Design & Realization of a Low Power DSP Architecture for GPS Receivers

Elham Firouzi

THÈSE SOUMISE À LA FACULTÉ DES SCIENCES
DE L'UNIVERSITÉ DE NEUCHÂTEL POUR L'OBTENTION
DU GRADE DE DOCTEUR ÈS SCIENCES

IMPRIMATUR POUR LA THESE

Design and Realisation of a Low-Power DSP Architecture for GPS Receivers

de Elham Firouzi

UNIVERSITE DE NEUCHATEL

FACULTE DES SCIENCES

La Faculté des sciences de l'Université de Neuchâtel sur le rapport des membres du jury,

MM. F. Pellandini (directeur de thèse), N. de Rooij, R. Riem-Vis et P.-A. Farine (Asulab, Marin)

autorise l'impression de la présente thèse.

Neuchâtel, le 2 octobre 2001

Le doyen:

J.-P. Derendinger

To my dear wife Catherine

and our son Lucas

Abstract

The Global Positioning System (GPS) provides accurate, continuous, worldwide, three-dimensional position and velocity information to users with the appropriate receiving equipment.

The fundamental navigation technique is to use one way ranging from the GPS satellites that also broadcast their estimated positions. With four satellites and appropriate geometry, four unknowns can be determined. Typically, these are latitude, longitude, altitude and a correction to the user clock.

With the enhancement and progresses in portable microelectronic devices (decrease of the power consumption; higher density and frequency for the integrated circuits), miniaturized GPS receivers are appearing and used in car navigation systems, cellular telephones or even watches.

The scope of this PhD research is the design and the realization of a low power Digital Signal Processing (DSP) architecture for GPS receivers. The DSP decodes the GPS signals transmitted by the Space Vehicles (SV) and transmits the GPS messages to the navigation unit. It occurs after the antenna and the radio frequency unit and until the processing of the GPS messages and the actual computation of the navigation solution. In most literature, the DSP is considered as the core of the GPS receiver and its performances have an important impact on the receiver characteristics.

The objectives were to be able to use the DSP architecture in small-volume portable applications like wristwatches. The overall system power requirement for the GPS receiver is in order of 100 mW. In order to be able to meat these severe requirements, this PhD report proposes a total

review of the DSP architecture and algorithms. Possible optimizations at the technology level are not the scope of this thesis.

The realization approach is based on Field Programmable Gate Array (FPGA) prototyping. In this manner, the circuit performances can be verified in real time before the achievement of an Application Specified Integrated Circuit (ASIC). It provides an efficient solution compared to a modeled approach, which would require very long simulation times.

The consumption of the DSP architecture proposed in this thesis is around 20 mW. Since the DSP architecture contains most of the GPS signal processing algorithms, the power consumption is good despite the common technology (e.g. 0.5-micron). Adding the radio frequency and navigation units, the total power consumption of the GPS receiver remains below 100 mW as targeted.

Résumé

Le Système de Positionnement Global (SPG abrégé en anglais par GPS), mis en place par le département américain de la défense dans les années 1970, permet de déterminer les coordonnées géographiques d'un point quelconque 24h sur 24h.

L'approche se base sur la mesure du temps de vol des ondes radios, émises par les satellites GPS. Ce procédé permet de déterminer l'éloignement de l'utilisateur par rapport aux satellites, dont les trajectoires sont connues. Quatre satellites sont nécessaires pour établir la position par triangulation avec une précision de $\pm$ 10 m.

Les progrès importants dans le domaine de la micro-électronique portable (diminution de la consommation; augmentation de la densité d'intégration et de la fréquence de travail) ont contribué à l'apparition des récepteurs GPS miniatures. Ces derniers sont utilisés dans des systèmes de navigation pour voitures, des téléphones cellulaires et même des montres bracelets.

Le but de cette recherche est la conception et la réalisation d'une architecture de Traitement Numérique du Signal (TNS abrégé en anglais par DSP) à basse consommation destinée aux récepteurs GPS. Le DSP décode les signaux GPS transmis par les satellites et fournit les messages GPS à l'unité de navigation. Dans la plupart des littératures, le DSP est considéré comme le cœur du récepteur et ses performances ont un impact important sur les caractéristiques du récepteur.

L'objectif est l'intégration de l'architecture DSP dans les applications portables et miniatures tel que les montres bracelets. La consommation totale du récepteur GPS ne devrait pas dépasser les 100 mW. Affin de respecter ces contraintes sévères, cette thèse propose des optimisations

architecturales et algorithmiques du DSP. Les éventuelles améliorations technologiques ne font pas parties des buts de ce travail.

L'approche de la réalisation est basée sur le prototypage avec des circuits logiques programmables. Ainsi, le comportement du DSP peut être vérifié en temps réel avant la réalisation d'un circuit intégré. Ce procédé est beaucoup plus efficace que l'approche par modélisation, qui nécessite des temps de simulation très longs.

La consommation de l'architecture DSP proposée dans cette thèse se situe autour de 20 mW. Comme l'architecture DSP contient la plus grande partie des algorithmes du traitement de signal GPS, ce résultat est bon en dépit de la technologie standard utilisée pour l'intégration (0.5 microns). En ajoutant les unités de radio fréquence et de navigation, la consommation totale du récepteur GPS reste en dessous de 100 mW selon les objectifs.

Acknowledgments

First of all, I wish to express my gratitude to Professor Fausto Pellandini, Head of the Electronics and Signal Processing Laboratory at the Institute of Microtechnology (IMT), for giving me the opportunity to work in his group and for accepting to supervise the writing of this thesis. I was privileged to have made this research work with such an extraordinary human and scientific person as him.

My effusive thanks go to Dr. Ruud Riem-Vis, for his interest in my work, for his availability and for sharing his large scientific knowledge. I am very grateful to him for his constant support, the comments and suggestions, which indeed improve the quality of this thesis.

I express my recognition to Professor Nico de Rooij and Dr. Pierre-André Farine, for their time and their efforts in evaluating this dissertation as members of the jury. Their positive feedbacks during the private examination are also largely appreciated.

I would also like to thank all my colleagues, who make IMT a very agreeable place to work. I think in particular to our respective system managers Heinz Burri and Laurent Jeanrenaud. Both let the whole infrastructure run very well, despite our continuously changing requests.

The work reported in this dissertation results from research supported by the Commission of Innovation and Technology (CTI 3050), the IMT and Asulab (the R&D Laboratories of the Swatch Group). Their supports are here thankfully acknowledged.

Finally, I express my gratefulness to my family, in particular to my parents for their supports and awaking my interests for scientific matters, and to my family-in-law for their encouragements. My personal thanks are going to my dear wife Catherine and our son Lucas, who make life even more exciting that just microtechnology research.

Table of Contents

Chapter 1

Introduction

This thesis draws the design and the realization of a low power Digital Signal Processing (DSP) architecture for Global Positioning System (GPS) receivers, which are dedicated to small-volume portable applications.

The strong constraints on the power consumption have led to a total review of the receiver architecture and algorithms. Possible optimizations at the technology level are not the scope of this thesis.

1.1 Introduction to Radio Navigation

Navigation is defined as the science of getting a craft or a person from one place to another. For most of us these skills require utilizing our eyes, common sense and landmarks. However, in some cases where more accurate knowledge of position is required, navigation aids other than landmarks are used. These may be in a form of a simple clock to determine the velocity over the known distance or the odometer in our car to keep track of the distance traveled. Some other navigation aids are more complex and transmit electronic signals. These are referred to as radio-navigation aids.

Signals from one or more radio-navigation aids enable persons to compute their position. It is important to note that it is the user's radio-navigation receiver that processes these signals and computes the position fixes. The receiver performs the necessary computations (e.g. range, bearing, estimate time of arrival) for the users to navigate to a desired location.

Various types of such aids exist. They can be categorized as either ground-based or space based. Mostly, the accuracy of ground-based systems is proportional to their operating frequency.

Early developed spaced-based systems (the United States (U.S.) Navy Navigation Satellite System – referred to as Transit - and the Russian Tsikada system) provide a two-dimensional high accuracy positioning service. Limitation application to both systems are that each position fix requires approximately 10 to 15 minutes of receiver processing and an estimate of the user's position. These attributes were suitable for shipboard navigation because of the low velocity, but not for aircraft and high dynamics. It was these shortcomings that led to the development of both the U.S. GPS and the Russian Global Navigation Satellite System (GLONASS). These space-based systems are in use today.

1.2 GPS Overview

The GPS provides accurate, continuous, worldwide, three-dimensional position and velocity information to users with the appropriate receiving equipment. The satellite constellation consists of 24 satellites arranged in 6 orbital planes with 4 satellites per plane. A worldwide ground control/monitoring network monitors the health and status of the satellites. This network also uploads navigation and other data to the satellites. GPS can provide service to an unlimited number of users since the user receivers operate passively. The system utilizes the concept of one-way Time Of Arrival (TOA) ranging.

GPS provides two services: the Standard Positioning Service (SPS) and the Precise Positioning Service (PPS). The SPS is designated for the civil community, whereas the PPS is slated for U.S. authorized military and selected government agency users. Access to the PPS is controlled through cryptography.

The PPS is specified to provide a predictable accuracy of at least 6.6 m in the horizontal plan (one sigma error when the Horizontal Dilution Of Precision HDOP=2.0) and 8.3 m in the vertical plane (one sigma error when the Vertical Dilution Of Precision VDOP=2.5) [Par94].

The SPS is available to all users worldwide. There are no restrictions on SPS usage. When the Selective Availability (SA) is on, this service provides predictable accuracy of 41.1 m in the horizontal plane and 51.4 m in the vertical plane (same conditions as above). With the discontinuation of the SA by the U.S. government in May 2000, the SPS horizontal error is 10.2 m and the vertical error 12.8 m [Par94].

1.3 GPS Application

Both GLONASS and GPS have evolved from dedicated military systems to true dual-use. Satellite navigation technology is being utilized in numerous civil and military applications that range from leisure hiking to spacecraft guidance. Numerous disciplines including all sectors of transportation have been affected. Users are no longer restricted to specific routes due to accuracy and/or coverage limitations of ground-based navigation aids. As long as a user is in line-of-sight to the satellites, accurate navigation is obtainable. To illustrate the diverse use of satellite navigation technology, several examples of current and projected applications are presented below.

1.3.1 Aviation

The aviation community has propelled the use of Global Navigation Satellite System (GNSS) and various augmentations to provide guidance for the route through precision approach phases of flight. The International Civil Aviation Organization (ICAO) defines a system that contains at least one or more satellite navigation systems as a GNSS. The continuous global coverage capability of GNSS permits aircraft to fly directly from one location to another provided factors such as obstacle clearance and required procedures are adhered to.

1.3.2 Maritime

Both the commercial and recreational maritime communities have embraced GNSS. Navigation is enhanced on all bodies of waters, from

oceanic travel to river-ways, especially in inclement weather. Several nations are developing local area differential GPS networks to increase system accuracy for harbor approach and river usage.

1.3.3 Land

The surveying community has relied on differential GPS to achieve measurement accuracy in the millimeter range. Similar techniques are in use within the railroad community to obtain train location with respect to an adjacent set of tracks. GPS is a key component in Intelligent Transportation Systems (ITS). In terms of vehicle applications, GNSS will be used for route guidance, tracking and emergency messaging. Integrating a GNSS receiver with a street database, digital moving map display and processor will allow the driver to obtain directions and/or the shortest most efficient route. Combining a cellular phone or data link function with this system will enable vehicle tracking and/or emergency messaging. A vehicle's position can be automatically reported to a control segment for fleet management. The activation of a "panic" button by the driver broadcasts an emergency message, vehicle characteristics and vehicle location to law enforcement authorities for assistance.

1.4 Scope of the Research

With the enhancement and progresses in portable microelectronic devices (higher density and frequency for the integrated circuits, and decrease of the power consumption), miniaturized GPS receivers are appearing and used in car navigation systems, cellular telephones or even watches (Casio).

The scope of this PhD research is the design and realization of a low power DSP architecture for GPS receivers. The DSP decodes the GPS signals transmitted by the Space Vehicles (SV) and transmits the GPS messages to the navigation unit. It occurs after the antenna and the radio frequency unit and until the processing of the GPS messages and the actual computation of the navigation solution. In most literature, the DSP is considered as the core of the GPS receiver and its performances have an important impact on the receiver characteristics.

The objectives were to be able to use the DSP architecture in small-volume portable applications like wristwatches. The overall system power requirement for the GPS receiver is in order of 100 mW. In order to be

able to meat these severe requirements, this PhD report proposes a total review of the DSP architecture and algorithms.

1.4.1 Signal Power

The constraints on the volume result in a reduced antenna size, which in turn is translated into lower signal levels. Therefore, a particular attention has been paid to be able to treat input signals with a low Signal to Noise Ratio (SNR). Compared to more ideal cases, the design had to be more sensitive and robust to ensure stable signal reception.

1.4.2 Power Consumption

To enable portable applications, much effort has been put on the reduction of the power consumption. From the many optimization methods that exist (e.g. technology, architectural and algorithmic), this thesis focuses on the architectural and algorithmic levels. This means that a potential exists to further reduce the energy that is required to operate the receiver.

1.5 Organization of the Report

After this short introduction, chapter 2 presents the GPS principles and applications. This chapter will accommodate the reader with the most important GPS receiver concepts, which should facilitate the understanding of the original contributions of this thesis.

Chapter 3 treats some aspects of the digital design and its power dissipation, which can be reduced by many existing optimization methods. At the end of this chapter, two original GPS examples of algorithmic optimization are presented. There are the complex sampling algorithm and the multiphase Numerical Controlled Oscillator (NCO).

The remaining chapters form the core of this work. Chapter 4 draws the design of a low power DSP architecture for GPS receivers. Chapter 5 presents the implementation of the DSP architecture into an Application Specified Integrated Circuit (ASIC), based on an effective “top-down” synthesis methodology.

The last chapter concludes the thesis with a summary of the most important results.

1.6 Publications & Patents

The innovative contributions of this thesis are protected by 4 patents, which have been deposed at the European Patent Office by Asulab (the R&D Laboratories of the Swatch Group). The title and a succinct description of each patent are presented bellow. However, for more details, the original documents should be consulted carefully.

Patent Nr 00202336.4-2206:

Low Power RF Receiver with Redistribution of Synchronization Tasks

This patent concerns the review of the receiver architecture presented at section 4.5.2, where all real-time DSP algorithms have been implemented into dedicated bit-parallel hardware. The claimed architecture contains 12 independent channels for the civilian L_1 band, which perform satellite acquisition and tracking without the help of an external microprocessor. When a channel is locked onto a satellite, the GPS message is transmitted to the microprocessor that performs the tasks of calculating the position, velocity and time, without any intervention of the DSP tasks.

Patent Nr 00204544.1-2215:

Numerically Controlled Oscillator in Particular for a Radio-Frequency Signal Receiver

This patent concerns the multiphase NCO described at the section 3.6, where the phase accumulator has been split in two parts. The Most Significant Bit Accumulator (MSBA) is clocked at the sampling frequency rate, whereas the Least Significant Bit Accumulator (LSBA) at a lower frequency rate. The claimed NCO has less power consumption thought maintaining the same frequency characteristics than a standard NCO.

Patent Nr 00204605.0-1248:

Correlation and Demodulation Circuit for a Receiver for Signals Modulated by a Specific Code

This patent concerns the Loop-Back test presented in section 4.13, which enables the integrity test of the DSP architecture without a real

GPS signal. This test should be realized at system boot and verifies whether the interaction between the sub-circuits operates as expected.

Patent Nr 00204606.8-1248:

Radio-Frequency Signal Receiver with Control Means for the Channels to be Controlled

This patent concerns the Priority Decoder described in section 4.12, which allows the channel selection. In normal mode, the claimed decoder enables the direct access to the different channels. In the virtual mode, it allows the access to those channels, which have generated an interruption. This selection is defined upon a given priority order. The channel with the highest identification number has the highest priority.

1.7 References

[Par94] B. –W. Parkinson, “GPS Error Analysis”, *Global Positioning System: Theory and Applications*, Vol. 1, pp 481-483, American Institute of Aeronautics and Astronautics Inc., 1996.

Chapter 2

GPS Principles & Applications

In this chapter, a description of the GPS process is presented. All state of the art GPS receivers have become digital prior to correlation and Doppler removal, because DSP systems are able to accomplish tasks, which are very difficult, or even impossible, to achieve with analogue electronics. To understand the GPS system more easily, a high-level block diagram of a digital GPS receiver is presented in the first part of this chapter. The second part of this chapter is dedicated to the descriptions of the DSP functions, starting with the hardware and following with the software.

In principle, this chapter will accommodate the reader with the most important digital GPS receiver concepts that should facilitate the understanding of the original contributions of this thesis. If desired, more detail on this subject can be found in the reference literature as presented in section 2.18.

2.1 GPS History & Applications

For six thousand years, humans have been developing ingenious ways of navigating to remote destinations. A fundamental technique developed by both ancient Polynesians and modern navies is the use of angular measurements of the natural stars.

With the deployment of radio signals, another class of navigation aids was born. The first systems were based on radio beacons. Some examples are VHF Omni-directional Radios (VOR); long range radio navigation (LORAN) and OMEGA. With the use of artificial satellites, more precise line-of-sight radio navigation signals became available. This practice started in the 1960s, when the U.S. Navy's Navigation Satellite System (NNSS) opened a new area of navigation technology and capability [Par94].

2.1.1 Navy Navigation Satellite System

The NNSS (also called Transit) was based on the Doppler shift measurement of a tone broadcast at 400 MHz. The Transit satellites had polar orbits at altitudes of about 600 nautical miles.

The tone broadcast was continuous. The maximum rate of change in the Doppler shift of the received signal corresponded to the point of closest approach of Transit satellite. The difference between "up" Doppler and "down" Doppler was used to calculate the range to the satellite at closest approach. Based on these measurements, the position could be determined within a few hundred meters, with known altitude (e.g. sea level) and the broadcast ephemeris of the satellites. The mutual interference restricted the number of satellites available word wide to about five. The intermittent availability of the signals was thus a limitation.

2.1.2 Timation Satellites

By 1972, another U.S.-Navy satellite system was extending the state of the art by orbiting very precise clocks. Known as Timation, these satellites were used principally to provide very precise time. In addition, they could provide navigation information. The ranging signals used a technique called side tone ranging, which broadcasts a variety of synchronized tones to solve phase ambiguities.

Initially, these spacecrafts used very stable quartz-crystal oscillators; later models used the first atomic frequency standards (rubidium and cesium). The atomic clocks had a frequency stability of several parts in 10^{12} per day. This frequency stability greatly improved the prediction of satellite orbits (ephemeris) and also extended the time between required control operations. Timation satellites were flown in inclined orbits: the first two at altitudes of 500 nautical miles and the last in the series at 7500 nautical miles. The third Timation satellite was also used as a technology demonstrator for GPS.

2.1.3 U.S. Air Force Project 612B

The third essential foundation for GPS was an U.S. Air Force program known as 612B. By 1972, this program had already demonstrated the operation of a new type of satellite ranging signal based on Pseudo Random Noise (PRN). The signal modulation was essentially a repeated digital sequence of fairly random bits (ones or zeros) with certain useful proprieties:

- *A navigation user can detect the start phase of the repeated sequence and use this for range determining.*
- *All satellites can broadcast on the same nominal frequency because the selected PRN coding sequences are orthogonal.*
- *Most forms of jamming or deliberate interference can be rejected.*
- *The sequence can easily be generated with a simple shift register and some logic.*

In addition, a communication channel could be added by inverting the whole sequence at a slow rate. These inversions can be used to indicate the ones or zeros of digital data streams.

The operational GPS system of today is virtually identical to the one proposed in 1972 where the satellites have expanded their functionality to support additional military capabilities. The orbits are slightly modified, but the equipment designed to work with the original four satellites would still perform that function today.

The orbital configuration is a total of 24 satellites, four in each of six circular rings with inclination of 55 degrees. The rings are equally spaced around the equator, and the orbital altitudes are 10`980 nautical miles. This altitude gives two orbital periods per day and produces repeating ground traces.

2.1.4 GPS Applications

The fundamental navigation technique for GPS is to use one way ranging from the GPS satellites that also broadcast their estimated positions. Ranges are measured to four satellites simultaneously in view by matching (correlation) the incoming signal with a user generated replica signal and measuring the received phase against the user's (relative crude) crystal clock. With four satellites and appropriate geometry, four unknowns can be determined. Typically, these are latitude, longitude, altitude and a correction to the user clock. If altitude or time is already known, a fewer number of satellites can be used.

The future position of the satellite is estimated from ranging measurements taken at worldwide monitoring stations. The Operational Control System (OCS) uses five monitor stations, which are located at Colorado Springs, Ascension Island, Diego Garcia, Kwajalein and Hawaii. These ranging measurements use the same signals that are employed by a typical user's receiver. The master control station forms estimates of future satellite locations and future satellite clock corrections and uploads these estimates to the satellites in the form of ephemeris.

2.2 GPS Signal Characteristics

The GPS SV transmits two synchronous ranging signals; the primary signal called L_1 and the secondary signal called L_2. The carrier frequencies are modulated by spread spectrum codes with a unique PRN sequence associated with each SV and by the navigation data message. All SVs transmit at the same two carrier frequencies, but their signals do not interfere significantly with each other because of the PRN code modulation. Since each SV is assigned to a unique PRN code and all of the PRN code sequences are nearly orthogonal with respect to each other, the SV signals can be separated and detected by a technique called Code Division Multiple Access (CDMA). In order to track one SV in common view with several other SVs by the CDMA technique, a GPS must replicate the PRN sequence for the desired SV along with the replica carrier signal.

Two carrier frequencies are provided to permit the two-frequency users to measure the ionosphere delay. Single frequency (L_1 only) users must estimate the ionosphere delay using modeling parameters that are broadcast to the user in the navigation message.

2.2.1 Frequencies and Modulation Format

Figure 2.1 shows the block diagram of the SV signal structure. The L_1 frequency (f_1=154 • f_{ref}) is modulated by the navigation message and two PRN codes, the Coarse/Acquisition code (C/A-code) and the Precision code (P-code). The L_2 frequency (f_2=120 • f_{ref}) is modulated by only one PRN code at time. The nominal reference frequency f_{ref} is 10.23 MHz. The C/A-code and the P-code have respectively a chipping rate of 1.023 • 10^6 chips/sec and 10.23 • 10^6 chips/sec.

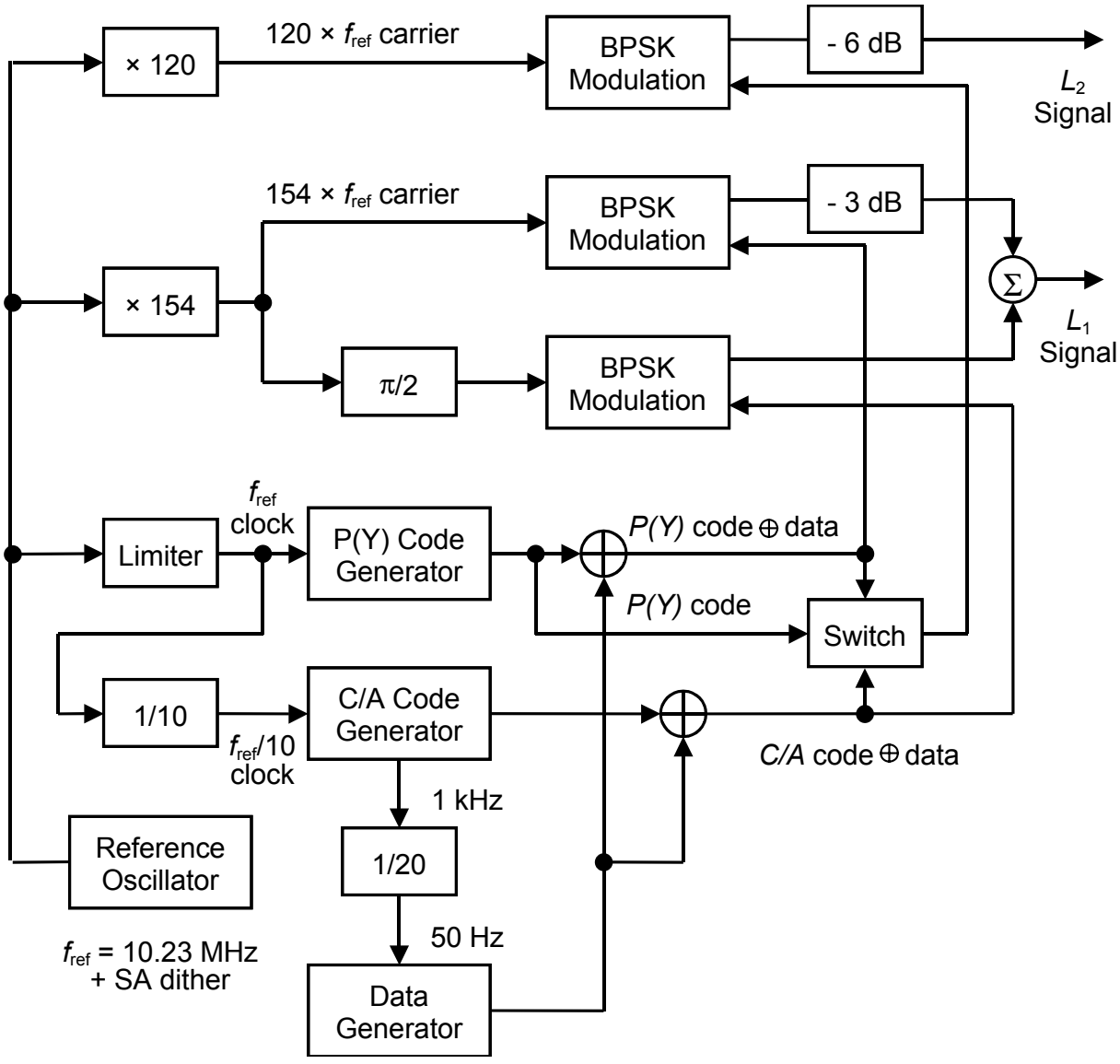


Figure 2.1: GPS satellite signal structure.

The precision code can be denied to the SPS users if the control segment activates Anti-Spoofing (AS) mode in the SV. When AS is

activated, the P-code is encrypted to form what is known as P(Y)-code. Both the C/A-code and the P(Y)-code as well as L_1 and L_2 carrier frequencies are subjected to the encrypted dither frequency of SA. This SA phase modulation effect creates a pseudo-random Doppler error on both the pseudorange and delta pseudorange measurements. Only the PPS user can remove this error. In addition, SA encrypts an offset error in the satellite's broadcast ephemeris and almanac data.

The combination of the 50-bits/sec data with both the C/A-code and the P(Y)-code uses the exclusive or process, denoted by $\oplus$. The C/A-code $\oplus$ data and P(Y)-code $\oplus$ data is a synchronous operation. Figure 2.2 illustrates the result of C/A-code $\oplus$ data and P(Y)-code $\oplus$ data. The exclusive-or process is equivalent to a binary multiplication of two one-bit values. One data bit contains either 204'600 P(Y)-code chips or 20'460 C/A-code chips.

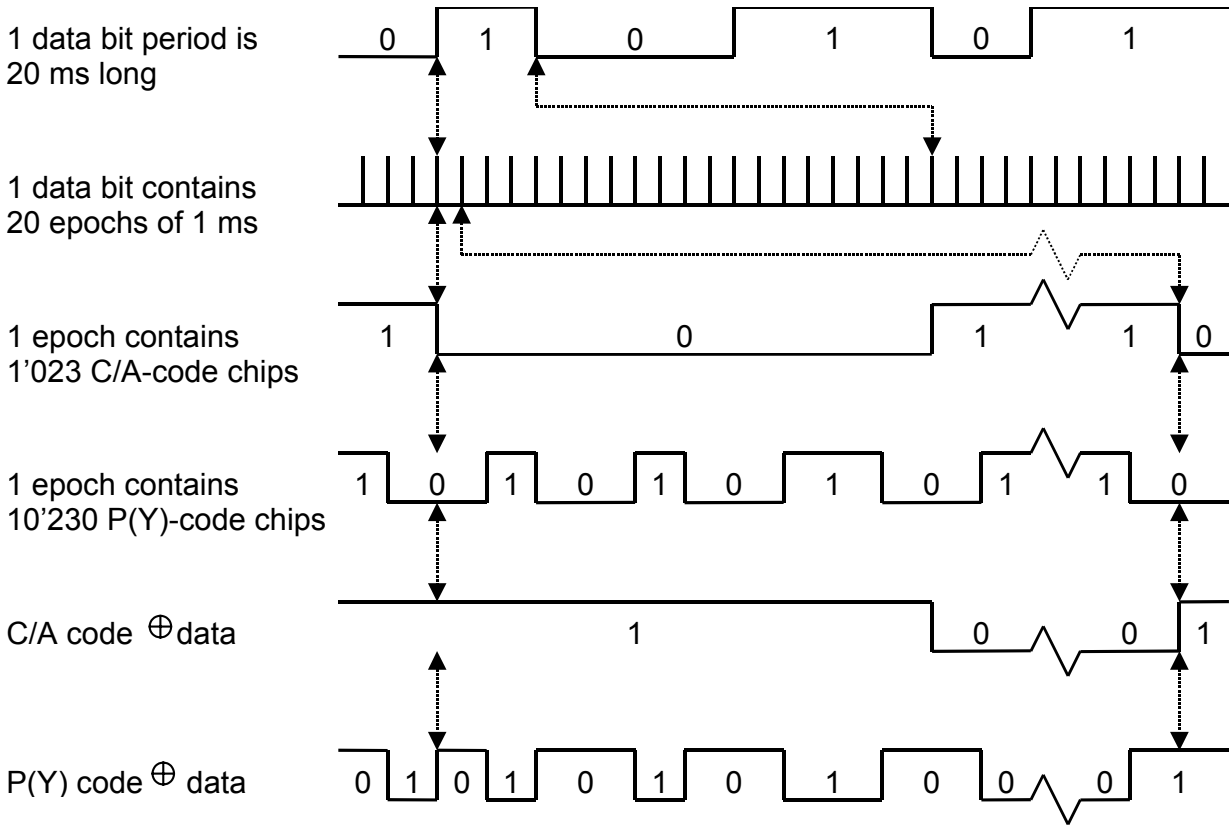


Figure 2.2: GPS code mixing with data.

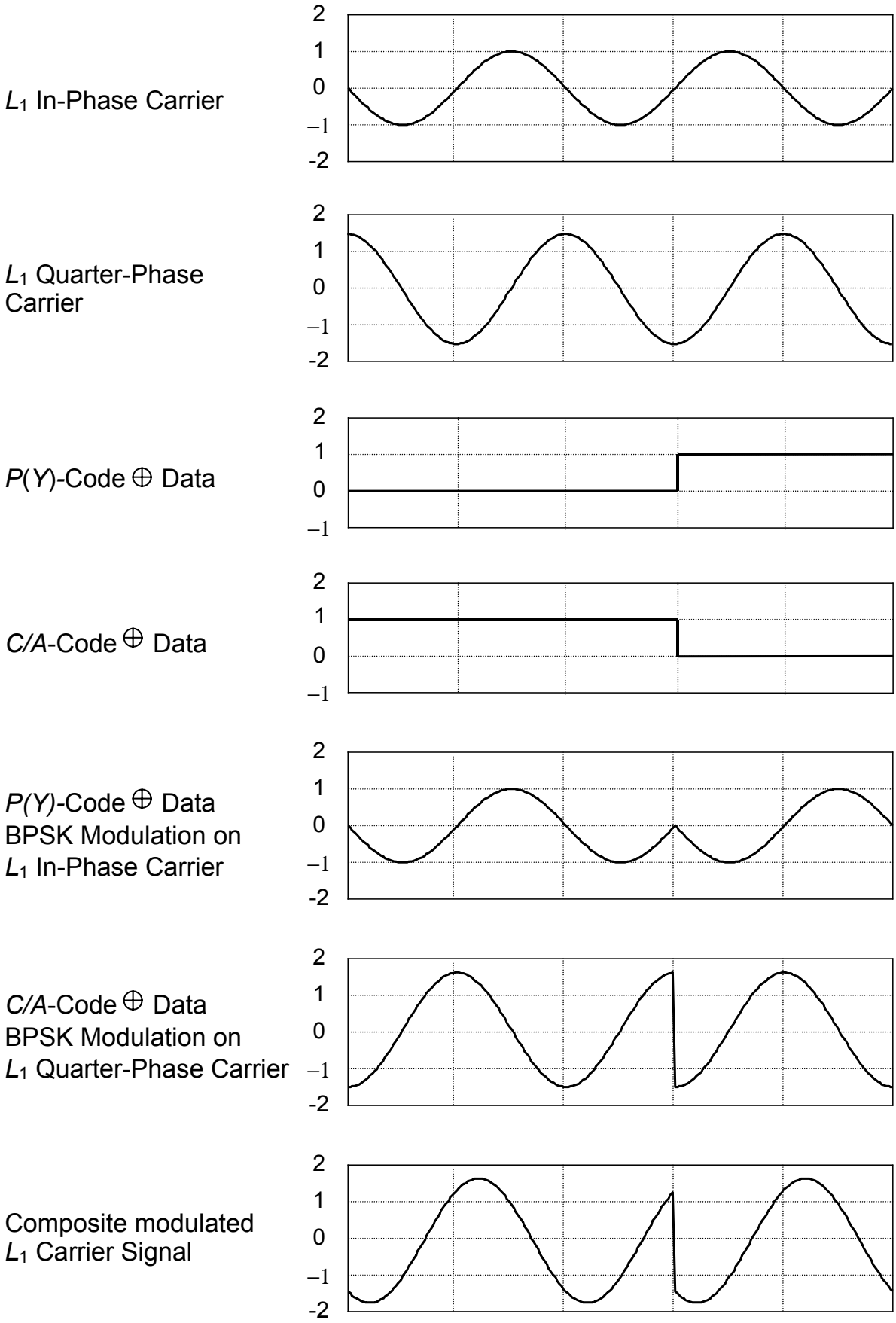


Figure 2.3: L_1 carrier modulation.

The type of modulation used in GPS is Bi-Phase Shift Key (BPSK). The L_1 frequency is modulated in-phase quarter-phase by the P(Y)-code $\oplus$ data and the C/A-code $\oplus$ data. Thus there is a $\pi/2$ phase shift between the C/A-code $\oplus$ data modulation and the P(Y)-code $\oplus$ data modulation on these two combined L_1 carrier frequencies. Figure 2.3 illustrates how the signal waveforms would appear before and after the BPSK modulation of one P(Y)-code $\oplus$ data transition and one C/A- code $\oplus$ data transition. There are 154 carrier cycles per P(Y)-code chip and 1'540 carrier cycles per C/A-code chip on L_1.

The L_2 frequency can be modulated by either the P(Y)-code $\oplus$ data or the C/A-code $\oplus$ data or with P(Y)-code alone as selected by the control segment. P(Y)-code and C/A-code are never present simultaneously on L_2, as is the case with L_1. There are 120 carrier cycles per P(Y) code chip on L_2. Table 2.1 summarizes the GPS signal structure on L_1 and L_2.

Signal Priority	**Primary**	**Secondary**
Signal Designation	L_1	L_2
Carrier Frequency [MHz]	f_1=1'575.42	f_2=1'227.60
PRN Codes [chips/sec]	$P(Y)=10.23 \cdot 10^6$ and $C/A=1.023 \cdot 10^6$	$P(Y)=10.23 \cdot 10^6$ or $C/A=1.023 \cdot 10^6$
Data [bits/sec]	50	50

Table 2.1: GPS signal structure.

2.2.2 Signal Representation

The GPS-signal is represented in the time domain with a sinusoidal function as follows [Die95a]:

$$s(t) = A_0 \cdot C(t) \cdot D(t) \cdot \cos\left[(\omega_0 + \Delta\omega) \cdot t + \Phi_0\right] \qquad (2:1)$$

In this relationship A_0 = signal amplitude; $C(t)$ = PRN code modulation (±1); $D(t)$ = 50 bps data modulation (±1); $\omega_0=2\pi \cdot f_0$ = carrier frequency (L_1 or L_2); $\Delta\omega=2\pi \cdot \Delta f$ frequency offset (Doppler etc.) and Φ_0 = nominal but ambiguous carrier phase.

In the frequency domain, the spectral density of the signal power is the spectral density of the PRN code power centered at $\pm(f_0 + \Delta f)$. In the baseband, this spectral density is as follows:

$$S_s(f) = \tfrac{1}{2} \cdot A_0^2 \cdot T_c \cdot \mathrm{sinc}^2(f \cdot T_c) \tag{2:2}$$

Where T_c is the PRN code chip width. The signal power in a two-sided bandwidth of $2B$ Hz is given by the following relationship:

$$P_s = \int_{-B}^{B} S_s(f) \cdot df \tag{2:3}$$

If $B=\infty$, $P_s=A_0^2/2$, the signal is thus completely preserved. If $B=1/T_c$, $P_s=0.9 \cdot A_0^2/2$, which corresponds to a 0.45 dB signal loss.

For a satellite at elevation angle above 5 degrees, the minimum specified received signal strength is given below in Table 2.2. The maximum spectral density for a PRN signal is P_s/f_c, where f_c represents the PRN code frequency. For the specified P_s=-160 dBW and f_c=1.023 MHz, the spectral density of the signal power is -220.1 dBW/Hz.

Link	P-Code	C/A-Code
L_1	- 163 dBW	- 160 dBW
L_2	- 166 dBW	- 166 dBW

Table 2.2: GPS minimum received signal power levels at output of a 0-dBIC antenna with right-hand circular polarization [Spi94b].

2.2.3 Noise Representation

Ambient noise is represented in the frequency domain as white noise with a constant spectral density of $N_0/2$=-205.2 dBW/Hz. Note that the spectral density of the noise power is 14.9 dB above the spectral density of the signal power. Figure 2.4 illustrates the spectral density of the noise power passed through a unit-gain band-pass filter, with a two-sided bandwidth of $2B$ Hz centered at $\pm f_0$.

The resulting noise power P_n in a $2B$ two sided bandwidth equals $2N_0 \cdot B$. The SNR in a $2B$ two-sided noise bandwidth is as follows:

$$SNR = \frac{P_s}{P_n} = \frac{A_0^2}{4N_0 \cdot B} \tag{2:4}$$

The output of the bandpass filter has the following time domain representation:

$$n(t) = x(t) \cdot \cos(\omega_0 \cdot t) - y(t) \cdot \sin(\omega_0 \cdot t) \tag{2:5}$$

In the relationship above, $x(t)$ and $y(t)$ represent respectively the in-phase and quarter-phase components. Both are bandlimited Gaussian processes with the properties defined as follows:

$$\begin{aligned} E[x(t)] &= E[y(t)] = 0 \\ E[x^2(t)] &= E[y^2(t)] = P_n \\ E[x(t) \cdot y(t)] &= 0 \end{aligned} \tag{2:6}$$

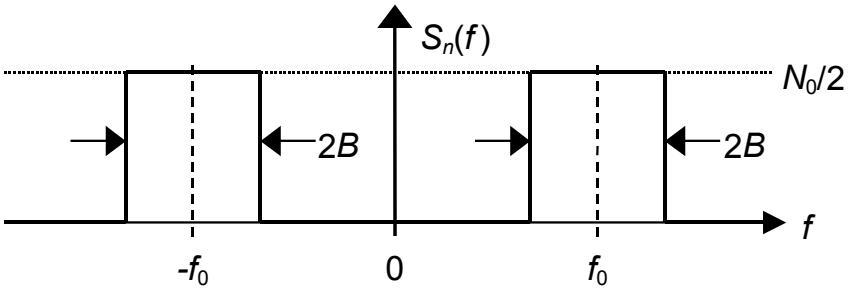


Figure 2.4: Spectral representation of ambient noise.

2.3 Spread Spectrum

Spread spectrum signaling is a method of taking a data signal $D(t)$ with bandwidth B_D and to spread its bandwidth to a new value B_S where $B_S \gg B_D$. Doing so, the spectral density of the signal power is reduced inversely. The bandwidth spreading is accomplished by multiplying $D(t)$ by a wide bandwidth spreading waveform $S(t)$ [Spi94a]:

$$S(t) = \sum_{n=-\infty}^{\infty} S_n \cdot p(t - n \cdot T_c) \tag{2:7}$$

In this relationship $p(t)$ represents a rectangular unit pulse over the interval $\{0, T_c=1/f_c\}$ and S_n is a random or pseudo-random sequence $S_n \pm 1$. Next, the spread spectrum signal is modulated by a sinusoidal carrier. For BPSK modulation, this form of spread spectrum modulation is called Direct Sequence-Spread Spectrum (DS-SS). Other forms of spread spectrum - called Frequency Hop Spread Spectrum (FH-SS) - include pseudo random frequency hoping. But only DS-SS provides a means to recover precise timing and recovery of the pure carrier.

In the receiver, the incoming signal is first demodulated. Next, an identical and precisely time-synchronized replica spreading signal $S(t)$ is generated and correlated with the received signal. The band-pass signal is thus converted to $D(t) \cdot S^2(t)=D(t)$, which follows from the auto-correlation properties of $S(t)$. So, the spread spectrum signal is compressed into its original bandwidth. The spectral density of the signal power is thus increased. The fact that the received signal must be accurately synchronized enables the system to extract accurate timing and ranging information.

2.3.1 Generation of the Spreading Signal

Linear feedback registers may be used to generate spreading signals. Figure 2.5 shows a simple four-stage linear feedback shift register. The taps after stages 1 and 4 are modulo-2 added to form a Pseudo-Noise (PN) sequence. The state vector components are defined by the state of the four binary shift register delay-elements. As long as this shift register is not set to the “all zero” state, the generator will cycle through its states. Only specific tap combinations produce maximal length PN sequences of 2^4-1=15 states.

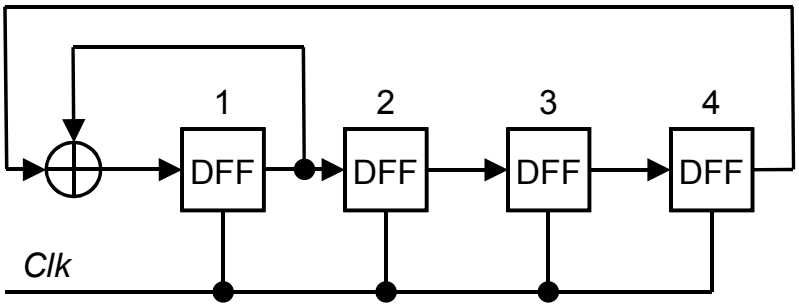


Figure 2.5: Generation of a PN sequence.

2.3.2 Auto-Correlation Function

The auto-correlation characteristics of the GPS PRN codes are fundamental to the demodulation process. To illustrate this, the auto-correlation function of a rectangular unite pulse is analyzed first (Figure 2.6). The equation for the rectangular unite pulse with a pulse width of one chip (T_c) is given by [War96a]:

$$\begin{aligned} p(t) &= 1 \qquad |t| \le T_c/2 \\ p(t) &= 0 \qquad \text{elsewhere} \end{aligned} \tag{2:8}$$

The auto-correlation function is defined as follows:

$$R_p(\tau) = \int_{-\infty}^{\infty} p(t) \cdot p(t+\tau) \cdot dt \tag{2:9}$$

In this relationship τ represents the phase shift of the replica code.

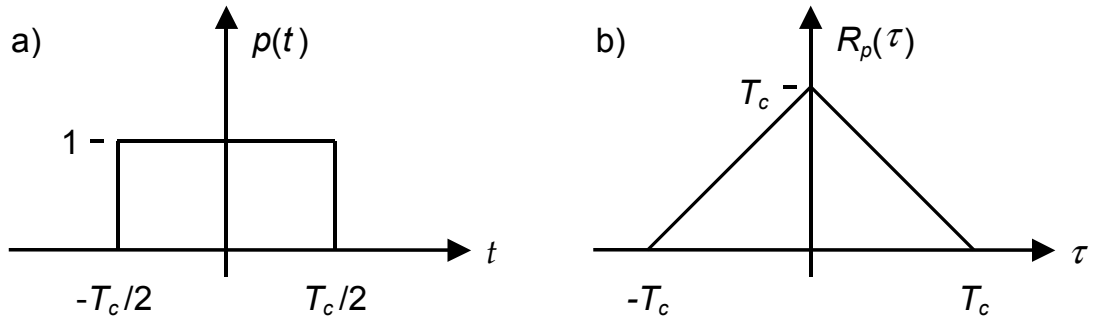


Figure 2.6: Rectangular pulse (a) and the auto-correlation function (b).

It is important to understand the mathematical process of auto-correlation, because a similar process takes place in a GPS receiver during the acquisition and tracking processes. The auto-correlation process involves replicating the function and then shifting its phase while multiplying it against the original function. When the phase of the replica is the same as the phase of the original, τ=0, maximum correlation is obtained. For the rectangular pulse, the auto-correlation is a triangular waveform. Its correlation amplitude drops linearly, as it is shifted forward or backward from τ=0 until the correlation amplitude becomes zero when τ is shifted by T_c or more.

A similar development can be made for the PRN code. The formula for the auto-correlation function for a maximum length PN sequence $PN(t)$ whose chipping period is T_c and whose period is $N \cdot T_c$ is the following:

$$R_{PN}(\tau) = \frac{1}{N \cdot T_c} \int_0^{N \cdot T_c} PN(t) \cdot PN(t+\tau) \cdot dt \tag{2:10}$$

Since PN is a maximum length sequence, the chip period is $N=2^n-1$, where n is the number of shift register stages used to generate the PN code. Figure 2.7 shows the auto-correlation function for a maximum

length PN sequence, which is the infinite series of triangular functions with period $N \cdot T_c$. The correlation amplitude outside the correlation interval is negative (-1/*N*), because in a maximum PN length sequence, the number of negative values (ones) is always one larger than the number of positive values (zeros).

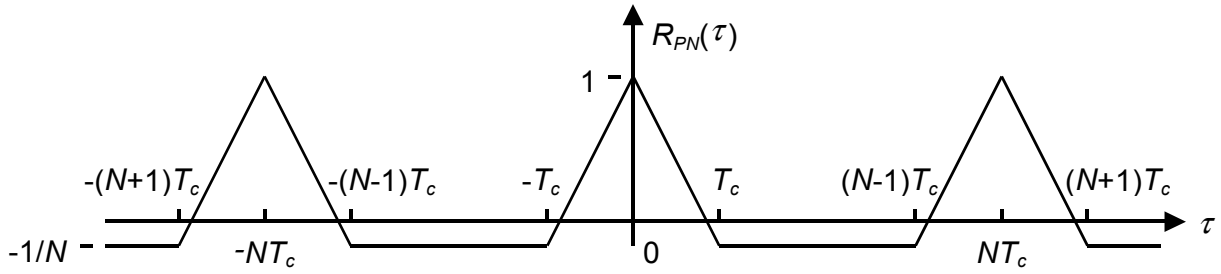


Figure 2.7: The auto-correlation function of a maximum length PN code.

2.4 Generic Receiver Description

As stated earlier, the fundamental navigation technique for GPS is to use one way ranging from the satellites that also broadcast their estimated positions. Ranges are measured to four satellites simultaneously in view, by matching (correlation) the incoming signal with a user replicated PRN code. It is important to understand that the GPS receiver must also detect the SV in the carrier-phase dimension by replicating the carrier frequency plus Doppler.

All modern GPS receivers use digital techniques to perform signal correlation and Doppler removal, because the DSP has many advantages over analogue signal processing. DSP systems are able to accomplish tasks, which would be very difficult, or even impossible with analogue electronics. Characteristics include flexibility and programmability, greater precision, and insensitivity to component tolerances. Analogue signal processing requires specific equipment, rewiring, and calibration for each new application, while digital techniques may be implemented, tested and easily modified on general-purpose computers. These advantages - coupled with the rapidly increasing density of digital Integrated Circuit (IC) manufacturing processes - have contributed to the evolution of the GPS receiver designs towards higher and higher levels of component integration. This trend is expected to continue. For this reason, a high-level block diagram

of a modern digital GPS receiver will be used to represent a generic GPS receiver architecture. Figure 2.8 illustrates the generic receiver, which consists of the following functions:

- *Antenna.*
- *Preamplifier.*
- *Reference Oscillator.*
- *Frequency Synthesizer.*
- *Down-Converter.*
- *Intermediate Frequency Unit (IFU).*
- *DSP.*
- *Application Processing.*

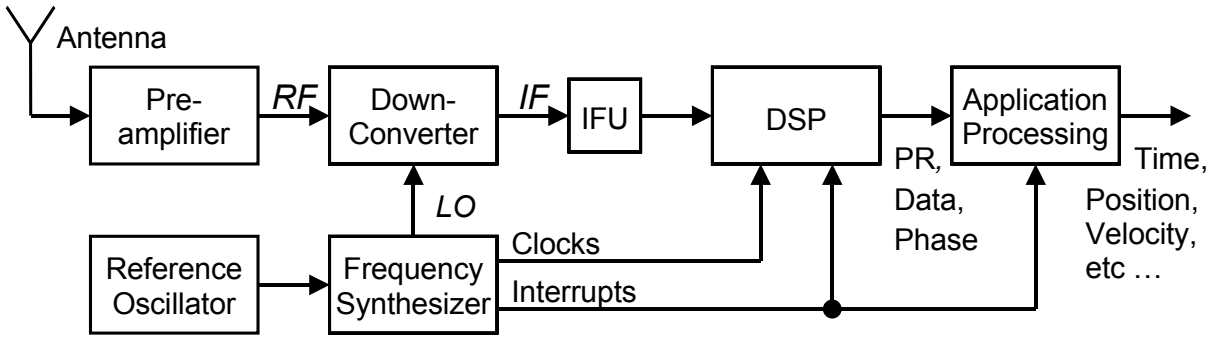


Figure 2.8: Generic GPS receiver functional block diagram.

The Antenna may be active or passive. Its function is to receive the GPS satellite signals while rejecting multipaths and interference signals. Multipath defines signals that arrive at the receiver via another path than the straight line-of-sight and is due to reflections from the Earth and nearby objects (e.g. buildings and vehicles).

The Preamplifier consists of burnout protection, filtering, and a Low-Noise Amplifier (LNA). Its primary function is to set the noise figure.

The Reference Oscillator output is used in the Frequency Synthesizer, where Local Oscillators (LO) and clocks are generated. One or more of these LOs are used by the Down-Converter to convert the Radio Frequency (RF) into Intermediate Frequency (IF).

The Down Converter mixes the LOs generated by the Frequency Synthesizer with the amplified RF input to IF frequencies and, if so designed, IF frequencies to lower IF frequencies.

The IFU provides further filtering of out of band noise and interference. Furthermore, it increases the amplitude of the signal plus noise level to a workable processing level. It may also contain Automatic Gain Control (AGC) circuits to control that level.

The core of the GPS receiver is the DSP, performing the following functions:

- *Sampling the signal plus noise.*
- *Splitting the signal plus noise samples into multiple signal-processing channels for simultaneously signal processing of multiple satellites.*
- *Generating the reference carrier and PRN code of the SV signals.*
- *Acquiring the satellite signals.*
- *Tracking the carrier and the PRN code of the SV signals.*
- *Demodulating the GPS system data from the SV signals.*
- *Extracting the code phase measurements from the PRN code of the SV signals.*
- *Extracting the carrier frequency and phase measurements from the carrier of the SV signals.*
- *Extracting SNR information from the satellite signals.*

The outputs of the DSP function are pseudoranges, pseudorange rates and/or delta pseudoranges, SNR, local receiver time tags and GPS system data for each of the GPS satellites being tracked.

The Application Processing controls the DSP functions and uses its outputs to satisfy application requirements. Although GPS is primary a satellite navigation system, the applications of GPS receiver are diverse and include:

- *Time and frequency transfer.*
- *Static and cinematic surveying.*
- *Differential GPS (DGPS) reference station receivers.*
- *GPS satellite signal integrity monitoring.*

2.4.1 Down-Converters

The amplified RF signal plus noise is down-converted to IF, using signal-mixing frequencies from LOs. The LOs are derived form the reference oscillator by the frequency synthesizer based on the frequency plan of the receiver design. One LO per down-converter stage is required. The LO signal mixing process generates both upper and lower side bands of

the SV signal, so the lower sideband are selected and the upper side band and leak-through signals are rejected by a post-mixer bandpass filter. The signal Doppler and the PRN codes are preserved after the mixing process. Only the carrier frequency is lowered. The IF is represented in the time domain with a sinusoidal function as follow:

$$I_s(t) = A_0 \cdot C(t) \cdot D(t) \cdot \cos\left[(\omega_{IF} + \Delta\omega) \cdot t + \Phi_0\right] \qquad (2:11)$$

In this relationship $\omega_{IF}=2\pi \cdot f_{IF}$ represents the carrier frequency of the IF. In the same way, the ambient noise becomes as follow.

$$I_n(t) = x(t) \cdot \cos(\omega_{IF} \cdot t) - y(t) \cdot \sin(\omega_{IF} \cdot t) \qquad (2:12)$$

The IF must be high enough to provide a single-sided bandwidth that will support the PRN code chipping frequency. An anti aliazing IF filter must suppress the stop band noise (unwanted out of band signals) at levels that are acceptable low when this noise is aliazed in DSP by the Analogue to Digital (A/D) conversion process.

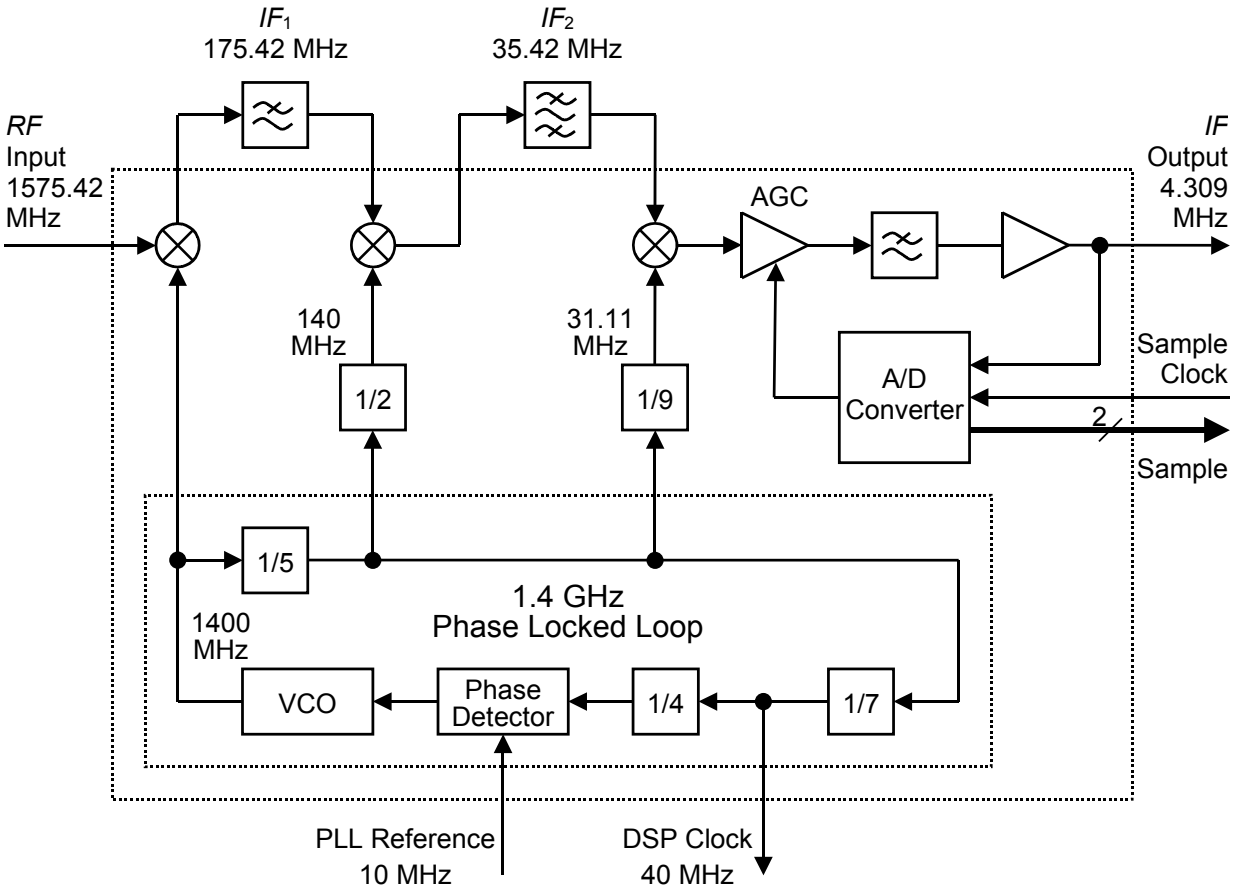


Figure 2.9: Block diagram of the GP2010 down-converter.

Figure 2.9 illustrates an example of down-converters manufactured by the Gec Plessey semiconductors. The GP2010 combines on chip Phase Locked Loop (PLL) synthesizer at 1400 MHz, 3 mixers and a 2-bit A/D converter. The down-converter receives the 1575.42 MHz signal transmitted by the GPS satellites and converts it to a 4.309 MHz IF, using a triple down-conversion [Gec95].

The input signal of the GP2010 is the GPS L_1 signal. The on chip PLL generates the first LO frequency at 1400 MHz. The output of the front-end mixer (Stage 1) at 175.42 MHz can then be filtered before being applied to the second stage. In the second stage the signal is mixed with a LO signal at 140 MHz giving a second IF at 35.42 MHz. The signal of the second stage is passed trough an external filter with 1 dB bandwidth of 1.9 MHz. The performance of this filter is critical to system performance and it is recommended to use a Surface Acoustic Wave (SAW) filter. The third mixer with an oscillator at 31.111 MHz gives a final IF at 4.309 MHz.

2.4.2 GPS Digital Signal Processing

The GPS receiver DSP is a two-dimensional signal replication process. In the code dimension, the GPS receiver accomplishes the cross correlation process by first searching for the phase of the desired SV code. Then, it tracks the SV code by adjusting the chipping rate of the replica code generator. In the carrier frequency dimension, the GPS receiver accomplishes the carrier matching process by first searching for the carrier Doppler frequency. Then, it tracks the SV carrier by adjusting the carrier frequency of the replica carrier generator. The Doppler induced effect on SV signal is therefore compensated. There is also an apparent Doppler effect on the carrier and code frequencies caused by the frequency offset of the reference oscillator.

The GPS receiver DSP processes dynamic signals that are to be analyzed in real time. Therefore, the processor must be able to keep up with the data flow. Most of small to medium size computers can keep up with data on the order of a few thousand samples per second. Above these rates, a general-purpose computer is not an effective solution. However, if the flexibility of a general-purpose computer is still needed, an array processor (a dedicated arithmetic processor that operates on

parallel data) or hardware accelerators are necessary to speed up the calculations.

In GPS applications, the receiver DSP algorithms are executed at different frequency rates as they operate on different stages of the signal processing. The algorithms that are executed at lower frequency rates, consist essentially of the following baseband functions:

- *Pre-detection bandwidth control.*
- *SV signal acquisition.*
- *SV signal tracking.*
- *Bit synchronization.*
- *Lock detection.*
- *Carrier-to-noise density determination.*
- *Data demodulation.*
- *Parity checking.*
- *Measurement processing.*

The frequency of these tasks varies from some Hz to some kHz. The implementation in software on general-purpose computers or microprocessors is therefore possible. The Software Signal Processing (SSP) requires important computational power, which can only be met with medium to big size computers or microprocessors. Unfortunately, the power consumption of this kind of microprocessors is very important. Generally, it represents the most important part of the total power consumption of a GPS receiver.

The receiver DSP algorithms, which are executed at higher frequency rates, consist essentially of the following processes:

- *Signal sampling & quantification.*
- *Demodulation.*
- *Number Controlled Oscillators (NCO).*
- *PRN code generation.*
- *Correlation.*
- *Post-correlation filtering.*

The sampling criteria govern the frequency of these tasks, which is around 5 MHz for C/A-code and 50 MHz for P(Y)-code. The implementation in software on general-purpose computers or microprocessors is therefore not possible. Generally, these tasks are executed on dedicated hardware called “correlator”.

Figure 2.10 represents the functional block diagram of one GPS receiver DSP channel, which is composed of a microprocessor and a correlator. First, the digitized IF signal plus noise is demodulated. Next, the base-band signal plus noise $I_{1,k}$ and $Q_{1,k}$ samples are correlated with the early, prompt and late replica codes, synthesized by the PRN Code Generator. After the carrier and code stripping process, a digital accumulation and dump operation occurs. This filtering provides a re-sampling at the processor baseband input rate. Usually, the hardwired accumulation process interval is about one millisecond, resulting in a 1 kHz dump rate. However, for more bandwidth narrowing, further signal accumulation may be performed by the microprocessor.

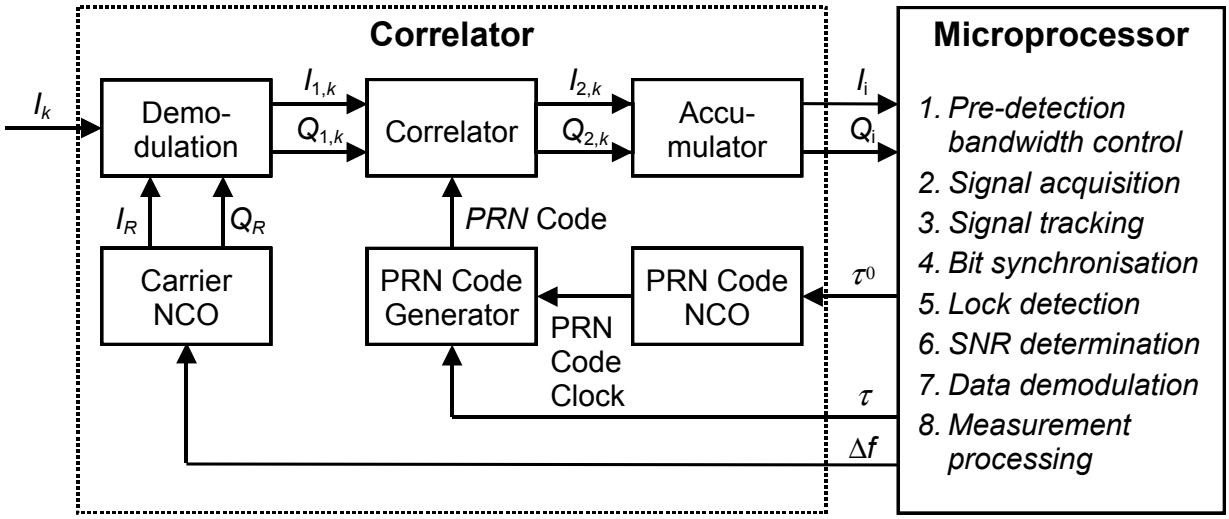


Figure 2.10: Generic digital receiver channel block diagram.

In the next sections, detailed descriptions of these functions are presented, starting with the hardware and following with the software part of the DSP.

2.5 Signal Sampling and Quantification

To process a continuous time signal by DSP techniques, it is necessary to convert the signal plus noise into a sequence of numbers. This procedure is called A/D conversion and the corresponding devices are called A/D Converters (ADC). Conceptually, the A/D conversion is a two-step process (Figure 2.11) [Pro89]:

- *Sampling: The conversion of a continuous-time signal into a discrete-time signal obtained by taking "samples" of the continuous-time signal at discrete-time instants. Therefore if the $x_a(t)$ is the input of the sampler, the output is $x_a(k \cdot T_s)=x(k)$, where T_s is called the sampling interval.*
- *Quantification and coding: The conversion of a discrete-time continuous value signal into a discrete-time, discrete-value (digital) signal. The value of each signal sample is represented by a value selected from a finite set of possible values. In the coding process, each discrete value is represented by a b-bit number, which may be denoted as $x_q(k)$. The difference between the non-quantified sample x(k) and the quantified value $x_q(k)$ is called the quantification error.*

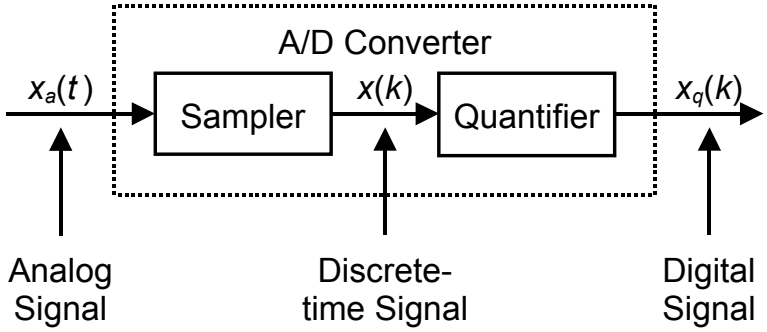


Figure 2.11: A/D conversion.

2.5.1 Time Domain Sampling

The sampling of an analogue signal $x_a(t)$ periodically every T_s seconds produces a discrete time signal $x(k)$ given by:

$$x(k) = x_a(k \cdot T_s) \qquad -\infty < k < \infty \tag{2:13}$$

The sampling frequency $f_s=1/T_s$ must be selected such that the sampling process will not result in any loss of spectral information (no aliazing). Indeed, if the spectrum of the analogue signal can be recovered from the spectrum of the discrete time signal, there is no loss of information. Consequently, the sampling process must be investigated by finding the relationship between the spectrum of $x_a(t)$ and $x(k)$.

In the time domain, the sampling process can be considered as a multiplication between the analogue signal and a Delta pulse train, which period correspond to the sampling period T_s [Pel90]:

$$x^*(t) = x_a(t) \cdot \sum_{k=-\infty}^{+\infty} \delta(t-k \cdot T_s) \tag{2:14}$$

To find the relationship between the spectrum of $x_a(t)$ and $x^*(t)$, it is necessary to calculate the Fourier transformation of the relationship above. For that purpose, the following propriety of the Fourier transformation can be used [Pel89a]:

$$2\pi \cdot [x_1(t) \cdot x_2(t)] \Leftrightarrow X_1(\omega) * X_2(\omega) \tag{2:15}$$

The relationship above shows that the multiplication between two signals in the time domain leads to a convolution of their spectrum in the frequency domain. The spectrum of the Delta pulse train remains a Delta pulse train, which frequency corresponds to the sampling frequency. Therefore, in the frequency domain, the sampling process corresponds to a convolution between the spectrum $X_a(\omega)$ of the analogue signal and the Delta pulse train of the sampling frequency:

$$X(\omega) \Leftrightarrow X_a(\omega) * f_s \cdot \sum_{n=-\infty}^{\infty} \delta(\omega - n \cdot \omega_s) \tag{2:16}$$

For example, consider the IF output of the Gec Plessey down-converter, as illustrated in Figure 2.9. If the sampling frequency is selected as 40/7 MHz, the power spectral density of the sampled IF signal (dashed lines) plus noise (solid lines) will appear as in Figure 2.12 and there is no aliazing. Hence the spectral density of the discrete time signal is identical to the spectral density of the analogue signal converted to 1.405 MHz.

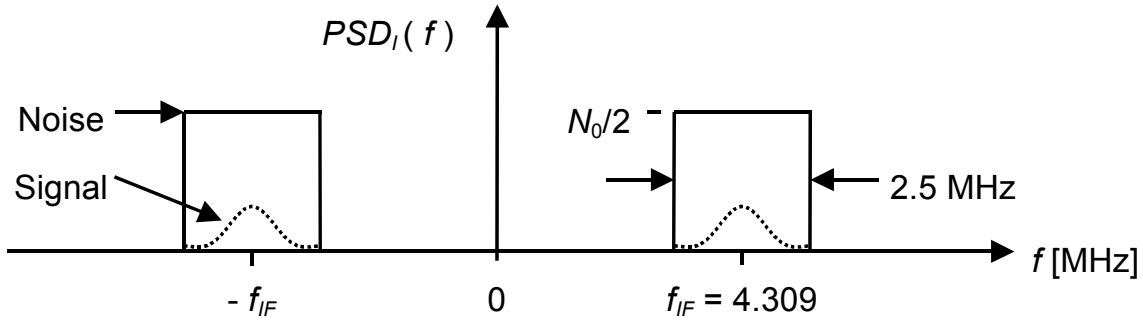
At the output of the A/D sampler, the discrete time signal is represented as follows:

$$I_{sk} = A_0 \cdot C_k \cdot D_k \cdot \cos(\Phi_k) \tag{2 17}$$

In this relationship C_k, D_k and $\Phi_k=(\omega_{IF} + \Delta\omega) \cdot t_k + \Phi_0$ represent respectively the code, the data and the phase of the samples attributable to the nominal phase and frequency offset at time t_k. In a similar way, the noise samples are represented as follows:

$$I_{nk} = x_k \cdot \cos(\Phi_k) - y_k \cdot \sin(\Phi_k) \tag{2:18}$$

a) Power spectral density of the analogue IF signal plus noise



b) Power spectral density of the sampled IF signal plus noise

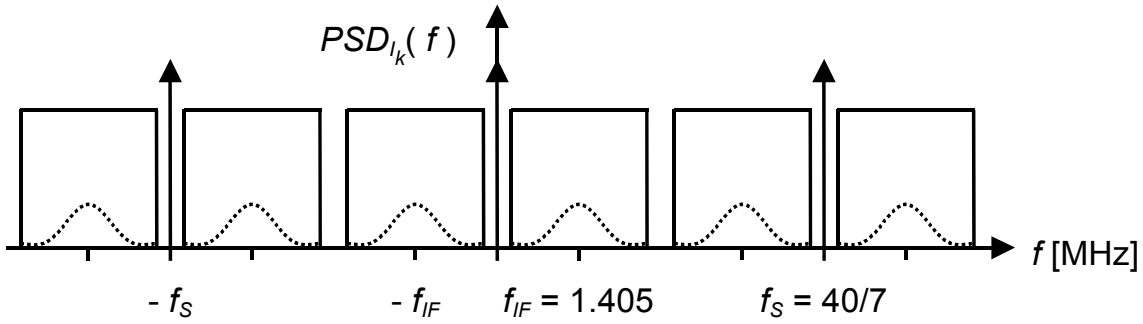


Figure 2.12: Gec Plessey time-domain sampling.

2.5.2 Pre-Sampling Filtering

In practice, the bandwidth of the original continuous-time signal may be larger than that containing the desired information. This commonly occurs when a low-frequency signal is corrupted by high frequency noise or in modulated signals, such as broadcast radio. If sampled at the rate dedicated by the sampling criterion for the desired analogue signal, unwanted high frequency signals may cause aliazing errors.

To prevent aliazing errors caused by these undesired signals, an analogue filter, called anti-aliazing filter, must be used. To reduce the DSP clock frequency, there is a desire to set the anti-aliazing filter bandwidth below the signal bandwidth. However, this filtering also causes correlation losses, because the side-lobes of the PRN code are eliminated.

For illustration, consider the GPS C/A-code sequence and its auto-correlation function, which is the infinite series of triangular functions of 1 millisecond period. Now, assume that the incoming code sequence is

filtered with a unit-gain rectangular filter with a two-sided bandwidth of 2*B*=2 MHz. This ideal filtering is known as the Gibbs effect and corresponds to a convolution between the incoming signal and a sinc function, where the position of the first zero is inversely proportional to the two-sided filter bandwidth (e.g. half microsecond) [Pel89b]:

$$\overline{PRN}(t) = 2B \cdot PRN(t) * \mathrm{sinc}(2B \cdot t) \tag{2:19}$$

In the GPS receiver, the filtered C/A-code is correlated with an identical but unfiltered C/A-code:

$$\overline{R}_{PRN}(\tau) = \overline{PRN}(-\tau) * PRN^{*}(\tau) \tag{2:20}$$

Considering the proprieties of the convolution function, the correlation function above becomes the infinite series of triangle functions convoluted with the sinc function:

$$\begin{aligned}
\overline{R}_{PRN}(\tau) &= 2B \cdot [PRN(-\tau) * \mathrm{sinc}(-2B \cdot \tau)] * PRN^{*}(\tau) \\
&= 2B \cdot [\mathrm{sinc}(-2B \cdot \tau) * PRN(-\tau)] * PRN^{*}(\tau) \;\Leftrightarrow\; \text{commutativity} \\
&= 2B \cdot \mathrm{sinc}(-2B \cdot \tau) * [PRN(-\tau) * PRN^{*}(\tau)] \;\Leftrightarrow\; \text{associativity} \\
&= 2B \cdot [PRN(-\tau) * PRN^{*}(\tau)] * \mathrm{sinc}(-2B \cdot \tau) \;\Leftrightarrow\; \text{commutativity} \\
&= 2B \cdot R_{PRN}(\tau) * \mathrm{sinc}(2B \cdot \tau) \;\Leftrightarrow\; \text{symmetry of the sinc function}
\end{aligned} \tag{2:21}$$

Figure 2.13 illustrates the difference between both correlation functions for one triangle, where the straight lines represent the correlation functions without filtering. The graphic shows that the correlation losses are not important outside the vertex of the correlation function.

However, in presence of multipath the filtering effects become more disadvantageous. Figure 2.14 illustrates the correlation losses in presence of a two-pass signal. In this graphic the multipath signal amplitude is attenuated by a factor 2 with respect to the true signal. The relative time delay between both incoming signals is half a chip. The graphic shows that correlation function with filtering is delayed almost by a quarter chip. This can represent the most important error contribution for pseudorange measurements.

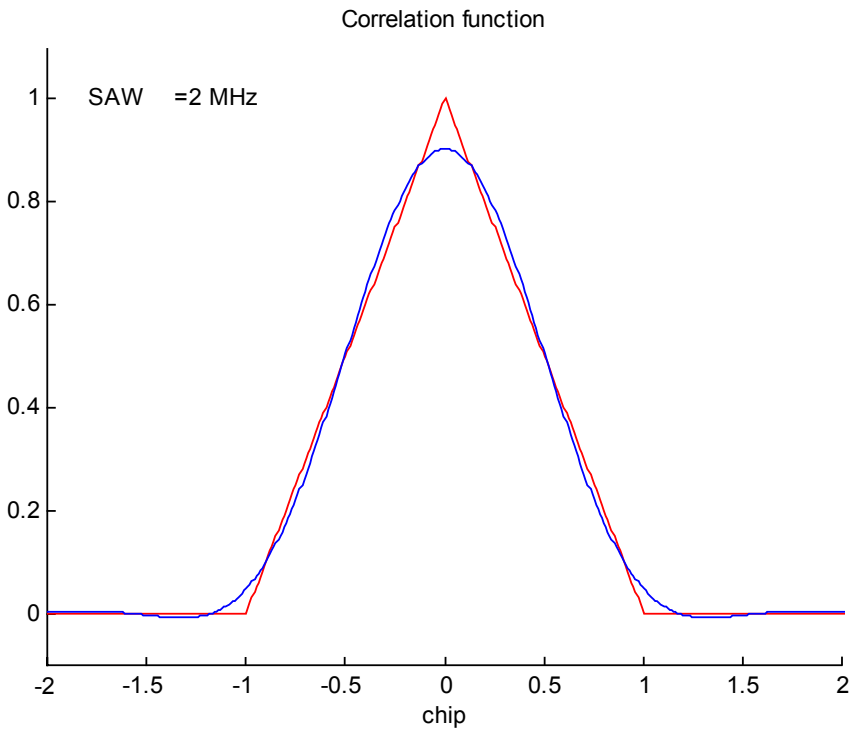


Figure 2.13: Correlation losses for a pre-correlation filtering of 2 MHz.

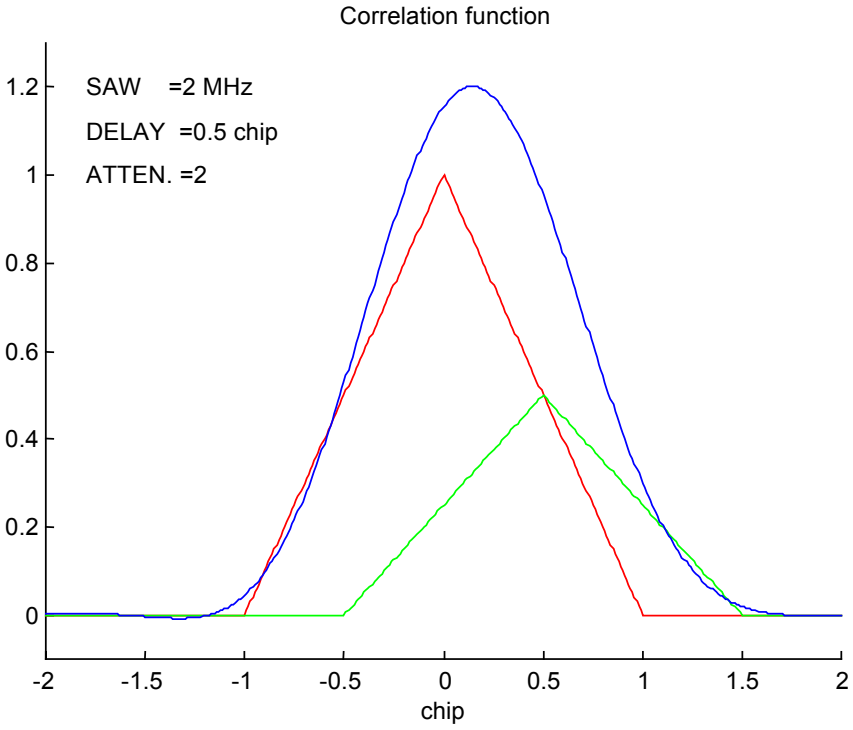


Figure 2.14: Correlation losses for a pre-correlation filtering of 2 MHz, in presence of multipath.

2.5.3 Quantification

The quantification is the process of converting a discrete-time signal into a digital signal by expressing each sample value as a finite (instead of infinite) number of digits. The errors introduced in representing the continuous-valued signal by a finite set of discrete value levels are called quantification errors or quantification noise. The excess digits can either be discarded (truncation) or discarded by rounding the resulting number to the nearest integer (Figure 2.15). The allowed values in the digital signal are called the quantification levels, whereas the distance $\Delta=2^{-b}$ between two successive quantification levels is called quantification step or resolution.

The rounding quantification process assigns each sample of $x(k)$ to the nearest quantification level. In contrast, the truncation quantification assigns each sample of $x(k)$ to the quantification level below it.

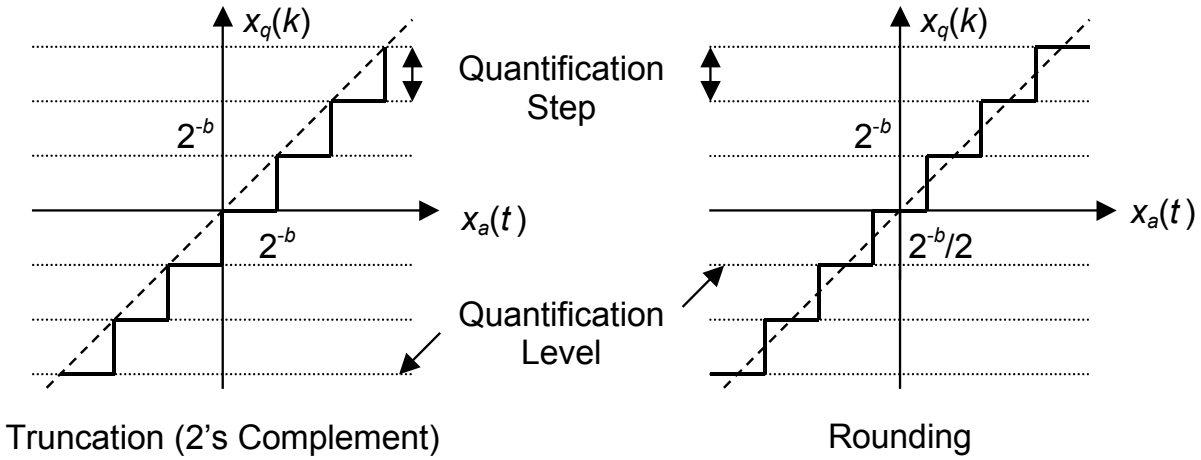


Figure 2.15: Illustration of quantification.

Figure 2.16 illustrates the quantification of the IF signal-plus-noise and the associated decoding. Three quantification cases are shown: 2-bit, 2.5-bit and 3-bit, using respectively three, five and seven threshold levels.

Theoretically, quantification of analogue signals always results in a loss of information. This is due to the ambiguity introduced by the quantification. Indeed, quantification is a non-reverse process since all samples in a $\Delta/2$ about a certain quantification level are assigned to the same value. This ambiguity makes the exact quantitative analysis of quantification extremely difficult.

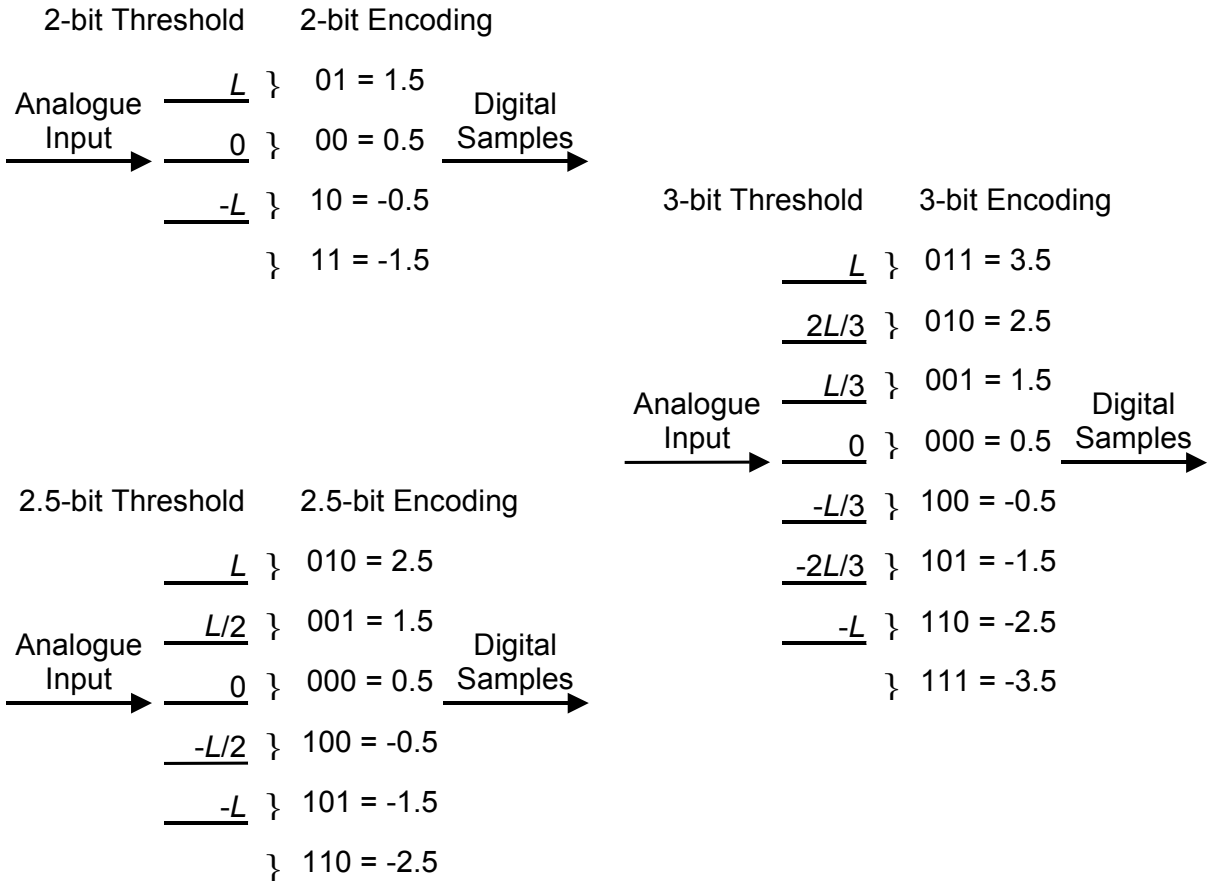


Figure 2.16: Quantification mechanism.

H. Chang has proposed statistical analysis of the degradation caused by quantification on the digital matched filter performance [Cha89]. These results can be extrapolated for the quantification losses for GPS receivers. The degradation caused by quantification for two different bandwidths is shown in Figure 2.17 for the 1-bit, 2-bit, 3-bit and 4-bit cases as a function of the ratio of the largest threshold *L* to Root Mean Square (RMS) noise level. Figure 2.17(a) presents the case where the single-sided bandwidth equals the code-chipping rate, while Figure 2.17(b) presents the case where the single sided bandwidth is five times the chipping rate. The reason why the opening of the pre-correlation bandwidth reduces the quantification loss and its threshold sensitivity is because there is more noise to dither the signal around the thresholds. Of course, the filtering loss is also less.

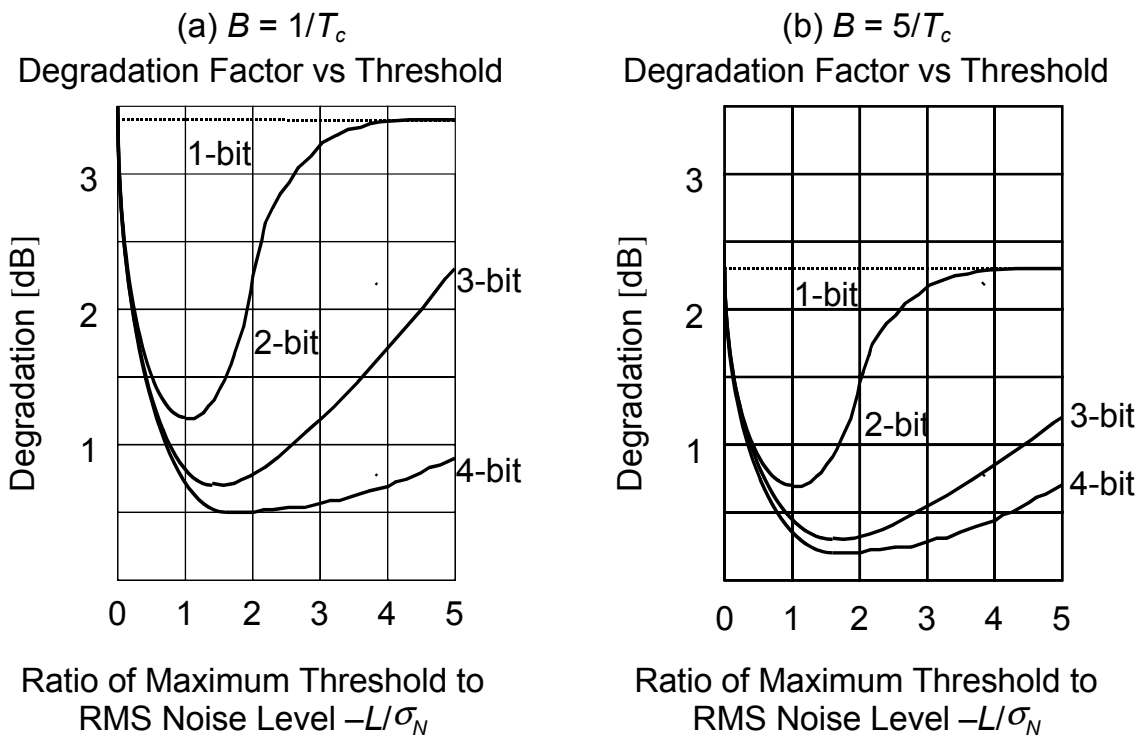


Figure 2.17: Quantification degradation in narrow & wide bandwidth.

2.5.4 Automatic Gain Control

Anytime multi-bit quantification is used, a pre-correlation AGC (or A/D threshold control, which has the same effect) is required. This AGC is used for three reasons:

- *Increased dynamic range.*
- *Quantification level control.*
- *Interference suppression.*

Some commercial receivers use 1-bit quantification (or hard limiting). This implementation does not require an AGC, provided that the front end has the required dynamic range to accommodate gain variation and interference.

2.6 Demodulation

The demodulation is accomplished with a digital implementation of the frequency mixing, which is nothing more than the execution of the following trigonometric identities:

$$\begin{aligned} I_{1,sk} &= 2I_{sk} \cdot \cos\Phi_{rk} \\ &= A_0 \cdot C_k \cdot D_k \cdot \left[\cos(\Phi_k - \Phi_{rk}) + \cos(\Phi_k + \Phi_{rk})\right] \\ Q_{1,sk} &= -2I_{sk} \cdot \sin\Phi_{rk} \\ &= A_0 \cdot C_k \cdot D_k \cdot \left[\sin(\Phi_k - \Phi_{rk}) - \sin(\Phi_k + \Phi_{rk})\right] \end{aligned} \quad (2:22)$$

In this relationship Φ_{rk} represents the phase of the reference IF carrier generated by the NCO at t_k. The mixing process generates both upper and lower side bands of the SV signal. It is important to fix the IF frequency in order to avoid interference between these components. In a well-designed frequency plan, the pre-detection accumulation process filters the upper sideband. In a similar way, the noise components are also mixed, but still random:

$$\begin{aligned} I_{1,nk} &= 2I_{nk} \cdot \cos\Phi_{rk} \\ &= x_k \cdot \left[\cos(\Phi_k - \Phi_{rk}) + \cos(\Phi_k + \Phi_{rk})\right] \\ &\quad - y_k \cdot \left[\sin(\Phi_k - \Phi_{rk}) + \sin(\Phi_k + \Phi_{rk})\right] \\ Q_{1,nk} &= -2I_{nk} \cdot \sin\Phi_{rk} \\ &= x_k \cdot \left[\sin(\Phi_k - \Phi_{rk}) - \sin(\Phi_k + \Phi_{rk})\right] \\ &\quad + y_k \cdot \left[\cos(\Phi_k - \Phi_{rk}) - \cos(\Phi_k + \Phi_{rk})\right] \end{aligned}$$

(2:23)

For example, consider the IF output of the Gec Plessey down-converter sampled at 40/7 MHz. Neglecting the quantification losses, the power spectral density of the digitized IF signal plus noise appears as the power spectral density of the analogue IF signal plus noise converted to 1.405 MHz (Figure 2.12). In the Gec Plessey correlators (GP2021), the IF signal plus noise samples are mixed with a reference carrier, which frequency is 1.405 MHz. The digital mixing process generates two components, the first at baseband and the second and at 2.81 MHz. Figure 2.18 illustrates the power spectral density of the signal (dashed lines) plus noise (solid lines), at the output of the mixer. Note that the upper side band is rejected at half of the sampling frequency. During the pre-detection accumulation process, the upper side band is filtered using digital integrators.

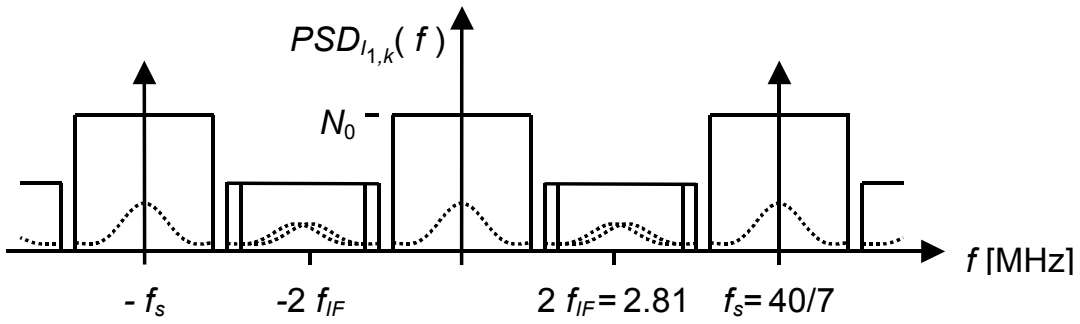


Figure 2.18: Power spectral density of the digitized signal plus noise samples, at the output of the Gec Plessey demodulation process.

2.6.1 Number Controlled Oscillator

The NCO (Figure 2.19) is the most efficient and flexible known method of generating a sinusoidal signal in the digital domain. First, a frequency increment Δf_{incr} is determined dependent on the frequency output resolution of the NCO. Next, the frequency number is accumulated at the clocking rate. Every time when a roll over occurs in the accumulator, a new cycle is generated.

For carrier frequency whip-off, the instantaneous digital phase is converted into digital amplitude using for example a look up table. For PRN clock generation, this conversion simply consists of taking the most significant bit of the NCO as the clock signal.

The illustrated N_b-bit NCO has the following proprieties:

- *The frequency output resolution is $\Delta f_{min}=2^{-N_b} \cdot f_{CLK}$, where f_{CLK} is the clock frequency, which is usually chosen equal to the sample rate entering the demodulation function.*
- *The frequency range of the NCO is governed by the Shannon criteria, which is given by $\Delta f_{max}=f_{CLK}/2$.*
- *The phase resolution $\Delta\phi_{min}=\Delta f_{incr}/f_{CLK}$ cycles is the step size of the phase accumulator. Normally, the frequency generated in the carrier NCO equals the Doppler frequency plus an intentional frequency offset. This offset is small compared to the clocking frequency of the NCO and therefore the resulting phase errors are negligible. Furthermore, these errors are random, if the clocking frequency is asynchronous with the intentional frequency offset.*
- *The number of output bits M_b does not need to be equal to the entire N_b bits.*

- *The number of bits representing the sinus and cosines of phase K_b does not have to be higher than three bits to provide good performance. With a single bit (square wave) the loss represents only 0.91 dB in the Doppler removal process.*
- *L_b bits can be latched and used by the microprocessor for real-time phase measurements.*

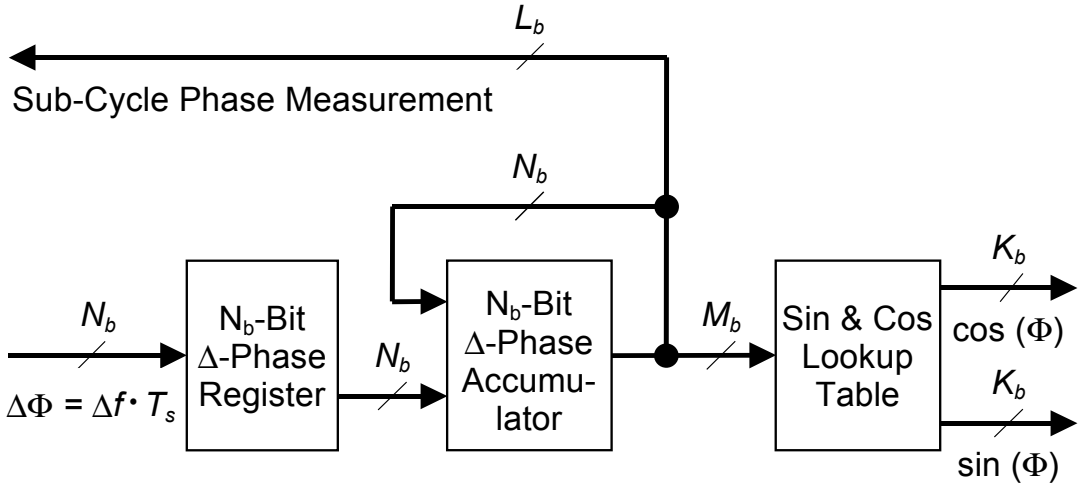


Figure 2.19: Functional block diagram of a NCO.

2.7 Coder Implementation

Both the acquisition and the tracking algorithms require the replication of the SV PRN code. The C/A-code is a Gold code, $G_i(t)$, of 1 millisecond in length and at a chipping rate of 1.023 megabits per second. The $G_i(t)$ sequence is a linear pattern generated by the modulo-2 addition of sub-sequences, $G_1(t)$ and $G_{2,i}(t)$, each of which is a 1023 chip long linear pattern. The epochs of the code are synchronized with the X1 epochs of the P(Y)-code. As shown in section 2.17, the $G_{2,i}(t)$ sequence is a $G_2(t)$ sequence selectively delayed from 5 to 950 chips. Doing so, a set of 36 mutually exclusive C/A-codes may be generated. The assignment of these codes to every SV is also given in section 2.17.

Two 10-stage shift registers generate the G_1 and G_2 sequences. The input of the corresponding shift registers is determined with the following polynomial equations:

$$G_1 = Z^{-10} \oplus Z^{-3}$$
$$G_2 = Z^{-10} \oplus Z^{-9} \oplus Z^{-8} \oplus Z^{-6} \oplus Z^{-3} \oplus Z^{-2} \qquad (2:24)$$

Restricting the implementation of the code generator to GPS C/A codes, many alternatives can be found to implement them. Figure 2.20 illustrates a C/A-coder implementation, which uses a selection of two taps of the G_2 register to generate the delay of the G_2 code. It can generate 45 possible codes, 9 of which are unbalanced. The remaining 36 are reserved for GPS [Nav91].

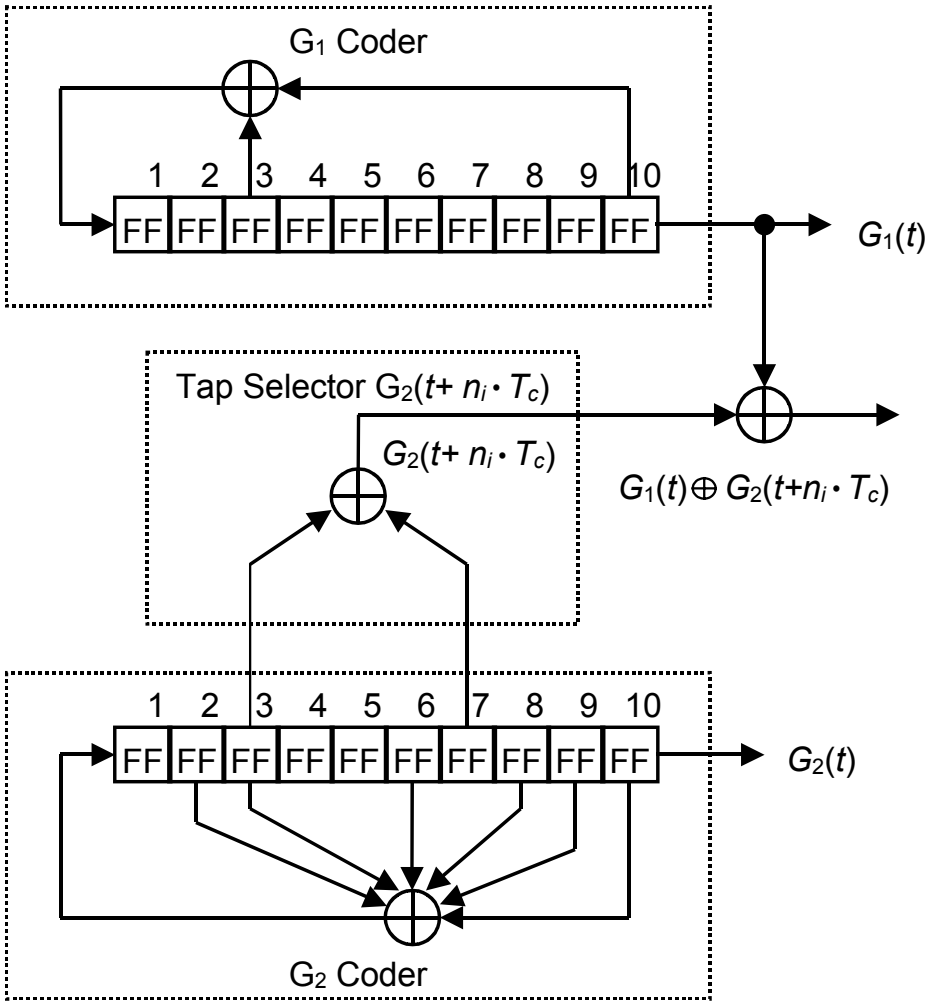


Figure 2.20: Two-tap selection C/A-code generator implementation.

2.7.1 Correlation

In the correlation process (Figure 2.21) the C/A code from the PRN code generator is shifted to obtain the Early, Punctual and Late codes (EPL). The early/late correlator spacing is defined by the clocking rate of the shift registers. At time t_k, the reference codes are output of the shift registers and multiplied by the $I_{1,sk}$ and $Q_{1,sk}$ samples of equations (2:22) [Die95b]:

$$\begin{aligned} I_{2,sk,j} &= I_{1,sk} \cdot C_{rk,j} \\ &= A_0 \cdot C_k \cdot C_{rk,j} \cdot D_k \cdot \left[\cos(\Phi_k - \Phi_{rk}) + \cos(\Phi_k + \Phi_{rk})\right] \\ Q_{2,sk,j} &= Q_{1,sk} \cdot C_{rk,j} \\ &= A_0 \cdot C_k \cdot C_{rk,j} \cdot D_k \cdot \left[\sin(\Phi_k - \Phi_{rk}) - \sin(\Phi_k + \Phi_{rk})\right] \end{aligned} \quad (2:25)$$

In this relationship $C_{rk,j}$ represents the shifted PRN code and j (j = E, P and L) denotes EPL. During the correlation process, the noise $x_{1,k}$ and $y_{1,k}$ are spread and become respectively $x_{2,kj}$ and $y_{2,kj}$.

The cross-correlation function between the incoming signal code and the referenced code j for a delay τ_{kj} at time t_k is given by:

$$\begin{aligned} E(C_k \cdot C_{rk,j}) &= R(\tau_{kj}) \\ &\approx 1 - \left|\tau_{kj}\right|, \left|\tau_{kj}\right| \le 1 \\ &\approx 0, \left|\tau_{kj}\right| > 1 \end{aligned} \quad (2:26)$$

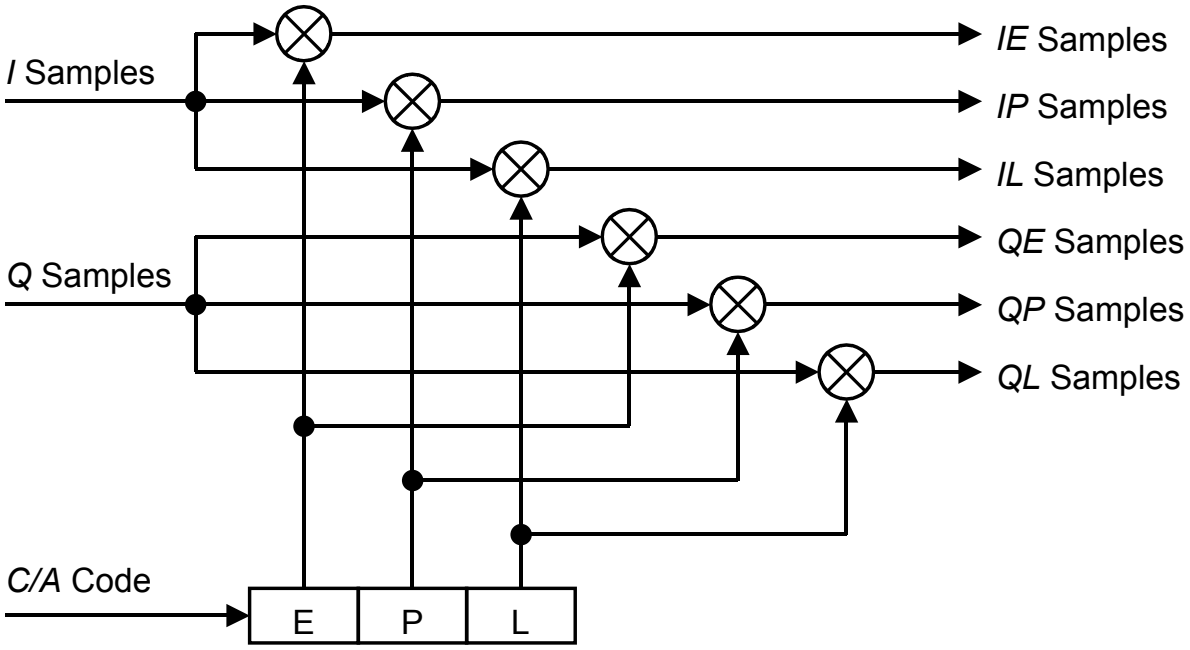


Figure 2.21: Correlation process.

Note that if the code-replica is identical and precisely time synchronized, the spread spectrum signal is converted into its original bandwidth, which corresponds to the 50 bps data modulation. In this case, the spectral density of the signal power will exceed the spectral density of the noise power.

2.8 Pre-Detection Bandwidth

Pre-detection is the signal processing after the carrier and code stripping process. During this process, the upper side band of the signal plus noise is eliminated using digital integrators. Furthermore, the baseband signal plus noise bandwidth is narrowed prior to being passed through the loop discriminators. At the output of the accumulators, the signal is resampled at the processor baseband input rate. The clock frequency for the integration process is around 5 MHz for C/A-code and 50 MHz for P(Y)-code. Generally, the hardwired accumulation process interval T is about 1 millisecond. So, the outputs of the accumulators are dumped at 1 kHz. However, for more bandwidth narrowing, further signal accumulation may be performed by the microprocessor.

The number of samples accumulated during the pre-detection interval T is $M_E = T/T_s$. Unless full correlation is achieved, the signal components are still random. Therefore, the accumulation process serves as a time average of the components of equation (2:25). We can only evaluate the result as an expected value [Die95b]:

$$\begin{aligned} E[I_{si}] &= E\left[\sum_{k=1}^{M_E} I_{2,sk,i}\right] \\ &= A_0 \cdot R(\tau_i) \cdot D_i \cdot \sum_{k=1}^{M_E} \left[\cos(\Phi_k - \Phi_{rk}) + \cos(\Phi_k + \Phi_{rk})\right] \\ E[Q_{si}] &= E\left[\sum_{k=1}^{M_E} Q_{2,sk,i}\right] \\ &= A_0 \cdot R(\tau_i) \cdot D_i \cdot \sum_{k=1}^{M_E} \left[\sin(\Phi_k - \Phi_{rk}) - \sin(\Phi_k + \Phi_{rk})\right] \end{aligned} \qquad (2:27)$$

In this relationship all M_E samples are within a data bit D_i period. Note that the subscripts have changed to i, indicating a sample with an end point at time t_i. The frequency and phase errors, Δf_i and $\Delta\Phi_i$, are assumed to be constant over the pre-detection interval T. The sums of the sine and cosine can therefore be approximated as an integration, which allows rewriting equation (2:27) as follows:

$$\begin{aligned} I_{si} &= A_0 \cdot M_E \cdot \operatorname{sinc}(\Delta f_i \cdot T) \cdot R(\tau_i) \cdot D_i \cdot \cos(\Delta\Phi_i) \\ Q_{si} &= A_0 \cdot M_E \cdot \operatorname{sinc}(\Delta f_i \cdot T) \cdot R(\tau_i) \cdot D_i \cdot \sin(\Delta\Phi_i) \end{aligned} \qquad (2:28)$$

In the pre-detection process, the noise samples are accumulated in the same manner. The accumulated noise samples have the following characteristics.

$$E(I_{ni}) = E(Q_{ni}) = 0$$
$$E(I_{ni}^2) = E(Q_{ni}^2) = \frac{N_0}{T_S} \cdot M_E = \sigma_n^2 \quad (2:29)$$

In this relationship σ_n^2 represents the variance of the noise samples at time t_i. Now, if the carrier power is expressed by $C=P_s=A_0^2/2$, the accumulated signal plus noise can be expressed in terms of SNR and RMS noise power σ_n.

$$\begin{aligned} I_i &= I_{si} + I_{ni} \\ &= \text{sinc}(\Delta f_i \cdot T) \cdot \sqrt{2 \cdot \frac{C}{N_0} \cdot T} \cdot R(\tau_i) \cdot D_i \cdot \cos(\Delta\Phi_i) \cdot \sigma_n + \sigma_n \\ Q_i &= Q_{si} + Q_{ni} \\ &= \text{sinc}(\Delta f_i \cdot T) \cdot \sqrt{2 \cdot \frac{C}{N_0} \cdot T} \cdot R(\tau_i) \cdot D_i \cdot \sin(\Delta\Phi_i) \cdot \sigma_n + \sigma_n \end{aligned} \quad (2:30)$$

Where C/N_0 represents the Carrier to Noise Ratio (CNR) and $(C/N_0) \cdot T$ the SNR at $1/T$ Hz two-sided noise bandwidth. For example, if the RF, the sampling and the quantification losses are ignored and if C=-160 dBW, the SNR=15.2 dB at 1 kHz two sided noise bandwidth. The relationship above is one of the most important concerning the GPS receiver DSP. The in-phase I_i and quarter-phase Q_i components will appear in most of the next relationships.

2.9 Signal Tracking

As stated earlier, the GPS receiver DSP is a two-dimensional signal replication process. In the code dimension, the cross correlation process is achieved by searching and tracking the phase of the SV code. In the carrier frequency dimension, the carrier matching process is accomplished by searching and tracking the carrier Doppler frequency.

The two-dimensional search and tracking process is easier to understand if the tracking process is explained first. In turn, the two-dimensional carrier and code tracking process is easier to understand if the carrier tracking process is explained first.

2.10 Carrier Tracking Loop

The programmable carrier pre-detection integrators, the carrier loops discriminators and the carrier loop filters characterize the carrier-tracking loop (Figure 2.22). These three functions determine the two most important performance characteristics of the carrier-tracking loop: the carrier loop noise errors and the maximum dynamic stress threshold.

The carrier loop discriminator defines the type of the tracking loop such as a Costas PLL or a Frequency Locked Loop (FLL). The Costas PLL is the most accurate but is more sensitive to dynamics stress than the FLL. The Costas PLL discriminator produces phase errors at its output, whereas the FLL discriminator generates frequency errors. There is a difference in the architecture of the loop filter as shown in Figure 2.23. For the same filter order there is an additional integrator in the FLL filter versus the PLL filter.

To tolerate dynamic stress, the pre-detection integration time should be short, the discriminator should be a FLL and the carrier loop filter noise bandwidth should be wide. However, for accurate carrier phase measurements, the pre-detection integration time should be long, the discriminator should be a PLL and the carrier loop noise bandwidth should be narrow.

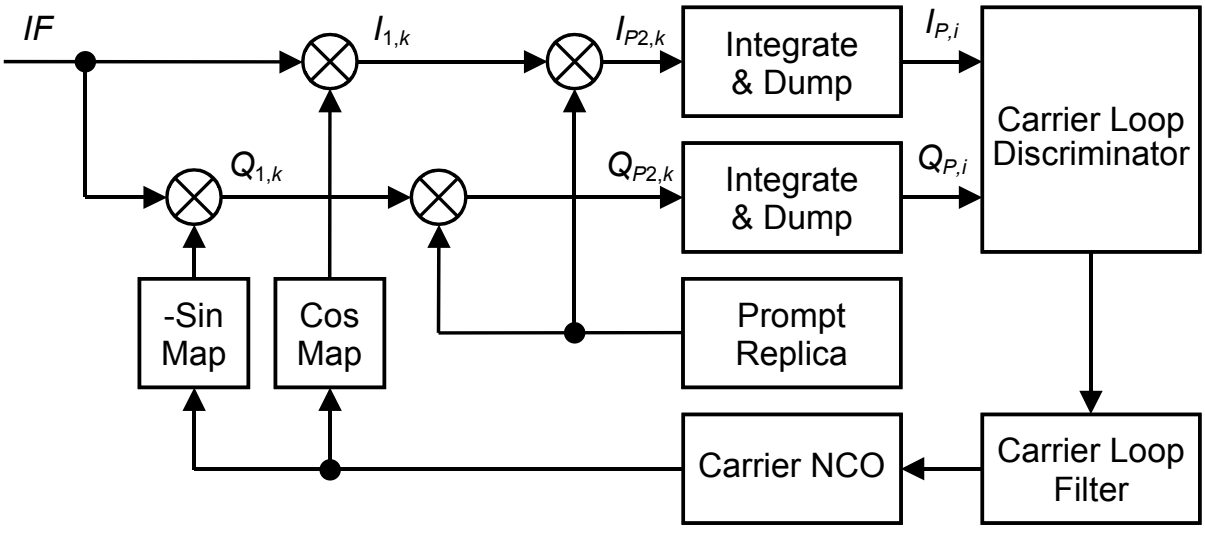


Figure 2.22: Generic block diagram of the carrier tracking loop.

2.10.1 Costas Phase Locked Loop

The Costas PLL discriminator tolerates the presence of data modulation on the baseband signal. Table 2.3 summarizes several Costas PLL discriminators, their output errors and their characteristics for GPS receivers [War96b]. Note that in the relationships below $I_{P,i}$ and $Q_{P,i}$ represent the accumulated signal plus noise in the punctual channels and $\Delta\Phi$ represents the phase error between the incoming and reference IF carriers.

Discriminator Algorithm	Output Phase Error	Characteristics
$\text{sign}(I_{P,i}) \cdot Q_{P,i}$	$\sin(\Delta\Phi)$	Near optimal at high SNR. Slope proportional to signal amplitude. Least computational burden.
$I_{P,i} \cdot Q_{P,i}$	$\sin(2\Delta\Phi)$	Near optimal at low SNR. Slope proportional to signal energy. Moderate computational burden.
$Q_{P,i} / I_{P,i}$	$\tan(\Delta\Phi)$	Sub-optimal at high and low SNR. Slope independent on signal amplitude. Higher computational burden.
$\text{atan}(Q_{P,i} / I_{P,i})$	$\Delta\Phi$	Two-quadrant arctangent. Optimal at high and low SNR. Slope independent on signal amplitude. Highest computational burden.

Table 2.3: Costas PLL discriminators.

Assume that the carrier wipe-off process of Figure 2.22 is a Costas PLL. The incoming SV carrier signal is respectively in-phase and $\pi/2$ out of phase (quarter-phase) with the replicates cosine and the inverted sine. The result is maximum $I_{P,i}$ amplitude at the in-phase output and minimum $Q_{P,i}$ amplitude at quarter-phase output. With a Costas PLL, it is straightforward to detect the bits in the SV data message stream. The $I_{P,i}$ samples are simply accumulated for one data bit interval and the sign of the result is the data bit. Since there is a π phase ambiguity with a Costas PLL, the detected data bit stream may be correct or inverted. This ambiguity is resolved during the frame synchronization process by comparing the known preamble at the beginning of each sub-frame.

2.10.2 Frequency Locked Loop

PLL replicates the exact frequency of the incoming SV signal to perform the carrier wipe-off function, whereas the FLL replicates the approximate frequency. The FLL of GPS must be insensitive to π reversals in the in-phase and quarter-phase signals. Table 2.4 summarizes several GPS receiver FLL discriminators, their output frequency errors, and their characteristics [War96b]. In this table, the cross and dot operations are achieved with the following identities:

$$\begin{aligned} cross &= I_{P,i-1} \cdot Q_{P,i} - I_{P,i} \cdot Q_{P,i-1} \\ dot &= I_{P,i-1} \cdot I_{P,i} + Q_{P,i-1} \cdot Q_{P,i} \end{aligned} \tag{2:31}$$

It is more complicated to detect the bits in the SV data message stream using an FLL. At the beginning, the sign of the first bit is set arbitrary. Then, a differential process (such as dot product) is used to detect the sign changes. Like for the PLL, the sign ambiguity is resolved later during frame synchronization process by comparing the known preamble at the beginning of the each sub-frame. Generally, in an integrated Costas PLL the Bit Error Rate (BER) performance is 1.5 dB better than in differential FLL [Die95d].

Discriminator Algorithm	Output Frequency Error	Characteristics
$\frac{cross}{2\pi \cdot (t_2 - t_1)}$	$\frac{\sin(\Delta\Phi_2 - \Delta\Phi_1)}{2\pi \cdot (t_2 - t_1)}$	Near optimal at low SNR. Slope proportional to SNR. Least computational burden.
$\frac{\text{sign}(dot) \cdot cross}{2\pi \cdot (t_2 - t_1)}$	$\frac{\sin(\Delta\Phi_2 - \Delta\Phi_1)}{2\pi \cdot (t_2 - t_1)}$	Near optimal at low SNR. Slope proportional to SNR. Moderate computational burden.
$\frac{\text{atan2}(cross, dot)}{2\pi \cdot (t_2 - t_1)}$	$\frac{\Delta\Phi_2 - \Delta\Phi_1}{2\pi \cdot (t_2 - t_1)}$	Four-quadrant arctangent. Optimal at high and low SNR. Slope independent on SNR. Highest computational burden.

Table 2.4: FLL discriminators.

2.10.3 FLL assisted PLL

A well-designed FLL will outperform a well-designed PLL under dynamic stress and Radio Frequency Interference (RFI) conditions. However, the PLL will significantly outperform the FLL for measurement accuracy and data demodulation. Phillip Ward has proposed an innovative carrier tracking loop technique, which integrates both the FLL and the PLL characteristics. This novel FLL-assisted-PLL solves the GPS receiver designer's dilemma when faced with the need for both the dynamic robustness of the FLL plus the accuracy performance of the PLL [War98].

Figure 2.23 represents the block diagram of a first order FLL assisting a second order PLL. Both tracking loops are insensitive to velocity stress, but are sensitive to acceleration stress. Note that if the PLL error is zeroed, the filter becomes a pure first order FLL design and if the FLL error is zeroed, the filter becomes a pure second order PLL.

When the FLL-assisted-PLL is used, both discriminators must be applied to the respective loop filter inputs. Thus, the dynamics and robustness of the FLL plus the accuracy performance of the PLL are combined. Compared to a pure FLL, the data demodulation is easier and more accurate.

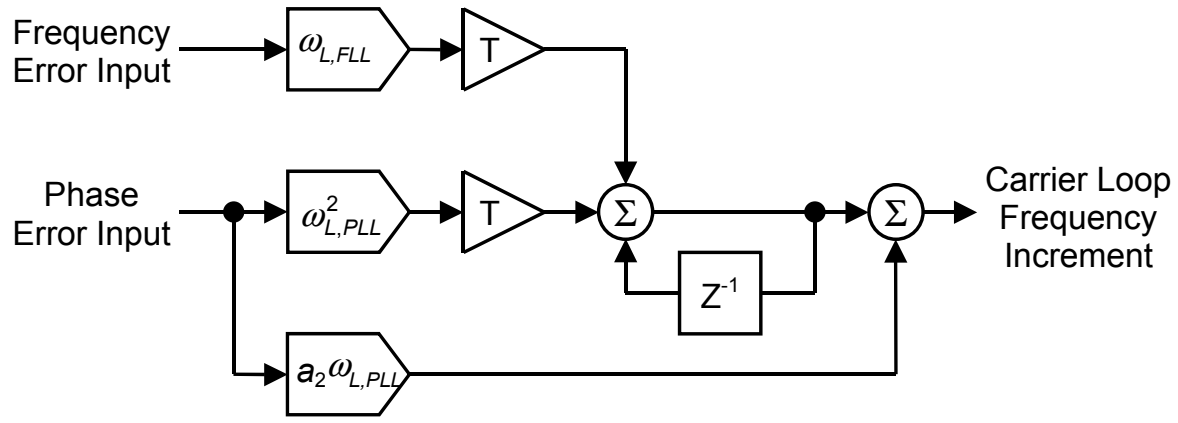


Figure 2.23: Second order PLL filter with first order FLL assist.

2.11 Code Tracking Loop

The design of the programmable pre-detection integrators, the code loop discriminators and the code loop filter characterize the receiver code-

tracking loop (Figure 2.24). Together, they determine the two most important performance characteristics of the receiver code-tracking loop design: the code loop thermal noise errors and the maximum line-of-sight stress threshold.

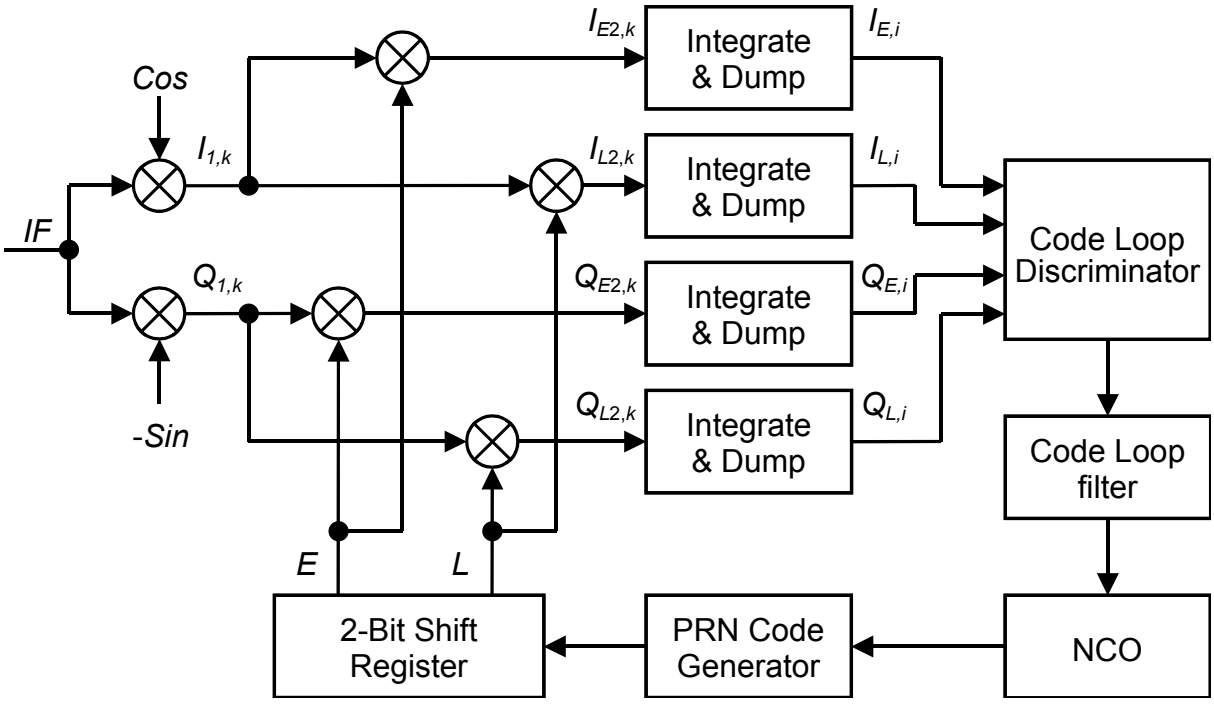


Figure 2.24: Generic block diagram of the code tracking loop.

2.11.1 Delay Locked Loop

There are two important Delay Locked Loop (DLL) discriminator families: the coherent and the non-coherent. The coherent DLL requires parallel carrier-phase tracking (and thus, a coherent qualifier). The non-coherent DLL does not require carrier tracking as long as the frequency offset is small with respect to the pre-detection bandwidth. Both discriminator families have their advantages and disadvantages. In particular, the non-coherent discriminators are more robust. The main advantage of the coherent DLL is that a hardwired “quarter-phase channel” is not required and thus “gate count” is reduced.

Table 2.5 summarizes four GPS receiver non-coherent DLL discriminators and their characteristics. The table assumes ½ chip spacing between the early, prompt and late correlator. The fourth DLL discriminator is a normalized version of the third discriminator. The other two DLL discriminators can also be normalized in a similar manner. Note

that in the relationship above, $I_{E,i}$ and $Q_{E,i}$ represent the accumulated signal plus noise in the early channels, whereas the $I_{L,i}$ and $Q_{L,i}$ the accumulated signal plus noise in the late channels.

Discriminator Algorithm	Characteristics
$I_{E-L,i} \cdot I_{P,i} + Q_{E-L,i} \cdot Q_{P,i}$	Dot product power. Lowest computational load. Produces nearly true errors output within ±½ chip of input errors.
$I_{E,i}^2 + Q_{E,i}^2 - \left(I_{L,i}^2 + Q_{L,i}^2\right)$	Early Minus Late (EML) power. Moderate computational load. Essentially the same performance as early minus late envelope within ±½ chip of input errors.
$\sqrt{\left(I_{E,i}^2 + Q_{E,i}^2\right)} - \sqrt{\left(I_{L,i}^2 + Q_{L,i}^2\right)}$	EML envelope. Higher computational load. Produces good tracking error output within ±½ chip of input errors.
$\frac{\sqrt{\left(I_{E,i}^2 + Q_{E,i}^2\right)} - \sqrt{\left(I_{L,i}^2 + Q_{L,i}^2\right)}}{\sqrt{\left(I_{E,i}^2 + Q_{E,i}^2\right)} + \sqrt{\left(I_{L,i}^2 + Q_{L,i}^2\right)}}$	Normalized EML envelope. Highest computational load. Produces good tracking error output within ±1½ chip of input errors.

Table 2.5: Non-coherent DLL discriminators.

2.12 Loop Filter

The objective of the loop filter is to reduce noise in order to produce an accurate estimate of the original signal at its output. The loop filter order and noise bandwidth (B_L) determine the loop filter's response to signal dynamics. The design approaches described here draws on existing knowledge of analogue loop filters, then adapts these into digital implementations. Figure 2.25 through Figure 2.27 show the block diagram of the first, second and third order analogue loop filters. In the figures, the analogue integrators are represented by $1/S$, which represents the Laplace transform of the time domain integration function.

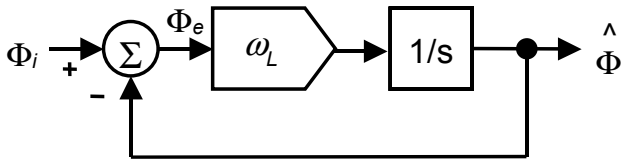


Figure 2.25: First order loop.

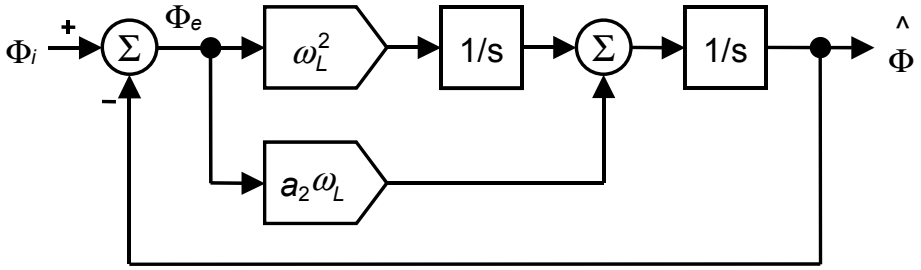


Figure 2.26: Second order loop.

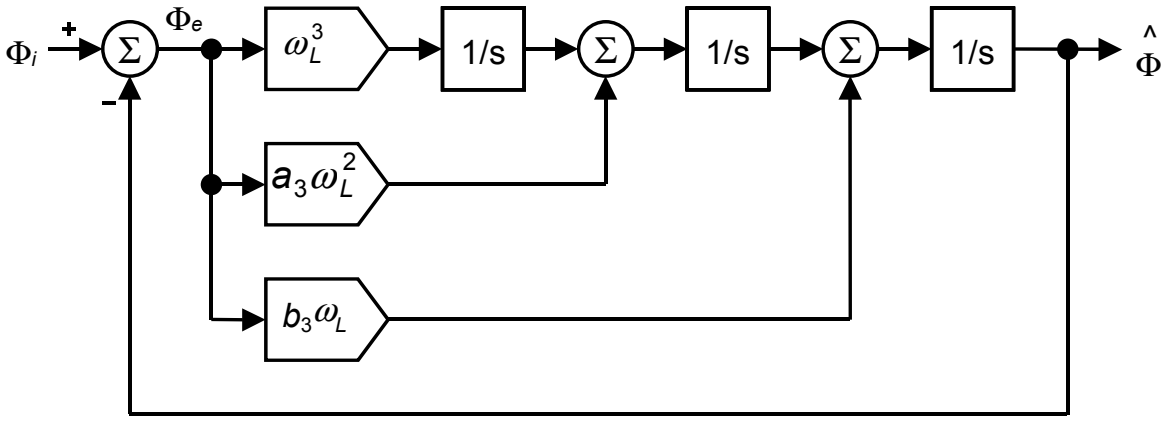


Figure 2.27: Third order loop.

Table 2.6 summarizes the filter characteristics and provides all the information required to compute the filter coefficients for the first, the second and the third order loop filters.

The first order loops are sensitive to velocity stress. They are used in aided code loops and sometimes in aided carrier loops. The second order loops are sensitive to acceleration. They are used in aided and unaided carrier loops. The third order loops are sensitive to jerk. They are used in unaided carrier loops.

The first and second order loops are unconditionally stable at all noise bandwidths, whereas the third order loops remains stable only if $B_L \leq 18$ Hertz [War96b].

Filter order	Transfer function	Typical Filter Values
First	$H(s) = \frac{\omega_L}{s + \omega_L}$	$B_L = 0.25 \cdot \omega_L$
Second	$H(s) = \frac{a_2 \cdot \omega_L \cdot s + \omega_L^2}{s^2 + a_2 \cdot \omega_L \cdot s + \omega_L^2}$	$a_2 = \sqrt{2}$ $B_L = 0.53 \cdot \omega_L$
Third	$H(s) = \frac{b_3 \cdot \omega_L \cdot s^2 + a_3 \cdot \omega_L^2 \cdot s + \omega_L^3}{s^3 + b_3 \cdot \omega_L \cdot s^2 + a_3 \cdot \omega_L^2 \cdot s + \omega_L^3}$	$a_3 = 1.1$ $b_3 = 2.4$ $B_L = 0.7845 \cdot \omega_L$

Table 2.6: Transfer function & typical filter values.

2.13 Measurement Errors & Tracking Thresholds

The GPS measurement errors and tracking thresholds are closely related because the receiver loses lock when the measurement errors exceed a certain boundary. As the code and carrier tracking loops are non-linear, especially near the threshold regions, only Monte Carlo simulations of the GPS receiver will determine the true tracking performance. However, rules of thumb can be used based on equations that predict the measurement errors of the tracking loops.

2.13.1 Phase Locked Loops

The dominant sources of phase errors in a GPS receiver are phase jitters and dynamic stress errors. The 3-sigma values of the PLL errors and the rule of thumb tracking threshold are computed as follows [War96b].

$$3 \cdot \sigma_{PLL} = 3 \cdot \sigma_i + \theta_e \leq \pi/4 \qquad (2:32)$$

Where σ_i represents the 1-sigma phase jitter from all sources except stress errors and θ_e the dynamic stress error in the PLL. The phase jitter is the root sum square of every source of non-correlated phase errors, such as thermal noise and oscillator noise. Oscillator noise includes vibration-induced jitters and Allan deviation induced jitters. In static applications the vibration-induced component can be ignored. Thus, it will not be taken in consideration furthermore.

The conservative rule-of-thumb threshold for a PLL is that the 3-sigma phase errors from all causes should not exceed $\pi/4$. This means that in 95% of all cases the loop measurement errors remain in the monotone fraction of the Costas phase discriminators. Therefore, the 1-sigma rule of thumb threshold for the PLL tracking loop is as follows:

$$\sigma_{PLL} = \sqrt{\sigma_{t,PLL}^2 + Q_A^2} + \frac{\theta_e}{3} \leq \frac{\pi}{12} \tag{2:33}$$

Where $\sigma_{t,PLL}$ and Q_A represent respectively the 1-sigma PLL thermal noise and the Allan variance induced by the oscillator jitters.

PLL Thermal Noise

Often the thermal noise is treated as the only sources of carrier tracking errors, because the other sources of PLL jitters may be either transient or negligible. Figure 2.28 shows a Costas PLL, which can be used for the determination of the PLL thermal noise [Die95c].

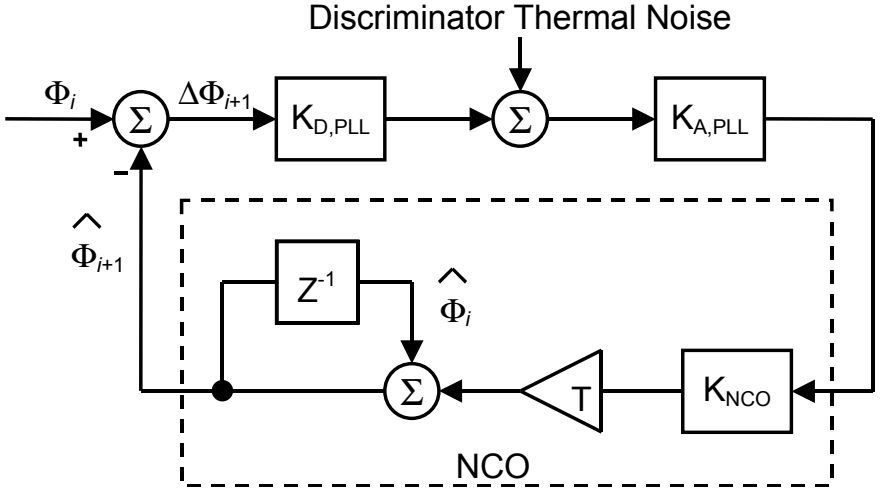


Figure 2.28: Generic linear tracking loop.

The NCO frequency gain K_{NCO} corresponds to the NCO frequency resolution. Assuming Gaussian approximation, the discriminator gain $K_{D,PLL}$ for the Costas PLL is given by:

$$K_{D,PLL} = \frac{\partial}{\partial \Phi}\left(I_{P,i} \cdot Q_{P,i}\right)\Big|_{\Delta\Phi=0} = 2T \cdot \frac{C}{N_0} \cdot \sigma_n^2 \tag{2:34}$$

Note that the discriminator gain varies with the CNR. To maintain the loop bandwidth, the loop gain must be compensated with the loop gain

adjustment factor $K_{A,PLL}$. Always assuming Gaussian approximation, the variance of the discriminator thermal noise is the following:

$$\sigma^2_{\delta\phi,i} = 2T \cdot \frac{C}{N_0} \cdot \left(1 + \frac{1}{2T \cdot C/N_0}\right) \cdot \sigma_n^4 \qquad (2:35)$$

For the first order loop, the closed loop time update errors $\Delta\Phi_i$ equation is of the following:

$$\Delta\Phi_{i+1} = (1 - 4B_{L,PLL} \cdot T) \cdot \Delta\Phi_i + K_{A,PLL} \cdot K_{NCO} \cdot T \cdot \sigma_{\delta\phi,i} \qquad (2:36)$$

In this relationship $\sigma_{\delta\phi,i}$ and $B_{L,PLL}$=$K_{D,PLL} \cdot K_{A,PLL} \cdot K_{NCO}/4$ represent respectively the discriminator thermal noise and the single sided loop noise bandwidth. The variance time update equation is the expected value of the square of both sides of equation (2:36). Accounting for the fact that the noise in $\sigma_{\delta\phi,i}$ is independent on the noise in all previous errors $\Delta\Phi_i$, the variance time update equation becomes:

$$\sigma^2_{\Phi,i+1} = (1 - 4B_{L,PLL} \cdot T)^2 \cdot \sigma^2_{\Phi,i} + \left(K_{A,PLL} \cdot K_{NCO} \cdot T\right)^2 \cdot \sigma^2_{\delta\phi,i} \qquad (2:37)$$

However, in steady state ${\sigma_{t,PLL}}^2$=${\sigma_{\Phi,i+1}}^2$=${\sigma_{\Phi,i}}^2$, therefore the 1-sigma thermal noise becomes:

$$\begin{aligned} \sigma_{t,PLL} &= \sqrt{\frac{\left(K_{A,PLL} \cdot K_{NCO} \cdot T\right)^2 \cdot \sigma^2_{\delta\phi,i}}{8B_{L,PLL} \cdot T \cdot (1 - 2B_{L,PLL} \cdot T)}} \\ &\approx \sqrt{\frac{B_{L,PLL}}{C/N_0} \cdot \left(1 + \frac{1}{2T \cdot C/N_0}\right)} \quad [\text{radian}] \end{aligned} \qquad (2:38)$$

Note that the term $2B_{L,PLL} \cdot T$ is neglected, which assumes that the loop gain is small enough not to cause stability problems (e.g. $2B_{L,PLL} \cdot T \ll 1$). Note also that the equation (2:38) does not include factors relating to C/A-code or P(Y)-code or the loop filter order. The PLL thermal noise is strictly dependent on the CNR (C/N_0), the PLL loop filter noise bandwidth ($B_{L,PLL}$), and the pre-detection integration time (T). If the CNR increases, the PLL thermal noises decrease. Reducing $B_{L,PLL}$ has the same effect at the expense of reduced dynamic stress threshold. Therefore, great care is recommended concerning the $B_{L,PLL}$ reduction. The part of the equation involving the pre-detection interval T is called the squaring loss, which in turn also decreases the thermal noise.

Allan Deviation Oscillator Phase Noise

The equations used to determine the Allan deviation σ_A are empirical. They are stated in terms of what the requirements are for the short-term stability of the reference oscillator as determined by the Allan variance method of stability measurement. The equation for a second order PLL is:

$$Q_{A,2} = \frac{\omega_0 \cdot \tau}{2.5} \cdot \sigma_A(\tau) \tag{2:39}$$

Where ω_0 and τ represent respectively the L-band input frequency in radian per second and the short-term stability gate time for Allan variance measurement in second. The short-term stability gate time must be evaluated at the noise bandwidth of the carrier loop filter τ=1/$B_{L,PLL}$. The equation for the third order PLL is similar:

$$Q_{A,3} = \frac{\omega_0 \cdot \tau}{2.25} \cdot \sigma_A(\tau) \tag{2:40}$$

Dynamic Stress Errors

Table 2.7 shows the dynamic stress errors obtained from the steady state error formulas. These errors depend on the loop bandwidth and the loop order.

Loop Order	Steady-State Errors [radian]
First	$\theta_{e,1} = \frac{\partial R/\partial t}{\omega_L}$
Second	$\theta_{e,2} = \frac{\partial^2 R/\partial t^2}{\omega_L^2}$
Third	$\theta_{e,3} = \frac{\partial^3 R/\partial t^3}{\omega_L^3}$

Table 2.7: Steady state errors.

2.13.2 Frequency Locked Loop

The dominant sources of frequency errors in a GPS receiver FLL are thermal noise frequency jitters and dynamic stress errors. The rule of thumb tracking threshold for the FLL is that the 3-sigma value of the jitters due to all sources of loop stresses must not exceed $\pi/2$ in one pre-detection interval T:

$$3\sigma_{FLL} = 3\sigma_{t,FLL} + f_e \leq \frac{1}{4T} \tag{2:41}$$

In this relationship $\sigma_{t,FLL}$ and f_e represent respectively the thermal noise frequency jitter and the dynamic stress error in the FLL. The reference oscillator vibration and Allan deviation induced frequency jitters are small order effects on the FLL and are considered as negligible.

The conservative rule-of-thumb threshold for a FLL is that the 3-sigma frequency errors from all causes should not exceed $1/4T$. So in 95% of all cases the loop measurement errors remain in the monotone fraction of the frequency discriminators. The 1-sigma frequency jitter would be $1/12T$.

The FLL thermal noise frequency jitter can be determined like the PLL thermal noise [War96b]:

$$\sigma_{t,FLL} = \frac{1}{2\pi \cdot T}\sqrt{\frac{4F \cdot B_{L,FLL}}{C/N_0}\left(1+\frac{1}{T \cdot C/N_0}\right)} \quad [\text{Hz}] \tag{2:42}$$

Where F=1 at high CNR and F=2 near threshold. Note that the thermal noise frequency jitters are independent of the PRN codes and the loop order. Because the FLL tracking loop involves one more integrator than PLL tracking loop of the same order, the dynamic stress errors are given by:

$$\begin{aligned} f_e &= \frac{\delta}{\delta t}\left(\frac{1}{2\pi \cdot \omega_L^n} \cdot \frac{\delta^n R}{\delta t^n}\right) \\ &= \frac{1}{2\pi \cdot \omega_{L,FLL}^n} \cdot \frac{\delta^{n+1} R}{\delta t^{n+1}} \quad [Hz] \end{aligned} \tag{2:43}$$

In this relationship n represents the loop order.

2.13.3 FLL Assisted PLL

According to P. Ward a rules of thumb cannot be used with the FLL-assisted-PLL. He has simulated two second-order FLL designs ($B_{L,FLL}$=2 Hz and $B_{L,FLL}$=4 Hz) and one third-order PLL design ($B_{L,PLL}$=18 Hz), under both high and low dynamic stresses. In addition, two FLL-assisted-PLL designs were also simulated under the same conditions. Table 2.8 summarizes the Monte Carlo simulator carrier tracking threshold results. It illustrates a very significant characteristic of the FLL-assisted-PLL design: it only outperforms the counterpart PLL only design, if the FLL only threshold is lower than the PLL only one.

Parameter	Units	FLL	FLL	FLL/ PLL	FLL/ PLL	PLL
Noise bandwidth	Hz	2	4	2/18	4/18	18
CNR threshold - 10 g/s	dB-Hz	31.3	23.9	28	25.8	28.5
CNR threshold - 1 g/s	dB-Hz	20.0	22.5	26.8	24.6	27.5

Table 2.8: Summary of Monte Carlo simulator tracking threshold results.

Using the design insight above, the preferred carrier loop design for the FLL should outperform the PLL under both high and low dynamic stresses. This can be determined with the carrier-tracking thresholds obtained from the rule of thumb analysis.

Now consider an FLL assisted PLL design, where the FLL outperforms the PLL under both high and low dynamic stresses. Every time when the CNR is below the PLL threshold, the phase discriminator is ineffective. This means that the FLL-assisted-PLL can be considered as a pure FLL where additional noise from the phase discriminator is injected. Since the thermal noise in the FLL is independent of the thermal noise in the PLL. The 1-sigma FLL-assisted-PLL thermal noise can be obtained by superposition of both FLL and PLL thermal noise contributions. However, the phase error must be converted into a frequency error. Therefore, the 1-sigma PLL thermal noise must be divided by $2\pi \cdot T$, where T represents the loop update interval.

$$\sigma_{t,FLL/PLL} = \sqrt{\sigma_{t,FLL}^2 + \left(\frac{\sigma_{t,PLL}}{2\pi \cdot T}\right)^2} \,[Hz] \qquad (2:44)$$

The rule of thumb tracking threshold for the FLL-assisted-PLL can be deduced from the FLL rule of thumb tracking threshold of equation (2:41), where the 1-sigma FLL thermal noise is replaced by the 1-sigma FLL assisted PLL thermal noise:

$$3\sigma_{FLL/PLL} = 3\sigma_{t,FLL/PLL} + \frac{\partial^{n+1} R / \partial t^{n+1}}{2\pi \cdot \omega_{L,FLL}^n} \leq \frac{1}{4T} \qquad (2:45)$$

The analyses using the presented rule of thumb threshold are quite accurate. Table 2.9 compares the analyzed threshold performances with the simulated ones, obtained by P. Ward. According to the Monte Carlo simulations, all analyses using the rule of thumb are accurate predictions.

Parameter	Units	FLL	FLL/PLL	PLL
Filter Order	-	2	2/3	3
Noise Bandwidth [Hz]	Hz	4	4/18	18
Pre-detection integration time	s	0.01	0.01/0.02	0.02
Simulation CNR threshold – 10 g/s	dB-Hz	23.9	25.8	28.5
Analysis CNR threshold – 10 g/s	dB-Hz	25.6	27.1	28.0
Difference in tracking threshold	dB	-1.7	-1.3	0.5
Simulation CNR threshold – 1 g/s	dB-Hz	22.5	24.6	27.5
Analysis CNR threshold – 1 g/s	dB-Hz	22.8	24.0	24.8
Difference in tracking threshold	dB	-0.3	0.6	2.7

Table 2.9: Threshold comparison at 10 & 1 g/s dynamic stress.

2.13.4 Delay Locked Loop

The dominant sources of range errors in a GPS receiver code-tracking loop (DLL) are thermal noise range errors and dynamic stress errors. A rule of thumb tracking threshold for the DLL is that the 3-sigma value of

the jitter due to all sources of loop stress must not exceed the correlator spacing *d*, in chips. So, in 95% of the cases the loop measurement errors remain in the monotone fraction of the DLL discriminators. Therefore, a rule of thumb tracking threshold may be:

$$3\sigma_{DLL} = 3\sigma_{t,DLL} + R_e \leq d \qquad (2:46)$$

Where $\sigma_{t,DLL}$ and R_e represent respectively the thermal noise code tracking jitter and the dynamic stress error in the DLL. The thermal noise code tracking jitter can be determined like the PLL thermal noise [War96b]:

$$\sigma_{t,DLL} = \sqrt{\frac{4F_1 \cdot d^2 \cdot B_{L,DLL}}{C/N_0}\left[2 \cdot (1-d) + \frac{4F_2 \cdot d}{T \cdot C/N_0}\right]} \quad [\text{chips}] \qquad (2:47)$$

In this relationship, F_1 represents the discriminator correlator factor. F_1=1 for time-shared tau-dither early/late correlator and F_1=1/2 for dedicated early and late correlator. F_2 represents the DLL discriminator type factor. F_2=1 for early/late type discriminators and F_2=1/2 for dot product type discriminators. As seen in (2:47), the DLL thermal noise code tracking jitter is directly proportional to the square root of the filter noise bandwidth ($B_{L,DLL}$) and the correlator spacing (*d*). Reducing $B_{L,DLL}$ and *d* results in lower DLL thermal noise code tracking jitter, resulting in a lower CNR. Furthermore, reducing *d* helps the C/A-code tracking vulnerability with respect to multipath. However, great care is recommended because both reductions have an important impact on the dynamic stress threshold. The choice between good dynamic stress performance and precise measurement ability is a dilemma. The Carrier aided code loop resolves this dilemma by putting the dynamic stress burden on the carrier-tracking loop. This innovative technique permits the use of a very narrow loop-filter noise bandwidth.

The part of the equation involving the pre-detection interval *T* is called the squaring loss. Increasing the pre-detection interval *T* reduces the squaring loss. The DLL tracking loop dynamics is determined by:

$$R_e = \frac{1}{\omega_{L,DLL}^n} \cdot \frac{d^n R}{dt^n} \quad [chips] \qquad (2:48)$$

2.14 Signal Acquisition

GPS signal acquisition is a two-dimensional search process in the time (code phase or chip) and frequency (bin) domain. This search process, like the track process, requires replication of both the code and the carrier of the SV (Figure 2.29). Note that there may be numerous power peaks, while only one is the valid peak. The extra peaks in the frequency direction are due to the side lobes of the filter. The additional peaks in the code-offset direction are minor-correlation peaks. Signal acquisition procedures must be implemented to avoid locking on these invalid peaks.

The signal acquisition process consists of the following receiver processing tasks:

- *Determination of the noise floor to set the signal acquisition detector threshold.*
- *Determination of initial estimates of carrier and code Doppler (due to satellite movement) and associated Doppler uncertainty (due to receiver clock offset). The initial estimates are set in the carrier and code NCO. The uncertainty is used to fix the frequency range that must be scanned. The search pattern is typically from the mean value of the Doppler uncertainty and then symmetrically one Doppler bin at time on either side of this value until 3-sigma Doppler uncertainty has been reached.*
- *Determination of the received code phase estimate and associated uncertainty based upon the estimated pseudorange. Normally, for initial acquisition when the exact receiver time is unknown, the uncertainty is bounded by the entire code, and the coder can be initialized at any phase. If the uncertainty is less than one half the entire C/A-code, or for P(Y)-code acquisitions, the coder phase is usually initialized at the code phase estimates minus some fraction of the code uncertainty. The code search is always performed from the early side of the estimated correlation peak to prevent the acquisition of multipath.*
- *The search for the correlation peak is accomplished by stepping the coder from early to late through the search aperture and measuring signal plus noise power at each code phase position and comparing the power to a determined threshold. Usually, the early to late code spacing is set to half a code chip to ensure that near peak power is*

encountered during the search. Each code phase search increment is called a code bin. The combination of a code bin and a Doppler bin is a cell.

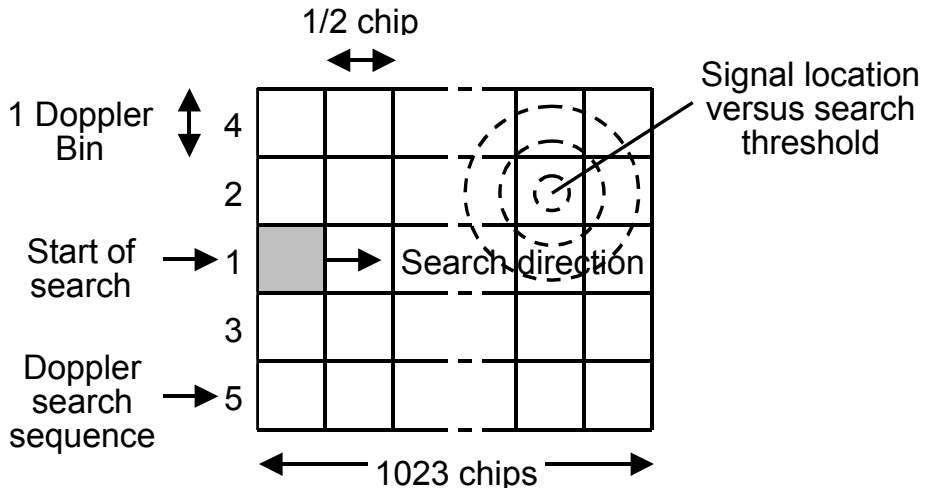


Figure 2.29: Two-dimensional C/A-code search pattern.

2.14.1 Fixed Integration Time Detection

Figure 2.30 illustrates the fixed integration time detection algorithm. At the input, the samples are those after demodulation and correlation. *M* of these samples are summed to narrow the predetection bandwidth by a factor of 1/*M*. Then, the energy of the signal is computed and compared against a threshold *TH,* to determine if the signal is present or not.

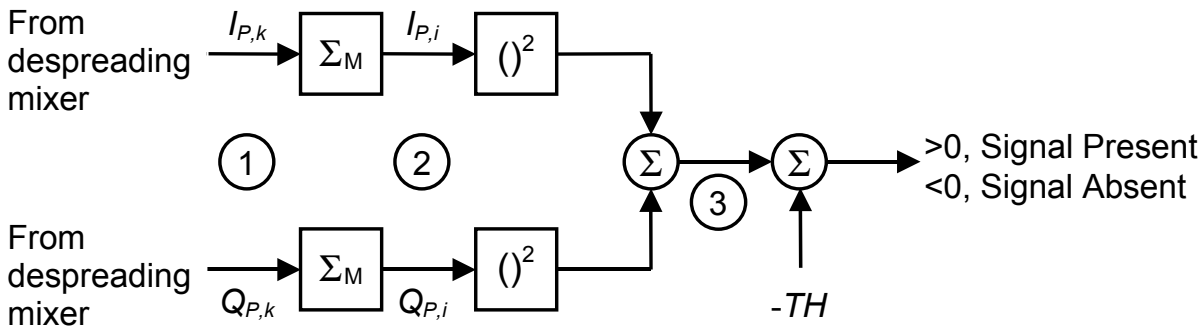


Figure 2.30: Signal acquisition model.

The detection of the signal is a statistical process, because each measurement contains noise whilst the signal is present or absent. Both cases have their own Probability Density Function (PDF). The two

statistics that are of most interest for the signal detection process are the probability of detection, P_d, and the probability of false alarm, P_{fa}. These are determined as follows:

$$P_d = \int_{TH}^{\infty} P_S(z) \cdot dz \tag{2:49}$$

$$P_{fa} = \int_{TH}^{\infty} P_N(z) \cdot dz \tag{2:50}$$

Here $P_S(z)$ and $P_N(z)$ represent respectively the PDF of the energy in the presence and absence of the signal. To determine these PDF, assume that I and Q have a Gaussian distribution. Then, $P_S(z)$ is a Ricean distribution as defined by [War96c].

$$P_S(z) = \frac{z}{\sigma_n^2} \cdot \exp\left\{ -\left(\frac{z^2}{2\sigma_n^2} + T \cdot C/N_0 \right) \right\} \cdot I_0\left(\frac{\sqrt{2T \cdot C/N_0}}{\sigma_n} \cdot z \right) \tag{2:51}$$

In the relationship above, $I_0(x)$ represents the modified Bessel function of the first kind. For the case where there is no signal present, evaluating (2:51) for C/N_0=0 yields a Rayleigh distribution for $P_N(z)$, which is defined by:

$$P_N(z) = \frac{z}{\sigma_n^2} \cdot \exp\left(-\frac{z^2}{2\sigma_n^2} \right) \tag{2:52}$$

The result of integrating (2:50) using the PDF of (2:52) is:

$$P_{fa} = \exp\left(-\frac{TH^2}{2\sigma_n^2} \right) \tag{2:53}$$

Rearranging (2:53) yields the threshold of the desired probability of false alarm:

$$TH = \sigma_n \cdot \sqrt{-2 \ln P_{fa}} \tag{2:54}$$

For example, if it is desired that P_{fa}=13.5%, then TH=2σ_n.

2.15 Fast Signal Acquisition

Fixed integration time detection is limited, because only the pre-detection interval T and threshold TH can be varied to reduce synchronization time. With the SNR and the number of uncertainty cells fixed, the selection of the T and TH completely determine P_d, P_{fa} and therefore average synchronization time. However, there is only one correct cell within the uncertainty region. So, most of the cells evaluated by the energy detector are noise alone and an energy detection scheme that is able to reject incorrect phase cells rapidly is desirable.

2.15.1 Multiple Dwell Detection

The Multiple-Dwell Detection (MDD) scheme reduces the synchronization time, using multiple evaluations of the same phase cell. The first evaluation is very short and results in immediate rejection of many incorrect cells. The short pre-detection integration time of this first evaluation also results in a high false alarm probability.

When a false alarm occurs on the first evaluation, a second evaluation of the same cell begins, using a longer pre-detection integration time. This second evaluation is able of rejecting most of the false alarms of the first evaluation. The second evaluation may be followed by a third, fourth or as many as desired to achieve a particular performance level [Pet95].

Figure 2.31 illustrates the flow diagram for a typical multiple dwell system. Using whatever a priori information available, the system begins by selecting a code phase and frequency bin for evaluation. At the end of the first detection interval T_1, the output of the energy detector V is compared with a threshold V_1. If $V<V_1$, a “miss” (incorrect phase) is declared and the phase cell is rejected. If $V>V_1$ a “hit” (potential correct phase cell) is declared and a second detection begins. At the end of the second detection interval T_2, the output of the energy detector V is compared with V_2. If $V<V_2$, the phase is rejected and if $V>V_2$, the phase remains potentially correct. When $V>V_2$, a third detection begins, after which the phase is either rejected or declared correct.

Figure 2.31 includes also a logic provision for continuing to evaluate the presence of signal during tracking. Detection 4 and 5 provide a mean for detecting lost lock. These two integration times may be quite large since an incorrect dismissal of the correct code phase is very serious.

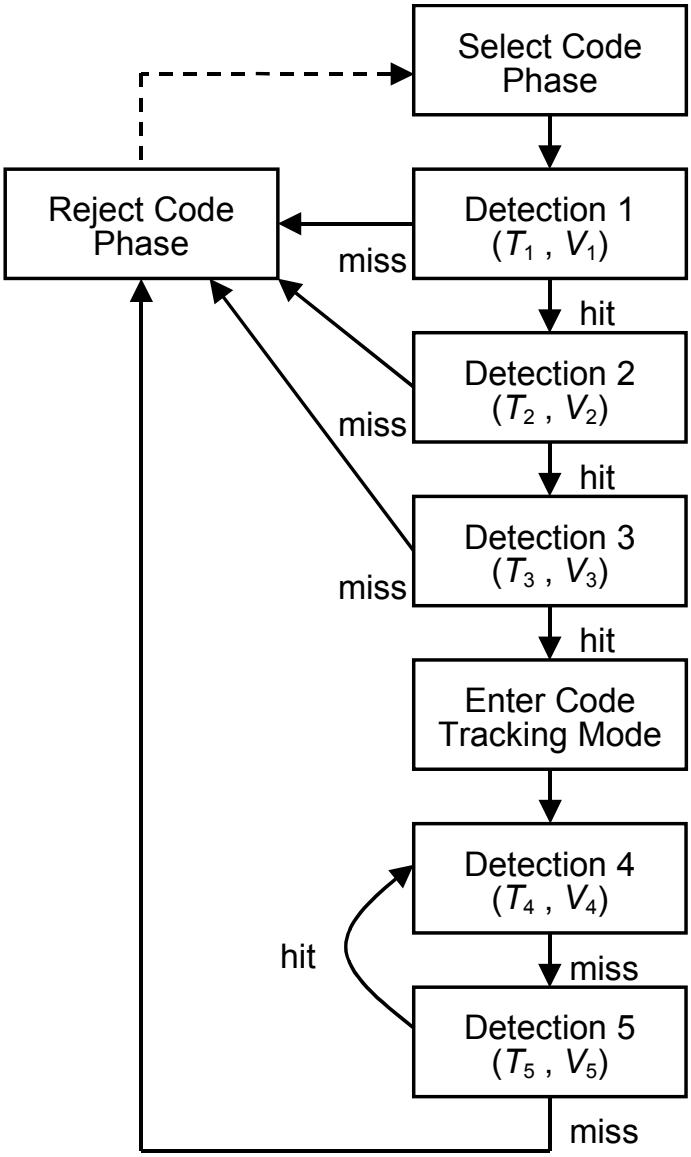


Figure 2.31: Typical logic diagram for a MDD.

2.16 Summary & Conclusions of the Chapter

In the first part of this chapter, a description of the GPS reception process is presented, followed by a high-level block diagram of a digital GPS receiver. The second part of this chapter treats the detailed descriptions of the DSP functions, starting with the hardware and following with the software.

Two novelties are also presented in this chapter. The first is a demonstration of the correlation losses, which can be used to determine the effect of the pre-sampling filtering on the PRN code discriminator (section 2.5.2). The second innovation is the determination of a rule of

thumb tracking threshold for the FLL assisted PLL that solves the GPS receiver's designer dilemma, when faced with both the dynamics and robustness of the FLL plus the accuracy of the PLL (section 2.13.3).

2.17 Appendix

SV	GPS PRN	G2-Code Tap Selection	G2-Code Delay [Chips]	SV	GPS PRN	G2-Code Tap Selection	G2-Code Delay [Chips]
1	1	2 ⊕ 6	5	19	19	3 ⊕ 6	471
2	2	3 ⊕ 7	6	20	20	4 ⊕ 7	472
3	3	4 ⊕ 8	7	21	21	5 ⊕ 8	473
4	4	5 ⊕ 9	8	22	22	6 ⊕ 9	474
5	5	1 ⊕ 9	17	23	23	1 ⊕ 3	509
6	6	2 ⊕ 10	18	24	24	4 ⊕ 6	512
7	7	1 ⊕ 8	139	25	25	5 ⊕ 7	513
8	8	2 ⊕ 9	140	26	26	6 ⊕ 8	514
9	9	3 ⊕ 10	141	27	27	7 ⊕ 9	515
10	10	2 ⊕ 3	251	28	28	8 ⊕ 10	516
11	11	3 ⊕ 4	252	29	29	1 ⊕ 6	859
12	12	5 ⊕ 6	254	30	30	2 ⊕ 7	860
13	13	6 ⊕ 7	255	31	31	3 ⊕ 8	861
14	14	7 ⊕ 8	256	32	32	4 ⊕ 9	862
15	15	8 ⊕ 9	257	…	33	5 ⊕ 10	863
16	16	9 ⊕ 10	258	…	34	4 ⊕ 10	950
17	17	1 ⊕ 4	469	…	35	1 ⊕ 7	947
18	18	2 ⊕ 5	470	…	36	2 ⊕ 8	948

Table 2.10: Code assignment table.

The C/A-code for the SV identification number *i* is a Gold code, $G_i(t)$, of 1 millisecond in length at a chipping rate of 1.023 megabits per second.

The $G_i(t)$ sequences are linear patterns generated by the modulo-2 addition of sub-sequences, $G_1(t)$ and $G_{2,i}(t)$, each of which is a 1023 chip long linear pattern. As shown in Table 2.10, the $G_{2,i}(t)$ sequence is a $G_2(t)$ sequence selectively delayed by 5 to 950 chips, thereby generating a set of 36 mutually exclusive C/A-codes. The assignment of these codes by SV identification is also illustrated in Table 2.10 [Nav91].

2.18 References

[Cha89] H. Chang, "Presampling Filtering, Sampling and Quantification Effects on the Digital Matched Filter Performance", *Proceedings of the international telemetering conference (San Diego CA)*, pp 889-915, 1989.

[Die95a] A. -J. Van Dierendonck, "GPS Receivers", *Global Positioning System: Theory and Applications*, Vol. 1, pp 338-340, American Institute of Aeronautics and Astronautics Inc., 1996.

[Die95b] A. -J. Van Dierendonck, "GPS Receivers", *Global Positioning System: Theory and Applications*, Vol. 1, pp 356-365, American Institute of Aeronautics and Astronautics Inc., 1996.

[Die95c] A. -J. Van Dierendonck, "GPS Receivers", *Global Positioning System: Theory and Applications*, Vol. 1, pp 369-372, American Institute of Aeronautics and Astronautics Inc., 1996.

[Die95d] A. -J. Van Dierendonck, "GPS Receivers", *Global Positioning System: Theory and Applications*, Vol. 1, pp 395-397, American Institute of Aeronautics and Astronautics Inc., 1996.

[Gec95] "GP2010 GPS Receiver RF Front End" *Advance Information*, Gec Plessey Semiconductors, June 1995.

[Nav91] "Navstar GPS Space Segment" *Navigation User Interfaces*, Interface Control Document 1991.

[Par94] B. -W. Van Parkinson, "Introduction and Heritage of NAVSTAR, the Global Positioning System", *Global Positioning System: Theory and Applications*, Vol. 1, pp 3-12, American Institute of Aeronautics and Astronautics Inc., 1996.

[Pel89a] F. Pellandini, "Signaux et Systèmes", *EPFL Département de microtechnique,* pp 74, Vol. 1, 1989.

[Pel89b] F. Pellandini, “Signaux et Systèmes”, *EPFL Département de microtechnique*, pp 100-113, Vol. 1, 1989.

[Pel90] F. Pellandini, “Signaux et Systèmes”, *EPFL Département de microtechnique*, pp 229-232, Vol. 2, 1990.

[Pet95] R. –L. Peterson, R. –E. Ziemer and D. –E. Borth, *Introduction to Spread-Spectrum Communications*, pp 253-256, Prentice-Hall, 1995.

[Pro89] J. –G. Proakis and D. –G. Manolakis, *Introduction to Digital Signal Processing*, pp 17-18, Maxwell Macmillan International Editions, 1989.

[Spi94a] J. -J. Spilker Jr., “GPS Signal Structure and Theoretical Performance”, *Global Positioning System: Theory and Applications*, Vol. 1, pp 59-62, American Institute of Aeronautics and Astronautics Inc., 1996.

[Spi94b] J. -J. Spilker Jr., “GPS Signal Structure and Theoretical Performance”, *Global Positioning System: Theory and Applications*, Vol. 1, pp 82-84, American Institute of Aeronautics and Astronautics Inc., 1996.

[War96a] P. Ward, “GPS Satellite Signal Characteristics”, *Understanding GPS Principles & Applications*, pp 100-111, Artech House Publishers, 1996.

[War96b] P. Ward, “Satellite Signal Acquisition and Tracking”, *Understanding GPS Principles & Applications*, pp 132-175, Artech House Publishers, 1996.

[War96c] P. Ward, “Satellite Signal Acquisition and Tracking”, *Understanding GPS Principles & Applications*, pp 193-198, Artech House Publishers, 1996.

[War98] P. Ward, “Performance Comparisons Between FLL, PLL and a Novel FLL-Assisted-PLL Carrier Tracking Loop Under RF Interference Conditions”, *Proceedings of the 11th International Technical Meeting of the Satellite Division of the Institute of Navigation*, ION GPS-1998.

Chapter 3

Digital Design

In the first part of this chapter basic definitions for design processes are presented, which include also some Computer Assisted Development (CAD) tool notions. These definitions will accommodate the reader with the fundamental digital design concepts.

The second part of this chapter treats some aspects of the Complementary Metal Oxide Silicon (CMOS) static and dynamic power dissipations, which can be reduced by many optimization methods.

The remaining part presents two GPS-examples of algorithmic optimizations. The first is the complex sampling algorithm, which reduces the sampling frequency to the information bandwidth, allowing for large potential power savings. The second is a multiphase NCO that has lower power consumption, though maintaining the same frequency characteristics as a standard NCO.

3.1 Design Concepts

In the next sections basic definitions concerning digital design are introduced. These definitions will accommodate the reader with the most important digital design concepts. However, more details on this subject can be found in the reference literature as presented in section 3.8.

The first definition presented in this chapter concerns the “design” concept. Armstrong and Gray propose the following terminology [Arm93]:

> *Design:* A series of transformations from one representation of a system to another until a representation exists that can be manufactured.

Thus, a design can be represented at different levels of abstraction. For digital design, this abstraction hierarchy consists of six distinguishable levels.

3.1.1 Abstraction Hierarchy

The abstraction hierarchy of a digital design can be expressed in two domains [Arm93]:

> *Structural domain*: A domain in which a component is described in terms of an interconnection of more primitive components.
>
> *Behavioral domain*: A domain in which a component is described by defining its input versus output response.

Table 3.1 illustrates the six levels of abstraction hierarchy used by the digital designers. This hierarchy consists of the silicon, circuit, gate, register, chip, and system levels. The silicon level is the lowest level in this classification, whereas the system level is the highest. The design can be represented independently in one of these levels. Moving down in the hierarchy, the design is closer to a physical implementation and the representation is less abstract. Correspondingly the required amount of detail increases.

At the silicon level, the basic primitives are geometric shapes that represent areas of diffusion, oxide, poly-silicon, and metal on the silicon surface. The viewpoint here is structural only.

At the circuit level, the representation is that of traditional passive and active electrical circuit elements: resistor, capacitors, bipolar and Metal Oxide Silicon (MOS) transistors etc. The interconnection of components

is used to model circuit behavior in terms of voltage and current. The behavioral content at this level can be expressed in terms of differential equations.

The gate level has traditionally been the major design level for digital design. The basic primitives are the AND, OR and INVERT operators and various types of flip-flops and latches. Interconnection of these primitives forms combinational and sequential logic circuits. Boolean equations define the behavior at this level.

At the register level, the basic primitives are more complex digital functions such as registers, counters, multiplexers and ALU. These primitives are sometimes referred to as functional blocks. They also correspond to VLSI design macros. Behavioral descriptions at this level are focused on the data flow.

At the chip level, the structural primitives are complete blocks such as microprocessors, memories, serial and parallel ports. The behavioral content of a chip level model is defined in terms of input versus output response of the device. Hardware Description Languages (HDL) are commonly used here to code these algorithmic descriptions.

The top level in the structural hierarchy is the system level. The primitive elements of this level are self-contained units like computers, bus interface units, disk units, radar etc. The behavioral content of this level is frequently expressed in terms of performance specifications.

Abstraction hierarchy	**Behavioral Domain Representation**	**Structural Domain Primitives**
System	Performance specification	Computer, disk unit, radar
Chip	Algorithm	Microprocessor, RAM, ROM
Register	Data flow	Register, ALU, counter, MUX
Gate	Boolean equations	AND, OR, XOR, FF
Circuit	Differential equations	Transistor, R, L, C
Layout/Silicon	None	Geometrical Shapes

Table 3.1: Abstraction hierarchy commonly used in digital design.

3.1.2 Design Representations

An important consideration in the design process is whether to use a pictorial or a textual description. Historically, pictures have been the preferred representation for digital design. For example, block diagrams, timing diagrams and logic diagrams were the principal forms of representation. However, with the advent of VHSIC Hardware Description Language (VHDL), textual design descriptions have gained in popularity because they allow the design to be parameterized in a very precise way.

Generally, structural descriptions are pictorial and behavioral representations are textual. Some exemptions to this classification scheme are state diagrams and timing diagrams, which are pictorial but represent behavior. As to the question when pictures should be employed or when text should be used, the following generalization can be made.

> **Text is better for representing complex behavior; pictures are better for illustrating interrelationship.**

Excessive use of either one results in a loss of perspective. Thus, real design systems balance the use of text and pictures.

3.1.3 Design Synthesis

Synthesis is another important terminology in the design concept. Armstrong and Gray propose the following definition for the synthesis process [Arm93]:

> *Synthesis*: The process of transforming one representation of the design abstraction hierarchy into another representation.

Each step in the design process is referred to as a synthesis step, each of which will be a transformation from level *i* to level *j* with *i<j*. This methodology is commonly called “top–down” design methodology. The design cycle consists of the following transformations:

- *Transformation from a spoken language to an algorithmic representation.*

- *Transformation from an algorithmic representation to a data flow representation or to a gate level representation (algorithmic synthesis).*
- *Transformation from data flow representation to a structural logic gate representation (logic synthesis). During this process, the design representation moves from the behavioral to the structural domain.*
- *Transformation from a logic-gate representation to a layout representation (layout synthesis).*

3.1.4 Digital Design Rules

In order to improve the chances to first silicon success in the design process, [Cha97] suggests a set of basic rules to be respected called "the Ten Commandments for Excellent Digital Design":

- *All state machine outputs must be registered.*
- *Latches should never be used instead of registers.*
- *The state machine inputs, including the resets, must be synchronous.*
- *Beware of fast paths.*
- *Minimize the clock skew of thin clocks.*
- *Cross clock domains with the greatest caution. Synchronize the signals.*
- *No dead states in the state machines.*
- *Have no logic with asynchronous feedback.*
- *All decode logic must be designed carefully.*
- *Do not trust the simulator. Some mistakes may occur when the design is not good.*

Although these rules do not guarantee the design to be correct, they do help to make the design more robust and may avoid problems that would only arise after the mapping into silicon.

3.2 CAD Tools

Armstrong and Gray propose the following definition for CAD tool [Arm93]:

> *CAD Tool*: Software program, which assists in performing or automates a particular design function.

Over the past 30 years a large number of CAD tools have been developed to aid in the design process. The terminology used by

manufactures to describe a tool's function is not consistent. However, the CAD tools can be divided into four classes: editors, simulators, checkers and synthesizers.

Editors are either textual or graphic. Textual editors can be used to edit the natural language of the specification at the system level or to edit HDL text at the circuit through chip level. At the circuit level, the HDL can be SPICE; for the gate, register and chip levels VHDL can be used. Graphic editors are used at all levels. At the silicon level they are used to create a pattern of geometric shapes that imply silicon treatment. At higher levels, the editors are used to create block diagrams and schematics.

Simulators are either stochastic or deterministic. Stochastic simulation is carried out at the system level to determine for example the percentage of time that a particular unit is busy. Deterministic simulation is carried out at all levels.

Checkers and analyzers are employed at all levels. At the silicon level, design rule checkers are used to insure that the layout implies a circuit that can be fabricated reliably. Timing analyzers can be used to check for the longest path through logic circuit or system.

The optimizer and synthesizer change the form of the design representation to a new form, which is regarded as improved. For example, at the gate level a minimization program can be used that yields a simpler Boolean expression. At the register level, optimizers are used to determine the best combination of control sequence and data path. And, as it was pointed out in the previous section, various forms of synthesis programs are employed to move the representation to a lower level in the hierarchy and thus closer to fabrication.

3.3 Power Dissipation

Most of the research and development effort in the area of digital products has been oriented toward increasing the speed and the complexity of a single chip. However, as the density and size increase, the difficulty in providing adequate cooling represents a limit and power consumption becomes a main concern. The increasing demand of portable systems (GSM, GPS, portable computers etc.) has aggravated the need of low power devices.

There are two components that establish the amount of power dissipated in a CMOS circuit:

- *Static power dissipation.*
- *Dynamic power dissipation.*

Each of them contributes to the total power consumption of a device and, in the case of a portable device, to the device's autonomy [Wes93].

3.3.1 Static Power Dissipation

The static power dissipation follows from leakage current or other current drawn continuously from the power supply.

To illustrate this, consider a complementary CMOS inverter (Figure 3.1). If the input is zero, the n-device is "off" and the p-device is "on". The output voltage is thus the logic one. Alternatively, if the input is one, the n-device is "on" and the p-device is "off". The output voltage is thus the logic zero. Since no current flows into the gate terminal, and there is no DC current path from V_{DD} to V_{SS}, the resultant steady-state current is zero.

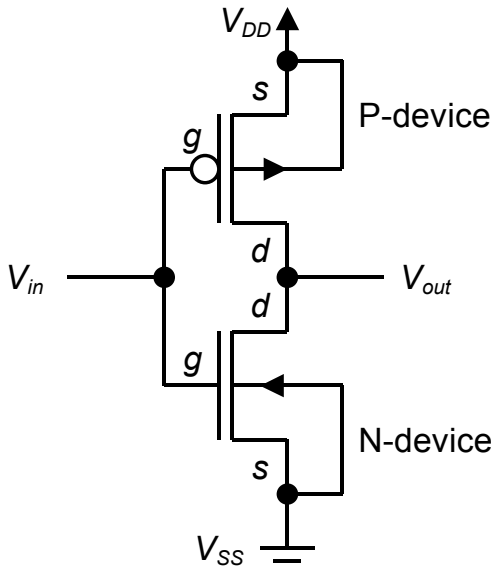


Figure 3.1: CMOS inverter.

However, there is some static dissipation due to reverse bias leakage current between diffusion regions and their substrate. Diffusion regions form parasitic diodes. Since the parasitic diodes are reverse-biased, only their leakage current contributes to static power dissipation. This current

is precisely described by the diode equation, but at room temperature, a useful estimation is to approximate the leakage current of 0.1 to 0.5 nA. The total static power dissipation P_S can now be obtained with the product of the leakage current and the supply voltage [Wes93]:

$$P_S = \sum_{1}^{n} \text{leackage current} \cdot \text{supply voltage} \qquad (3:1)$$

Where n represent the total number of devices.

3.3.2 Dynamic Power Dissipation

The dynamic power dissipation follows from:

- *Switching transient current.*
- *Charging and discharging of load capacitance.*

During transition either from zero to one or, alternatively, from one to zero, both transistors are “on” for a short period of time. The result is in a short current pulse from V_{DD} to V_{SS}. Current is also required to charge and discharge the output capacitive load, which is usually the dominant term.

The current pulse from V_{DD} to V_{SS} results in a “short-circuit” dissipation that is dependent on the input rise/fall time, the load capacitance and the gate design. In general, slow rising and falling edges increase the short circuit current.

Capacitive Dissipation

The capacitive dissipation can be modeled by assuming that the rise and fall time of the step input is much less than the repetition period. In this case, the average dynamic power dissipation (P_D), for a load capacitance (C_L), and a square-wave input having a repetition frequency of f_{CLK} is given by [Wes93]:

$$P_D = C_L \cdot V_{DD}^2 \cdot f_{CLK} \qquad (3:2)$$

This equation shows that the capacitive power dissipation is proportional to the switching frequency and the square of V_{DD}. However, it is independent on the device parameters.

Short Circuit Dissipation

Assume that both transistors in an inverter have the same electrical characteristics (e.g. the same β and same threshold voltage V_{TH}). Then, for an unloaded inverter, the short current dissipation (P_{SC}) is given by the following relationship [Wes93]:

$$P_{SC} = \frac{\beta}{12} \cdot t_{rf} \cdot f_p \cdot (V_{DD} - 2V_{TH})^3 \qquad (3:3)$$

Here, $t_{rf}=t_r=t_f$ represents the rising and falling time of the input waveform. The equation shows that short-circuit current depends on β and the input waveform rise and fall times. Slow rise times on nodes can result in significant (20%) short-circuit dissipation for loaded inverters. Therefore, it is a good practice to keep all edges fast if power dissipation is a concern. As the load capacitance is increased, the significance of short-circuit dissipation is dominated by the capacitive dissipation P_D.

3.3.3 Total Power Dissipation

The total power dissipation can be obtained from the sum of the three dissipation components:

$$P_{total} = P_S + P_D + P_{SC} \qquad (3:4)$$

When calculating the power dissipation, a rule of thumb is to add all capacitance operating at a particular frequency and calculate the power. Then, the power from the other groups operating at different frequencies is summed.

The dynamic power dissipation may be used to estimate the total power consumption of the circuit and also the size of V_{DD} and V_{SS} conductors to minimize transient-induced voltage drops.

3.4 Power Consumption Optimization

Minimizing power may be achieved in a number of ways. Static power dissipation may be reduced to leakage dissipation by only using complementary logic gates. This leakage is proportional to the area of the diffusion, so minimum sized devices are advantageous.

Reducing supply voltage, switch capacitance and the clock frequency, attenuates the dynamic power. Supply voltage tends to be a system

design consideration and low-power systems use 3 down to 1-Volt supplies (or even less).

Using minimum sized devices and optimal allocation of resources, such as adders and registers, reduces the switching capacitance. Manual layout techniques are also used to minimize routing capacitance.

Another big gain can be obtained at the circuit design level by operating the minimum amount of circuitry at high speeds or having a variable clock depending on how much computation has to be completed.

3.4.1 Technology Optimization

As already mentioned, reducing the supply voltage results in a drastic improvement of the dynamic power dissipation. Unfortunately, as V_{DD} approaches the threshold V_{TH}, a speed penalty is paid and the delay significantly increases. Since the objective is to reduce the power dissipation while keeping the throughput of the overall system fixed, the "technical" solution is to reduce the threshold voltage.

The technology engineers of the various foundries perform this optimization. The minimum to which the IC engineer can optimize V_{TH} is constrained by choosing a technology that will satisfy yield constraints while guaranteeing a minimal power supply.

3.4.2 Architectural Optimization

Lowering the supply voltage increases the delay. Therefore, architectural optimization techniques must be used to compensate the speed reduction.

One possibility is to parallelise the architecture. In this case, two modules in parallel are used instead of one module to implement the same function. Indeed, the throughput is maintained but each module works at half the frequency. Thus, the supply voltage can be reduced until the delay would be doubled. However, as a drawback, the hardware size is increased and some extra routing is required leading to an effective capacitance that is slightly more than doubled compared to the reference architecture.

Another solution to reduce the longest path is to pipeline the operations of the module by inserting registers. Again, the delay of each operation can be lengthened by reducing the supply voltage, until the most critical

path delay equals the desired throughput time. Each operation is still performed at the original rate, but the area is slightly increased due to the additional registers.

In both solutions, the effective capacitance will increase. Nevertheless, the quadratic dependence on V_{DD} still ensures a substantial improvement in the final power consumption.

Pipelining and parallelism can be explored together. At a certain point though, the overhead circuitry will dominate and no further power consumption improvement will be obtained.

3.4.3 Algorithmic Optimization

Algorithmic optimization is the most efficient way to reduce power consumption. It compromises the following strategies:

- *Choice of the algorithms.*
- *Simplification of the algorithms in order to reduce the complexity and to decrease the dynamic range needs.*
- *Study of the fixed-point quantification effects to determine the optimum scaling and minimum word length required at every node of the algorithm.*
- *Simplification of the interactions among the different algorithms inside the whole system.*
- *Good interaction between the algorithms and the target architecture.*

In the sections 3.5 and 3.6, two original and effective examples for algorithmic optimization are presented. The first is the complex sampling algorithm, which reduces the sampling frequency to the information bandwidth. The second example is a multiphase NCO, which consists of two phase-accumulators clocked at different frequency rates.

3.5 Complex Sampling

The complex sampling methodology is an effective and well-known algorithmic optimization, which is used in many applications to reduce both the power consumption and the frequency bandwidth. However, despite its excellent proprieties, the complex sampling algorithm is not used in the GPS domain.

This section contains the description of the complex sampling algorithm and its possible application in the GPS receivers. The impact on power

savings is also illustrated at the end of this section using a conventional GPS receiver.

3.5.1 Conversion to Baseband

In some GPS receivers, an additional conversion to baseband occurs at the output of the down converter. Figure 3.2 illustrates the process, which consists to convert the IF signal to that of in-phase and quarter-phase components of the signal envelope.

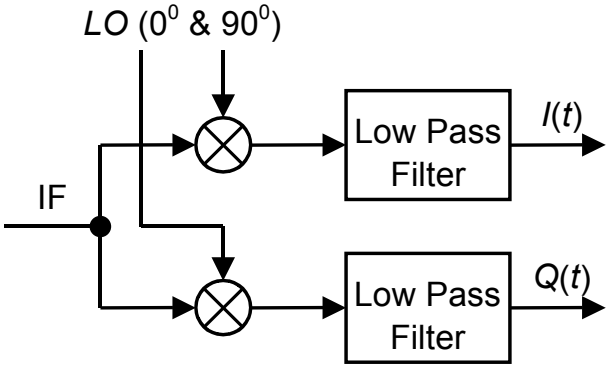


Figure 3.2: Conversion to baseband.

The conversion to baseband is achieved by mixing the IF signal with two LOs, one of which is shifted $\pi/2$ in the phase with respect to the other (in quarter phase). The low-pass filters reject the upper side bands. The in-phase and quarter phase combined LOs are respectively:

$$\begin{aligned} LO_I(t) &= 2\cos(\omega_{LO} \cdot t) \\ LO_Q(t) &= 2\cos(\omega_{LO} \cdot t + \pi/2) \\ &= -2\sin(\omega_{LO} \cdot t) \end{aligned} \qquad (3:5)$$

The resulting analogue in-phase and quarter-phase baseband signal components are then as follow:

$$\begin{aligned} I_s(t) &= A_0 \cdot C(t) \cdot D(t) \cdot \cos[(\omega_B + \Delta\omega) \cdot t + \Phi_0] \\ Q_s(t) &= A_0 \cdot C(t) \cdot D(t) \cdot \sin[(\omega_B + \Delta\omega) \cdot t + \Phi_0] \end{aligned} \qquad (3:6)$$

In the relationship above, $\omega_B = \omega_{IF} - \omega_{LO}$ contains always an intentional frequency offset. This is necessary, because the signal envelope is still modulated with residual Doppler. Therefore, to prevent folding of $\omega_B = 2\pi \cdot f_B$ thought DC, the baseband frequency is elevated above DC

and the offset is removed digitally in the DSP. In the same way, the ambient noise becomes as follow:

$$I_n(t) = x(t) \cdot \cos(\omega_B \cdot t) - y(t) \cdot \sin(\omega_B \cdot t)$$
$$Q_n(t) = x(t) \cdot \sin(\omega_B \cdot t) + y(t) \cdot \cos(\omega_B \cdot t) \quad (3:7)$$

Example

Figure 3.3 illustrates an example of a baseband converter manufactured by Trimble. The down-converter combines a first mixer, a 1400 MHz PLL and a pair of quarter-phase mixers [Tri96].

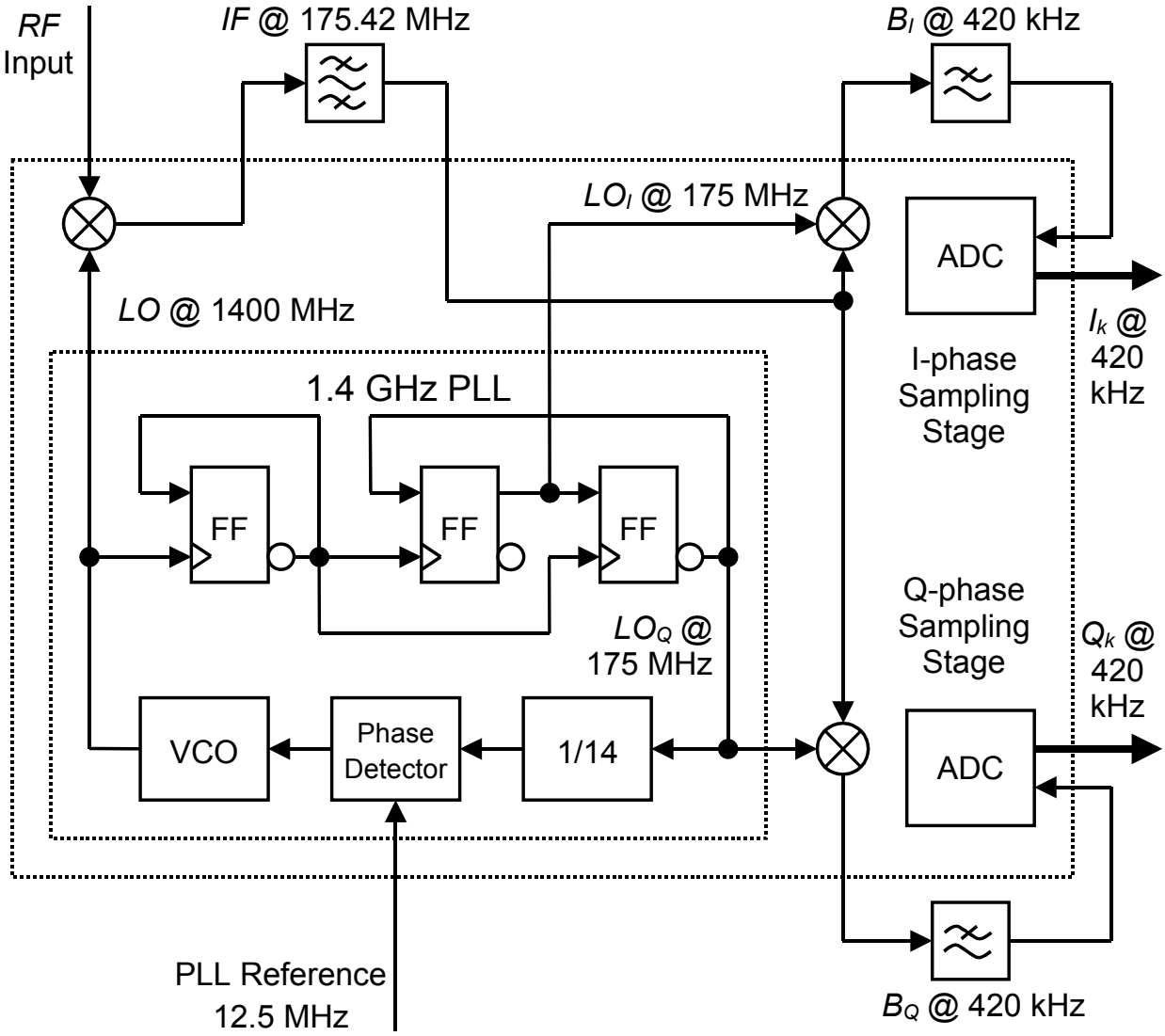


Figure 3.3: Trimble down-converter.

Given that the satellite transmits on two frequencies, L_1 at 1575.42 MHz and L_2 at 1227.6 MHz, the Voltage Controlled Oscillator (VCO) at 1400 MHz will produce a first intermediate frequency of 175.42 MHz or 172.4 MHz. The division by 8 action of the flip-flops causes a second intermediate frequency output of approximately 420 kHz or 2.6 MHz. This product is a pseudo-baseband signal.

3.5.2 Time Domain Sampling

Both $I_s(t)$ and $Q_s(t)$ can be combined together to form an analytical signal:

$$\begin{aligned} I_s(t)+i\cdot Q_s(t) &= A_0\cdot C(t)\cdot D(t)\cdot\{\cos[(\omega_B+\Delta\omega)\cdot t+\Phi_0]+i\cdot\sin[(\omega_B+\Delta\omega)\cdot t+\Phi_0]\} \\ &= A_0\cdot C(t)\cdot D(t)\cdot\exp\{i\cdot[(\omega_B+\Delta\omega)\cdot t+\Phi_0]\} \end{aligned} \tag{3:8}$$

The analytical signal above has a two-sided bandwidth of $2B$ Hz centered only at $f_B \pm \Delta f$. Figure 3.4 illustrates the power spectral density of the analytical signal (dashed lines) plus noise (solid lines) at the output of the GPS baseband converter.

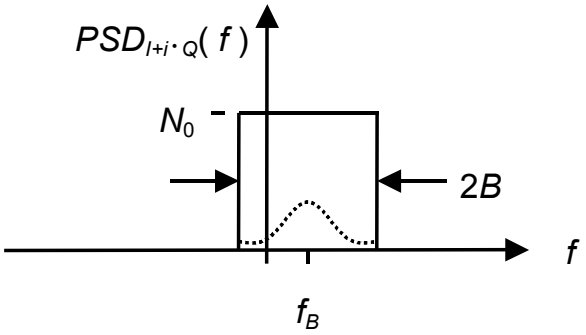


Figure 3.4: Power spectral density of the analytical signal plus noise.

The sampling frequency $f_s=1/T_s$ must be selected such that the sampling process will not result in any loss of spectral information (no aliazing). Consequently, the sampling process must be investigated by finding the relationship between the spectrum of the analogue signal and the spectrum of the digital signal. In the frequency domain, the sampling process results as a convolution between the spectrum of the analogue signal and the delta pulse train of the sampling frequency $\omega_s=2\pi\cdot f_s$:

$$S_{I_k+i\cdot Q_k}(\omega) \Leftrightarrow S_{I+i\cdot Q}(\omega) * f_s\cdot\sum_{n=-\infty}^{\infty}\delta(\omega-n\cdot\omega_s) \tag{3:9}$$

Therefore, the analytical baseband signal can be sampled at $2B$ samples per second, without information losses. Figure 3.5 illustrates the power spectral density of the analytical signal (dashed lines) plus noise (solid lines) after the complex sampling process.

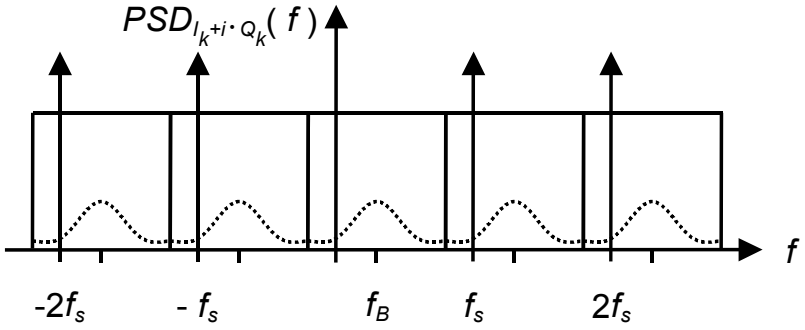


Figure 3.5: Complex sampling of an analytical signal.

The criterion above remains true, if the sampling is performed on each of the low-pass filter outputs. However, the in-phase and quarter-phase samples must be combined together to form an analytical signal. Neglecting the quantification losses, the digitized in-phase and quarter-phase signals can be represented as follow:

$$\begin{aligned} I_{sk} &= A_0 \cdot C_k \cdot D_k \cdot \cos(\Phi_k) \\ Q_{sk} &= A_0 \cdot C_k \cdot D_k \cdot \sin(\Phi_k) \end{aligned} \tag{3:10}$$

In this relationship C_k, D_k and $\Phi_k=(\omega_B+\Delta\omega)\cdot t_k+\Phi_0$ represent respectively the code, the data and the phase of the samples attributable to the nominal phase and frequency offset at time t_k. In the same way, the noise samples can be represented as follows:

$$\begin{aligned} I_{nk} &= x_k \cdot \cos(\Phi_k) - y_k \cdot \sin(\Phi_k) \\ Q_{nk} &= x_k \cdot \sin(\Phi_k) + y_k \cdot \cos(\Phi_k) \end{aligned} \tag{3:11}$$

3.5.3 Demodulation

The digitized baseband signal plus noise is demodulated at the input of the correlator. However, remember that the output of the complex sampling stage is an analytical signal. Therefore, the demodulation process must be implemented with the single sideband modulator.

In the time domain, the single sideband modulator is accomplished by the execution of the following trigonometric identities [Die95a]:

$$
\begin{aligned}
I_{1,sk} &= \mathrm{Re}\{(I_{sk} + i \cdot Q_{sk}) \cdot [\cos(\Phi_{rk}) - i \cdot \sin(\Phi_{rk})]\} \\
&= A_0 \cdot C_k \cdot D_k \cdot \cos(\Phi_k) \cdot \cos(\Phi_{rk}) + A_0 \cdot C_k \cdot D_k \cdot \sin(\Phi_k) \cdot \sin(\Phi_{rk}) \\
&= A_0 \cdot C_k \cdot D_k \cdot \cos(\Phi_k - \Phi_{rk}) \\
Q_{1,sk} &= \mathrm{Im}\{(I_{sk} + i \cdot Q_{sk}) \cdot [\cos(\Phi_{rk}) - i \cdot \sin(\Phi_{rk})]\} \\
&= A_0 \cdot C_k \cdot D_k \cdot \sin(\Phi_k) \cdot \cos(\Phi_{rk}) - A_0 \cdot C_k \cdot D_k \cdot \cos(\Phi_k) \cdot \sin(\Phi_{rk}) \\
&= A_0 \cdot C_k \cdot D_k \cdot \sin(\Phi_k - \Phi_{rk})
\end{aligned}
\tag{3:12}
$$

In the relationship above Φ_{rk} represents the phase of the reference carriers generated by the NCO at t_k. Note that the reference carriers are also combined together to form an analytical signal. Note also that the multiplication with the imaginary number vanishes in both cases, which simplifies the Doppler removal implementation. If both the incoming signal and the reference carrier are coded on 2 bits, the realization of the Doppler removal process is very easy and cheap.

In the frequency domain, the single sideband modulator leads to a convolution between the spectrum of the incoming baseband signal and the spectrum of the analytical carrier reference. Figure 3.6 illustrates the power spectral density of the analytical signal (dashed lines) plus noise (solid lines) before (Figure 3.6.a) and after (Figure 3.6.c) the Doppler Removal process.

In a similar way, the noise components are also rotated, but still random:

$$
\begin{aligned}
I_{1,nk} &= \mathrm{Re}\{(I_{nk} + i \cdot Q_{nk}) \cdot [\cos(\Phi_{rk}) - i \cdot \sin(\Phi_{rk})]\} \\
&= I_{nk} \cos(\Phi_{rk}) + Q_{nk} \sin(\Phi_{rk}) \\
&= x_{1,k} \cos(\Phi_k - \Phi_{rk}) - y_{1,k} \sin(\Phi_k - \Phi_{rk}) \\
Q_{1,nk} &= \mathrm{Im}\{(I_{nk} + i \cdot Q_{nk}) \cdot [\cos(\Phi_{rk}) - i \cdot \sin(\Phi_{rk})]\} \\
&= Q_{nk} \cos(\Phi_{rk}) - I_{nk} \sin(\Phi_{rk}) \\
&= x_{1,k} \sin(\Phi_k - \Phi_{rk}) + y_{1,k} \cos(\Phi_k - \Phi_{rk})
\end{aligned}
\tag{3:13}
$$

Note that the sampling frequency is only limited by the signal bandwidth. This means that the carrier frequency has no influences on the sampling criterion. The information bandwidth remains the unique condition. Therefore, complex sampling may lead to important power savings.

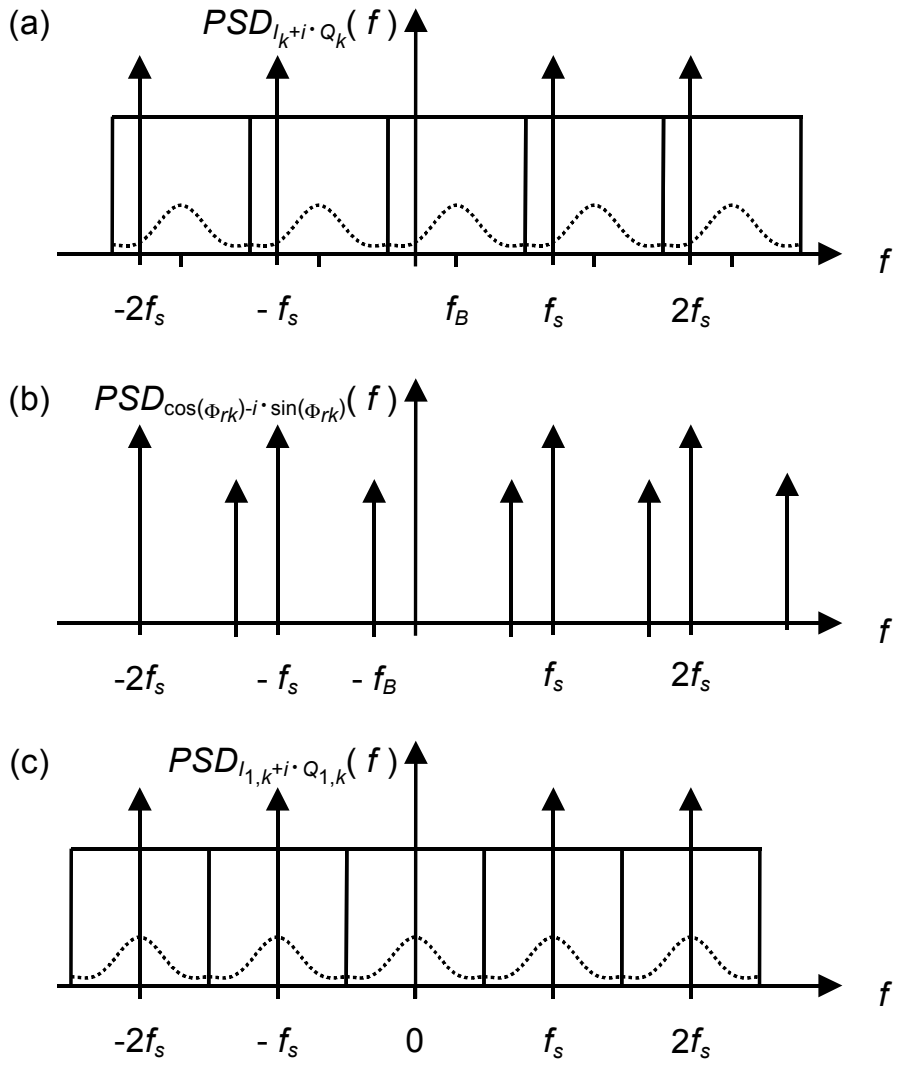


Figure 3.6: Doppler Removal in the frequency domain.

3.5.4 Complex Sampling Impact on Power Savings

The complex sampling impact on power savings can be illustrated with the GPS Builder 12 Channel GPS Development System. This conventional and well-known GPS receiver is manufactured by Gec Plessey and contains the GP2010 down-converter and the GP2021 correlator.

The GP2010 does not include a quarter-phase sampling stage like the Trimble down-converter. However, if the GP2010 were designed to furnish a pseudo-baseband signal, then it would be possible to realize the complex sampling algorithm. This would allow the sampling frequency reduction, from 40/7 MHz down to 40/16 MHz, without SNR

losses. Since the sampling frequency determines the correlator clock frequency, its reduction would represent an important consumption gain.

There are two ways to illustrate the impact of the complex sampling on power savings. The first approach is to considering the number of flip-flops implicated in the GP2021 DSP, where each channel contains the following components [Gec95]:

- *26-bit carrier NCO.*
- *25-bit code NCO.*
- *20-bit carrier cycle counter.*
- *C/A code generator.*
- *Four 16-bit accumulators.*
- *11-bit code phase counter.*
- *11-bit code slew counter.*
- *11-bit epoch counter.*

Considering the list of the components above, the number of flip-flop in one DSP channel can be estimated at 188. Since the GPS Builder contains 12 channels this number becomes 2256. The complex sampling will reduce the clock frequency of these flip-flops by half, which is not negligible.

The second manner to illustrate the complex sampling frequency effect on power savings is to consider the power consumption of the GP2021 correlator, which is typical 150 mW. Neglecting the peripheral functions, the complex sampling would reduce this consumption almost by half.

The example above illustrates that the complex sampling algorithm can represent a large potential on power savings. Therefore, a succinct aide memoir of the complex sampling algorithm is furnished at the end of this section.

> **For baseband or pseudo-baseband signals, the complex sampling frequency is only limited by the information bandwidth. This can represent an important impact on power savings. However, both the in-phase and quarter-phase components must be combined together to form an analytical signal. Therefore, at the input of the correlator, the demodulation process must be implemented with the single sideband modulator.**

The application of the complex sampling in GPS receivers can be considered as the first original contribution of this thesis.

3.6 Multiphase NCO

Figure 3.7 shows the functional block diagram of a standard NCO. The phase is accumulated upon a frequency increment and a lookup table converts the phase in a sine and a cosine.

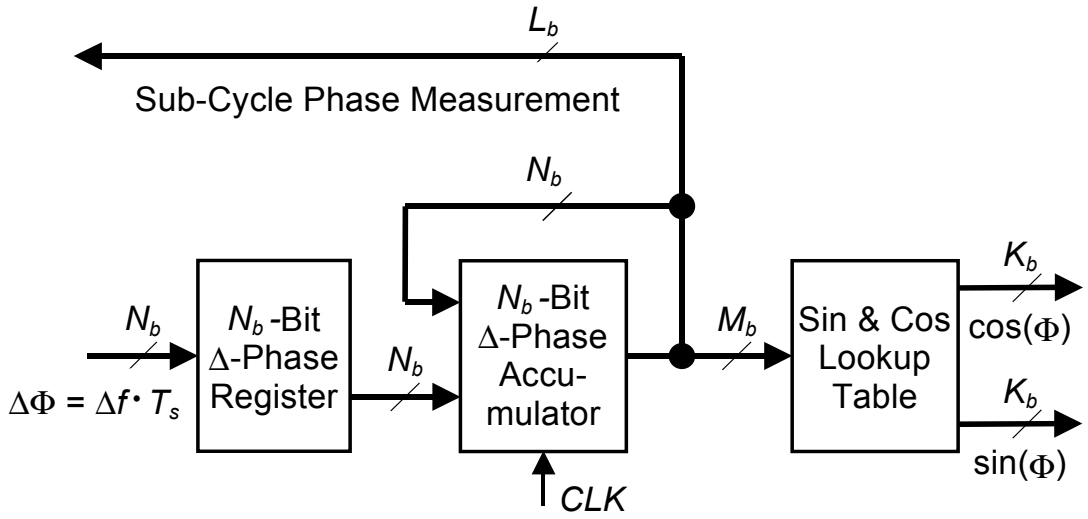


Figure 3.7: Functional block diagram of a NCO.

The illustrated N_b-bit NCO has the following characteristics:

- *Frequency output resolution $\Delta f_{min}=2^{-Nb} \cdot f_{CLK}$, where f_{CLK} is the clocking rate.*
- *The frequency range of the NCO $\Delta f_{max}=f_{CLK}/2$ is governed by the Shannon criterion.*

In GPS applications, long time constants and long coasting times are usual. Therefore, the NCO should be large enough, in order to have a high frequency resolution. However, phase accumulation in a large NCO consumes lot of energy, which is a drawback in low power environments.

For power savings, the phase accumulator can be split in two parts. Figure 3.8 illustrates the multiphase NCO, where each part is clocked at a different frequency rate. The MSBA is clocked with the main clock *CLK,* whereas the LSBA is clocked at a lower frequency rate with *CLK/N*, where *N* represents the clock division factor. At the beginning of a *CLK/N* clock cycle, the overflow of the LSBA is multiplied by *N* and added to the

MSBA. Therefore, The multiphase NCO has the same frequency characteristics as a conventional NCO. However, the phase accumulator splitting generates quantification noise in the tracking loop. In noisy environment like GPS, this additional phase noise can be set below other noise contributions. This will be demonstrated in the next sections.

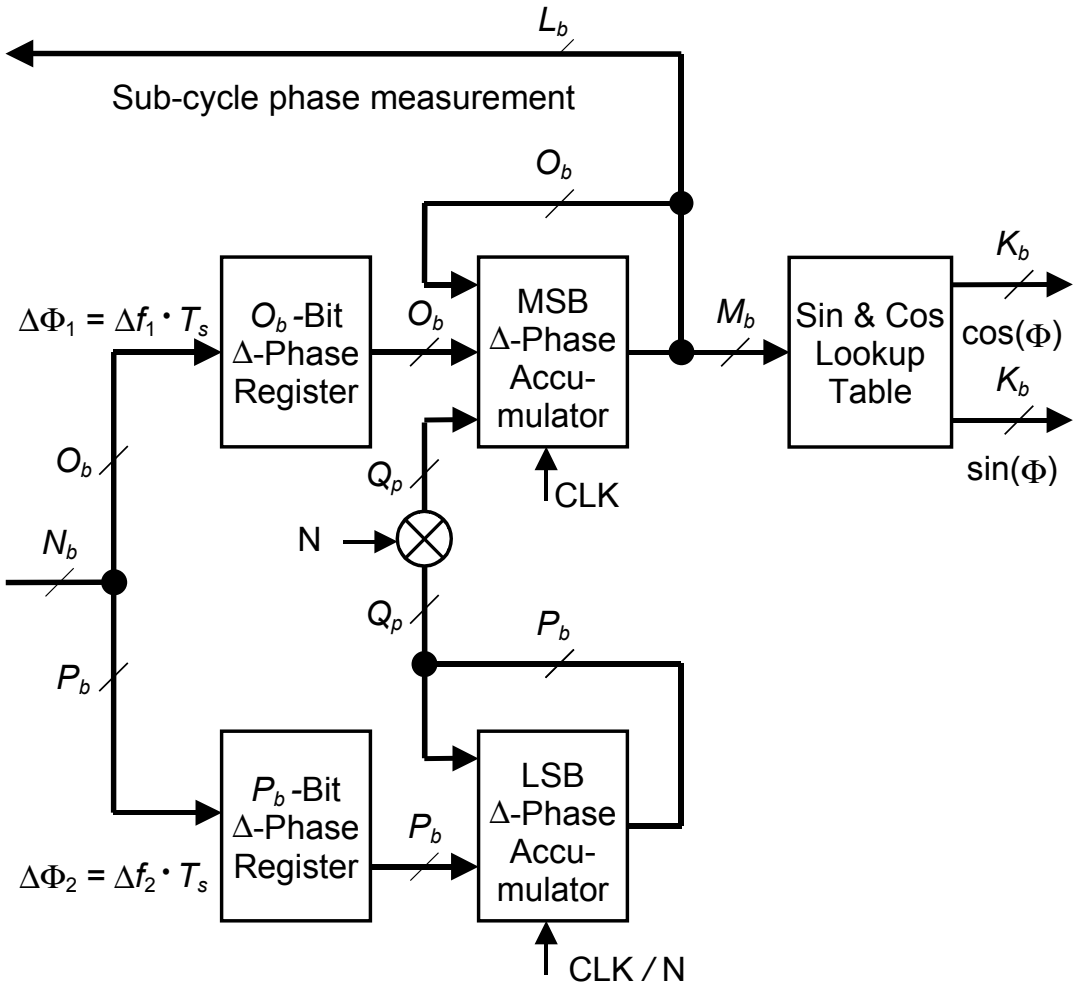


Figure 3.8: Multiphase NCO.

3.6.1 Quantification Noise

When the overflow of the LSBA is multiplied by N and added to the MSBA, the MSBA quantification error Q=0. At every *CLK* clock cycle, Q increases systematically by 2^{-Ob} amount, where O_b represents the size of the MSBA. After N iterations, Q reaches its maximum value of $Q=2^{-Ob} \cdot (N-1)$ and the overflow of the LSBA is shifted and added to the MSBA. At this moment, Q is zero again. Figure 3.9 illustrates the behavior of Q as a function of *CLK* for N=5.

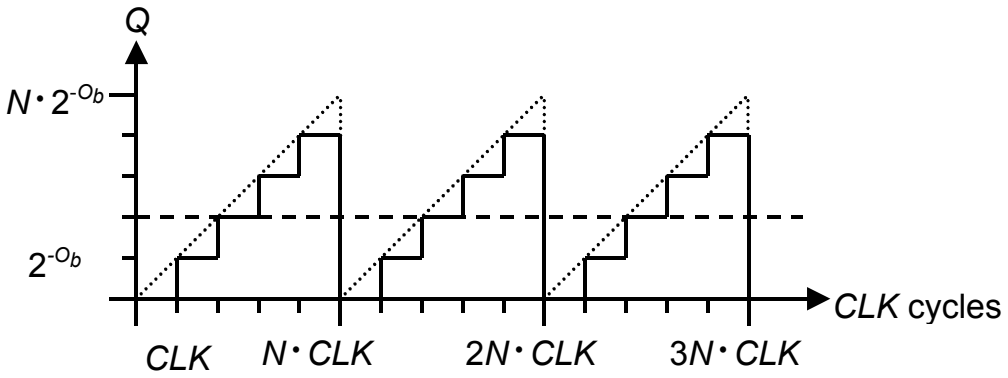


Figure 3.9: MSBA quantification in function of CLK.

The quantification type is two's complement truncation and its variance is given by:

$$\sigma_Q^2 = \frac{2}{N} \cdot \sum_{i=0}^{N/2} \left[\left(i \cdot 2^{-O_b} \right)^2 \right] = \frac{2 \cdot 2^{-2O_b}}{N} \cdot \sum_{i=0}^{N/2} \left(i^2 \right)$$
$$\approx \frac{2 \cdot 2^{-2O_b}}{N} \cdot \int_0^{N/2} i^2 \cdot di = \frac{\left(2^{-O_b} \cdot N \right)^2}{12} \qquad (3:14)$$

The frequency increments for both accumulators are obtained in the same way as the frequency increment for a standard NCO (Δf_{incr}):

$$\Delta f_{incr} = f_{sig} \cdot \frac{2^{(O_b + P_b)}}{f_{clk}} \qquad (3:15)$$

In this relationship, f_{sig} represents the frequency of the output signal.

3.6.2 Multiphase NCO Impact on Power Savings

The multiphase NCO impact on power savings can be illustrated again with the Gec Plessey GP2021 correlator. The components of each channel are listed in section 3.5.4. However depending on the application, all the elements are not used continually. Therefore, it is possible to redefine the heart of the correlator as containing the following components:

- *26-bit carrier NCO.*
- *25-bit code NCO.*
- *C/A code generator.*
- *Four 16-bit accumulators.*

Considering the list of the components above, the total number of continually used flip-flops per channel can be estimated to 135. If both NCOs were replaced with 12-bit MSBA multiphase NCOs and if the clock division factor N=16, then the power consumption would be reduced by 20%.

3.6.3 Splitting Error in the Tracking Loops

The splitting of the phase accumulator adds quantification noise. Therefore, additional phase noise will appear in the tracking loops. For example, this noise with the Costas PLL of Figure 3.10 yields:

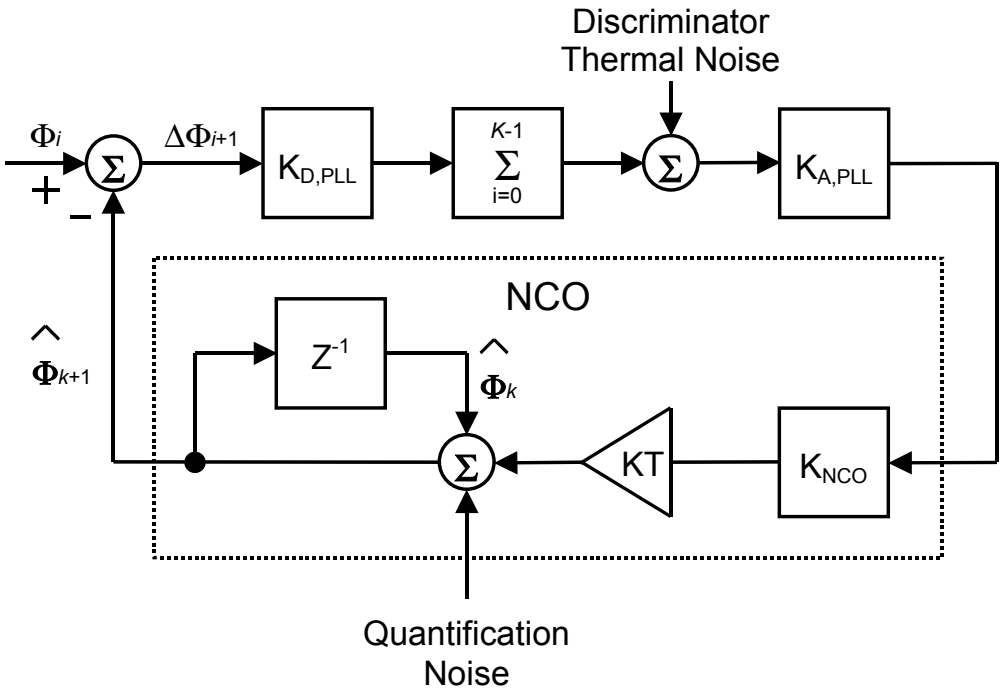


Figure 3.10: Generic linearized tracking loop.

There is an additional averaging operation at the output of the Costas PLL discriminator. In some tracking loops, this post detection filtering is introduced to reduce the noise bandwidth of the detector. Assuming Gaussian approximation and small phase tracking errors, the accumulated discriminator gain $K_{D,PLL,K}$ for the Costas PLL is given by:

$$K_{D,PLL,K} = \sum_{i=0}^{K-1} K_{D,PLL,i} = K \cdot K_{D,PLL} = 2\,K \cdot T \cdot \frac{C}{N_0} \cdot \sigma_n^2 \tag{3:16}$$

In the relationship above, σ_n represents the RMS noise power in one accumulator. Note that $K_{D,PLL}$ is determined in section 2.13.1. The discriminator gain varies with the CNR. Therefore, to maintain loop bandwidth, the loop gain must be compensated with the loop gain adjustment factor $K_{A,PLL}$. The PLL discriminator error at time i is independent of that at time $j \neq i$ and, still assuming Gaussian approximation, the variance of the accumulated discriminator thermal noise is given by the following relationship:

$$\sigma_{\delta\phi,K}^2 = \mathrm{var}\left(\sum_{i=0}^{K-1} K_{D,PLL,i}\right) = K \cdot \mathrm{var}\left(K_{D,PLL}\right) = K \cdot \sigma_{\delta\phi,i}^2 = 2K \cdot T \cdot \frac{C}{N_0} \cdot \left(1 + \frac{1}{2T \cdot C/N_0}\right) \cdot \sigma_n^4 \tag{3:17}$$

Assuming that $\Delta\Phi_i = \Delta\Phi_k$ over K samples, the closed loop noise time update error $\Delta\Phi_k$ equation is the following [Die95b]:

$$\Delta\Phi_{k+1} = (1 - 4B_{L,PLL} \cdot K \cdot T) \cdot \Delta\Phi_k + K_{A,PLL} \cdot K_{NCO} \cdot K \cdot T \cdot \sigma_{\delta\phi,K} + 2\pi \cdot \sigma_{Q,PLL} \tag{3:18}$$

In this equation, $\sigma_{\delta\phi,K}$, $\sigma_{Q,PLL}$ and $B_{L,PLL} = K_{D,PLL,K} \cdot K_{A,PLL} \cdot K_{NCO}/4$ represent respectively the accumulated discriminator thermal noise, the quantification noise and the single sided loop noise bandwidth. The variance time update equation is the expected value of the square of both sides of equation (3:18). Accounting for the fact that the noise in $\sigma_{\delta\Phi,K}$ and $\sigma_{Q,PLL}$ are independent on all previous errors $\Delta\Phi_k$, this variance becomes:

$$\sigma_{\Phi,k+1}^2 = (1 - 4B_{L,PLL} \cdot K \cdot T)^2 \cdot \sigma_{\Phi,k}^2 + \left(K_{A,PLL} \cdot K_{NCO} \cdot K \cdot T\right)^2 \cdot \sigma_{\delta\phi,K}^2 + \left(2\pi \cdot \sigma_{Q,PLL}\right)^2 \tag{3:19}$$

However, in steady state $\sigma_{t\,\&\,Q,PLL}{}^2 = \sigma_{\Phi,k+1}{}^2 = \sigma_{\Phi,k}{}^2$ and assuming that $1 \gg 2B_{L,PLL} \cdot K \cdot T$, the 1-sigma phase noise becomes:

$$\sigma_{t\,\&\,Q,PLL} = \sqrt{\frac{\left(K_{A,PLL}\cdot K_{NCO}\cdot K\cdot T\right)^2\cdot\sigma^2_{\delta\phi,K}+\left(2\pi\cdot\sigma_{Q,PLL}\right)^2}{8B_{L,PLL}\cdot K\cdot T\cdot\left(1-2B_{L,PLL}\cdot K\cdot T\right)}}$$
$$\approx \sqrt{\frac{B_{L,PLL}}{C/N_0}\cdot\left(1+\frac{1}{2T\cdot C/N_0}\right)+\frac{\left(2\pi\cdot\sigma_{Q,PLL}\right)^2}{8B_{L,PLL}\cdot K\cdot T}} \qquad (3:20)$$
$$\approx \sqrt{\sigma^2_{t,PLL}+\frac{\left(2\pi\cdot\sigma_{Q,PLL}\right)^2}{8B_{L,PLL}\cdot K\cdot T}}\ [\text{radian}]$$

This relationship above shows that the variance of the quantification noise in the Costas PLL is divided by twice the desired loop gain. Note that the post detection filtering has no direct effect on the tracking loop performance, because $\sigma_{t,PLL}$ has the same expression in both equations (2:38) and (3:20).

The post detection filtering reduces the variance of the normalized discriminator thermal noise by 1/*K*. However, this filtering increases also the loop update interval of *K.* Since the NCO integrates the smaller discriminator thermal noise longer, the post detection filtering has no direct effect on the tracking loop performances.

3.6.4 GPS Application

The determination of the quantification noise contribution in the DLL loop is simple. For that purpose, it is sufficient to replace the 1-sigma Costas PLL thermal noise in (3:20) by the 1-sigma DLL thermal noise and to convert the quantification noise units:

$$\sigma_{t\,\&\,Q,DLL} = \sqrt{\sigma^2_{t,DLL}+\frac{\sigma^2_{Q,DLL}}{8B_{L,DLL}\cdot K\cdot T}}\ \ [\text{chips}] \qquad (3:21)$$

The 1-sigma DLL thermal noise is defined in section 2.13.4.

The determination of the quantification noise contribution in the FLL assisted PLL loop is more complicated. However, the dynamic stress can be considered as mild most of the time. Therefore, when the CNR is above the PLL tracking threshold, the frequency discriminator will be superfluous. Consequently, the FLL-assisted-PLL can be considered as a pure PLL, where additional noise from the frequency discriminator is injected. Since, the thermal noise in the PLL is independent of that in the

FLL. The 1-sigma FLL-assisted-PLL thermal noise is obtained by superposition of both FLL and PLL thermal noise contributions, with the constraint that the frequency error must be converted into a phase error. Therefore, the 1-sigma FLL thermal noise is multiplied by $2\pi \cdot K \cdot T$, where $K \cdot T$ represents the loop updates period expressed in second.

$$\sigma_{t,FLL/PLL} = \sqrt{\left(2\pi \cdot K \cdot T \cdot \sigma_{t,FLL}\right)^2 + \sigma_{t,PLL}^2} \quad [\text{radian}] \qquad (3:22)$$

The 1-sigma FLL thermal noise is defined in section 2.13.2.

Now, the quantification noise in the FLL-assisted-PLL can be determined. Therefore, it is sufficient to replace the 1-sigma Costas PLL thermal noise in (3:20) by the 1-sigma FLL assisted PLL thermal noise as defined above.

$$\sigma_{t\,\&\,Q,FLL/PLL} = \sqrt{\sigma_{t,FLL/PLL}^2 + \frac{\left(2\pi \cdot \sigma_{Q,PLL}\right)^2}{8B_{L,PLL} \cdot K \cdot T}} \quad [\text{radian}] \qquad (3:23)$$

Table 3.2 summarizes the thermal and quantification noise variances in both loops, where the thermal noise variance has been computed with C/N_0=45 dB-Hz. The table show that in well-designed tracking loops the quantification noise can be set below the thermal noise.

Tracking Loop	FLL-assisted-PLL	DLL
Noise bandwidth	$B_{L,FLL}$=4 Hz $B_{L,FLL}$=18 Hz	$B_{L,DLL}$=1 Hz
Pre-detection interval	T=1 ms	T=1 ms
Post-detection averaging	K=1	K=16
NCO MSBA	12 bit	14 bit
NCO LSBA	12 bit	14 bit
Clock division factor	N=16	N=16
Thermal Noise variance	$1.10 \cdot 10^{-3}$ radian^2	$6.05 \cdot 10^{-6}$ chips^2
Quantification Noise variance	$3.49 \cdot 10^{-4}$ radian^2	$6.21 \cdot 10^{-7}$ chips^2

Table 3.2: Thermal & quantification noise contributions in the carrier and code tracking loops.

In both tracking loops of the Table 3.2, the MSBA and the LSBA have the same size. Therefore, the power consumption of each NCO is reduced almost by half. In standard correlators, the third of the total power consumption is attributable to the NCO. The use of multiphase NCO has thus an important impact on power savings. This is illustrated in section 3.6.2 with the Gec Plessey correlators.

The NCO splitting is the second original contribution of this thesis. Therefore, a succinct aide memoir is furnished at the end of this section:

> **The multiphase NCO is recommended for all applications where high frequency resolution and low power dissipation are required. However, the splitting of the phase accumulator adds quantification noise. In noisy environments like GPS and in well-designed tracking loops, this additional phase noise remains non significant in comparison with the other noise contributions.**

3.7 Summary & Conclusions of the Chapter

In the first part of this chapter, basic definitions that relate to the digital design process are presented, including some CAD tool notions. These definitions accommodate the reader with the fundamental digital design concepts. In the second part of this chapter, optimization methods that may contribute to reduce power consumption in CMOS circuits are presented. It appears, that the most effective methodology is the algorithmic optimization.

At the end of this chapter, two effective examples of algorithm optimization are presented. The first is the complex sampling algorithm, which reduces the sampling frequency to the information bandwidth, allowing for important power savings. The second example is a multiphase NCO with the same frequency characteristics than a standard NCO, but reduced power consumption. The additional quantification noise in the tracking loop remains non-significant in comparison with the thermal noise.

3.8 References

[Arm93] J. –R. Armstrong & F. –G. Gray, *Structured Logic Design with VHDL*, pp 1-20, Prentice-Hall, 1993.

[Cha97] P. Chambers, *The Ten Commandments of Excellent Design*, VHDL Code Examples, Electronic Design, April 14, 1997.

[Die95a] A. -J. Van Dierendonck, “GPS Receivers” *Global Positioning System: Theory and Applications*, Vol. 1, pp 356, American Institute of Aeronautics and Astronautics Inc., 1996.

[Die95b] A. -J. Van Dierendonck, “GPS Receivers” *Global Positioning System: Theory and Applications*, Vol. 1, pp 369-372, American Institute of Aeronautics and Astronautics Inc., 1996.

[Gec95] “GP2021 GPS 12 Channel Correlator With Microprocessor Support Functions”, *Advance Information*, Gec Plessey Semiconductors, June 1995.

[Tri96] E. –B. Rodal, G. –L. Wagner & C. –Y. Lau, Trimble Navigation Limited, *Ultra Low-Power Integrated Circuit For Pseudo-Baseband Down-Conversion of GPS RF Signals*, Patent Number 5’564’098, United States Patent, 1996.

[Wes93] N. –H. -E. Weste & K. Eshraghian, *Principles of CMOS VLSI Design*, pp 231-240, Addison-Wesley, 1993.

Chapter 4

Design of a Low Power DSP Architecture for GPS Receivers

This chapter draws the design of a low power DSP architecture for GPS receivers, which are dedicated to small-volume portable applications.

With strong constraints on the power consumption, significant dissipation gains are only conceivable with the size reduction of the microprocessor. Therefore, all real-time DSP algorithms have been implemented into dedicated bit-parallel hardware. Only a small microprocessor for the navigation processing is necessary.

The DSP architecture, called IMT3050M12, has 12 independent channels for the civilian L_1 band. It performs satellite acquisition and tracking, without the help of an external microprocessor. When a channel is locked onto a satellite, the IMT3050M12 synchronizes the GPS signal and transmits the GPS message to the microprocessor. In addition, the circuit records pseudo-ranges on request.

In the first part of the chapter, the specifications for the DSP architecture are introduced. Then, the SV signal acquisition and tracking algorithms are completely defined. These algorithms set the most important performance characteristics for the GPS receiver. In the second part of the chapter, the processor target architecture is presented, followed by a detailed description of each element.

4.1 Specifications

The scope of this thesis is the design and the realization of a low power DSP architecture for GPS receivers, dedicated to miniature and portable applications, like time synchronization or position determination in a wristwatch.

The volume constraint results in a reduced antenna size, which can generate SNR losses as large as 4.5 dB. Therefore, special care has to be taken with respect to the receiver sensitivity.

The planned battery is a lithium-ion type, which supplies 3.6 Volts and has a 350-mAh capacity. The autonomy of the GPS receiver was targeted to cover its use during a weekend. With a projected power consumption of 30 mW for the radio front-end, the baseband processing should be in the same order of magnitude [Far2000].

4.1.1 Signal Sensitivity

As mentioned before, the small antenna volume results in large SNR losses and therefore DSP algorithms with robust acquisition and tracking characteristics are required. Generally, the price of these algorithms is complexity and high power dissipation. Thus, much effort has been spend to understand the existing DSP algorithms.

Beside the robustness, the signal acquisition speed has also been investigated. The aim was time-to-first-fix minimization, in order to limit the power consumption.

4.1.2 Power Consumption

The most important constraint for the receiver is the power consumption, because the final application is portable. Unfortunately, the technology has not been a parameter in this optimization process and other directions had to be investigated.

An interesting alternative is the size reduction of the microprocessor, because it has a significant impact on power consumption with respect to the other units (e.g. LNA, RF, IF and correlator). The reason for that is the real time SSP, which requires important computational capacity like 32-bit Reduced Instruction Set Computers (RISC), as shown in Table 4.1.

Table 4.1 shows that half of the total power consumption is attributable to the Computer Programmable Unit (CPU) core. In most of the cases it exceeds 200 mW, which is more than the specified power consumption of the complete receiver (e.g. 100 mW). Therefore, a new architecture is proposed where some (or even all) SSP is implemented into dedicated hardware, which introduces the key point of this thesis:

> **"A GPS receiver architecture is proposed, where all DSP algorithms are implemented into dedicated hardware".**

The statement above can be considered as a novelty in the GPS domain because no other implementation uses this paradigm. In addition, only a small microprocessor for the navigation processing is required because this is not a real time process. Therefore, the designed GPS baseband receiver can be considered as a new combination of a **"GPS message receiver"** and a small microprocessor for the navigation processing.

Company	Chipset	CPU	CPU Size	Total Power Dissipation	CPU Core Dissipation
Gec Plessey	GPS Architect	P60ARM-B	32-bit RISC		150 mW @ 20 MHz [Mit99]
Motorola	Oncore	Motorola MC68331	32-bit RISC	900 mW	520 mW @ 19.1 MHz [Mot96]
Philips	SC1575 DN1575	Philips XA-G3	16-bit RISC	650 mW	300 mW @ 30 MHz [Phi99]
µ-Blox	GPSMS1	Hitachi SH-7020	32-bit RISC	500 mW	215 mW @ 12.5 MHz [Hit97]

Table 4.1: The commercial RISC microprocessors used in GPS applications and their power consumption.

4.2 DSP Architecture Description

The DSP architecture, called IMT3050M12, is a one chip 12-channel GPS message receiver with low power architecture. The circuit performs satellite acquisition and tracking, without the help of an external microprocessor. When a channel is locked onto a satellite, the circuit synchronizes the GPS data stream that is output for further processing. In addition, the IMT3050M12 records pseudo-ranges on request.

After loading the PRN code number and setting up the frequency parameters, the channel can be started to search for a satellite. It scans all possible chip phases in a single frequency bin with a ½ chip resolution. When the satellite has been found, the circuit locks onto the signal while constantly adapting the internally generated carrier and code frequencies. As long as the signal strength is sufficient, the channel performs bit synchronization and outputs the GPS message. Only frame synchronization is to be implemented in software.

Pseudo-range recording is supported through an external signal. When asserted, this signal latches the state of the C/A code generator for each channel simultaneously, as well as the current phase of the code NCO. This allows recovering of sub-microsecond information.

Power consumption is reduced in active coasting mode. In this configuration the code generation process continuous whereas the carrier generation and the accumulation process are stopped. The internal state may still be accessed through the microprocessor interface. Reactivating the normal mode allows resuming in track mode with the current NCO settings.

Maximum power reduction may be obtained by inhibiting the clock. In this stand-by mode, the CPU may still access the registers.

4.3 Signal Acquisition

A short synchronization time has a great influence on the power consumption, because long synchronization times consume more energy than short ones. Therefore, a modified MDD scheme has been realized for the signal acquisition in the DSP architecture.

4.3.1 Modified Multiple-Dwell Detection

Figure 4.1 shows the block diagram of the modified MDD. Compared with the standard MDD, there are essentially two mayor differences:

- *In the modified MDD, the pre-detection integration times T remain constant for all evaluations, whereas for the standard MDD, the pre-detection intervals T are variable.*
- *The energy is accumulated over the full detection time thereby reducing detection variance.*

At the end of the first detection, a threshold *TH* is subtracted to the output of the energy detector V_1. If the resulting V_1-*TH*<0, a “miss” (incorrect phase) is declared and the phase cell is rejected. If the resulting V_1-*TH*>0, a “hit” (potential correct cell) is declared and the second detection of the same cell can begin.

At the end of the second detection, a threshold *TH* is subtracted again from the output of the energy detector V_2 and the new difference is added to the preceding difference. If the resulting V_1+V_2-2*TH*<0, a “miss” (incorrect phase) is declared and the phase cell is rejected. If the resulting V_1+V_2-2*TH*>0, a “hit” (potential correct cell) is declared and the third detection of the same cell can begin.

This confirmation process is repeated for *N* cycles, until the signal is declared present and the tracking process is started. In a standard MDD the previous outputs of the energy detector are not used during the confirmation process.

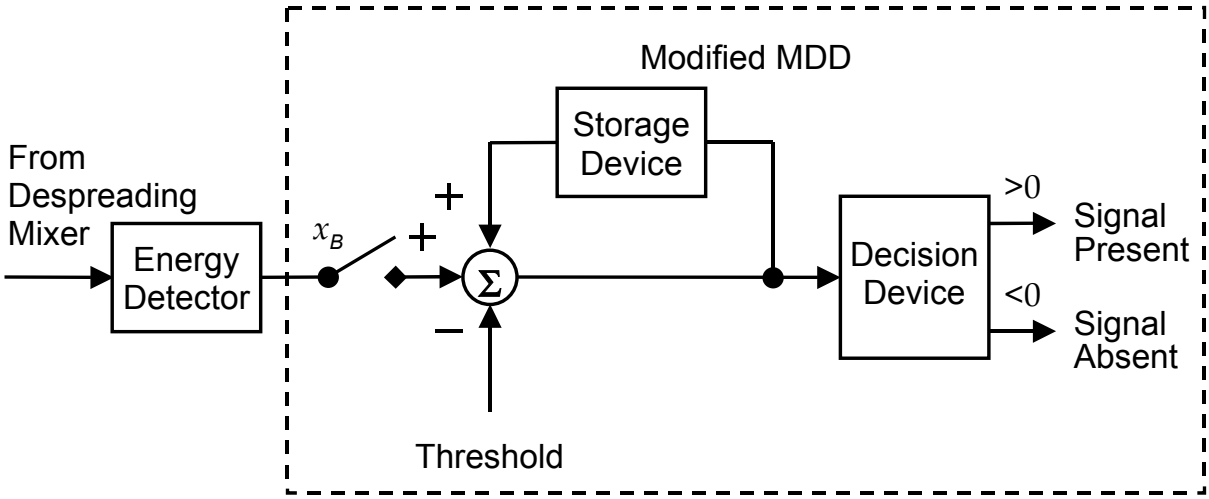


Figure 4.1: Modified MDD.

4.3.2 Advantages of the Modified MDD

The modified MDD has two advantages over the standard one, which follows from their differences.

The first advantage is the complexity reduction of the algorithm. In the modified MDD, the pre-detection intervals *T* are limited to 1 millisecond and remain constant for all evaluations. Therefore, data bit transitions do not influence the energy detector. For the standard MDD, the pre-detection integration times *T* are variable. Generally, *T* is short at the beginning and becomes larger during the confirmation process. The realization of a variable integration time *T* is complex and expensive, especially when the pre-detection interval *T* exceeds 1 millisecond. Then the data bit transitions influence the energy detector, which should be insensitive to them. The moderate complexity of the modified MDD reduces the implementation cost, which has an important impact on the power consumption.

The second advantage of the modified MDD appears during the confirmation process, because the energy-samples are accumulated, narrowing the post-detection bandwidth. The modified MDD is therefore more accurate than the standard one. Figure 4.2 illustrates the PDF of the accumulated energy during the acquisition process, with *K* representing the number of confirmation iterations. The dashed lines correspond to the PDF of the noise power (e.g. when no signal is present), whereas the plain lines represent the PDF of the signal plus noise power (e.g. when the SV signal is present). The SNR is set to 6 dB, which corresponds to a standard value for a pre-detection interval of 1 millisecond. At the input of the MDD, the accumulated signals are assumed to have a Gaussian distribution. Therefore, the general PDF of the signal plus noise power at the output of the modified MDD is given by the following relationship [Die95c]:

$$P_{S,K}(y) = \frac{1}{2} \cdot \left(\frac{y}{2K \cdot \beta} \right)^{\frac{K-1}{2}} \cdot \exp\left\{ -\frac{1}{2}(y + 2K \cdot \beta) \right\} \cdot I_{K-1}\left(\sqrt{2K \cdot \beta \cdot y} \right) \tag{4:1}$$

In this relationship, $\beta = (C/N_0) \cdot T$ and $I_{K\text{-}1}(x)$ represent respectively the SNR and the modified Bessel function of the first kind. Note also that in the relationship above the Mean Noise Power (MNP) of the predetection

accumulator has been normalized (e.g. σ_n^2=1). Likewise, the general PDF of the normalized noise power at the output of the MDD is given by:

$$P_{N,K}(y) = \frac{1}{2^K (K-1)!} \cdot y^{K-1} \cdot \exp\left(-\frac{y}{2}\right) \qquad (4:2)$$

For K=1 and through the change of variable $y=z^2$, the PDF of the signal plus noise power is a Rician distribution. Likewise, the PDF of the noise power is a Rayleigh distribution (section 2.14.1). The relationships above are complex and the method for their determination is not in the scope of this thesis. However, the curves presented in Figure 4.2 are meaningful. When the number of confirmations K increases, the presence of the SV can be determined with more accuracy. Since the variances of the PDFs are inversely proportional to K.

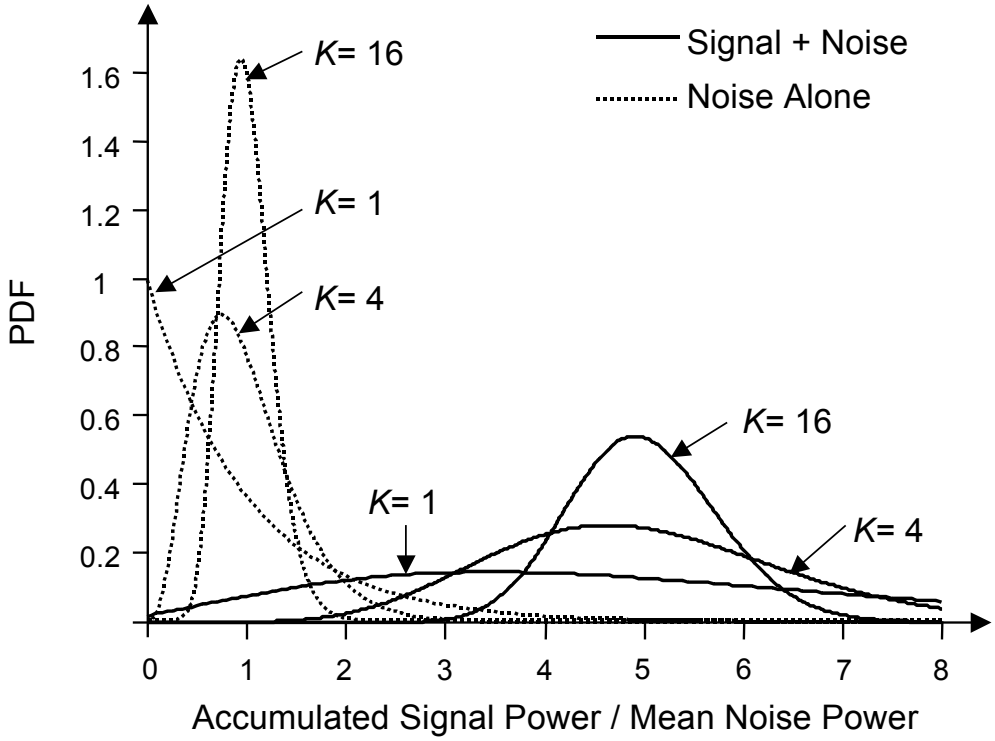


Figure 4.2: PDF of the energy in the presence and absence of the SV signal.

The modified MDD takes benefit of the decreasing PDF variances whereas the standard implementation doesn't have this benefit. At the beginning, when K=1, the respective PDF are almost indistinguishable and the determination of the threshold *TH* is quite difficult. In this case, a recommended value for the threshold would be the MNP of the in-phase

plus quarter phase predetection accumulators (e.g. $TH=2\,\sigma_n^2$). During the confirmation process, when K increases, the respective PDFs can be distinguished more easily and it is recommended to fix a higher threshold.

In the implemented DSP architecture, the threshold has been fixed to twice the MNP. In addition, the energy detector is initialized systematically to MNP for the first evaluation (e.g. K=1). The apparent threshold corresponds therefore to the MNP for the first evaluation. When a false alarm has occurred on the first evaluation, the apparent threshold will correspond to 1.5 times the MNP for the second evaluation (e.g. K=2). The third confirmation (e.g. K=3) will have a 1.66 times apparent threshold and so on. This detection scheme is very simple and has an excellent acquisition performance.

4.4 Signal Tracking Algorithm

In an unaided (stand-alone) GPS receiver, a well-designed DLL will track at considerably lower levels of CNR than the carrier-tracking loop. Since, both the code and carrier tracking loops must successfully track their respective signals, it is sufficient to analyze the weaker tracking threshold only.

4.4.1 Carrier Tracking Loop

Figure 4.3 illustrates the carrier-tracking loop block diagram of the DSP architecture. The design of this loop concerns the pre-detection integrators, the loop discriminators and the loop filter. These three functions determine the thermal noise error and the maximum line-of-sight stress threshold. All functions are realized by dedicated hardware.

To minimize the number of correlators in each channel, the punctual (or prompt) complex correlator has been eliminated. Instead, the pseudo prompt values are obtained from the average values of the early and late correlators. The early/late correlator spacing is set to half a chip. Therefore, the pseudo prompt channel suffers from a 2.5 dB SNR loss.

During initial loop closure tracking conditions, the first carrier-tracking mode is a first order FLL, to remove the potential large frequency uncertainty due to the crude Doppler search process. After initial loop closure in FLL, if the phase-lock detector indicates that PLL can be

supported, the carrier tracking loop transitions into a first order FLL assisting a second order PLL. As presented in section 2.10.3, this innovative tracking loop combines both: the dynamics and robustness of the FLL plus the accuracy performance of the PLL. Both tracking loops are insensitive to velocity stresses, but are sensitive to acceleration stresses.

A sensitive and accurate phase lock detector is necessary for a fail-safe transition form FLL only to FLL-assisted-PLL mode. This new sensitive and accurate phase lock detector (called BitSync) is introduced at section 4.10. Note that the FLL only mode is also used as fallback mode when phase lock is lost.

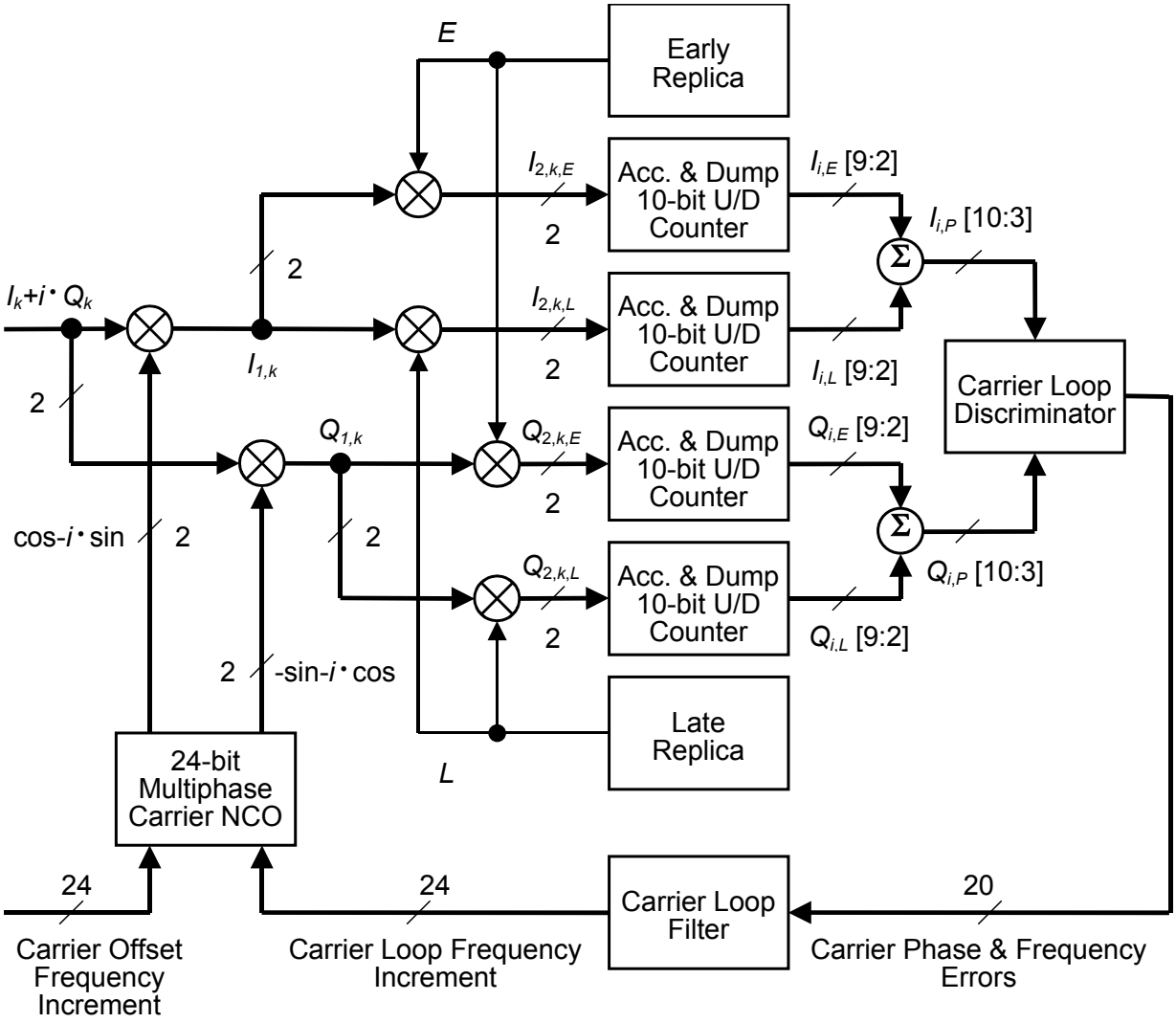


Figure 4.3: Carrier tracking loop block diagram.

Discriminator

During the initial FLL mode, the cross product discriminator detects the potentially large frequency errors. In the FLL-assisted-PLL mode, the FLL and PLL discriminators are respectively a decision-directed cross product and an $I \cdot Q$.

The cross product discriminator is one of the simplest frequency discriminators. It is near optimal at low SNR and has the lowest computational complexity, which has a positive influence on power consumption. This discriminator is obtained with the following relationship:

$$cross = \frac{I_{P,i-1} \cdot Q_{P,i} - I_{P,i} \cdot Q_{P,i-1}}{2\pi \cdot T} \tag{4:3}$$

Figure 4.4 illustrates the output frequency error of the cross product discriminator (solid lines). The error is proportional to $\sin(2\pi \cdot \Delta f_k \cdot T)$ attenuated by the sinc^2 function and has more than one zero point as a function of Δf_k, some with a positive slope and some with a negative slope. However, the cross-product discriminator is affected by the data-bits. So it must be computed within a data-bit period. Otherwise, if the bit-sign changes, the discriminator results in the wrong sign. Unfortunately, during initial loop closure tracking conditions, the data-bit transitions are unknown. Therefore, some additional noise quantities are injected in the carrier tracking loops.

The decision-directed cross product solves that problem by modulating the cross product with the sign of the dot product:

$$dot = I_{P,i-1} \cdot I_{P,i} + Q_{P,i-1} \cdot Q_{P,i} \tag{4:4}$$

The inconvenience of this type of discriminator is its discontinuity, narrowing its effective frequency range by two. Figure 4.4 illustrates the output frequency error of the decision-directed cross product discriminator (dashed lines).

In the retained solution, both discriminator types are combined. At the beginning the frequency uncertainty due to the Doppler search process is large, so a discriminator with a large pull in frequency range is necessary. Next, the frequency uncertainty becomes smaller and the data demodulation can begin. Here, a less noisy discriminator is required.

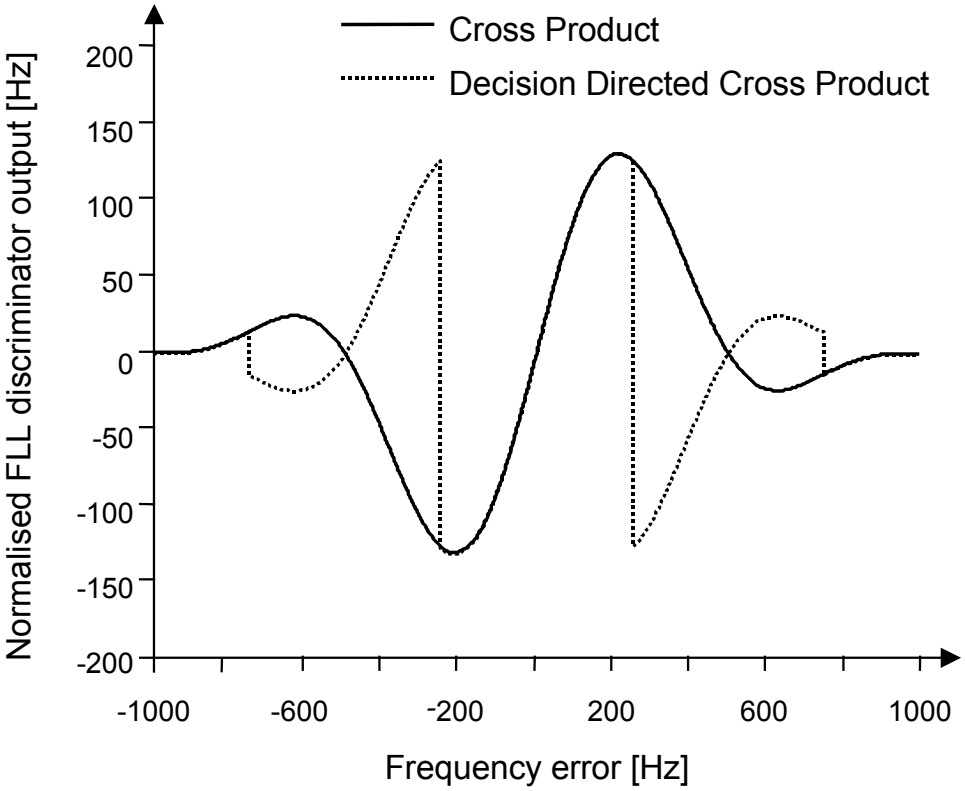


Figure 4.4: Decision-directed cross product and cross product discriminators.

Threshold

The GPS measurement errors and tracking thresholds are closely related, because the receiver loses lock when the measurement errors exceed a certain boundary. The conservative tracking threshold is defined as follows: the 3-sigma errors from all sources combined with the dynamic stress errors should not exceed the monotone fraction of the discriminator. Ignoring the oscillator noise, this relationship has been determined at section 2.13.3:

$$3\sigma_{FLL/PLL} = 3\sqrt{\sigma_{t,FLL}^2 + \left(\frac{\sigma_{t,PLL}}{2\pi \cdot T}\right)^2} + \frac{\partial^{n+1}R/\partial t^{n+1}}{2\pi \cdot \omega_{L,FLL}^n} \leq \frac{1}{4T} \qquad (4:5)$$

Section 2.13.3 demonstrates that the analyses using rule of thumb thresholds are quite accurate predictions. Table 4.2 presents the fixed loop noise bandwidth of the DSP architecture and the corresponding rule of thumb carrier tracking thresholds. Both noise bandwidths are the same than those proposed by P. Ward. Under both high and low

dynamic stresses the FLL outperforms the PLL. Considering the design insight of P. Ward, the FLL-assisted-PLL should have better performance than a pure PLL. The FLL assisted PLL rule of thumb analyses confirm this trends.

Parameter	Units	FLL	FLL/PLL	PLL
Filter Order	-	1	1/2	2
Noise Bandwidth	Hz	4	4/18	18
Pre-Detection interval *T*	ms	1	1/1	1
Analysis CNR Threshold – 100 m/s^2	dB-Hz	26.8	27.7	Lost
Analysis CNR Threshold – 10 m/s^2	dB-Hz	26.1	27.0	29.9
Analysis CNR Threshold – 1 m/s^2	dB-Hz	26.1	26.7	27.3

Table 4.2: Rule of thumb analysis tracking threshold for predefined acceleration.

Loop Parameters

Figure 4.5 illustrates the detailed block diagram of the FLL-assisted-PLL discriminators and filter. An 8-bit multiplier and a 20-bit accumulator are necessary for the implementation of these operations. In addition, there is a shift operator at the input of the accumulator to fix the desired loop gain.

For small frequency errors and 13 dB SNR in the early and late complex correlators, the gain of the cross product discriminator $K_{D,FLL}$ is given by the following relationship [Die95a]:

$$K_{D,FLL} = 4\pi \cdot \frac{C}{N_0} \cdot T^2 \cdot \sigma_{n,8MSB}^2 = 34.26 \tag{4:6}$$

In the relationship above, $\sigma_{n,8MSB}$ represents the RMS noise power of the pre-detection accumulators, which value is determined in section 4.6.6. As a compromise between accurate carrier tracking and implementation efficiency, the size of the NCO has been set to 24 bits. Considering the clock frequency of 4.3625 MHz, the resolution of the NCO becomes:

$$\Delta f_{\min} = 2^{-N_b} \cdot f_{clk} = 260 \quad [\text{mHz}] \tag{4:7}$$

Considering the discriminator gain, the NCO resolution and a 13 dB SNR, the frequency noise bandwidth of the FLL becomes as follows:

$$\begin{aligned} \omega_{L,FLL} \cdot T &= K_{D,FLL} \cdot K_{A,FLL} \cdot \Delta f_{\min} \cdot T \\ \frac{B_{L,FLL} \cdot T}{0.25} &= \frac{K_{D,FLL} \cdot \Delta f_{\min}}{2^9} \\ B_{L,FLL} &= \frac{0.25 \cdot K_{D,FLL} \cdot \Delta f_{\min}}{2^9 \cdot T} \\ &= 4.35 \quad [\text{Hz}] \end{aligned} \tag{4:8}$$

This value is close to the expected value of 4 Hz. In the relationship above, the loop update interval T represents the unit delay Z^{-1} in the digital integrator of the loop filter (Figure 2.23). Note also that the $1/2^9$ factor corresponds to the loop adjustment gain multiplied by the loop update interval (e.g. $K_{A,FLL} \cdot T$), which is realized with the 5-bit and 4-bit shift-to-the right operations.

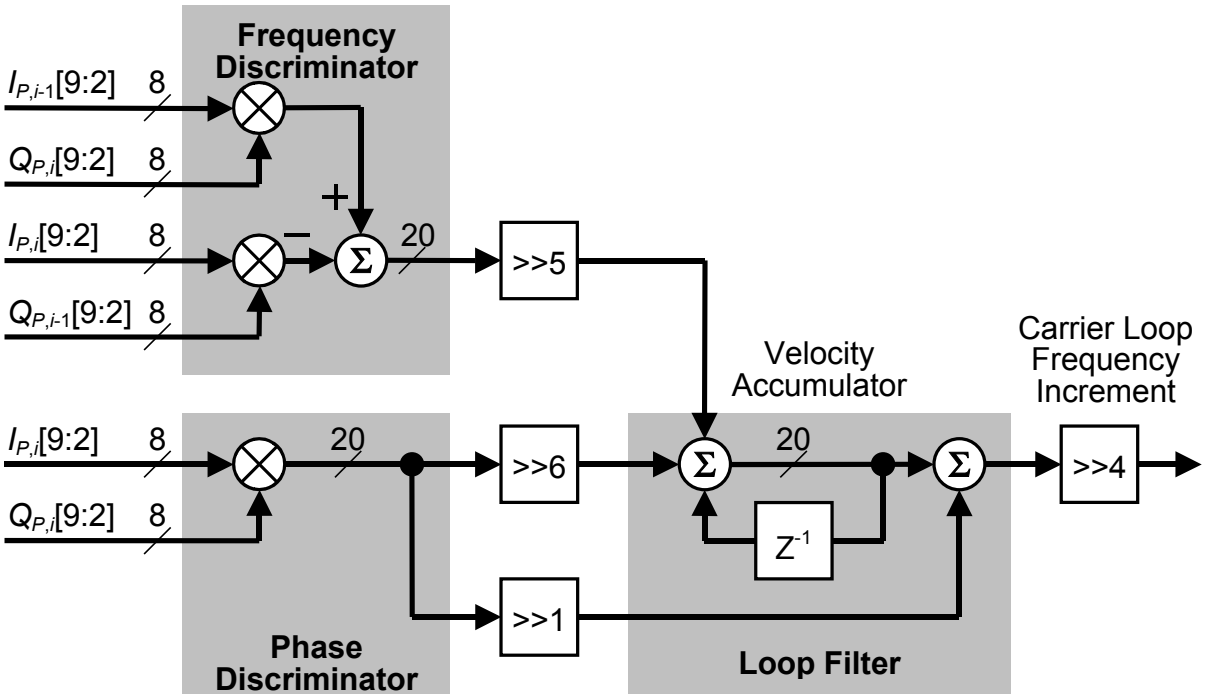


Figure 4.5: Block diagram of the FLL assisted PLL discriminators and filter.

The PLL noise bandwidth can be determined in the same way with the $I \cdot Q$ discriminator gain obtained at section 2.13.1, which is given by:

$$\begin{aligned} K_{D,PLL} &= 2 \cdot \frac{C}{N_0} \cdot T \cdot \sigma^2_{n,8MSB} \\ &= 5453 \end{aligned} \tag{4:9}$$

Taking the discriminator gain, the unchanged NCO resolution and a 13 dB SNR, the PLL noise bandwidth in the direct branch becomes as follows:

$$\begin{aligned} a_2 \cdot \omega_{L,PLL} &= K_{D,PLL} \cdot K_{A,PLL,D} \cdot \Delta f_{\min} \\ \frac{a_2 \cdot B_{L,PLL}}{0.53} &= K_{D,PLL} \cdot K_{A,PLL,D} \cdot \Delta f_{\min} \\ B_{L,PLL} &= \frac{0.53 \cdot K_{D,PLL} \cdot \Delta f_{\min}}{2^5 \cdot a_2} \\ &= 16.6 \quad [\text{Hz}] \end{aligned} \tag{4:10}$$

Again, the $1/2^5$ factor corresponds to the loop adjustment gain, realized with the 1-bit and 4-bit shift-to-the right operations.

The PLL noise bandwidth in the integrated branch is defined as follows:

$$\begin{aligned} \omega^2_{L,PLL} \cdot T &= K_{D,PLL} \cdot K_{A,PLL,I} \cdot \Delta f_{\min} \cdot T \\ \left(\frac{B_{L,PLL}}{0.53} \right)^2 \cdot T &= \frac{K_{D,PLL} \cdot \Delta f_{\min}}{2^{10}} \\ B_{L,PLL} &= 0.53 \cdot \sqrt{\frac{K_{D,PLL} \cdot \Delta f_{\min}}{2^{10} \cdot T}} \\ &= 19.7 \quad [\text{Hz}] \end{aligned} \tag{4:11}$$

Note that the noise bandwidths in the integrated branch and in the direct branch are not the exactly the same, but this will not have a significant impact on the loop performance. Both bandwidths correspond approximately to the expected value of 18 Hz. The $1/2^{10}$ factor corresponds to the loop adjustment gain multiplied by the loop update period (e.g. $K_{A,PLL,I} \cdot T$) and is realized with the 6-bit and a 4-bit shift-to-the right operations.

All carrier loop noise bandwidths depend on the SNR. Therefore, the discriminator-shift controls $B_{L,FLL}$ and $B_{L,PLL}$ as a function of this ratio. If the SNR exceeds 14.5 dB, an additional 2-bit shift-to-the-right is applied at the output of both discriminators.

Another important design parameter is the saturation of the accumulator. In the block diagram of Figure 4.5 the velocity accumulator will saturate at the following frequencies:

$$Vel_{Sat} = \pm \frac{2^{(20-1)}}{2^4} \cdot \Delta f_{\min} = \pm 8.54 \quad [\text{kHz}] \tag{4:12}$$

These frequencies will not be attained with a GPS application, because the Doppler frequency varies from + 4.5 kHz to – 4.5 kHz.

4.4.2 Code Tracking Loop

Figure 4.6 illustrates the PRN code tracking loop block diagram. The design of this loop concerns the pre-detection integrators, the loop discriminators and the loop filter. These three functions determine the thermal noise error and the maximum line-of-sight stress threshold. All functions are realized by dedicated hardware.

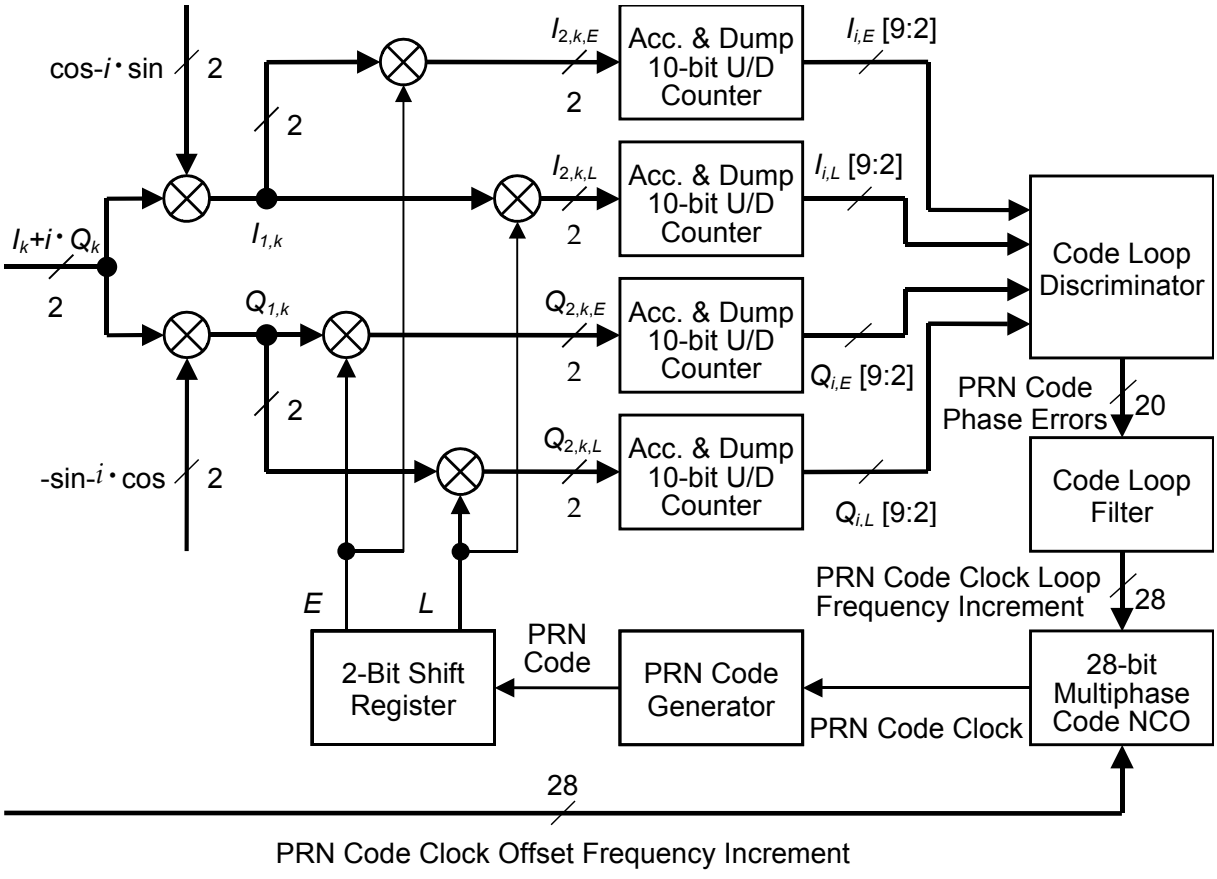


Figure 4.6: PRN code tracking loop block diagram.

Discriminator Type

The non-coherent early-minus-late power DLL discriminator has been chosen for the DSP architecture, which is justified by three arguments. The first is the robustness of the discriminator, which is useful for small SNR. The second argument is the moderate computational load of the discriminator. Finally, the discriminator produces good tracking errors within ±½ chip of input error. The discriminator is obtained with the following relationship [Die95a]:

$$E(\delta\tau_i) = I_{E,i}^2 + Q_{E,i}^2 - \left(I_{L,i}^2 + Q_{L,i}^2\right) \qquad (4:13)$$

The carrier-aided DLL permits the use of a very narrow noise bandwidth to produce very precise pseudo-range measurements, because the DLL does not need to track the full dynamics. For this DLL, the standard filter is of the first order and has a noise bandwidth of 1 Hz (Figure 4.7).

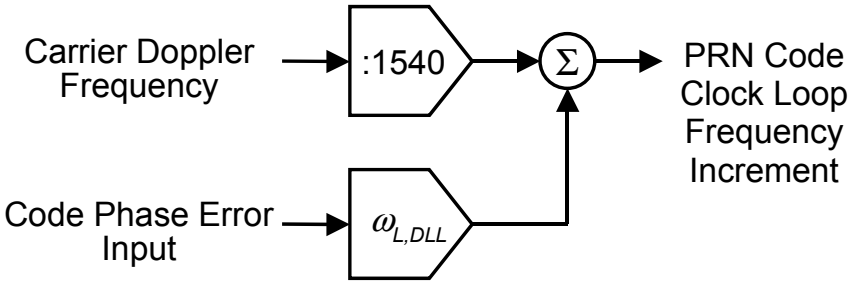


Figure 4.7: Carrier assisted first order DLL filter.

The code frequency discriminator is obtained from the carrier Doppler frequency. As the ratio between the carrier and the code frequencies is exactly 1540, the carrier Doppler frequency is divided by this ratio. The implementation of a divider into dedicated hardware is complex and expensive. However, the division operation can be accomplished in a very simple way.

Successive accumulation of the pre-shifted carrier Doppler frequency furnishes a good approximation. For example, if the carrier Doppler frequency is pre-shifted to the right by 11-bit, 13-bit, 15-bit and 17-bit, the result is a division by 1542 instead of 1540. The extra hardware for the shift operations at the input of the accumulator is far less than the supplementary cost of a divider implementation. The division by 1542

presented here is an excellent compromise between processing precision and implementation cost.

In the implemented DSP architecture, there is an additional averaging operation at the output of the code discriminator, which increases the robustness and the precision of the DLL. Figure 4.8 illustrates the code discriminator accumulation during 16 cycles. Next, the accumulated value is shifted to the right by 4 bits and introduced into the loop filter. Section 3.6.3 demonstrates that the discriminator averaging operations have no direct effects on the tracking loop performances. However, the loop-update period increases to K=16 epochs, which corresponds to the period of the signal presence confirmation. Both processes are synchronized during the SV signal tracking. Therefore, the DLL is only updated after the SV signal presence confirmations. The advantage of this method appears during the coasting, when the SV signal disappears for a little while and reappears again. Then, it is important to confirm the presence of the SV signal before loop updates. Otherwise, important noise quantities are injected in the DLL.

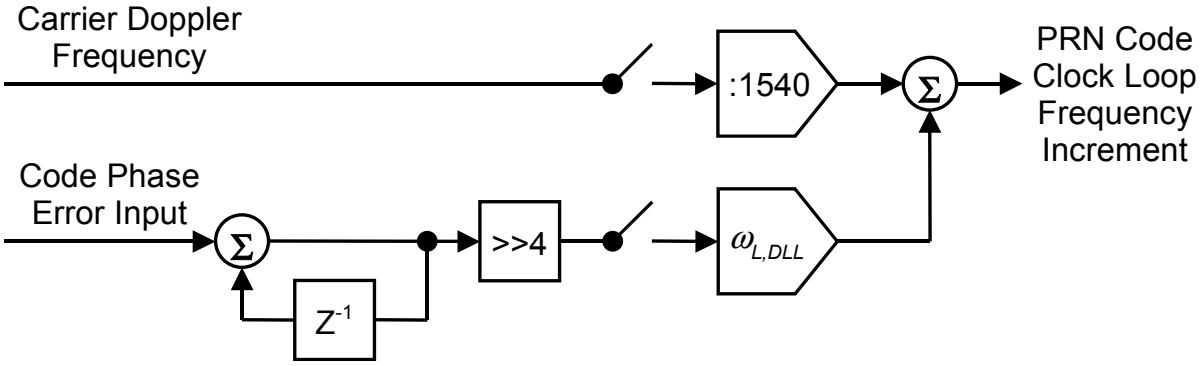


Figure 4.8: Code discriminator averaging.

Loop Parameters

Figure 4.9 illustrates the block diagram of the carrier aided DLL discriminators and filter. An 8-bit multiplier and a 20-bit accumulator are necessary for the implementation of these operations. In addition, the accumulator should have a shifting operator at the input, to fix the desired loop gain.

For small frequency errors and 13 dB SNR in both early and late complex correlators, the gain of the cross product discriminator $K_{D,DLL,K}$ is given by the following relationship [Die95a]:

$$\begin{aligned} K_{D,DLL,K} &= \sum_{i=0}^{K} K_{D,DLL,i} \\ &= 4K \cdot \frac{C}{N_0} \cdot T \cdot \sigma_{n,8MSB}^2 \cdot (2-d) \\ &= 2.62 \cdot 10^5 \end{aligned} \tag{4:14}$$

In this relationship *d* represents the spacing between the early and late correlator expressed in chips. The resolution of the code NCO has to be set higher than the carrier NCO, because it is important to maintain the code phase precisely for accurate phase measurements and good coasting performance. It is selected to 28 bits:

$$\Delta f_{\min} = 2^{-N_b} \cdot f_{clk} = 16{,}3 \quad [\text{mHz}] \tag{4:15}$$

Considering the discriminator gain, the NCO resolution and a 13 dB SNR, the frequency noise bandwidth of the DLL becomes as follows:

$$\begin{aligned} \omega_{L,DLL} &= K_{D,DLL,K} \cdot K_{A,PLL} \cdot \Delta f_{\min} \\ \frac{B_{L,DLL}}{0.25} &= \frac{K_{D,PLL,K} \cdot \Delta f_{\min}}{2^{10}} \\ B_{L,DLL} &= 1.04 \quad [\text{Hz}] \end{aligned} \tag{4.16}$$

The $1/2^{10}$ factor corresponds to the 10-bit shift-to-the-right at the output of the code discriminator. The noise bandwidth corresponds to the desired 1 Hz. Unfortunately, this loop noise bandwidth depends also on the SNR. Therefore, the discriminator shift controls $B_{L,DLL}$ as a function of this value. For example, if the SNR exceeds 14.5 dB, a 12-bit shift-to-the-right is applied instead of 10 bits.

The carrier aiding frequency is obtained from the velocity accumulator of the carrier-tracking loop (Figure 4.5). Where, the frequency increment is shifted to the right by 4 bits at the output of the loop filter. The PRN code clock NCO has thus the same frequency resolution than the velocity accumulator. Therefore, the velocity components are shifted by 11-bit, 13-bit, 15-bit and 17 bit.

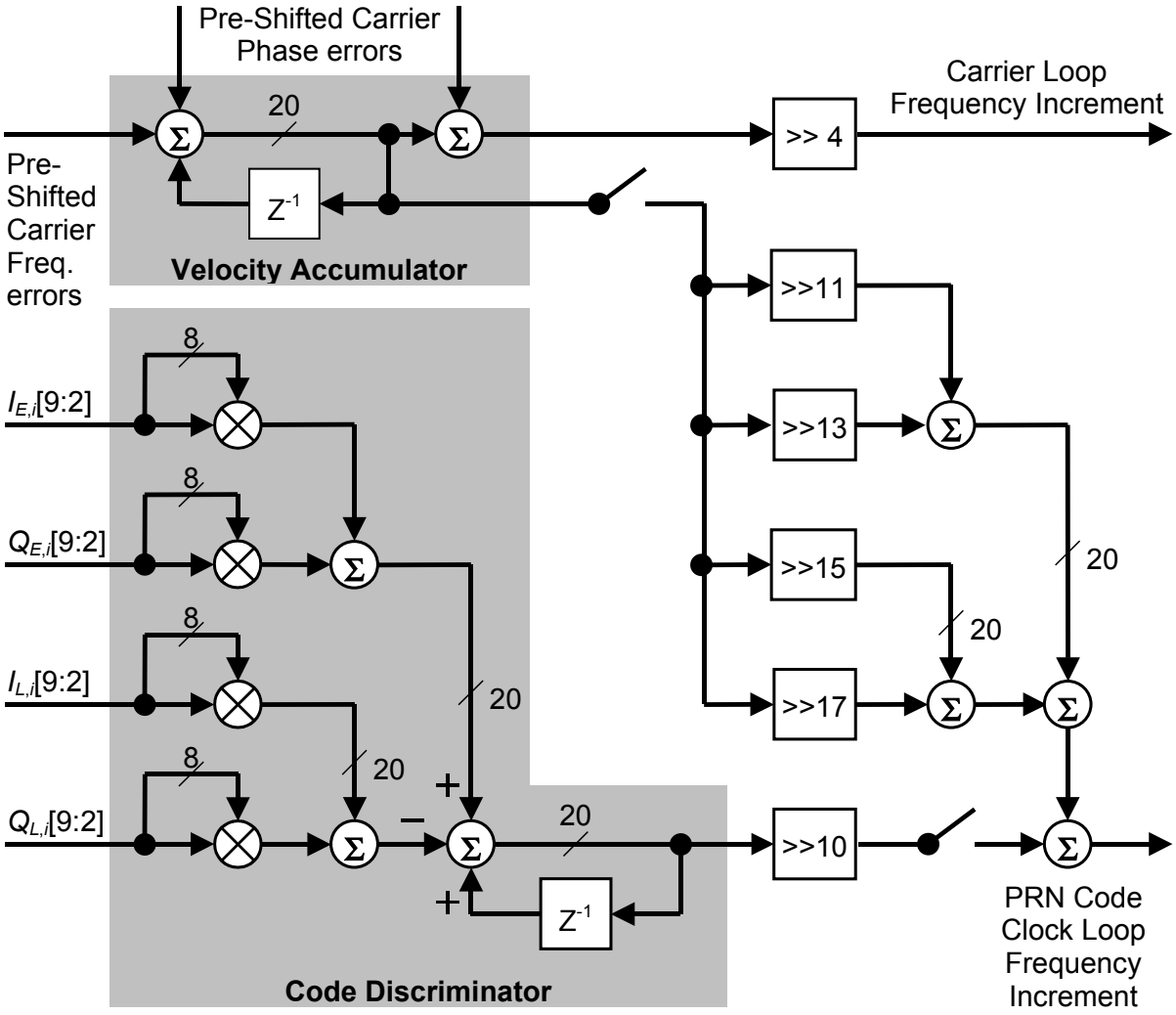


Figure 4.9: Block diagram of the carrier assisted DLL discriminators and filter.

4.5 DSP Implementation

In the preceding sections, the SV signal acquisition and tracking algorithms are completely defined. These algorithms constitute the most important performance characteristics for the GPS receiver. The following sections describe the DPS implementation into dedicated hardware.

4.5.1 Task Distribution

In GPS applications, most of the SSP is performed in real time. It requires important computational power, which can be met only with medium to big size computers or microprocessors. Unfortunately, these

kinds of processors have very important power consumption with respect to the other units as shown in Table 4.1. However, if an important part of the SSP algorithms is implemented into dedicated hardware, a small microprocessor for the navigation processing may become sufficient.

4.5.2 DSP Target Architecture

The GPS receiver DSP contains large block of samples to which complex algorithms are applied. Therefore, the bit-parallel architecture has been chosen for the DSP implementation, where each channel has its own dedicated processor. In principle, a commercially available processor could be used in this case and an 8-bit processor would be sufficient. However, in order to reduce further the power consumption, a dedicated solution has been implemented.

The target processor architecture is essentially composed of Memory Units (MU) and Arithmetic Unit (AU), communicating via a dedicated bus network. Figure 4.10 shows the block diagram of the DSP architecture, IMT3050M12, wherein each channel is composed of the following basic operational units:

- *The MU is composed of Read Only Memory (ROM) and Random Access Memory (RAM).*
- *The AU is composed of a Multiplier and an Accumulator (MAC).*

In addition, there are also more specific units, which can be regarded as hardware accelerators. These units perform the time critical operation more efficiently than the standard operational units:

- *The correlator is dedicated for the carrier and code wipe-off applications and signal filtering.*
- *The BitSync is responsible for the synchronization with the data bits.*
- *The Interruption Control Unit (ICU) records the interruptions generated in each channel and furnishes them to the CPU, upon a predefined order.*

The Correlator Control Unit (CCU) controls the basic operational units and the hardware accelerators.

In the DSP top unit, the main clock "*CLK"* is divided by 16 to form another clock "*CLK16"* at a lower frequency rate. The frequencies of these clocks are respectively 4.3625 MHz and 272.656 kHz. Both clocks

are used in the correlator, whereas the remaining processor units are exclusively clocked with *CLK16*.

The top unit also contains a test signal generator for hardware checks. The most important part is a five-stage PRN sequence generator, which can generate a GPS-like satellite signal with similar properties as the real SV signals. However, the repetition period of the PRN sequence is reduced to 31 chips instead of the 1023. With this generator, on-chip fast-acquisition and tracking tests are possible locally without external RF signals.

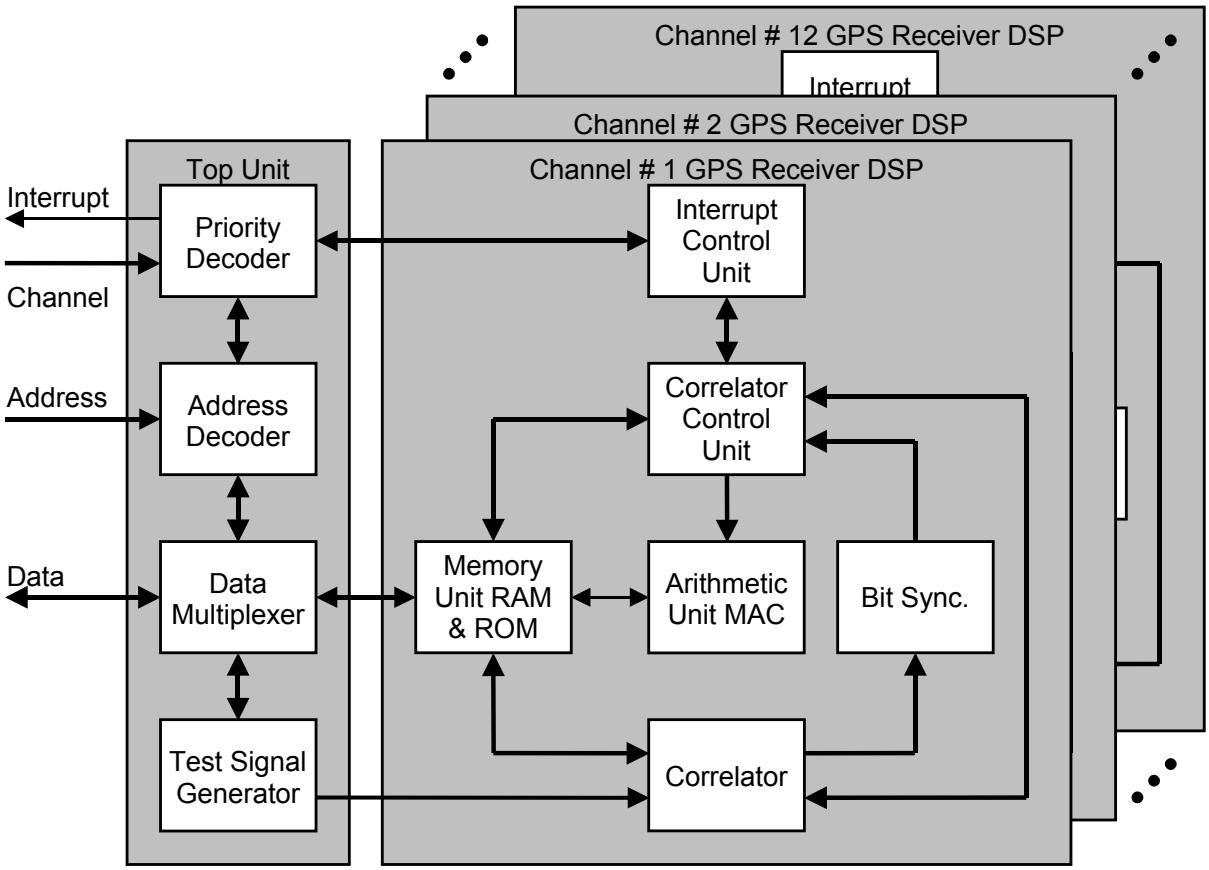


Figure 4.10: DSP Architecture.

4.6 Correlator

The correlator is dedicated to the carrier and code wipe-off and signal filtering. Figure 4.11 represents the functional block diagram of one correlator. First, the digitized pseudo-baseband signal plus noise is stripped off the carrier at 400 kHz. The resulting in-phase ($I_{1,k}$) and quarter-phase ($Q_{1,k}$) signals are correlated with the early and late

replicas. After the carrier and code striping processes, a digital pre-detection integrate and dump operation occurs. This filtering provides a re-sampling at 1 kHz.

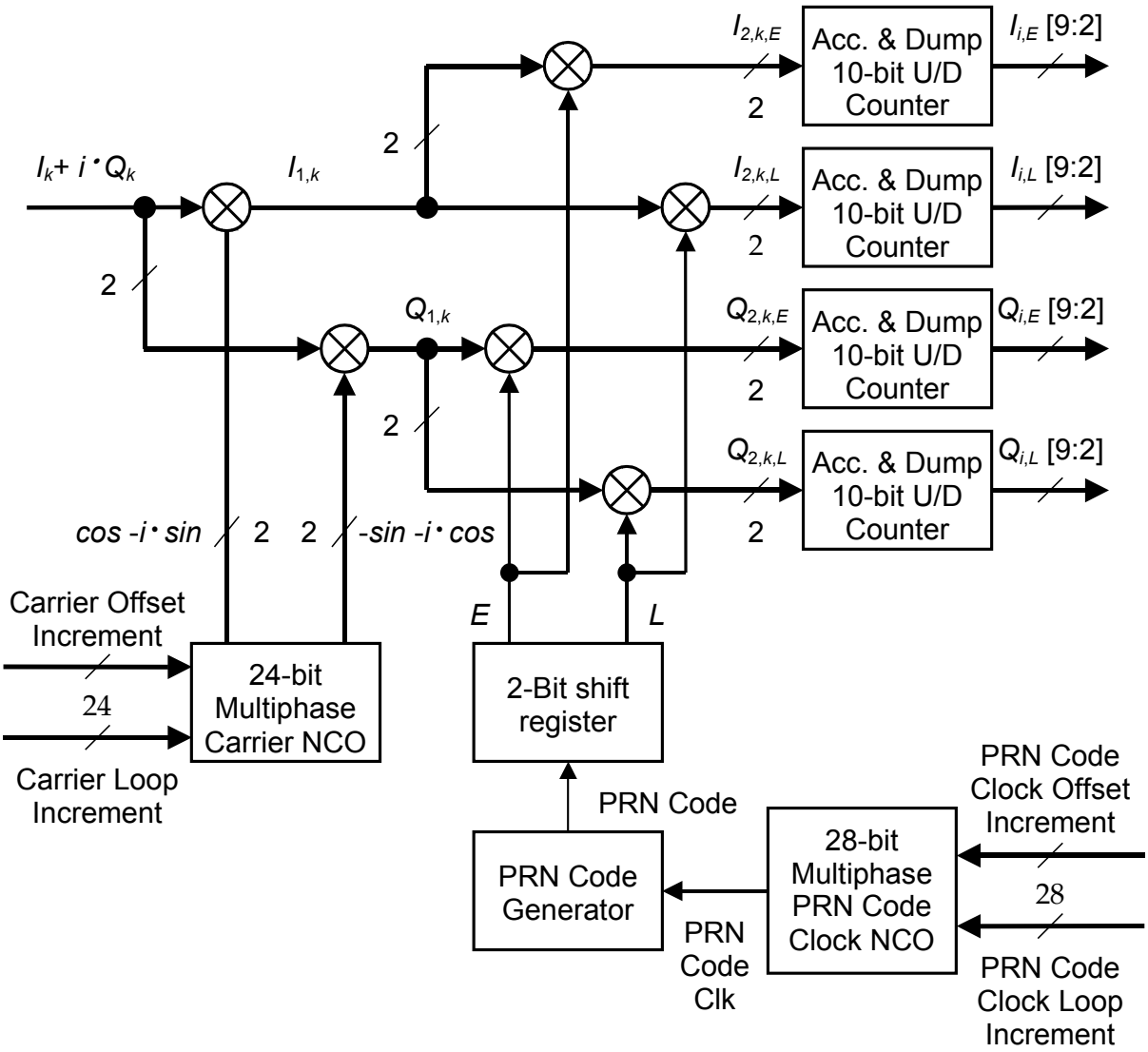


Figure 4.11: Correlator block diagram.

4.6.1 Signal Sampling & Quantification

Generally, the sample frequency defines the clock frequency of the DSP, which remains the most important power consumption factor. So, it is important to reduce this sampling frequency as much as possible without degrading the receiver performances.

Sampling

The complex sampling algorithm is by far the best sampling method for low power applications, because the sampling frequency f_s is only limited by the signal bandwidth. However, both the in-phase and the quarter-phase samples must be combined together to form a complex signal (section 3.5). Only the single side-band modulator uses the complex combination of both I_{sk} and Q_{sk}. This demodulation is executed with the following trigonometric identities:

$$\begin{aligned} I_{1,sk} &= (I_{sk} + i \cdot Q_{sk}) \cdot [\cos(\Phi_{rk}) - i \cdot \sin(\Phi_{rk})] \\ &= A_0 \cdot C_k \cdot D_k \cdot \cos(\Phi_k - \Phi_{rk}) \\ Q_{1,sk} &= -i \cdot (I_{sk} + i \cdot Q_{sk}) \cdot [\cos(\Phi_{rk}) - i \cdot \sin(\Phi_{rk})] \\ &= A_0 \cdot C_k \cdot D_k \cdot \sin(\Phi_k - \Phi_{rk}) \end{aligned} \qquad (4:17)$$

In this relationship Φ_{rk} is the referenced phase generated in the NCO at time t_k, and Φ_k is the phase of the I_k and Q_k samples.

The complex sampling has been chosen for the DSP implementation. In the RF stage the signal plus noise bandwidth is limited at 3 MHz. Therefore, the sampling frequency must be bigger than 3 MHz.

Pre Sampling Filtering

Pre-sampling filtering is necessary to prevent aliazing after sampling in a digital receiver. To limit the DSP clock frequency, there is a desire to set the filter bandwidth below the signal bandwidth. However, this limitation causes also correlation losses, which has been determined analytically in section 2.5.2.

The pre-sampling filtering is performed in the RF stage with a SAW filter, which bandwidth is 3 MHz. Figure 4.12 compares the resulting correlation function (dashed lines) with the ideal correlation function (straight line). The graphic shows that the correlation losses are not very important.

Quantification

The quantification process is the conversion of a discrete time signal into a digital signal, by expressing each sample value as a finite (instead of infinite) number of digits. The signal plus noise is quantified to 1 bit in the IFU. An AGC is therefore not required, which saves additional power

consumption. However, the 1-bit quantification and the pre-sampling filtering reduce the SNR by 3 dB [Cha89].

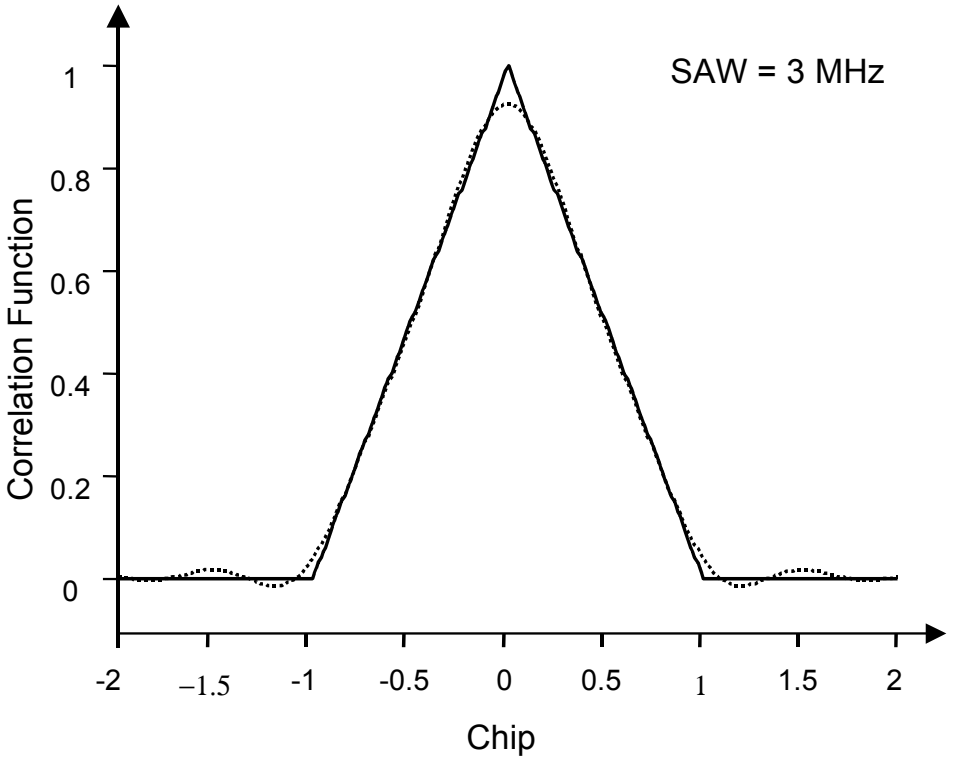


Figure 4.12: Correlation losses for a pre-correlation filtering of 3 MHz.

4.6.2 Carrier and Doppler Removal

The carrier and Doppler removal process is based on the trigonometric identities of equation (4:17). The IFU furnishes an equal proportion of either positive samples (+1) or negative samples (-1). As, the carrier reference is quantified to 1 bit; the implementation of the carrier and Doppler frequency removal process becomes very easy and simple. However, the 1-bit carrier reference quantification generates an additional 0.91 dB SNR loss in the demodulation process [Die95a].

Table 4.3 represents the truth table of the single side band modulator, where the signal (subscript S) and the reference (subscript R) carriers are quantified to 1 bit. If the signal is positive the encoded bit is “false”. Conversely, if the signal is negative the encoded bit is “true”. The truth table shows that the output of the modulator is ± 2 during half of the time. In the remaining cases, there are no operations. Therefore, the post-correlation accumulators can be realized with synchronous up-down

counters through the control of the up/down (UD) and the clock enable (CE) inputs. The counters operate at half the clock frequency, which saves additional power consumption.

The digital implementation of the single side band modulator is accomplished with the following logic operations:

$$\begin{aligned} UD_I &= I_S \oplus I_R \\ CE_I &= \overline{I_S \oplus I_R \oplus Q_S \oplus Q_R} \\ UD_Q &= I_R \oplus Q_S \\ CE_Q &= \overline{CE_I} \end{aligned} \tag{4:18}$$

I_S	I_R	Q_S	Q_R	$I_S I_R + Q_S Q_R$	$Q_S I_R - I_S Q_R$	UD_I	CE_I	UD_Q	CE_Q
0	0	0	0	+2	0	+	Yes	+	No
0	0	0	1	0	+2	+	No	+	Yes
0	0	1	0	0	-2	+	No	-	Yes
0	0	1	1	+2	0	+	Yes	-	No
0	1	0	0	0	-2	-	No	-	Yes
0	1	0	1	-2	0	-	Yes	-	No
0	1	1	0	-2	0	-	Yes	+	No
0	1	1	1	0	+2	-	No	+	Yes
1	0	0	0	0	+2	-	No	+	Yes
1	0	0	1	-2	0	-	Yes	+	No
1	0	1	0	-2	0	-	Yes	-	No
1	0	1	1	0	-2	-	No	-	Yes
1	1	0	0	+2	0	+	Yes	-	No
1	1	0	1	0	-2	+	No	-	Yes
1	1	1	0	0	+2	+	No	+	Yes
1	1	1	1	+2	0	+	Yes	+	No

Table 4.3: Truth table of the 1 bit single side band modulator.

4.6.3 Carrier NCO

The carrier generator has been realized with a multiphase NCO (see section 3.6), where both phase accumulators have 12 bits and the clock division factor *N*=16. Its power consumption is reduced almost by half with respect to a standard 24-bit NCO. The NCO splitting generates phase noise in the carrier-tracking loop, which has been determined for the FLL-assisted-PLL:

$$\sigma_{t\,\&\,Q,FLL/PLL} = \sqrt{\sigma^2_{t,FLL/PLL} + \frac{\left(2\pi \cdot \sigma_{Q,PLL}\right)^2}{8B_{L,PLL} \cdot K \cdot T}} \tag{4:19}$$

Both noise variances have been calculated in section 3.6.4 with the following design parameters:

- *The FLL noise bandwidth $B_{L,FLL}$=4 Hz.*
- *The PLL noise bandwidth $B_{L,PLL}$=18 Hz.*
- *The pre-detection integration time T=1 ms.*
- *The Post-detection averaging K=1.*
- *The CNR C/N_0=45 dB-Hz.*

The comparison between both noise variances shows that the NCO splitting has almost no effect on the carrier tracking loop noise performance:

$$\begin{aligned} \sigma^2_{t,FLL/PLL} &= 1.100 \cdot 10^{-3} \quad \left[\text{radian}^2\right] \\ \frac{\left(2\pi \cdot \sigma_{Q,PLL}\right)^2}{8B_{L,PLL} \cdot K \cdot T} &= 3.486 \cdot 10^{-4} \quad \left[\text{radian}^2\right] \end{aligned} \tag{4:20}$$

Figure 4.13 shows the schematics of the multiphase NCO. Three Carrier Save Adders (CSA), one Carry Propagate Adder (CPA) and one Brent & Kung Adder (BKA) compose the design. In the CSA structure, each bit has the binary output encoded in the sum and carry bit. The carry bit is not saved in a flip-flop, but it is shifted to feed the carry input of the next rank. This kind of addition structure is often called Redundant Binary Adder (RBA), because two bits are used to represent each digit.

In the LSBA, the binary output is obtained by summing each sum and carry bit with a CPA. In the MSBA, a faster BKA architecture is used for the final sum, because its clock frequency is higher.

At the beginning of each *CLK16* cycle, the *MX-NCO* signal enables the overflow of the LSBA to be shifted and added to MSBA.

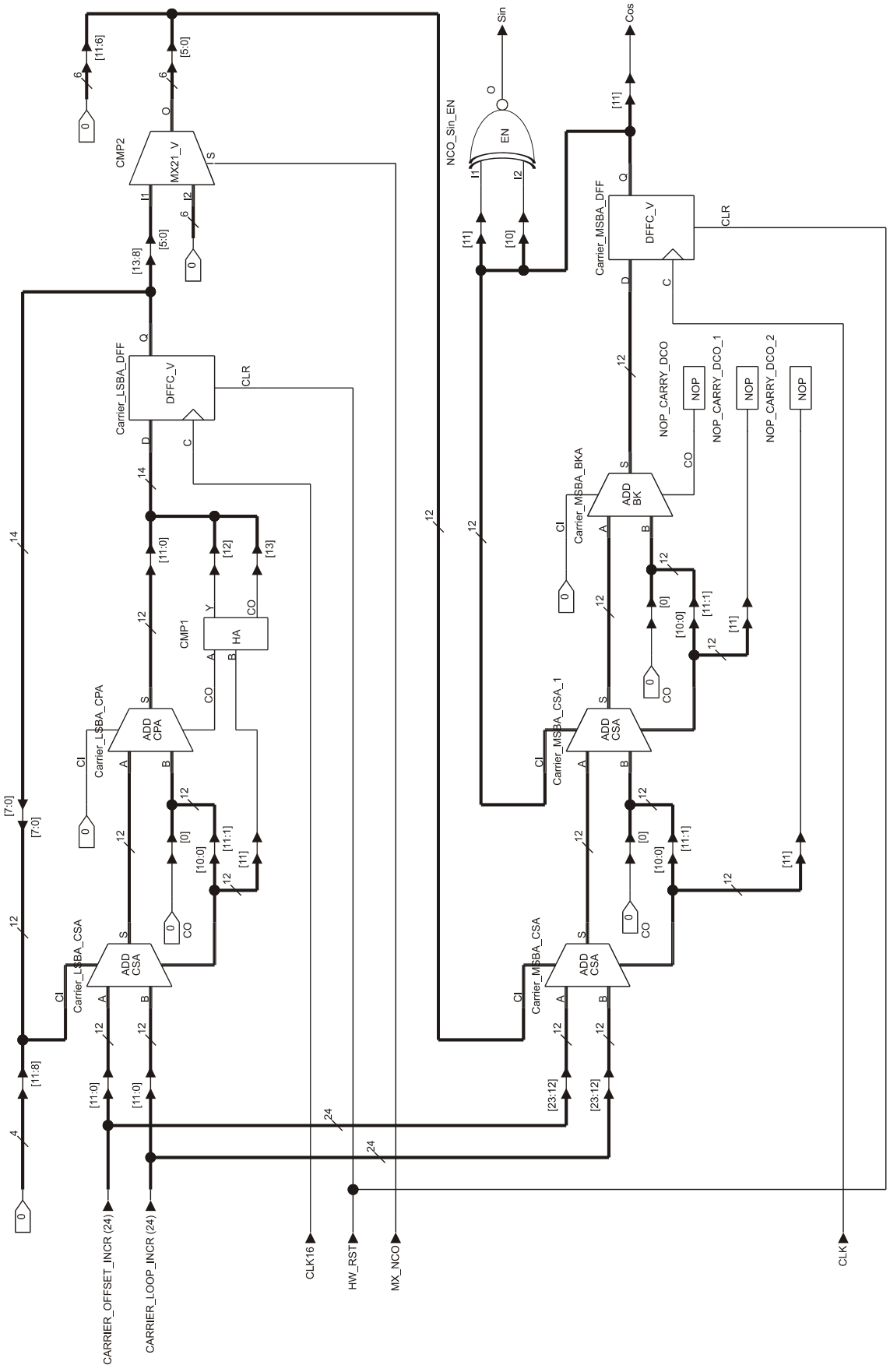


Figure 4.13: Schematic of the multiphase NCO.

4.6.4 PRN Code Clock NCO

The PRN code clock generator has been also realized with a multiphase NCO, where both phase accumulators have 14 bits and the clock division factor *N*=16. Again, the power consumption is almost decreased by half with respect to a standard 28-bit NCO.

The additional phase noise in the PRN code-tracking loop is:

$$\sigma_{t\,\&\,Q,DLL} = \sqrt{\sigma_{t,DLL}^2 + \frac{\sigma_{Q,DLL}^2}{8B_{L,DLL} \cdot K \cdot T}} \tag{4:21}$$

Both noise variances have been calculated in section 3.6.4 with the following design parameters:

- *The DLL noise bandwidth $B_{L,DLL}$=1 Hz.*
- *The predilection integration time T=1 ms.*
- *The post-detection averaging K=16.*
- *The CNR C/N_0=45 dB-Hz.*

The comparison between both noise variances shows that the NCO splitting has almost no effect on the code tracking loop noise performances:

$$\begin{aligned} \sigma_{t,DLL}^2 &= 6.05 \cdot 10^{-6} \quad \left[\text{chips}^2\right] \\ \frac{\sigma_{Q,DLL}^2}{8B_{L,DLL} \cdot K \cdot T} &= 6.21 \cdot 10^{-7} \quad \left[\text{chips}^2\right] \end{aligned} \tag{4:22}$$

The equation above demonstrates again that the noise induced by the NCO splitting is less than the DLL thermal noise.

4.6.5 Coder Implementation

The implementation of a PRN code generator is straightforward and has been illustrated in chapter 2.

To prevent the acquisition of multipaths during the signal acquisition process, the coder is delayed every pre-detection interval from early towards late. This delay equals one chip and is achieved by canceling one PRN clock slope. The signal power at each code phase position is measured in both complex channels (e.g. early and late) and compared with a pre-determined threshold. Therefore, the resolution of the signal acquisition process corresponds to ½ chip. The decision to retain a

certain code phase is made at the end of the pre-detection interval. At this moment, the next code phase has already been generated. For this purpose, the Code Slide multiplexer allows recovering the previous code phase (Figure 4.14).

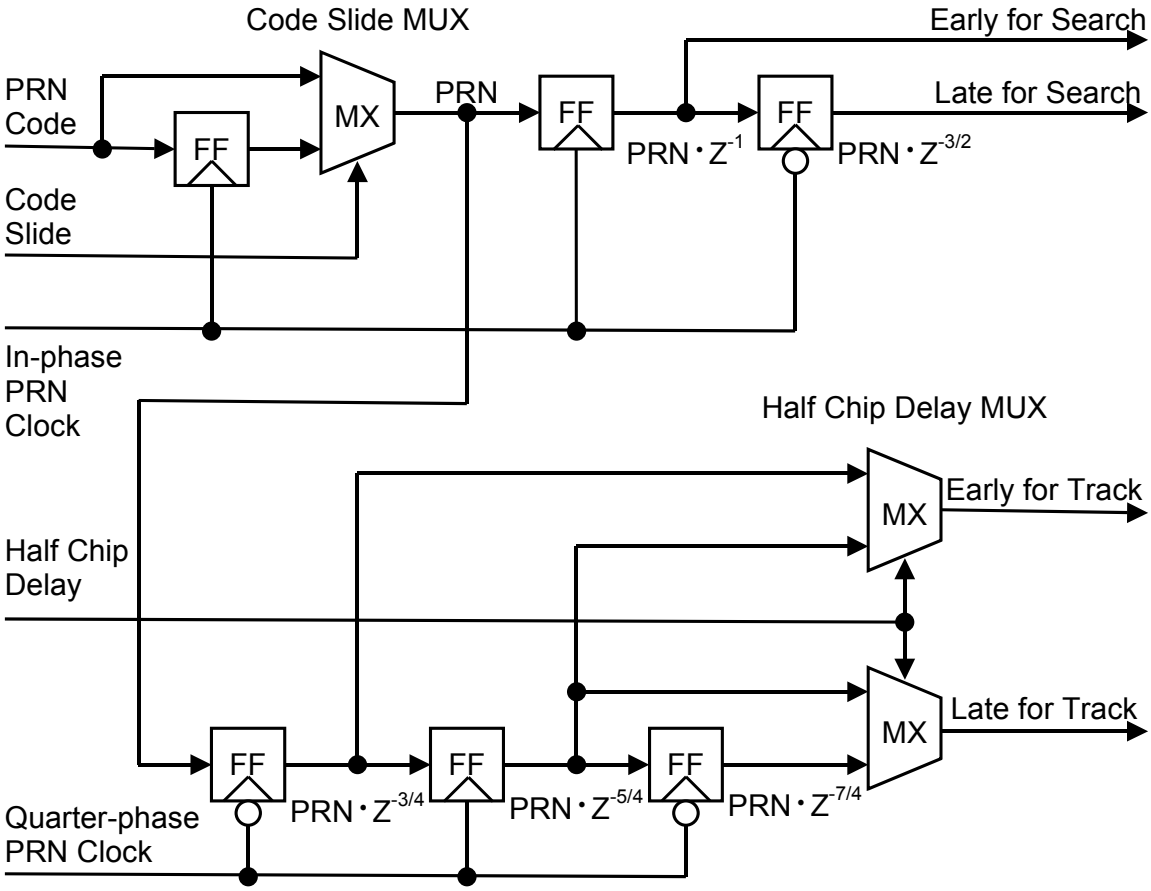


Figure 4.14: PRN code shift register.

When the tracking process begins, the complex channel with the biggest power is set as the pseudo prompt channel. The pseudo term is used here, because the correlator does not have a physical prompt channel. If the power of the early channel is bigger than the power of the late channel, the PRN code is advanced by a quarter-chip. Inversely, if the power of the late channel is bigger than the power of the early channel, the PRN code is delayed by a quarter-chip.

The quarter chip shift operations described above require a PRN clock, which is the double of the PRN code frequency. This represents an important drawback, because three important criteria govern the *CLK* frequency until yet:

- *The complex sampling frequency algorithm, where the bandwidth of the anti aliazing filter limits the CLK frequency at a minimum of 3 MHz.*
- *The minimal clock frequency necessary for both the carrier and the PRN code clock NCOs.*
- *The CLK frequency should be asynchronous with those of the carrier and the PRN code.*

The carrier and PRN code frequencies are respectively 400 kHz and 1.023 MHz. Thus, the ideal clock frequency is 3.5 MHz, which is to small to generate the quarter shift operations mentioned above. Fortunately, the quarter chip shift operations can be generated in a different way, using the quarter-phase component of the PRN code clock NCO. The in-phase component is used during signal acquisition whereas the quarter-phase component is used during signal tracking (Figure 4.14).

4.6.6 Pre-Detection

Pre-detection is the signal processing that is performed after the carrier and code stripping process. During this process the signal bandwidth is narrowed prior to being passed through the signal discriminators. Digital integrators provide the accumulation filtering over *T* seconds. The required dynamics of these accumulators can be determined by considering the square root of the maximum signal power:

$$P_s = \frac{C}{N_0} \cdot T \cdot \sigma_n^2 \qquad (4:23)$$

Here, σ_n represents the RMS noise power in one accumulator. This can be obtained by simply combining the mean-square value of each sample with their likelihood of occurrence and the sampling rate. The truth table (Table 4.3) shows that in 50% of the time the output of the modulator is ±2. The rest of time there is no operation. This proportion is not changed during correlation with the reference PRN codes. Therefore the mean noise power in the up-down counter becomes:

$$\sigma_n^2 = \frac{f_{CLK} \cdot T}{2} = 2181 \qquad (4:24)$$

This relationship allows the determination of the up-down counter size in the following way:

$$N = \log_2\left(\sqrt{\frac{C \cdot f_{CLK} \cdot T^2}{2N_0}}\right) + 1 \qquad (4:25)$$

For a CNR of 50 dB-Hz, a clock frequency of 4.3625 MHz and the integration interval of 1 ms, a 10-bit accumulator would be sufficient. Most of the time the CNR is below 50 dB-Hz, but sometimes even with small antennas, the CNR can exceed 50 dB-Hz. An 11-bit accumulator is then preferred. An elegant solution resides in a 10-bit accumulator with saturation, which reduces the size of the arithmetic operations that follow. The consequence is simply a 20-dB SNR limitation when the CNR exceeds 50 dB-Hz. However, if the CNR exceeds often 50 dB-Hz, a larger accumulator is recommended because the discriminators loose their linearity.

For the saturation solution, it is not necessary to use all 10 bits for the acquisition and tracking processes. For example, if the 8 Most Significant Bits (MSB) are only retained for the following arithmetic operations, a large amount of hardware resources can be saved. The mean noise power of the 8 MSB of the accumulator is given by:

$$\sigma^2_{n,8MSB} = \frac{\sigma^2_n}{16} = 136 \qquad (4:26)$$

The accumulator quantification variance becomes:

$$\sigma^2_Q = \frac{Q^2}{12} = \frac{4}{3} \qquad (4:27)$$

The accumulator quantification variance is small with respect to the mean noise power given at (4:26).

4.7 Arithmetic Unit

The key to high performance in DSP hardware processing is to have the hardware computational elements organized in such a way, that they match the structure of the computation algorithm at hand. The computational structure of the acquisition and tracking algorithms are of the "sum of product" type (e.g. $X \times Y + A \Rightarrow A$). The basic building blocks should be interconnected in such a way that the instruction cycle time is minimized and most algorithm steps can be accomplished in one

instruction. Figure 4.15 shows the block diagram of the required four-ported register, which is composed of three independent ports for the read operations and a fourth for the write operation. The addresses decoding, the operands fetching, the arithmetic operations and the results storing are performed in one clock cycle.

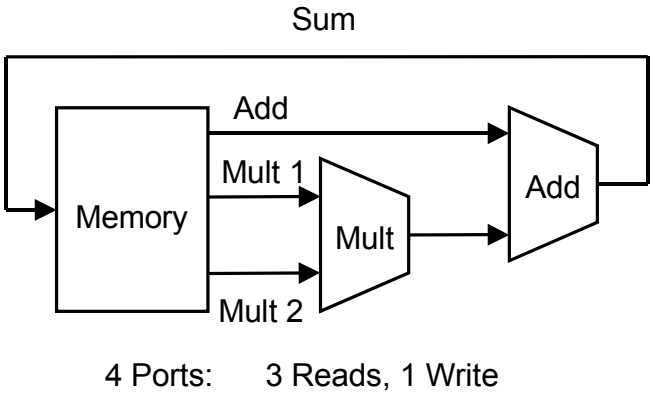


Figure 4.15: Multiple register address architecture.

The register file can be implemented using D-flip-flops. The size of this block depends on the maximum number of intermediate results that needs to be held at a time. This can be determined by implementing the chip function with a high-level language (such as C) first.

The AU module performs multiplication, addition, subtraction and shift operations. Since arithmetic operations dominate the execution of the acquisition and tracking algorithm, a high-speed AU is required.

Figure 4.16 illustrates the schematic diagram of the required four-port AU. Three operands can be read independently. "*Mult1*" and "*Mult2*" inputs are dedicated to the multiplication whereas the "*Add*" input is used for the addition. The "*Acc*" signal enables the accumulation of the internal sum, which sign is defined by the "*Sub-or-Add*" signal.

A parallel 8×8 Redundant Binary Multiplier (RBM) and a RBA compose the AU. RBA are structures where each bit has the binary output encoded in the sum and carry of each bit. They are composed of CSA where the carry output of each stage is left shifted to feed the carry input of the next rank. The BKA achieves the Redundant Binary (RB) conversion in to 2's complement. This structure is highly parallel. Therefore, no carry propagation delay penalty slows down the AU.

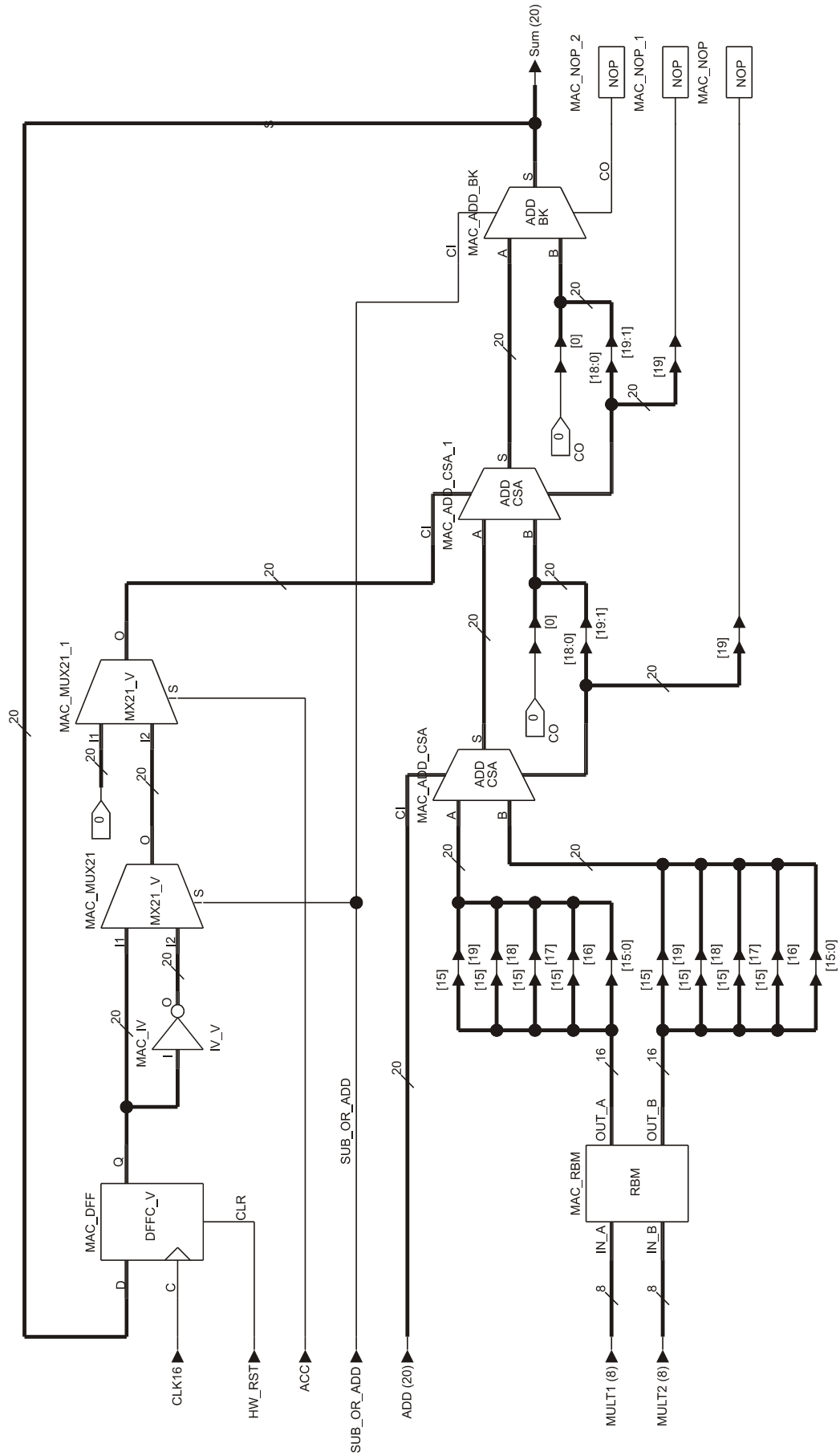


Figure 4.16: Schematic of the arithmetic unit.

The parallel 8×8 multiplier uses the Modified Booth Algorithm (MBA), proposed by Mac Sorley. The principle of the MBA follows from the standard Booth Algorithm, which starts by appending a 0 to the right of the operand *X*'s Least Significant Bit (LSB) (denoted X_{-1}) and then looking at binary pairs beginning with X_{-1}. According to Table 4.4, a multiple of operand Y is added to the partial products at every step and the result is shifted to the right by 1 bit.

X_{i+1}, X_i	Add to Partial Product
00	+0
01	+*Y*
10	-*Y*
11	-0

Table 4.4: Booth algorithm.

In the MBA, triplet of n bits instead of pairs is looked at (Table 4.5). In this way, if n represents the (even) number of bits, the number of partial product decreases from n to n/2. This improve both speed and hardware requirements.

X_{i+2}, X_{i+1}, X_i	Add to Partial Product
000	+0
001	+*Y*
010	+*Y*
011	+2*Y*
100	-2*Y*
101	-*Y*
110	-*Y*
111	-0

Table 4.5: Modified Booth algorithm.

Various classes of parallel multipliers use the MBA. Figure 4.17 illustrates the dominant scheme where the MBA is used along with a RBA. In this structure the product remains redundant and is highly

parallel. Therefore, no carry propagation delay penalty appears. As mentioned above the BKA achieves the RB conversion in to 2's complement.

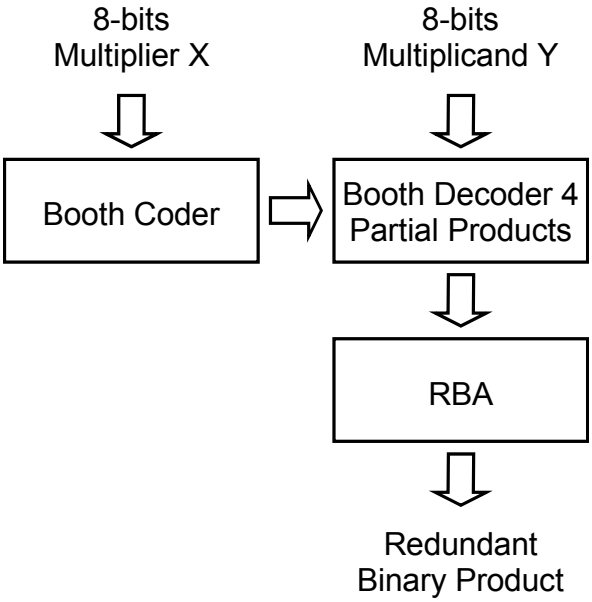


Figure 4.17: Block diagram of a multiplier using the MBA.

4.8 Memory Unit

Figure 4.18 illustrates the MU, which is responsible for providing operands to the AU and storing the results of the AU. The MU contains both the processing data and the fixed coefficient constants. The data are stored in a RAM, which can perform both read and write operations during the same clock cycle. A ROM contains the fixed coefficients.

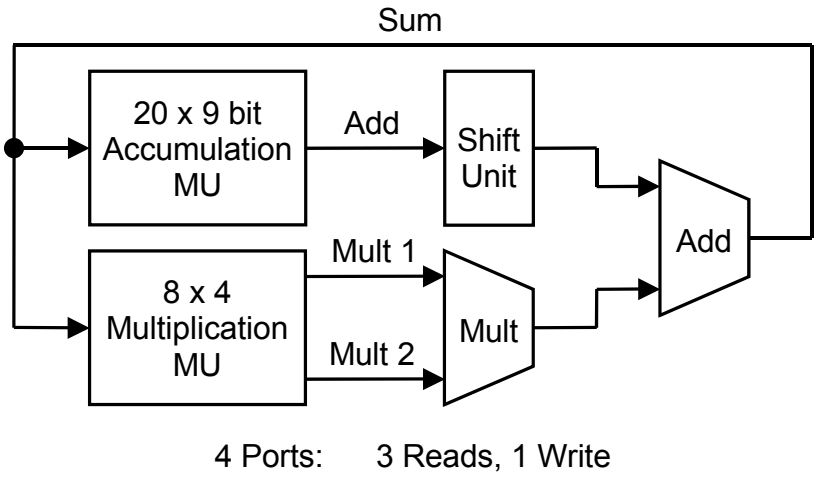


Figure 4.18: Multiple register address architecture.

Table 4.6 and Table 4.7 furnish the data structure of the MU.

Variable to hold for the Multiplication AU	Connection with the AU ports	Location of the variable
Accumulated In-Phase Early (AIE)	*Mult1* & *Mult2*	Correlator
Accumulated In-Phase Late (AIL)	*Mult1* & *Mult2*	Correlator
Accumulated Quarter-Phase Early (AQE)	*Mult1* & *Mult2*	Correlator
Accumulated Quarter-Phase Late (AQL)	*Mult1* & *Mult2*	Correlator
In-Phase Prompt (IP)	*Mult1*, *Mult2* & *Sum*	Multiplicand MU
Quarter-Phase Prompt (QP)	*Mult1*, *Mult2* & *Sum*	Multiplicand MU
In-phase PromptZ^{-1} (IPZ)*	*Mult2* & *Sum*	Multiplicand MU
Quarter-Phase PromptZ^{-1} (QPZ)*	*Mult2* & *Sum*	Multiplicand MU

Table 4.6: Data structure of the Multiplication MU.

Variable to hold for the Accumulation AU	Connection with the AU ports	Location of the variable
Early Energy (EE)	*Add* & *Sum*	Addition MU
Prompt Energy (PE)	*Add* & *Sum*	Addition MU
Late Energy (LE)	*Add* & *Sum*	Addition MU
Frequency Discriminator (FD)	*Add* & *Sum*	Addition MU
Phase Discriminator (PD)	*Add* & *Sum*	Addition MU
Code Discriminator (CD)	*Add* & *Sum*	Addition MU
Accumulated Velocity (AV)	*Add* & *Sum*	Addition MU
Carry Loop Increment (CaLI)	*Sum*	Addition MU
Code Loop Increment (CoLI)	*Sum*	Addition MU

Table 4.7: Data structure of the Accumulation MU.

4.9 Correlator Controller Unit

In the preceding sections, the acquisition and tracking algorithms are introduced together with the DSP architecture. This section will present the challenging process of implementing the acquisition and tracking algorithms into the dedicated hardware, where each channel has its own processor (Figure 4.10).

In each processor, the operational units are controlled by the CCU, which determines the data transfers between the MU, the AU and the correlator. The CCU can be realized either with dedicated hardware or using a micro-code program [Arm93].

The DSP has to accomplish the computational task of acquisition and tracking algorithms in one instruction. Address decoding, operands fetching, arithmetic operations and results storing must be performed in one clock cycle. Therefore, the speed advantage of the hardwired controller becomes the decisive argument. However, the hardwired controller has two important disadvantages: a greater complexity and the need for a specific design. Both problems can be solved with the new generation of CAD tools, which allow high-level circuit representation, either with VHDL or with a state diagram. Therefore, a complex description of the design at the gate level is not necessary.

4.9.1 Flow Chart

A flow chart is used to help the mapping of the acquisition and tracking algorithms into the dedicated hardware (Figure 4.19). This chart presumes that the CPU has already initialized the PRN code number and the frequency parameters.

The DSP first waits for the channel to be started. If so, it enters into an idle mode, waiting for the Accumulate and Dump Occurred (ADO) interrupts. The correlator generates these interruptions at the end of every PRN code epoch (e.g. after 1 ms). When an ADO interrupt has occurred, the DSP accomplishes either the signal acquisition or the signal tracking algorithms.

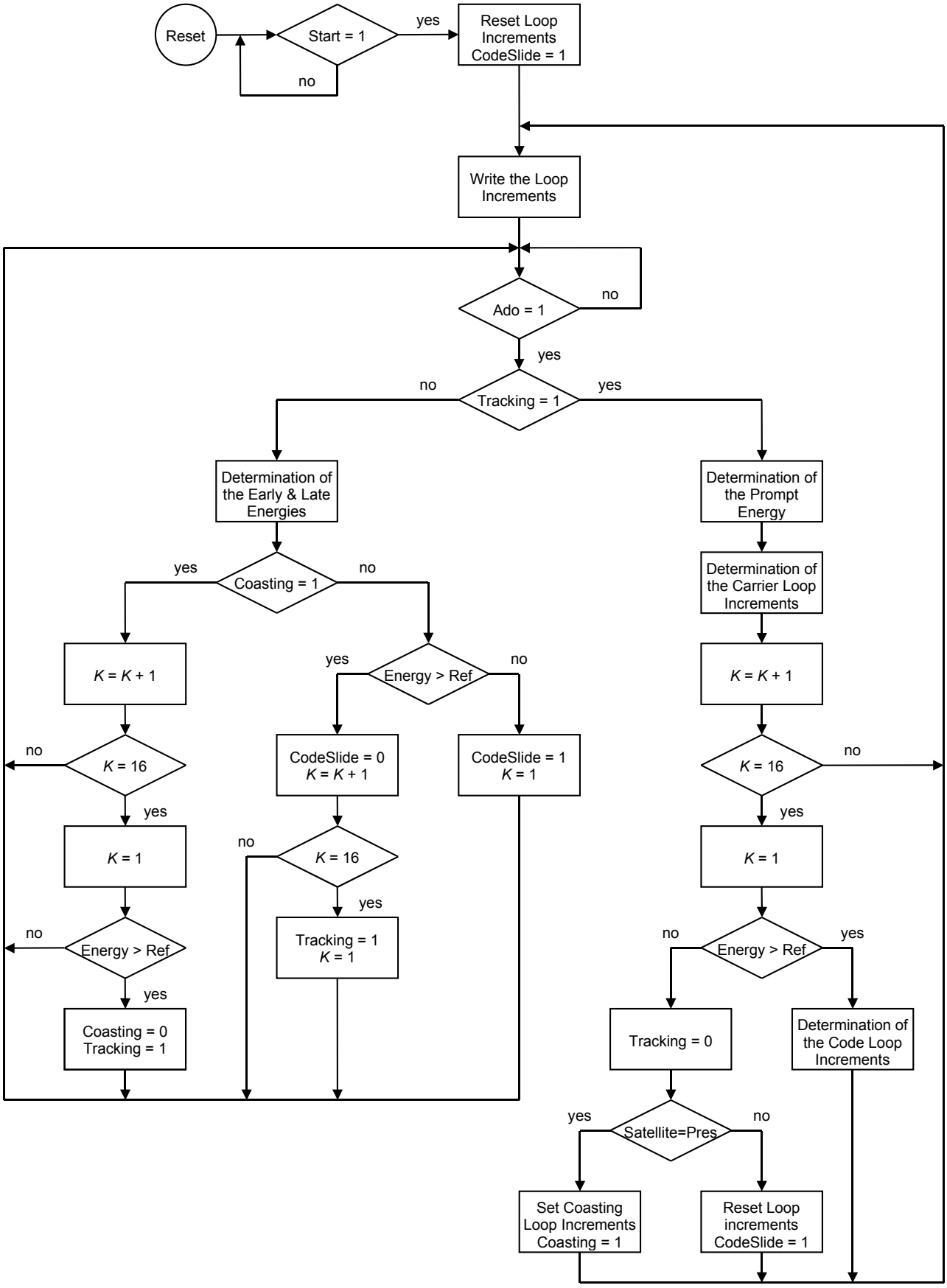


Figure 4.19: GPS receiver DSP Flow Chart.

Signal Acquisition

The signal acquisition algorithms are executed either in the searching mode or in the coasting mode, according to the modified MDD algorithm introduced in section 4.3.1. During both modes, the DSP computes the signal energy in the early and late channels and compares their values with the predefined thresholds. If the SV signal is declared “present”, the DSP enters in the tracking mode.

Signal Tracking

In the tracking mode, the DSP executes the signal tracking algorithms defined in section 4.4. During this mode, the DSP computes the signal energy of the pseudo prompt channel and compares its value with a predefined threshold every 16 epochs (e.g. K=16).

As long as the SV signal strength is sufficient, the channel performs bit synchronization and outputs the GPS message. However, if the signal strength drops bellow the threshold, then the DSP enters respectively in the coasting mode or in the searching mode, upon the presence of the SV has been confirmed or not.

It is important to distinguish between the presence of the SV and the presence of its signal. A SV can be present without reception of its signal. For example, if an object masks the antenna, the signal is lost but the SV may always be present. In the DSP, the SV is declared “present” when its data bit stream has been demodulated.

In the coasting mode the DSP set up the carrier and PRN code frequencies according to the preceding loop increments. Next, it waits until the SV signal comes back. During this mode, both carrier and code loop updates are frozen. When the SV signal reappears, the DSP enters into the tracking mode. However, if the coasting mode exceeds 2 minutes, the microprocessor restarts the DSP channel.

4.9.2 State Diagram

The translation of the flow chart into a Finite State Machine (FSM) description is a tedious task, especially when applied to a real-time process where testing is difficult.

The resulting FSM is quite complex and its detailed representation is not in the scope of this thesis. However, note that the FSM contains exactly

32 states, and therefore, it can be coded with 5 flip-flops without generating forbidden or dead states.

4.10 Bit Synchronization

Upon initial wide-band C/A-code acquisition, the position and time uncertainty are large. The C/A-code epoch ambiguity results in the lack of knowledge of data-bit timing, which requires bit synchronization. There are numerous techniques available to achieve this.

The histogram approach breaks an assumed data-bit period (20 ms) into 20 C/A-code epoch periods (1 ms) and senses sign changes between successive epochs. For each sensed sign change, a corresponding histogram cell count is incremented until the value in one specific cell exceeds the other 19 bins by a pre-specified amount.

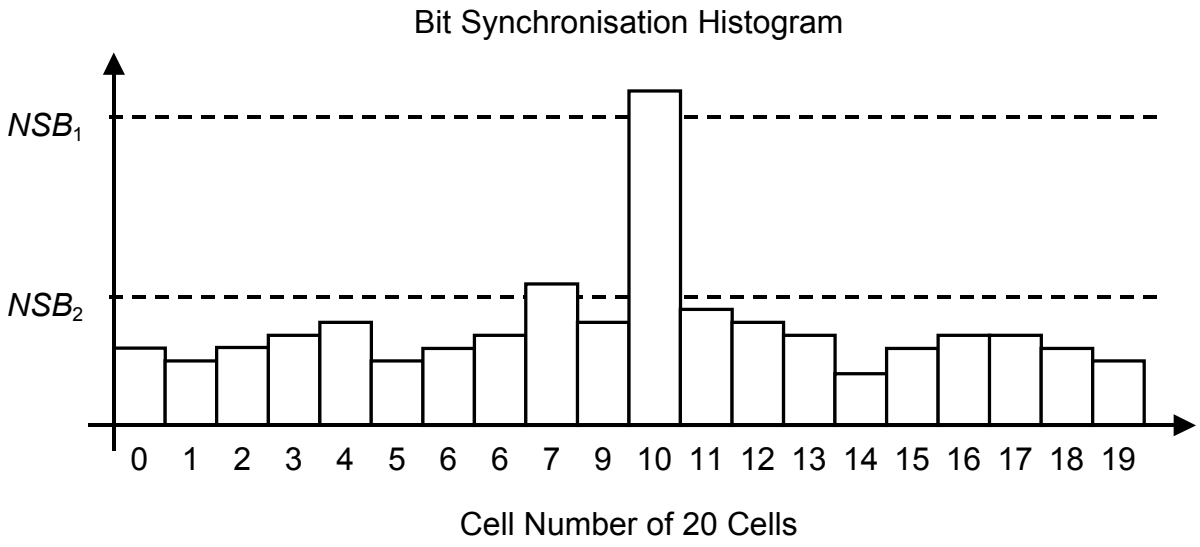


Figure 4.20: Example of a successful synchronization histogram.

Figure 4.20 illustrates such a histogram for successive bit synchronization along with count threshold. The procedure is as follows [Die95b]:

- *Cell counter Kcell is arbitrarily set and runs from 0 to 19.*
- *Each sensed sign change is recorded by adding 1 to the histogram cell corresponding to cell Kcell.*
- *The process continuous until one of the following cases occurs.*
  a) Two cell counts exceed threshold NBS_2.

b) Loss of lock.
c) One cell count exceeds threshold NBS_1.

- *If (a) occurs, bit synchronization fails because of low CNR or lack of bit sign transitions, and bit synchronization is reinitialized. If (c) occurs, bit synchronization is successful, and the C/A-code epoch counter is reset to the correct value.*

The realization of the histogram approach is complex and expensive to implement in hardware. Therefore, a more efficient bit synchronization algorithm has been developed for the DSP architecture. Figure 4.21 illustrates the block diagram of the new bit synchronization algorithm, called BitSync.

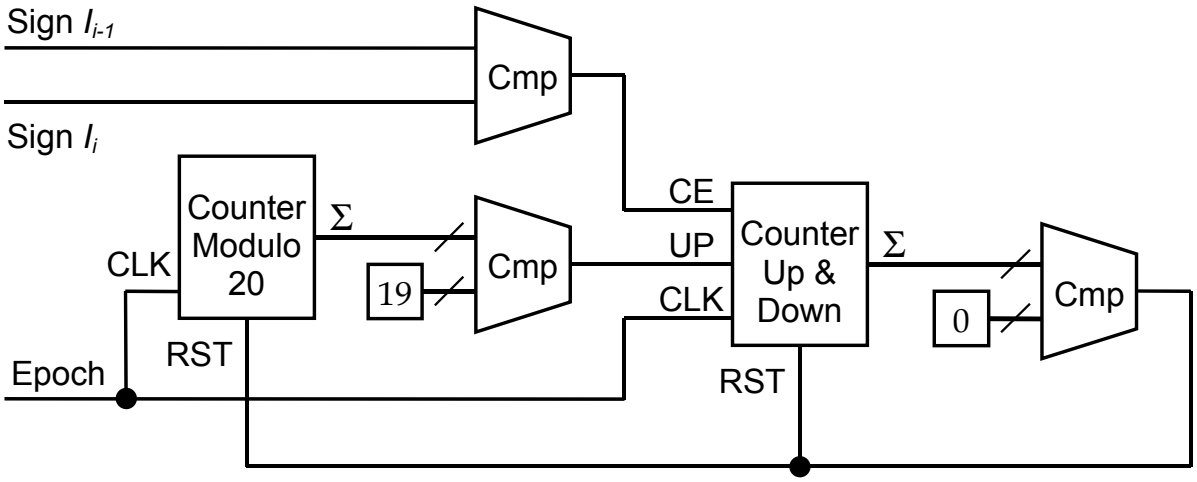


Figure 4.21: New bit synchronization algorithm.

The principle of the BitSync is the following. An arbitrary C/A-code epoch is assumed to be the first epoch of the data-bit and the up & down counter is initialized at threshold TH_1. Every time when a sign change appears at intervals of 20 C/A-code epochs (20 ms), the up & down counter is incremented. Otherwise, if the sign changes appear outside the 20 C/A code epochs, the up & down counter is decremented.

The bit synchronization process fails when the up & down counter reaches zero. In this case the “BitSync” process is reinitialized at the next C/A-code epoch. The bit synchronization process is successful when the up & down counter reaches the threshold $TH_2 > TH_1$. In this case the C/A-code epoch counter is reset.

A second use of the up-down counter is to provide an indication when the FLL-assisted-PLL loop discriminator may be started. Therefore, another threshold TH_3 has been defined, which has been fixed between TH_2 and TH_1.

4.11 Interruption Control Unit

Each DSP channel can generate specific interruptions to indicate its actual state (Table 4.8). The Interruption Control Unit (ICU) records these interruptions and furnishes them one by one to the microprocessor, upon a predefined order. The interruption causes can be read in the Status Register. The confirmation process is achieved by writing the received value in the same register. The ICU has been generated with a FSM, which detailed description is not in the scope of this thesis.

Interrupt Cause	**IC_2 down to IC_0**
No Interrupts	000
New Data Bit Available	001
Satellite Search Done	010
Coasting Started	011
New Pseudo-Range Recorded	100
Bit Synchronization Lost	101
Tracking Mode Entered	110
Accumulate & Dump Occurred (only in FSM test mode)	111

Table 4.8: Bit description of the Interrupt causes.

4.12 Priority Decoder

To reduce the complexity of the channel selection, a new priority based access scheme has been realized (Figure 4.22). In normal mode, it enables the direct access to the different channels. In the virtual mode the priority decoder enables the access to those channels, which have generated an interrupt. The access is defined upon a given priority order.

The channel with the highest identification number has the highest priority.

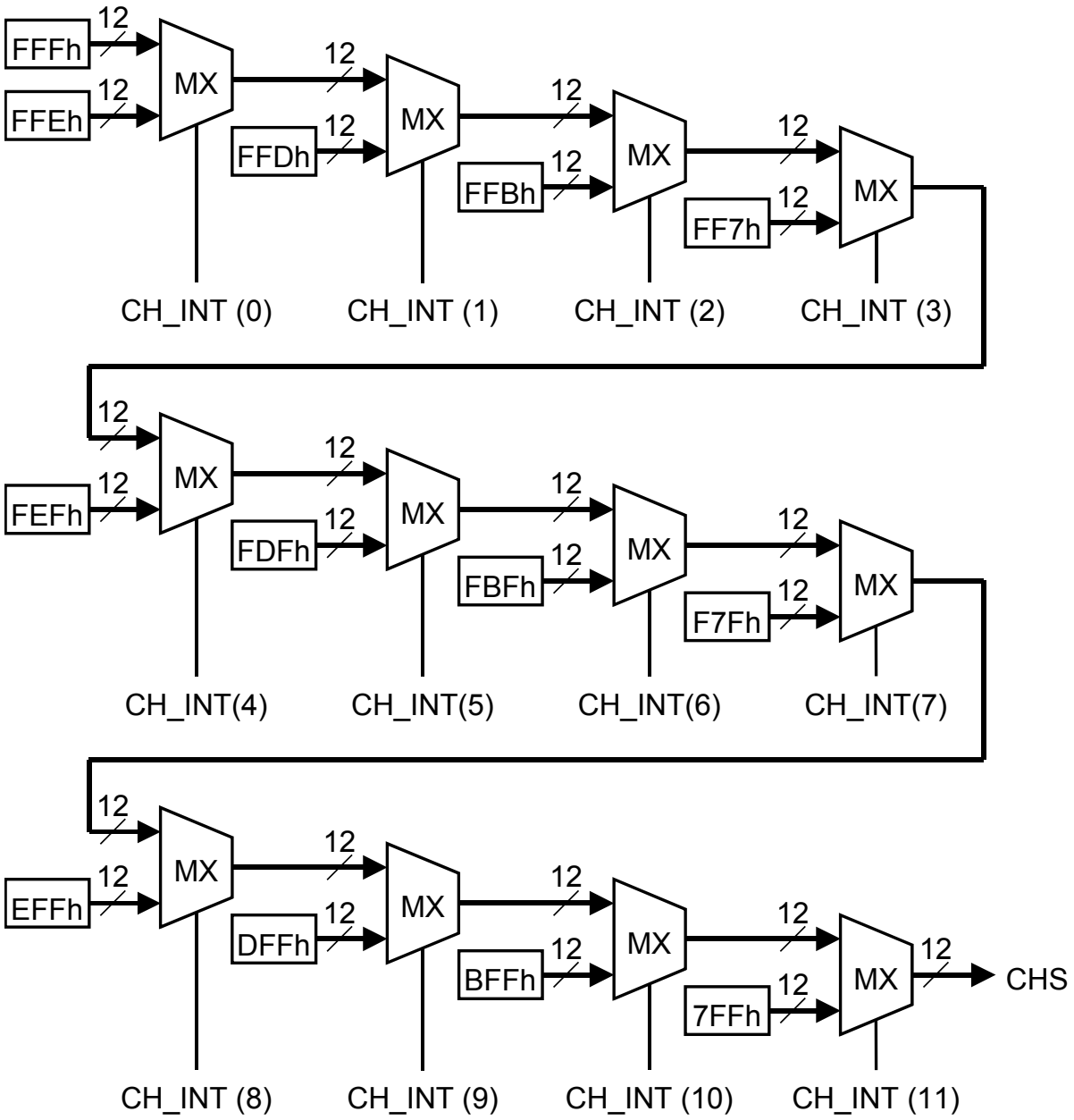


Figure 4.22: Block diagram of the Priority Decoder.

The advantage of the priority decoder appears when the DSP generates interruptions. Using the priority decoder, the microprocessor does not need to determine, which channel has generated the interruption. Instead, it is sufficient to access the channel through the reserved virtual entry that directly presents the highest channel in an interrupt state with the highest priority. This avoids the microprocessor to poll every channel.

4.13 Test Signal Generator

The integrity of the system can be tested without the need of a real GPS input signal. This test should be realized at system boot and verifies whether the interaction between the sub-circuits operates as expected. It is assumed that the sub-modules have past the elementary tests successfully.

The major purpose of the Loop-Back test is to check the complete circuit operation in near to real conditions in a small amount of time. As the use of real GPS signals will make the test set up impractical and will require too much time; special arrangements have been made. In particular, the length of the PRN code generator is reduced from 2^{10}-1 to 2^5-1, and this for two main reasons:

- *The code search procedure is speed up, because both the code length and the number of code phases are reduced. This results in a speed up of 1000.*
- *The possibility to test the circuit with a non-noisy signal while using the same control circuitry.*

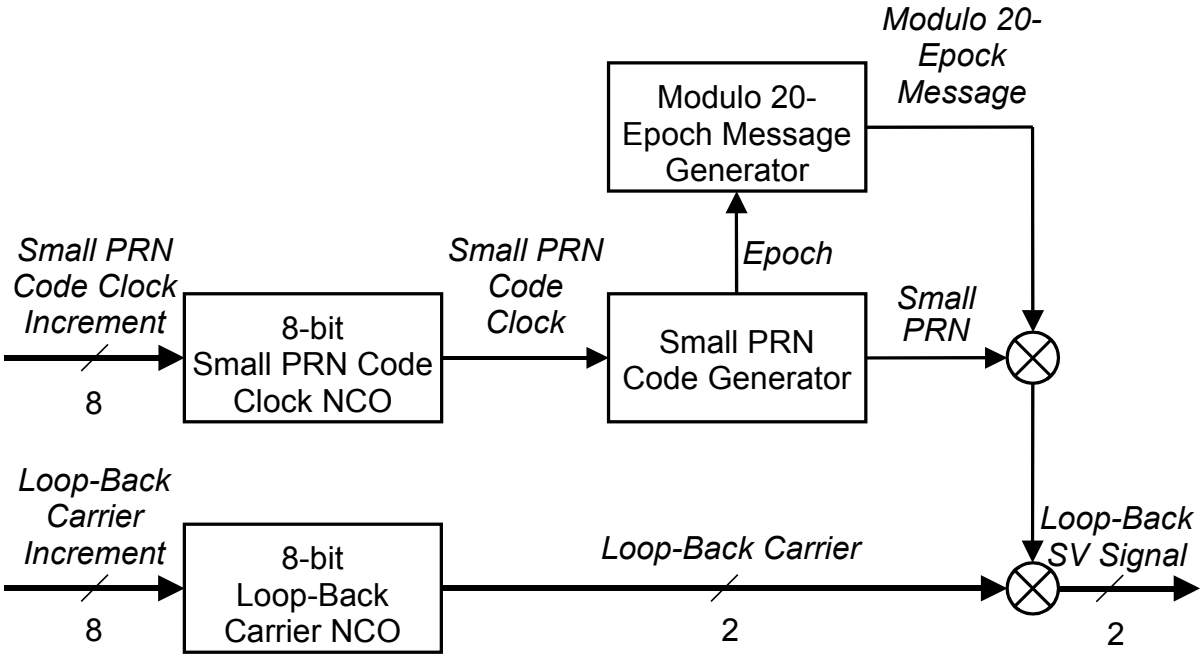


Figure 4.23: Loop-Back SV Signal Generator.

Figure 4.23 illustrates the block diagram of the Loop-Back SV signal generator, which is located in the top unit of the DSP architecture. The block diagram contains a small PRN code generator, which clock

frequency is generated by an 8-bit NCO. The “Modulo 20-Epoch Message Generator” acts as the SV communication channel. It inverts the whole small PRN code at a slow rate to indicate the ones or the zeros of digital data streams.

The block diagram of the small PRN code generator is shown at Figure 4.24. The code sequence $G_s(t)$ is a linear pattern generated by the modulo-2 addition of sub-sequences, $G_{1,s}(t)$ and $G_{2,s}(t)$, each of which is a 31 chip long linear pattern. The input of the corresponding shift registers is determined with the following polynomial equations [Dix84]:

$$G_{1,s} = Z^{-5} \oplus Z^{-3}$$
$$G_{2,s} = Z^{-5} \oplus Z^{-4} \oplus Z^{-3} \oplus Z^{-2} \qquad (4:28)$$

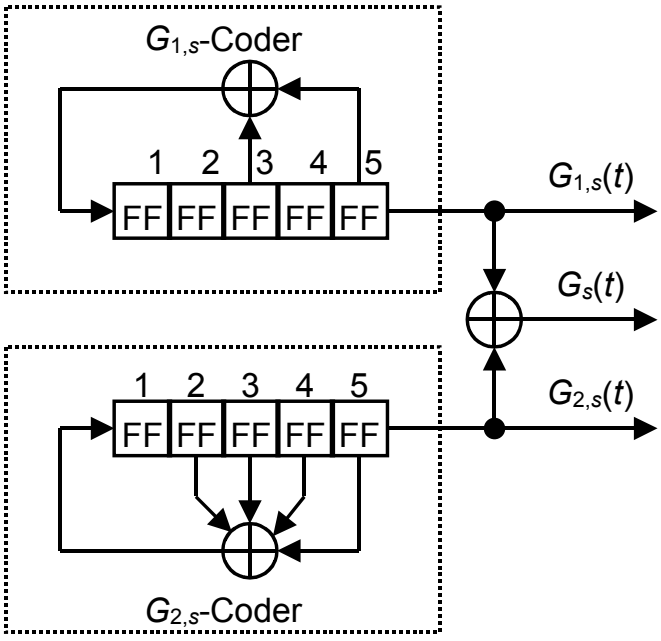


Figure 4.24: Small PRN code generator for Loop-Back test.

Beside the small PRN code, in-phase and quarter-phase carriers are also available. Therefore, the test of the carrier demodulation and the BitSync is possible as well. The frequency of the carrier can be set through the Test Carrier Increment Register (TCaIR). This 8-bit register contains the frequency increment for the 8-bit carrier NCO.

In all channels the small PRN code can be replicated with the standard PRN generator at the same frequency. Table 4.9 illustrates the Auto-

Correlation Function (ACF) of the small PRN code, in function of the phase offset between both codes.

Phase Offset [Chip]	Auto Correlation function	Phase Offset [chip]	Auto Correlation function
0	31	16	-9
1	7	17	-9
2	7	18	-9
3	7	19	-9
4	7	20	-9
5	-1	21	-1
6	7	22	-1
7	-1	23	-1
8	-1	24	-1
9	-1	25	7
10	-1	26	-1
11	-9	27	7
12	-9	28	7
13	-9	29	7
14	-9	30	7
15	-9	31	31

Table 4.9: Auto correlation function of the small PRN, in function of the delta phase.

If both codes are in phase, the accumulated value *ACC* in the accumulator registers will be:

$$ACC = \frac{ACF \cdot f_{clk}}{f_{PRN,s} \cdot Q} = \frac{31 \cdot 256}{24 \cdot 4} \approx 84 \qquad (4:29)$$

In the relationship above *ACF* and *Q* represent respectively the auto-correlation function and the accumulator quantification factor. In the contrary, when both code are not in phase, the accumulated value in the accumulator registers will be in the worth case:

$$ACC = \frac{-9 \cdot 256}{24 \cdot 4} \approx -24 \qquad (4:30)$$

In both cases, the energy of the signal corresponds approximately to the square of the accumulated signal. Therefore, the energy of the signal will correspond respectively to 7056 and 576. The predefined threshold for the signal presence determination is 1536. The threshold setting is achieved by writing -96 in the Search Reference Register (SRR), because the Search Reference value is shifted to the left by 4 bits in the DSP.

4.14 Summary & Conclusions of the Chapter

This chapter draws the design of a low power DSP architecture for GPS receivers, destined to small-volume portable applications. All the baseband applications together must not consume more than 70 mW.

The proposed architecture is a new combination of a “GPS message receiver” and a small microprocessor for the navigation processing. The DSP architecture, called IMT3050M12, has 12 independent channels. It performs satellite acquisition and tracking, without the help of an external circuitry. When a channel is locked onto a satellite, the circuit synchronizes the GPS signal and transmits the GPS message to the navigation unit. In addition, the IMT3050M12 records pseudo-ranges on request.

The bit-parallel architecture has been chosen for the DSP implementation, where each channel has its own dedicated processor. Each processor in turn is composed of a MU, an AU and a correlator. A FSM controls these units via dedicated bus network.

The computational structure of the acquisition and tracking algorithms is of the “sum of product” type. Since most algorithm steps should be accomplished in one step, the MU is composed of a four-port register file: three independent ports for the read operations and a fourth for the write operation. The AU involves a parallel 8×8-bit RBM and 20-bit RBA.

In the DSP top unit, the main clock "*CLK"* is divided by 16 to form another clock "*CLK16"* at a lower frequency rate. Both clocks are used in the correlator, whereas the remaining processor units are exclusively clocked with *CLK16*. Therefore, excellent power consumption characteristics are obtained.

The minimum *CLK* frequency is approximately 3.5 MHz. Unfortunately, other considerations in the RF unit have defined the *CLK* frequency to equal 4.3625 MHz. Additional power savings could thus be obtained by further reducing the clock frequency.

4.15 References

[Arm93] J. –R. Armstrong & F. –G. Gray, *Structured Logic Design with VHDL*, pp 237-240, Prentice-Hall, 1993.

[Cha89] H. Chang, "Pre-sampling Filtering, Sampling and Quantification Effects on Digital Matched Filter Performance", Proceedings of the international telemetering conference (San Diego CA), pp 889-915, 1989.

[Die95a] A. -J. Van Dierendonck, "GPS Receivers", *Global Positioning System: Theory and Applications*, Vol. 1, pp 369-390, American Institute of Aeronautics and Astronautics Inc., 1996.

[Die95b] A. -J. Van Dierendonck, "GPS Receivers", *Global Positioning System: Theory and Applications*, Vol. 1, pp 395-396, American Institute of Aeronautics and Astronautics Inc., 1996.

[Die95c] A. -J. Van Dierendonck, "GPS Receivers", *Global Positioning System: Theory and Applications*, Vol. 1, pp 402-405, American Institute of Aeronautics and Astronautics Inc., 1996.

[Dix84] R. –C. Dixon, *Spread Spectrum Systems*, pp 79-84, John Wiley & Sons, 1984.

[Far2000] P. -A. Farine, "An Analogue Watch Including a Very Low Power GPS Receiver", *Proceedings of the International Technical Meeting of the Satellite Division of the Institute of Navigation*, ION GPS-2000.

[Hit97] Hitachi, *SH-7020/SH-7021 Product Brief*, pp 1-6, Hitachi Inc., 1997.

[Mit99] Mitel Semiconductor, *ARM60 Data Sheet*, pp Preface-ii, Copyright Mitel Corporation, 1999.

[Mot96] Motorola, *MC68331 User's Manual*, pp A-2, Copyright 1996 Motorola, Inc.

[Phi99] Philips Semiconductors, *Data Sheet XA-G3*, pp 22, Copyright Philips Electronics North America Corporation 1999.

Chapter 5

Realization & Test of a Low Power DSP Architecture for GPS Receivers

This chapter draws the realization and test of the low power DSP architecture described in the previous chapter.

The approach is based on Field Programmable Gate Array (FPGA) prototyping followed by an ASIC realization. In this manner, the circuit performances can be verified in real time before integration. It provides an efficient solution compared to a modeled approach, which would require very long simulation times.

After the presentation of the applied "top-down" synthesis methodology, the VHDL test bench modules are introduced, including the Loop-Back test system. Then, the DSP architecture characteristics are presented.

5.1 Design Flow

For the DSP architecture realization, a new top-down synthesis methodology has been developed and used, which principle is shown at Figure 5.1.

The design entry of the DSP architecture is defined at high level with SpeedCHART. After translation of the design, the VHDL description is first used to implement a prototype with 8 channels, using 4 Altera Flex10k100 FPGAs. With this set up, the low power DSP architecture is tested and validated in real time.

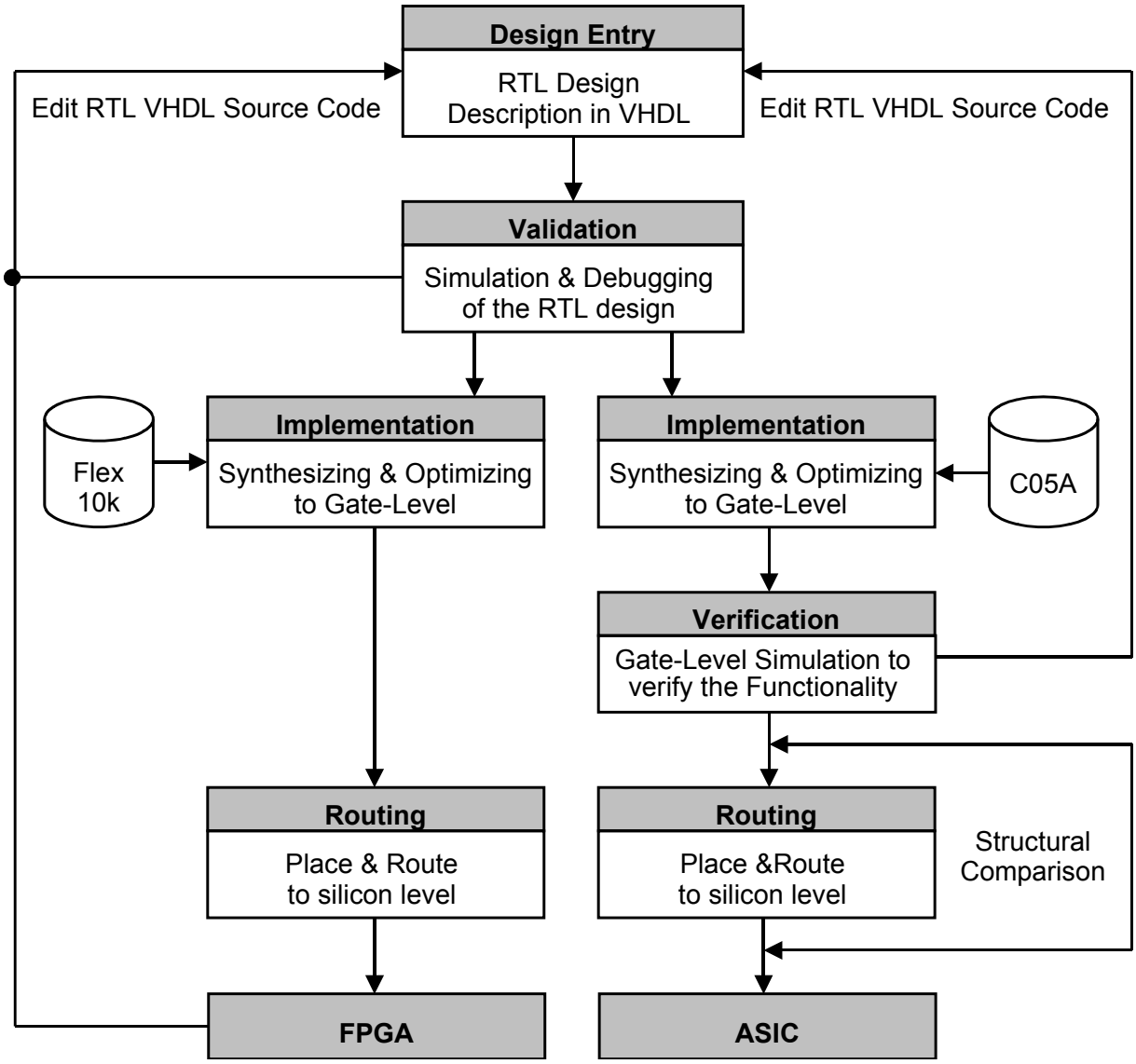


Figure 5.1: Design Flow.

Once the FPGA design meets the specifications, the synthesis of the ASIC begins. To minimize the error risks, the same VHDL description is taken as that one for the prototype realization. However, the ASIC design is extended to 12 channels. Since no errors are tolerated for an ASIC synthesis, a complete VHDL test bench is also necessary. The ASIC is simulated before and after synthesis.

5.2 Design Entry

The design has been edited at high level with SpeedCHART, which is a CAD editor using VHDL. It comprises a comprehensive suite of programs to support all phases of the design edition including a project, state and code designer. A stimuli editor, a simulator and a waveform viewer are also available for conceptual level simulation.

The design is represented with a graphical hierarchy of data flow diagrams. From this source, structural VHDL can be automatically generated. State machines and user specified VHDL descriptions might be created to describe the behavioral aspect of the design. Since a behavioral model can exist at the circuit, gate, register or chip level in a given design, different parts of the design can have their behavior specified at different levels. Figure 5.2 illustrates an example of this structural design decomposition. The design is represented as a tree, where different levels of the tree correspond to levels in the abstraction hierarchy (see section 3.1.2).

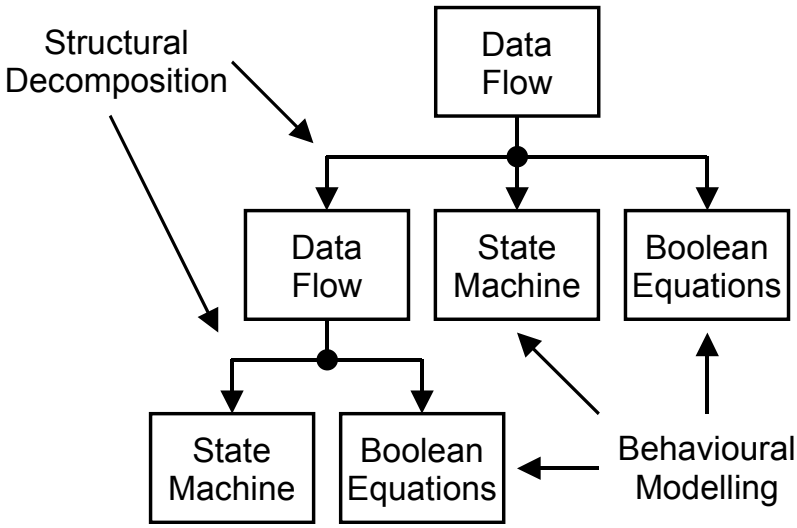


Figure 5.2: Structural design decomposition.

The tool generates a complete VHDL Model of the design, including the user generated VHDL. When compiled, this VHDL can be simulated to explore system behavior and used as the input to evaluate design architectures.

5.3 Logic Synthesis

Logic synthesis is the transformation from data flow representation to a structural logic gate representation. Both the FPGA and the ASIC logic synthesis have been realized with the Synopsys Design Compiler, which translates the VHDL description into a technology dependent gate-level design. During this procedure, the design is also optimized for speed, area and routing ability. The synthesis libraries were respectively the Flex10k and the C05A of EM Marin.

Design Compiler provides links to other CAD tools, such as place and route, and post layout re-synthesis techniques, such as in-place optimization. The synthesis flow follows this general process:

- *Reading the design and its sub-designs.*
- *Set design attributes on the top-level design.*
- *Set realistic timing or area goals for the design.*
- *Run the check_design to verify the design.*
- *Perform Design Compiler optimization.*
- *Run area and constraint reports to determine whether design goals are met.*
- *Re-optimize after modifying attribute or constraints if goals are not met.*
- *Run additional reports and schematics to analyze results further.*

5.3.1 ASIC Synthesis Results

Table 5.1 displays the ASIC synthesis result with the C05A library. This table demonstrates that only 25% of the total flip-flops are clocked with *CLK*. Therefore, approximately only the quarter of all transistors works at the full speed.

The remaining parts are clocked either with the *CLK16* or during the register writing operations. As the register access frequency is far less than *CLK16*, its contribution to the total power consumption may be neglected. Compared to the *CLK* part, the *CLK16* will approximately add 15% to the total average current.

Section 5.7.2 illustrates that the IMT3050M12 ASIC consumes about 20 mW at 3.3 Volt. Since, the ASIC contains most of the GPS receiver DSP algorithms. The power consumption characteristics are good, despite the common technology (e.g. 0.5-micron).

	Correlator	1 Channel GPS Receiver DSP (Without Correlator)	Top Unit	IMT3050-M12 (Total)
Combinational Cells	2262	5407	637	92665
DFF clocked with CLK	165	0	68	2048
DFF Clocked with CLK16	36	372	0	4896
Register DFF Clocked with NWR	0	87	44	1088
Combinational Area [mm^2]	0.476	1.279	0.178	21.238
Non-combinational Area [mm^2]	0.105	0.239	0.060	4.188
Total Cell Area [mm^2]	0.580	1.518	0.238	25,426

Table 5.1: Synthesis results.

The 8-bit CoolRISC816 micro-controller has been chosen for the navigation processing, which core consumes 230-micro Watt/MIPS. Being clocked at 4.3625 MHz, the processor core contributes only 1 mW to the total power balance.

The positioning software contains 48-kilo instructions and is stored in an AMD FLASH memory of 8 Mb as no smaller devices was available. The navigation processing data and constants are stored in a RAM and an E^2PROM, of 16 kB each. When the GPS is switched on, the navigation constants (e.g. almanac and ephemeris) are transferred from the E^2PROM into the RAM. On the contrary, when the GPS is switched off, the navigation constants are transferred back to the E^2PROM.

All the navigation units together consume about 50 mW, mainly due to the memory units (e.g. the consumption of the AMD FLASH memory is around 35 mW whereas the RAM and E^2PROM consume together 15 mW). The typical power consumption of the LNA, RF and IF is in the range of 30 mW [Far2000]. Therefore, the global power consumption remains below 100 mW, as targeted. These dissipation characteristics are excellent with respect to other receivers.

5.4 Place & Route

The place and route operation is the transformation from a logic gate representation into a layout representation (layout synthesis or routing). For the FPGA prototype version, this operation was realized with MAX+Plus II. Once the prototype was characterized, different versions of the ASIC were integrated:

- *The ICORREL is an IMT3050M12 version, which was integrated by IMEC.*
- *The TCOR1B is an IMT3050M12 version, which was integrated by TSMC.*
- *The TCCO1B is a composition of the DSP architecture and the CoolRISC, which was integrated by TSMC.*

All ASIC circuits have the same technology specific gate–level description. However, the IMEC and the TSMC circuits were routed respectively with Compass V9.0 and with Silicon Ensemble V5.1 (Cadence).

5.4.1 MAX+Plus II

The Altera's MAX+PLUS II development tool provides a complete FPGA design environment that is easily adaptable to all specific designs needs. Regardless of whether a personal computer or a workstation, MAX+PLUS II ensures easy design entry, fast processing and straightforward device programming.

MAX+PLUS II software is a fully integrated, architecture-independent package for designing logic with Altera programmable logic devices. MAX+PLUS II offers a full spectrum of design capabilities:

- *Three design entry methods for hierarchical designs.*
- *Floor plan editing.*

- *Powerful logic synthesis.*
- *Design partitioning.*
- *Functional, timing and board level type simulations.*
- *Detailed timing analyses.*
- *Automatic error location.*
- *Device programming and verifications.*

The design entry format was a standard edf net-list. Next the Compiler achieves the silicon implementation of the project. Finally the binary output file can be used to program the FPGA.

5.4.2 Compass

Compass design automation tools v9.2 provide a broad of tools for gate array, standard cell and full custom chip design. According to their function they are divided into six families:

- *The Design Specification tools.*
- *The Design Verification tools.*
- *The Basic tools.*
- *The Physical Design tools.*
- *The Design Test tools.*
- *The Physical Verification tools.*

For the IMEC design routing the Chip Compiler V9.5 was used, which belongs to the Physical Design family tools. The Chip Compiler is an integrated block/standard cell placement and routing system with a floor-planning stage and an automatic floor-plan estimator. The Chip Compiler offers:

- *Channel compaction for two-layer metal routing.*
- *Two-layer and poly routing and optional three layer metal routing.*
- *Over The Cell (OTC) routing.*
- *Timing driven-placement.*
- *File generation for timing verifications.*

The design entry format was a standard verilog net-list. This net-list must be converted into a specific Compass net-list description (e.g. nls file). Next the Chip Compiler places and routes the design.

5.4.3 Cadence

For the TSMC design routing the Silicon Ensemble V5.1 was used. This tool is a complete place and route system for standard cell design. It is designed to produce single-pass, timing-correct, 2 to N-layer designs consisting of any combinations of blocks, data-paths and cells. Based on the production-proven foundation of cell3 Ensemble technology, Silicon Ensemble is embedded with leading technologies, which enables designers to tackle Deep Sub-Micron (DSM) and System On Chip (SOC) design challenges. The design entry format was a standard verilog net-list.

5.4.4 ASIC Place and Route Results

Figure 5.3 illustrates the TCCO1B integrated by TSMC. This IC contains, the low power DSP architecture, the CoolRISC micro-controller, 16 kB integrated SRAM memory and the needed peripherals of the microprocessor (parallel and serial interfaces, I^2C, interrupt controller etc.). The transistor count implemented on this CMOS IC is roughly 1.9 million, of which 1 million are attributable to the low power DSP architecture. The total circuit measures 8.99×7.68 mm^2, including the pad ring.

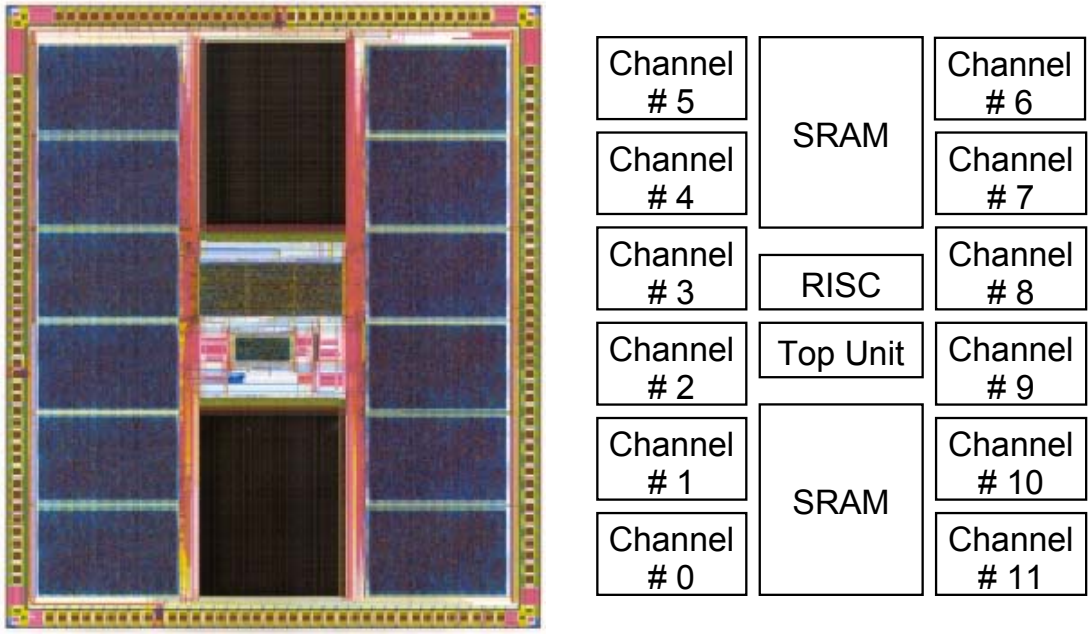


Figure 5.3: TCCO1B integrated circuit in CMOS 0.5-micron technology.

5.5 Verification

Since no errors are tolerated during an ASIC synthesis, the ASIC must be controlled at each hierarchy level. The simulation environment was the Synopsys VSS, which consists of before and after synthesis simulations. Furthermore, the post synthesis simulations have been executed with and without delay back-annotation. However, no simulations have been executed after the ASIC routing, because the technology libraries were not available. As an alternative, a structural comparison of the design before and after routing has been realized. Note that the structural comparison is only valid for common technologies. For sub-micron technologies, simulations after routing are highly recommended.

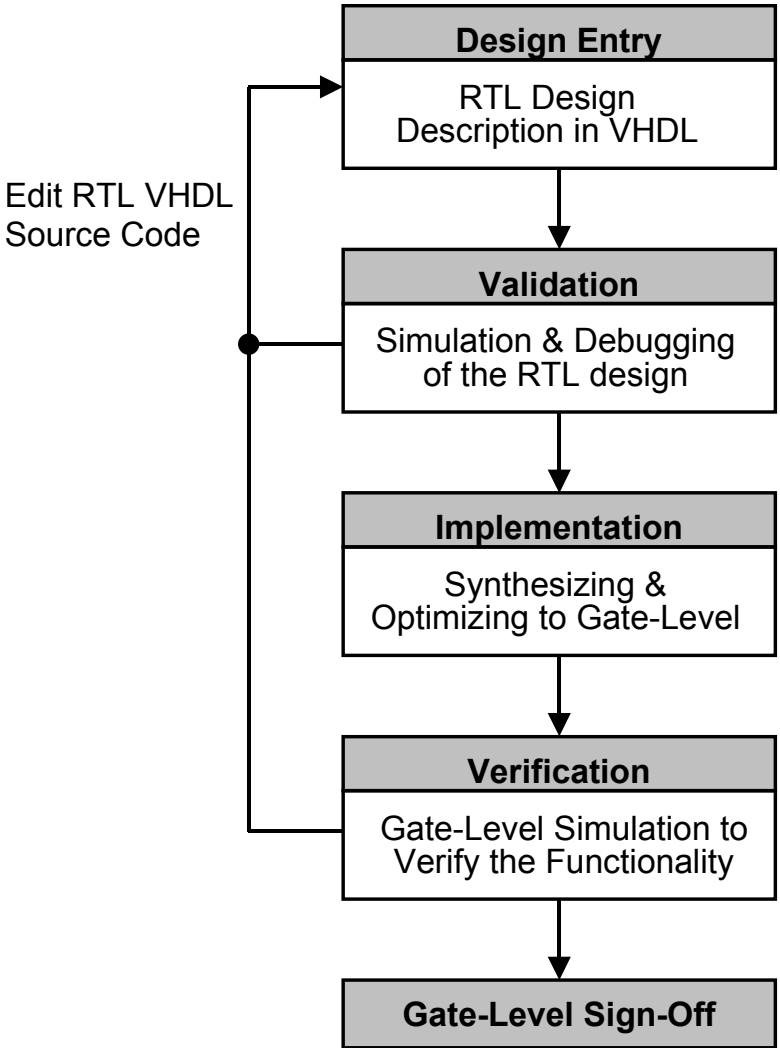


Figure 5.4: VSS in a top-down design methodology.

The Synopsys VSS simulator is a powerful simulator that can be used from the design level through ASIC sign-off. This simulator includes an optimized gate-level simulation engine, sign-off features, and interfaces to C-language models and other commercial simulators. These features allow to compile and to simulate descriptions written in IEEE VHDL 1076-1987. Synopsys VSS capabilities allow to perform source-level debugging, display the results of simulation and analysis of simulation results. Figure 5.4 shows how to use the Synopsys VSS simulator in a high-level (VHDL) design methodology.

5.6 VHDL Test Bench

The testing of the DSP architecture is an important but quite complicated task as many parts interact and twelve channels are to be verified. To solve this, a hierarchical test bench has been developed, which allows testing the functionality of each module individually, followed by a Loop-Back test system.

The complete test bench is composed of 11 module tests, which correspond to the different components of the DSP architecture. In addition, the Loop-Back test allows the test of the interconnection of the different modules. These tests have been described in VHDL for the Synopsys VSS simulator.

5.6.1 Module Tests

The general principle of the module tests is to set up the TMS pins at “01”. Next, specific values are written into the DSP registers. After this initialization, a certain number of clock flags are applied to the DSP architecture. Finally, the particular DSP register values are read back and verified against precomputed values. The test sequences are applied to all channels in parallel in order to speed up simulation times. Table 5.2 illustrates the module test acronyms and their succinct descriptions. An example of the accumulator test is given below, whereas a full description of the test benches can be found in [Fir2000].

Acronym	Succinct Test Description
ACC	Accumulator test
CA_NCO	Carrier NCO test
CHAN_DEC	Channel decoder test
CO_NCO	Code clock NCO test
INTRPT	Test of the Interrupt Control Unit
PRN_C	Standard PRN code generator test
PRN_L	Length test of the standard PRN code generator
PRN_P	Phase test of the standard PRN code generator
PRN_31	Small PRN code generator test
REC	Recording test of the PRN phase
REG	Read & write test of the registers

Table 5.2: Module test description.

5.6.2 Accumulator Test

The Test Signal Register (TSigR) allows fixing the up/down counters. First, the in-phase early and in-phase late accumulators are set to count up, whereas the quarter-phase early and the quarter-phase late accumulators remain idle. Next, 256 CLK cycles are generated. During the test process the In-Phase Early Register (IPER) and the In-Phase Late Register (IPLR) are controlled every 16 CLK clock cycles. These registers contain respectively the 8 MSB of the in-phase early and in-phase late accumulators. Therefore, the expected values should increment systematically by 4 units. When the in-phase early and the in-phase late accumulators have been tested successfully, the same tests are executed on the quarter-phase accumulators.

Figure 5.5 displays the accumulator tests before synthesis with the Synopsys VSS simulator.

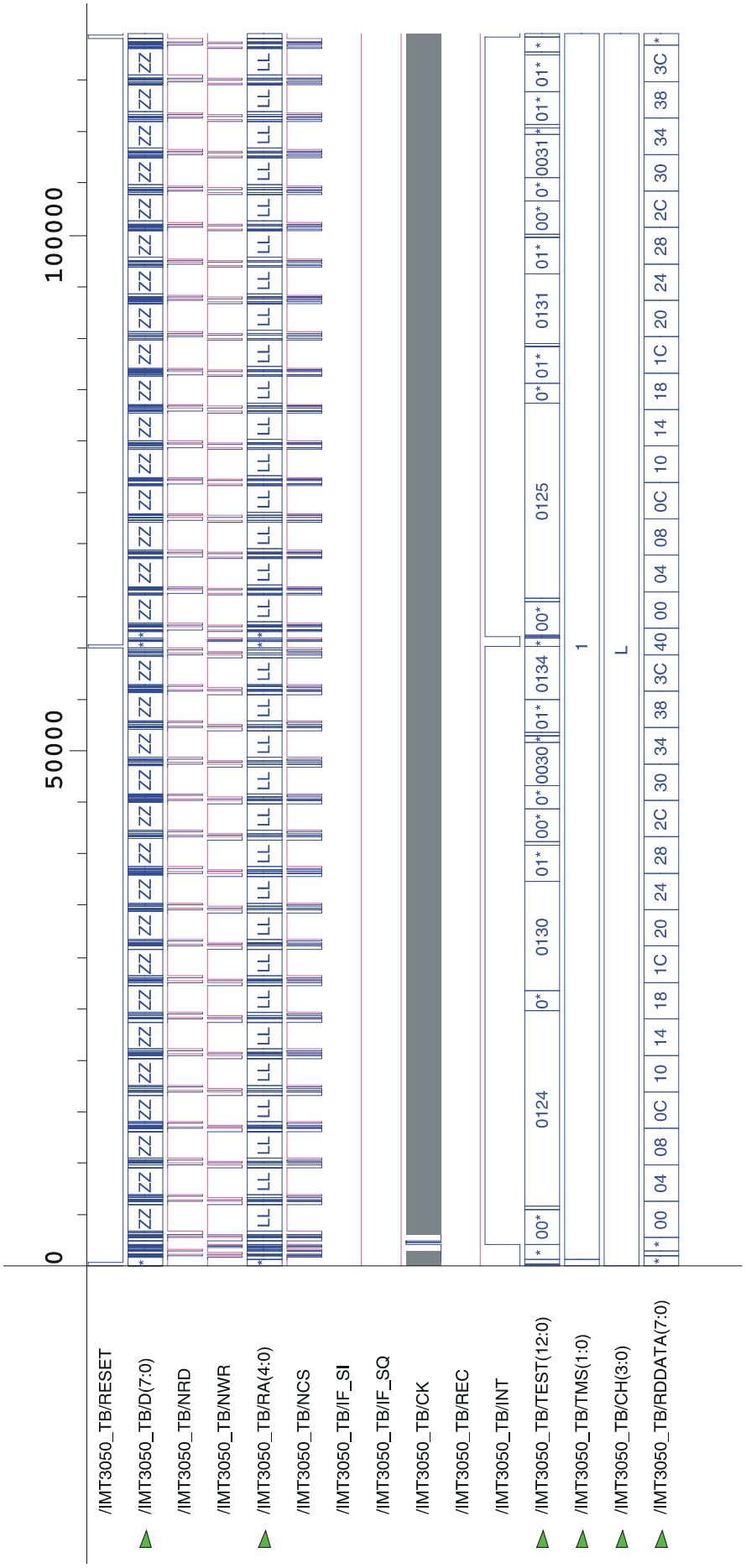


Figure 5.5: Before synthesis accumulator test with the Synopsys VSS simulator.

5.6.3 Loop-Back Test

When TMS is set to “11”, the integrity of the system can be tested without the need of a real GPS signal. This test can be realized at system boot and verifies whether the interaction between the sub-circuits operates as expected. It is assumed that the sub-modules have past the elementary tests successfully.

The major purpose of the Loop-Back test is to check the complete circuit operation near to real conditions in a small amount of time. Indeed, the use of real GPS signals would make the test set up impractical and would require too much time. Therefore, the length of the PRN code generator is reduced from 1023 to 31 chips. Section 4.13 illustrates the block diagram of the Loop-Back SV signal generator, which is located in the top unit of the DSP architecture.

Beside the small PRN code, in-phase and quarter phase carriers are also available. Therefore, the test of the carrier demodulation and the BitSync is possible as well.

In all channels the small PRN code can be replicated with the standard PRN generator at the same frequency. When the channel is started in the Loop back mode, it scans all possible chip phases of the small PRN code with a ½ chip resolution. When the correct phase has been found, the channel locks onto the signal while constantly adapting the internally generated carrier and small PRN code frequencies. As long as the signal strength is sufficient, the channel performs bit synchronization and outputs the GPS message.

5.7 ASIC Test Measurements

The different ASIC versions that have been tested are the following:

- *The ICORREL is an IMT3050M12 version, which was integrated by IMEC.*
- *The TCOR1B is an IMT3050M12 version, which was integrated by TSMC.*
- *The TCCO1B is a composition of the DSP architecture and the CoolRISC, which was integrated by TSMC.*

Both functional and dynamic tests have been realized on the circuits. They correspond respectively to the module and Loop-Back tests described above. The functional tests are used to check the functionality of the DSP modules, whereas the dynamic tests are destined to system level validation and consumption measurements.

5.7.1 Test Outlook

The hardware of the test outlook is composed by:

- *The ProTest test bench.*
- *Three PCB adapters realized by ASULAB.*
- *The Hewlett Packard 54645D numerical oscilloscope.*
- *An ampere meter.*
- *A personal computer.*

The software part is composed by:

- *The CAT-ProTest, which is designated to control the ProTest.*
- *The VSS VHDL simulator form Synopsys, which is used in the interactive mode to visualize the results graphically.*

ProTest Test bench

Figure 5.6 illustrates the ProTest test bench of Microlab (Engineering School Biel), which allows functional test of integrated ASIC circuits or FPGA.

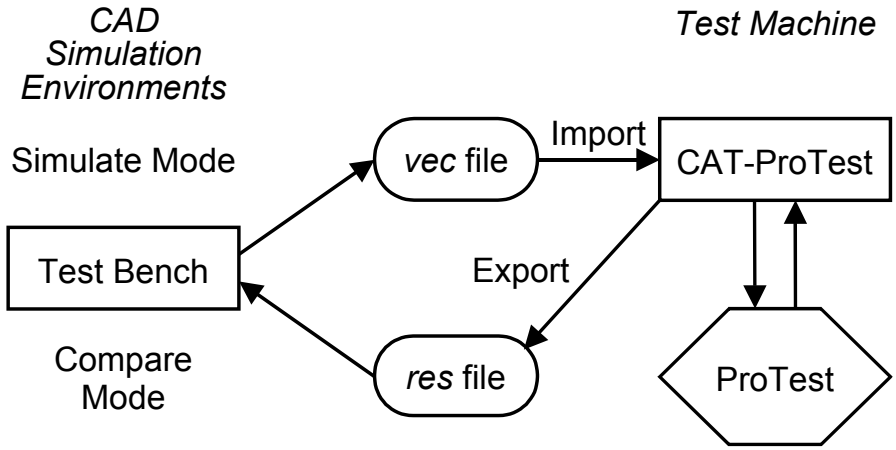


Figure 5.6: ProTest system [Mic98].

The test vectors can be translated from the VHDL test bench. The design flow is the following:

- *Development of the ASIC circuit and the VHDL test bench.*
- *Extension of the VHDL test bench for ProTest.*
- *Simulation and creation of the test vectors (.vec files).*
- *Creation of a test project and program with the CAT-ProTest tool.*
- *Test of the ASIC circuit and creation of a result file (.res files).*
- *Simulation (compare mode) and comparison with the test result.*

Figure 5.7 shows a ProTest sequence with several test cycles. Each of them is composed of input signal writing, a clock impulse generation, and result reading. The timing is respected only during one test cycle. As the system is particularly slow between two cycles, the test device was adapted to make it possible to generate several clock impulses during one ProTest test sequence.

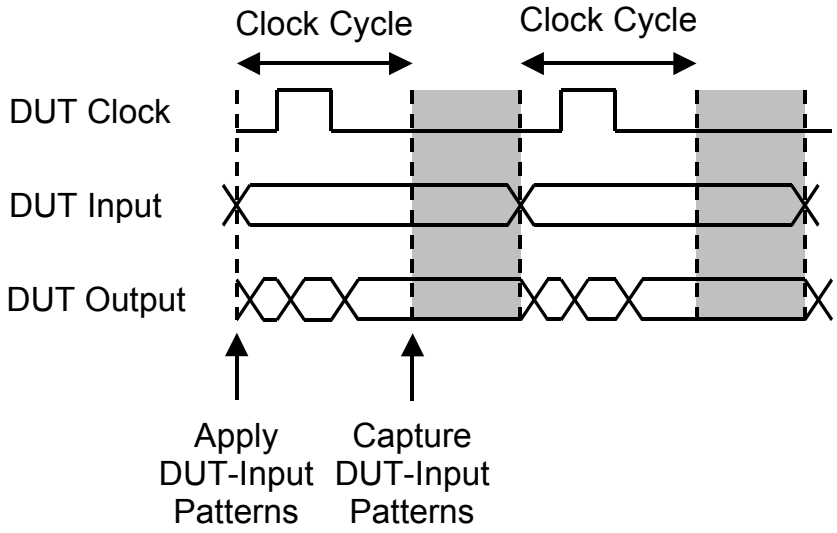


Figure 5.7: ProTest test cycles [Mic98].

A special hardware adapter was designed to enable the link between the ProTest and either an FPGA or an ASIC circuit under test. This adapter contains circuitry that is able to generate clock bursts used during the execution of the tests. A Flex10k100 is programmed either to act as a one channel IMT3050M12 or as a signal amplitude converter when used together with an ASIC.

5.7.2 Test Results

The principal test objective is to characterize the performances of the IMT3050M12 DSP. Five TCOR1B designs were tested completely. The aim of this first test series was design validation and consumption measurement.

Then, six ICORREL designs were tested in the Loop-Back mode. The aim of this second test series was to compare the power consumption of both technologies.

Finally one sample of the TCCO1B was tested to validate the GPS baseband receiver of the combined circuit.

TCOR1B

Table 5.3 displays the test results. Circuit 3 passes all tests, but circuit 5 does not function at all. The remaining circuits pass the tests partially.

Detailed investigation shows that the NCO test does not pass correctly. The reason may reside in the fact that the *MX-NCO* signal is not correctly synchronized at the input of the correlator. This flag enables the overflow of the LSBA to be shifted and added to MSBA and it is possible that the NCO phase accumulation is not realized correctly.

The appearance of this kind of problem can be avoided with a more defensive approach during the design of the control signals. Another solution would be accurate simulations after place and route.

Circuit 3 shows the lowest current consumption in the Loop-Back mode, which is 5.92 mA. Since, the Loop-Back DSP algorithms are the same than the standard ones, it is possible to assert that the TCOR1B will consume 20 mW during regular operating mode. These measured power dissipation characteristics are good, despite the common technology (e.g. 0.5-micron).

Acronym	Circuit # 1	Circuit # 2	Circuit # 3	Circuit # 4	Circuit # 5
ACC	OK	OK	OK	OK	Not OK
CA_NCO	OK Except Channel 8	OK Except Channel 8	OK	OK Except Channel 1 & 8	Not OK
CHAN_DEC	OK	OK	OK	OK	Not OK
CO_NCO	OK Except Channel 8	OK Except Channel 8	OK	OK Except Chan.1 & 8	Not OK
INTRPT	OK	OK	OK	OK	Not OK
LB	OK Except Channel 6 & 8	OK	OK	OK Except Channel 6	Not OK
PRN_C	OK	OK	OK	OK	Not OK
PRN_L	OK	OK	OK	OK	Not OK
PRN_P	OK	OK	OK	OK	Not OK
PRN_31	OK	OK	OK	OK	OK
REC	OK Except Channel 8	OK Except Channel 8	OK	OK Except Channel 1 & 8	Not OK
REG	OK	OK	OK	OK Except Channel 1	Not OK
Consumption	7.48 mA @ 3.3V	5.93 mA @ 3.3V	5.92 mA @ 3.3V	8.57 mA @ 3.3V	75 mA @ 3.3V

Table 5.3: Test result of the TCOR1B.

ICORREL

Table 5.4 displays the test results of the six IMEC circuits. In this case, the Loop-Back test never functions completely. Furthermore, the power consumption of the IMEC series is a little bit more important than that of the TSMC series. The difference between the IMEC & TSMC performances can result either from the technology or from the layout differences, as both circuits have the same structural logic gate representation. However, the IMEC circuit has been routed with Chip Compiler V9.5 (Compass) and the TSMC with Silicon Ensemble V5.1 (Cadence).

Acro.	Circuit # 1	Circuit # 2	Circuit # 3	Circuit # 4	Circuit # 5	Circuit # 6
LB	OK Except Channel 2	Not OK	Not OK	OK Only Channel 0, 1 & 3	OK Only Channel 0, 2, 3 & 5	OK Except Channel 4
Cons.	7.02 mA @ 3.3 V			6.76 mA @ 3.3 V	6.90 mA @ 3.3 V	6.88 mA @ 3.3 V

Table 5.4: Test result of the ICORREL.

TCCO1B

Table 5.5 displays the test result obtained for the TCCO1B circuit where only the GPS base-band DSP has been tested. These tests validate the separation of both subsystems. This circuit works correctly with the same exceptions observed for the TCOR1B circuit. However, there are other channels, which generate difficulties with the NCO. This can occur from a different channel placement around the top unit. Note that the consumption is bigger than the consumption of the TCOR1B.

Acronym	TCCO1B # 1
ACC	OK
CA_NCO	OK Except Channel 3
CHAN_DEC	OK
CO_NCO	OK Except Channel 3
INTRPT	OK
LB	OK Except Channel 6
PRN_C	OK
PRN_L	OK
PRN_P	OK
PRN_31	OK
REC	OK Except Channel 3
REG	OK
Consumption	8.20 mA @ 3.3V

Table 5.5: Test result of the TCCO1B.

5.8 Summary & Conclusion of the Chapter

This chapter shows the realization and the test of the DSP architecture based on a top-down design methodology. FPGA prototyping is used to validate and to test the design before the implementation of an ASIC. VHDL test benches are intensively used to verify the design at different stages. An original Loop-Back test is included that allows autonomous fast system tests without the need for external RF circuitry.

The test results illustrate that the design of the IMT3050M12 is correct. However, some timing problems may appear in the control signals of certain channels, probably due to routing capacitances.

The consumption of the IMT3050M12 is around 20 mW. Since the DSP architecture executes most of the GPS signal processing algorithms, the power consumption is good despite the common technology (0.5-micron). Adding the navigation processing, the baseband power consumption remains below 70 mW. The ASIC – containing the low

power DSP architecture, the CoolRISC micro-controller and the 16-kB SRAM - measures 8.99×7.68 mm^2 in a quite old technology. Further power reductions are therefore possible. Anyhow, the receiver is already amongst the most economic commercial GPS receivers today.

5.9 References

[Far2000] P. -A. Farine, “An Analogue Watch Including a Very Low Power GPS Receiver”, *Proceedings of the International Technical Meeting of the Satellite Division of the Institute of Navigation*, ION GPS-2000.

[Fir2000] E. Firouzi, V. Moser & R. Riem-Vis, “Description of the Hardware Design and Test Methodology”, *A Low Power GPS Receiver Chip,* Internal Report Document ASU4UM-2000.

[Mic98] http://www.microlab.ch/industry/products/protest/

Chapter 6

Conclusions

This thesis draws the design and the realization of a low power DSP architecture for GPS receivers, destined to small-volume portable applications.

The volume constraints result in reduced antenna and battery sizes. Therefore, the DSP should have excellent consumption characteristics without important SNR losses.

In this thesis architectural and algorithmic optimizations are presented, which have an important impact on both the power consumption and the signal processing efficiency. This chapter concludes the thesis with a summary of the most important results.

6.1 Main Contributions

The main contributions of this thesis concern either architectural or algorithmic optimizations of the GPS receiver DSP. They improve both the power dissipation and the signal processing efficiency.

6.1.1 Chapter 3

The chapter 3 contains two examples of algorithmic optimizations. Both concern the power consumption in standard GPS receivers.

The first is the complex sampling algorithm, which limits the sampling frequency to the information bandwidth. In this manner, the power dissipation of standard correlator can be reduced almost by half.

The second is a multiphase NCO, which has the same frequency characteristics than a standard NCO but half of its power consumption. The accumulator splitting adds quantification noise in the tracking loops that can be maintained bellow the thermal noise.

6.1.2 Chapter 4

The chapter 4 draws the design of a low power DSP architecture for GPS receivers, destined to small-volume portable applications.

The proposed architecture is a combination of a “GPS message receiver” and a small microprocessor for the navigation processing. The DSP architecture, called IMT3050M12, has 12 independent channels and performs satellite acquisition and tracking without the help of an external circuitry. When a channel is locked onto a satellite, the circuit synchronizes the GPS signal and transmits the GPS message to the navigation unit. In addition, the IMT3050M12 records pseudo-ranges on request.

The bit-parallel architecture has been chosen for the DSP implementation, where each channel has its own dedicated processor. Each processor in turn is composed of a MU, an AU and a correlator, communicating via a dedicated bus network.

In the DSP top unit, the main clock “*CLK”* is divided by 16 to form another clock “*CLK16”* at a lower frequency rate. Both clocks are used in the correlator, whereas the remaining processor units are exclusively clocked with *CLK16*. Therefore, excellent power consumption characteristics are obtained.

The minimum required *CLK* frequency is approximately 3.5 MHz. Unfortunately, other considerations in the RF unit have defined the *CLK* frequency to equal 4.3625 MHz. Additional power savings could thus be obtained by further reducing the clock frequency.

6.1.3 Chapter 5

This chapter draws the realization and the test of the DSP architecture described in chapter 4, based on a top-down design methodology. FPGA prototyping is used to validate and to test the design before the implementation of an ASIC. VHDL test benches are intensively used to verify the design at different stages. An original Loop-Back test is included that allows autonomous fast system tests without the need for external RF circuitry.

The test results illustrate that the design of the IMT3050M12 is correct. However, some timing problems may appear in the control signals of certain channels, probably due to routing capacitance.

The consumption of the IMT3050M12 is around 20 mW. Since the DSP architecture executes most of the GPS signal processing algorithms, the power consumption is good despite the common technology (0.5-micron). Adding the navigation processing, the baseband power consumption remains below 70 mW. The ASIC - containing the low power DSP architecture, the CoolRISC micro-controller and the 16-kB SRAM - measures 8.99×7.68 mm^2 in a quite old technology. Further power reductions are therefore possible. Anyhow, the receiver is already amongst the most economic commercial GPS receivers today.

6.2 Final Remarks

The low power consumption and the small size of the baseband unit are compatible with the requirements of a normal sized watch. Therefore, Asulab (the R&D Laboratories of the Swatch Group) has integrated the baseband IC in an analogue wristwatch (Figure 6.1).

The GPS analogue watch is characterized by low power consumption and with dimensions slithery smaller than a Swatch Beat ($47 \times 57 \times 20$ mm^3). The antenna is a small rectangular patch antenna with reduced size ($13 \times 13 \times 4$ mm^3) and is located under the “12 hour” location in the wristwatch dial. The current consumption is in the range of one μA in the watch mode and less than 40 mA in the GPS mode.

Figure 6.1: GPS analogue watch [Far2000].

6.3 References

[Far2000] P. -A. Farine, “An Analogue Watch Including a Very Low Power GPS Receiver”, Proceedings of the International Technical Meeting of the Satellite Division of the Institute of Navigation, ION GPS-2000.

Annexe

Notations & Abbreviations

The notations and abbreviations, used throughput this thesis, are listed in this sections in function of the chapter. A same letter may unfortunately have different meaning. However, the context and additional explanations in the text should avoid confusion.

A.1. Notations

A.1.1. Scalars

		Chapter
A_0	SV signal amplitude……	2
B	Single side bandwidth……	2
B_L	Loop noise bandwidth……	2
$B_{L,i}$	B_L where ($i \Rightarrow$ *DLL*, *FLL* or *PLL*)……	2
C/N_0	Carrier to noise ratio……	2
d	Correlator spacing……..	2
Δf	Frequency offset (Doppler etc.)……	2
Δf_i	Frequency offset at time t_i……	2

P_{fa}	False alarm probability	2
P_n	Noise power	2
P_s	Signal power	2
P_S	Total static power dissipation	3
P_{SC}	Short current power dissipation	3
P_{total}	Total power dissipation	3
Q	Quantification factor	3
Q_A	Allan variance-induced oscillator jitters	2
$Q_{A,i}$	Q_A where ($i \Rightarrow$ 2 or 3) represents the PLL order	2
θ_e	PLL dynamic stress errors	2
R_e	DLL dynamic stress errors	2
SNR	Signal to Noise Ration	2
S_n	Pseudo-random sequence	2
$\sigma_A(\tau)$	Allan deviation	2
σ_i	RMS phase errors causes from all sources where ($i \Rightarrow$ *DLL*, *FLL* or *PLL*)	2
σ_n	RMS noise power of the accumulator	2
σ_Q	RMS of the multiphase NCO quantification noise	3
$\sigma_{t,i}$	RMS of the tracking loop thermal noise where ($i \Rightarrow$ *DLL*, *FLL* or *PLL*)	2
T	Post-correlation integration time	2
T_c	PRN code chip width	2
TH	Threshold	2
T_s	Sampling period	2
τ	Phase offset between two replicas PRN codes	2
V_{TH}	Threshold voltage	3
ω_B	Pseudo baseband carrier circular frequency ($\omega_B = 2\pi \cdot f_B$)	3
ω_{IF}	IF carrier circular frequency ($\omega_{IF} = 2\pi \cdot f_{IF}$)	2
ω_L	Loop natural gain	2
$\omega_{L,i}$	ω_L where ($i \Rightarrow$ *DLL*, *FLL* or *PLL*)	2
ω_0	SV carrier circular frequency ($\omega_0 = 2\pi \cdot f_0$)	2
x_k	In-phase noise sample at time t_k	2

A.1.2. Signals

$I_{2,sk,j}$	In-phase signal after the carrier and code stripping process sampled at time t_k	2
L_1	Primary ranging SV signal (1575.42 MHz)	2
L_2	Secondary ranging SV signal (1227.6 MHz)	2
M_E	Number of accumulated samples	2
Mult1	Primary MAC multiplicand	4
Mult2	Secondary MAC multiplicand	4
MX_NCO	Enables the overflow of the LSBA to be shifted and added to MSBA	4
$n(t)$	Noise representation in time domain	2
P(Y)	Precise code	2
$p(t)$	Rectangular unit pulse	2
$PN(t)$	PN sequence	2
$PRN(t)$	PRN sequence	2
Q_i	Accumulated quarter-phase signal plus noise after the carrier and code stripping process sampled at time t_i	2
Q_{ni}	Accumulated quarter-phase noise after the carrier and code stripping process sampled at time t_i	2
Q_{nk}	Quarter-phase IF noise sampled at time t_k	3
$Q_n(t)$	Quarter-phase IF noise in the time domain	3
Q_{si}	Accumulated quarter-phase signal after the carrier and code stripping process sampled at time t_i	2
Q_{sk}	Quarter-phase IF signal sampled at time t_k	3
$Q_s(t)$	Quarter-phase IF signal in the time domain	3
$Q_{1,nk}$	Quarter-phase base band noise sampled at time t_k	2
$Q_{1,sk}$	Quarter-phase base band signal sampled at time t_k	2
$Q_{2,sk,j}$	Quarter-phase signal after the carrier and code stripping process sampled at time t_k	2
Sub/Add	Defines the sign of the internal sum	4
Sum	MAC output	4
$S_n(f)$	Noise representation in the frequency domain	2
$S_s(f)$	SV signal representation in the frequency domain	2
$s(t)$	SV signal representation in the time domain	2
$S(t)$	Spreading waveform	2

$x_a(t)$	Analogue signal in the time domain……	2
$X_a(\omega)$	Analogue signal in the frequency domain……	2
$x_q(k)$	Digital signal in the time domain……..……	2
$x(k)$	Discrete time signal in the time domain…….……	2
$X(\omega)$	Discrete time signal in the frequency domain……	2

A.1.3. Functions

$H(s)$	Transfer function……	2
$I_K(x)$	Modified Bessel function of the first kind……..	2
$P_N(y)$	General PDF of the noise power……	2
$P_{N,K}(y)$	$P_N(y)$ where ($K \Rightarrow$ number of iteration)……	4
$P_{S,}(y)$	General PDF of the signal plus noise power……	2
$P_{S,K}(y)$	$P_{S,}(y)$ where ($K \Rightarrow$ number of iteration)……	4
$R(\tau)$	Auto-correlation function……..	2

A.2. Abbreviations

ACF	Auto-Correlation Function…….	4
ADC	Analogue to Digital Converters…….	2
ADO	Accumulate and Dump Occurred…….	4
AGC	Automatic Gain Control……..	2
AS	Anti Spoofing……	2
ASIC	Application Specified Integrated Circuit……..	1
AU	Arithmetic Unit……..	4
A/D	Analogue to Digital……..	2
BER	Bit Error Rate……	2
BKA	Brent & Kung Adder……	4
BPSK	Bi-Phase Shift Key……..	2
CAD	Computer Assisted Development…….	3
CCU	Correlator Control Unit……	4
CDMA	Code Division Multiple Access……..	2
CMOS	Complementary Metal Oxide Silicon……	3

CNR	Carrier to Noise Ratio	2
CPA	Carry Propagate Adder	4
CPU	Computer Programmable Unit	4
CSA	Carry Save Adder	4
C/A	Coarse Acquisition	2
DGPS	Differential GPS	2
DLL	Delay Locked Loop	2
DSM	Deep Sub Micron	5
DSP	Digital Signal Processing	1
DS-SS	Direct Sequence - Spread Spectrum	2
EML	Early Minus Late	2
EPL	Early, Punctual and Late	2
FH-SS	Frequency Hop - Spread Spectrum	2
FLL	Frequency Locked Loop	2
FPGA	Field Programmable Gate Array	5
FSM	Finite State Machine	4
GLONASS	Global Navigation Satellite System	1
GNSS	Global Navigation Satellite System	1
GPS	Global Positioning System	1
HDL	Hardware Description Language	3
HDOP	Horizontal Dilution Of Precision	1
IC	Integrated Circuit	2
ICAO	International Civil Aviation Organisation	1
ICU	Interruption Control Unit	4
IF	Intermediate Frequency	2
IFU	Intermediate Frequency Unit	2
ITS	Intelligent Transportation Systems	1
LNA	Low Noise Amplifier	2
LO	Local Oscillators	2
LORAN	Long Range Radio Navigation	2
LSB	Least Significant Bit	4
LSBA	Least Significant Bit Accumulator	1

MBA	Modified Both Algorithms	4
MDD	Multiple D-well Detector	2
MIPS	Mega-Instruction Per Second	5
MNP	Mean Noise Power	4
MOS	Metal Oxide Silicon	3
MSB	Most Significant Bit	4
MSBA	Most Significant Bit Accumulator	1
MU	Memory Unit	4
NCO	Numerical Controlled Oscillator	1
NNSS	Navy's Navigation Satellite System	2
OCS	Operation Control Segment	2
OTC	Over The Cell	5
PDF	Probability of Density Function	2
PLL	Phase Locked Loop	2
PN	Pseudo Noise	2
PPS	Precise Positioning Service	1
PRN	Pseudo Random Noise	2
P(Y)	Precise	2
RAM	Random Access Memory	4
RB	Redundant Binary	4
RBA	Redundant Binary Adder	4
RBM	Redundant Binary Multiplier	4
RF	Radio Frequency	2
RFI	RF Interference	2
RISC	Reduced Instruction Set Computer	4
RMS	Root Mean Scare	2
ROM	Read Only Memory	4
SA	Selective Availability	1
SAW	Surface Acoustic Wave	2
SNR	Signal to Noise Ratio	1
SOC	System On Chip	5
SPS	Standard Positioning Service	1

SSP	Software Signal Processing……..	2
SV	Space Vehicle……..	1
TOA	Time Of Arrival…….	1
VCO	Voltage Controlled Oscillator…….	3
VDOP	Vertical Dilution Of Precision…….	1
VHDL	VHSIC Hardware Description Language…….	3
VOR	VHF Omni-directional Radios……	2