A 64B/67B Channel Code Implementation and Its Application in Seismic Data Acquisition System

Cheng MA, Shikai WANG, Xue YANG, Jianping XIONG
State Key Laboratory of Precision Measurement Technology and Instruments, Department of Precision Instruments, Tsinghua University, Beijing, China

ABSTRACT: Based on guided scrambling technique, the 64B/67B channel code has two remarkable advantages of high bit rate of 64/67 and DC-balance. In this paper, we realize the FPGA implementation of 64B/67B coding scheme and make out a comparison with 8B/10B coding scheme. Testing results indicate that taking resource occupation, bit error rate and effective transmission speed into consideration 64B/67B channel code is the best choice for seismic data acquisition system.

KEYWORD: 64B/67B; Guided scrambling; DC-balance; Bit rate

1 INTRODUCTION

1.1 *Seismic data acquisition system*

Large-scale Seismic Data Acquisition and Recording (LSDAR) system uses long distance UTP channel for its data transmission. To ensure the reliability and stability in data transmission process, small low frequency content, high efficiency and low error multiplication are some of the typical requirements that the coding scheme needs to fulfill. Fortunately, the guided scrambling (GS) theory proposed a designed method to select best GS coding parameters which can meet the low-frequency requirements of the long distance UTP channel. [1]

1.2 *64B/67B channel code*

According to GS theory, if we set the width of source data as 64 bits, choose $d(x) = x^3 + 1$ for scrambling polynomial (number of augmenting bits is 3), we will get 64B/67B coding scheme. [2-3] Its bit rate is as high as 0.9552 and also it has an excellent performance of DC-balance characteristics theoretically by using absolute RDS (ABSRDS) as its selection criteria. [4]

The process of 64B/67B coding scheme can be described as 3 steps. Firstly, expanding the data to 67 bits by inserting 3 augmenting bits before the 64 bits of source data; secondly, multiplying the scrambling polynomial with the data generated in step 1 to get candidate codes; finally, picking out the optimal code for outputting from all the candidate codes according to the selection criteria.

Figure 1 shows the 3 steps of 64B/67B coding process. Here, *s(x)* means 64 bits' source data; *d(x)* is the scrambling polynomial; *b(x)* represents for candidate codes.

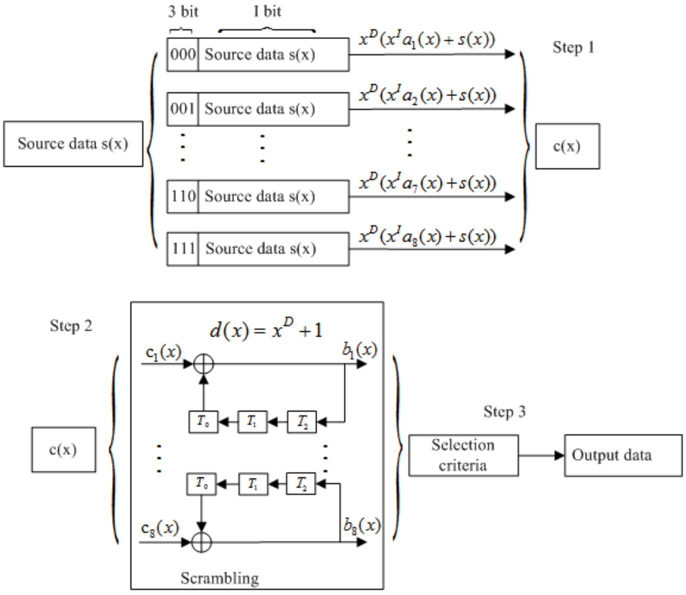

Figure 1. 64B/67B coding process

2 IMPLEMENTATION

2.1 *Frame*

The hardware implementation is mainly consist of encoder and decoder 2 blocks; specifically encoder can be divided into source data generation, scrambler, and selection & transition three parts; decoder can be divided into receiver, descrambler, and transition three parts. [5] Figure 2 shows the internal structure of encoder and decoder.

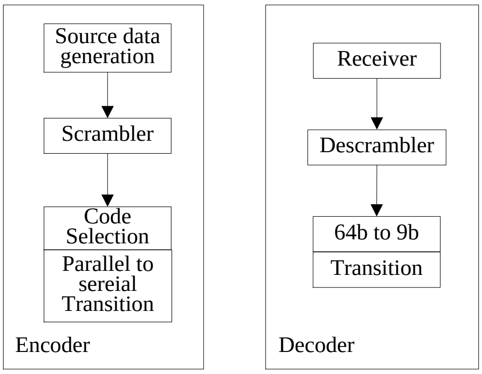

Figure 2. Internal structure of encoder and decoder

2.2 *Encoder*

There should be no clearance between characters in the channel. So encoder needs to send idle characters in the source data generation part when there is no signal data appears.

2.2.1 *Scrambler*

Scrambling was implemented using a series of interconnected delay elements and XOR gates, as shown in Figure 3.

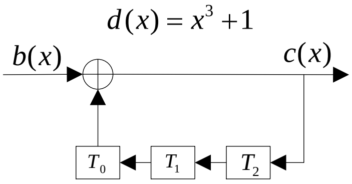

Figure 3. Scrambling process

2.2.2 *State machine for code selection*

Absolute RDS uses absolute operation to replace square operation, its performance is similar to minimum squared weight (MSW).

Figure 4 displays the state machine in FPGA to realize optimal code selection. Its main work focus on obtaining the candidate codes' ABSRDS values and select the optimal code by comparing their ABSRDS values.

2.3 *Decoder*

The most important part of decoder is descrambler, which works to translate the code received into original data. It is the inverse process of scrambling.

2.3.1 *Descrambler*

Similar to scrambler, descrambler was implemented using a series of interconnected delay elements and XOR gates, the descrambling process is shown in Figure 5.

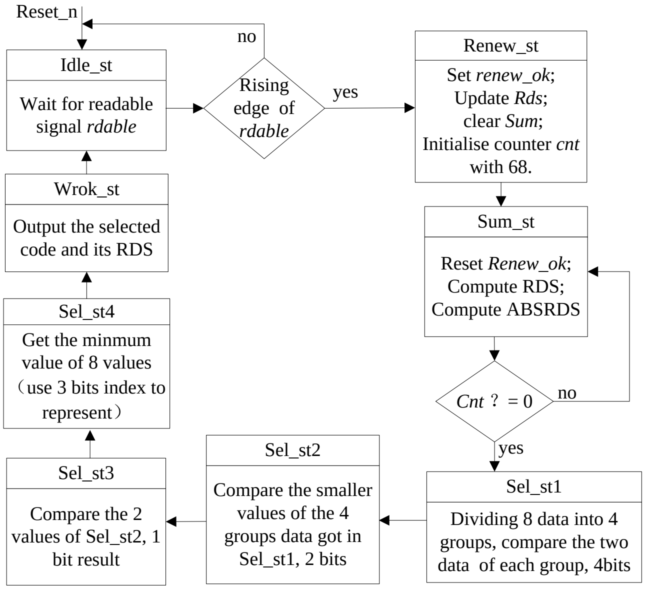
Reset_n
Idle_st
Wait for readable signal *rdable*
no
Rising edge of *rdable*
yes
Renew_st
Set *renew_ok*; Update *Rds*; clear *Sum*; Initialise counter *cnt* with 68.
Wrok_st
Output the selected code and its RDS
Sum_st
Reset *Renew_ok*; Compute RDS; Compute ABSRDS
Sel_st4
Get the minmum value of 8 values （use 3 bits index to represent）
Cnt ? = 0
no
yes
Sel_st3
Compare the 2 values of Sel_st2, 1 bit result
Sel_st2
Compare the smaller values of the 4 groups data got in Sel_st1, 2 bits
Sel_st1
Dividing 8 data into 4 groups, compare the two data of each group, 4bits

Figure 4. State machine for optical code selection

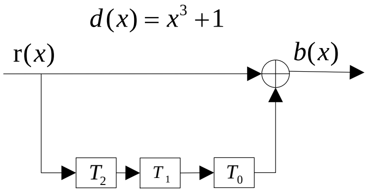

Figure 5. Descrambling process

2.4 *FPGA logic resource occupation*

8B/10B coding scheme is one of the superior modulation coding schemes, and it has been used in digital cable communication for a long time. In order to assess the performance of 64B/67B coding scheme, we implement both of these two kinds of coding schemes respectively on the same FPGA circuit to make evaluation on their performance.

Table 1 shows the percent of logic resource expense of 8B/10B and 64B/67B.

Table 1 Logic resource expense of 8B/10B and 64B/67B

	Encoder logic resource expense (LE)	Decoder logic resource expense (LE)	Percent of total resource
8B/10B	42	39	0.3%
64B/67B	70*	67	0.4%
	1365**		4.3%

* Scrambler, ** Optimal code selection.

The scrambler in 64B/67B coding scheme just consumed 0.4% of all system logic resource, is almost the same as 8B/10B coding scheme. But the code selection part consumed 4.3% of the system logic resource. Since 4.7% resource occupation is still a small scale for FPGA core, it can meet the design requirements of seismic data acquisition system.

3 VERIFICATION

The developed 64B/67B code had been applied to our large scale seismic data acquisition system. Experiments had been made to test its performance. Figure 6 displays the picture of the experiment system and Figure 7 shows its block diagram.

Figure 6. FPGA implement

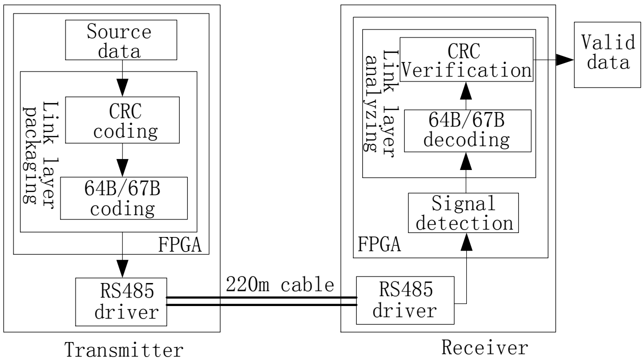

Figure 7. Frame of testing system

3.1 *Bit error rate*

We can easily get the bit error rate by minoring the CRC model. Table 2 shows the testing results at 5MHz and 8MHz. When the clock frequency is 5MHz, the bit error rate of both 8B/10B and 64B/67B coding scheme are 0. When clock frequency is 8MHz, the bit error rate is 2.4×10^{-5}, but it can be eliminated by error correction model in the system.

Table 2. Bit error rate at 5MHz and 8MHz

	5MHz	8MHz
8B/10B	0%	0%
64B/67B	0%	2.4×10^{-5}

3.2 *Effective transmission speed*

In the effective transmission speed testing process, we set the frame length of source data as 200 bytes. According to the 8B/10B and 64B/67B encoding rules, it is necessary to insert synchronization characters in front of each data frame. So the effective bit rate will be less than 8/10 and 64/67. Taking the synchronization characters into consideration, the effective bit rate of these two coding scheme mentioned above should have the relation below,

8B/10B, effective bit rate, $\frac{200*8}{205*10}=0.780$;

64B/67B, effective bit rate, $\frac{200*8}{26*67}=0.918$;

$$v_{64B67B}/v_{8B10B}=\frac{0.918}{0.78}=1.178 \qquad (1)$$

Theoretically, the effective transmission speed of 64B/67B should be 1.178 times faster than 8B/10B coding scheme when works at the same clock frequency.

Table 3 presents the effective data transmission speed in testing experiments.

Table 3. Effective data transmission speed at 8MHz

Frame/Second	8MHz
v_{8B10B}	3190
v_{64B67B}	3682
v_{64B67B}/v_{8B10B}	1.154

It is easy to find that the effective transmission speed of 64B/67B is 1.15 times faster than 8B/10B. The effective transmission speed has been improved by 15%.

The difference between theoretical values (1) with the values measured is caused by the idle characters for channel sequence padding.

4 CONCLUSIONS

The hardware implementation based on FPGA presented proves that 64B/67B coding technique is practical in seismic data acquisition system. By comparing the source occupation status, bit error rate and effective transmission speed, we assessed the performance of these 3 kinds of coding scheme. And 64B/67B channel code is highly suitable for seismic data acquisition system.

REFERENCES

[1] Fair, I.J. &Grover, W.D. 1991. Guided scrambling: a new line coding technique for high bit rate fiber optic transmission systems. *Communications, IEEE Transactions* 39(2): 289-297.

[2] Xue, Yang et al. 2012. A high-bit-rate (64/67) DC-balance line code scheme based on guided scrambling. *Applied Mechanics and Materials* 229-231: 1572-1576.

[3] Fair, I.J. & Wang, Q. 1995. Polynomials for guided scrambling line codes. *Selected Areas in Communications, IEEE Journal* 13(3): 499-509.

[4] Immink, S.A. & Wilson, H.Y. 2000. A comparison of two schemes for generating DC-free RLL sequences. *IEEE International Symposium on Information Theory*. Sorrento: IEEE.

[5] Hua, X.H. & Chen, D.L. 2008. The Design and Simulation Based on Scrambler and Descrambler of AHDL. *Journal of Huangshi Institute of Technology* 2008 (3): 53-56.