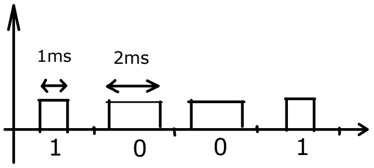
Line Codes

Line codes are the waveforms used to transmit data over baseband channels. Many different line codes have been developed. After this lecture you should be able to: distinguish between baseband and passband channels; explain reasons for use of bit stuffing; apply and remove bit stuffing; identify characteristics of a line code including: number of transitions per bit, number of levels, unipolar vs bi-polar, use of differential encoding, block vs bit-by-bit encoding, bandwidth, DC balance, maximum time between transitions; and encode/decode data to/from the line codes described below.

Lines Codes

Line codes encode data bits into waveforms that are suitable for transmission over baseband channels.

There are many ways this conversion of bits to waveforms could be done. For example, a data value of 0 could be represented by a 1ms pulse and a data value 1 could be represented by a 2ms pulse.



There are dozens of different line codes in use. They have been developed to match the characteristics of different channels, different data rates, different implementation technologies and different cost/performance requirements. The choice of line code will depend on the requirements of the application and will include compromises between bandwidth, DC content, transition density and implementation complexity.

This lecture describes the characteristics of line codes and a few representative examples.

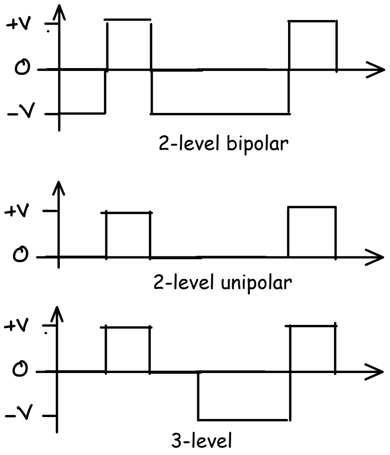
Characteristics of Line Codes

Voltage Levels

Most line codes use only two voltage levels. These are the simplest to implement and have the largest noise margin for a given signal power.

The two voltages may be balanced about zero (sometimes called a bipolar code) or one of the levels may be zero (sometimes called unipolar).

Multi-level codes are also common. A typical example is a bipolar line code that alternates a zero value and two polarities (negative and positive voltages of equal magnitude). This allows the average value of the waveform (the DC content) to be zero.



Exercise 1: What is the noise margin for a unipolar line code using levels of ± 1 V? What are the voltage levels for a bipolar lines with the same noise margin? What are the RMS voltages of these two line codes when transmitting a dotting sequence (alternating 1's and 0's)? Why might you use unipolar line codes anyways?

Transitions

Some codes allow a maximum of one transition per bit, while others allow two. More transitions per symbol typically means a higher bandwidth for a given symbol rate (but the actual spectrum will depend on the statistics of the transitions which in turn depend on the details of the line code).

Some codes guarantee a minimum number of transitions in a given time interval. This can help the receiver synchronize itself to the received waveform so that it can accurately recover the data.

Differential Coding

It is possible to encode each bit as a waveform that is either different or the same as the previous waveform. For example, with two voltage levels a data value of '1' could be transmitted as a change from the previous level and '0' as no change.

This "differential" encoding makes the data insensitive to the channel polarity. This is convenient since it allows cable pairs to be connected without regards to polarity.

One disadvantage of differential encoding is that errors always occur in pairs and this doubles the bit error rate.

Exercise 2: Assume a 1 is transmitted as 5V and 0 as 0V. Draw the waveform for the bit sequence 1011. Draw the waveform if the bits are transmitted differentially with a 1 encoded as a change in level. Assume the initial value of the waveform is 0. Invert the waveform and decode it.

Block Line Codes

Some line codes operate on blocks of bits. They read a number of data bits from the source and output a larger number of bits (or symbols). For example, a transmitter can read a block of 3 bits bits and output 4 bits. This called a 3B4B block code (B stands for binary).

Exercise 3: How many combinations are there of 3 bits? Of 4 bits? How many bits might be input and output by an 8B10B code? What might a 4B3T code mean?

The advantage of a block code is that the possible output patterns do not have to include those with undesirable properties. For example, the all-high or all-low waveforms might be omitted because the average DC content is too high or too low. Other output waveforms might be excluded because there is too long an interval between transitions.

Exercise 4: Design your own 2B3B line code by choosing the output waveforms that have the lowest average DC value and giving preference to those that start and end at different levels (assume bipolar signalling).

Block codes can be more efficient because the ratio of output to input bits can be low (10/8, for example). But this comes at the cost of additional complexity for encoding, decoding and detecting the boundaries of the blocks. Synchronization is usually done by using output bit patterns that do not correspond to any possible input data sequence.

Violations

Some line codes allow for the encoding algorithm to be violated to indicate infrequent events such as the start of a frame. A typical example is a bipolar code that transmits two adjacent bipolar pulses using the same (instead of alternating) polarity.

In other cases violation of the line coding rules indicates an error. This can be useful for detecting problems at the physical layer (we will study other techniques that are used at higher layers).

Properties of Line Codes

Bandwidth

Line codes determine the bandwidth required to transmit the signal. It is often desirable to minimize the bandwidth of the baseband signal.

One reason is to maximize the symbol rate that can be transmitted over a channel with limited bandwidth.

Even when ISI due to limited bandwidth is not a concern, another reason to limit the bandwidth of signals is that higher frequencies in the spectrum of the signal radiate more easily than low-frequency components and are more likely to cause crosstalk and interference.

However, the spectrum of the line-coded signal depends not just on the waveforms (symbols) used, but also on the way these waveforms follow each other.

Low-Frequency Content

In many cases transformers are used by baseband channels for impedance matching or protection purposes. In these cases low frequencies (typically below a few 100 Hz) are greatly attenuated. For these types of channels it is useful if the line code has little frequency content at low frequencies since this will minimize distortion of the waveform.

DC Balance

Line codes can be designed so that the average voltage of the waveform (its DC value) is zero. This is different than having low low-frequency content. For example an NRZ signal with voltages balanced about zero will have significant low frequency content but zero average voltage.

Maximum Time Between Transitions

Digital systems use a periodic signal called a clock to synchronize the transfer of data between devices. Few communications systems transmit a clock signal along with their data because it would require an additional channel (an extra twisted pair, for example). The receiver thus has to recover the clock from the received data waveform.

Both the transmitter and receiver know the approximate clock frequency so the clock recovery problem is limited to correcting for an the accumulated phase difference between the clocks due to a relatively small frequency difference.

Modern receiver and transmitter clocks are controlled by quartz crystal oscillators with accuracies of tens of ppm (parts per million).

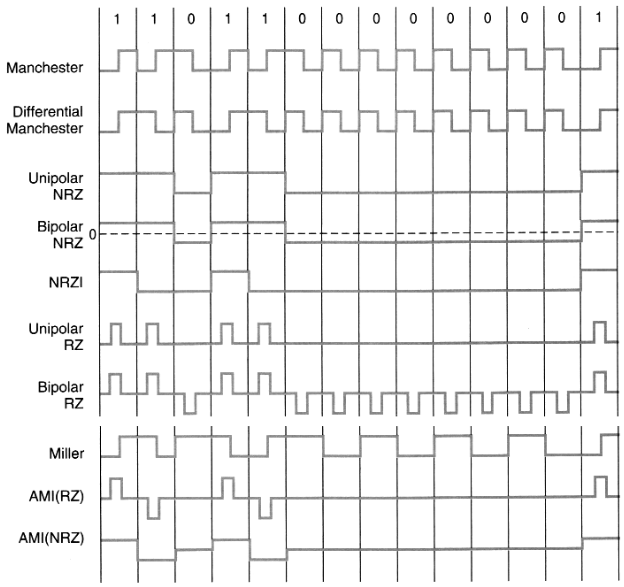
Exercise 5: A link operates at 100 Mb/s. What is the bit period? The transmitter and receiver have independent clocks (oscillators) with accuracies of 100ppm. What is the maximum difference between the two clock periods in ppm? In seconds?

The timing error due to a frequency (period) difference accumulates over time. How many bits will it take for the accumulated error to equal 10% of the clock period?

Receivers look at the transitions between signal levels to help estimate the phase (or timing) error between their clocks. They typically average these estimates to correct the point at which the receiver samples the received signal. More signal transitions provide more information for the receiver to resynchronize its clock.

Commonly-Used Line Codes

This section describes some common or representative codes. The figure above shows some of these codes and is from the text by Hioki:



NRZ - Non-Return to Zero

The two waveforms are two different voltage levels with durations equal to the bit period.

NRZ is the line “code” typically used between logic devices and is also used for short links. Short means short relative to the propagation time.

Exercise 6: A data link operates over a distance of 10m at 100 kb/s. If the velocity factor of the cable is 0.66, what is the propagation delay in microseconds? What fraction of the bit period does this represent?

NRZ can use differential encoding so that a 1 is encoded as a change in level and 0 is encoded by no change.

For example, the 12 Mb/s USB protocol uses differential NRZ. A change in voltage represents a ‘1’, no change represents a ‘0’. To avoid long periods without transitions a ‘1’ bit is bit-stuffed after 5 consecutive 0’s.

RZ

The waveforms in the RZ line code return to zero in the second half of each bit interval.

The first part of each waveform (1 or 0) can be encoded as unipolar or bipolar. The first part can also be encoded differentially if desired.

RZ ensures a transition in every bit if bipolar encoding is used. The disadvantage is the increase in bandwidth.

Manchester

A ‘1’ is transmitted as a waveform that goes from low to high half-way into the bit period. A ‘0’ is transmitted as a waveform that goes from high to low half-way into the bit period.

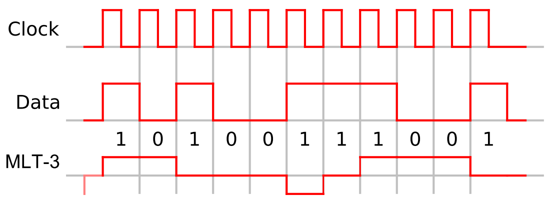
Manchester coding can also be transmitted differentially. A ‘1’ is encoded by using a different waveform than was used for the previous bit and a ‘0’ is encoded by the same waveform as the previous bit.

Manchester coding is used by 10Mb/s Ethernet (10BaseT).

MLT-3

There are three voltage levels. Each ‘1’ bit switches to the next level. When the highest or lowest level is reached the direction reverses. This produces something that looks like a sine wave and thus has a compact spectrum.

This is the line code used for 100 Mb/s Ethernet (100BaseTX). The following example is from Wikipedia:



4B5B

This code converts a sequence of 4 bits into a sequence of 5 bits.

Since there are 16 input sequences and 32 output sequences, certain problematic output sequences can be omitted (not used). Those sequences include those that do not have transitions and those that have more than a certain number of consecutive 1’s or 0’s.

The advantage of 4B5B is that only a relatively small increase in bandwidth is required while still ensuring low DC content and a minimum transition density.

This code is used for 100 Mb/s Ethernet (100BaseTX) (before MLT-3 encoding).

The table below, taken from the IEEE 802.3 (Ethernet) standard, shows the 4B5B encoding:

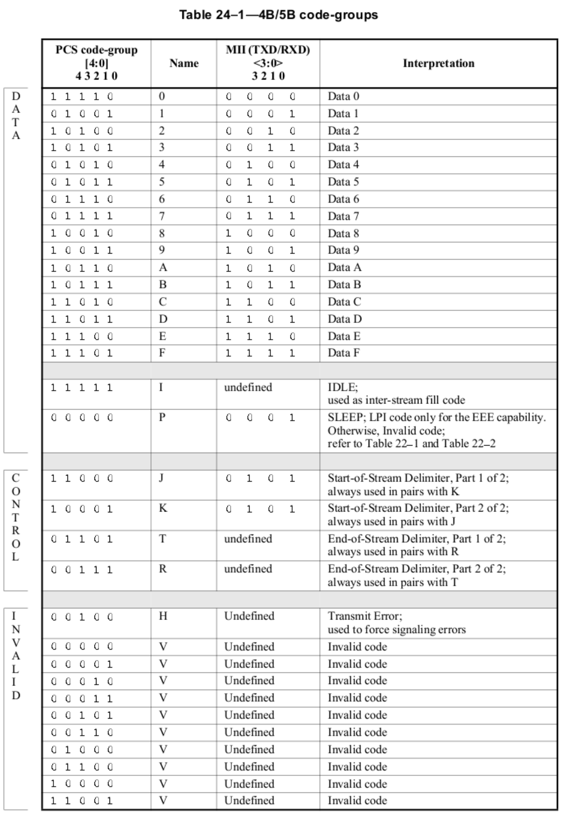


Table 24–1—4B/5B code-groups

	PCS code-group [4:0] 4 3 2 1 0	Name	MII (TXD/RXD) <3:0> 3 2 1 0	Interpretation
DATA	1 1 1 1 0	0	0 0 0 0	Data 0
	0 1 0 0 1	1	0 0 0 1	Data 1
	1 0 1 0 0	2	0 0 1 0	Data 2
	1 0 1 0 1	3	0 0 1 1	Data 3
	0 1 0 1 0	4	0 1 0 0	Data 4
	0 1 0 1 1	5	0 1 0 1	Data 5
	0 1 1 1 0	6	0 1 1 0	Data 6
	0 1 1 1 1	7	0 1 1 1	Data 7
	1 0 0 1 0	8	1 0 0 0	Data 8
	1 0 0 1 1	9	1 0 0 1	Data 9
	1 0 1 1 0	A	1 0 1 0	Data A
	1 0 1 1 1	B	1 0 1 1	Data B
	1 1 0 1 0	C	1 1 0 0	Data C
	1 1 0 1 1	D	1 1 0 1	Data D
	1 1 1 0 0	E	1 1 1 0	Data E
	1 1 1 0 1	F	1 1 1 1	Data F
	1 1 1 1 1	I	undefined	IDLE; used as inter-stream fill code
	0 0 0 0 0	P	0 0 0 1	SLEEP; LPI code only for the EEE capability. Otherwise, Invalid code; refer to Table 22–1 and Table 22–2
CONTROL	1 1 0 0 0	J	0 1 0 1	Start-of-Stream Delimiter, Part 1 of 2; always used in pairs with K
	1 0 0 0 1	K	0 1 0 1	Start-of-Stream Delimiter, Part 2 of 2; always used in pairs with J
	0 1 1 0 1	T	undefined	End-of-Stream Delimiter, Part 1 of 2; always used in pairs with R
	0 0 1 1 1	R	undefined	End-of-Stream Delimiter, Part 2 of 2; always used in pairs with T
INVALID	0 0 1 0 0	H	Undefined	Transmit Error; used to force signaling errors
	0 0 0 0 0	V	Undefined	Invalid code
	0 0 0 0 1	V	Undefined	Invalid code
	0 0 0 1 0	V	Undefined	Invalid code
	0 0 0 1 1	V	Undefined	Invalid code
	0 0 1 0 1	V	Undefined	Invalid code
	0 0 1 1 0	V	Undefined	Invalid code
	0 1 0 0 0	V	Undefined	Invalid code
	0 1 1 0 0	V	Undefined	Invalid code
	1 0 0 0 0	V	Undefined	Invalid code
	1 1 0 0 1	V	Undefined	Invalid code

Exercise 7: How would the bit sequence 0110 be encoded using 4B5B followed by MLT3 assuming the starting level is 0V?

Spectra

The following figure shows the approximate spectrums (power spectral densities, in W/Hz) of three line codes. The power of an NRZ-encoded signal is maximum at DC and the first null is at the bit (or symbol) rate, *R*. The power of a bipolar RZ signal using AMI is zero at DC and peaks at half the symbol rate. A Manchester-coded signal also has no DC content and little power and low frequencies but has a wider first-null bandwidth than NRZ.