

# Bab 5

## Transmisi Sinyal Digital

### Tujuan dari Bab ini:

- Pembaca memahami berbagai jenis pengkodean data digital menjadi sinyal digital beserta aplikasinya.
- Pembaca memahami berbagai jenis pengkodean sinyal analog menjadi data digital beserta aplikasinya.
- Pembaca memahami karakteristik pencacahan sinyal analog untuk menghasilkan data digital.

Dalam Bab 3 kita telah mempelajari bahwa dalam proses transmisi data selalu dikonversi menjadi sinyal terlebih dahulu. Data tersebut bisa berbentuk data analog dan data digital. Sinyal juga dapat berupa sinyal analog dan sinyal digital. Transmisi *baseband* adalah representasi data analog atau data digital menjadi sinyal digital pada proses transmisi. Sedangkan transmisi *passband* (akan dibahas dalam Bab 6) adalah representasi data analog atau data digital menjadi sinyal analog pada proses transmisi. Transmisi *passband* ditandai dengan pergeseran frekuensi dari frekuensi data yang umumnya rendah menjadi frekuensi sinyal yang tinggi sesuai dengan frekuensi gelombang pembawa (*carrier frequency*).

Pada bagian ini kita hanya membicarakan proses konversi dari data digital menjadi sinyal digital dan proses konversi dari data analog menjadi sinyal digital. Sedangkan pada Bab berikutnya kita akan mendiskusikan proses konversi dari data digital dan data analog menjadi sinyal analog. Proses konversi data menjadi sinyal seringkali juga disebut dengan istilah pengkodean (*encoding*) (Stalling, 2001).

## 5.1. Konversi (Pengkodean) Data Digital Menjadi Sinyal Digital

Terdapat tiga macam cara untuk melakukan proses konversi dari data digital menjadi sinyal digital, yaitu *line coding*, *block coding* dan *scrambling*. Namun sebelum membicarakan ketiga macam teknik konversi tersebut mari kita bahas terlebih dahulu hubungan antara kecepatan data (*data rate*) dan kecepatan sinyal (*signal rate*) dan syarat-syarat agar transmisi sinyal digital dapat berlangsung dengan baik.

**Terdapat tiga macam cara untuk melakukan proses konversi dari data digital menjadi sinyal digital, yaitu *line coding*, *block coding* dan *scrambling*.**

Kecepatan pengiriman sinyal diwakili oleh beberapa istilah, antara lain: *baud rate*, *modulation rate* atau *pulse rate*. Dalam buku ini kita akan menggunakan istilah *baud rate* dengan satuan *baud* untuk menyatakan kecepatan pengiriman sinyal digital. Secara logis kita tahu bahwa dalam komunikasi data diharapkan agar kecepatan data dapat dicapai setinggi-tingginya sedangkan kecepatan pengiriman sinyal dapat dicapai serendah-rendahnya.

Kecepatan data tinggi dalam proses transmisi berarti bahwa sejumlah besar data dapat dikirimkan dalam satu satuan waktu. Karena itu semakin tinggi data rate berarti semakin besar jumlah data yang dapat dikirimkan dalam satu satuan waktu. Sedangkan kecepatan pengiriman sinyal diharapkan menjadi rendah karena berkaitan dengan bandwidth dari sinyal. Semakin rendah *baud rate*, berarti semakin kecil pula jumlah bandwidth yang dibutuhkan untuk mentransmisikan sinyal.

Hubungan antara kecepatan sinyal dan kecepatan data dinyatakan dalam persamaan 5.1.

$$S = k \times \frac{1}{m} \times R \quad (5.1)$$

Simbol  $S$  merepresentasikan kecepatan sinyal (signal rate) rata-rata dalam satuan *baud*,  $k$  adalah konstanta yang dapat berubah-ubah tergantung pada jenis modulasi yang

digunakan,  $m$  adalah jumlah elemen data yang dapat dibawa oleh setiap elemen sinyal (*waveform*), dan  $R$  adalah kecepatan data (*data rate*). Untuk pengkodean data digital menjadi sinyal digital, nilai rata-rata dari  $k$  adalah  $\frac{1}{2}$ .

*Contoh 5.1.*

Sebuah sinyal digital menggunakan High-bit-rate Digital Subscriber Line membawa data sedemikian rupa sehingga 2 buah elemen data dikodekan sebagai 1 buah elemen sinyal. Jika kecepatan data rata-rata yang dapat dicapai pada saat itu adalah 1,544 Mbps, berapa kecepatan sinyal rata-rata dari sinyal HDSL?

*Jawaban:*

Pada pengkodean sinyal HDSL diketahui bahwa nilai  $m=2$ . Maka  $S=1/2 \times 1/2 \times 1,544$  Mbps = 0,386 Mbaud.

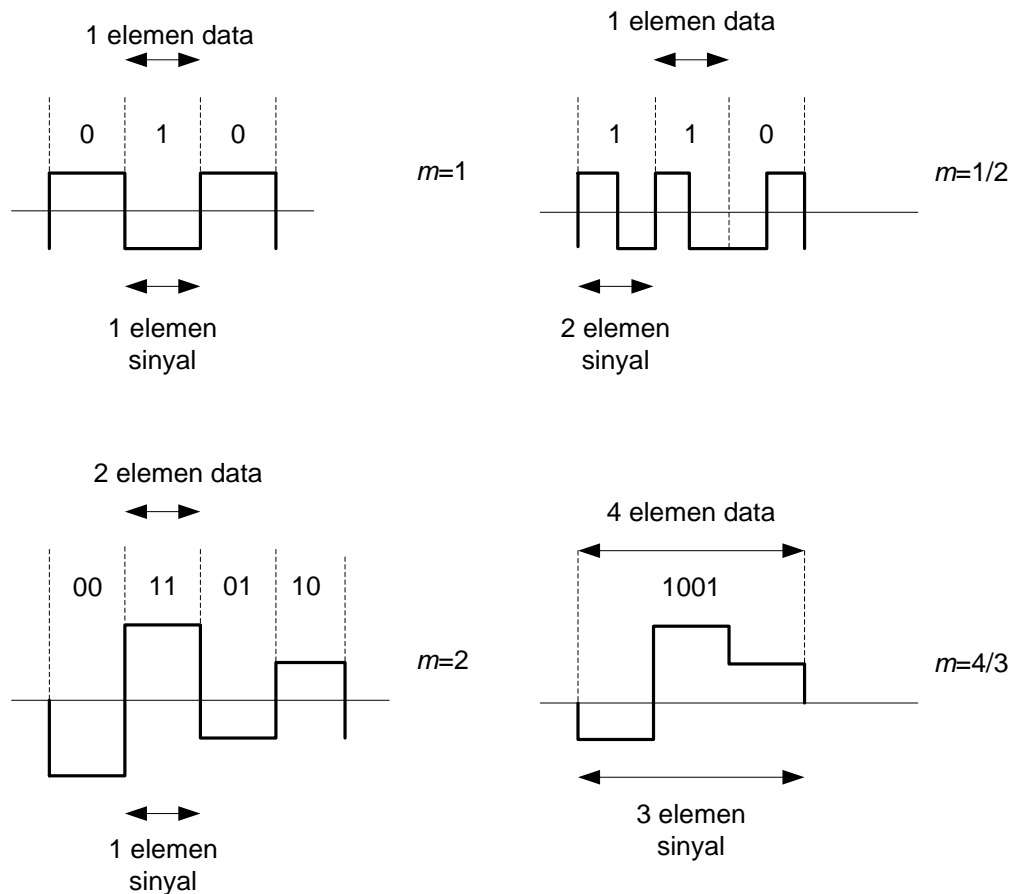
**Dalam komunikasi data diharapkan agar  
kecepatan data dapat dicapai setinggi-tingginya sedangkan  
kecepatan pengiriman sinyal dapat dicapai serendah-rendahnya.**

Dalam persamaan 5.1, terdapat variabel  $m$  yang belum mendapatkan penjelasan gamblang. Kasus paling sederhana adalah apabila kita mengirimkan 1 bit data dimana setiap 1 bit data tersebut diwakili oleh 1 buah sinyal digital. Nilai konstanta  $m$  untuk kasus semacam ini adalah 1. Tetapi hal semacam ini tidak efisien ditinjau dari sisi pemakaian bandwidth dari sinyal. Agar tujuan komunikasi data untuk memaksimalkan jumlah data yang dapat dibawa oleh sebuah sinyal dapat dicapai, sebagai contoh kita dapat membuat nilai  $m$  menjadi 2. Nilai  $m=2$  berarti 1 buah sinyal *waveform* membawa 2 bit data. Lihat Gambar 5.1. Dalam komunikasi data, jumlah data yang dibawa oleh sebuah elemen sinyal bervariasi mulai dari 2, 4, 8, 16, 32, 64, dan 128. Namun perlu diingat bahwa semakin besar nilai  $m$  akan menyebabkan semakin sulit proses deteksi pada sisi penerima.

Konversi data digital menjadi sinyal digital dengan nilai  $m=1/2$  sekalipun meningkatkan bandwidth dari sinyal, dalam praktek lebih sinyal semacam ini dibutuhkan untuk proses

sinkronisasi. Sebagai contoh, pada komunikasi serial yang digunakan oleh komputer, apabila jenis komunikasi serial sinkron digunakan maka setiap beberapa byte selalu disisipkan bit-bit sinkronisasi untuk memberikan kesempatan bagi terminal penerima melakukan sinkronisasi waktu dengan terminal pengirim.

Kita dapat melihat sekarang bahwa cukup banyak pertimbangan-pertimbangan yang harus dilakukan sebelum proses transmisi data terjadi. Selain pertimbangan bandwidth dan sinkronisasi, pengkodean data digital menjadi sinyal digital juga harus mempertimbangkan struktur dari deretan bit yang akan ditransmisikan.



Gambar 5.1. Modulasi data digital untuk berbagai nilai  $m$

Sinyal dengan level tegangan konstan tidak disukai dalam transmisi sinyal digital, misalnya pengiriman data dengan jumlah bit 1 berderet panjang akan dikonversi menjadi tegangan konstan (komponen DC) sebesar -1 Volt sampai seluruh bit selesai dikirimkan.

Pertama, tegangan konstan seperti ini tidak diharapkan karena penurunan energi sinyal (atenuasi) di sisi penerima dapat menyebabkan kesulitan deteksi. Kedua, di samping atenuasi, tegangan konstan seperti itu memiliki frekuensi nol (seperti telah dibahas dalam Bab 3), padahal frekuensi nol tidak dapat dilewatkan melalui saluran komunikasi. Sebagai contoh saluran-saluran telepon kabel tidak dapat melewatkan sinyal dengan frekuensi di bawah 200 Hz. Ketiga, tegangan konstan juga dapat menyebabkan pergeseran daya rata-rata dari sinyal. Padahal proses deteksi pada sisi penerima sangat mendasarkan pada perhitungan daya rata-rata dari sinyal. Pergeseran daya rata-rata sinyal seperti ini seringkali disebut dengan istilah *baseline wandering*.

Faktor-faktor lain yang harus dimiliki oleh sinyal digital adalah: memiliki kemampuan untuk mendeteksi kesalahan dalam proses transmisi, memiliki ketahanan terhadap gangguan-gangguan transmisi seperti derau dan interferensi, memiliki kompleksitas rendah pada saat diimplementasikan.

---

---

**Unsur-unsur dalam transmisi sinyal digital: sinkronisasi, bandwidth, struktur deretan bit, kemampuan mendeteksi kesalahan, ketahanan terhadap gangguan, kompleksitas rendah.**

---

---

Setelah membicarakan kriteria dan syarat-syarat yang harus dipenuhi dalam pengkodean, sekarang mari kita bicarakan jenis-jenis pengkodean sinyal digital seperti telah disebutkan pada bagian awal dari bab ini satu-persatu.

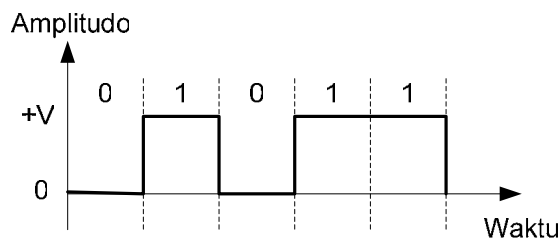
### **5.1.1. Line Coding**

Pada *line coding* selalu diasumsikan bahwa data yang berupa teks, gambar, suara, video telah tersimpan dalam memori komputer sebagai deretan bit. *Line coding* akan mengkonversi deretan bit tersebut menjadi sinyal digital untuk ditransmisikan. Pada sisi penerima, harus dilakukan proses sebaliknya yaitu konversi dari sinyal digital menjadi data digital.

*Pengkodean Unipolar: Non-Return-to-Zero*

Pengkodean data digital menjadi sinyal digital yang paling sederhana adalah *non-return-to-zero* (NRZ). NRZ juga disebut sebagai pengkodean digital *unipolar* karena sinyal yang dibangkitkan hanya menggunakan tegangan positif atau negatif saja. Perhatikan Gambar 5.2 untuk memahami bagaimana pengkodean digital dengan NRZ yang dibangkitkan dengan tegangan positif.

Pada modulasi NRZ, bit 0 direpresentasikan oleh sinyal dengan tegangan 0 volt, sedangkan bit 1 direpresentasikan oleh sinyal dengan tegangan +V volt. Karena 1 elemen sinyal hanya membawa 1 elemen data, maka  $m=1$ . Berdasarkan persamaan 5.1 kita dapat bahwa kecepatan sinyal rata-rata adalah  $S=R/2$  baud. Pengkodean ini disebut dengan NRZ karena sinyal tidak kembali ke 0 volt di tengah-tengah bit (bandingkan dengan modulasi manchester). Pengkodean NRZ dalam aplikasi nyata tidak digunakan karena jumlah daya yang dibutuhkan untuk membangkitkan 1 buah sinyal pada NRZ lebih besar daripada jenis pengkodean NRZ-L atau NRZ-I yang akan segera kita bahas.



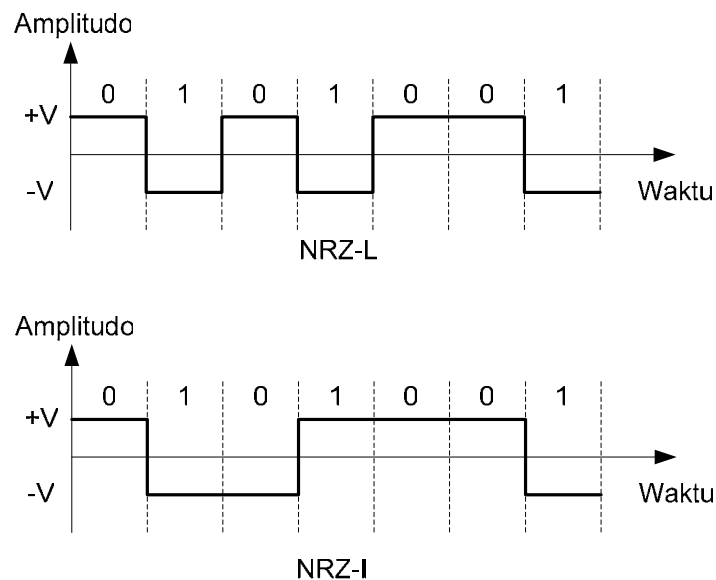
Gambar 5.2. Pengkodean digital NRZ

#### *Pengkodean Polar: NRZ-L, NRZ-I dan RZ*

NRZ-Level (NRZ-L) dan NRZ-Invert (NRZ-I) merupakan pengkodean digital *polar*. Disebut demikian karena keduanya menggunakan baik tegangan positif maupun tegangan negatif untuk membangkitkan sinyal digital. Pada NRZ-L bit 1 dan bit 0 direpresentasikan dengan level tegangan dari sinyal, sedangkan pada NRZ-I bit 1 dan bit 0 dibedakan oleh ada atau tidaknya perubahan level tegangan dari sinyal. Konversi data digital menjadi sinyal digital dengan menggunakan NRZ-L dan NRZ-I ditunjukkan dalam Gambar 5.3.

**Pada NRZ-L bit 1 dan bit 0 direpresentasikan dengan level tegangan dari sinyal, sedangkan pada NRZ-I bit 1 dan bit 0 dibedakan oleh ada atau tidaknya perubahan level tegangan dari sinyal.**

Seperti terlihat dalam Gambar 5.3, NRZ-L dan NRZ-I menggunakan tegangan positif dan negatif sebagai representasi bit. Pada NRZ-I tegangan dari sinyal akan berubah (berinversi) apabila bit berikutnya adalah bit 1. Sedangkan apabila bit berikutnya adalah bit 0, tidak ada perubahan sinyal. Dengan mengamati bentuk sinyal NRZ-L dan NRZ-I kita dapat melihat bahwa kedua modulasi polar ini masih akan mengalami apa yang disebut dengan *baseline wandering*. Pada NRZ-L *baseline wandering* akan terjadi apabila terdapat deretan panjang bit 1 atau bit 0, sedangkan pada NRZ-I *baseline wandering* hanya terjadi pada deretan panjang bit 0 saja. Dalam hal ini NRZ-I sedikit lebih baik daripada NRZ-L.

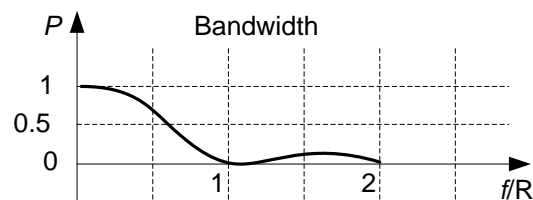


Gambar 5.3. Pengkodean digital dengan NRZ-L dan NRZ-I

Pada NRZ-L dan NRZ-I terlihat bahwa 1 bit elemen data direpresentasikan oleh 1 elemen sinyal *waveform*, sehingga  $m=1$ . Dengan demikian kecepatan sinyal rata-rata dari modulasi digital NRZ-L dan NRZ-I adalah  $S=R/2$  baud.

Bagaimana dengan bandwidth dari sinyal NRZ-L dan NRZ-I? Pertanyaan bagus. Karakteristik dari bandwidth dari kedua model modulasi ditunjukkan dalam Gambar 5.4.

Variabel  $P$  pada sumbu vertikal dari gambar adalah densitas dari daya (*Power density*), yaitu jumlah daya pada setiap 1 Hz dari bandwidth. Terlihat bahwa sebagian besar daya berada di sekitar frekuensi 0 Hz. Hal ini berarti terdapat komponen DC yang membawa energi besar sekali. Dari sini dapat disimpulkan bahwa energi yang dibawa oleh NRZ-L dan NRZ-I tidak tersebar merata di kedua tegangan positif dan tegangan negatif. Dengan kata lain, masalah *baseline wandering* tak terhindarkan oleh kedua jenis modulasi digital ini.



Gambar 5.4. Karakteristik bandwidth dari NRZ-L dan NRZ-I

*Contoh 5.2.*

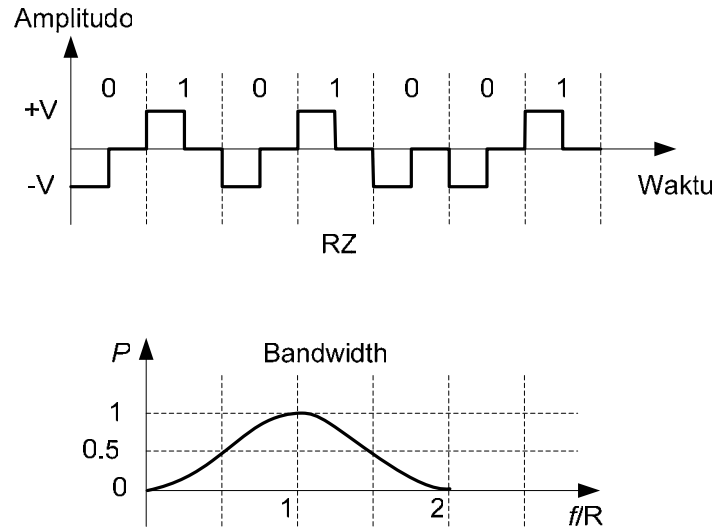
Komunikasi sistem jaringan menggunakan *fast-ethernet* dengan menggunakan pengkodean NRZ-I. Kecepatan pengiriman data adalah 100 Mbps. Berapa kecepatan pengiriman sinyal dari sistem tersebut?

*Jawaban:*

$R=100$  Mbps dan  $m=1$ . Maka  $S= R/2 = 100 \text{ Mbps}/2 = 50$  Mbaud.

Kekurangan dari NRZ-L dan NRZ-I diperbaiki oleh pengkodean digital *return-to-zero* (RZ). RZ menggunakan tiga level tegangan yaitu: tegangan positif, tegangan nol dan tegangan negatif seperti terlihat dalam Gambar 5.5. Dengan demikian persoalan munculnya komponen DC pada NRZ dapat dieliminasi oleh RZ.





Gambar 5.5. Pengkodean digital RZ dan karakteristik bandwidth RZ

Pengkodean RZ selalu mengembalikan sinyal ke tegangan nol pada saat sinyal telah mencapai separo dari durasi sinyal. Tetapi karena RZ menggunakan 2 sinyal elemen untuk merepresentasikan sebuah elemen data, hal ini berakibat pada kenaikan bandwidth sebanyak dua kali lipat dibandingkan dengan bandwidth yang digunakan oleh NRZ. Perhatikan bahwa nilai  $m=1/2$  dan kecepatan sinyal rata-rata adalah  $S=N$  baud.

Selain itu, karena RZ membutuhkan tiga level tegangan maka perangkat dengan kompleksitas tinggi dibutuhkan untuk membangkitkan sinyal RZ. Kelemahan-kelemahan sinyal RZ tersebut di atas menjadi alasan sehingga dalam praktek komunikasi data RZ tidak digunakan. Modulasi digital yang cukup efisien saat ini adalah *manchester* dan *differential mancheseter* yang akan dibicarakan pada bagian berikutnya.

#### *Pengkodean Dua-Fasa: Manchester dan Differential Manchester*

Pengkodean *Manchester* dan *Differential Manchester* dapat dilihat dalam Gambar 5.6. Pengkodean Manchester membagi durasi bit menjadi dua bagian. Level

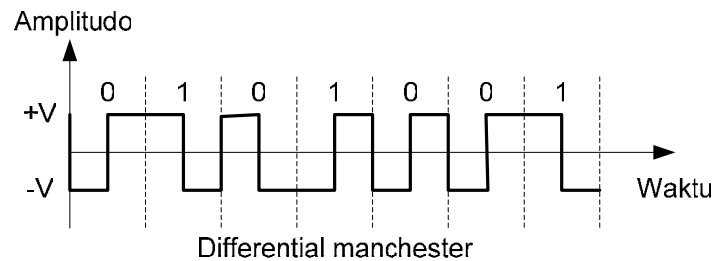
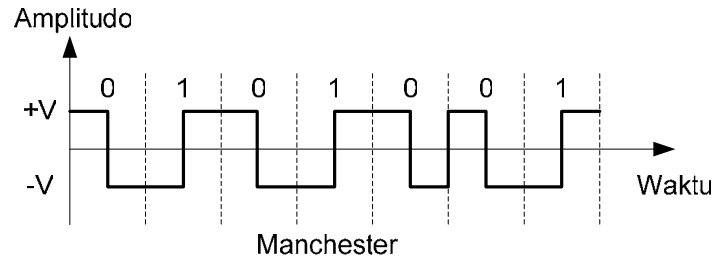
tegangan akan berubah saat separo dari durasi bit terlampaui. Sinyal yang merepresentasi bit 0 berubah dari tegangan positif (+V) menjadi tegangan negatif (-V), sedangkan bit 1 direpresentasikan dengan perubahan sinyal dari tegangan negatif (-V) menjadi tegangan positif (+V).

Pada pengkodean *differential manchester* selain terdapat perubahan sinyal pada separo dari durasi bit, juga terdapat inversi sinyal pada saat bit berikut adalah bit 0. Apabila bit berikut adalah bit 1, maka tidak ada inversi sinyal.

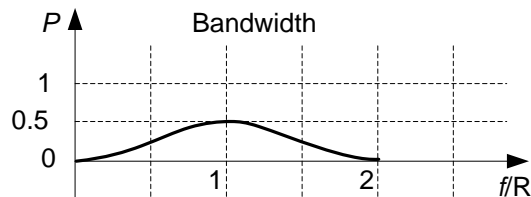
Seperti terlihat dalam gambar, pada pengkodean dua-fasa setiap 1 bit elemen data diwakili oleh 2 elemen sinyal, sehingga  $m=1/2$ . Dengan menggunakan persamaan 5.1, kecepatan sinyal rata-rata didapatkan  $S=R$  baud.

**Level tegangan pengkodean Manchester akan berubah saat separo dari durasi bit terlampaui. Pada *differential manchester* terdapat perubahan sinyal pada separo dari durasi bit dan inversi sinyal bila bit berikut bit 0. Apabila bit berikut adalah bit 1, maka tidak ada inversi sinyal.**

Dengan adanya transisi pada separo waktu dari durasi bit yang dapat diprediksikan sebelumnya, maka antara pengirim dan penerima terjadi proses sinkronisasi pada transisi tersebut. Keuntungan lain menggunakan pada pengkodean dua-fasa adalah tidak adanya komponen DC, sehingga *baseline wandering* tidak mungkin terjadi pada pengkodean ini. Satu-satunya kelemahan pada pengkodean dua-fasa adalah kebutuhan bandwidth transmisi yang dua kali lebih besar daripada pengkodean NRZ. Karakteristik bandwidth dari pengkodean dua-fasa dapat dilihat dalam Gambar 5.7.



Gambar 5.6. Pengkodean digital dengan *Manchester* dan *Differential Manchester*



Gambar 5.7. Karakteristik bandwidth dari pengkodean dua-fasa.

*Contoh 5.3.*

Jaringan Local Area Network (LAN) menggunakan Ethernet dengan kecepatan pengiriman data 10 Mbps. Apabila pengkodean Manchester digunakan, berapa kecepatan pengiriman sinyal jaringan Ethernet?

*Jawaban:*

Pengkodean Manchester menggunakan  $m=1/2$ . Maka kecepatan sinyal  $S=R=10$  Mbaud.

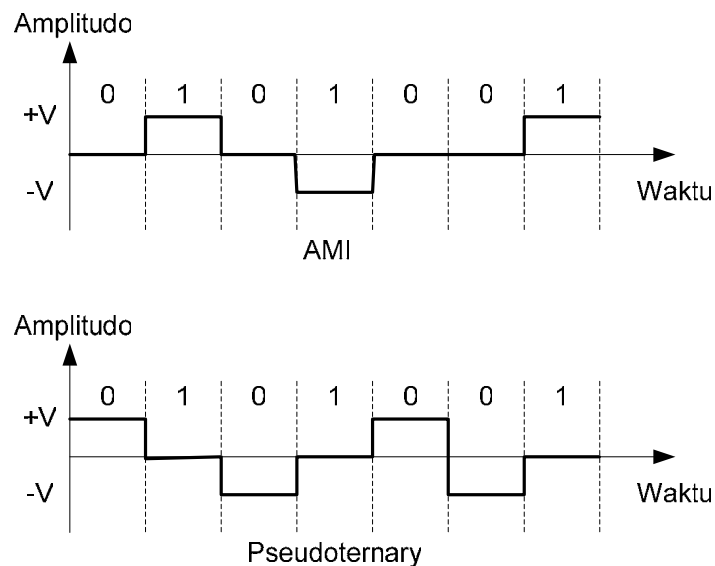
*Pengkodean Bipolar: AMI dan Pseudoternary*

Pada bagian ini kita akan melihat dua macam pengkodean bipolar yang dikenal dengan nama *Alternate Mark Inversion* (AMI) dan *Pseudoternary*. Pengkodean

bipolar dibuat untuk mengeliminasi kekurangan-kekurangan yang ada pada NRZ. Pada pengkodean AMI, elemen data dengan bit 1 direpresentasikan oleh sinyal yang beriversi bolak balik dari tegangan positif ke tegangan negatif atau sebaliknya dari tegangan negatif ke tegangan positif. Sedangkan elemen data dengan bit 0 direpresentasikan oleh tegangan 0 volt.

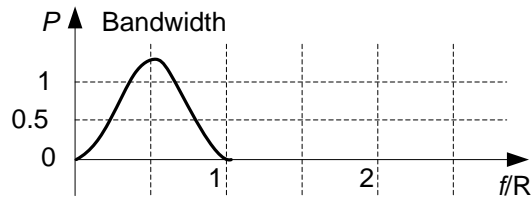
Pada pengkodean *pseudoternary*, elemen data dengan bit 0 direpresentasikan oleh sinyal yang beriversi bolak balik dari tegangan positif ke tegangan negatif atau sebaliknya dari tegangan negatif ke tegangan positif. Sedangkan elemen data dengan bit 1 direpresentasikan oleh tegangan 0 volt.

Kedua jenis pengkodean bipolar ini direpresentasikan dalam Gambar 5.8. Seperti terlihat dalam gambar, pada pengkodean bipolar ini 1 elemen data direpresentasikan oleh 1 elemen sinyal, sehingga didapatkan nilai  $m=1$ . Dengan menggunakan persamaan 5.1 didapatkan bahwa kecepatan sinyal rata-rata adalah  $S=R/2$  baud. Dengan memperhatikan pada Gambar 5.9, kita tahu bahwa konsentrasi sebagian energi dari pengkodean bipolar berada pada frekuensi  $R/2$ .



Gambar 5.7. Pengkodean digital dengan AMI dan Pseudoternary

Keuntungan menggunakan menggunakan pengkodean bipolar adalah: pertama, tidak memiliki komponen DC, dan kedua, membutuhkan bandwidth dua kali lebih kecil daripada pengkodean dua-fasa yang telah kita bicarakan sebelumnya.



Gambar 5.8. Karakteristik bandwidth dari pengkodean bipolar

**Keuntungan menggunakan menggunakan pengkodean bipolar adalah: pertama, tidak memiliki komponen DC, dan kedua, membutuhkan bandwidth dua kali lebih kecil daripada pengkodean dua-fasa.**

*Contoh 5.4.*

Pada sistem komunikasi dengan memanfaatkan jalur komunikasi T1 menggunakan pengkodean digital AMI untuk pengiriman data dengan kecepatan 1,544 Mbps. Berapa kecepatan pengiriman sinyal pada T1 tersebut?

*Jawaban:*

Pada pengkodean AMI diketahui  $m=1$ . Maka kecepatan pengiriman sinyal  $S=R/2 = 0,772$  Mbaud.

*Pengkodean Multilevel: 2B1Q, 8B6T dan 4D-PAM5*

Tujuan dari pengkodean multilevel adalah meningkatkan kecepatan data tetapi pada saat yang sama menurunkan kecepatan sinyal (menurunkan bandwidth). Untuk mencapai tujuan ini pengkodean multilevel melakukan pengkodean dari  $p$  elemen data menjadi  $q$  elemen sinyal.

Elemen data terdiri atas bit 0 dan bit 1, sehingga jumlah kombinasi pola bit yang mungkin dibuat adalah  $2^p$  pola. Sedangkan elemen sinyal dengan level  $L$  akan menghasilkan kombinasi pola sinyal sebanyak  $L^q$ . Karena itu apabila kita buat agar  $2^p=L^q$ , maka setiap pola data akan dapat direpresentasikan tepat pada setiap pola sinyal. Namun dalam aplikasi dibutuhkan agar  $2^p \leq L^q$  sehingga tidak semua pola sinyal merupakan representasi dari pola data. Pola sinyal selebihnya dapat digunakan untuk sinkronisasi dan pedeteksi kesalahan.

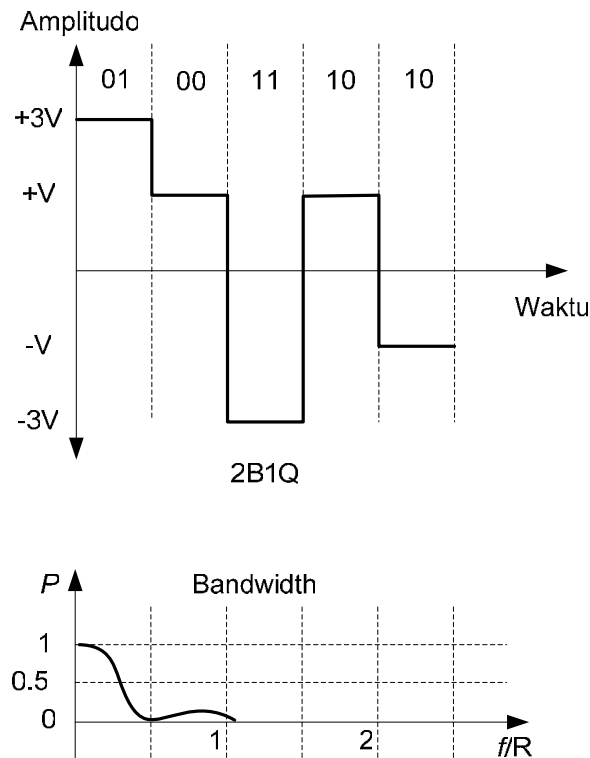
Model pengkodean multilevel yang saat ini digunakan oleh Digital Subscriber Line (DSL) adalah *two binary, one quaternary* (2B1Q). 2B1Q berarti setiap 2 bit data dikodekan ke dalam 1 elemen sinyal yang memiliki 4 level tegangan. Sehingga nilai  $m=2$  dan kecepatan sinyal rata-rata adalah  $S=R/4$  baud. Dengan demikian nilai  $p=2$ ,  $q=1$  dan  $L=4$ . Pengkodean 2B1Q mengikuti aturan seperti yang ada dalam Tabel 5.1.

Tabel 5.1. Pengkodean 2B1Q

<i>Bit</i>	<i>Level</i>	<i>Level</i>
	<i>Sebelumnya</i>	<i>Sebelumnya</i>
	<b>: Positif</b>	<b>: Negatif</b>
	<i>Level</i>	<i>Level</i>
	<i>Berikutnya</i>	<i>Berikutnya</i>
00	+1	-1
01	+3	-3
10	-1	+1
11	-3	+3

Contoh pengkodean 2B1Q dapat dilihat dalam Gambar 5.9. Dengan menggunakan 2B1Q data dikirimkan dengan kecepatan dua kali lipat kecepatan yang dapat dicapai oleh NRZ-L. Namun satu hal perlu diperhatikan, pada NRZ-L penerima hanya mendeteksi dua level tegangan, sedangkan pada 2B1Q penerima harus

mampu mendeteksi sampai empat level tegangan. Keuntungan lain menggunakan 2B1Q adalah kebutuhan bandwidth yang kecil untuk mentransmisikan sinyal.



Gambar 5.9. Pengkodean digital dan karakteristik bandwidth 2B1Q

**Dalam aplikasi dibutuhkan agar  $2^p \leq L^q$  sehingga tidak semua pola sinyal merupakan representasi dari pola data. Pola sinyal selebihnya dapat digunakan untuk sinkronisasi dan pedeteksi kesalahan.**

*Contoh 5.5.*

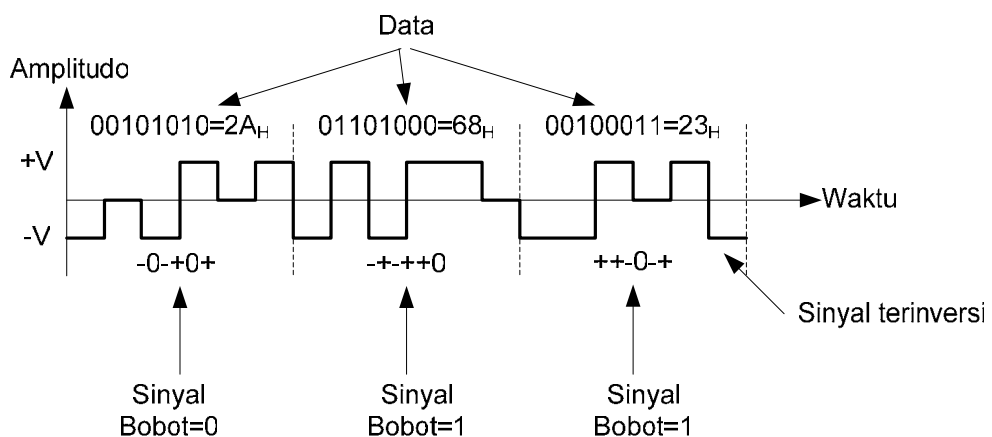
Sistem komunikasi *Symmetric Digital Subscriber Line* (SDSL) menggunakan pengkodean 2B1Q dengan kecepatan pengiriman data 768 kbps. Berapa kecepatan pengiriman sinyal SDSL?

*Jawaban:*

Pada pengkodean 2B1Q diketahui bahwa  $m=2$ . Maka kecepatan pengiriman sinyal SDSL  $S=R/4=768 \text{ kbps}/4 = 192 \text{ kbaud}$ .

Model pengkodean multilevel yang lain disebut dengan nama 8B6T. Dinamakan demikian karena pengkodean ini melakukan konversi dari 8 bit bilangan biner data menjadi 6 pola sinyal dengan tiga (ternary) level tegangan (positif, negatif dan nol). Sehingga kita dapat menghitung bahwa ada sebanyak  $2^8 = 256$  pola data dan ada sebanyak  $3^6 = 478$  pola sinyal. Karena itu akan terdapat sisa pola sinyal yang tidak digunakan merepresentasikan pola data sebanyak 222 pola. Sisa pola sinyal yang tidak digunakan ini dipakai untuk sinkronisasi dan pendeteksi kesalahan. Dengan demikian pada pengkodean ini nilai  $m=8/6$ , dan kecepatan sinyal rata-rata adalah  $S=3N/8$  baud. Proses pengkodean ini menggunakan tabel seperti ditunjukkan dalam Lampiran 1. Contoh pengkodean data dengan 8B6T ditunjukkan dalam Gambar 5.10.

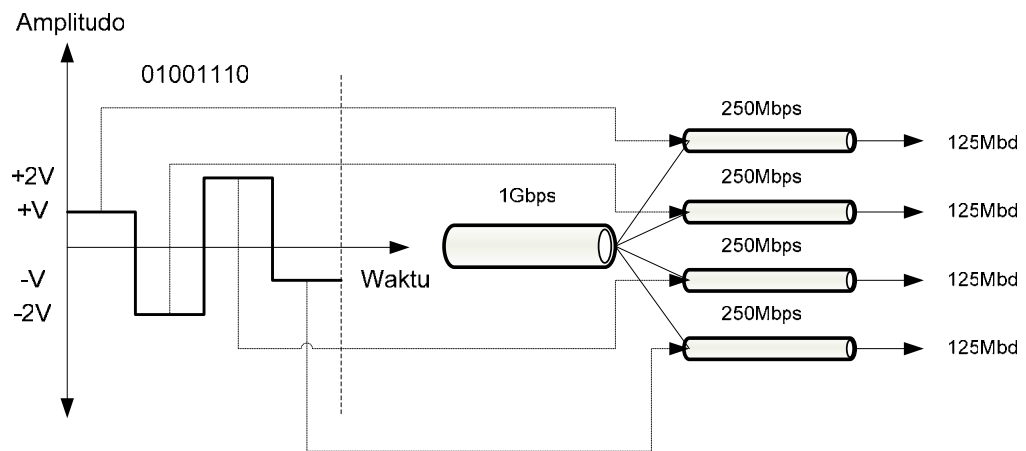
Seperti terlihat dalam gambar, salah satu penggunaan pola sinyal yang tidak memiliki representasi pola data adalah untuk mengidentifikasi bobot dari setiap sinyal. Perhatikan tabel dalam Lampiran 1, setiap sinyal memiliki representasi bobot 0 atau 1. Karena itu apabila terdapat dua buah sinyal dengan bobot yang sama berurutan, maka akan dilakukan inversi terhadap sinyal terakhir seperti terlihat dalam Gambar 5.10.



Gambar 5.10. Pengkodean digital 8B6T



Teknologi terbaru dalam bidang komunikasi jaringan adalah teknologi Giga Bit Ethernet. Gigabit Ethernet menggunakan model pengkodean multilevel dengan nama 4D-PAM5. Pengkodean ini menggunakan 5 level sinyal yang ditransmisikan melalui 4 buah kabel pada saat yang sama. Sinyal dengan level tegangan 0 volt tidak digunakan untuk mengirimkan data melainkan digunakan untuk deteksi kesalahan. Sehingga 4D-PAM5 mengkodekan 8 bit elemen data menjadi 4 level sinyal dan masing-masing sinyal ditransmisikan melalui 4 buah kabel seperti terlihat dalam Gambar 5.11. Dengan demikian nilai  $m=4$  dan kecepatan sinyal rata-rata adalah  $S=N/8$ . Jika dibandingkan dengan metode pengkodean yang lain, 4D-PAM5 merupakan jenis pengkodean dengan kecepatan sinyal rata-rata paling rendah.



Gambar 5.11. Pengkodean digital 4D-PAM5

#### *Pengkodean Multiline Transmission: MLT-3*

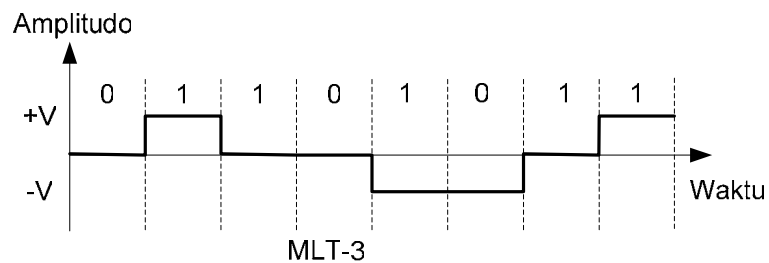
*Multiline transmission, three level (MLT-3)* menggunakan 3 level tegangan sinyal dan 3 aturan transisi untuk berpindah dari satu level tegangan ke level tegangan yang lain. Tiga aturan transisi tersebut dapat dirumuskan sebagai berikut:

- a. Apabila bit berikutnya adalah 0, maka tidak ada transisi level sinyal.
- b. Apabila bit berikutnya adalah 1 dan level sinyal saat ini tidak 0, maka level sinyal berikutnya adalah 0.

- c. Apabila bit berikutnya adalah 1 dan level sinyal saat ini adalah 0, maka level sinyal berikutnya adalah kebalikan dari level sinyal tidak 0 yang terakhir.

***Multiline transmission, three level (MLT-3) menggunakan 3 level tegangan sinyal dan 3 aturan transisi untuk berpindah dari satu level tegangan ke level tegangan yang lain.***

Contoh sinyal dengan pengkodean MLT-3 ditunjukkan dalam Gambar 5.12.



Gambar 5.12. Pengkodean digital dengan MLT-3

Secara sepintas pengkodean MLT-3 mirip dengan pengkodean AMI, yaitu: 1 elemen data dikonversikan menjadi 1 elemen sinyal. Namun karena adanya aturan transisi di atas, dalam praktek pengkodean MLT-3 dapat mencapai kecepatan sinyal rata-rata  $S=R/4$ , lebih kecil daripada AMI. Karena itu MLT-3 dalam aplikasi nyata digunakan untuk Fast Ethernet dengan kecepatan 100 Mbps melalui kawat tembaga, karena kawat tembaga tidak dapat mendukung bandwidth dari sinyal lebih dari 32 MHz.

*Contoh 5.6.*

Sebagian besar jaringan LAN pada saat ini menggunakan teknologi Fast Ethernet dengan kecepatan pengiriman data sebesar 100 Mbps. Apabila jenis pengkodean MLT-3 digunakan untuk pengiriman data, berapa kecepatan sinyal Fast Ethernet?

*Jawaban:*

Pada pengkodean MLT-3 diketahui bahwa nilai  $m=2$ . Maka kecepatan pengiriman sinyal  $S=R/4 = 100 \text{ Mbps}/4 = 25 \text{ Mbaud}$ .

Ringkasan dari pembicaraan tentang pengkodean digital beserta lebar bandwidth dan aplikasinya dapat dilihat dalam Tabel 5.2. Seperti terlihat dalam tabel, sebagian besar aplikasi pengkodean digital diterapkan untuk komunikasi jaringan *Local Area Network* (LAN).

Tabel 5.2. Ringkasan berbagai macam pengkodean digital

<i>Kategori</i>	<i>Kode</i>	<i>Bandwidth</i>	<i>Aplikasi</i>
Unipolar	NRZ	$R/2$	-
Polar	NRZ-L	$R/2$	Komunikasi serial RS-232
	NRZ-I	$R/2$	Fast Ethernet dengan Serat optik: 100Base-FX
Dua-fasa	Manchester	$R$	LAN Ethernet bus: 10Base5, 10Base2, 10Base-T, 10Base-F.
	Diff-Manchester	$R$	LAN dengan Token Ring
Bipolar	AMI	$R/2$	ISDN
	Pseudoternary	$R/2$	ISDN
Multilevel	2B1Q	$R/4$	DSL

	8B6T	3R /4	Fast-Ethernet 100Base-T4
	4D- PAM5	R /8	Gigabit Ethernet 1000Base-T4
Multil ne	MLT-3	R /4	Fast Ethernet 100Base-TX

Terkait dengan penamaan standar, beberapa keterangan perlu ditambahkan di sini. LAN yang menggunakan teknologi Ethernet distandarkan dengan beberapa nama dan kecepatan pengiriman data. Ethernet memiliki kecepatan 10 Mbps, distandarkan dengan nama 10Base2 dan 10Base5 untuk komunikasi menggunakan media kabel koaxial. 10Base-T adalah Ethernet dengan media twisted-pair, dan 10Base-F adalah Ethernet dengan media serat optik. Pengembangan berikutnya meningkatkan kecepatan Ethernet menjadi 100 Mbps disebut dengan nama Fast-Ethernet. Fast-Ethernet dengan standar 100Base-TX menggunakan media dua pasang kabel twisted pair (UTP Cat. 5) sedangkan 100Base-T4 menggunakan media 4 pasang kabel twisted pair (UTP Cat. 5E, Cat. 6) dan 100Base-FX menggunakan media serat optik. Versi terakhir dari Ethernet adalah Gigabit Ethernet dengan kecepatan 1 Gbps. Standar 1000Base-T4 adalah Gigabit Ethernet menggunakan media twisted pair 4 pasang, sedangkan 1000Base-SX, 1000Base-LX dan 1000Base-CX adalah Gigabit Ethernet dengan menggunakan media serat optik.

### 5.1.2. Pengkodean Blok

Pengkodean dengan menggunakan blok secara teknis mengkodekan sebuah blok data dengan panjang  $p$  bit menjadi blok data dengan panjang  $q$  bit. Pengkodean blok disimbolkan dengan menggunakan tanda '/' untuk membedakan dengan pengkodean multilevel yang telah kita bicarakan dalam sub-bab 5.1.1. Sebagai contoh pengkodean 4 biner menjadi 5 biner dituliskan dengan simbol 4B/5B. Proses pembentukan pengkodean blok terdiri atas tiga tahap, yaitu:

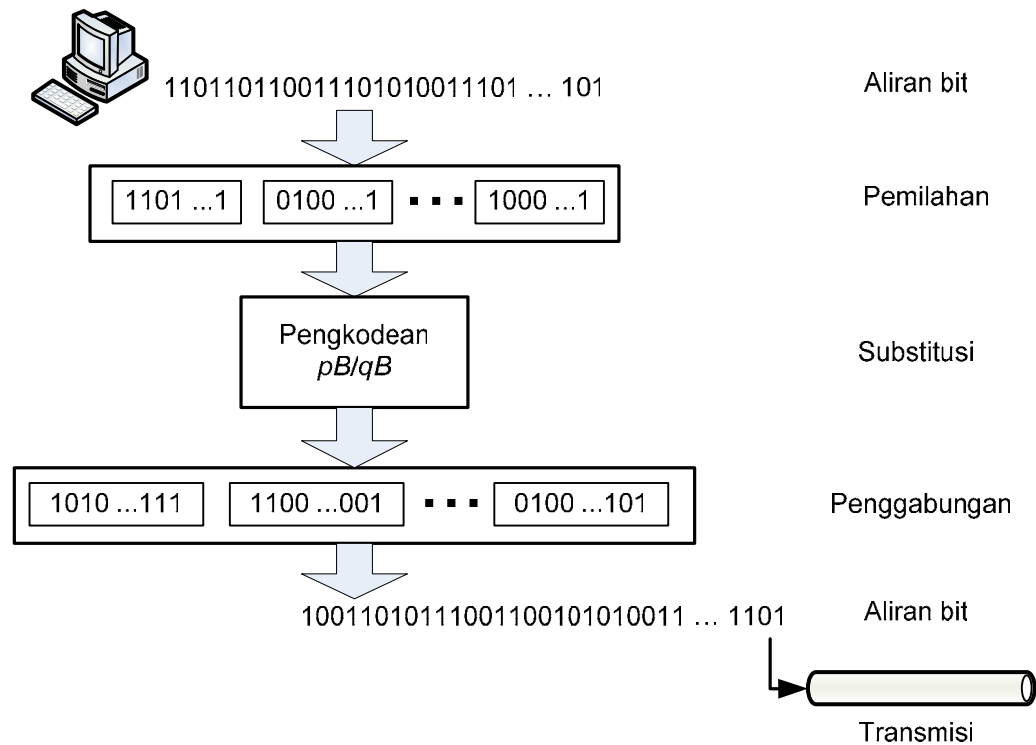
- a. Tahap pemilahan aliran bit data menjadi blok.

- b. Tahap substitusi blok data yang telah dibuat menjadi blok data baru dengan ukuran blok lebih besar.
- c. Tahap penggabungan blok data baru menjadi aliran bit data.

Ilustrasi ketiga proses pengkodean blok dapat dilihat dalam Gambar 5.13.

**Proses pembentukan pengkodean blok terdiri atas tiga tahap:  
pemilahan aliran bit, substitusi blok, penggabungan blok.**

Dalam proses substitusi dibutuhkan agar blok data baru memiliki ukuran bit lebih besar daripada blok data sebelum substitusi. Hal ini dikarenakan bit data yang akan ditransmisikan harus memiliki kemampuan untuk melakukan sinkronisasi dan deteksi kesalahan di dalam dirinya. Sebagai contoh, pada pengkodean blok 4B/5B, blok data lama berukuran 4 bit sedang blok data baru berukuran 5 bit. Karena blok data lama hanya memiliki 16 pola sedangkan blok data baru memiliki variasi sebanyak 32 pola, maka hanya dibutuhkan sebanyak 16 pola dari blok data baru untuk substitusi. Sisa pola data digunakan untuk sinkronisasi dan deteksi kesalahan.

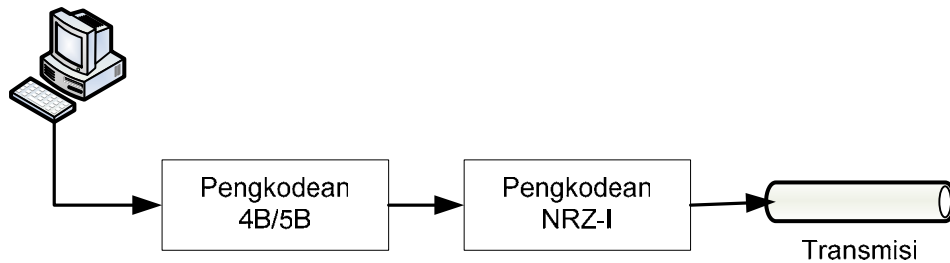


Gambar 5.13. Ilustrasi proses pengkodean blok

#### *Pengkodean Blok 4B/5B*

Pengkodean blok 4B/5B melakukan konversi blok data yang terdiri atas 4 bit bilangan biner menjadi blok data berukuran 5 bit bilangan biner. Dalam praktek, pengkodean 4B/5B digunakan bersama-sama dengan pengkodean NRZ-I seperti terlihat dalam Gambar 5.14. Sebelum ditransmisikan data dikodekan terlebih dahulu dengan menggunakan pengkodean blok 4B/5B, selanjutnya data dikodekan menjadi sinyal dengan menggunakan NRZ-I. Seperti kita lihat dalam sub-bab 5.1.1 bahwa NRZ-I memiliki kelemahan apabila terdapat deretan data bit 0 yang cukup panjang, namun kelemahan tersebut telah dapat dieliminasi dengan adanya pemilahan aliran bit data yang panjang menjadi blok-blok data berukuran kecil.

Pengkodean NRZ-I bukan satu-satunya jenis pengkodean yang dapat digunakan bersama-sama dengan pengkodean blok 4B/5B. Apabila efek dari komponen DC masih belum dapat ditolerir, maka pengkodean blok 4B/5B juga dapat digabungkan dengan pengkodean dua-fasa atau pengkodean bipolar.



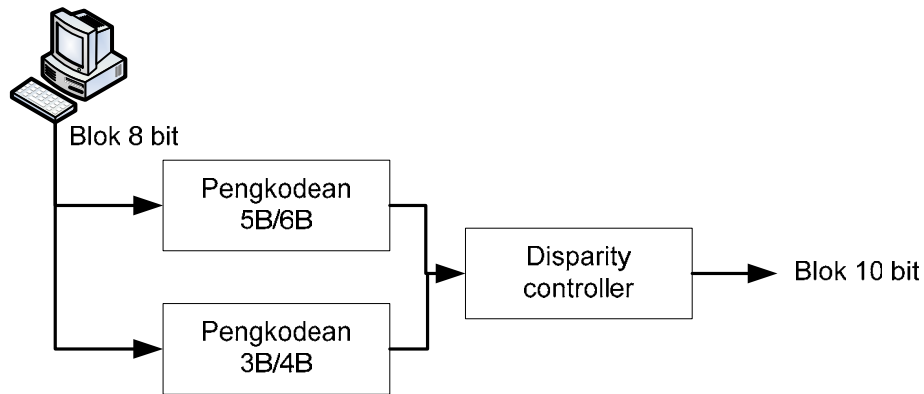
Gambar 5.14. Pengkodean digital dengan 4B/5B dan NRZ-I

Tabel konversi dari berbagai pola 4 bit data menjadi 5 bit data ditunjukkan dalam Lampiran 2. Seperti terlihat dalam tabel, hasil pengkodean 5 bit menunjukkan bahwa tidak ada lagi blok data yang diawali oleh lebih dari satu bit 0. Juga tidak ada lagi blok data yang diakhiri oleh bit 0 dengan jumlah bit lebih dari dua. Sehingga dengan menggunakan pengkodean 4B/5B tidak akan ada lagi sebanyak tiga elemen bit 0 berada dalam posisi berjajar.

#### *Pengkodean blok 8B/10B*

Pengkodean blok 8B/10B mengkodekan 8 bit data biner menjadi 10 bit data biner. Keuntungan menggunakan pengkodean blok 8B/10B adalah kemampuan deteksi kesalahan yang lebih baik daripada pengkodean blok 4B/5B. Namun dalam dunia nyata untuk dapat menghasilkan pengkodean blok 8B/10B dilakukan penggabungan antara pengkodean blok 5B/6B dengan pengkodean blok 3B/4B seperti terlihat dalam Gambar 5.15. Tujuan dari penggabungan dua jenis pengkodean ini adalah semata-mata untuk menyederhanakan tabel konversi.

Dalam proses pengkodean 8B/10B, 5 bit pertama (*the most significant bit*) dari 8 bit data yang akan dikodekan diinputkan ke dalam pengkode digital 5B/6B, sedangkan 3 bit terakhir diinputkan ke dalam pengkode digital 3B/4B. *Disparity controller* digunakan untuk mendeteksi apabila terdapat elemen data bit 0 atau bit 1 berjajar dalam jumlah banyak.



Gambar 5.15. Skema pengkodean digital 8B/10B

### 5.1.3. Scrambling

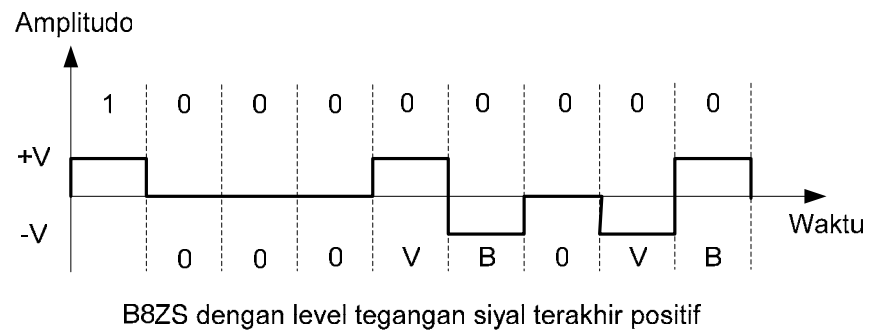
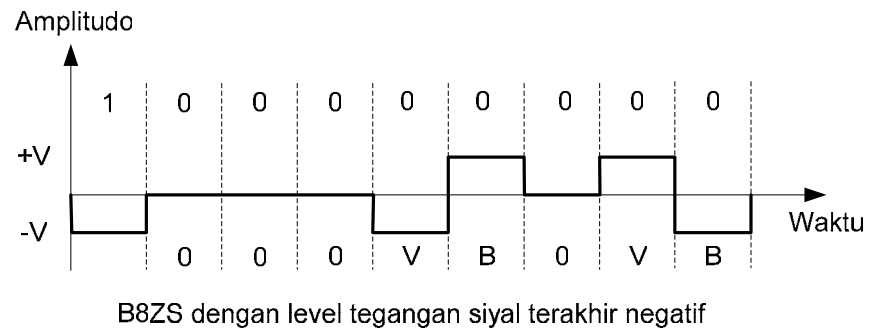
Seperti telah kita diskusikan dalam sub-bab 5.1.1 bahwa pengkodean digital AMI mengandung masalah tersendiri apabila terdapat level tegangan nol berderet panjang. Kelemahan tersebut dapat diperbaiki dengan menggunakan teknik pengkodean *scrambling*. Tujuan dari pengkodean scrambling adalah melakukan substitusi dengan aturan tertentu apabila dideteksi sejumlah level tegangan nol berderet panjang. Pada dasarnya teknik scrambling adalah pengkodean AMI dengan modifikasi apabila dideteksi level tegangan nol berderet panjang.

**Tujuan dari pengkodean scrambling adalah melakukan substitusi dengan aturan tertentu apabila dideteksi sejumlah level tegangan nol berderet panjang.**

#### *Pengkodean scrambling B8ZS*

Pengkodean B8ZS adalah *Bipolar with 8-Zero Substitution*. Dengan menggunakan pengkodean ini apabila terdapat 8 level tegangan nol berurutan, maka kedelapan level tegangan tersebut disubstitusi oleh level tegangan 000VB0VB. V adalah singkatan dari *violation* dan B adalah singkatan dari *bipolar*. Level tegangan dengan nilai V adalah level tegangan yang memiliki level tegangan inversi dari level tegangan yang seharusnya (inversi dari aturan AMI), sedangkan level tegangan B adalah level tegangan yang mengikuti aturan AMI.





Gambar 5.16. Pengkodean scrambling dengan B8ZS

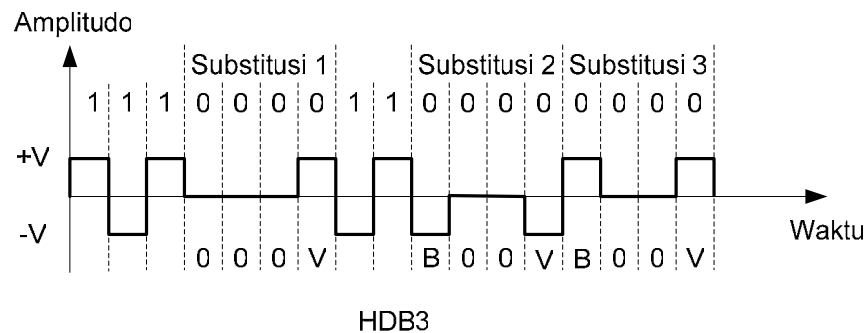
Sebagai contoh perhatikan contoh pengkodean B8ZS dalam Gambar 5.16. Dalam Gambar 5.16 terdapat dua macam kondisi yang mungkin terjadi. Kondisi pertama adalah saat bit 1 direpresentasikan oleh tegangan negatif dalam pengkodean AMI. Karena itu level tegangan V berinversi dari level tegangan yang seharusnya positif (menurut aturan AMI) menjadi negatif. Sedangkan nilai level tegangan B menjadi positif karena level tegangan tidak nol terakhir adalah level tegangan negatif. Lihat Gambar 5.16 atas. Gambar 5.16 bagian bawah mengilustrasikan kondisi kedua, yaitu saat bit 1 direpresentasikan oleh tegangan positif dalam pengkodean AMI. Pembaca dapat menelusuri sendiri hasil pengkodean dengan menggunakan pengkodean scrambling B8ZS.

*Pengkodean scrambling HDB3*

Pengkodean *High-Density Bipolar 3-Zero* (HDB3) mirip dengan pengkodean B8ZS yang telah di ulas pada bagian sebelumnya. HDB3 akan melakukan substitusi dengan level tegangan 000V atau B00V apabila menjumpai empat level tegangan nol berurutan. Aturan substitusi adalah sebagai berikut:

1. Jika jumlah sinyal tidak nol setelah substitusi terakhir adalah ganjil, maka substitusi dilakukan dengan menggunakan level tegangan 000V.
2. Jika jumlah sinyal tidak nol setelah substitusi terakhir adalah genap, maka substitusi dilakukan dengan menggunakan level tegangan B00V.

Ilustrasi pengkodean sinyal dengan HDB3 dapat dilihat dalam Gambar 5.17.



Gambar 5.17. Pengkodean scrambling HDB3

Seperti terlihat dalam gambar, sebelum substitusi 1 terjadi terdapat 3 level tegangan tidak nol. Menurut aturan HDB3 maka substitusi 1 dilakukan dengan menggunakan level tegangan 000V. Setelah substitusi 1 terjadi, maka level tegangan berikutnya akan mengikuti aturan pada AMI. Selanjutnya sebelum substitusi 2 terjadi terdapat 2 level tegangan tidak nol (setelah substitusi 1), maka menurut aturan HDB3, substitusi dilakukan dengan menggunakan level tegangan B00V. Sebelum substitusi 3 terjadi, tidak ada sinyal dengan level tegangan tidak nol. Bila hal ini terjadi maka jumlah level tegangan sebelum substitusi dianggap sebagai genap. Karena itu substitusi 3 menggunakan level tegangan B000V.

## 5.2. Konversi Sinyal Analog Menjadi Data Digital

Di sekitar kita, dalam kehidupan sehari-hari, sebenarnya lebih banyak sinyal yang direpresentasikan dalam bentuk analog daripada sinyal dalam bentuk digital. Misalnya, suara, cahaya, suhu, bau dan sebagainya. Namun sinyal-sinyal analog semacam itu akan lebih mudah disimpan, diolah, direproduksi kembali apabila disimpan dalam bentuk data digital. Sebagai contoh, *Compact Disc* yang dijual di pasaran dapat menampung sejumlah besar lagu adalah hasil konversi sinyal suara analog ke dalam bentuk digital. Film-film yang dapat dinikmati melalui DVD juga merupakan hasil dari rekayasa digital. Dan masih banyak lagi manfaat yang dapat kita rasakan saat ini dengan adanya teknologi digital.

---

**Sinyal analog lebih mudah disimpan, diolah, direproduksi kembali apabila disimpan dalam bentuk data digital.**

---

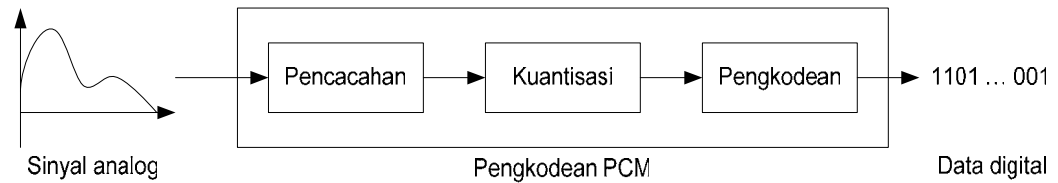
Untuk memperoleh data digital dibutuhkan suatu proses untuk mengubah sinyal analog menjadi data digital. Ada beberapa metode yang dapat digunakan. Namun dalam sub-bab ini kita hanya akan membahas dua metode yang paling banyak digunakan, yaitu *Pulse Code Modulation (PCM)* dan Modulasi Delta (Delta modulation).

### **5.2.1. Pulse Code Modulation (PCM)**

PCM merubah sinyal analog menjadi data digital melalui proses awal yang disebut dengan *sampling*. Sampling adalah proses mencacah sinyal analog menjadi potongan-potongan sinyal dengan amplitudo sesuai dengan sinyal asli. Setelah didapatkan sinyal hasil sampling, sinyal tersebut selanjutnya dikuantisasi. Kuantisasi adalah proses pembulatan amplitudo sinyal terkuantisasi ke bilangan integer terdekat. Proses terakhir adalah melakukan pengkodean digital terhadap kode hasil kuantisasi. Jadi dapat disimpulkan bahwa PCM menggunakan tiga langkah utama di dalam mengubah sinyal analog menjadi data digital, yaitu:

1. Proses pencacahan (*sampling*),
2. Proses kuantisasi, dan
3. Pengkodean data digital.

Gambar 5.18 memberikan ilustrasi seluruh proses mengubah sinyal analog menjadi data digital dengan menggunakan PCM.



Gambar 5.18. Pengkodean PCM

### *Proses pencacahan*

Proses pencacahan dilakukan dengan mencacah sinyal analog dalam periode waktu tertentu disebut dengan periode pencacahan ( $T_s$ ). Kebalikan dari periode pencacahan adalah frekuensi pencacahan ( $f_s$ ), yaitu  $f_s=1/T_s$ .

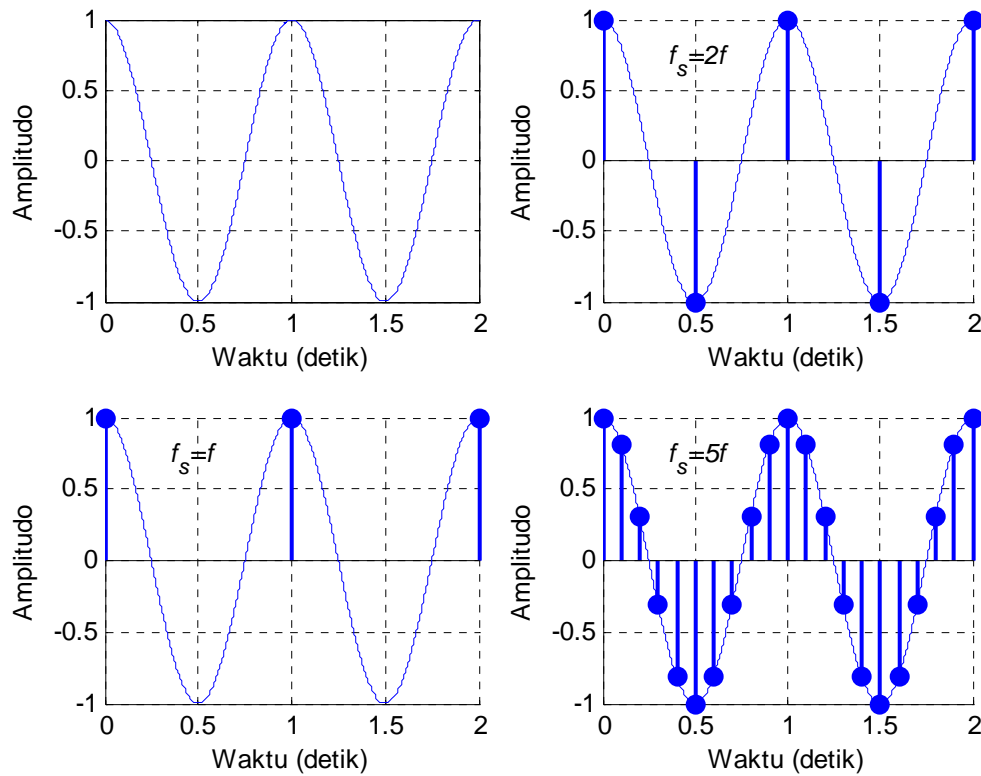
Semakin tinggi frekuensi pencacahan, atau semakin kecil periode pencacahan maka sinyal hasil cacahan akan semakin menyerupai sinyal analog asli. Sinyal hasil cacahan seringkali disebut juga istilah sinyal Pulse Amplitude Modulation (PAM). Namun semakin tinggi frekuensi pencacahan membawa konsekuensi pada harga keseluruhan dalam proses pencacahan semakin mahal.

Sebaliknya, menggunakan frekuensi pencacahan rendah akan menurunkan harga proses pencacahan tetapi mengandung konsekuensi pada representasi sinyal PAM yang kurang dapat mewakili sinyal analog asli. Karena itu secara natural akan muncul pertanyaan, berapa frekuensi terendah yang dapat digunakan agar hasil pengkodean digital nantinya dapat dikembalikan ke bentuk asli dari sinyal analog?

Pertanyaan ini dijawab oleh Teorema Nyquist yang berbunyi demikian: *frekuensi pencacah harus minimal dua kali frekuensi tertinggi (bukan bandwidth) yang dikandung oleh sinyal asli*. Dalam Bab 3 sudah kita bicarakan representasi domain frekuensi dari suatu sinyal. Dengan menggunakan representasi domain frekuensi

tersebut kita dapat melihat frekuensi tertinggi yang dikandung oleh suatu sinyal. Karena itu kita dapat menarik acuan umum bahwa proses pencacahan hanya dapat dilakukan apabila sinyal memiliki bandwidth terbatas (*band-limited*). Apabila bandwidth dari suatu sinyal tak terbatas, maka pencacahan tidak dapat dilakukan. Dengan kata lain, akan dibutuhkan frekuensi tak terhingga untuk mencacah sinyal dengan bandwidth tak terbatas.

**Teorema Nyquist berbunyi demikian: *frekuensi pencacah harus minimal dua kali frekuensi tertinggi(bukan bandwidth) yang dikandung oleh sinyal asli.***



Gambar 5.19. Pencacahan dengan berbagai frekuensi pencacah

Efek dari variasi frekuensi pencacah ditunjukkan dalam Gambar 5.19. Gambar sebelah kanan atas adalah contoh pencacahan sinyal dengan menggunakan frekuensi pencacah sama dengan frekuensi yang disyaratkan oleh Nyquist, yaitu

$f_s=2f_{max}$ . Gambar bawah sebelah kiri adalah pencacahan dengan frekuensi pencacah kurang dari syarat Nyquist. Karena jumlah sinyal pencacah kurang dari syarat minimal, maka sinyal pencacah tidak akan dapat merepresentasikan sinyal analog asli. Sedangkan pada gambar terakhir terlihat bahwa frekuensi pencacah jauh di atas syarat Nyquist, karena itu sinyal pencacah dapat merepresentasikan sinyal analog asli dengan sangat baik.

#### *Contoh 5.7.*

Dalam Gambar 5.19, sinyal memiliki frekuensi 1 Hz. Tentukan frekuensi pencacah yang dibutuhkan untuk mencacah sinyal tersebut.

#### *Jawaban:*

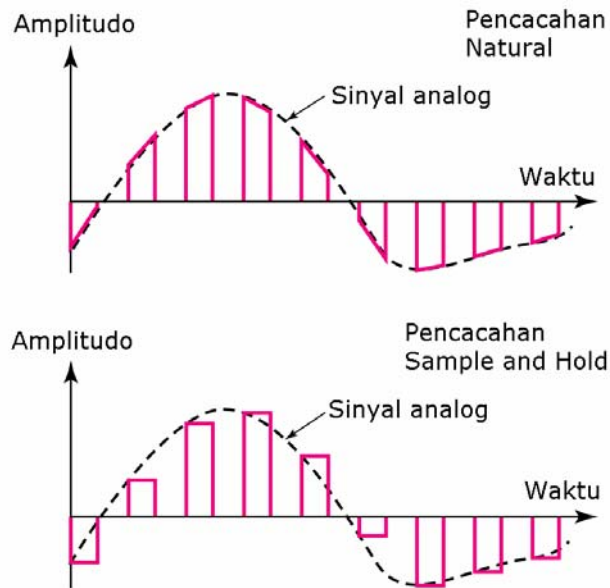
Sesuai dengan kriteria Nyquist, maka frekuensi pencacah minimal adalah  $f_s=2f_{max}$ , maka nilai  $f_s=2$  Hz. Frekuensi pencacah tersebut adalah pencacah minimal. Apabila frekuensi pencacah ditingkatkan menjadi 5 kali frekuensi maksimal, maka  $f_s=5$  Hz. Seperti terlihat dalam Gambar 5.19 dengan menggunakan frekuensi pencacah 5 Hz, sinyal hasil sampling lebih menyerupai sinyal asli.

Proses pencacahan seperti dalam Gambar 5.19 disebut dengan pencacahan ideal. Pencacahan ideal tidak mungkin dicapai dalam aplikasi nyata, karena membutuhkan peralatan yang dapat menghasilkan periode waktu setiap cacahan pendek sekali (setiap cacahan hanya berupa garis). Pencacahan natural akan menghasilkan cacahan berupa persegi panjang dengan tinggi sesuai dengan amplitudo gelombang, dan lebar sesuai dengan periode cacahan. Lihat ilustrasi dalam Gambar 5.20. Namun perangkat elektronik pencacah biasanya menggunakan metode *sample and hold* ketimbang menggunakan pencacahan ideal atau pencacahan natural.

#### *Proses Kuantisasi*

Pencacahan menghasilkan deretan pulsa PAM dengan amplitudo bervariasi dari nilai minimum tegangan sampai nilai maksimum tegangan sinyal analog asli.

Jumlah variasi amplitudo tak terhingga. Karena itu langkah selanjutnya adalah melakukan proses kuantisasi amplitudo.



Gambar 5.20. Pencacahan natural dan *sample and hold* (Forouzan, 2007)

Lebar kuantisasi ( $\Delta$ ) ditentukan dengan rumusan berikut:

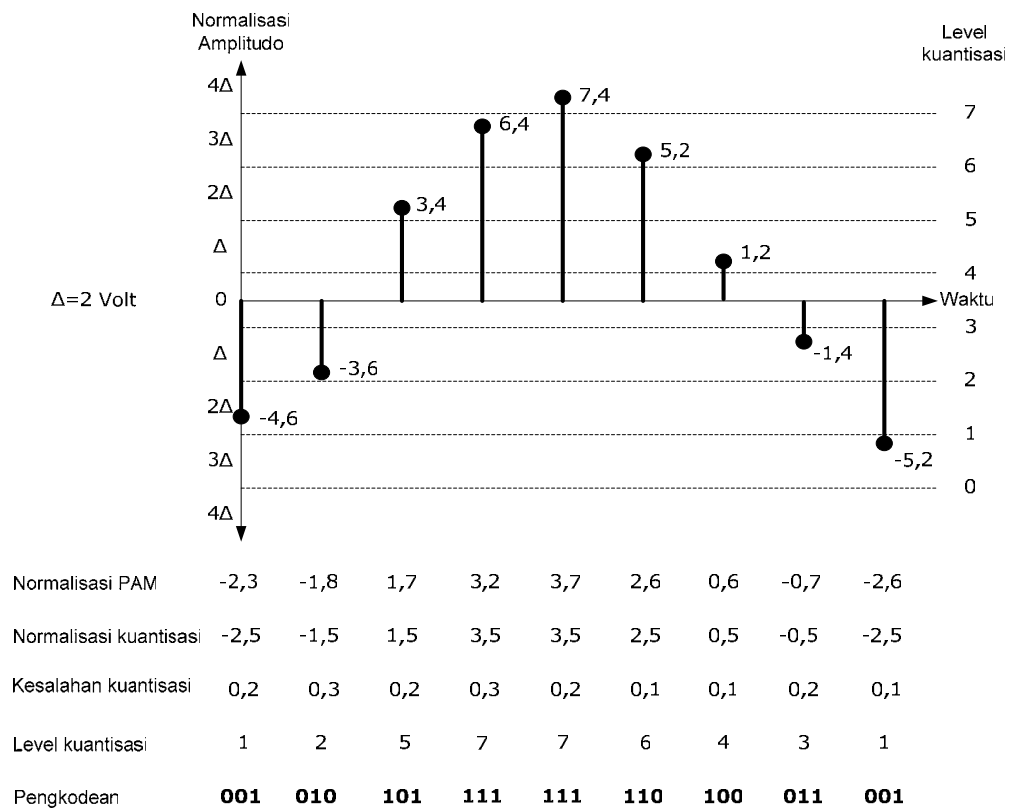
$$\Delta = \frac{V_{\max} - V_{\min}}{L} \quad (5.2)$$

Yang mana  $V_{\max}$  adalah tegangan maksimal dari sinyal analog asli dan  $V_{\min}$  adalah tegangan minimum yang dapat dicapai oleh sinyal analog asli dan  $L$  adalah jumlah level kuantisasi yang diinginkan.

Ilustrasi proses kuantisasi dapat dilihat dalam Gambar 5.21. Tegangan sinyal analog bervariasi antara -8 volt sampai 8 volt. Apabila diinginkan level kuantisasi sebanyak 8 level, maka dengan menggunakan persamaan 5.2 didapatkan lebar kuantisasi ( $\Delta$ ) = 2 volt. Normalisasi PAM dalam Gambar 5.21 adalah nilai tegangan PAM hasil dari pencacahan dibagi dengan  $\Delta$ . Sedangkan

normalisasi kuantisasi adalah hasil pembulatan normalisasi PAM ke level kuantisasi terdekat, dalam gambar level kuantisasi ditandai dengan garis terputus-putus yaitu pada:  $-3,5\Delta$ ;  $-2,5\Delta$ ;  $-1,5\Delta$ ;  $-0,5\Delta$ ;  $0,5\Delta$ ;  $1,5\Delta$ ;  $2,5\Delta$  dan  $3,5\Delta$ .

PCM dengan lebar kuantisasi ( $\Delta$ ) yang memiliki nilai tetap seperti terlihat dalam gambar disebut dengan kuantisasi seragam (*uniform quantization*). Dalam kasus yang lain, misalnya perubahan amplitudo sinyal analog lebih sering terjadi pada tegangan rendah, tidak digunakan kuantisasi seragam tetapi digunakan kuantisasi tidak seragam. Kuantisasi tidak seragam akan menghasilkan lebar kuantisasi berbeda-beda untuk setiap level kuantisasi.



Gambar 5.21. Proses kuantisasi dan pengkodean digital

Hal lain yang perlu mendapatkan perhatian khusus adalah adanya kesalahan kuantisasi akibat adanya pembulatan level tegangan PAM ke level kuantisasi terdekat. Nilai kesalahan dari setiap cacahan tidak akan melebihi  $\Delta/2$ , karena itu



kesalahan kuantisasi akan berada pada nilai  $-\Delta/2 \leq \text{kesalahan kuantisasi} \leq \Delta/2$ . Kesalahan kuantisasi berkontribusi pada peningkatan SNR dari sinyal yang tentu saja akan berakibat langsung pada penurunan kapasitas kanal (ingat teori Shannon dalam Bab 3). SNR akibat adanya kesalahan kuantisasi dirumuskan oleh persamaan 5.3.

$$SNR(dB) = 6,02 \times \log_2 L + 1,76 \quad (5.3)$$

Variabel  $L$  adalah jumlah level kuantisasi. Rata-rata kesalahan kuantisasi dapat dikurangi dengan memberikan penambahan derau dalam jumlah kecil. Proses penambahan derau seperti ini disebut dengan *dithering*. Perlu diketahui bahwa tidak semua derau bersifat mengganggu, justru sebaliknya derau yang terkendali akan sangat bermanfaat sebagaimana halnya implementasi *dithering* dalam proses kuantisasi. Untuk pengayaan terhadap materi ini, penulis menggunakan proses *dithering* dan mengajukan skema *dithering* dalam proses kuantisasi yang disebut dengan *sinusoidally-distributed dithering*. Untuk pembaca yang tertarik memahami proses *dithering* lebih lanjut, dipersilahkan untuk melihat literatur dalam Digital Communication (Sklar, 2001; Jusak, 2004).

*Contoh 5.8.*

Berapakah SNR akibat adanya kesalahan kuantisasi dari proses kuantisasi dalam Gambar 5.21.

*Jawaban:*

Dalam Gambar 5.21 terlihat bahwa proses kuantisasi menggunakan 8 level kuantisasi, berarti untuk setiap cacahan dibutuhkan representasi kode digital sebanyak 3 bit. Karena itu nilai SNR (dB) =  $6,02 \times \log_2(8) + 1,76 = 19,82$  dB. Sesuai dengan persamaan 5.3., apabila level kuantisasi dinaikkan, maka nilai SNR juga akan meningkat.

**Tidak semua derau bersifat mengganggu, justru sebaliknya derau yang terkendali akan sangat bermanfaat sebagaimana halnya implementasi *dithering* dalam proses kuantisasi. Pembaca yang tertarik memahami proses *dithering* lebih lanjut, dipersilahkan untuk melihat literatur dalam *Digital Communication* (Sklar, 2001; Jusak, 2004).**

#### *Proses Pengkodean Data Digital*

Langkah terakhir dalam metode PCM adalah pengkodean data digital. Seperti terlihat dalam Gambar 5.21, pengkodean digital terletak pada baris terakhir dalam gambar. Pengkodean ini mengubah level kuantisasi seperti dalam Gambar 5.21 ke bentuk digital. Misalnya level kuantisasi 7 memiliki bentuk digital 111, level kuantisasi 3 memiliki bentuk digital 011, dan seterusnya. Dengan cara demikian, sinyal analog sekarang telah berubah menjadi bentuk digital.

Kecepatan data dapat dihitung dengan rumusan dalam persamaan 5.4.

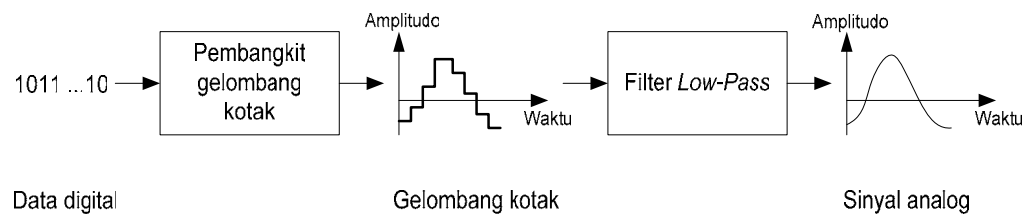
$$R = f_s \times \log_2 L \quad (5.4)$$

Yang mana  $R$  adalah kecepatan data dalam satuan bps, dan  $f_s$  adalah frekuensi cacahan dalam satuan Hz. Dalam persamaan 5.4,  $\log_2 L$  pada dasarnya adalah jumlah bit yang digunakan untuk merepresentasikan  $L$  level, sebagai contoh untuk  $L=8$ , maka dibutuhkan jumlah bit 3 seperti dapat dilihat dalam Gambar 5.21.

#### *PCM Pembalik Kode*

PCM pembalik kode (decoder) adalah peralatan yang digunakan untuk mendapatkan kembali sinyal analog yang telah dikodekan menjadi data digital. Perlu diingat bahwa untuk dapat melakukan pembalikan kode, persyaratan Nyquist harus dipenuhi pada saat melakukan pencacahan.

PCM pembalik kode terdiri atas dua bagian penting. Pertama, kode digital diubah menjadi sinyal dengan amplitudo sesuai dengan nilai integer kode digital, selanjutnya peralatan menahan tegangan pada nilai amplitudo tersebut sesaat sampai sinyal berikutnya datang. Sehingga peralatan ini akan menghasilkan sinyal dengan gelombang kotak yang merupakan representasi dari kode digital. Kedua, sinyal dengan gelombang kotak tersebut dihaluskan agar menjadi sinyal analog dengan menggunakan filter *low-pass*. Lihat Gambar 5.22.



Gambar 5.22. Ilustrasi pembalik kode PCM

#### *Bandwidth dari PCM*

Sekarang kita akan menghitung berapa bandwidth yang dihasilkan oleh pengkodean PCM. Kita telah melihat bahwa kecepatan sinyal rata-rata dapat dihitung dengan persamaan 5.1. Sekarang mari kita lakukan substitusi persamaan 5.4 ke dalam persamaan 5.1, menjadi:

$$S = k \times f_s \times \log_2 L \times \frac{1}{m} \quad (5.5)$$

$$S = k \times 2 \times f_{\max} \times \log_2 L \times \frac{1}{m}$$

Apabila kita asumsikan bahwa nilai  $k$  adalah  $\frac{1}{2}$  dan nilai  $m$  adalah 1, kita juga melakukan asumsi bahwa sinyal telah melewati filter low-pass ( $f_{\max}$ =Bandwidth) maka persamaan 5.5 dapat disederhanakan menjadi:

$$B_{pcm} = B_{analog} \times \log_2 L \quad (5.6)$$

Dari persamaan 5.6 dapat disimpulkan bahwa bandwidth minimum dari sinyal digital adalah  $\log_2 L$  kali lebih besar daripada bandwidth sinyal analog. Ini adalah nilai tambah dari penggunaan sinyal digital.

*Contoh 5.9.*

Sinyal analog dengan bandwidth 3 KHz jika dilewatkan pada sebuah kanal komunikasi membutuhkan bandwidth dari kanal sebesar 3 KHz. Apabila sinyal tersebut diubah menjadi sinyal digital dengan menggunakan metoda PCM dengan 3 bit per cacahan, berapa bandwidth dari kanal yang dibutuhkan?

*Jawaban:*

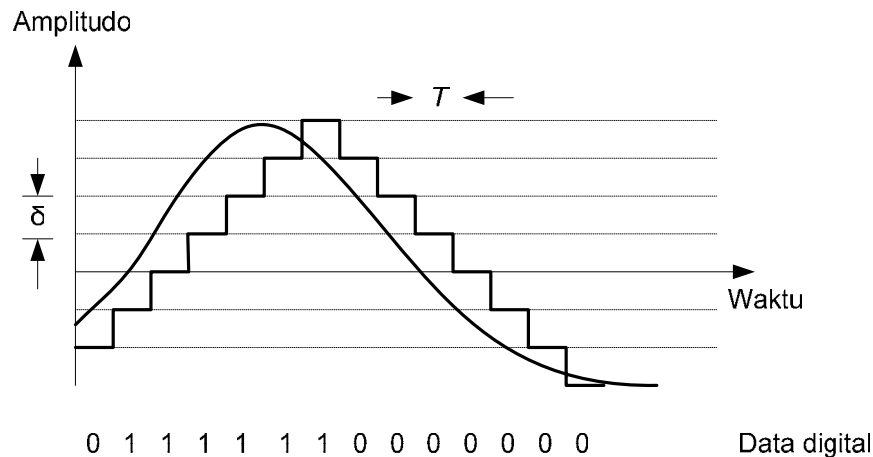
Bandwidth yang dibutuhkan adalah  $B_{pcm} = 3 \text{ KHz} \times 3 = 9 \text{ KHz}$ .

**Dari persamaan 5.6 dapat disimpulkan bahwa  
bandwidth minimum dari sinyal digital adalah  $\log_2 L$  kali lebih besar  
daripada bandwidth sinyal analog.**

### 5.2.2. Modulasi Delta

Teknik konversi dari sinyal analog menjadi data digital akan menjadi lebih sederhana apabila diimplementasikan dengan menggunakan Modulasi Delta daripada menggunakan PCM. Modulasi Delta tidak mendeteksi amplitudo sebagaimana halnya pada PCM, melainkan mendeteksi perubahan amplitudo antara cacahan saat ini dengan cacahan sebelumnya. Perbedaan antara amplitudo saat ini dengan amplitudo sebelumnya disebut dengan  $\delta$ . Perhatikan ilustrasi Modulasi Delta dalam Gambar 5.23. Apabila  $\delta$  bernilai positif, maka Modulasi Delta akan membangkitkan bit 1, sebaliknya apabila  $\delta$  bernilai negatif maka Modulasi Delta akan membangkitkan bit 0. Dengan demikian keluaran dari Modulasi Delta merupakan deretan bit yang menggambarkan perubahan amplitudo dari sinyal analog.

Untuk dapat menghasilkan unjuk kerja Modulasi Delta yang lebih baik,  $\delta$  dapat dibuat menjadi adaptif. Dengan menggunakan Modulasi Delta Adaptif nilai  $\delta$  akan berubah-ubah mengikuti amplitudo dari sinyal analog.



Gambar 5.23. Ilustrasi Modulasi Delta

Sebagaimana halnya pada PCM, kesalahan akibat kuantisasi juga terjadi pada Modulasi Delta. Namun secara umum dapat dikatakan bahwa kesalahan kuantisasi dari Modulasi Delta lebih kecil daripada PCM.

**Kesalahan kuantisasi dari Modulasi Delta lebih kecil daripada PCM.**

### 5.3. Soal Pengayaan

1. Mengapa pengkodean digital diperlukan? Carilah 3 alasan utama!
2. Kriteria-kriteria atau pertimbangan-pertimbangan apa yang digunakan untuk menentukan pemilihan jenis pengkodean digital dalam suatu implementasi?
3. Apa yang dimaksud dengan *dithering*? Carilah referensi bagaimana *dithering* dapat mengurangi kesalahan kuantisasi!
4. Berikan penjelasan mengapa pengkodean dua-fasa banyak digunakan dalam implementasi komunikasi data?
5. Apa keuntungan menggunakan pengkodean blok daripada menggunakan *line-coding*?

6. Buktikan bahwa persamaan untuk menentukan kecepatan sinyal rata-rata dari pengkodean MLT-3 adalah  $S=R/4$ .
7. Diketahui data digital 1010000011011110 akan ditransmisikan dengan kecepatan 25kbps. Tentukan:
  - a. Bentuk sinyal digital dengan menggunakan pengkodean NRZ-I.
  - b. Bentuk sinyal digital dengan menggunakan pengkodean Differential Manchester.
  - c. Bentuk sinyal digital dengan menggunakan pengkodean AMI.
  - d. Tentukan kecepatan sinyal untuk masing-masing pengkodean dalam soal a-c.
8. Sebuah sinyal analog memiliki bandwidth 128 KHz. Jika kita melakukan pencacahan terhadap sinyal tersebut dan mengirimkannya melalui kanal dengan kecepatan 160 Kbps, berapakah SNR (dB)?
9. Deratan bit 11100000000000 akan dikodekan dengan menggunakan metode *scrambling* B8ZS dan HDB3. Bagaimana bentuk sinyal hasil *scrambling* bila diasumsikan bahwa bit terakhir yang tidak nol memiliki tegangan positif?
10. Berapakah frekuensi cacahan minimal yang dapat ditentukan untuk sinyal-sinyal berikut ini:
  - a. Sinyal analog yang dilewatkan melalui filter low-pass dengan bandwidth 300 KHz.
  - b. Sinyal analog yang dilewatkan melalui filter band-pass dengan bandwidth 300 KHz jika frekuensi terendah adalah 100 KHz.
11. Berapakah kecepatan data pada keluaran PCM apabila digunakan frekuensi cacahan sebesar 500 cacah per menit dan pengkodean digital dengan menggunakan 128 level?
12. Sinyal analog dikirimkan dengan menggunakan lebar pita frekuensi 20 KHz. Jika sinyal tersebut diubah menjadi sinyal digital dengan proses sampling untuk dilewatkan pada kanal dengan kecepatan 30Kbps, berapakah nilai dari SNR (dalam dB)?