

Data Communications and Networking Fourth Edition



Chapter 4 Digital Transmission

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

4-1 DIGITAL-TO-DIGITAL CONVERSION

In this section, we see how we can represent digital data by using digital signals. The conversion involves three techniques: line coding, block coding, and scrambling. Line coding is always needed; block coding and scrambling may or may not be needed.

Topics discussed in this section:

- Line Coding
- Line Coding Schemes
- Block Coding
- Scrambling

Line Coding

- It's the process of converting digital data to digital signals.
- The data are stored in computer memory as sequences of bits
- Converting a string of 1's and 0's (digital data) into a sequence of signals that denote the 1's and 0's.
- For example a high voltage level (+V) could represent a "1" and a low voltage level (0 or -V) could represent a "0".

Figure 4.1 *Line coding and decoding*



Mapping Data symbols onto Signal levels

- A data symbol (or element) can consist of a number of data bits:
 - 1,0 or
 - **11**, 10, 01,
- A data symbol can be coded into a single signal element or multiple signal elements

■ 1 -> +V, 0 -> -V

- 1 -> +V and -V, 0 -> -V and +V
- The ratio 'r' is the number of data elements carried by a signal element.

Signal Element Versus Data Element

- A data element is the smallest entity that can represent a piece of information
- A signal element is the shortest unit (timewise) of a digital signal.
- Data elements are what we need to send; signal elements are what we can send.
- Suppose each data element is a person who needs to be carried from one place to another.
- We can think of a signal element as a vehicle that can carry people.
- When r = 1, it means each person is driving a vehicle. When r > 1, it means more than one person is travelling in a vehicle (a carpool, for example).
- We can also have the case where one person is driving a car and a trailer (r = 1/2).

Relationship between data rate and signal rate

- The data rate defines the number of bits sent per sec - **bps**. It is often referred to the bit rate.
- The signal rate is the number of signal elements sent in a second and is measured in **bauds**. It is also referred to as the modulation rate.
- Goal is to increase the data rate while reducing the baud rate.
- We now need to consider the relationship between data rate (N) and signal rate bauds (S)
 S = N/r

Figure 4.2 Signal element versus data element



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



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



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



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

Data rate and Baud rate

The **baud** or signal rate can be expressed as:

S_ave = c x N x 1/r bauds where N is data rate c is the case factor (worst, best & avg.) r is the ratio between data element & signal element A signal is carrying data in which one data element is encoded as one signal element (r = 1). If the bit rate is 100 kbps, what is the average value of the baud rate if c is between 0 and 1?

Solution

We assume that the average value of c is 1/2. The baud rate is then

$$S = c \times N \times \frac{1}{r} = \frac{1}{2} \times 100,000 \times \frac{1}{1} = 50,000 = 50$$
 kbaud

The Bandwidth

- As learned previously, a digital signal that carries information is nonperiodic and its bandwidth is continuous and infinite
- Also, many of its components have very small amplitude that can be ignored
- Therefore, the "effective bandwidth" of the reallife digital signal is finite
- The bandwidth is proportional to the signal rate
- Given N: $B_{\min} = c \times N \times \frac{1}{m}$

Given B:

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

4.10



Although the actual bandwidth of a digital signal is infinite, the effective bandwidth is finite.

Example 4.2

The maximum data rate of a channel (see Chapter 3) is $N_{max} = 2 \times B \times \log_2 L$ (defined by the Nyquist formula). Does this agree with the previous formula for N_{max} ?

Solution

A signal with L levels actually can carry log_2L bits per level. If each level corresponds to one signal element and we assume the average case (c = 1/2), then we have

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

Considerations for choosing a good signal element referred to as line encoding

- Baseline wandering a receiver will evaluate the average power of the received signal (called the baseline) and use that to determine the value of the incoming data elements. If the incoming signal does not vary over a long period of time, the baseline will drift and thus cause errors in detection of incoming data elements.
- A good line encoding scheme will prevent long runs of fixed amplitude.

Baseline Wandering

- While decoding the received signal, the receiver, calculates the running average of the received signal power, called the "baseline"
- The incoming signal power is evaluated against the baseline in order to determine the value of the data element
- Long strings of 0's or 1's can cause a drift of the baseline, called "baseline wandering", which makes it hard for the receiver to correctly decode the signal
- A good line coding scheme should prevent baseline wandering

- DC components when the voltage level remains constant for long periods of time, there is an increase in the low frequencies of the signal. Most channels are bandpass and may not support the low frequencies.
- This will require the removal of the dc component of a transmitted signal.

- Self synchronization the clocks at the sender and the receiver must have the same bit interval.
- If the receiver clock is faster or slower it will misinterpret the incoming bit stream.

Figure 4.3 Effect of lack of synchronization



a. Sent





Example 4.3

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 sent1001 bits received1 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

- Error detection errors occur during transmission due to line impairments.
- Some codes are constructed such that when an error occurs it can be detected. For example: a particular signal transition is not part of the code. When it occurs, the receiver will know that a symbol error has occurred.

- Noise and interference there are line encoding techniques that make the transmitted signal "immune" to noise and interference.
- This means that the signal cannot be corrupted, it is stronger than error detection.

Complexity - the more robust and resilient the code, the more complex it is to implement and the price is often paid in baud rate or required bandwidth.

Figure 4.4 *Line coding schemes*



Unipolar

- All signal levels are on one side of the time axis - either above or below
- NRZ Non Return to Zero scheme is an example of this code. The signal level does not return to zero during a symbol transmission.
- Scheme is prone to baseline wandering and DC components. It has no synchronization or any error detection. It is simple but costly in power consumption.

Figure 4.5 Unipolar NRZ scheme



Polar - NRZ

- The voltages are on both sides of the time axis.
- Polar NRZ scheme can be implemented with two voltages. E.g. +V for 1 and -V for 0.
- There are two versions:
 - NZR Level (NRZ-L) positive voltage for one symbol and negative for the other
 - NRZ Inversion (NRZ-I) the change or lack of change in polarity determines the value of a symbol. E.g. a "1" symbol inverts the polarity a "0" does not.

Figure 4.6 Polar NRZ-L and NRZ-I schemes





In NRZ-L the level of the voltage determines the value of the bit. In NRZ-I the inversion or the lack of inversion determines the value of the bit.



NRZ-L and NRZ-I both have an average signal rate of N/2 Bd.



NRZ-L and NRZ-I both have a DC component problem and baseline wandering, it is worse for NRZ-L. Both have no self synchronization &no error detection. Both are relatively simple to implement.

A system is using NRZ-I to transfer 1-Mbps data. What are the average signal rate and minimum bandwidth?

Solution

The average signal rate is $S = c \times N \times R = 1/2 \times N \times 1 = 500$ kbaud. The minimum bandwidth for this average baud rate is $B_{min} = S = 500$ kHz.

Note c = 1/2 for the avg. case as worst case is 1 and best case is 0

Polar - RZ

- The Return to Zero (RZ) scheme uses three voltage values. +, 0, -.
- Each symbol has a transition in the middle.
 Either from high to zero or from low to zero.
- This scheme has more signal transitions (two per symbol) and therefore requires a wider bandwidth.
- No DC components or baseline wandering.
- Solve Self synchronization transition indicates symbol value.
- More complex as it uses three voltage level. It has no error detection capability.

Figure 4.7 Polar RZ scheme





Polar - Biphase: Manchester and Differential Manchester

- Manchester coding consists of combining the NRZ-L and RZ schemes.
 - Every symbol has a level transition in the middle: from high to low or low to high. Uses only two voltage levels.
- Differential Manchester coding consists of combining the NRZ-I and RZ schemes.
 - Every symbol has a level transition in the middle. But the level at the beginning of the symbol is determined by the symbol value. One symbol causes a level change the other does not.

Figure 4.8 Polar biphase: Manchester and differential Manchester schemes





In Manchester and differential Manchester encoding, the transition at the middle of the bit is used for synchronization.



The minimum bandwidth of Manchester and differential Manchester is 2 times that of NRZ. The is no DC component and no baseline wandering. None of these codes has error detection.

Bipolar - AMI and Pseudoternary

- Code uses 3 voltage levels: +, 0, -, to represent the symbols (note not transitions to zero as in RZ).
- Voltage level for one symbol is at "0" and the other alternates between + & -.
- Bipolar Alternate Mark Inversion (AMI) the "0" symbol is represented by zero voltage and the "1" symbol alternates between +V and -V.
- Pseudoternary is the reverse of AMI.

Figure 4.9 Bipolar schemes: AMI and pseudoternary



Bipolar C/Cs

- It is a better alternative to NRZ.
- Has no DC component or baseline wandering.
- Has no self synchronization because long runs of "0"s results in no signal transitions.
- No error detection.

Multilevel Schemes

- In these schemes we increase the number of data bits per symbol thereby increasing the bit rate.
- Since we are dealing with binary data we only have 2 types of data element a 1 or a 0.
- We can combine the 2 data elements into a pattern of "m" elements to create "2^m" symbols.
- If we have L signal levels, we can use "n" signal elements to create Lⁿ signal elements.

Code C/Cs

- Now we have 2^m symbols and Lⁿ signals.
- If 2^m = Lⁿ then we have an exact mapping of one symbol on one signal.
- If 2^m < Lⁿ then we have more signals than symbols and we can choose the signals that are more distinct to represent the symbols and therefore have better noise immunity and error detection as some signals are not valid.
- If 2^m > Lⁿ then we cannot represent the data elements, we don't have enough signals.
- mBnL : Binary patterns of length m, signal patterns of length n with L signal levels
- L can be replaced by:
- B (Binary) : L=2 (e.g: 2B2B)
- T (Ternary) : L=3 (e.g: 8B6T)
- > Q (Quaternary) : L=4 (e.g: 2B1Q)



In *m*B*n*L schemes, a pattern of *m* data elements is encoded as a pattern of *n* signal elements in which $2^m \leq L^n$.

Figure 4.10 Multilevel: 2B1Q scheme



4.44

Redundancy

- In the 2B1Q scheme we have no redundancy and we see that a DC component is present.
- If we use a code with redundancy we can decide to use only "0" or "+" weighted codes (more +'s than -'s in the signal element) and invert any code that would create a DC component. E.g. `+00++-' -> `-00--+'
- Receiver will know when it receives a "-" weighted code that it should invert it as it doesn't represent any valid symbol.

Figure 4.11 Multilevel: 8B6T scheme



Multilevel using multiple channels

- In some cases, we split the signal transmission up and distribute it over several links.
- The separate segments are transmitted simultaneously. This reduces the signalling rate per link -> lower bandwidth.
- This requires all bits for a code to be stored.
- xD: means that we use 'x' links
- YYYz: We use 'z' levels of modulation where YYY represents the type of modulation (e.g. pulse ampl. mod. PAM).
- Codes are represented as: xD-YYYz

Figure 4.12 Multilevel: 4D-PAM5 scheme



Multitransition Coding

- Because of synchronization requirements we force transitions. This can result in very high bandwidth requirements -> more transitions than are bits (e.g. mid bit transition with inversion).
- Codes can be created that are differential at the bit level forcing transitions at bit boundaries. This results in a bandwidth requirement that is equivalent to the bit rate.
- In some instances, the bandwidth requirement may even be lower, due to repetitive patterns resulting in a periodic signal.

Figure 4.13 *Multitransition: MLT-3 scheme*





c. Transition states

b. Worse case

MLT-3

- Signal rate is same as NRZ-I
- But because of the resulting bit pattern, we have a periodic signal for worst case bit pattern: 1111
- This can be approximated as an analog signal a frequency 1/4 the bit rate!

Table 4.1 Summary of line coding schemes

| 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 | |

Block Coding

- For a code to be capable of error detection, we need to add redundancy, i.e., extra bits to the data bits.
- Synchronization also requires redundancy transitions are important in the signal flow and must occur frequently.
- Block coding is done in three steps: division, substitution and combination.
- It is distinguished from multilevel coding by use of the slash - xB/yB.
- The resulting bit stream prevents certain bit combinations that when used with line encoding would result in DC components or poor sync. quality.



Block coding is normally referred to as *m*B/*n*B coding; it replaces each *m*-bit group with an *n*-bit group.

Figure 4.14 Block coding concept



Division of a stream into m-bit groups

Combining n-bit groups into a stream

Figure 4.15 Using block coding 4B/5B with NRZ-I line coding scheme



Table 4.24B/5B mapping codes

| Data Sequence | Encoded Sequence | Control Sequence | Encoded Sequence |
|---------------|------------------|---------------------|------------------|
| 0000 | 11110 | Q (Quiet) | 00000 |
| 0001 | 01001 | I (Idle) | 11111 |
| 0010 | 10100 | H (Halt) | 00100 |
| 0011 | 10101 | J (Start delimiter) | 11000 |
| 0100 | 01010 | K (Start delimiter) | 10001 |
| 0101 | 01011 | T (End delimiter) | 01101 |
| 0110 | 01110 | S (Set) | 11001 |
| 0111 | 01111 | R (Reset) | 00111 |
| 1000 | 10010 | | |
| 1001 | 10011 | | |
| 1010 | 10110 | | |
| 1011 | 10111 | | |
| 1100 | 11010 | | |
| 1101 | 11011 | | |
| 1110 | 11100 | | |
| 1111 | 11101 | | |

Figure 4.16 Substitution in 4B/5B block coding



5-bit blocks

Redundancy

- A 4 bit data word can have 24 combinations.
- A 5 bit word can have 25=32 combinations.
- We therefore have 32 26 = 16 extra words.
- Some of the extra words are used for control/signalling purposes.

We need to send data at a 1-Mbps rate. What is the minimum required bandwidth, using a combination of 4B/5B and NRZ-I or Manchester coding?

Solution

First 4B/5B block coding increases the bit rate to 1.25 Mbps. The minimum bandwidth using NRZ-I is N/2 or 625 kHz. The Manchester scheme needs a minimum bandwidth of 1.25 MHz. The first choice needs a lower bandwidth, but has a DC component problem; the second choice needs a higher bandwidth, but does not have a DC component problem.

Figure 4.17 8B/10B block encoding



More bits - better error detection

The 8B10B block code adds more redundant bits and can thereby choose code words that would prevent a long run of a voltage level that would cause DC components.

Scrambling

- The best code is one that does not increase the bandwidth for synchronization and has no DC components.
- Scrambling is a technique used to create a sequence of bits that has the required c/c's for transmission - self clocking, no low frequencies, no wide bandwidth.
- It is implemented at the same time as encoding, the bit stream is created on the fly.
- It replaces `unfriendly' runs of bits with a violation code that is easy to recognize and removes the unfriendly c/c.

Figure 4.18 AMI used with scrambling

