# DESIGN OF A SOFTWARE GPS RECEIVER AND ITS MATLAB IMPLEMENTATION

**Yun Zhao, Qishan Zhang**

**Beijing University of Aeronautics and Astronautics, Beijing, P.R. China**

## ABSTRACT

The embedded system related hardware technology has experienced rapid development, and it provided the software technology with a huge space for growth. Therefore using software approaches to perform GPS receiver functions in a powerful and generic hardware platform is becoming more feasible.

In this paper, the software GPS receiver technology and the design basics of the software receiver are discussed. Further in the Matlab simulation environment, the implementation of a software receiver for replacing the processing functions of ASIC in traditional GPS receivers, i.e. RF front end and multi-channel correlator, is presented. Some simulation results and implementation details are included.

## KEYWORDS

Software GPS Receivers, Matlab, Embedded Systems.

## INTRODUCTION

As a result of the recent development in both embedded systems and wireless communications, nowadays the software GPS (Global Positioning System) receiver is becoming a technical trend of GPS receivers.

Although it's currently hard to define the software GPS receiver, it is generally considered to be the design pattern to which the software radio technology is applied. There are two primary design goals in developing a software radio application. First, the ADC (A/D converter) should be positioned as close to the antenna as possible. Second, the software should perform the radio functions as many as possible [1]. Consequently, the software GPS receiver typically uses a generic hardware platform while minimizing the dedicated hardware, and relies on software approaches to implement most of the receiver functions.

Compared to traditional GPS receivers, software GPS receivers offer many advantages. Some of these include that a closer position of the digital signal processing to the antenna allows the GPS measurements to be derived to high levels of accuracy, the software approach ensures a extensible, flexible and reconfigurable design which is easy to maintain and upgrade, and the introduction of the generic hardware platform simplifies the hardware components, and then the device cost is reduced and power saving is also achieved.

The paper begins by describing the software GPS receiver architecture. Then the design basics for the software receiver are discussed. Next, the core functions of the traditional GPS receiver RF front end and multi-channel correlator are implement in software models in Matlab. Finally, a discussion of this work is presented.

## SOFTWARE GPS RECEIVER ARCHITECTURE

The focus from here will be on the $L_1$ C/A-code GPS receiver. In the traditional receiver design we have implemented previously, a GPS antenna, a RF front end ASIC (GP2010), a 12-channel correlator ASIC (GP2021) and a programmable DSP processor are employed. And the software on the DSP processor performs signal acquisition and tracking, and navigation processing to provide the position solution.

According to the current level of the hardware technology development, it is impossible to move the ADC to the antenna yet, and then the RF front end in hardware form is still necessary. The hardware components of which the software GPS receiver consist include the antenna, the RF front end, and a programmable DSP（digital signal processor）of high performance or a generic PC platform on which all the processing is done in software. The architecture of the software GPS receiver is illustrated in Figure1.

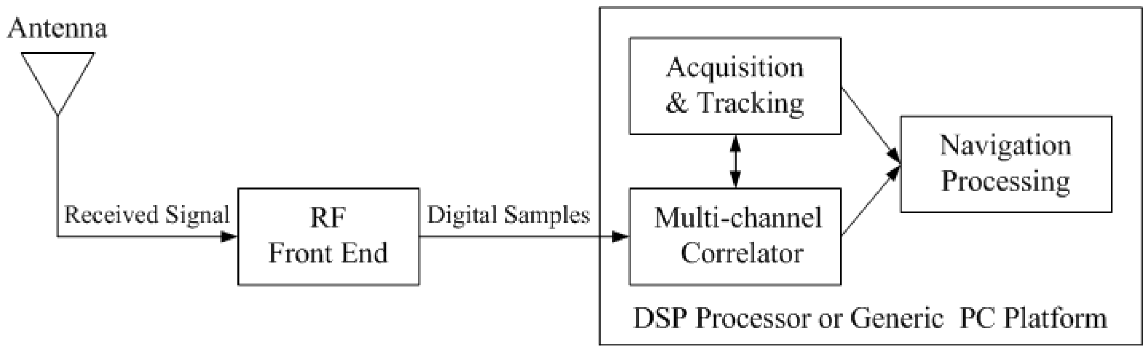


Figure 1. Architecture of the software GPS receiver

For both the DSP processor and the generic PC platform, the software running on them consists of the following components: multi-channel correlaor, signal acquisition and tracking, and navigation processing.

The generic PC platform can be a general-purpose computer, a laptop, or an embedded system product like PDA or mobile phone. Considering the current limitation of real-time processing for the microprocessor of the generic PC platform, the multi-channel correlaor is usually accomplished in a firmware, prior to the microprocessor, for accelerating the correlation operation.

## BASICS OF SOFTWARE GPS RECEIVER DESIGN

The module-based design is important for the software GPS receiver application, which allows the receiver to be quickly and easily adapted to meet a broad range of current and future requirements, such as to support characteristic multiple frequencies and new codes signals of the next generation GPS satellites.
The rough modules for the software GPS receiver are described below.

*RF front end module*

Although performed in hardware, it is required to use minimal components and be of low power. The traditional receiver usually consists of multiple stages of frequency down-conversion (there are 3 stages frequency down-conversion in GP2010). Since multiple stages introduce additional analog components with some potential negative consequences, a single or two stages frequency translation is preferred in this front end design. However the final IF frequency can be much higher than that of the traditional RF front end, to reach dozens of MHz typically.

*Multi-channel correlator module*

In the traditional receiver, it is performed in dedicated hardware. As the software based correlator, a challenging aspect is high throughput. For instance, 2MHz bandwidth signal will produce $2 \cdot 10^6$ complex samples per second. Approximately the same amount of local replica samples needs to be generated along with correlation calculation. Furthermore in a multi-channel receiver, where several independent correlation results are to be managed, this real time task gets hardly to be accomplished even with modern DSP processors [2]. Therefore, the DSP processor is required to be of high throughput, and the algorithms involved are required to be faster.

*Signal acquisition and tracking module*
*Navigation processing module*

The two modules are responsible for acquiring and tracking the GPS satellites, and calculating position when a sufficient number of satellites are tracked. The processing involved are complicated and also extremely important to the receiver, however the design is similar to that in the tradition receiver, so out of the scope of this paper.

## MATLAB IMPLEMENTATION

Having served as key ASIC for traditional GPS receivers, the RF front end GP2010 and the multi-channel correlator GP2021, in particular the signal processing functions, are implemented in software in the Matlab simulation environment.

*RF front end module*

A generic representation of the GP2010 is shown in Figure 2. The GP2010 receives the 1575.42MHz RF signal from a GPS antenna including a LNA, and converts it to a 4.309MHz IF via three stages of frequency down-conversion and the corresponding filters. The 4.309MHz IF is an output and further fed to a 2-bit quantiser, which produces sign and magnitude outputs. If the GP2010 is used in conjunction with the GP2021, then the GP2021 provides a sampling clock of 5.714MHz. This converts the IF to a 1.405MHz 2-bit digital output. The AGC loop controls the quantiser, and allows the distribution of the digital output with a numerical variance of 3.4 and a mean of 0 [3].

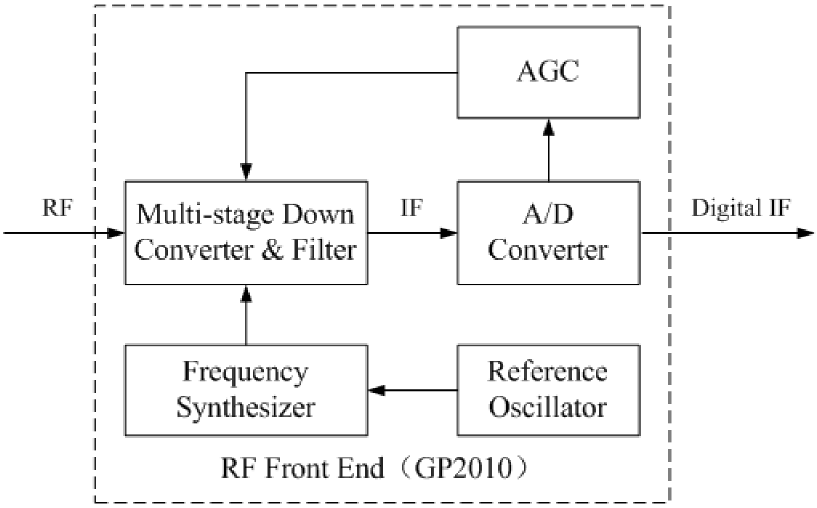


Figure 2. Generic GPS RF front end

A received satellite signal model is required for the simulation, and the received signal from the antenna is down-converted directly to the final IF in the software implementation, i.e. a single stage frequency translation is accomplished. Based on the above-described principles of the GP2010, the final model for the received signal from $i$ th GPS satellite at the point of sampling only with the receiver clock error is defined by [4]:

$$R_{L1DC} = \sqrt{2P_r}\, G_i(t_r - T_d - \delta t_r) D_i(t_r - T_d - \delta t_r) \cos\{(\omega_{IF3} - \frac{\Delta f}{f} f_{TDC}) t_r - \omega_{L1}(T_d + \delta t_r)\} + n(t_r) \quad (1)$$

Where $P_r$ = Received signal power  $G_i(t)$ = PRN spreading sequence

$D_i(t)$ = Data message modulation  $\omega_{L1}$ = $L_1$ carrier frequency

$t_r$ = Receiver clock time  $T_d$ = Signal propagation time(delay)

$\delta t_r$ = Advance of the receiver clock with respect to system time

$\omega_{IF_3}$ = Sampled IF frequency  $\Delta f / f$ = Receiver clock frequency error

$f_{TDC}$ = Total frequency down conversion from $L_1$ to sampled IF

$n(t_r)$ = Filtered noise, modeled as additive white Gaussian noise

This model is implemented in software, which generates each quantity in Equation (1) respectively, and then synthesizes the final signal gradually. Some intermediate simulation results, prior to the quantization, are shown in Figures 3 ~ 6, which indicate that the software model is effective and accurate.

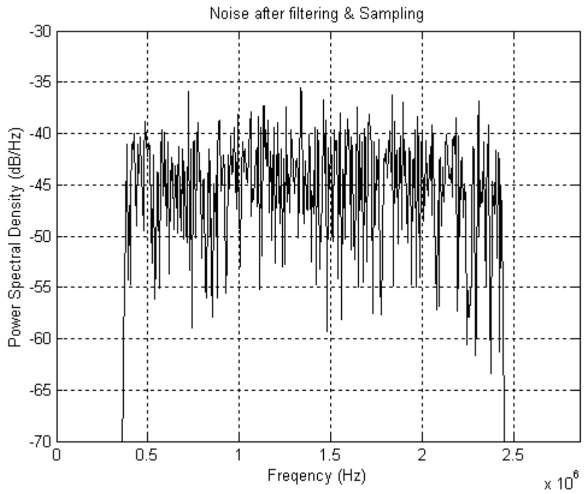


Figure 3. PSD of noise after filtering & sampling

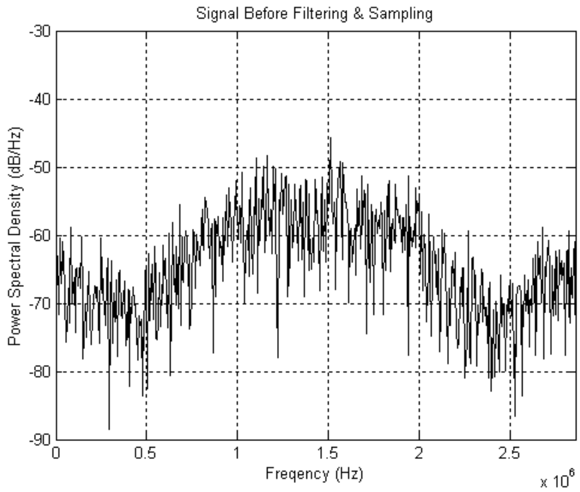


Figure 4. PSD of signal before filtering

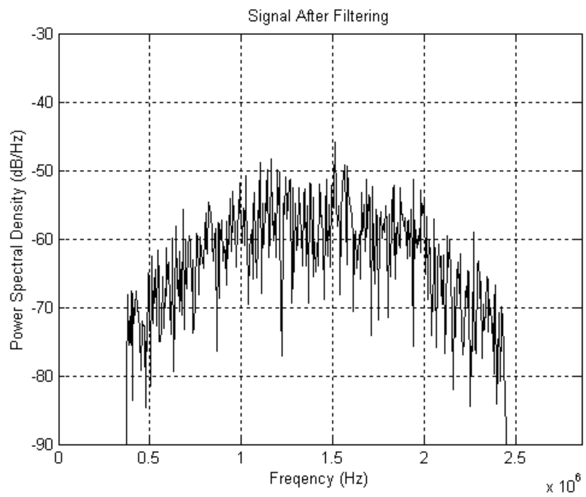


Figure 5. PSD of signal after filtering

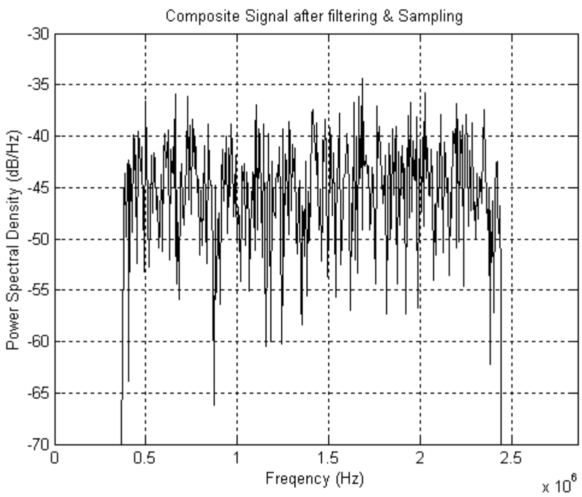


Figure 6. PSD of signal containing noise and clock error after sampling

The 2-bit quantiser in software converts the analog IF to digital outputs, which uses an iterative technique for simulating the AGC function to implement a 15% occurrence of +3 or -3, and the 35% occurrence of +1 or -1.

*Multi-channel correlator module*

A block diagram of code and carrier tracking is shown in Figure 7, in which the left represents a single channel of the multi-channel correlator like GP2021, while the right represents the carrier and code tracking loops [5,6]. Note the carrier DCO and the code DCO should be included in the channel, but the correction for either DCO depends on the corresponding loop control.

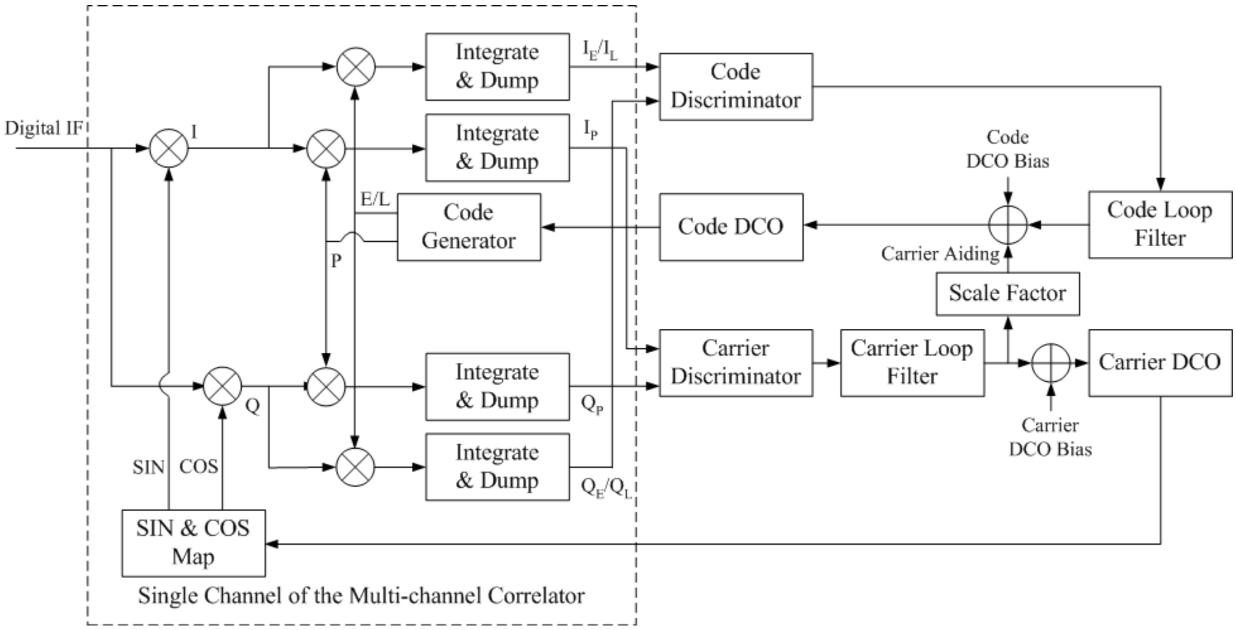


Figure 7. Block diagram of code and carrier tracking

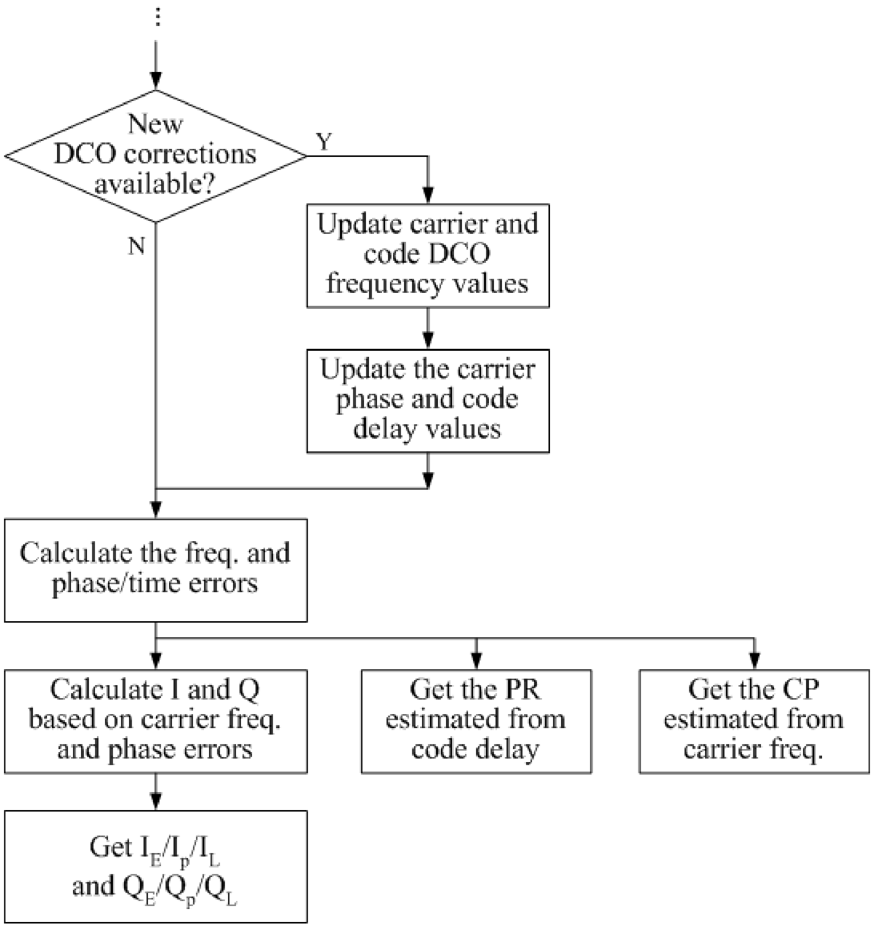


Figure 8. The software flow for correlator

For the GP2021, there are two correlators per channel. The correlator receives digital data from the RF front end, and processes the data for early, prompt and late code phase tracking. The results are accumulated and output. The correlator also performs the measurement for pseudo-range and carrier phase calculations.
The software for correlator is the model, which generates the accumulated I (In-phase) and Q (Quadrature-phase) samples of the received signal, the pseudo-range (PR) and carrier phase (CP) measurements. A proven mathematic model for approximating the accumulated I and Q values as a function of carrier DCO frequency and phase errors is used, which is explained in detail in [7]. Furthermore, the software model generates the early, prompt and late I and Q based on the accumulated I and Q values, according to the code misalignment. The software flow is shown in Figure 8.

## CONCLUSIONS

The core processing functions of the traditional GPS receiver RF front end and multi-channel correlator have been implemented in the software models in Matlab. The work involved demonstrates it is possible to replace the hardware in traditional GPS receivers with software implementation, and futher accomplish the complete software based GPS receiver.
For these software models, there is a much room for enhancement and further validation, and there is still a considerable distance to the actual software GPS receiver. Nonetheless, the implemented models reveal a huge potential of the software approach for the GPS receiver implementation.

## REFERENCES

[1] Niu, X.X., Yang, Y.X., Software radio technology and applications, Beijing university of posts and telecommunications press, Beijing, China, Sept. 2000, pp1-8.
[2] Botchkovski, A.L., Chistyyakov, V.V., Golubev, M.A. and et al, "SoftFlex: An Advanced Approach to Design of GNSS Receiver with Software Correlator", Proc. of ION GPS'99, Nashville, TN, US, 14-17 Sept. 1999, pp353-362.
[3] Zarlink Semiconductor, "GP2010 GPS receiver RF front end", DS4056 Issue3.5, Feb. 2002.
[4] Corbell, Phillip M., "Design and Validation of an Accurate GPS Signal and Receiver Truth Model for Comparing Advanced Receiver Processing Techniques", MS thesis, Air Force Institute of Technology and Management, Wright-Patterson AFB, OH, US, Mar. 2000.
[5] Kaplan, E.D., Understanding GPS: principles and applications, Artech House Publishers, Norwood, MA, US, 1996, pp 75-97.
[6] Zarlink Semiconductor, "GP2021 GPS 12-Channel Correlator", DS4077 Issue3.2, Apr. 2001.
[7] Ward, Phillip W., "Using a GPS Receiver Monte Carlo Simulator to predict RF Interference Performance", Proc. of ION GPS'97, Kansas City, US, Sept. 1997, pp1473-1482.