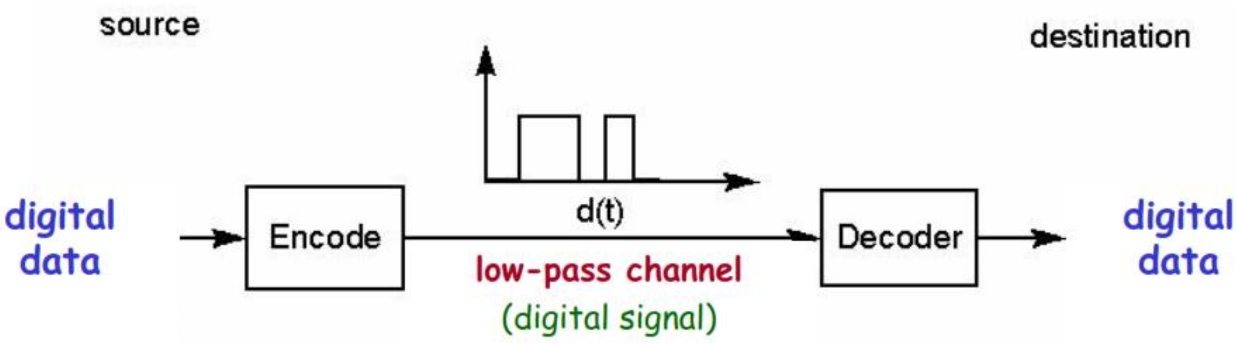
Line Encoding

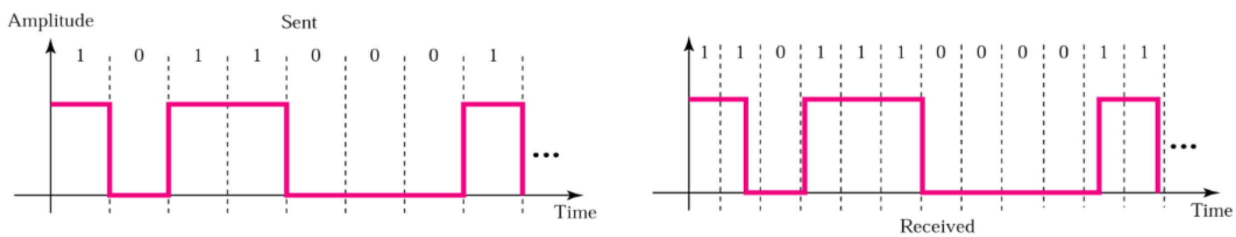
- The assignment of pulses (an electrical signal) to the binary digits that come out of the PCM or DPCM system.
- Line coding encodes the bit stream for transmission through a line, or a cable.
- It is used for communications between the CPU and peripherals, and for short-distance baseband communications, such as the Ethernet.



Two Design Considerations

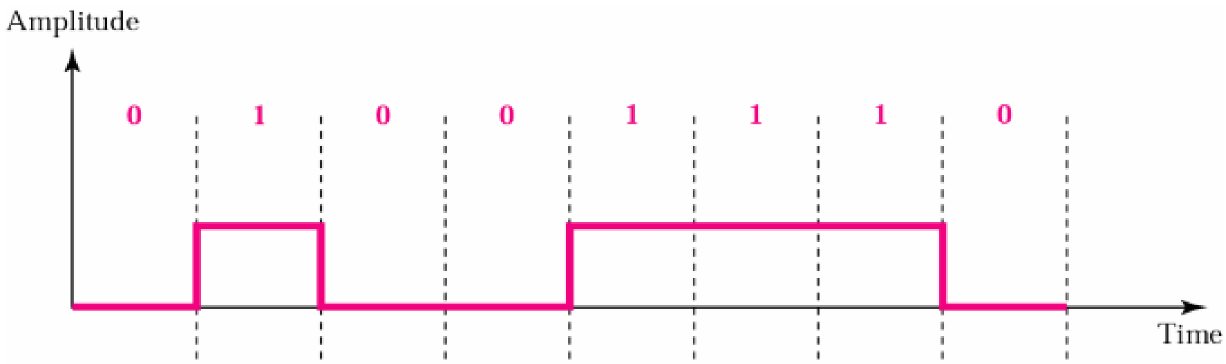
- **DC Component in Line Coding:** Some line coding schemes have a residual (DC) component, which is generally undesirable.
  - Transformers do not allow passage of DC component.
  - DC component ⇒ extra energy – useless!

- **Self-Synchronization (clocking):** To correctly interpret signal received from sender, receiver's bit interval must exactly correspond to sender's bit intervals
  - If receiver clock is faster/slower, bit intervals not matched ⇒ receiver misinterprets signal
  - Self-synchronizing digital signals include timing information in itself, to indicate the beginning & end of each pulse



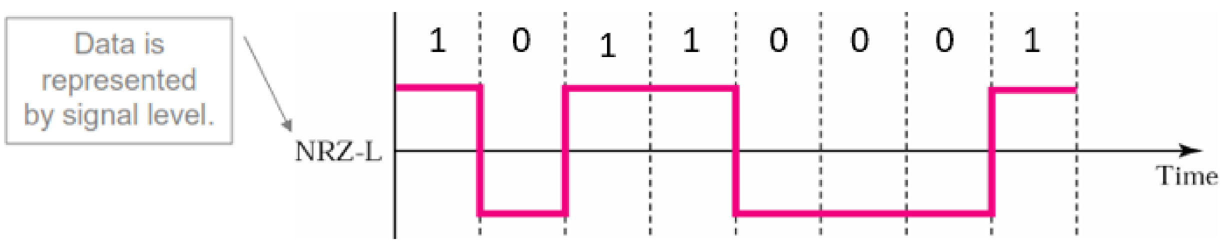
Unipolar Line Coding (Unipolar non-return to zero)

- Uses only one non-zero and one zero voltage level to represent binary digits 1 and 0
- Simple to implement, but obsolete due to two main problems:
  - Presence of a DC component.
  - Lack of synchronization for long series of 1-s or 0-s



Polar Line Coding: Polar non-return to zero

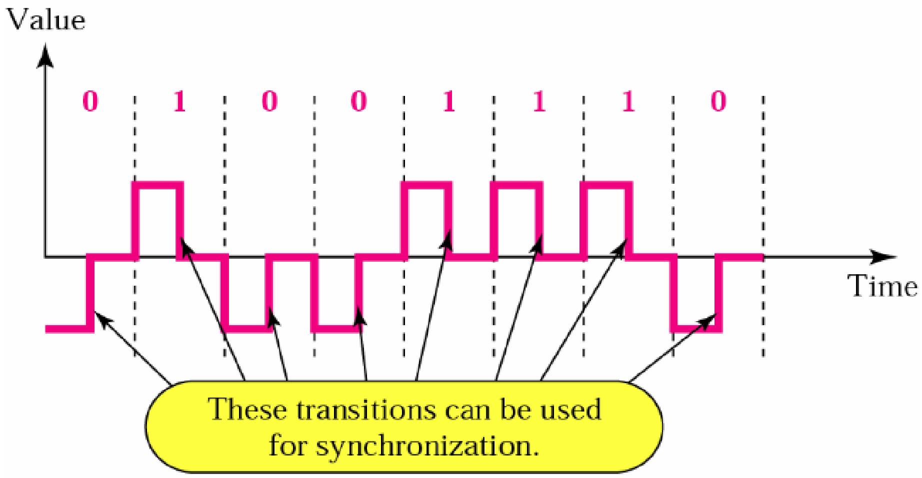
- Uses two non-zero voltage level to represent digits 1 and digit 0. +ve for 1 and –ve for 0
- No DC component is present
- Poor synchronization for long series of 1-s or 0-s



Polar Non-return to Zero

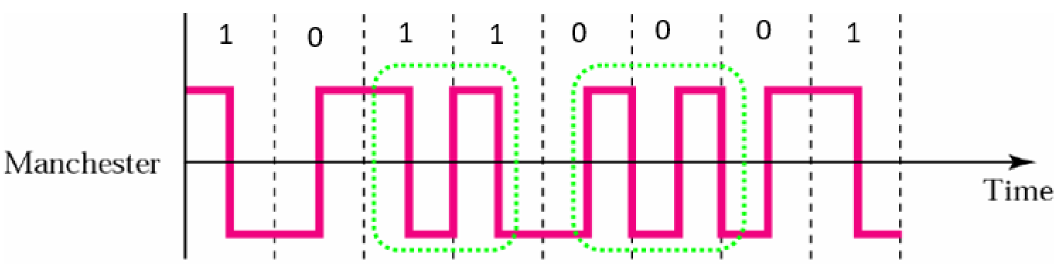
Polar Line Coding: Polar return to zero

- Uses two non-zero voltage level to represent digits 1 and digit 0. +ve for 1 and –ve for 0. **Must** return to zero halfway through each bit interval.
- No DC component is present.
- Perfect synchronization for long series of 1-s or 0-s.
- Twice the bandwidth required for polar non-return to zero, $B.W \propto \frac{1}{pulse\ width}$.



Manchester Line Coding

- Inversion at the middle of each bit interval is used for both synchronization and bit representation
- Digit 0 ⇒ neg-to-pos transition, 1 ⇒ pos-to-neg transition
- Perfect synchronization for long series of 1-s or 0-s
- There is always transition at the middle of the bit, and maybe one transition at the end of each bit.
- Fine for alternating sequences of bits (10101), but wastes bandwidth for long runs of 1-s or 0-s.
- Used by IEEE 802.3 (Ethernet).
- No DC component is present.
- Twice the bandwidth required for polar non-return to zero. Two pulses are used to represent one bit.

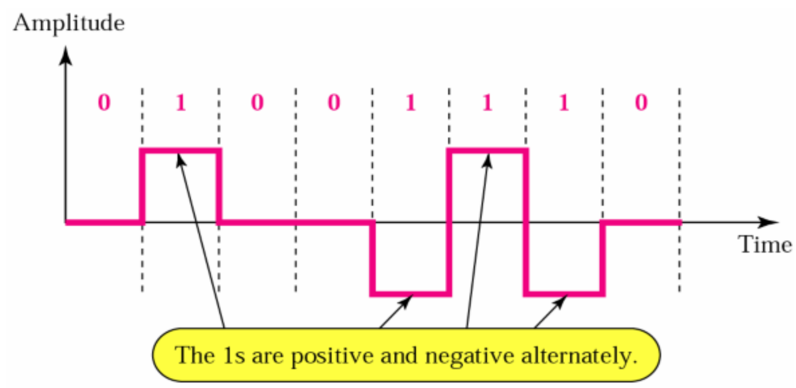


Manchester Encoding

Bipolar Line Coding

- Uses two non-zero and zero voltage level for representation of two data levels.
- **0 = zero level; 1 = alternating positive and negative level.**

- If first bit 1 is represented by a positive amplitude, second will be represented by negative amplitude, third by positive, etc.
- Less bandwidth required than with Manchester coding (for any sequence of bits).
- Loss of synchronization is possible for long runs of 0-s.
- No DC component is present.



2B1Q (2 Bipolar to 1Quaternary) Line Coding

- Data patterns of size 2 bits are encoded as one signal element belonging to a four-level signal.
- Data is sent two time faster than with polar non-return to zero.
- Receiver has to discern 4 different thresholds

Binary Input	Output Voltage
00	-3
01	-1
10	1
11	3