Development of a Prototyping Platform for Software GPS Receiver

Falin WU*, Nobuaki KUBO**, Akio YASUDA** and Harumasa HOJO**

Abstract

In this paper we develop a prototype software GPS receiver. The receiver consists of a RF front-end, an ADC, and a software GPS program that runs on PC. The RF front-end down-converts the signal from RF to IF, and the ADC samples the IF signal. All the other processing including signal acquisition, tracking, data decoding, and solving position are all implemented in software using signal processing techniques. The local C/A codes and carrier replica signal are pre-generated, stored in memory, and used repetitively during signal acquisition and tracking. The concepts of the prototype software GPS receiver can be applied to next generation GNSS and SBAS receivers design.

Keywords: Software GPS receiver, GNSS, QZSS

1. INTRODUCTION

The modernization of the Global Positioning System (GPS)\(^1\), the revival of Russian GLONASS and the advent of European Galileo system\(^2\) will construct the next generation Global Navigation Satellite System (GNSS). To augment GPS for aviation users the U.S., Europe and Japan have deployed the Wide Area Augmentation System (WAAS), the European Geostationary Navigation Overlay System (EGNOS) and MTSAT Satellite Augmentation System (MSAS), respectively. Each of these Space Based Augmentation System (SBAS) services is free and interoperable, providing coverage for most of North America, most of Europe and Japan, respectively. At same time Japan launches a new regional SBAS plan, Japanese Quasi-Zenith Satellite System (QZSS)\(^3,4,5\), a constellation of three to seven satellites that will broadcast GPS signals from orbits optimized for East Asian hemisphere. Although GNSS and SBAS will provide users more navigation satellite signals, it is a challenge for users to develop the next generation GNSS receivers. Currently GPS receivers consist of a radio frequency (RF) front-end, an Applications Specific Integrated Circuit (ASIC) for signal processing, and a CPU core for higher level functions. Firmware can be loaded into the CPU to change the performance parameters of the receiver. However, design flexibility is very constrained by an ASIC that is hardwired with predefined tracking channels, correlator and control loop characteristics. So a software GNSS receiver is the best solution for next generation GNSS receivers design. Software GNSS receivers are those that implement signal acquisition and tracking process not in hardware, but in software\(^6,7,8,9,10,11\). Each of GNSS, although different, operates in a similar manner and there is much commonality in the required underlying signal processing. As such, the software GPS receiver architecture will be used throughout this paper as an example but the concepts can be applied to any

* Student Member: Tokyo University of Mercantile Marine (2-1-6, Etchujima, Koto-ku, Tokyo 135-8533), fwu@e.kaiyodai.ac.jp
** Member: Tokyo University of Mercantile Marine (2-1-6, Etchujima, Koto-ku, Tokyo 135-8533)
software GNSS receivers design.

To design the next generation GNSS receiver, a prototype personal computer (PC) based and intermediate frequency (IF) sampling L1 software GPS receiver is developed in this paper. Firstly, the structure of software GPS receiver and data collection hardware will be presented in Sec. 2. Secondly, the GPS Coarse/Acquisition (C/A) code acquisition will be described in Sec. 3. Thirdly, the GPS signal tracking will be discussed in Sec. 4. Finally, the conclusion will be given in Sec. 5.

2. SOFTWARE GPS RECEIVER

Software GPS receiver is to use an analog-to-digital converter (ADC) to change the input signal into digital data at the earliest possible stage in the receiver(6). Software GPS receivers have been developed for a number of years and different classes of software GPS receivers have emerged. For example, in References (7) and (8), a GPS receiver has been implemented using a software radio approach; Block Adjustment of Synchronizing Signal (BASS) has been introduced in References (6) and (9); a complete tutorial about IF software GPS has been given in Reference (11); a high-bandwidth GPS/Galileo signals simulator has been developed in Reference (12). All these software GPS receivers can be grouped in four classes: PC based genuine software GPS receiver, digital signal processors (DSP) based genuine software GPS receiver, field programmable gate array (FPGA) based software GPS receiver, and simulation tools. Software GPS receivers can also be divided into RF sampling and IF sampling GPS receiver. The basic hardware requirements include an antenna, a low noise amplifier (LNA), and ADC in both cases. Only down conversion can be left out for a RF sampling receiver(12).

Fig. 1 illustrates a general structure of software GPS receiver(12). The signals transmitted from the GPS satellites are received from the antenna. Through a RF front-end the input signal is amplified to a proper amplitude and the frequency is converted to a desired output frequency. An ADC is used to digitize the input signal. The antenna, RF front-end, and ADC are the hardware used in the software GPS receiver. After the signal is digitized, software is used to process it. Acquisition means to find the signal of a certain satellite. The tracking program is used to find the phase transition of the navigation data. From the navigation data phase transition, the sub-frames and navigation data can be obtained. Ephemeris data and pseudorange can be obtained from the navigation data. The ephemeris data are used to obtain the satellite positions. Finally, the user position, velocity, and time can be calculated(6).

Table 1 summarizes the characters of the prototype software GPS receiver developed in this paper. The GPS C/A code signals are down-converted from 1575.42 MHz to 15.42 MHz and sampled with a sampling frequency from 2 MHz to 20 MHz in an increment of 1 KHz using a GPS SIGNAL TAP of Accord Software and Systems Private Company. The signal samples can be processed in real-time and/or stored on the hard disk for post processing(13). In
future work we will build a new front-end to sample the civil signals of modernized GPS, GLONASS and Galileo.

3. GPS C/A CODE ACQUISITION

3.1 GPS C/A Code Properties and Generation

The C/A code is a bi-phase modulated signal with a chip rate of 1.023 MHz. The null-to-null bandwidth of the main lobe of the spectrum is 2.046 MHz. Each chip is about 977.5 ns (1/1.023 MHz) long. The total code period contains 1,023 chips. With a chip rate of 1.023 MHz, 1,023 chips last 1 ms; therefore, the C/A code is 1 ms long. The code repeats itself every millisecond.

The GPS C/A signals belong to the family of Pseudo-Random Noise (PRN) codes known as the Gold codes(6). Different C/A codes are used for different satellites. The signals are generated from the product of two 1,023-bit PRN sequence G1 and G2. The positions of the feedback circuit determine the output pattern of the sequence. The feed back of G1 is from bits 3 and 10. The feed back of G2 is from bits 2, 3, 6, 8, 9 and 10. The corresponding polynomials are

\[ G1: 1 + x^3 + x^{10}, \]
\[ G2: 1 + x^2 + x^3 + x^6 + x^8 + x^9 + x^{10}. \]

The local C/A codes are pre-generated, stored in memory and used repetitively during signal acquisition and tracking. One of the most important properties of the C/A codes is their correlation result. High autocorrelation peak and low cross-correlation peaks can provide a wide dynamic range for signal acquisition(6). The autocorrelation of C/A codes of SV 12 and the cross correlation of SV 19 and SV 31 are shown in Fig. 2. These satellites are arbitrarily chosen. The maximum of the autocorrelation peak is 1023, which equals the C/A code length. The position of the maximum peak is deliberately shifted to the center of the figure for a clear view. The rest of the correlation has three values 63,-1 and -65. The cross-correlation has three values 63,-1 and -65.

3.2 GPS C/A Code Acquisition

The acquisition algorithm is used to determine whether a certain satellite is in the input signals, and so determine its code phase and carrier frequency. In this section two different acquisition methods: conventional approach and fast Fourier transform (FFT) approach will be described.

The conventional approach performs signal acquisition in time domain. A noncoherent correlator in time domain, shown in Fig. 3, is used since the phase of received signal is random(14). The correlation is approximated by the discrete sum,

\[ R^2[m] = \sum_{j=0}^{K-1} \left( \sum_{n=jL}^{(j+1)L-1} x[n] \cdot CA[n+m] \cdot \cos[\Omega n] \right)^2 \]
\[ + \sum_{j=0}^{K-1} \left( \sum_{n=jL}^{(j+1)L-1} x[n] \cdot CA[n+m] \cdot \sin[\Omega n] \right)^2, \]

where \( \Omega \) is radian frequency. First the digital IF, \( x[n] \), is multiplied by the replicated C/A code, \( CA[n+m] \). Here \( n \) represents the \( n^{th} \) sample and \( m \) represents the number of samples the replicated C/A code is phase shifted. \( L \) is the samples length of one C/A code period. After the code removal the in-phase \( I \) and quadrature-phase \( Q \) components are generated. The \( I \) and \( Q \) components are accumulated for one or more code periods, \( N \). The accumulated
Fig. 4 Noncoherent correlator in frequency domain

The sum is squared. Next $K$ correlations are accumulated to produce an averaged correlation point. If the correlation point is larger than a certain threshold it is assumed that the satellite is acquired.

The conventional approach can also be performed in circular convolution. The received data are correlated with the replica code by circularly shifting the replica code. This resembles the circular convolution and may be expressed as

$$R[m] = \sum_{n=0}^{L-1} x[n] \cdot CA[(n+m)_L]$$

The FFT approach performs circular convolution in the frequency domain. The discrete Fourier transform (DFT) and its inverse is used to calculate $R$,

$$R[m] = x[n] \otimes CA[-n]$$

$$= F^{-1} \left( F(x[n]) \cdot F(CA[n])^* \right)$$

where $F$ and $F^{-1}$ denote FFT and inverse FFT. A noncoherent correlator in frequency domain can be adapted to the acquisition of GPS signals as shown in Fig. 4. Here the input signal is mixed to baseband and the $I$ and $Q$ components are used as the real and imaginary inputs when calculating the DFT. The result is multiplied by the complex conjugate of DFT of the C/A code. The circular convolution is obtained by taking the magnitude of the inverse DFT. The fast Fourier transform algorithm is used to implement the DFT and inverse DFT; hence the acquisition method may be called FFT approach.

Fig. 5, 6, 7 and 8 show the results of acquisition using FFT approach. Fig. 5 shows the correlation matrix for SV 4. After the circular correlation, the beginning of the C/A code of SV 4 is shown in Fig. 6. The beginning of the C/A code is at 7547. The
amplitudes of 21 frequency components of SV 4 separated by 1 KHz are shown in Fig. 7. The highest component occurs at 4.578 MHz. Fig. 8 illustrates how the signal to noise ratio increases as K increase. For clarity the correlation powers in Fig. 8 are normalized and the acquisition is performed at a fixed replica carrier frequency.

4. GPS SIGNAL TRACKING

Once a signal is acquired, the signal must be tracked in order to obtain the navigation data. The tracking program uses two parameters obtained from the acquisition process: the beginning of the C/A code period and the carrier frequency of the input signal to start the processing. Two loops are needed to track one GPS satellite signal. One loop is often referred to as code loop, which tracks the C/A code. The other one is phase (or frequency) locked loop, which tracks the carrier frequency of the down-converted input signal. These two loops must be coupled together as shown in Fig. 9.

The code loop uses three locally generated C/A code to track the C/A code of the input signal. The three locally generated codes are usually used: a prompt, an early and a late replica C/A code. The early and late replica C/A codes are the prompt replica C/A code shifted a few samples (e.g. samples in 1 ms) to the right and left, respectively. The prompt code is applied to the digitized input signal and strips the C/A code from the input signal. The output will be a continuous wave (cw) signal with phase transition caused only by the navigation data. This signal is applied to the input of the carrier loop.

The output from the carrier loop is a cw with the carrier frequency of the input signal. This signal is used to strip the carrier from the digitized input signal. The output is a signal with only a C/A code and no carrier frequency, which is applied to the input of the code loop.

4.1 Code Tracking

The code loop is known as a delay lock loop (DLL). The prompt code is intended to match the beginning of the C/A code in the input signal. The correlation outputs from three codes can be used to accurately determine the beginning of the C/A code in the input signal. This information is used to adjust the initial phase of the locally generated prompt code to better match the code phase of the input signal. Fig. 10 shows the correlations between input C/A codes and local replica C/A codes. The discriminator algorithm is to give a calculated number on the code phase error. The discriminator output signal \( \varepsilon \) is calculated,

\[
\varepsilon = \frac{y_E}{y_L}, \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots (5)
\]

where \( y_E \) and \( y_L \) are the correlation powers of early code and late code, respectively. If \( \varepsilon = 1 \), the prompt code is perfectly aligned with the C/A code in the input signal; if \( \varepsilon > 1 \), the local codes should be shifted to the right; if \( \varepsilon < 1 \), the local codes should be shifted to the left.
4.2 Carrier Tracking

The phase locked loop is used to track the input IF signal, which is obtained from the acquisition program. The locally generated reference IF signal for the tracking program should be closer to the input IF signal.

A second-order loop can track the rate of change in Doppler and a third-order loop can be used to track constant acceleration\(^6\). For a software GPS receiver, the phase and frequency of the reference signal are adjusted every ms or 10 ms. This operation may put the IF of the locally generated signal very close to the IF of the input signal, but may not be exactly equal to it. In this paper a second order PLL is used as the filter to track the input IF signal. The filter function is

\[
F(z) = \frac{(C_1 + C_2) - C_2 z^{-1}}{1 - z^{-1}}, \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots (6)
\]

where \(C_1\) and \(C_2\) are coefficients,

\[
C_1 = \frac{1}{k_0 k_1} \frac{8 \zeta \omega_n t_s}{4 + 4 \zeta \omega_n t_s + (\omega_n t_s)^2}, \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots (7)
\]

\[
C_2 = \frac{1}{k_0 k_1} \frac{4 (\omega_n t_s)^2}{4 + 4 \zeta \omega_n t_s + (\omega_n t_s)^2}, \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots (8)
\]

where \(\omega_n = 2 \beta / (\zeta + 1/4 \zeta)\), \(\omega_n\) is natural frequency, \(\zeta\) is damping ratio, \(\beta\) is bandwidth, \(t_s\) is the sampling interval. The amplifier \(k_0\) represents the gain of the phase comparator and the low-pass filter limits the noise in the loop, and \(k_i\) is the gain of the voltage-controlled oscillator (VCO)\(^6,16\).

Fig. 12 shows outputs from conventional tracking method of SV 4. The amplitude changes with time; this is the transit effect of the tracking loop. Finally, the amplitude reaches a steady state.
5. CONCLUSION

In this paper we have developed a prototype PC based and IF sampling software GPS receiver. Two different acquisition methods: conventional approach and FFT approach have been presented. The code and carrier tracking methods have been described. The concepts of the prototype software GPS receiver can be applied to the next generation GNSS and SBAS receivers design. In future work we will develop a genuine software GNSS and SBAS receiver to acquire and track the signals of modernized GPS, GLONASS, Galileo and QZSS.

ACKNOWLEDGEMENT

The authors would like to acknowledge Dr. Thomas Pany at the Institute of Geodesy and Navigation of the University FAF Munich, Germany, for discussions on software GNSS receiver.

REFERENCES


(14) F. Johansson, R. Mollaei, J. Thor, and J. Uusitalo,
"GPS Satellite Signal Acquisition and Tracking."
Division of Signal Processing, Luleå University of Technology, S-971 87 Luleå, Sweden,


QUESTIONS AND ANSWERS

Yasuo ARAI (Marine Technical College):
How about the performance of the software GPS receiver what you developed?

Falin WU:
The performance of the software GPS receiver depends on a lot of facts, such as sampling frequency, acquisition approach, and tracking algorithm. We only developed a prototype software GPS receiver, and we will test its performance in future work.

Ivan PETROVSKI (The Institute of Advanced Satellite Positioning Technology):
What language did you use for the development?

Falin WU:
We used MATLAB to develop the software GPS receiver, and we will code it in C/C++ after the algorithms have been validated.