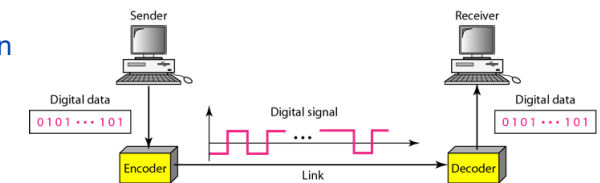




Konversi Data Digital ke Sinyal Digital

- Pada transmisi digital, data yang dihasilkan oleh transmitter berupa data digital dan ditransmisikan dalam bentuk sinyal digital menuju ke receiver (penerima).
- Pada bagian ini, kita melihat bagaimana kita dapat mewakili data digital dengan menggunakan sinyal digital.
- Konversi ini melibatkan tiga teknik :

- a. Line coding
- b. Blok coding, dan
- c. Scrambling.



Tujuan Line Coding

1. Merekayasa spektrum sinyal digital agar sesuai dengan medium transmisi yang akan digunakan.
2. Dapat dimanfaatkan untuk proses sinkronisasi antara pengirim dan penerima (sistem tidak memerlukan jalur terpisah untuk clock).
3. Dapat digunakan untuk menghilangkan sinyal yang mengandung komponen DC (sinyal dengan frekuensi 0). Komponen DC tidak mengandung informasi apapun tetapi menghamburkan daya pancar.
4. Line coding dapat digunakan untuk menaikkan data rate.
5. Beberapa teknik line coding dapat digunakan untuk pendeteksian kesalahan.

Karakteristik Line Coding

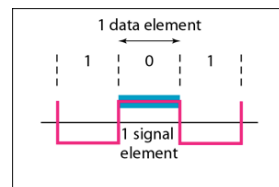
Karakteristik Line Coding:

1. Elemen data dan Elemen Sinyal
2. Data Rate dan Signal Rate
3. Bandwidth
4. Self Synchronization
5. Baseline Wandering
6. Komponen DC
7. Sinkronisasi Bit
8. Deteksi Bit Error
9. Mengurangi Noise
10. Kompleksitas

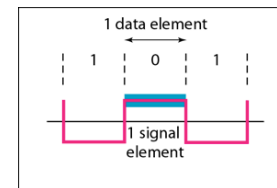
1. Elemen Data dan Elemen Sinyal

- Pada komunikasi data, elemen data merupakan satuan terkecil yang dapat mewakili beberapa informasi yang berupa bit, sedangkan elemen sinyal merupakan unit terpendek dari sinyal digital dimana elemen sinyal membawa elemen data.
- Dengan kata lain elemen data adalah apa yang kita butuhkan untuk dikirim, sedangkan elemen sinyal adalah apa yang dapat kita kirim.

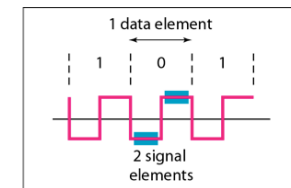
- Dalam Komunikasi Data, tujuannya adalah untuk mengirim Elemen Data



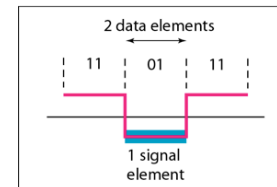
a. One data element per one signal element ($r = 1$)



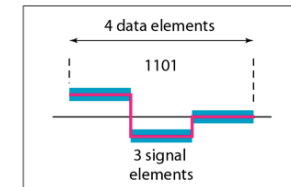
a. One data element per one signal element ($r = 1$)



b. One data element per two signal elements ($r = \frac{1}{2}$)



c. Two data elements per one signal element ($r = 2$)



d. Four data elements per three signal elements ($r = \frac{4}{3}$)

2. Data Rate dan Signal Rate

- Data rate adalah sejumlah elemen data dalam satu detik, satuannya adalah bit per detik (bps) sedangkan sinyal rate adalah sejumlah elemen sinyal yang dikirim dalam 1 detik, satuannya adalah Baud.

Data Rate=Bit Rate,

Signal Rate=Pulse Rate/modulation Rate/Baud Rate.

- Rumus yang digunakan untuk menghitung sinyal rate adalah:

$$S = C \times N \times \frac{1}{r}$$

S = Signal Rate
r = elemen sinyal
N = data rate (bps)
C = Nilai konstanta (nilai bisa bervariasi untuk tiap kasus)

Contoh soal

Sebuah sinyal yang membawa data di mana salah satu elemen data yang dikodekan sebagai salah satu elemen sinyal ($r = 1$). Jika bit rate adalah 100 kbps, berapakah nilai rata-rata baud rate jika c adalah antara 0 dan 1?

Jawab:

Kita asumsikan bahwa nilai rata-rata c adalah $\frac{1}{2}$.

Maka baud rate ini adalah sebagai berikut :

$$S = C \times N \times \frac{1}{r} = \frac{1}{2} \times 100.000 \times \frac{1}{1} = 50.000 = 50 \text{ kbaud}$$

3. Bandwidth (Lebar Pita)

- Bandwidth adalah suatu sistem komunikasi elektronika yang mengirimkan informasi dengan memancarkan energi elektromagnetik.
- Energi elektromagnetik ini dapat berjalan sebagai sebuah tegangan atau arus yang melalui kawat sebagaimana emisi radio melintasi udara dan cahaya. Untuk mengirim informasi, sistem komunikasi harus menggunakan spektrum elektromagnetik dalam jumlah atau range tertentu.
- Bandwidth menunjukkan ukuran kapasitas jalur transmisi yang dinyatakan dalam satuan, yakni:
 - Baud (Bd) adalah kecepatan modulasi
 - Bit perdetik (bps) adalah kecepatan sinyal
 - Karakter perdetik (cps) adalah kecepatan transmisi

Contoh soal

Saluran data rate maksimum adalah $N_{\max} = 2 \times B \times \log_2 L$ (didefinisikan oleh rumus Nyquist). Apakah ini sama dengan rumus sebelumnya untuk N_{\max} ?

Penyelesaian :

Sebuah sinyal dengan tingkat L benar-benar dapat membawa bit $\log_2 L$ per tingkat. Jika setiap tingkat sesuai dengan salah satu elemen sinyal dan kita asumsikan rata-ratanya jika ($c = 1/2$), maka kita harus

$$N_{\max} = 1/c \times B \times r = 2 \times B \times \log_2 L$$

- Rumus mencari bandwidth minimum :

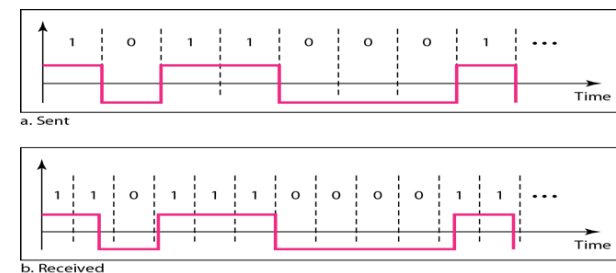
$$B_{\min} = c \times N \times 1/r$$

- Dan data rate jika bandwidth saluran diberikan :

$$N_{\max} = 1/c \times B \times r$$

4. Self Synchronization

- Untuk mengoreksi sinyal yang diterima dari pengirim dengan benar, interval bit penerima harus sesuai dengan pengirim interval bit. Jika clock penerima lebih cepat atau lebih lambat, interval bit tidak cocok dan penerima mungkin akan salah mengoreksi sinyal.



5. Baseline Wandering

- Didalam decoding sinyal digital, penerima menghitung rata-rata menjalankan kekuatan sinyal yang diterima. Rata-rata ini disebut baseline.
- Kekuatan sinyal yang masuk dievaluasi dibandingkan dengan data dasar untuk menentukan nilai dari elemen data. String 0 atau 1 yang sangat panjang dapat menyebabkan penyimpangan dalam baseline (*baseline wandering*) dan membuat sulit bagi penerima untuk memecahkan kode dengan benar.
- Skema line coding yang baik perlu untuk mencegah baseline wandering

Contoh

In a digital transmission, the receiver clock is 0.1 percent faster than the sender clock. How many extra bits per second does the receiver receive if the data rate is 1 kbps? How many if the data rate is 1 Mbps?

Solution

At 1 kbps, the receiver receives 1001 bps instead of 1000 bps.

1000 bits sent	1001 bits received	1 extra bps
----------------	--------------------	-------------

At 1 Mbps, the receiver receives 1,001,000 bps instead of 1,000,000 bps.

1,000,000 bits sent	1,001,000 bits received	1000 extra bps
---------------------	-------------------------	----------------

6. Self-synchronization

- Untuk dapat menafsirkan secara benar sinyal yang diterima dari pengirim, interval bit penerima harus sesuai persis dengan interval bit pengirim. Jika *clock* di penerima lebih cepat atau lebih lambat, maka interval bit tidak cocok dan penerima mungkin salah menafsirkan sinyal.
- Men-sinkronisasi sendiri (*self synchronization*) sinyal digital termasuk mengatur waktu (timing) data yang dikirim. Hal ini dapat dicapai apabila ada transisi didalam sinyal yang memberitahukan penerima untuk awal, tengah, atau akhir pulsa. Jika clock penerima tidak sinkron, maka pada titik tersebut dapat mengatur ulang clock.

7. Deteksi bit error

Deteksi bit error adalah kemampuan yang dimiliki oleh sinyal digital untuk mendeteksi kesalahan dalam proses transmisi.

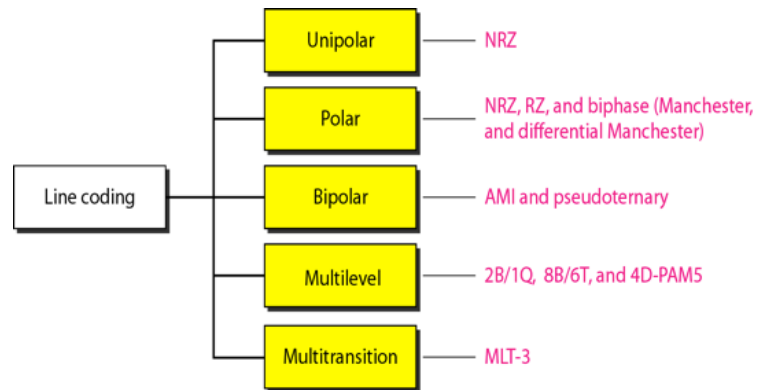
8. Mengurangi noise

Noise adalah tambahan sinyal yang tidak diinginkan yang masuk dimana pun diantara pengirim dan penerima. Terdapat beberapa skema yang dapat tahan terhadap noise atau gangguan yang lain.

9. Kompleksitas

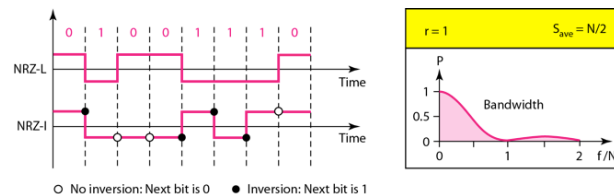
Skema yang kompleks lebih mahal untuk diterapkan daripada yang sederhana. Misalnya, skema yang menggunakan sinyal empat tingkat lebih sulit untuk diinterpretasikan dibandingkan dengan hanya menggunakan dua tingkat

Skema Line Coding



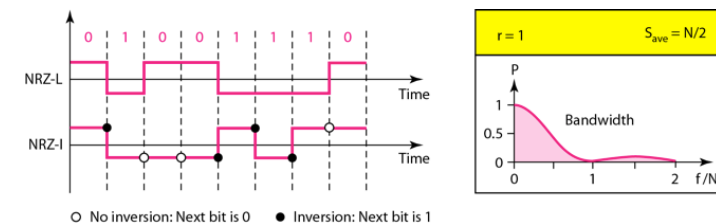
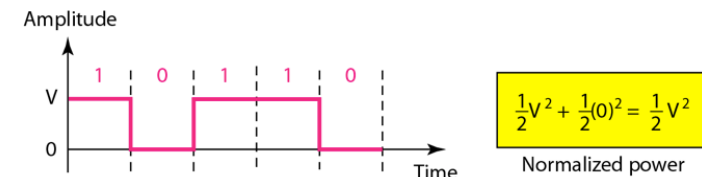
NRZ (Non Return To Zero)

- Awalnya, skema unipolar dibuat untuk non-return to zero (NRZ) yang mana tegangan positif mendefinisikan bit 1 dan tegangan 0 mendefinisikan bit 0. Disebut non return to zero level dikarenakan sinyal tidak kembali ke nol pada tengah dari bit.
- Pada pengkodean polar NRZ, digunakan dua level tegangan. Terdapat dua jenis NRZ polar yaitu NRZ-L dan NRZ-I. Sebagaimana ditunjukkan pada gambar dibawah.



Skema Unipolar

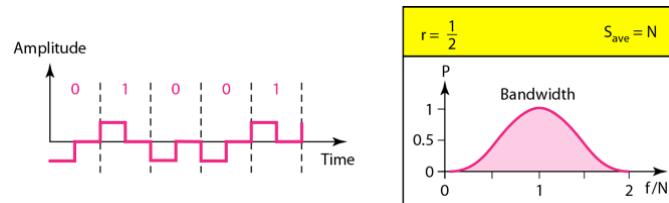
- Skema Unipolar adalah semua elemen sinyal yang mempunyai tanda yang sama, yaitu positif semua atau negatif semua.
- Dalam skema unipolar, semua level sinyal berada di salah satu sisi dari sumbu waktu, baik di atas atau di bawah.



- Pada NRZ-L, level tegangan mendefinisikan nilai dari bit, sedangkan pada NRZ-I, perubahan atau tidak adanya perubahan level tegangan menentukan nilai dari bit. Jika tidak ada perubahan, bit nya adalah 0, jika terdapat perubahan nilai bit nya adalah 1.
- Kelemahan dari skema pengkodean ini adalah baik NRZ-L dan NRZ-I sama-sama memiliki masalah komponen DC.

Return to Zero (RZ)

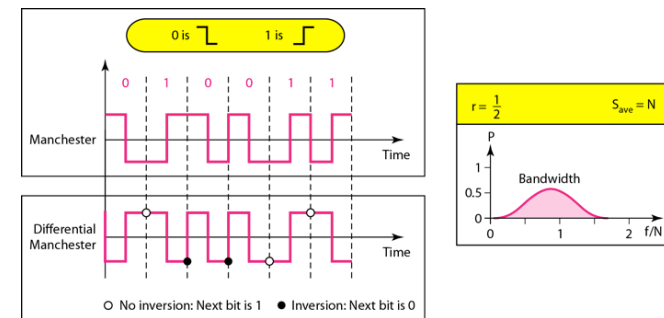
- Masalah utama dengan pengkodean NRZ terjadi ketika pengirim dan penerima clock tidak disinkronkan. Penerima tidak tahu kapan salah satu bit telah berakhir dan bit berikutnya mulai. Salah satu solusinya adalah kembali ke nol yaitu skema RZ, yang menggunakan tiga nilai positif, negatif dan nol.



Keuntungan Biphas

- Skema manchester mengatasi beberapa masalah yang terkait dengan NRZ-L, dan Diferensial manchester mengatasi beberapa masalah yang terkait dengan NRZ-I.
- Pada skema ini tidak terdapat
 - baseline wandering*.
 - Tidak ada komponen DC karena setiap bit memiliki kontribusi tegangan positif dan negatif.
- Satu-satunya kelemahan adalah tingkat sinyal. tingkat sinyal untuk manchester dan diferensial manchester adalah dua kali lipat untuk NRZ. Alasannya adalah bahwa selalu ada satu transisi di tengah bit dan mungkin satu transmisi pada akhir setiap bit.

Biphase: Manchester dan Differential Manchester

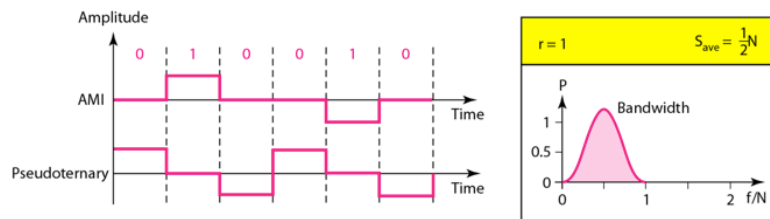


- Di Manchester dan diferensial Manchester encoding, transisi di tengah bit yang digunakan untuk sinkronisasi.
- Bandwidth minimum dari Manchester dan Manchester diferensial adalah 2 kali dari NRZ.

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.
- Dalam encoding bipolar, kita menggunakan tiga tingkatan: positif, nol, dan negatif.

Keuntungan Bipolar



- Pada pengkodean AMI, elemen data dengan bit 1 direpresentasikan oleh sinyal yang berinversi 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 berinversi 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.

Skema Pengkodean Multilevel

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

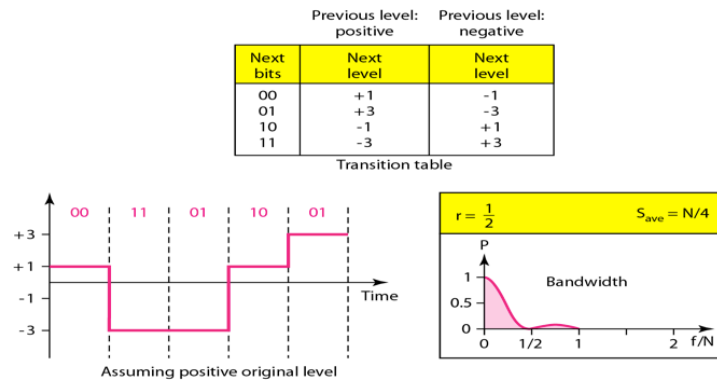
- Keuntungan menggunakan menggunakan pengkodean bipolar adalah:
 - tidak memiliki komponen DC,
 - membutuhkan bandwidth dua kali lebih kecil daripada pengkodean yang telah kita bicarakan sebelumnya.

2B1Q (Two Binary One Quaternary)

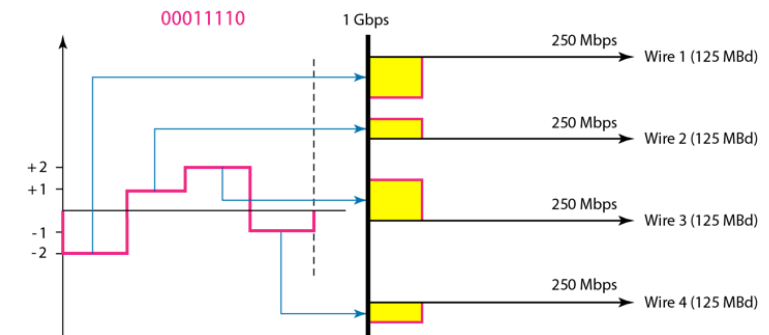
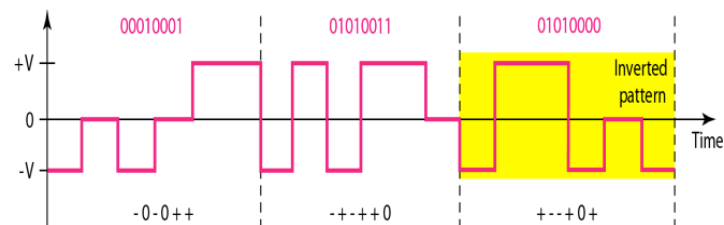
- Model pengkodean multilevel yang saat ini digunakan oleh Digital Subscriber Line (DSL) adalah two binary one quaternary (2B1Q). 2B1Q berarti setiap 2 bit data dikodekan ke dalam 1 elemen sinyal yang memiliki 4 level tegangan. Sehingga nilai $m=2$ dan kecepatan sinyal rata-rata adalah $S=R/4$ baud. Dengan demikian nilai $m=2$, $n=1$ dan $L=4$.
- Dalam aplikasi dibutuhkan agar $2m \leq Ln$ sehingga tidak semua pola sinyal merupakan representasi dari pola data. Pola sinyal selebihnya dapat digunakan untuk sinkronisasi dan pendeteksi kesalahan.

8B6T (Eight Binary Six Ternary)

- 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 $3^6 = 729$ pola sinyal. Karena itu akan terdapat sisa pola sinyal yang tidak digunakan merepresentasikan pola data sebanyak 223 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.

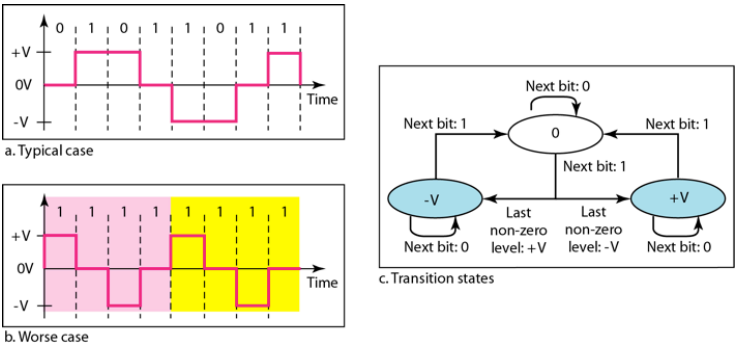


4D-PAM5 (Four Dimensional Five Level Pulse Amplitude Modulation)



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



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

Category	Scheme	Bandwidth (average)	Characteristics
Unipolar	NRZ	$B = N/2$	Costly, no self-synchronization if long 0s or 1s, DC
Unipolar	NRZ-L	$B = N/2$	No self-synchronization if long 0s or 1s, DC
	NRZ-I	$B = N/2$	No self-synchronization for long 0s, DC
	Biphase	$B = N$	Self-synchronization, no DC, high bandwidth
Bipolar	AMI	$B = N/2$	No self-synchronization for long 0s, DC
Multilevel	2B1Q	$B = N/4$	No self-synchronization for long same double bits
	8B6T	$B = 3N/4$	Self-synchronization, no DC
	4D-PAM5	$B = N/8$	Self-synchronization, no DC
Multiline	MLT-3	$B = N/3$	No self-synchronization for long 0s

- Next.....
 - Block Coding
 - Scrambling