DIGITAL DATA IN CABLE SYSTEMS: THE BASICS OF ERROR CONTROL

Victor A. Nowakowski
Senior Systems Engineer

Oak Communications Inc.
Satellite Systems Division

ABSTRACT

Analog video and audio have long been the mainstream signals in cable systems. With the appearance of scrambled programming and addressable subscriber decoders, cable systems silently entered the digital world.

As in any communications system, the cable networks must contend with and compensate for imperfect or noisy signals. The analog world can tolerate noise much easier than the digital world. Moreover, analog methods for digital noise filtering are ineffective against errors in digital data channels. This paper examines digital error control, its effects on customers and equipment, and presents an overview of a few techniques used by equipment designers to improve data reliability and system performance.

1.0 INTRODUCTION

The cable system of yesterday was a distributor of simple analog video and audio programs to domestic subscribers unable to receive direct television broadcasts. The cable system of today is a sophisticated multi-channel network whose operators offer a number of programming packages and services to a larger and much more diverse subscriber population.

Although communication satellites have been with us since the 1960s, it has taken nearly twenty years to design and build affordable technology that could link local cable system networks with nationwide broadcast satellites. Cable television now reaches not only into the rural market, but into the urban and fast growing suburban markets as well.

The marketplace today offers a wide variety of television programming that includes entertainment, news and information. To provide the subscriber with a variety of program packages and billing rates, as well as to contain and secure the product, programmers and cable operators installed scrambling equipment that, to varying degrees, made the video and audio unrecognizable.

Recently, scrambling equipment came between the programmer and the cable operator, in addition to scrambling equipment between the operator and subscribers. It was a natural step for the manufacturers of scrambling systems to consider innovative designs that have since changed the industry. Some of these design features are: encrypted digital high-fidelity multi-channel audio, addressable decoder control, and digital data channels.

The art of scrambling became a science. Assigning each decoder a unique address, or name, permitted selective decoder authorization which has lead to pay-per-view and other impulse mode products. Computerized decoder control and individual decoder attention is now the basis for most scrambling systems. All these necessarily depend upon digital remote control.

2.0 BASE-BAND TRANSPARENCY

So, it is not just analog anymore. Your cable systems and satellite receivers relay control information that is often inserted in the Vertical Blanking Interval (VBI) of the video base-band signal. Descrambling decoders and re-encoders are controlled by these digital pulses that are organized into meaningful messages which the decoding equipment understands. Analog audio is digitized and usually inserted in the Horizontal Blanking Interval (HBI) along with other information. These systems are described as being base-band transparent, i.e. all video, audio and related digital information resides within the allocated video bandwidth.

Sub-carriers are also a design option for digital pathways, but these are used at the expense of additional bandwidth and complexity. The same sub-carrier space might just as well be used for additional commercial channels and the capture of otherwise lost revenues. The same may be said of systems that require extended video bandwidths to pass all required control signals. From this point of view, base-band transparency, the ability to send digital information in the allocated video bandwidth, is the economical "best-buy" in terms of price/performance.

3.0 THE NOISE PROBLEM

What makes all this exciting is the one-way control traffic - we can tell a decoder what to do, but it will be the subscribers that call us whenever their decoders are in a most undesirable state. Decoders are "Listeners", i.e. they cannot talk back to their controller. In the presence of noise, however, just what is it the Decoders are listening too? This situation provides a design challenge to which there is no perfect solution, but a lot of thought and clever implementation has made the open-loop control problem a manageable one.

Noise is a corrupting influence. Depending upon its relative intensity, a message may be crystal clear or entirely unrecognizable. In many cases, noise has a subtle effect, not as much noticable as it is potentially irritating.

The analog world perceives noise as a "distortion" while the digital world may perceive noise as a "bit-bomber". Figure 1 illustrates this point. Analog signals

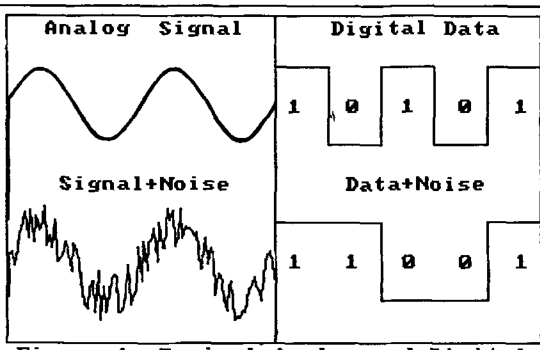

Figure 1. Typical Analog and Digital noise effects.

affected by additive noise exhibit amplitude and frequency component mutation. Most analog signals that we send are naturally smooth and continuous. But noisy analog signals can be treated with a well defined science that: (1) compensates for amplitude losses with increased gain, and (2) uses band-limiting filters that narrow the permissable frequency spectrum within which our signals appear. Other design factors, e.g. modulation, are also an important part in noise elimination, but gain and filtering are basic to analog noise reduction.

In contrast to analog, digital noise has a significantly different and potentially serious effect. First, digital signals are not smooth and continuous. They usually contain elements with discrete levels called bits, i.e. BInary digiTs. Each level is assigned a unique value or interpretation. Second, these bits are grouped together to represent predefined packets of information; the original information could be digitized analog such as voice and music, or computer generated command messages (e.g. "Decoder-#123-Authorized").

In "binary" or two-level digital systems, each bit takes on a value of either a "0" or a "1". Figure 1 shows an example of a transmitted digital signal with a value "10101". Imagine being a subscriber and that somewhere along the way to the decoder, impulse noise (like that from your neighbors' lawnmower) instantaneously alters two bits in this message that now is decoded as "11001". Suddenly, before your very eyes, your favorite movie has turned into a kalidescope of colors, zig-zags and rolling black lines. Your family may well ask, "what happened?" Without digital error control, this decoder may have translated the received command as a "Decoder-#123-Deauthorize" command and so it immediately stopped descrambling your movie (with no thanks to your neighbor, of course). Such a pity. Is this a case where the cable system gets the blame? Who would you call?

In short, analog noise can be tolerated and effectively minimized using classical design techniques. However, digital noise cannot be tolerated so it must, therefore, be error controlled but in a completely different manner.

4.0 ERROR CONTROL FUNDAMENTALS

The science of digital communications is well described mathematically and has been successfully implemented over the past two decades. Rather than dwell on information theory and esoteric concepts such as the Shannon Limit and channel capacity (which are best left to other references [1]), it is usefull instead to discuss some fundamental concepts used in digital error control.

4.1 The Error Control Rule

When casually discussing with friends the merits of any digital error control method, it is wise to remember two rules-of-thumb:

1. "Some is better than none."

2. "More is not necessarily better than less."

Although these rules may apply to many choice decisions, they especially apply to digital error control in cable system networks. The first rule recommends some type of effective error control. Just having "error detection" capability alone would have prevented the inadvertant deauthorization of Decoder-#123 as previously discussed. Advanced methods allow "error correction," i.e. actually recovering the original message even though is was received incorrectly.

Error detection and correction (EDC) methods vary widely; the simple methods are less expensive (of course) but have definite limitations. The more exotic methods prove to be very effective but resort to expensive overhead and processing demand. Levels of improvement are in terms of dB communication efficiency, i.e. a computed difference, in Signal/Noise per bit for a given bit-error rate, between the coded and uncoded messages. Without EDC there is 0 dB gain. Modest coding gains of 1 to 3 dB can be achieved using simple "block-type" codes; moderate coding gains of 4 to 7 dB and maximum coding gains of 8 dB or greater can be achieved using sophisticated "block-type" codes, "convolution-type" codes and "concatenation-type" methods [1].

The designer must match overall system performance criteria such as response time and low C/N survivability, with customer demand, end-product cost targets, development time and the nature of the communication channel itself. The second rule-of-thumb, therefore, applies to this evaluation and selection process.

4.2 The Noise Limitation Rule

C. E. Shannon was the founder of modern information theory. His work in 1948 and throughout the 1950s, together with a rapidly progressing technology, provided the foundation for all modern communication systems. An important idea he introduced stated that [1]:

> Noise does not limit the level of reliability of a communication channel, but it does set a limit on the rate of reliable information transfer.

This implied that the message rate, or rate of information transfer, was directly related to channel noise. The lower the noise, the greater the allowable message rate; inversely, the higher the noise, the slower the message rate to achieve reliable information transfer. If you send a message slow enough or often enough, the message will eventually get through. There are, as discussed previously, practical considerations that restrict these message rates in real systems. But, in general, this rule has lead to a wide variety of error control techniques that attempt to maximize information transfer within noisy communication channels.

5.0 BASIC ERROR CONTROL CONCEPTS

There are three elementary concepts in digital error control. These are:

1. redundancy
2. interleaving
3. parity

Each is briefly discussed below.

5.1 Redundancy

The first of these is redundancy. The first of these is redundancy. Redundant information is information that is repeated in the same or equivalent form, as demonstrated at the beginning of this paragraph. The Noise Limitation Rule suggested that with enough redundancy, any message could be reliably sent in any noisy channel. Indeed, this rule was well applied to space exploration satellites that relied heavily on redundant message transmission. And at the millions and billions of miles that separate the Earth with these far gone travelers, it seems like a miracle that tracking stations can still hear their whisper amongst the galactic noise within which they exist.

The obvious disadvantage of redundancy is the penalty of duplication. Decoding time, storage and effort are required for each message. If there are a lot of messages, then for a simple 2-of-3 redundant majority technique, the entire series of messages would take 3 times as long to send and decode as it would if the series were sent but once. (By the way, always send an odd number of repeats for majority voting decisions. A firm decision cannot be made with only an even number of choices - one says yes and one says no, which way do we go?)

5.2 Interleaving

In the pursuit of reliability, add-on error control codes necessarily increased the length of the original message. In modest- and moderate-gain methods, the amount of benefit was somewhat limited. If a massive noise pulse obscured more than one or two bits in a message, it was impossible to reconstruct the original information from the received noisy message.

Rather than allow a single message to receive the noise burst, why not spread the noise around? This at first might seem ludicrous - why corrupt a perfectly good message? The answer is explained as follows.

Burst noise has localized effects, i.e. bits tend to get corrupted in a short sequence corresponding to the actual noise burst interval. Where there is burst noise there are bit errors. Error detection and correction methods have a limited capability to detect and correct bits; usually one or two bits can be efficiently corrected in real-time systems. If the messages are organized such that each transmitted message is a collection of bits from two or more messages, then bit errors in the received message can be spread across more than one message when the information is reassembled into original form. Thus, bit errors per message are reduced.

This result is desireable since the number of bit errors per message is very small and can be readily detected and corrected with relatively straighforward and fast EDC methods. As an example, say that we must send two messages to a specific Decoder (see Figure 2):

Message #1: ABCDEF
Message #2: 123456

A simple transmitted message stream would look like

ABCDEF,123456

where Message #2 follows Message #1 in sequence. If there was, for example, a noise burst in the first message such that the receiver collected

ABxyzF,123456

where "CDE" were changed to look like "xyz", and the receiver could only correct up to two-bit errors, then Message #1 would have to be discarded altogether and the information would be lost.

On the other hand, if Message #1 were interleaved with Message #2 such as swapping every other bit between messages, i.e. B with 2, D with 4, etc., then the new transmitted message stream would look like

A2C4E6,1B3D5F

and if that same noise burst blasted three bits in Message #1 such that the received messages were

A2xyz6,1B3D5F

then the reassembled or deinterleaved messages would have the patterns

ABxDzF,123y56

Now the receiver error processing can detect and recover the original messages since, in this example, there were no more than two bit errors resident per message.

So, by interleaving and deinterleaving, bit errors can be reduced per message. This allows the implementation of less expensive and faster EDC methods at the cost of additional bit-swapping in the encoding and decoding stations.

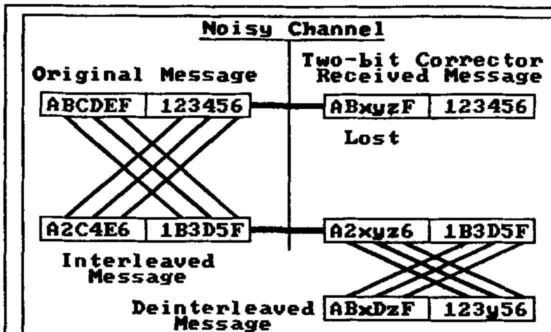

Figure 2. An example of Interleaving to reduce message error.

There are a number of interleaving methods that are commercially in use. Compact audio disk (CD) players, as an example, make use of interleaving and redundancy, together with sophisticated EDC algorithms, to recover audio with a typical dynamic range of well over 70 dB. The coding gains provide a minimum of 5 dB with a raw burst error rate of 10 (-3) (as in 10 to the -3 power or 0.001) to gains of 13 dB with raw burst error rates of 10 (-4). These gains produce uncorrectable error rates of 10 (-8) to 10 (-17), respectively [2]. As can be seen, coding gain is not linear with error rates, and errors still exist but at a dramatically reduced rate.

5.3 Parity

Parity, as applied in digital error control, is the logical result of adding binary digits. If the logical sum is zero then this condition is defined as "even parity". When the logical sum is one then this condition is defined as "odd parity." Understanding the parity concept is extremely important since it is the basis of nearly all sophisticated digital error detecting and correcting techniques.

The sums we refer to are from the logical addition of all bits in a message. Say that an incoming message was "10101", like that in Figure 1. From the above discussion, we would determine the parity of this message by adding together the bits and testing the resulting sum.

Before we can proceed, it is important to know how to add with binary numbers. The elementary rules of binary addition are [3]:

$$0 + 0 = 0$$
$$0 + 1 = 1$$
$$1 + 0 = 1$$
$$1 + 1 = 0$$

Any two identical bits produce a zero sum, while any combination otherwise produces a sum value of one. This, by the way, is the truth table for a two-input exclusive-or gate. In general, binary addition is commonly termed "modulo-2" addition. Special note: here is a case were one plus one does not equal two!

Anyway, with this set of rules we can determine the parity of any number of bits. For the given example, the parity is "odd" because

$$1 + 0 + 1 + 0 + 1 = 1$$

where the sum was evaluated to a value of a logical one. Observe that there are always an odd number of ones in "odd" parity, and consequently "even" parity has a even number of ones.

So how is parity applied to error control? A simple technique is to send a message with a known parity. To force a message into a desired parity, one needs to evaluate the parity of the original message first, and then append the resulting bit, often called the parity-bit, to the original message. This has, of course, increased the message by one bit, but now the message is guaranteed to be transmitted with a known parity. In general, if M is a message composed of b-bits such as

$$M = [\ b_1 \ b_2 \ . \ . \ . \ b_n \ ]$$

then the parity-bit is computed by

$$p = b_1 + b_2 + . \ . \ . + b_n$$

and the parity-corrected message actually transmitted is

$$M = [\ b_1 \ b_2 \ . \ . \ . \ b_n \ p \ ]$$

Note that the parity-bit was appended to the original message. When the true "p" is appended to M, then M will always have "even" parity. When "p" is complemented (if 0 then 1, or if 1 then 0) then M will have "odd" parity. In the case of our example, these steps are shown to produce:

$$M1 = [\ 10101\ ]$$

$$p = 1+0+1+0+1 = 1$$

$$M2 = [\ 101011\ ]$$

where M1 was the original message and M2 was the parity-corrected message. The M2 message is necessarily one bit longer than the M1 message. This is a penalty since we have started to add overhead, i.e. bits sent that are not the actual message, and overhead reduces the overall message rate.

But the reason for adding a parity-bit to our message was to guarantee message parity. The encoder and decoder were both in aggreement as to the parity of the messages. Thus, if the decoder evaluates a message with a parity different than what was expected, an error can be announced. This idea was the basis for many high-powered error detection and correction techniques.

Obviously, this method alone, although better than nothing, is a weak defense against digital noise. It is quite possible for noise to corrupt the message but retain the original parity, thus giving a false sense of protection. Therefore, more powerful techniques must be considered.

6.0 ADVANCED TOPICS

Section 4.1 alluded to several performance classes of EDC techniques. As it turned out, each class had its advantages and disadvantages. What is one willing to pay for EDC? The addage "you get what you pay for" applies perfectly in this case.

In all cases, one must examine and process the original message, perhaps changing or adding code to the original information. These codes are not unlike the parity-bit we added to our little binary message. But a simple parity-bit may be the least effective code we could generate, next to none at all. If expense and response time were no object, then one would pursue methods that produced the highest gain, i.e. those methods that generated the biggest difference between coded and non-coded Signal-to-Noise levels (coding gain).

This leads to a brief description of "block" and "convolution" type codes. In general, simple block codes provide an effective low-cost EDC solution. They typically can correct one-bit errors, and detect two or more bit errors. Block codes take a fixed bit-size information block and generate a special codeword that is appended to the message. The codeword is usually a parity-like function that uniquely describes the information block. This uniqueness permits effective error detection and single-bit correction. Each block of information is considered independent of preceding data blocks.

Convolution codes differ from block codes in that convolution codes take a stream of information bits and generate a stream of encoded bits by applying feedback (or recursive) equations often called generator polynomials. Unlike block codes, these type of codes require memory of recently processed bits, and so have structures like that of some digital filters.

Concatenated codes are encoding techniques that merge more than one EDC method into an overall error control scheme. These have been found to be highly effective, but necessarily require the highest cost of all. Table 1 provides a short list of some well known coding methods and their respective coding gains.

Communication scientists and information theorists have, over the past thirty years, developed a wide variety of techniques that have application in many different systems. This includes cable system networks, too. Your equipment should incorporate some form of effective EDC, if not then chances are the complaint levels may be excessive. The interested reader is encouraged to investigate this most fascinating field of digital error control.

TABLE 1. A partial list of some EDC codes grouped by performance. The gain is the difference between coded and non-coded Signal-to-Noise levels [1,4].

CODING GAIN	CODES
1-3 dB	Hamming Golay
4-7 dB	BCH Reed-Solomon (RS) Viterbi
+8 dB	RS-Viterbi Fano

Note: Use of redundancy and interleaving in addition to the error control coding significantly increases coding gain.

7.0 SUMMARY

Cable system networks have evolved from television signal distribution systems to multi-service advanced communication systems employing digital remote control at a number of levels in the network. Satellite links have delivered attractive programming to cable operators, with encryption and scrambling equipment providing effective program and revenue containment.

Individual decoder control has blossomed as a de facto standard. Digital computer commands are issued to each or all decoders by inserting this data and other digitized information in the VBI and HBI of the video signal in base-band transparent designs, or on sub-carriers or extended bandwidth channels.

The serious effects of digital data errors demand error control techniques radically different from that offered by classical analog noise control solutions. Significant gains using digital error control methods can be obtained, but at the minimal cost of added complexity.

Redundancy, interleaving and parity were basic concepts upon which elaborate techniques were designed and implemented in most digitally oriented communications equipment found commercially in use today. The box on your TV is not just a decoder, it is truely a marvel in communications technology.

8.0 REFERENCES

[1] Michelson, A.M. and Levesque, A.H. "Error-Control Techniques for Digital Communication." Wiley and Sons, Inc., New York, 1985

[2] Ohr, Stephan "Error checking and correcting IC slashes optical disk defects." Electronic Design, December 12, 1985 pp. 37-38.

[3] Berlekamp, Elwyn R. "Algebraic Coding Theory." Revised 1984 Edition. Aegean Park Press, Laguna Hills, CA.

[4] Fthenakis, Emanuel "Manual of Satellite Communications" McGraw-Hill, New York, 1984 Chapter 12, pp. 233-263.

9.0 ACKNOWLEDGEMENT

The author is very grateful to Mr. Frank Pennypacker, Director of Engineering at Oak Satellite Systems Division, for his insightful comments and suggestions.

Additional reader comments may be addressed to the author at:

Oak Communications Inc.
Satellite Systems Division
100 South Main Street
Crystal Lake, Illinois
60014