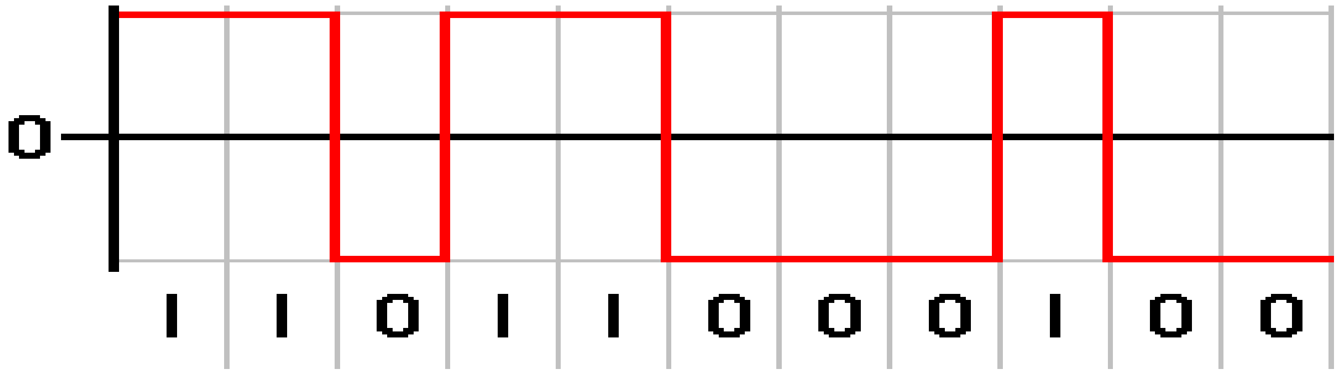
Signal Encoding

NRZ



In telecommunication, a non-return-to-zero (NRZ) line code is a binary code in which 1's are represented by ones ignificant condition (usually a positive voltage) and 0's are represented by some other significant condition (usually a negative voltage), with no other neutral or rest condition. The pulses have more energy than a RZ code. Unlike RZ, NRZ does not have a rest state. NRZ is not inherently a self-synchronizing code, so some additional synchronization technique (for example a run length limited constraint, or a parallel synchronization signal) must be used to avoid bit slip.

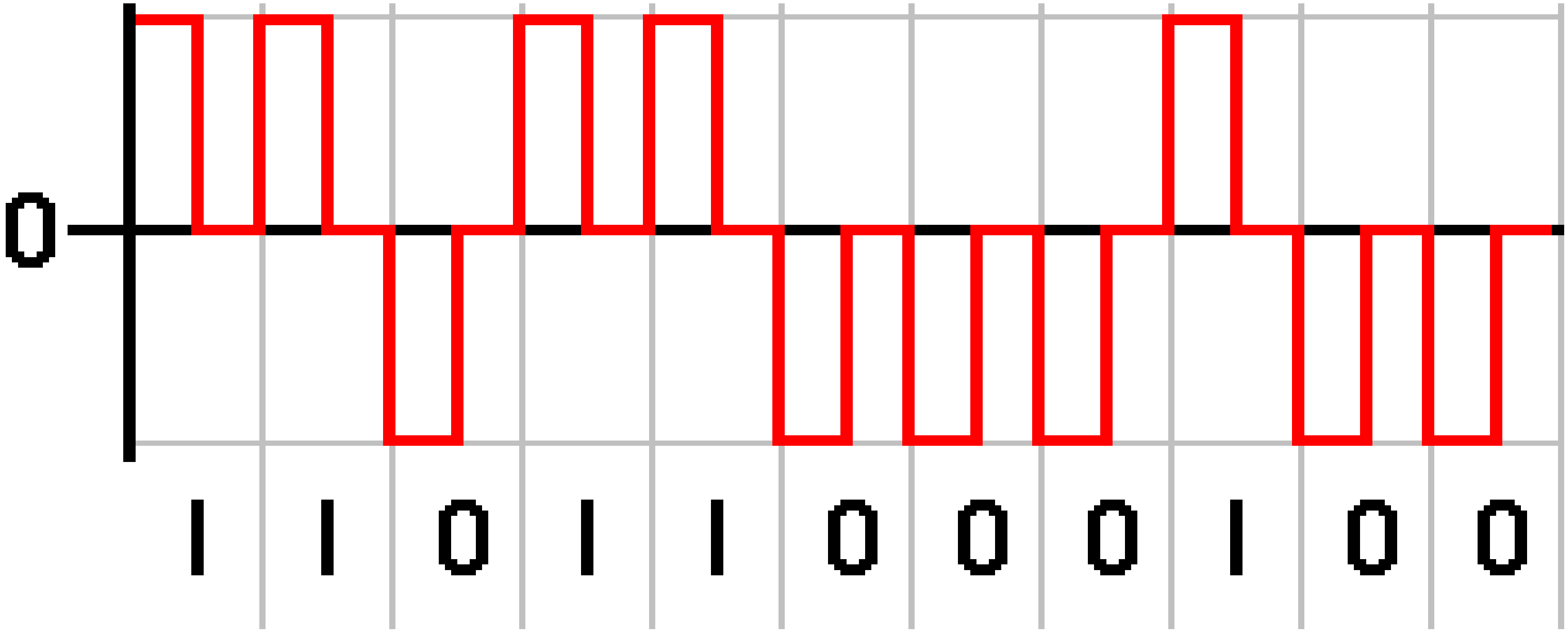
For a given data signaling rate, i.e., bit rate, the NRZ code requires only half the bandwidth required by theManchester code.

When used to represent data in an asynchronous communication scheme, the absence of a neutral state requires other mechanisms for bit synchronization when a separate clock signal is not available.

NRZ-Level itself is not a synchronous system but rather an encoding that can be used in either a synchronous or asynchronous transmission environment, that is, with or without an explicit clock signal involved. Because of this, it is not strictly necessary to discuss how the NRZ-Level encoding acts "on a clock edge" or "during a clock cycle" since all transitions happen in the given amount of time representing the actual or implied integral clock cycle. The real question is that of sampling--the high or low state will be received correctly provided the transmission line has stabilized for that bit when the physical line level is sampled at the receiving end.

However, it is helpful to see NRZ transitions as happening on the trailing (falling) clock edge in order to compare NRZ-Level to other encoding methods, such as the mentioned Manchester code, which requires clock edge information (is the XOR of the clock and NRZ, actually) and to see the difference between NRZ-Mark and NRZ-Inverted.

RZ



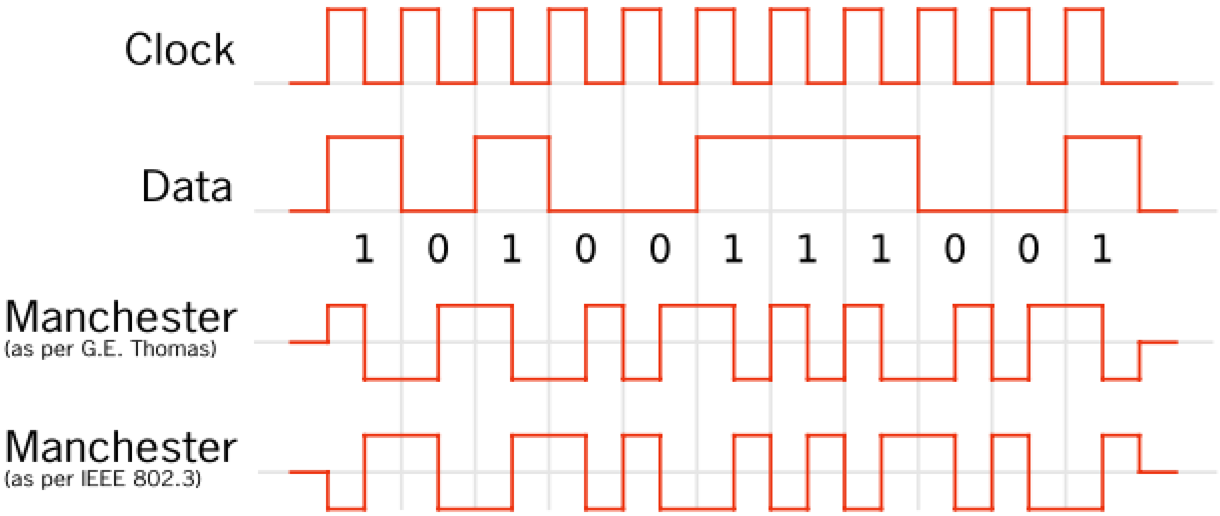
Return-to-zero (RZ) describes a line code used in telecommunications signals in which the signal drops (returns) to zero between each pulse. This takes place even if a number of consecutive 0's or 1's occur in the signal. The signal is self-clocking. This means that a separate clock does not need to be sent alongside the signal, but suffers from using twice the bandwidth to achieve the same data-rate as compared to non-return-to-zero format.

The "zero" between each bit is a neutral or rest condition, such as a zero amplitude in pulse amplitude modulation(PAM), zero phase shift in phase-shift keying (PSK), or mid-frequency in frequency-shift keying (FSK). That "zero" condition is typically halfway between the significant condition representing a 1 bit and the other significant condition representing a 0 bit.

Although return-to-zero (RZ) contains a provision for synchronization, it still has a DC component resulting in “baseline wander” during long strings of 0 or 1 bits, just like the line code non-return-to-zero.

[edit]

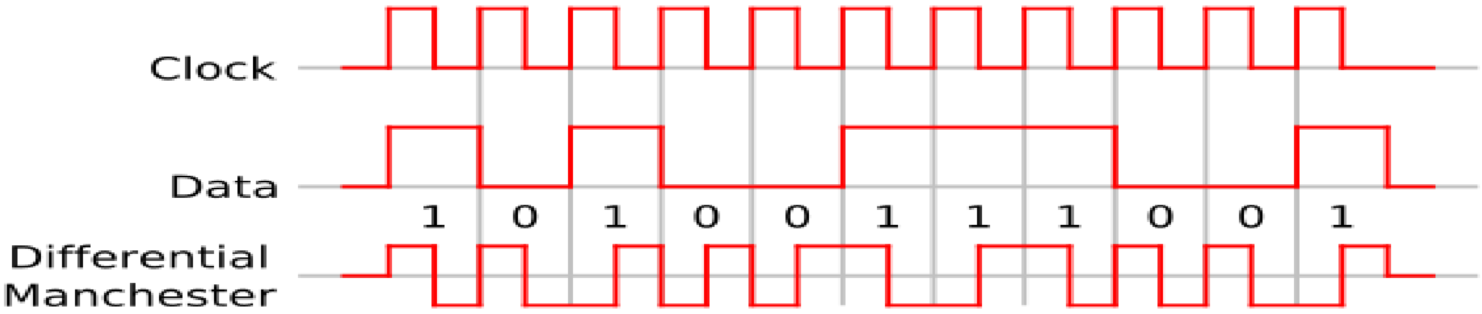
Manchester



Each bit is transmitted in a fixed time (the "period").
A 0 is expressed by a low-to-high transition, a 1 by high-to-low transition (according to G.E. Thomas' convention -- in the IEEE 802.3 convention, the reverse is true).
The transitions which signify 0 or 1 occur at the midpoint of a period.
Transitions at the start of a period are overhead and don't signify data.
Manchester code always has a transition at the middle of each bit period and may (depending on the information to be transmitted) have a transition at the start of the period also. The direction of the mid-bit transition indicates the data. Transitions at the period boundaries do not carry information. They exist only to place the signal in the correct state to allow the mid-bit transition. The existence of guaranteed transitions allows the signal to be self-clocking, and also allows the receiver to align correctly; the receiver can identify if it is misaligned by half a bit period, as there will no longer always be a transition during each bit period. The price of these benefits is a doubling of the bandwidth requirement compared to simpler NRZ coding schemes (or see also NRZI).
In the Thomas convention, the result is that the first half of a bit period matches the information bit and the second half is its complement.

```
original data XOR clock = Manchester value 0 0 0 0 1 1 1 0 1 1 1 0
```
Summary:

Differential Manchester



Differential Manchester encoding, also called biphase mark code (BMC) or FM1, is a line code in which data and clock signals are combined to form a single 2-level self-synchronizing data stream. It is a differential encoding, using the presence or absence of transitions to indicate logical value. It has the following advantages over some other line codes:
A transition is guaranteed at least once every bit, allowing the receiving device to perform clock recovery.
Detecting transitions is often less error-prone than comparing against a threshold in a noisy environment.
Unlike with Manchester encoding, only the presence of a transition is important, not the polarity. Differential coding schemes will work exactly the same if the signal is inverted (wires swapped). (Other line codes with this property include NRZI, bipolar encoding, coded mark inversion, and MLT-3 encoding).
If the high and low signal levels have the same voltage with opposite polarity, coded signals have zero average DC voltage, thus reducing the necessary transmitting power and minimizing the amount of electromagnetic noise produced by the transmission line.
The symbol rate is twice the bitrate of the original signal. Each bit period is divided into two half-periods: clock and data. The clock half-period always begins with a transition from low to high or from high to low. The data half-period makes a transition for one value and no transition for the other value. One version of the code makes a transition for 0 and no transition for 1 in the data half-period; the other makes a transition for 1 and no transition for 0. Thus, if a "1" is represented by one transition, then a "0" is represented by two transitions and vice versa, making Differential Manchester a form of frequency shift keying. Either code can be interpreted with the clock half-period either before or after the data half-period.

Differential Manchester (Example 2)