

What's the Difference Between NRZ, NRZI, and Manchester Encoding?

NRZ, NRZI, and Manchester are popular serial encoding mechanisms. Find out how they differ from each other.

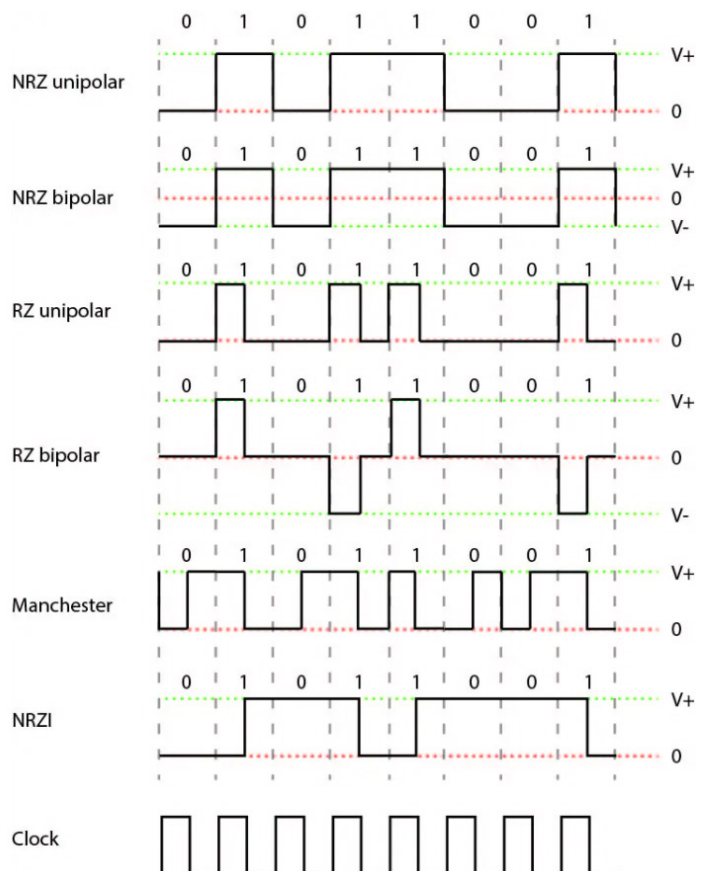
Non-return-to-zero (NRZ), non-return-to-zero-inverted (NRZI), return-to-zero (RZ), and Manchester are popular serial encoding mechanisms (see figure). Each has different characteristics that make it useful in different applications. Those dealing with UARTs will find NRZ to be the most familiar.

There is also a unipolar and bipolar version for many of these encodings. Unipolar has a transition between a zero and positive. The actual measurement can be one of many types of attributes from like voltage, current, pressure, or optical. A bipolar system has a transition between a positive and negative. Any method can employ a bipolar encoding but logically they may be the same as shown with the NRZ example.

The figure shows signals synchronized with a clock at the bottom. The types and locations of the transitions are what make the encoding mechanisms different.

NRZ has a level the matches the logical signal. In this case the clock would latch the value being transmitted at the start of each clock cycle and it would be sampled at the receiving end in the middle of the clock cycle. NRZ is commonly used with serial ports. The receiver normally has a faster clock that is synchronized with a rising or falling edge of the data transmission. This allows the determination of the center of the clock cycle to be more accurately determined (Fig. 1).

The clock highlights the different transition methods. NRZ is level sensitive related to the values being sent. NRZ has a transition for a 1 value with respect to the clock. Manchester and NRZI have transitions based on the value. Manchester has a falling transition for a 1, while NRZI has a transition for a 1.



The clock highlights the different transition methods. NRZ is level sensitive related to the values being sent. RZ has a transition for a 1 value with respect to the clock. Manchester and NRZI have transitions based on the value. Manchester has a falling transition for a 1, while NRZI has a transition for a 1.

transitions based on the value. Manchester has a falling transition for a 1, while NRZI has a transition for a 1.

RZ uses pulses at the start of the clock cycle to indicate a 1 value. In the bipolar version, sequential 1 values have a transition in the opposite directions. RZ will always have a transition when there are 1 values in the data.

Manchester and NRZI encodings have transitions in the middle of the clock cycle, with the type depending on the encoding. Manchester is an NRZ encoding that is exclusively-ORed with the clock. This provides at least one transition per bit. NRZI also uses a transition in the middle of the clock cycle, but this only occurs when there is a 1 value. Manchester makes clock recovery easier.

The challenge with Manchester and RZ is the frequency of the data transmission is twice that as NRZ and NRZI. The tradeoffs depend upon the mechanisms used to implement the communication. Sometimes bandwidth can be an issue.

These encoding schemes are just the starting point used for most data transmission systems. For example, universal asynchronous receiver-transmitters (UART) map an asynchronous per/byte sequence that includes a start bit and one or more stop bits. A parity bit can be included to assist in error detection. Higher level protocols often forego the parity bit and compute parity or a CRC over a data packet. This helps reduce the overhead for error detection.

Synchronous systems typically have an idle character that can be recognized in addition to a clock recovery/synchronization scheme. Encoding schemes like Manchester encoding make this easier to implement. The synchronous support often uses encoding schemes like 4B/5B and 8B/10B. The 4B/5B uses 5 bits to encode 16 (4-bit) symbols that are unique enough that clock recovery/synchronization is possible using encodings like NRZ.

There is a bandwidth penalty with these higher level encoding schemes. The larger the collection, the lower the overhead but they require tighter timing and more complex synchronization logic. The approach scales and 64B/66B is used with 10 Gbit Ethernet.