**Explanations**

**NRZ-L**

Contains two levels, 0 = high level and 1 = low level.

When the input bit is 1 the output should be a -5.

If it’s a 0, it should be 5.

**NRZI**

Contains two levels, 0 = no transition at the beginning of the interval and 1 = transition at the beginning of interval.

When the input is 1 the output transitions to the alternate voltage.

**Bipolar-AMI**

Three levels, 0 = no line signal and 1 = positive or negative level, it alternates for successive ones.

When the input bit is 0, the output should be ‘neutral’ or at the zero level. If the input bit is 1, the output should alternate from the previous 1 bit. For instance, if the first bit is 1 then say we place it at the 5 level. But if the 2nd bit is another 1, the output should alternate to a -5.

**Manchester**

Contains two levels, 0 = transition from high to low in middle of interval and 1 = transition from low to high in middle of interval.

If the input bit is 1, then the output should alternate from -5 to 5 in the middle of the bit interval. If it is a 0 then it should alternate in the middle of the bit interval from 5 to -5.

**D-Manchester**

Contains two levels, it always transitions in the middle of the interval. 0 = transition at the beginning of interval and 1 = no transition at the beginning of interval.

If the input bit is 0 then the output should transitions in the beginning of the bit interval. If it is 1, then the output alternates in the middle of the bit interval.

**Summary**

The program starts by opening the input data using fstream. Then it asks the user which encoding scheme he/she wants to use. Using a switch case and depending on the user’s decision the program passes the line from the input into the respective function.

The code creates an array of size 8 called enc. This was because the input size is 8 bits. This array is used for the output, and depending the explanation (discussed earlier of the scheme), the output will be given. The encoded output was relatively straight forward for NRZ-L and NRZI with NRZI only requiring a simple modification of the NRZ-L code.

*Special Cases:*

For the B-AMI, I created a counter to save the previous bit output. This was used for an input bit of 0 which required for the output to remain the same from the previous interval.

For Manchester, I created another array called “trans” because this scheme required a transition in the middle of the bit interval. In order to print this I first printed the encoded output (enc array) and then it’s middle interval transition (trans array).

D-Manchester was an alternative of Manchester and only required for a modification in the Manchester function’s code.