

N 9 4 - 2 4 3 9

1993
NASA/ASEE SUMMER FACULTY FELLOWSHIP PROGRAM
MARSHALL SPACE FLIGHT CENTER

THE UNIVERSITY OF ALABAMA OF HUNTSVILLE
ERROR CODING SIMULATIONS

Prepared by:	Viveca K. Noble
Academic Rank:	Instructor
Institution and Department:	Tuskegee University
MSFC Colleague(s):	Bernd Seiler Helen L. Thomas
NASA/MSFC:	
Office:	Astrionics Laboratory
Division:	Computers and Communications
Branch:	Flight Data Systems

Introduction

There are various elements such as radio frequency interference (RFI) which may induce errors in data being transmitted via a satellite communication link. When a transmission is affected by interference or other error-causing elements the transmitted data becomes indecipherable. It becomes necessary to implement techniques to recover from these disturbances. The objective of this research is to develop software which simulates error control circuits and evaluate the performance of these modules in various bit error rate environments. The results of the evaluation provides the engineer with information which helps determine the optimal error control scheme.

The Consultative Committee for Space Data Systems (CCSDS) recommends the use of Reed-Solomon (RS) and Convolutional encoders and Viterbi and RS decoders for error correction (Reference [2]). The use of forward error correction techniques greatly reduces the received signal-to noise needed for a certain desired bit error rate. The use of concatenated coding, e.g. inner convolutional code and outer RS code, provides even greater coding gain. The 16-bit Cyclic Redundancy Check (CRC) code is recommended by CCSDS for error detection (Reference [2]).

Evaluation and Implementation

The initial development phase of the simulator required evaluation of custom error correction software generated for Goddard Space Flight Center (GSFC) to determine what modules were applicable to Marshall Space Flight Center's (MSFC) planned laboratory capabilities as illustrated in Figure 1. A block diagram which illustrates the operation of the GSFC software is shown in Figure 2.

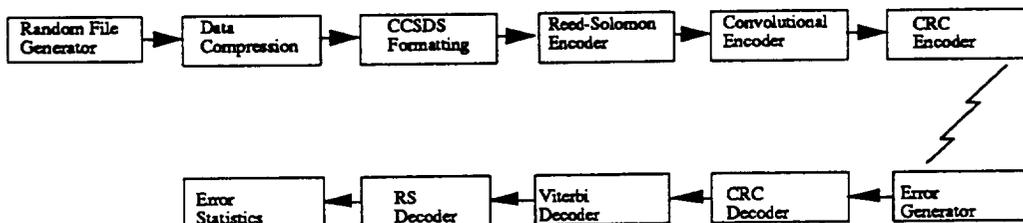


Figure 1

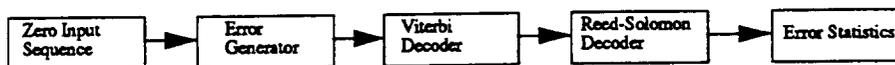


Figure 2

Since the software assumes an all zero input sequence, there is no need for an encoder because the encoded sequence will still be all zeros. This makes the task of determining the error rate a matter of only determining the percentage of non-zero decoder outputs (Reference [5]). Since MSFC's desired system requires random or user-specific data the software from Goddard is unusable in its present form. In order to provide error control capabilities for the Solar Xray Imager (SXI), the remaining modules of the CCSDS telemetry system simulator were developed. These modules include a multiplicative congruential random number generator (RNG), a random error generator, a CCSDS formatter and a CCSDS recommended CRC error detection encoder/decoder. The error statistics generator is currently being developed.

The RNG uses Equation 1 (Reference [3]):

$$X_{n+1} \equiv X_n p \pmod{2^k} \quad [1]$$

where $X_n = 1$, $p = 37$ and $k = 15$. These variables may be assigned any value but X_n and p must be odd. The RNG produces 8968 (8920 bits, maximum transfer frame length plus 48 bits, primary header length) decimal values ranging from 0 to 8191 with a period of 2^{k-2} . Binary values are generated by dividing the decimal values by 4000 and assigning 1 to resulting values greater than 0.5 and 0 to resulting values less than or equal to 0.5. The binary values are used as the random input data and the decimal values are used to access elements in the CRC-encoded message to generate errors in random order.

The CCSDS formatter inserts the sync marker $1ACFFC1D_{hex}$ (Reference [1]) at the beginning of the binary data file to conform to the CCSDS transfer frame format shown in Figure 3 (Reference [2]).

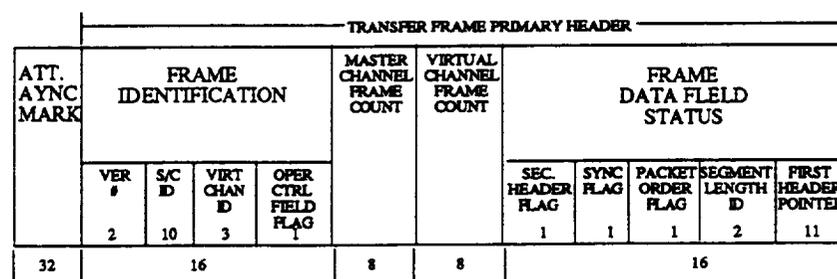


Figure 3

The CRC encoder looks for the 32-bit sync marker, encodes the remaining information bits after synchronization is established and stores the first forty-eight (48) bits of the remaining bits in a header array. The error detection encoder module is the software implementation of the circuit in Figure 4 (Reference [2]).

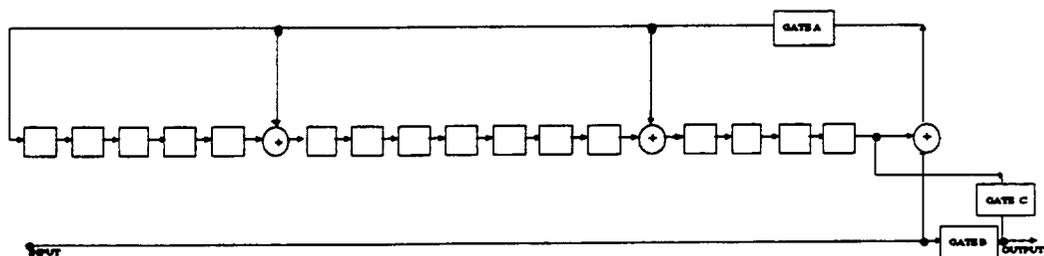


Figure 4

This procedure generates a $(n, n-16)$ code where n is the number of bits in the encoded message and $n-16$ is the unencoded message. Equation 2 is the 16-bit Frame Check Sequence (FCS)

$$FCS = [X^{16} \cdot M(X) + X^{(n-16)} \cdot L(X)] \text{ modulo } G(X) \quad [2]$$

where $M(X)$ is the unencoded message in the form of a polynomial, $L(X)$ is the polynomial

Conclusion and Future Tasks

All of the previously discussed software is written in FORTRAN 77. Due to the inflexible nature of this language, e.g. input data arrays must be given a declared size, it is recommended that the code be converted to C and all future code be written in C. Appropriate error distributions must be determined so that customized error control environments may be developed. The current error correction portion of the system must be written for use with random data and user specific data. Convolutional and RS encoders and a more refined and flexible error generator must be developed. Data compression modules need to be added for the handling of "housekeeping" data. Testing of the code for various bit error rates must be continued in order to gather statistical data on the performance of the code. The process presented above provides a modular, inexpensive error control environment. Its use will allow an engineer to create an optimal error control environment for a given error distribution prior to implementing the procedure in hardware.

References

- [1] *Telemetry Channel Coding*, Recommendation CCSDS 101.0-B-3, Issue 3, Blue Book, Consultative Committee for Space Data Systems, May 1992 or later issue, p. 5- 1.
- [2] *Telemetry*, Recommendation CCSDS 100.0-G-1, Issue 1, Green Book, Consultative Committee for Space Data Systems, December 1987 or later issue, pp. 3-19 - 3-20 and pp. D-1 - D-4.
- [3] Hamming, R. W., *Numerical Methods for Scientists and Engineers*, Dover Publications, Inc., New York, 1986
- [4] Jain, Raj, *Error Characteristics of Fiber Distributed Data Interface (FDDI)*, IEEE Transaction on Communications, Vol. 38, No. 8, August 1990, p. 1249.
- [5] Odenwalder, Joseph P., *Error Control Coding Handbook*, Final Report, Contract F44620-76-C-0056, July 15, 1976, p. 4 and p. 122.