | V | V | 54 | 3 1 1 1 2 3 |
| 55 | W | W | 55 | 3 1 1 3 2 1 |
| 56 | X | X | 56 | 3 3 1 1 2 1 |
| 57 | Y | Y | 57 | 3 1 2 1 1 3 |
| 58 | Z | Z | 58 | 3 1 2 3 1 1 |
| 59 | [ | [ | 59 | 3 3 2 1 1 1 |
| 60 | \ | \ | 60 | 3 1 4 1 1 1 |
| 61 | ] | ] | 61 | 2 2 1 4 1 1 |
| 62 | ^ | ^ | 62 | 4 3 1 1 1 1 |
| 63 | _ | _ | 63 | 1 1 1 2 2 4 |
| 64 | NUL | ` | 64 | 1 1 1 4 2 2 |
| 65 | SOH | a | 65 | 1 2 1 1 2 4 |
| 66 | STX | b | 66 | 1 2 1 4 2 1 |
| 67 | ETX | c | 67 | 1 4 1 1 2 2 |
| 68 | EOT | d | 68 | 1 4 1 2 2 1 |
| 69 | ENQ | e | 69 | 1 1 2 2 1 4 |

| | | | | |
|---|---|---|---|---|
| 70 | ACK | f | 70 | 1 1 2 4 1 2 |
| 71 | BEL | g | 71 | 1 2 2 1 1 4 |
| 72 | BS | h | 72 | 1 2 2 4 1 1 |
| 73 | HT | i | 73 | 1 4 2 1 1 2 |
| 74 | LF | j | 74 | 1 4 2 2 1 1 |
| 75 | VT | k | 75 | 2 4 1 2 1 1 |
| 76 | FF | I | 76 | 2 2 1 1 1 4 |
| 77 | CR | m | 77 | 4 1 3 1 1 1 |
| 78 | SO | n | 78 | 2 4 1 1 1 2 |
| 79 | SI | o | 79 | 1 3 4 1 1 1 |
| 80 | DLE | p | 80 | 1 1 1 2 4 2 |
| 81 | DC1 | q | 81 | 1 2 1 1 4 2 |
| 82 | DC2 | r | 82 | 1 2 1 2 4 1 |
| 83 | DC3 | s | 83 | 1 1 4 2 1 2 |
| 84 | DC4 | t | 84 | 1 2 4 1 1 2 |
| 85 | NAK | u | 85 | 1 2 4 2 1 1 |
| 86 | SYN | v | 86 | 4 1 1 2 1 2 |
| 87 | ETB | w | 87 | 4 2 1 1 1 2 |
| 88 | CAN | x | 88 | 4 2 1 2 1 1 |
| 89 | EM | y | 89 | 2 1 2 1 4 1 |
| 90 | SUB | z | 90 | 2 1 4 1 2 1 |
| 91 | ESC | { | 91 | 4 1 2 1 2 1 |
| 92 | FS | \| | 92 | 1 1 1 1 4 3 |
| 93 | GS | } | 93 | 1 1 1 3 4 1 |
| 94 | RS | ~ | 94 | 1 3 1 1 4 1 |
| 95 | US | DEL | 95 | 1 1 4 1 1 3 |
| 96 | FNC3 | FNC3 | 96 | 1 1 4 3 1 1 |
| 97 | FNC2 | FNC2 | 97 | 4 1 1 1 1 3 |
| 98 | SHIFT | SHIFT | 98 | 4 1 1 3 1 1 |
| 99 | ODEC | CODEC | 99 | 1 1 3 1 4 1 |
| 100 | ODEB | FNC4 | CODEB | 1 1 4 1 3 1 |
| 101 | FNC4 | CODEA | CODEA | 3 1 1 1 4 1 |
| 102 | FNC1 | FNC1 | FNC1 | 4 1 1 1 3 1 |
| 103 | StartA | StartA | StartA | 2 1 1 4 1 2 |
| 104 | StartB | StartB | StartB | 2 1 1 2 1 4 |
| 105 | StartC | StartC | StartC | 2 1 1 2 3 2 |
| 106 | Stop | Stop | Stop | 23 3 1 1 1 2 |

Code 128 memiliki fitur untuk dapat bergeser dari sub set yang satu ke sub set yang lainnya dengan menggunakan karakter CODE dan SHIFT, CODEX menyebabkan seluruh message bergeser menjadi sub set X (misalnya CODE A pada sub set B membuat message menjadi sub set A), sedangkan SHIFT

menyebabkan satu karakter didepannya bergeser subset (ini hanya berlaku untuk sub set A ke sub set B atau sebaliknya).

Struktur code 128 barcode seperti terlihat dibawah ini :




Gambar 2.6. contoh barcode code 128

Dimana tinggi barcode minimum 0.15 kali lebar barcode dan lebar barcode dinyatakan dalam rumus :

**L = (11C+35) X untuk alpha numeric (CODE A dan CODE B)**

**L = (5.5C+35) X untuk double density numeric only (CODEC)**

Dimana :

L : Lebar barcode total termasuk quietzone

C : Jumlah karakter

X : Lebar module (elemen yang tersempit)

CODE C dikatakan sebagai double densisty numeric only dan dalam perhitungan lebar perkarakternya hanya 5.5X sebab satu karakter CODE C mewakili 2 digit numeric (lihat tabel diatas).

Perhitungan check character code 128 sbb:

1. Message : CODE 128
2. Karakter : StartA CODE 128
3. Nilai karakter : 103 35 47 36 37 0 17 18 24
4. Posisi : - 1 2 3 4 5 6 7 8
5. Perhitungan Total : 103 + (35 x 1) + (47 x 2) + (36 x 3) + (37 x 4) + (0 x 5) + (17 x 6) + (18 x 7) + (24 x 8) = 908

   908/103 = 8 remainder 84

   84 = DC4
6. Message akhir : (StartA) CODE 128 (DC4) (STOP)

## 2.5. Software Pendukung

### 2.5.1. Borland Delphi

Delphi adalah suatu program berbasis bahasa Pascal yang berjalan dalam lingkungan Windows. Delphi telah memanfaatkan suatu teknik pemrograman yang disebut RAD yang telah membuat pemrograman menjadi lebih mudah. Delphi adalah suatu bahasa pemrograman yang telah memanfaatkan metode pemrograman Object Oriented Programming (OOP). Elemen-elemen dan fungsi-fungsinya dalam pembuatan progam dari delphi sebagai berikut :

a) Object Inspector : suatu window yang berguna untuk mengatur suatu object baik properti, events, dan method.

b) Form : Digunakan sebagai layar/window yang digunakan sebagai lembar kerja kita. Di form-lah semua komponen seperti tombol dan komponen lainnya disimpan.

c) Window Unit/Source Code : Window/layar yang berisi perintah-perintah yang akan dieksekusi oleh komputer. Di layar inilah kita mengisikan program-program.

d) Component Palette : Layar yang berisikan komponen-komponen yang dipakai dipakai dalam program kita.

### 2.5.2. MySQL

MySQL dikembangkan oleh sebuah perusahaan Swedia bernama MySQL AB, yang kala itu bernama TcX DataKonsult AB, sejak sekitar 1994–1995, meski cikal bakal kodenya bisa disebut sudah ada sejak 1979. TcX adalah perusahaan pengembang software dan konsultan database.

Versi awal MySQL memiliki banyak keterbatasan, tetapi sekitar tshun 1998-1999 MySQL menjadi semakin popular dan banyak dilirik orang karena stabilitasnya sudah mulai baik dan kecepatannya meningkat, sudah tersedia di berbagai platform, termasuk Windows. Versi akhir MySQL yaitu seri 3.22 yang banyak dipakai di berbagai instalasi, sehingga MySQL AB tetap memberikan dukungan *technical support*.

Kelebihan utama yang dimiliki MySQL sehingga begitu banyak digunakan adalah:

a) MySQL tersedia di berbagai platform Linux dan berbagai varian Unix
b) Fitur-fitur yang dimiliki MySQL memang yang biasanya banyak dibutuhkan.
c) MySQL memiliki *overhead* koneksi yang rendah.

### 2.5.3. MyODBC

MyODBC merupakan komponen yang penting bagi pengembang. Ini memungkinkan kita menawarkan pada client, antarmuka yang mudah dan plat form yang sudah familiar. MyODBC merupakan driver yang khusus digunakan untuk berkomunikasi dengan database MySQL.