4B/5B

The four binary/five binary (4B/5B) coding scheme was designed to be used in combination with NRZ-I. Recall that NRZ-I has a good signal rate, one-half that of the biphase, but it has a synchronization problem. A long sequence of as can make the receiver clock lose synchronization. One solution is to change the bit stream, prior to encoding with NRZ-I, so that it does not have a long stream of as. The 4B/5B scheme achieves this goal. The block-coded stream does not have more that three consecutive as, as we will see later. At the receiver, the NRZ-I encoded digital signal is first decoded into a stream of bits and then decoded to remove the redundancy.

In 4B/5B, the 5-bit output that replaces the 4-bit input has no more than one leading zero (left bit) and no more than two trailing zeros (right bits). So when different groups are combined to make a new sequence, there are never more than three consecutive as. (Note that NRZ-I has no problem with sequences of Is.) Table shows the corresponding pairs used in 4B/5B encoding. Note that the first two columns pair a 4-bit group with a 5-bit group. A group of 4 bits can have only 16 different combinations while a group of 5 bits can have 32 different combinations. This means that there are 16 groups that are not used for 4B/5B encoding. Some of these unused groups are used for control purposes; the others are not used at all. The latter provide a kind of error detection. If a 5-bit group arrives that belongs to the unused portion of the table, the receiver knows that there is an error in the transmission.

4B/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 |  |  |

4B/5B encoding solves the problem of synchronization and overcomes one of the deficiencies of NRZ-1. However, we need to remember that it increases the signal rate of NRZ-1. The redundant bits add 20 percent more baud. Still, the result is less than the biphase scheme which has a signal rate of 2 times that of NRZ-1. However, 4B/5B block encoding does not solve the DC component problem of NRZ-1. If a DC component is unacceptable, we Need to use biphase or bipolar encoding.

4-bit blocks

5-bit blocks

8RIlOR:

The eight binary/ten binary (SBIlOB) encoding is similar to 4B/5B encoding except that a group of 8 bits of data is now substituted by a lO-bit code. It provides greater error detection capability than 4B/5B. The 8BIlOB block coding is actually a combination of 5B/6B and 3B/4B encoding, The most five significant bits of a 10-bit block is fed into the 5B/6B encoder; the least 3 significant bits is fed into a 3B/4B encoder. The split is done to simplify the mapping table. To prevent a long run of consecutive Os or Is, the code uses a disparity controller which keeps track of excess Os over Is (or Is over Os). If the bits in the current block create a disparity that contributes to the previous disparity (either direction), then each bit in the code is complemented (a 0 is changed to a 1 and a 1 is changed to a 0). The coding has 210 - 28 =768 redundant groups that can be used for disparity checking and error detection. In general, the technique is superior to 4B/5B because of better built-in error-checking capability and better synchronization.

8B/10B encoder

R8ZS:

Bipolar with S-zero substitution (BSZS) is commonly used in North America. In this technique, eight consecutive zero-level voltages are replaced by the sequence OOOVBOVB. The V in the sequence denotes violation; this is a nonzero voltage that breaks an AMI rule of encoding (opposite polarity from the previous). The B in the sequence denotes bipolm; which means a nonzero level voltage in accordance with the AMI rule. There are two cases, Note that the scrambling in this case does not change the bit rate. Also, the technique balances the positive and negative voltage levels (two positives and two negatives), which means that the DC balance is maintained. Note that the substitution may change the polarity of a 1 because, after the substitution, AMI needs to follow its rules.One more point is worth mentioning. The letter V (violation) or B (bipolar) here is relative. The V means the same polarity as the polarity of the previous nonzero pulse; B means the polarity opposite to the polarity of the previous nonzero pulse.

AMI used with scrambling:

HDB3:

High-density bipolar 3-zero (HDB3) is commonly used outside of North America. In this technique, which is more conservative than B8ZS, four consecutive zero-level voltages are replaced with a sequence of OOOV or BOO\: The reason for two different substitutions is to maintain the even number of nonzero pulses after each substitution. The two rules can be stated as follows:

1. If the number of nonzero pulses after the last substitution is odd, the substitution pattern will be OOOV, which makes the total number of nonzero pulses even.

2. If the number of nonzero pulses after the last substitution is even, the substitution pattern will be BOOV, which makes the total number of nonzero pulses even. There are several points we need to mention here. First, before the first substitution, the number of nonzero pulses is even, so the first substitution is BODY. After this substitution, the polarity of the 1 bit is changed because the AMI scheme, after each substitution, must follow its own rule. After this bit, we need another substitution, which is OOOV because we have only one nonzero pulse (odd) after the last substitution. The third substitution is BOOV because there are no nonzero pulses after the second substitution (even).

First second Third

Substitution substitution substitution