A Highly Efficient and Fast Algorithm for Implementing a Real-Time Software GNSS Receiver

*Sung-Hyuck Im[1], Gyu-In Jee[2], Hak-Sun Kim[3], Sang-Do Cho[4], Sun-Jun Ko[5]

[1]A Department of Electronics Engineering, Konkuk Univ.(E-mail: shim@konkuk.ac.kr)
[2]A Department of Electronics Engineering, Konkuk Univ.(E-mail: gijee@konkuk.ac.kr)
[3]Samsung Electro-Mechanics (E-mail: hszic.kim@samsung.com)
[4] Samsung Electro-Mechanics (E-mail: sdcho@samsung.com)
[5] Samsung Electro-Mechanics (E-mail: sunjun.ko@samsung.com))

Abstract

In this paper, for implementing a real-time software GNSS receiver we propose the highly efficient and fast algorithms such as partial down-conversion, phase rotator, composite I&Q accumulation, Virtual DCO technique, and parallel acquisition using FFT. When the proposed algorithms are used, more 30 tracking channels with 3 tracking arm(early-prompt-late) is operated real-time on Intel 2.8GHz personal computer. Also, the partial down-conversion reduces the FFT size, for parallel acquisition, to 1/8 of conventional FFT-size and the program size includes map is not exceed 1Mbyte. Finally, the proposed real-time software GNSS receiver using the proposed algorithms provides the navigation solution with below 10 meter rms error.

Keywords: Software correlator, GNSS receiver, signal acquisition, FFT, real-time.

1. Introduction

A real-time software GNSS receiver architecture can provide GNSS user equipment with operational flexibility. A conventional receiver that uses a hardware correlator will require hardware modifications in order to use new GNSS signals. A software GNSS receiver can use new signals, such as GPS L2, L5, and Galileo, without the requirement of a new correlator chip. If the software is changed, new signals could be traced by software GNSS receiver

General hardware GNSS receiver has RF front-end, A/D sampler, correlator and microprocessor. Figure 1 shows the general receiver architecture. New hardware corrlator chip and RF front-end are required to track new GNSS signals. But, in software GNSS receiver, only change of baseband processing software is required.

The notion of a software GNSS receiver has been around for several years. Among the proposed algorithms, the correlation using parallelization and bit-wise processing was attractive [4]. But the proposed algorithm has limitations such as storage requirement for code/carrier map, additive hardware for bit-composing, and processing capacity.

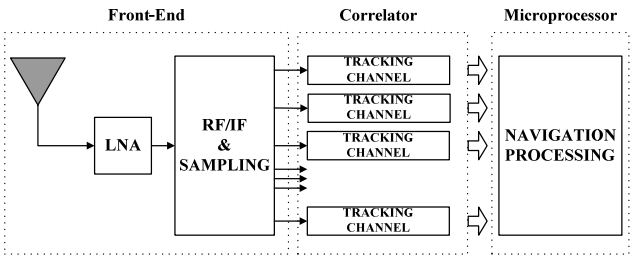

Figure 1. Conventional hardware GNSS receiver

In this paper, we proposed the highly efficient and fast algorithms for implementation of a real-time software GNSS receiver. Newly proposed innovative algorithms which include partial down-conversion, phase rotator, composite I&Q accumulation, Virtual DCO technique, and parallel acquisition using FFT.

2. Fast Acquisition and Correlation

2.1 System Design

Table 1. Representation of input IF signal

Physical value	2 Bit representation
-1	00
-3	01
1	10
3	11

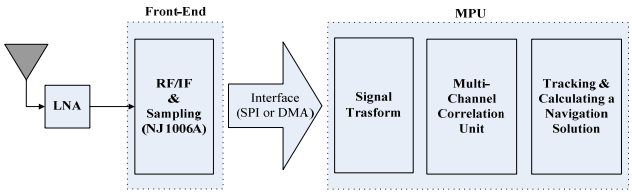

Figure 2. The proposed real-time software GNSS receiver structure

The system is consisted of a Nemerix NJ-1006A RF front-end, NI 653x data acquisition board, and a PC with 2.8GHz Intel Processor running the Windows XP operating system. The RF front-end performs A/D conversion. Its sampling frequency and intermediate frequency are 16.367MHz and 4.188MHz. The digitized signals are a binary 2-bit data which is consisted of (00), (01), (10), (11). The MSB(most significant bit) is sign and the

other bit is magnitude. Table 1 shows binary 2-bit data and it's physical value.

The data acquisition system uses the NI-653x digital I/O card. The NI-653x DAQ supports 20Mbyte/s I/O interface through PCI. DAQ software use C++ language and API functions which is supported by National Instruments. Figure 3 shows the software GPS receiver structure and the developed system.

The whole software GPS receiver program is coded in C++ language and no processor-specific codes are used.

2.2 Partial Down-Conversion

Generally, the sampling frequency is much faster than Doppler frequency. For example, if maximum Doppler frequency is 10000 Hz, sampling frequency with 16.367 MHz is 1636.7 times of Doppler frequency. Therefore, for several samples, carrier replica's phase change, which results from Doppler, can be neglected. Equation (1) represents the IF(intermediate frequency) carrier replica signals.

$$e^{j2\pi\left(f_{IF}+\hat{f}_d\right)nT}, \quad n=0,1,2,... \tag{1}$$

where

f_{IF} : IF frequency

$\hat{f}_d$: Estimated Doppler frequency

$T=1/f_S$: Sampling period

f_S : Sampling frequency

For the processing of several samples, $e^{j2\pi\hat{f}_d nT}$ is regarded to zero. Therefore, equation (1) can be approximated like equation (2).

$$e^{j2\pi\left(f_{IF}+\hat{f}_d\right)nT}=e^{j2\pi f_{IF}nT}e^{j2\pi\hat{f}_d nT}\approx e^{j2\pi f_{IF}nT}, \quad n=0,1,2,...,N-1 \tag{2}$$

where

N : Depended on sampling frequency and IF frequency

In the implemented system, sampling frequency is 8 times of C/A code chipping rate. Therefore, 8 samples are converted to one sample value through being multiplied by the fixed eight carrier replicas which has the nominal IF frequency. Figure 3 shows the partial down-conversion process of the input sampled signals.

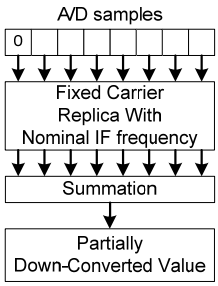

Figure 3. Partial down-conversion process

Therefore, when partial down-conversion is used, sampling frequency f_S is reduced to $f_S/8$. The value, through partial down-conversion process, is used in every correlation channels which have different Doppler values. Figure 4 shows the fixed carrier replicas. When the combination of every possible input values are inputted figure 5 shows the output results through partial down-conversion process.

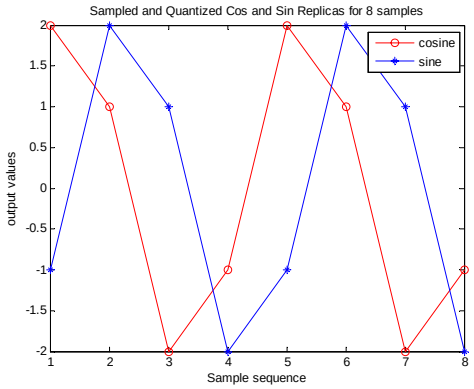

Figure 4. Fixed carrier replica for down-conversion

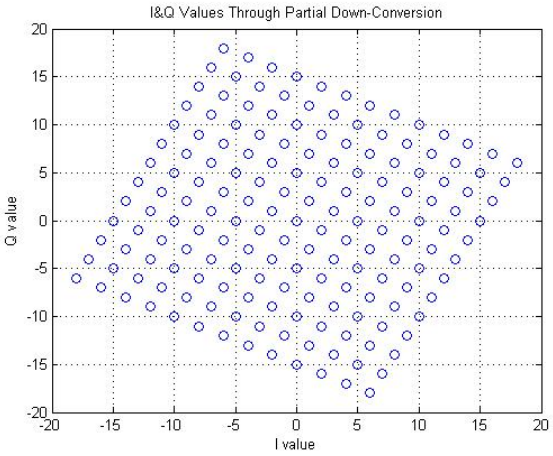

Figure 5. I&Q values through partial down-conversion

I&Q values through partial down-conversion are represented as the index value with 169 kinds. This gives extreme efficiency to the generation of mapping function.

2.3 Fast Correlation

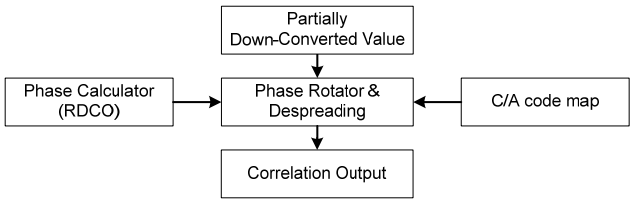

Figure 6. Fast correlation process

Figure 6 shows the correlation process. In the correlation process for each tracked satellite channel, phase rotation which results from Doppler have to be compensated. To compensate the phase rotation, therefore, the partially down-converted value is rotated by the phase rotator. For the calculation of phase rotation the reduced DCO(RDCO) is used. The function of RDCO is different from the function of DCO of a hardware. RDCO is working at every 32 samples. RDCO provides the removal of carrier replica map, which occupies the large storage, for the software reciever

When the fast correlation process is implemented the phase rotator and dispreading block require 128Kbyte map storage. Also, C/A code map requires 128Kbyte map storage for 32 satellites.

2.4 Composite I&Q Accumulation

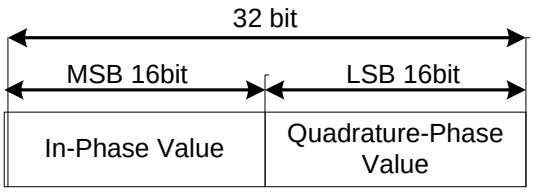

Figure 7. I&Q value structure

Generally, In-phase and Quadrature-phase values are separately accumulated on micro-processor. But the I&Q values have limited values such as less than 10000 in the implemented system. The values through phase rotator&dispreading process don't exceed 19 in both I and Q value. Therefore, to remove minus term I and Q value is added by 19. After ending of 1ms accumulation, additive value can be easily removed. Figure 7 shows the I&Q value structures.

2.5 Virtual Carrier&Code DCO (VDCO)

A conventional hardware receiver uses the DCO for generation of carrier and code replica. The proposed virtual DCO(VDCO) isn't to work at every sampling clock but at every 1mili-second(C/A code period). VDCO calculate the code phase and carrier phase at the start of C/A code and the point of TIC. The calculation method is simple. We can estimate the number of sample with 1 C/A code period. This operation is calculated without float point operation through separating DCO values to MSB 16bits and LSB 16bits. [16 x 16367] is not exceed 32bit integer value.

2.6 Parallel Acquisition

The partial down-conversion process reduces the FFT size for the parallel signal acquisition. Figure 8 and 9 shows the conventional parallel acquisition and the proposed parallel acquisition.

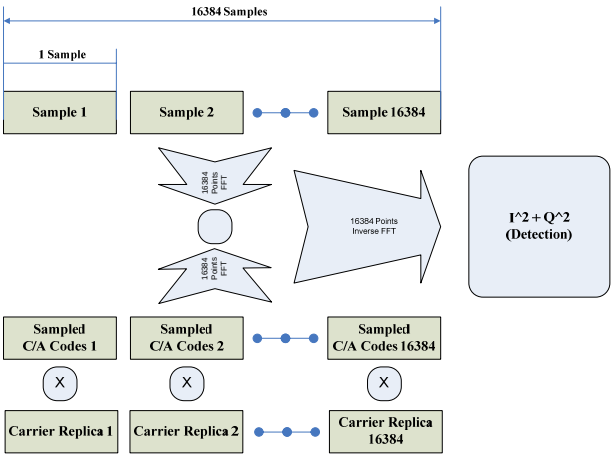

Figure 8. A structure of conventional parallel acquisition

But partial down-conversion can cause the matching error of code phase. The signal leakage equals maximum value with about 1.5dB. However, if 2 partial down-conversion value with different sample phase is made, The signal leakage is reduced to about 0.5dB and this value is not remarkable.

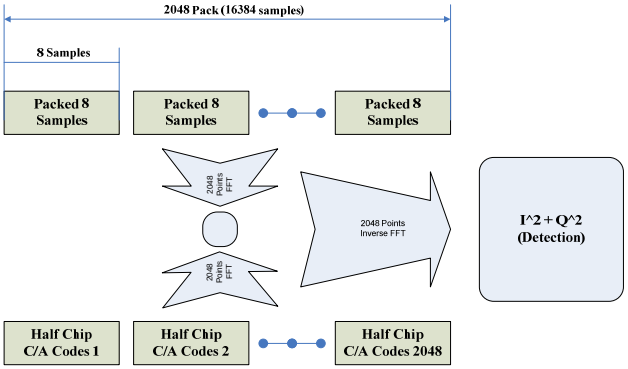

Figure 9. The proposed structure of parallel acquisition

When the proposed parallel acquisition algorithm is used, the computational requirement for parallel acquisition of one satellite is reduce to 1/30 of conventional parallel acquisition.

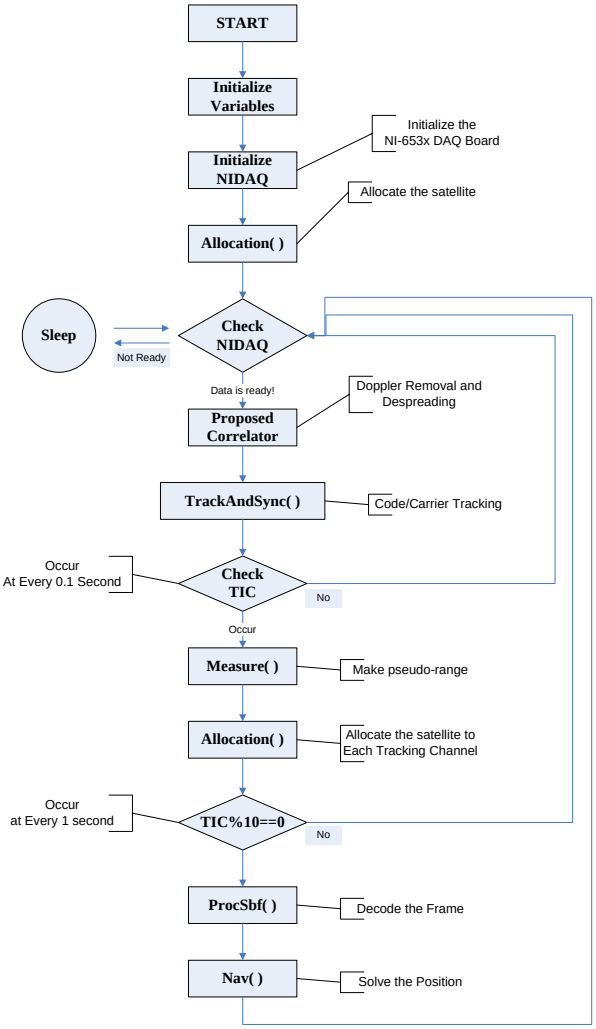

Figure 10. Software flow of the implented system

2.7 Implementation of Real-Time Software GNSS Receiver and Test Results

We implement the real-time software GNSS receiver which

the proposed algorithms are adapted. Figure 10 shows the software flow of the implantation system. Figure 11 shows the monitoring graphic. For the implementation, no processor specific codes are used and C++ language is only used.

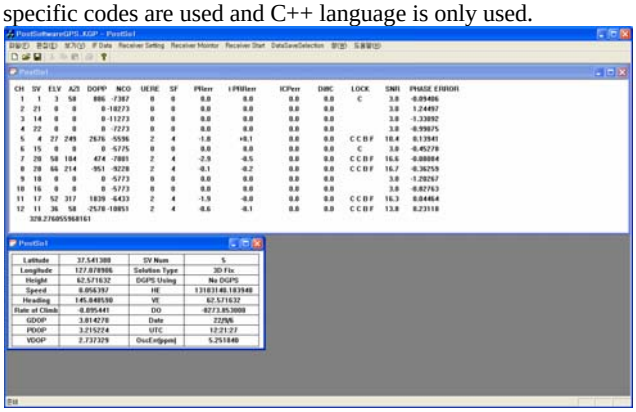

Figure 11. Monitoring graphic of implemented system

We test the implemented real-time software GNSS receiver. Figure 12~13 shows the position error from mean position. Table 2 shows the standard deviation of position error. In the result, the real-time software GNSS receiver, which the newly proposed correlation algorithms is adopted, works the same as conventional hardware receiver in general environment.

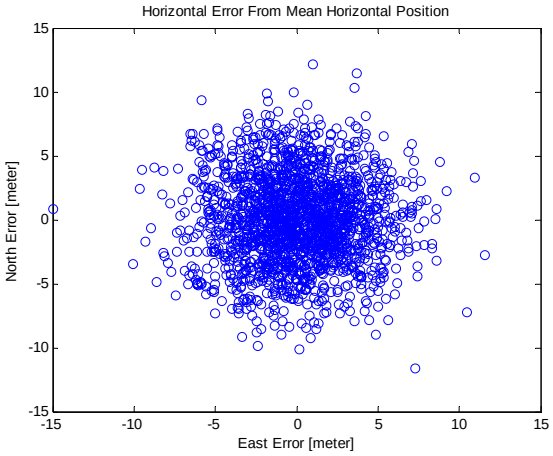

Figure 12. Horizontal position error

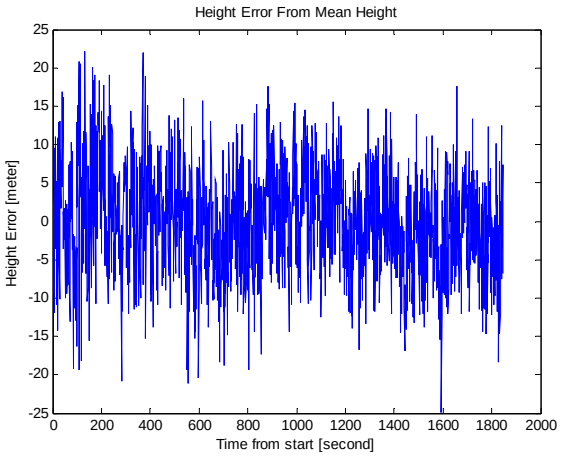

Figure 13. Height error

Table 2. Standard deviation of errors

	Standard deviation
East	3.1597
North	3.4301
Height	6.8088

3. Conclusion

In this paper, for implementing a real-time software GNSS receiver we propose the highly efficient and fast algorithms such as partial down-conversion, phase rotator, composite I&Q accumulation, Virtual DCO technique, and parallel acquisition using FFT. Noteworthy, when the proposed algorithms are used, more 30 tracking channels with 3 tracking arms(early-prompt-late) are operated real-time on Intel 2.8GHz personal computer which the hyper-threading is working(when the hyper-threading is working, the application don't use the full resource because of limitation of resource). Also, the partial down-conversion reduces the FFT size, for parallel acquisition, to 1/8 of conventional FFT-size and the program size includes map is not exceed 1Mbyte. Finally, the proposed real-time software GNSS receiver using the proposed algorithms provides the navigation solution with below 10 meter rms error.

Acknowledgement

This research was supported by BK21 project and SEM(Samsung Electro-Mechanics) in Korea. The authors wish to thank for their technical and financial supports.

Reference

1. B. W. Parkinson, J. J. Spiler Jr, *GPS Positioning System : Theory and Application*, AIAA, 1996.
2. Kaplan E. D., *Understanding GPS : Principles and Applications*, Artech House Publisher, Norwood, MA, 1996.
3. James Bao-Yen Tsui, *Fundamentals of Global Positioning System Receivers : A Software Approach*, John Willey&Sons, INC., 2000.
4. B.M. Ledvina, P.M. Kintner and M.L. Psiaki, "A 12-Channel Real-Time GPS L1 Software Receiver", ION GPS 2003, 2003.