CCSDS PN PROCESSING SPEED OPTIMIZATION

Sang-Il AHN[1*], Tae-Hoon KIM[2] , In-Hoi KOO[1]

Ground System Development Department, Korea Aerospace Research Institute (KARI) [1]
siahn@kari.re.kr, freewill@kari.re.kr
SOLETOP Inc. Satellite Image Department[2]
freekid99@soletop.com

ABSTRACT: Telemetry processing system requires minimum bit transition level in data streams to maintain a bit synchronization while receiving telemetry signal. PN code has a capability of providing the bit transition and is widely used in the packet communication of CCSDS. CCSDS PN code that generator polynomial is $h(x) = x^8 + x^7 + x^5 + x^3 + 1$, and the random bit sequence that is generated from this polynomial is repeated with the cycle of 255 bits. As the resolution of satellite image increases, the size and transmission rate of data increases. To process of huge and bulky size of satellite image, the speed of CCSDS PN Processing is very important.

This paper introduces the way of improving the CCSDS PN Processing speed through processing 128 bits at one time using the feature of cyclic structure that repeats after first 255 bytes by grouping the random bit sequence with 1 byte and Intel Streaming SIMD Extensions 2. And this paper includes the comparison data of processing speed between SSE2-applied implementation and not-applied implementation, in addition, the measured value of speed improvement.

KEY WORDS: CCSDS, PN Processing, High-speed Processing, SSE2

1. INTRODUCTION

Telemetry processing system needs the bit transition to the input signal at minimum level to maintain a bit synchronization while receiving telemetry signal. In order to ensure proper receiver operation, the data stream must be sufficiently random [1]. Pseudo-Noise (PN) code has a capability of providing the bit transition.

The Consultative Committee for Space Data Systems (CCSDS) defined the Pseudo-Randomizer that ensures sufficient randomness for all combinations of CCSDS-recommended modulation and coding schemes. The packet communication of CCSDS is widely used in many satellite systems. CCSDS Pseudo-Randomizer that generator polynomial is $h(x) = x^8 + x^7 + x^5 + x^3 + 1$, and the random bit sequence that is generated from this polynomial is repeated with the cycle of 255 bits.

As the application of high resolution satellite image such as over 1 m is growing, the need of the fast acquisition of image data is increasing. Because the high resolution satellite image has the huge amount of data size, the need of high speed processing is increasing. And the real time processing of receiving system is required. But the general CCSDS PN Processing software spends much time.

This paper introduce the way of improving the CCSDS PN Processing speed through processing 128 bits at one time using the feature of cyclic structure that repeats after first 255 bytes by grouping the random bit sequence with 1 byte and Intel Streaming SIMD Extensions2 (SSE2). And this paper includes the comparison data of processing speed between SSE2-applied implement and not-applied implement, in addition, the measured value of speed improvement.

2. CCSDS PN PROCESSING

2.1 Pseudo-Randomizer

In order to maintain bit (or symbol) synchronization with the received telemetry signal, every ground data capture system requires that the incoming signal has a minimum bit transition density. In order to ensure proper receiver operation, the data stream must be sufficiently random [1]. CCSDS defined the Pseudo-Randomizer to ensure sufficient randomness for all combinations of CCSDS-recommended modulation and coding schemes. The method for ensuring sufficient transitions is to exclusive-OR each bit of the Code block or Transfer Frame with a standard pseudo-random sequence [1]. The configuration at the sending end is shown in Figure 1. The pseudo-random sequence shall not be exclusive-ORed with the Sync Marker.

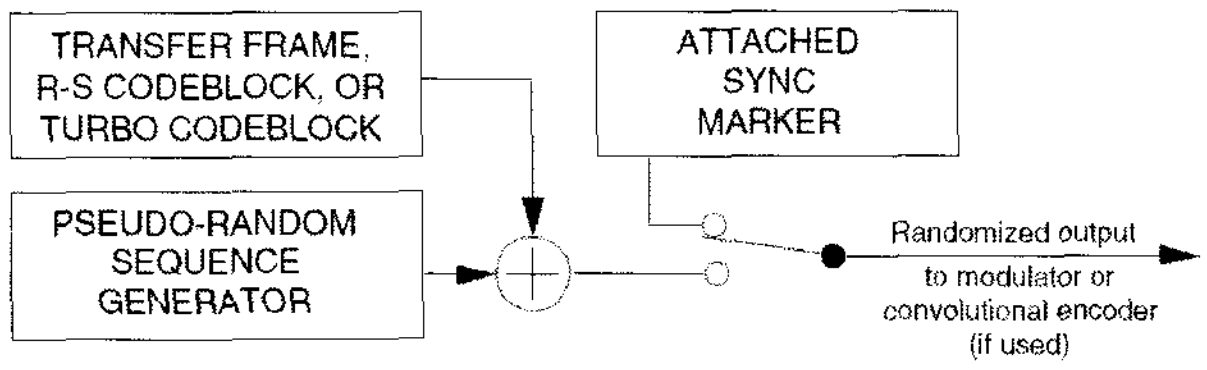

TRANSFER FRAME, R-S CODEBLOCK, OR TURBO CODEBLOCK
ATTACHED SYNC MARKER
PSEUDO-RANDOM SEQUENCE GENERATOR
Randomized output to modulator or convolutional encoder (if used)

Figure 1. Pseudo-Randomizer Configuration

On the receiving end, the original Code block or Transfer Frame is reconstructed using same pseudo-random sequence.

2.2 Pseudo-Random Sequence Generator

The CCSDS pseudo-random sequence shall be generated using following polynomial:

$$h(x) = x^8 + x^7 + x^5 + x^3 + 1 \quad (1)$$

The sequence begins at the first bit of the Code block or Transfer Frame and repeats after 255 bits, continuing repeatedly until the end of the Code block or Transfer frame. The sequence generator is initialised to the all-ones state at the start of each Code block or Transfer Frame.

The logic diagram of CCSDS Pseudo-Randomizer is shown in Figure 2.

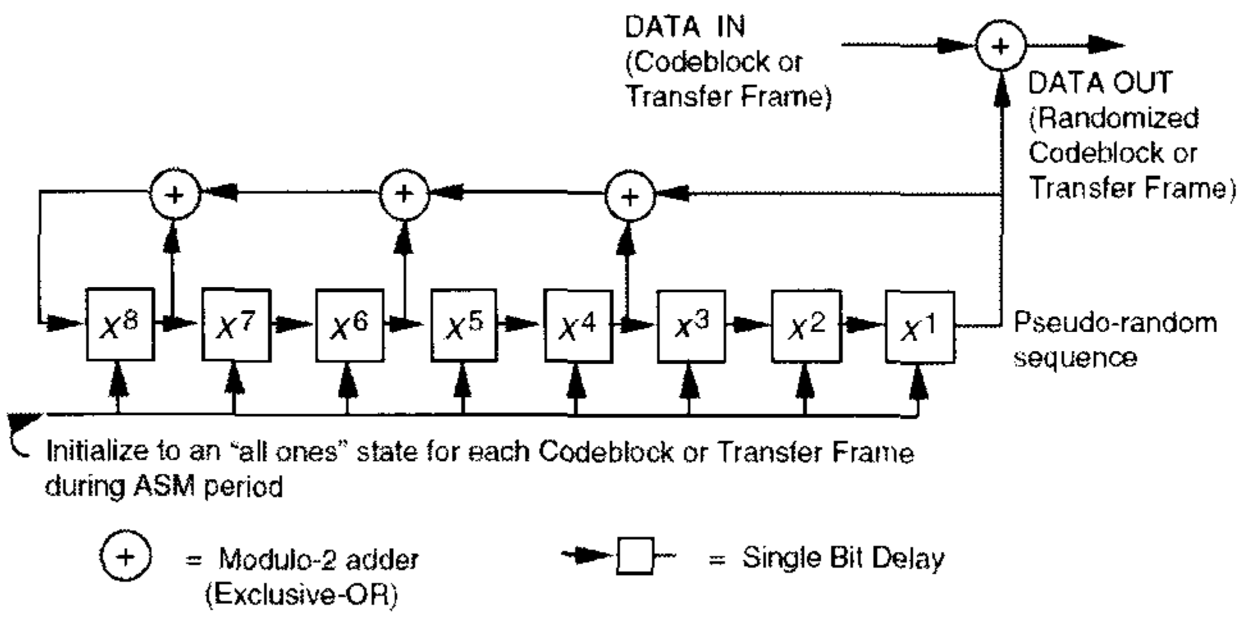


Figure 2. CCSDS Pseudo-Randomizer Logic Diagram

The first 40 bits of the pseudo-random sequence from the generator are shown below;

1111 1111 0100 1000 0000 1110 1100 0000 1001 1010

The leftmost bit is the first bit of the sequence.

3. TECHNIQUES FOR PARALLEL PROCESSING

3.1 Streaming SIMD Extensions 2

SIMD is the one of three performance enhancement factor in MMX technology. With the help of this, just single instruction can be used for iterative loop with multiple instructions.

For example, the 32 bytes data can be processed by 32 times of 1 byte data operation but 32 bytes data can be processed by 2 time instruction with SIMD as shown in Figure 3.

SSE2 is extended technology for SIMD and Intel applied this technology into P-4 since Nov, 2000. SSE2 includes new 144 instruction commands using SIMD technology and both 128-bit length integer and float instruction command were added.

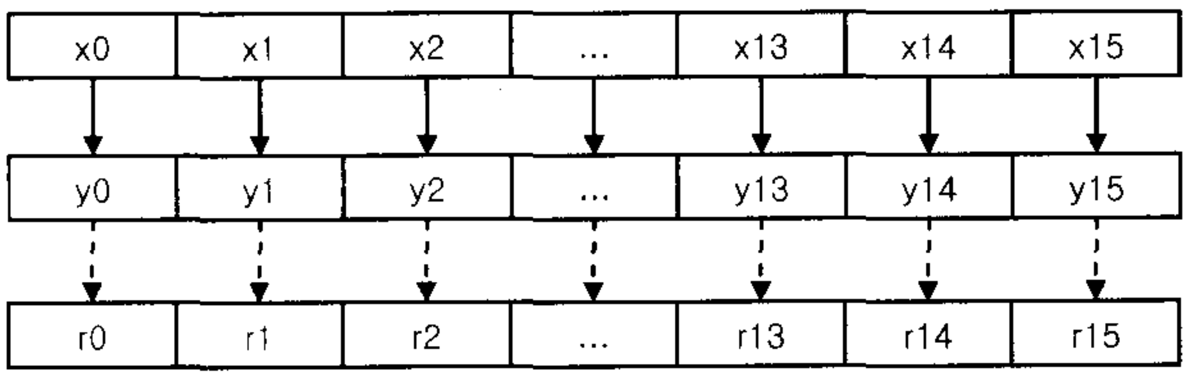


Figure 3. SSE2 Processing Scheme

3.2 Cyclic Structure of PN code

The logic diagram in Figure 2 is suitable for implementing Pseudo-Randomizer using simple hardware structure.

When this diagram is directly used for software implementation, it normally undergoes the time-consuming because includes calculation for all bits. But when the feature of cyclic pattern by 255bit is used the processing time can be reduced.

To apply the cyclic pattern characteristic into software, the Pseudo-Random sequence should be aligned by byte, minimum memory size.

Pseudo-random sequence shows the pattern repeat by every 255bytes when packed by 1 byte.

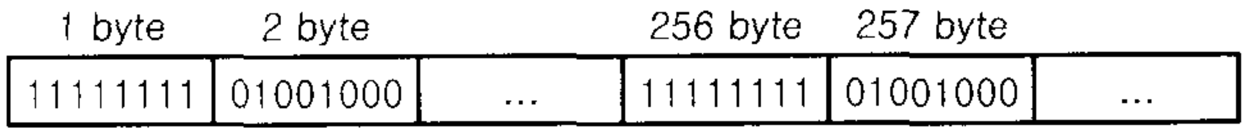


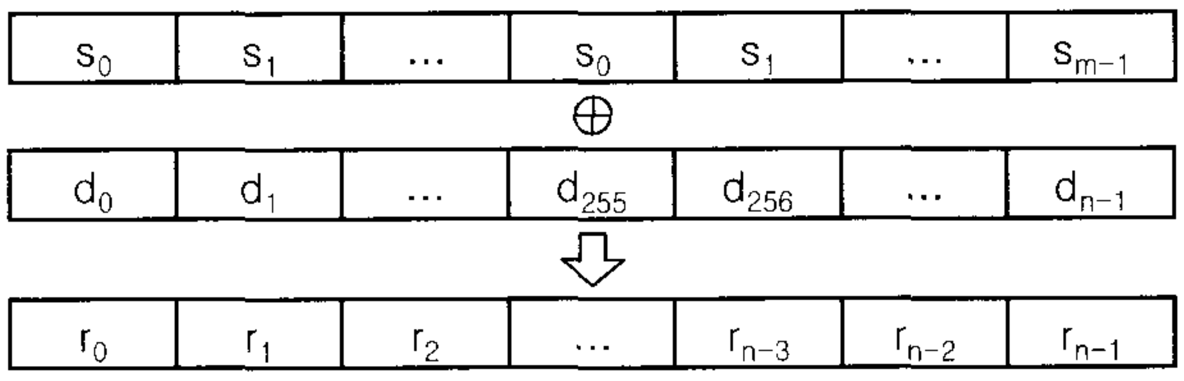
Figure 4. Cyclic Characteristic in Byte Aligned Pseudo-Random Sequence

The obtained byte aligned sequence is shown in Figure 5.

```
000 : FF 48 0E C0 9A 0D 70 BC
008 : 8E 2C 93 AD A7 B7 46 CE
...
248 : 08 78 C4 4A 66 F5 58 FF
```

Figure 5. Byte Aligned CCSDS Pseudo-Random Sequence

Figure 6 shows the LUT usage based on Pseudo-random sequence's cyclic pattern by 255bytes.



where s = Pseudo-random sequence
d = Code block or Transfer Frame
n = Length of d
m = index of last pseudo-random sequence,

$$n - \left\lfloor \frac{n}{255} \right\rfloor \times 255$$

Figure 6. Byte Processing Using Cyclic Structure

3.3 Implementation using SSE2

Though LUT processing in Figure 6 can reduce the processing time, the very high speed downlink data, for example, KOMPSAT2 downlink, 320Mbps, can not be processed in real-time.

Only possible approach is to apply the SSE2 technology for 1 instruction for upto 16byte operation at a time. With SSE2, the processing speed can be remarkably increased.

Figure 6 shows the operation used LUT for CCSDS Pseudo-Randomizer is just simple exclusive-OR.

Therefore if special LUT for 16bytes to be processed is available, the processing itself is not much difficult.

The size of LUT is minimum number of multiples of 16 above code block or transfer frame.

$$y = \left\lceil \frac{x}{16} \right\rceil \times 16 \qquad (2)$$

where x = Code block or Transfer Frame size
y = Look-up table size

The processing time consumed for PN processing of 100,000 to 500,000 times for normal algorithm and for SSE2 application and its results was summarized in Table 1 and 2. Intel(R) Pentium(R)4 3.2GHz was used for simulation test. As shown in Table 2, the data rate for just cyclic operation is about 108.13Mbps, which can not meet the realtime downlink speed of 320Mbps, but the the SSE2 case shows data rate of 2.57Bbps, about 24 times higher speed than normal byte cyclic structure.

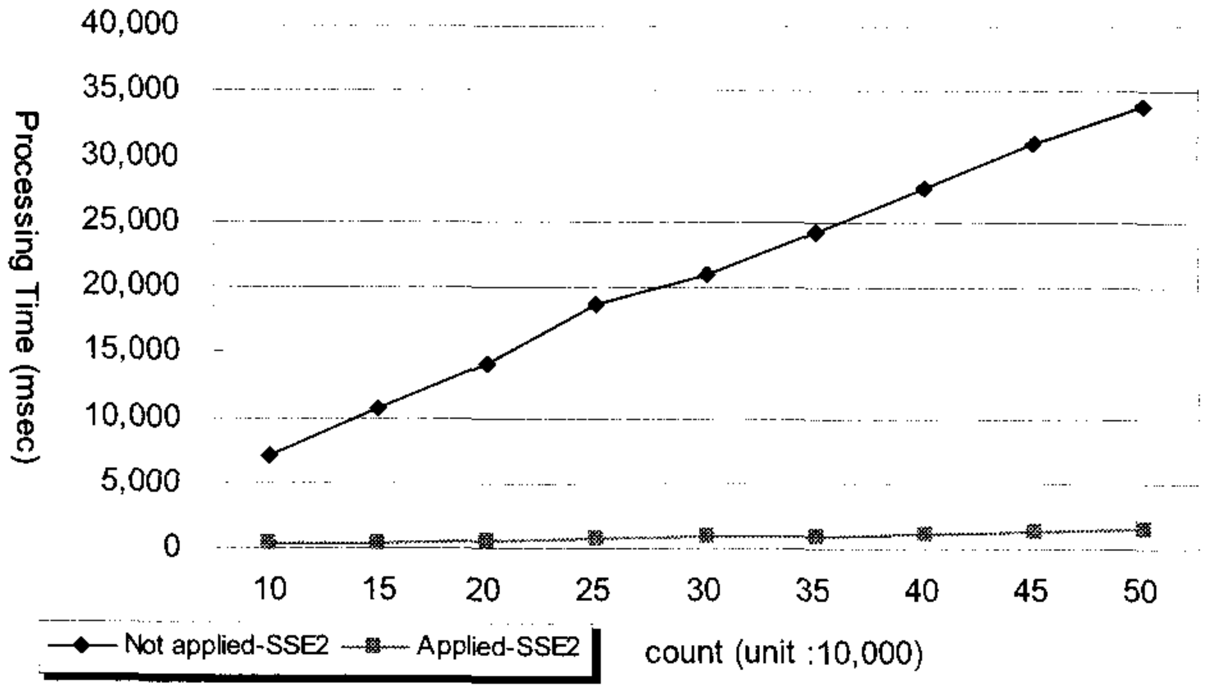


Figure 7. Processing Time Comparison Chart

Count	Not applied-SSE2	Applied-SSE2
100,000	7127.7454	291.3187
150,000	10694.7705	448.0983
200,000	14122.2156	607.3614
250,000	18757.9919	749.4070
300,000	21002.2900	891.9766
350,000	24140.9375	1038.1447
400,000	27555.6543	1174.3733
450,000	31058.9844	1327.5116
500,000	33965.7178	1457.0274

Table 1. Processing Time (msec)

Count	Not applied-SSE2	Applied-SSE2
100,000	0.1066	2.6087
150,000	0.1066	2.5439
200,000	0.1076	2.5025
250,000	0.1013	2.5352
300,000	0.1086	2.5560
350,000	0.1102	2.5621
400,000	0.1103	2.5885
450,000	0.1101	2.5761
500,000	0.1119	2.6079

Table 2. Data Rate (Gbps)

4. CONCLUSION

This paper shows the CCSDS PN processing for high-speed processing and the cyclic structure of CCSDS PN sequence was analyzed to be used for SSE2 application.

The time consumption and processed data rate for 2 cases were measured and found that the SSE2 case shows data rate of 2.57Bbps, about 24 times higher speed than normal byte cyclic structure.

The developed CCSDS PN processing software was applied to KOMPSAT-2 image processing system and validated successfully again. Due to its supporting data rate, this software is expected to be applied to other high speed CCSDS downlink scheme.

5. REFERENCES

[1] Telemetry Channel Coding, CCSDS 101.0-B-6, Blue Book, Oct 2002.

[2] IA-32 Intel® Architecture Software Developer's Manual Volume 2A: Instruction Set Reference, A-M.

[3] IA-32 Intel® Architecture Software Developer's Manual Volume 2A: Instruction Set Reference, N-Z

[4] The Software Vectorization Handbook, Aart J.C.Bik