1 Error Detection
CIS748 Class Notes

During transmission, signals are generally subjected to attenuation, distortion, and noise. In order to ensure an error free transmission, we need a system that will allow us to detect errors when they occur. We are not talking about error correction at this time; just error detection.

There are two primary errors that we have to worry about: single-bit errors, and burst errors. In serial communications, burst errors are fairly common—if for example there is a 0.01 second distortion on a 1200 bps connection, then about 12 bits may be corrupted.

Error detection usually involves calculating something with the input, lets say N, and attaching that to the data sent. On the receiving end, the same exact calculation is carried about, and the result is compared to the sent N. If the values match, then data is assumed to be error free.

The added redundancy (the stuff we add to the sent message), can take several forms.

1.1 VRC: Vertical Redundancy Check

These take the form of a simple parity check. To every data unit (a byte, or more), we attach a bit to make the total count of bits even (or odd). If we make number of bits even, then it's called even parity, etc., otherwise it's called odd parity.

1.2 LRC: Longitudinal Redundancy Check

This is similar to VRC, except instead of taking parity of each data unit individually, we arrange the data units (bytes) in a table, and take parity vertically across that table. Same idea as with even/odd parity applies.

1.3 CRC: Cyclic Redundancy Check

The CRC does something similar to parity, except it ensures that the number is divisible by some pre-determined value. A divisor is chosen, the data is then divided, the remainder is attached to the data to be sent.

When the receiver gets the data, it divides the data by the same exact divisor, and looks at the remainder. If remainder is zero, then it is assumed there was no corruption.