IMPLEMENTATION OF SOFTWARE GPS RECEIVER

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

EZGİ GÜNAYDIN

IN PARTIAL FULFILLMENT OF THE REQUIRMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

JULY 2005

Approval of the Graduate School of Natural and Applied Sciences.

_____

Prof. Dr. Canan Özgen

Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

_____

Prof. Dr. İsmet Erkmen

Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

_____

Assoc. Prof. Dr. T.Engin Tuncer

Supervisor

Examining Committee Members

Prof. Dr. Mübeccel Demirekler (Chairman),(METU, EE) _____

Assoc. Prof. Dr. T.Engin Tuncer, (METU, EE) _____

Assoc. Prof. Dr. Tolga Çiloğlu, (METU, EE) _____

Assist. Prof. Dr. Arzu Koç, (METU, EE) _____

Dr. Tolga Sönmez, (TÜBİTAK-SAGE) _____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name : Ezgi Günaydın

Signature :

ABSTRACT

IMPLEMENTATION OF SOFTWARE GPS RECEIVER

Günaydın, Ezgi
MS, Department of Electrical and Electronics Engineering
Supervisor: Assoc. Prof. Dr. T.Engin Tuncer
July 2005, 144 pages

A software GPS receiver is a functional GPS receiver in software. It has several advantages compared to its hardware counterparts. For instance, improvements in receiver architecture as well as GPS system structure can be easily adapted to it. Furthermore, interaction between nearby sensors can be coordinated easily. In this thesis, a SGR (software GPS receiver) is presented from a practical point of view. Major components of the SGR are implemented in Matlab environment. Furthermore, some alternative algorithms are implemented. SGR implementation is considered in two main sections namely a signal processing section and a navigation section. Signal processing section is driven by the raw GPS signal samples obtained from a GPS front-end of NordNav^{TM} R-25 instrument. The conventional and the block adjustment of synchronizing signal (BAAS) processing methods are implemented and their performances are compared in terms of their speed and outputs. Signal processing section outputs raw GPS measurements and navigation data bits.

Since the output data length is insufficient in our case, navigation section input is fed from Ashtech™ GPS receiver for a moving platform and Trimble™ GPS Receiver for a stationary platform. Satellite position computation, pseudorange corrections, Kalman filter and LSE (least squares estimation) are implemented in the navigation section. Kalman filter and LSE methods are compared in terms of positioning accuracy for a moving as well as a stationary platform. Results are compared with the commercial GPS outputs. This comparison shows that the software navigation section is equivalent to the commercial GPS in terms of positioning accuracy.

Keywords: GPS receiver, GPS signal, spread spectrum modulation, pseudorandom code, signal acquisition, signal tracking, pseudorange, Doppler frequency, LSE (least squares estimation), Kalman filter

ÖZ

YAZILIM ORTAMINDA GPS ALICISI TASARIMI

Günaydın, Ezgi
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Doç. Dr. T.Engin Tuncer
Temmuz 2005, 144 sayfa

Yazılım ortamındaki GPS alıcısının, donanımsal GPS alıcısına gore pek çok avantajları vardır. Örneğin sistemdeki veya alıcı tasarımındaki güncellemeler kolayca uygulanabilir. Bunun yanı sıra, çevre algaçlarıyla koordinasyonu kolaylıkla sağlanabilmektedir. Bu tezde, yazılım ortamında GPS alıcısı geliştirilmesi ile ilgili detaylı inceleme ve araştırma sunulmaktadır. GPS alıcısının temel bileşenleri Matlab ortamında gerçeklemiştir. Alternatif yöntemler greçeklenip birbirleriyle karşılaştırılmışlardır. Bu çalışma ile hazırlanan GPS alıcısı programı iki ana kısımdan oluşmaktadır. Birinci kısım ham GPS sinyalinin işlendiği birimdir. Ham ve örneklenmiş GPS sinyali, NordNav^{TM} R-25 GPS ön basamağından sağlanmıştır. Geleneksel ve yazılım ortamı için hazırlanmış BAAS (blok ayarlamalı sinyal senkronizasyonu) yöntemleri hazırlanmıştır. Bu iki yöntemin çıktıları ve hızları karşılaştırılmıştır. Sinyal işleme birimi, GPS ölçümlerinin oluşturulması ve seyrüsefer bitlerinin yapılandırılması ile yükümlüdür. Sinyal işleme kısmının veri uzunluğu yerli olmadığı için, seyrüsefer algoritmaları için girdiler ticari GPS alıcılarından toplanmaktadır. İkinci kısım alıcının pozisyonunun ve hızının tahmin edildiği

seyrüsefer birimidir. Hareketli durum için Ashtech™ GPS alıcısı, durağan durum için Trimble™ GPS alıcısı kullanılmıştır. Seyrüsefer birimi, uydu konumunun hesaplanmasını, sözde-uzaklığın düzeltilmesini, Kalman filtre ve LSE (en küçük kareler toplamı) yöntemlerini içermektedir. Pozisyon ve hız hesaplaması ancak ve ancak görülen uydu konumlarının bulunması ve ölçüm-zamanlama düzeltimlerinin yapılması ile mümkündür. Hazırlanmış olan bütün GPS alıcısı programı performansı ve sonuçları elde edilmiştir. Sonuçlar gerçek GPS alıcısının çıktıları ile karşılaştırıldığında, alıcı programının uygun bir şekilde çalıştığını görülmüştür.

Anahtar Sözcükler: GPS alıcısı, GPS sinyali, yayılı izge modulasyonu, sözde-rastgele kod, sinyal yakalama, sinyal izleme, sözde-uzaklık, Doppler frekansı, En küçük kareler tahmin yöntemi, Kalman filtresi

ACKNOWLEDGEMENTS

I wish to express my special thanks to my supervisor Assoc. Prof. Dr. T. Engin Tuncer for his guidance, support and suggestions.

I also wish to thank to my colleagues at TÜBİTAK SAGE, especially to Dr. A. Pınar Koyaz and Dr. Tolga Sönmez for their support and advices.

Beside, I would like to thank to my family for their love and support. I also would like to thank to my friends, especially to Rahşan Erdoğan, Elvan Kahyaoğlu and Kadriye Tiryaki for their patient, encouragements and support.

Finally, the support provided by TÜBİTAK-SAGE to this thesis is greatly acknowledged.

TABLE OF CONTENTS

PLAGIARISM ..... **iii**

ABSTRACT ..... **iv**

ÖZ ..... **vi**

ACKNOWLEDGEMENTS ..... **viii**

TABLE OF CONTENTS ..... **ix**

LIST OF TABLES ..... **xiv**

LIST OF FIGURES ..... **xv**

CHAPTER.

1. INTRODUCTION **1**

1.1 Thesis objective **1**

1.1.1 Software GPS Receiver Concept 2

1.1.2 Reference Sources 2

1.1.3 GPS Receiver 3

1.1.4 Implemented Software GPS Receiver 4

1.1.4.1 Implemented Signal Processing Section 6

1.1.4.2 Implemented Navigation Section 8

1.2 Global Positioning System **9**

1.2.1 History 9

1.2.2 GPS Segments 9

1.2.2.1 Space Segment 9

1.2.2.2 Control Segment 11

1.2.2.3 User Segment 12

1.2.3 GPS Services 12

1.2.3.1 Precise Positioning Service 13

1.2.3.2 Standard Positioning Service 13

1.3 Thesis outline 13

2. GPS SIGNAL and RECEIVER 15

2.1. GPS Signal Structure 15

2.1.1. Spread Spectrum Modulation 16

2.1.1.1 Direct Sequence Spread Spectrum, DS-SS 17

2.1.1.2 Pseudo Random Codes 20

2.1.1.2.1 C/A Code Generator in GPS 21

2.1.1.3 Code Division Multiple Access Technique 22

2.1.2. GPS Signal 23

2.1.3. GPS Signal Content 26

2.2. GPS Receiver 28

2.2.1. GPS Antenna 28

2.2.2. RF stage (front end) 28

2.2.3. DSP module 29

2.2.4. Navigation Algorithms 30

3. DSP MODULE 32

3.1 Operation of the signal processing module 32

3.1.1 Signal Acquisition 33

3.1.2 Signal Tracking 34

3.1.3 Bit and Frame Synchronization 36

3.2 Conventional Signal Processing 36

3.2.1 Conventional Signal Acquisition Method 36

3.2.2 Conventional Signal Tracking 40

3.2.2.1 Phase Locked Loop 41

3.2.2.2 C/A Code Tracking Loop 41

3.2.2.3 Carrier Frequency Tracking Loop 45

3.2.2.3.1 Costas Loop 45

3.2.2.3.2 Frequency Locked Loop (FLL) 48

3.3 Block Adjustment of Synchronizing Signal (BAAS) Signal Processing 50

3.3.1 Acquisition by Circular Correlation 50

3.3.2 Fine Frequency Estimation 54

3.3.3 BAAS Tracking 58

3.3.3.1 BAAS Carrier Tracking Algorithm 58

3.3.3.2 BAAS Code Tracking Algorithm 59

4. GPS POSITIONING 64

4.1 GPS Navigation Data Structure and Content 64

4.2 GPS Measurements 68

4.2.1 Time of transmission (TOT) of GPS signal from the satellite 69

4.2.2 Satellite Position Calculation 72

4.2.3 Pseudorange..... 73

4.2.3.1 Atmospheric Delay Correction 75

4.2.3.1.1 Ionospheric Model..... 75

4.2.3.1.2 Tropospheric Model..... 78

4.3 GPS Equations.....78

4.3.1 Pseudorange..... 79

4.3.2 Doppler frequency..... 80

4.4 GPS Position and Velocity Estimation Methods.....81

4.4.1 Least Squares Estimation 82

4.4.2 Kalman Filter..... 84

4.4.2.1 Discrete Extended Kalman Filter (EKF)..... 85

5. SIMULATION RESULTS..... 90

5.1 Software Signal Processing Module Results 90

5.1.1 Raw GPS IF Signal..... 91

5.1.2 Software C/A code generator 93

5.1.3 Conventional Signal Processing Results 95

5.1.4 BAAS Signal Processing Results..... 104

5.1.5 Comparison of conventional and BAAS signal processing methods. 117

5.2 Software Navigation Algorithm Results.....118

5.2.1 Dynamic case 118

5.2.1.1 Pseudorange correction 119

5.2.1.2 Path experiment results 120

5.2.1.2.1 LSE results 120

5.2.1.2.2 Kalman Filter results 125

5.2.1.2.3 Kalman filter and LSE methods comparison 131

5.2.2 Stationary case 131

6. CONCLUSION 137

REFERENCES 142

LIST OF TABLES

Table 1. Minimum power levels of the GPS signals 26

Table 2. Common Delay Locked Loop Discriminators 44

Table 3. Common Costas Loop Discriminators 48

Table 4. Common frequency locked loop discriminators 50

Table 5. GPS Data Structures 65

Table 6. TLM and HOW words structures 65

Table 7. GPS Navigation Subframes 66

Table 8. Subframe-1 navigation parameters 67

Table 9. Subframe-2 navigation parameters 67

Table 10. Subframe-3 navigation parameters 68

Table 11. Candidate satellites information 91

Table 12. Acquisition results of the conventional algorithm for PRN id 11 98

Table 13. Acquisition results of the conventional algorithm for PRN id 20 102

Table 14. Acquisition results of the circular correlation algorithm for PRN id 11 109

Table 15. Acquisition results of the circular correlation algorithm for PRN 20 113

Table 16. Ashtech G12 Board GPS receiver Messages 118

Table 17. Pseudorange errors 120

Table 18. LSE method errors 122

Table 19. Kalman Filter method errors 127

Table 20. Trimble GPS receiver Messages 133

LIST OF FIGURES

Figure 1. GPS Receiver Stages 4

Figure 2. Implemented GPS Software Receiver 5

Figure 3. Code and carrier shifts in the GPS signal 7

Figure 4. GPS Satellite Constellation 10

Figure 5. GPS Signals 11

Figure 6. Power Spectral Densities in Spread Spectrum Technique 16

Figure 7. Transmission channel in spread spectrum communication 17

Figure 8. DS-SS transmitter and receiver ([22]) 18

Figure 9. Transmitted signals in time and frequency domains ([22]) 19

Figure 10. Received signals in time and frequency domains ([22]) 20

Figure 11. G1 and G2 generators 22

Figure 12. GPS Signal 23

Figure 13. GPS signal timings 25

Figure 14. L1 GPS signal and the noise levels 26

Figure 15. GPS Data structure 27

Figure 16. Signal acquisition and tracking flow diagram 30

Figure 17. GPS Navigation Algorithms 31

Figure 18. Conventional acquisition process 38

Figure 19. Conventional acquisition loop 38

Figure 20. Combination of the code and the carrier tracking loops 40

Figure 21. Basic PLL diagram 41

Figure 22. Conventional code tracking 42

Figure 23. Correlation function difference of the early and the late codes 44

Figure 24. Conventional carrier tracking by using Costas PLL 45

Figure 25. Phasor diagram for Costas PLL 47

Figure 26. Conventional carrier tracking by using FLL..... 49

Figure 27. Phasor diagram for FLL..... 49

Figure 28. Circular correlation process..... 53

Figure 29. Circular correlation acquisition loop 53

Figure 30. Correlation lag in unit of chip (case 1) 60

Figure 31. Correlation lag in unit of chip (case 2) 62

Figure 32. Correlation lag in unit of chip (case 3) 62

Figure 33. Timing representations of the GPS signal..... 70

Figure 34. Elevation and Azimuth angles..... 76

Figure 35. DSP module set-up of the implemented GPS receiver..... 90

Figure 36. A portion of the time domain GPS signal (first 1000 samples) 92

Figure 37. Power spectral density of the IF signal..... 92

Figure 38. 0.1 milliseconds duration of the 31st satellite C/A code..... 93

Figure 39. Auto-correlation of the 19th satellite C/A code 94

Figure 40. Cross-correlation of the 19th and 31st satellite C/A codes 94

Figure 41. Acquisition metric for the satellite with PRN 11 97

Figure 42. Frequency search of PRN 11 97

Figure 43. Code search of PRN 11 98

Figure 44. Navigation bits, output of the carrier tracking loop for PRN 11 99

Figure 45. Doppler frequency obtained from carrier tracking for PRN 11..... 99

Figure 46. C/A code delay obtained from code tracking for PRN 11 100

Figure 47. Acquisition metric for the satellite with PRN 20 101

Figure 48. Frequency search of PRN 20 101

Figure 49. Code search of PRN 20 102

Figure 50. Navigation bits, output of the carrier tracking loop for PRN 20 103

Figure 51. Doppler frequency obtained from carrier tracking for PRN 20 103

Figure 52. C/A code delay obtained from code tracking for PRN 20 104

Figure 53. Acquisition metric for the satellite PRN 15 105

Figure 54. Thresholds of a phase shift 107

Figure 55. Code search of PRN 11, C/A code step vs. correlation value 108

Figure 56. Frequency search of PRN 11, frequency step vs. correlation value 108

Figure 57. Prompt channel output, i.e. navigation bits of PRN 11 110

Figure 58. Raw phase of the input signal for satellite PRN 11 110

Figure 59. Phase of the input signal for satellite PRN 11 111

Figure 60. Doppler frequency of PRN 11 during tracking 111

Figure 61. C/A code delay of PRN 11 during tracking 112

Figure 62. Code search of PRN 20, C/A code step vs. correlation value 112

Figure 63. Frequency search of PRN 20, frequency step vs. correlation value 113

Figure 64. Prompt channel output, i.e. navigation bits of PRN 20 114

Figure 65. Raw phase of the input signal for satellite PRN 20 115

Figure 66. Phase of the input signal for satellite PRN 20 115

Figure 67. Doppler frequency of PRN 20 116

Figure 68. C/A code delay of PRN 20 116

Figure 69. Dynamic experiment set-up 119

Figure 70. Three dimensional X-Y-Z path results (meters) 121

Figure 71. X-Y-Z position errors 121

Figure 72. X-Y-Z velocity errors 122

Figure 73. Azimuth and elevation plot 123

Figure 74. Calculated bias of the receiver clock 124

Figure 75. Calculated drift of the receiver clock 124

Figure 76. Three dimensional X-Y-Z path results (meters)..... 125

Figure 77. X-Y-Z position errors..... 126

Figure 78. X-Y-Z position mean square errors..... 126

Figure 79. X position results 127

Figure 80. Y position results 128

Figure 81. Z position results 128

Figure 82. X-Y-Z velocity errors 129

Figure 83. Azimuth-elevation plot..... 130

Figure 84. Calculated bias of the receiver clock 130

Figure 85. Calculated drift of the receiver clock 131

Figure 86. Stationary experiment set-up 132

Figure 87. X position error of the software receiver 133

Figure 88. Y position error of the software receiver 134

Figure 89. Z position error of the software receiver..... 134

Figure 90. X position error of the Trimble GPS 135

Figure 91. Y position error of the Trimble GPS 135

Figure 92. Z position error of the Trimble GPS 136

CHAPTER 1

INTRODUCTION

This study is aimed to implement a SGR (software GPS receiver). As an introduction, thesis objective, overview of Global Positioning System structure and the thesis outline are presented.

1.1 Thesis objective

Main objective of this study is to implement a software based GPS receiver. However, in order to achieve this main objective, this thesis has certain targets as below:

1. Overall GPS technology is understood by considering several main sources in GPS literature. Therefore, this thesis includes detailed information about GPS technology.

2. GPS literature is surveyed about SGR design. Alternative methods in SGR design are implemented and comparison of their performances are presented as well. These methods are explained in the thesis for practical use.

Thesis work in order to achieve these objectives is introduced by presenting SGR concept, reference sources, overviews of GPS receiver and implemented SGR.

1.1.1 Software GPS Receiver Concept

As expressed in [18], hardware (ASIC-based) GPS receivers provide minimal flexibility since hardware platform does not provide flexible environment for any new generation Global Navigation Satellite Systems (GNSS) such as Galileo. Thus, flexible software based receiver design conception would enter to the scene of the GNSS technology. SGR concept provides developmental comforts in navigation technology. Software based receivers' main advantages are listed below:

1. SGR can be embedded to any platform easily.
2. Improvements in receiver technology can be updated to SGR easily.
3. GPS system updates can be copped easily. For example, new frequency bands or new pseudo random code types may be defined in GPS structure.
4. SGR can be integrated with other devices easily. For example, GPS/INS (inertial navigation system) integration can be developed and improved by using flexible SGR.

1.1.2 Reference Sources

In order to implement a SGR, GPS technology should be understood by surveying satellite navigation literature.

Most of the GPS sources refer to [10] because GPS signal specifications, signal contents and navigation algorithms are defined in this document. Unfortunately, signal processing part is not explained in [10].

[8] is the fundamental source for a SGR designer. However, Kalman filter used in navigation algorithms is not included in this book. This thesis includes Kalman filter positioning as well as the field test results.

[3], [9] and [6] are the best overall references on the GPS system overview and GPS receiver architecture.

[19], [17] and [11] are the papers presenting the SGR implementation study. Furthermore, [19] includes the introduction of the BAAS (the block adjustment of synchronizing signal) method which is implemented in this thesis. [25] is the SGR tutorial, which is useful for SGR design. [17] also uses BAAS algorithm with some improvements on FFT operations. [21] is a good introduction to GPS receiver architecture.

The ION (Institute of navigation) is well known for its technical publications in satellite navigation. [13] includes six volumes of the published material in 1980, 1984, 1986, 1993, 1998 and 1999 prepared by ION. It is a well-known GPS reference and it covers GPS system structure including receiver architecture studies. Moreover, web site of ION provides journal, proceeding and paper search on navigation technology.

[18], [16] and [15] are the lecture notes that were distributed in Navtech seminars. GPS receiver architecture, GPS measurements and software GPS receiver design are explained in these notes. Especially, signal processing section of a SGR is detailed in [18].

[20], [2] and [1] include detailed information about the navigation algorithms designing and positioning methods.

[4], [22] and [12] are important to understand the modulation technique used in GPS signal. Theoretical base about the signal processing section is presented in these references.

Even though many topics related to GPS can be found in these sources, we believe that this thesis presents more complete overview of a GPS receiver with practical aspects by presenting realizable blocks for signal processing and navigation algorithms.

1.1.3 GPS Receiver

General GPS receiver (no matter if hardware or software) sections and the functionality of the sections are shown briefly in Figure 1.

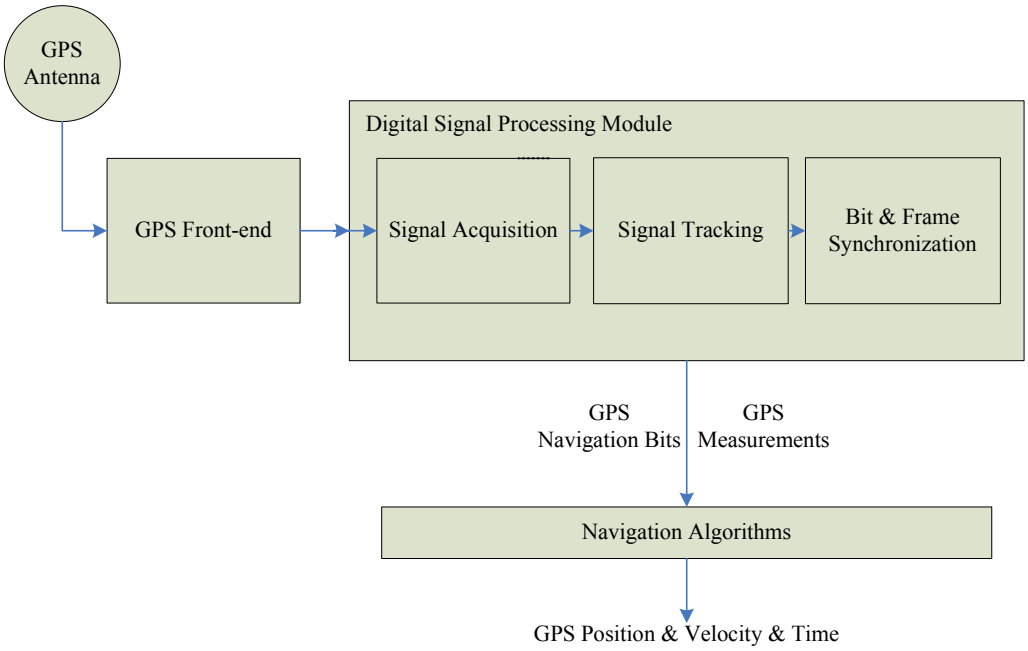

Figure 1. GPS Receiver Stages

As shown in the figure, GPS receiver consists of GPS antenna, front-end, digital signal processing module and navigation algorithms. PVT (position-velocity-time) information is obtained at the output of the GPS receiver.

1.1.4 Implemented Software GPS Receiver

We have implemented complete software GPS receiver except the bit and frame synchronization in Figure 1. Matlab is selected as the software environment. Thus, block diagram of the implemented software GPS receiver can be illustrated in Figure 2.

We have implemented the signal processing module and the navigation algorithms separately as shown in Figure 2. Raw IF GPS digital data is obtained from a GPS front-end and it is provided by Dr. Dennis Akos from Colorado University. Duration of this raw data is not long enough to provide necessary navigation bits to the navigation algorithms. So, GPS measurements and

navigation data are fed from a commercial GPS receiver to the software navigation algorithms.

Raw IF GPS digital data are collected from a GPS front-end of NordNav™ R-25 instrument. Navigation bits are observed at the output of the signal processing module. Moreover, fine measurements in the pseudorange and the Doppler frequency are also obtained at the output of the signal processing module.

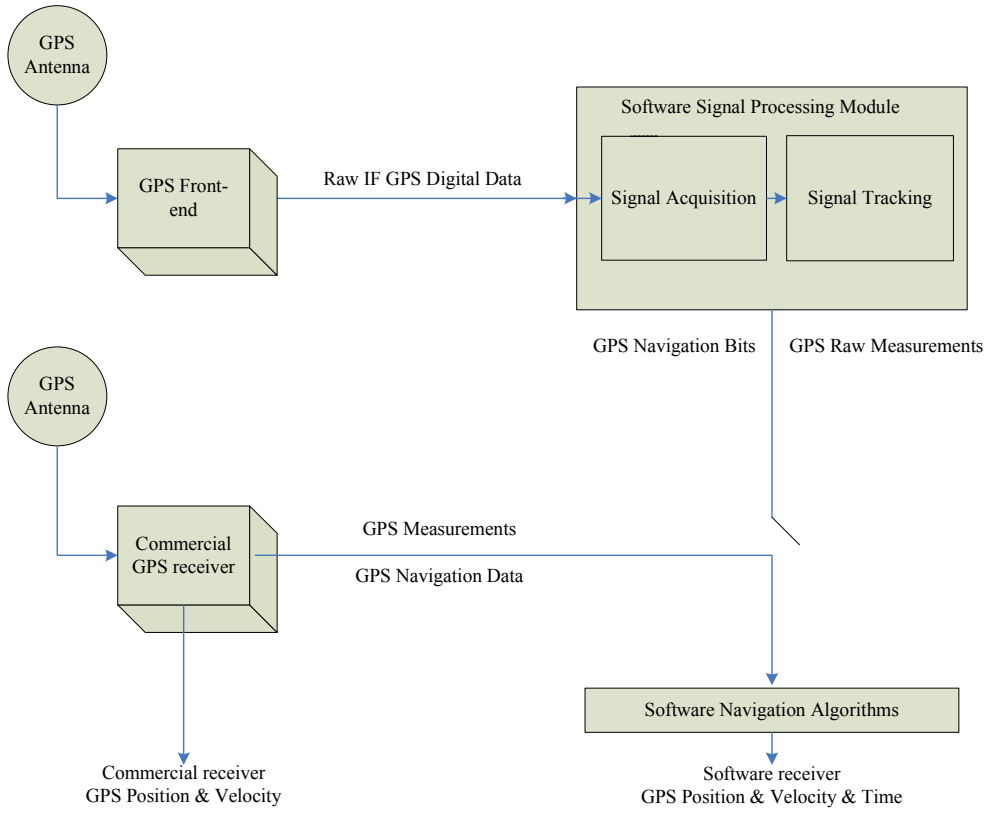

Figure 2. Implemented GPS Software Receiver

The commercial GPS receivers provide GPS measurements and navigation data to the software navigation algorithms. Ashtech™ G-12 Board GPS receiver is used for a moving platform and Trimble™ Force-5 GPS receiver is used for a

stationary platform. Results of the implemented navigation algorithms are compared with these GPS receivers PVT (position-velocity-time) outputs. Outcome of the programs show that the implemented GPS software navigation algorithms closely match with the commercial.

Operation of the implemented SGR is explained in two parts as signal processing and navigation sections.

1.1.4.1 Implemented Signal Processing Section

In order to explain the implemented signal processing operation, first GPS signals must be introduced. GPS satellites broadcast dual frequency signal at L_1 (1575.42 MHz) and L_2 (1227.60 MHz). In this thesis, L_1 signal is considered since it is the GPS civil signal. GPS signals are modulated by spread-spectrum techniques. Mathematical representation of the L_1 signal is shown below as defined in [9]:

$$s(t) = \sqrt{2P_I}\, d(t)c(t)\cos(2\pi f_{L_1} t + \theta) + \sqrt{2P_Q}\, d(t)p(t)\sin(2\pi f_{L_1} t + \theta) \quad (1.1)$$

where:

P_I, P_Q : signal power

$d(t)$: 50 bps navigation data

$c(t)$, $p(t)$: C\A (1.023MHz) and P (10.23 MHz) pseudo random codes.

f_{L_1} : 1575.42MHz

θ : phase difference.

Navigation data in equation (1.1) is a low frequency signal while the C/A (coarse/acquisition) code and carrier signal are high frequency signals. In order to receive and decode GPS signal from the space, these two high frequency components of the GPS signal must be acquired and tracked. Therefore, GPS signals have two-dimensional indefiniteness, which are the pseudo code shift and Doppler effect. Pseudo code of the GPS signal (*c(t)* or *p(t)*) is shifted during the signal transmission from satellite to the GPS antenna. Furthermore, Doppler

effect shifts the carrier signal frequency from the central frequency of 1575.42 MHz due to the high satellite velocities. Code delay and carrier signal frequency change are illustrated in Figure 3. Digital signal processing module of a receiver finds out this code delay and Doppler effect in order to extract *d(t)*.

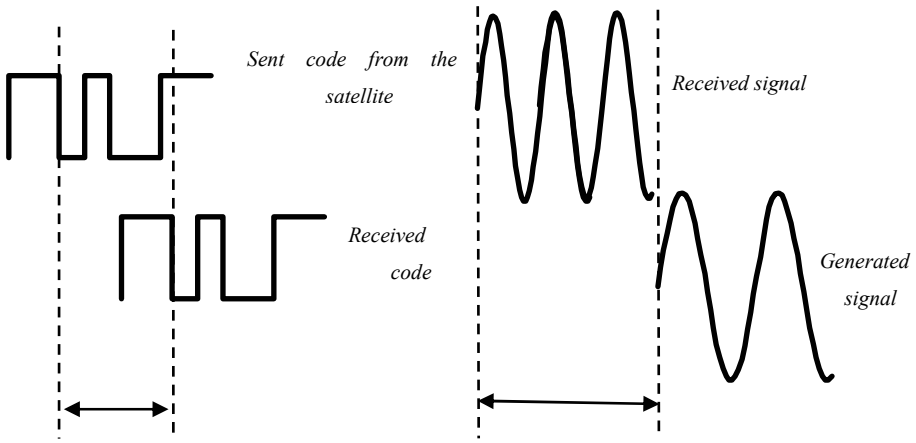

Figure 3. Code and carrier shifts in the GPS signal

Signal processing section is composed of signal acquisition and signal tracking as explained below:

1. Signal acquisition section searches the GPS signal in two dimensions and finds out the code delay and carrier frequency. Acquisition process speed is important in order to achieve positioning as fast as possible. All possible satellite signals are searched in the GPS signal.

2. Signal tracking traces the GPS signal after the signal is acquired. Tracking process aims to follow the satellite signal and to provide navigation data bit extraction. GPS signal tracking requires two loops, which are code tracking loop and carrier tracking loop. Code tracking loop is to strip off the C/A code from the GPS signal. On the other hand, carrier tracking loop detects the frequency change in the GPS signal. Tracking loops must be sensitive enough since small variations on the code and the carrier signals should be detected for accurate operation. Besides, tracking process should control whether the satellite signal is

missed or not. If so, signal acquisition should be re-processed. Another mission of the tracking is to extract the navigation bits by tracing the phase reversals of the GPS signals.

In this thesis, two types of signal processing methods are implemented. These methods are the conventional signal processing and BAAS (the block adjustment of synchronizing signal) signal processing approach as explained below:

1. In conventional signal processing, acquisition is managed by serial search in time domain. Correlation of all possible code delays and Doppler frequency steps are calculated. Maximum correlation point occurs at the correct code delay and Doppler frequency. After the signal acquisition, conventional tracking loops are designed. Conventional approach uses DLL (delay locked loops) for the code tracking and FLL (frequency locked loops) for the carrier tracking. Both of the DLL and the FLL operations are based on PLL (phase locked loop) operation concept.

2. In BAAS signal processing approach, acquisition is achieved by circular correlation. The circular correlation method operates in the frequency domain. After the signal acquisition, BAAS tracking approach tracks the GPS signal in frequency domain. GPS navigation bits are obtained by tracing the phase of the GPS signal.

1.1.4.2 Implemented Navigation Section

In navigation algorithms, GPS positioning is provided by processing the navigation data and the measurements. GPS measurements are the pseudorange and Doppler frequency obtained in signal processing module of the receiver. In navigation algorithms, satellite position is calculated first. Next, pseudorange measurement is corrected according to the satellite clock error and the atmospheric delays. Either LSE(least square estimation) or Kalman filter is used for PVT (position, velocity and time) calculation.

1.2 Global Positioning System

The Global Positioning System is a space-based navigation and time-transfer system, which was developed by the U.S. Air Force (USAF). The system provides accurate, continuous, worldwide, three-dimensional position, velocity and timing information to the GPS users. GPS was firstly developed as a military force enhancement system. Nevertheless, GPS has also demonstrated important potential to benefit the civil usage in an increasing variety of applications. GPS can be integrated with the other navigation systems as well as individual usage.

1.2.1 History

The Global Positioning System (GPS) was developed by the Department of Defense under Air Force Management through the GPS Joint Program Office at the USAF Space Division. Actually, the U.S Air Force and the U.S Navy have had navigation satellite programs from the early 1960s. GPS JPO established on 1 July 1973 and the first satellite was launched in 1978. In February 1994, Federal Aviation Agency declared that GPS navigation service was ready to be used.

1.2.2 GPS Segments

GPS comprises three major system segments as space, control and user segments.

1.2.2.1 Space Segment

Global positioning system performs the positioning by using the GPS satellites. The space segment is the GPS satellites. The satellite constellation includes 24 satellites arranged in 6 orbital planes with 4 satellites per plane. GPS baseline satellite constellation consists of satellites placed in nearly circular orbits with a radius of 26.560 km., circulating period of nearly 12 hours. The satellites are arranged in six orbital planes inclined at 55º relative to the equatorial plane, with four satellites distributed somewhat unevenly in each orbit. With this constellation, almost all users with a clear view of the sky have a

minimum of four satellites in view. It is more likely that a user would see six to eight satellites. Satellite constellation is shown in Figure 4.

The satellites broadcast ranging signals and a navigation message allowing the users to measure their distances from the space vehicle and estimate their positions. Space segment is responsible for preparing the navigation data. During the preparation of the navigation data, time information is measured by the atomic clocks in the satellites.

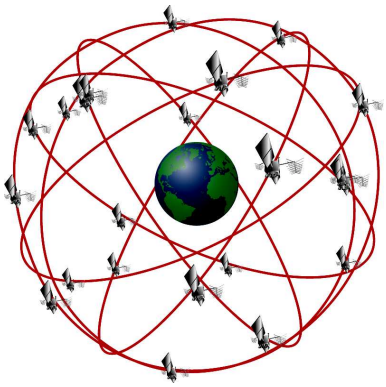

Figure 4. GPS Satellite Constellation

Satellites are the transmitters for the GPS signals. GPS navigation data is transmitted to the user from the satellite in the form of binary data over the ranging signal via a spread-spectrum communication technique. The type of spread-spectrum employed by GPS is known as binary phase shift keying direct sequence spread spectrum (BPSK DSSS). Each satellite generates navigation message at the L-band (L_1 band is at 1575.42 MHz and L_2 band is at 1227.6 MHz).The signal intended for unrestricted use is broadcasted by each satellite at L_1, and it is modulated by a pseudo-random noise (PRN) code called C/A code. Each satellite also broadcasts a pair of signals for the Department of Defense (USA) authorized users, one at L_1 in phase quadrature with the civil signal, and the other at L_2 as shown in Figure 5.

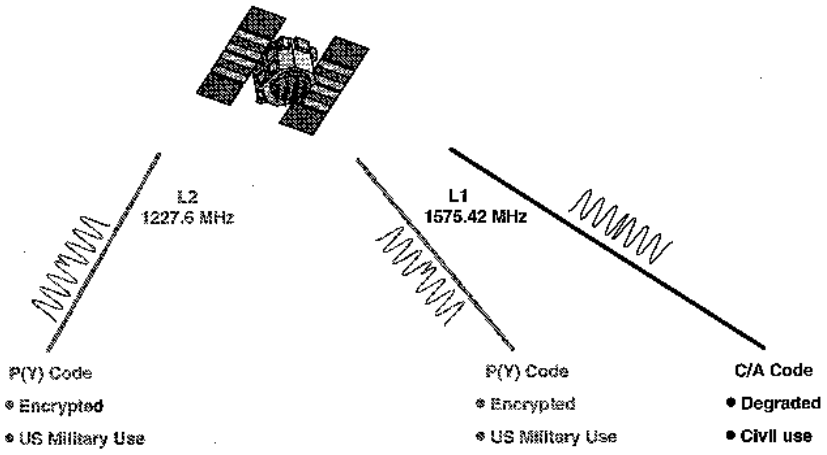

Figure 5. GPS Signals

1.2.2.2 Control Segment

GPS control segment deals with the management of the space segment. Objectives of the control segment can be presented as the following steps:

1) Maintain each of the satellites in its proper orbit.
2) Make corrections and adjustments to the satellite clocks and payload as needed.
3) Track the GPS satellites, generate and upload the navigation data to each of the GPS satellites.

The operational control segment started in 1985 and consists of five monitor stations; four ground antennas upload stations, and the Operational Control Center (master control station).

Master Control Station: The master control station, located at Falcon Air Force Base in Colorado Springs, Colorado, is responsible for overall management of the remote monitoring and transmission sites. As the center for support operations, it calculates any position or clock errors for each individual satellite, based on information received from the monitor stations, and then "orders" the appropriate ground antennas to relay the requisite corrective information back to that satellite.

Monitor Stations: Five monitor stations are located at Falcon Air Force Base in Colorado, Hawaii, Ascension Island in the Atlantic Ocean, Diego Garcia Atoll in

the Indian Ocean, and Kwajalein Island in the South Pacific Ocean. Each of the monitor stations checks the exact altitude, position, speed, and overall health of the orbiting satellites. The control segment uses measurements collected by the monitor stations to predict the behavior of each satellite's orbit and clock. The prediction data is up-linked, or transmitted, to the satellites for transmission back to the users. The control segment also ensures that the GPS satellite orbits and clocks remain within the acceptable limits. A station can track up to 11 satellites at a time. This "check-up" is performed twice a day, by each station, as the satellites complete their journeys around the earth. Noted variations, such as those caused by the gravity of the moon, sun and the pressure of solar radiation, are passed along to the master control station.

Ground Antennas: Ground antennas monitor and track the satellites from horizon to horizon. They also transmit correction information to individual satellites.

1.2.2.3 User Segment

The user segment covers the GPS receiver that outputs PVT (Position-Velocity-Time) information. GPS receiver determines the position by decoding the navigation data of the space vehicle.

GPS receivers do not generally have sensitive atomic clocks because of their high cost. Receivers clocks are less sensitive than the atomic clocks. Due to this clock difference, receivers have time bias from the satellite time, which is called receiver clock offset, or receiver clock bias. Each receivers' clock offset are not same but clock offset is unique for a receiver at a time. This unique offset is calculated together with the user position.

1.2.3 GPS Services

GPS provides two levels of service, which are Precise Positioning Service (PPS) and the Standard Positioning Service (SPS).

1.2.3.1 Precise Positioning Service

PPS is the most accurate positioning velocity and timing service that is provided to only authorized users. PPS is defined for military purposes of United States. The U.S. Department of Defense controls the authorization. PPS receivers can use either the P(Y)-code or C/A-code or both. Maximum GPS accuracy is obtained using the precise code at both frequencies L_1 and L_2.

1.2.3.2 Standard Positioning Service

SPS is a less accurate positioning and timing service, which is available to all GPS users. SPS is defined for civilian purposes. The United States Government defines GPS SPS as defined in [23]:

SPS is a positioning and timing service, and is provided on the GPS L_1 frequency. The GPS L_1 frequency, transmitted by all GPS satellites, contains a coarse acquisition (C/A) code and a navigation data message. The GPS L_1 frequency also contains a precision (P) code that is reserved for military use and is not part of the SPS. The P code can be altered without notice and will not normally be available to users that do not have valid cryptographic keys. GPS satellites also transmit a second ranging signal known as L_2. This signal is not part of the SPS, although many civil receivers have incorporated technologies into their design that enables them to use L_2 to support two-frequency corrections without recourse to code tracking logic. SPS performance standards are not predicated upon use of L_2.

1.3 Thesis outline

This thesis consists of six chapters including the introduction chapter. Chapter-2 gives details about the GPS signal structure, GPS receiver sections, and their duties. Chapter-3 provides information on the signal processing module and the implemented software algorithms. Chapter-4 is the part that the navigation algorithms are introduced and the implemented navigation algorithms are presented in detail. Finally, chapter-5 reports the performance of the implemented software receiver in comparison of real GPS results. Conclusion is

the last chapter of the thesis to present the importance and the results of this study.

CHAPTER 2

GPS SIGNAL and RECEIVER

This chapter includes the detailed information about the GPS signal and the GPS receiver structure.

2.1. GPS Signal Structure

The design of the GPS signal structure is done by considering frequency bands for various specifications ([6]). Several frequency bands were taken into account according to their path loss, atmospheric effects, etc. At the end of the studies, L-band (1-2 GHz) was chosen. Dual carrier frequency L_1 and L_2 were selected in order to provide group delay ionospheric measurement. Signal bandwidth is defined as 20 MHz.

GPS satellites broadcast dual frequency signal at L_1:1575.42 MHz and L_2:1227.60 MHz. Each of the carrier frequency is selected as a multiple of f_o =10.23 MHz master clock:

$$f_{L1} = f_o \text{ x } 154 = 1575.42 \text{ MHz}$$

$$f_{L2} = f_o \text{ x } 120 = 1227.60 \text{ MHz}$$

L_1 and L_2 carrier signals are modulated with the two ranging codes (C/A and P codes) and navigation data according to the spread spectrum signaling.

2.1.1. Spread Spectrum Modulation

Spread spectrum is a radio-frequency communications system. Bandwidth of the base band signal is intentionally spread over a larger bandwidth by injecting a higher-frequency signal. Consequently, energy used in transmitting the signal is spread over a wider bandwidth, and appears as noise. The ratio (in dB) between the spread base band and the original signal is called processing gain. Typical spread spectrum processing gain may vary from 10dB to 60dB as stated in [8]. The spreading effect to the power spectral density is figured in Figure 6 from [24]. As shown from the figure, spread waveform is below the noise level even though the original signal exceeds the noise level significantly.

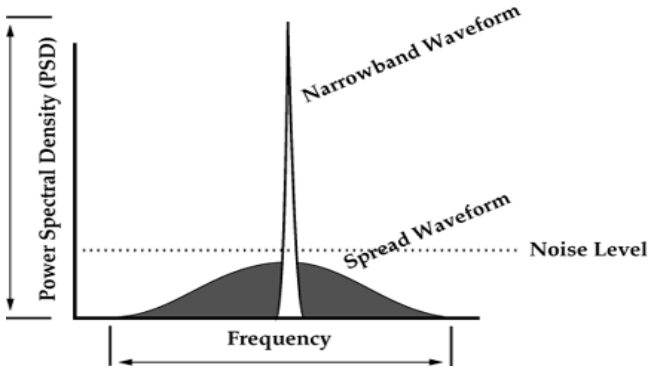

Figure 6. Power Spectral Densities in Spread Spectrum Technique

Main process of the spread spectrum signaling is multiplying the corresponding spread spectrum code with the transmitted signal. Multiplication is to extend the information in a larger bandwidth. Conversely, in the receiver despreading operation is performed at a point in the receive chain before data retrieval. The effect of a despreading operation is to construct the information in its original bandwidth. It is obvious that the same code must be known in advance at both ends of the transmission channel. Transmission channel from input data to output data is shown in Figure 7 from [24].

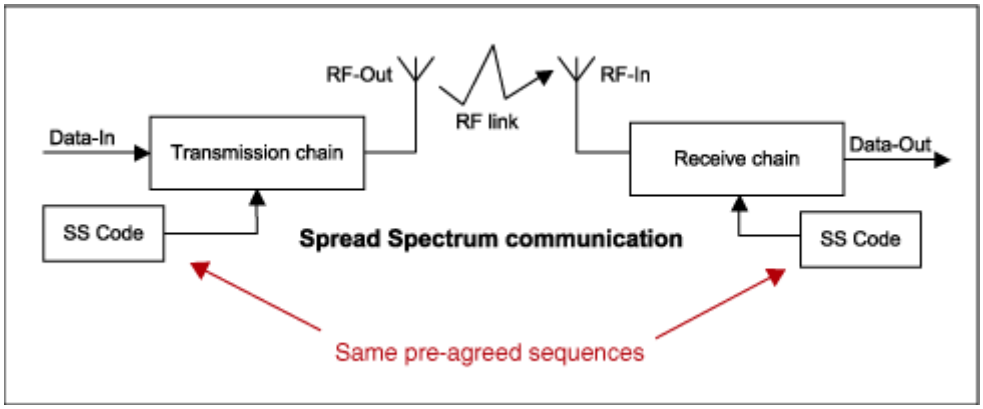

Figure 7. Transmission channel in spread spectrum communication

There are two common forms of spread spectrum; DS-SS (Direct Sequence Spread Spectrum) and FH-SS (Frequency Hopping Spread Spectrum) ([22]). There are also various hybrid forms of DS-SS and FH-SS.

In GPS signal structure, DS-SS is selected because it provides a means to recover precise timing, and at the same time, it permits recovery of the pure radio frequency carrier. In addition to this, in GPS spread spectrum modulation technique, spread spectrum code is a pseudo-noise code that is independent of the signal. Despreading of the pseudo-noise code is the GPS receiver task that can be achieved if and only if the GPS receiver generates a synchronized replica of the code.

2.1.1.1 Direct Sequence Spread Spectrum, DS-SS

In the DS-SS modulation technique, the bandwidth of the transmitted data is extended by using a spreading signal. Spreading signal selection is important discussion in the aspect of the demodulation part. In DS-SS, binary pseudo-random signals are used as spreading signal. Clock frequency of a pseudo-random signal should exceed the input data bit frequency significantly. In Figure 8 ([22]), typical DS-SS transmitter and receiver units are shown.

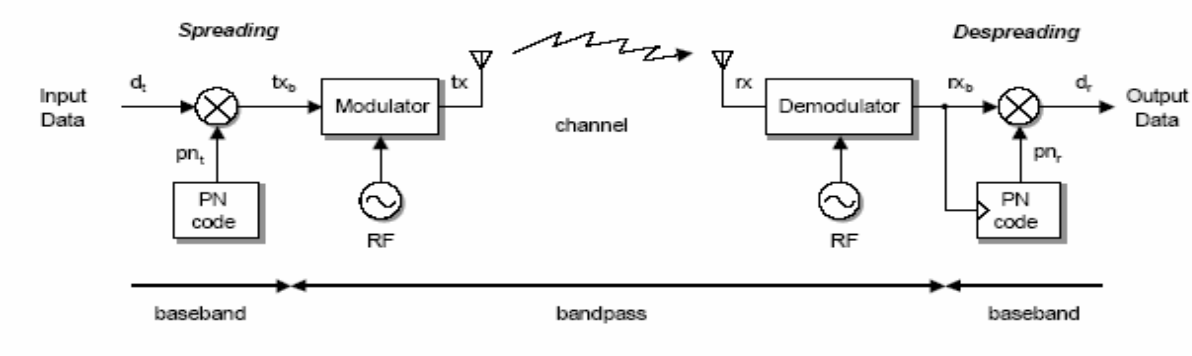

Figure 8. DS-SS transmitter and receiver ([22])

In Figure 9 ([22]), time domain and frequency domain representation of the signals are shown. Excluding the carrier modulation, input data d_t's bandwidth spreading is accomplished by multiplying the data with the PRN (pseudorandom) sequence pn_t in the transmitter. In the figure, bit rate of the input data is R_s ($f_s=1/R_s$), pseudo random code chip rate is R_c ($f_c=1/R_c$). R_c should be an integer multiple of R_s in order to simplify the signal processing building. Since the timing of the data and clock transitions are synchronous, the spread spectrum product $tx_b = d_t \, x \, pn_t$ has exactly the same spectrum as that of pn_t alone as shown in Figure 9.

When it comes to the receiver part of DS-SS, despreading of the signal is succeeded by multiplying the received signal with a replica of the spreading signal PN code. In other words, despreading can not be achieved unless the pseudo random code is synchronized with the code in the received signal. In Figure 10 from [22], pn_r is the receiver generated pseudo random code that is synchronized with the code of the incoming signal. As observed from the time domain signals, recovered signal d is same as the transmitted data d_t. Besides, frequency domain signal is recovered as desired.

The most important issue of despreading is the synchronization of the PN code. If pn_r and pn_t are not time synchronized, multiplying the generated code results meaningless signal. Thus, despreading action does not happen.

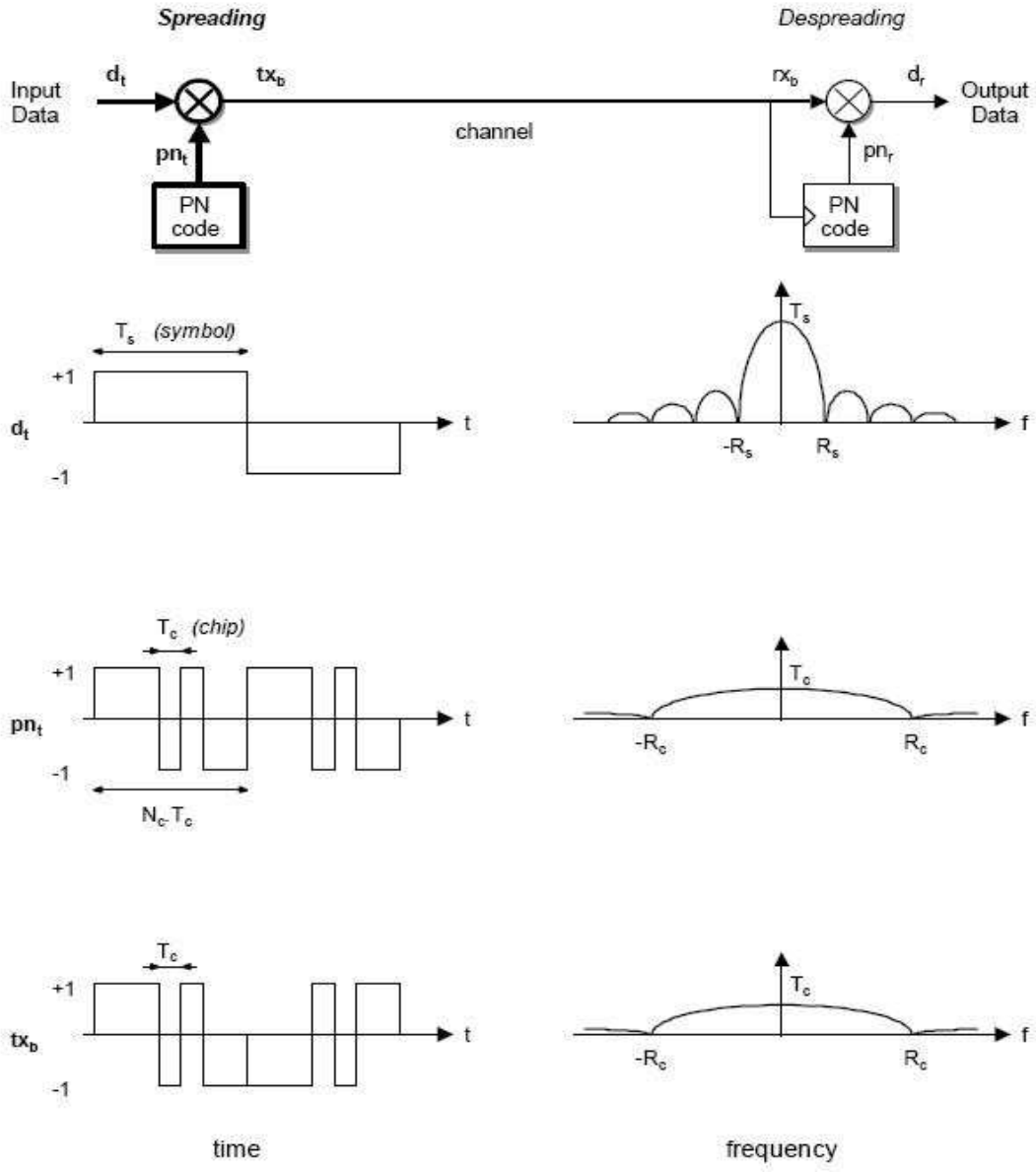

Figure 9. Transmitted signals in time and frequency domains ([22])

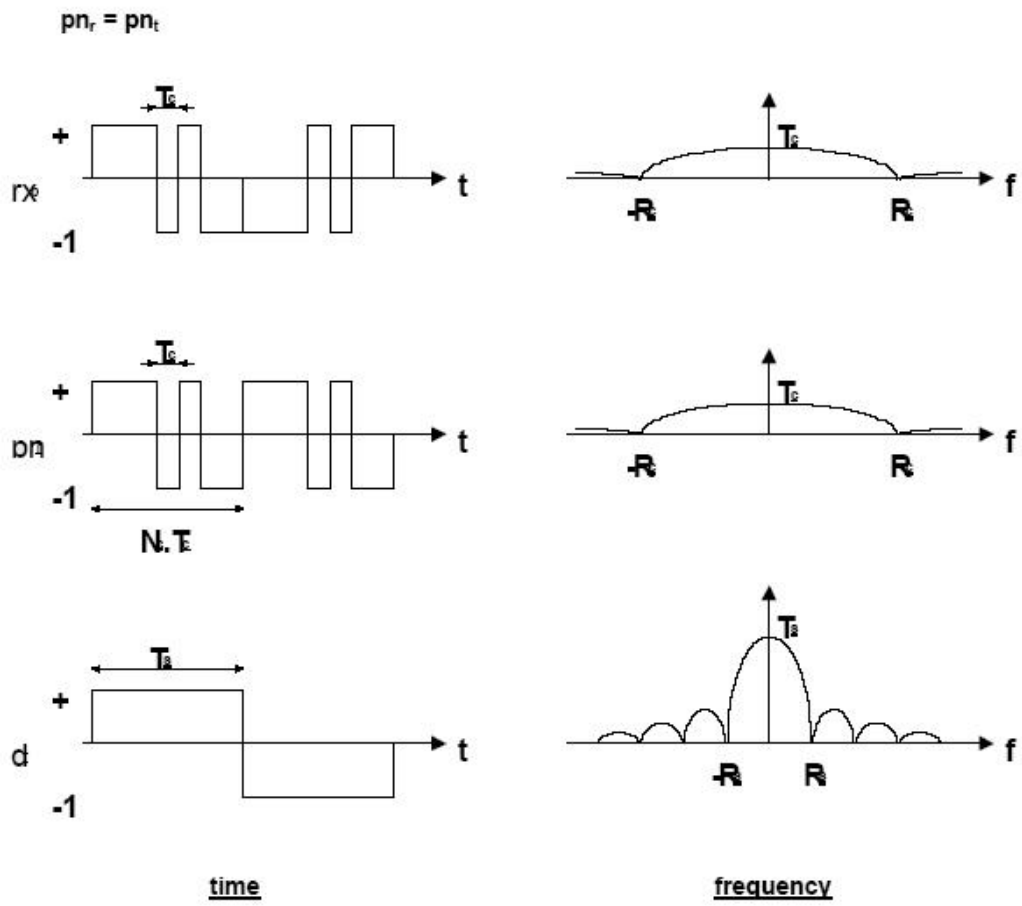

Figure 10. Received signals in time and frequency domains ([22])

2.1.1.2 Pseudo Random Codes

As [8], [6] and [10] state PRN code is a noise like sequence. Nevertheless, code should be deterministic in order to be used in despreading operation.

Main property of the PRN code is correlation functions. Its auto-correlation function is actually similar to that of a white noise signal. Hence, the autocorrelation function is impulse like signal. On the other hand, cross-correlation function is almost zero for different PRN codes since the codes are constituted as orthogonal to each other. This provides CDMA (code division multiple access) property.

GPS spreading uses three kinds of PRN codes, which are C/A (coarse-acquisition), P (precise) and P(Y) code. P is the encrypted code for authorized users. Y code is more secure than P code in case of spoofing. Y code is converted from P code at the same clock frequency. Civil users are able to decode the C/A code. P code is the unavailable crypto code. Therefore, C/A code type will be presented.

GPS signal at L_1 signal band includes the navigation data multiplied with C/A code (coarse/acquisition) and multiplied with the carrier signal. C/A code is unique for each GPS satellite. GPS receiver should be able to generate the PRN codes of the all satellites to be able to use in the tracking loop. Thus, GPS receiver should have PRN generator in itself.

2.1.1.2.1 C/A Code Generator in GPS

One period of C/A code signal is composed of 1023 bits. Code bits are obtained from the multiplication of the G_1 and G_2 PRN bit sequences. G_1 and G_2 generators have the same principles. They have 10 stages linear feedback shift register (LFSR). In GPS receivers, clock of the shift register is 1.023 MHz.

If a linear feedback shift register, LFSR has n bit register, length of the generated sequence is (2n-1). In the simulation program, D flip-flops are used for the shift register stages. In the G_1 generator, 3^{rd} and 10^{th} bits are modulo-2 added and used for feedback. Therefore, generator polynomial of G_1 is G_1:$1 + x^3 + x^{10}$. On the other hand, G_2 generator polynomial is G_2: $1 + x^2 + x^3 + x^6 + x^8 + x^9 + x^{10}$. Output of G_1 is same for all GPS satellite codes. Nevertheless, the output of the G_2 generator differs from satellite to satellite. G_2 generator output is modulo-2 addition of the two bits of the G_2 bits. The selection of these two bits changes with the satellite PRN number. In [10], G_2 output bits are defined for all GPS satellites. Hence, unique C/A code is produced for each satellite. G_1 and G_2 generators are presented in Figure 11.

2.1.1.3 Code Division Multiple Access Technique

Spread spectrum signaling has the capability to provide a form of multiple access signaling called code division multiple access The GPS signal is code division multiple access (CDMA). In CDMA signaling, multiple signals can be transmitted in the same frequency channel.

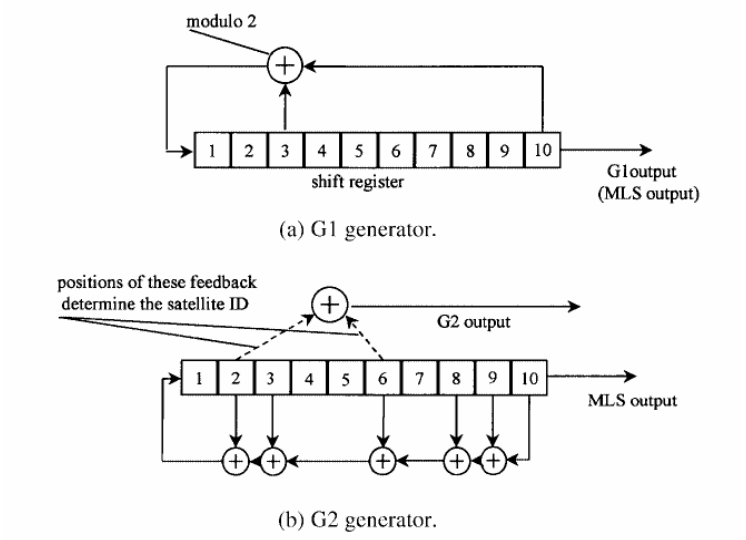

Figure 11. G_1 and G_2 generators

CDMA technique provides entire GPS satellites transmitting their navigation data at L_1 and L_2 frequencies. This multiple access property is important since any GPS receiver needs to lock to at least four satellites to make a position fix. Signals in CDMA do not interfere since they are modulated by a set of orthogonal (or near-orthogonal) codes. Hence, correlation of an individual signal with its code is maximum whereas correlation with other codes is minimum. Interference between the satellite codes is avoided by setting the whole satellite signals to approximately the same power level.

2.1.2. GPS Signal

GPS satellites broadcast dual frequency signals at L_1 and L_2. The power spectral densities of the signals are shown in Figure 12 as illustrated in [6].

Civil users can only process the L_1 signal to obtain the position and velocity information. In this thesis, L_1 signal will be used. The L_1 carrier signal is modulated by the navigation data and the pseudo random code. The pseudo random codes are modulo-2 added to the navigation data as defined in [10]. Both of the P and the C/A codes are used in L_1 signal. Quadrature component of the carrier is modulated by the P signal (P+D) where D is denoting for the 50 bps navigation data. In-phase component of the carrier is modulated by the C/A signal (C/A+D).

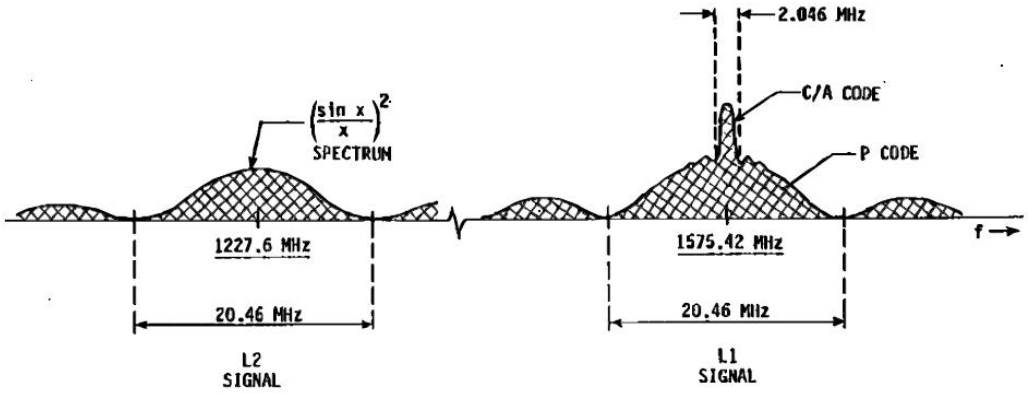

Figure 12. GPS Signal

Mathematical representation of the L_1 signal is as shown below ([9]):

$$s(t) = \sqrt{2P_I}\,d(t)c(t)\cos(2\pi f_{L_1}t + \theta) + \sqrt{2P_Q}\,d(t)p(t)\sin(2\pi f_{L_1}t + \theta) \qquad (2.1)$$

where:

P_I, P_Q : signal power

d(t) : 50 bps navigation data

c(t) , p(t) : C\A (1.023MHz) and P (10.23 MHz) PRN codes.

f_{L_1} : L_1 carrier frequency (1575.42MHz)

θ : phase difference.

d(t) in (2.1) conveys the relevant navigation information to determine the position. Therefore, GPS receiver aims to extract *d(t)* from *s(t)* in (2.1). While the GPS signal transmits from the satellite to the receiver on Earth, shifts occur on the *s(t)* because of the Doppler effect on the frequency and the delay in the pseudorandom code. These delays should be found out in order to find the navigation data *d(t)*.

The navigation data bit rate is only 50 Hz or one data bit is 20 milliseconds long. There are 20 C/A code in each data bit since one C/A code period is 1 millisecond. Hence, in one data bit all 20 C/A codes have the same phase. Phase transition occurs due to the data bit change. In one data bit, there are so many carrier signal cycles that carrier signal can be processed easily in the interval. GPS L_1 signal timings are shown in Figure 13 that represents the components of the *s(t)* in one bit interval.

The L_2 signal is bi-phase modulated by either the civil or the encrypted signal. L_2 modulation may be commanded by the ground stations. In general, operation of the L_2 signal should include P(Y) code modulation. The L_2 signal is modulated with only the 50 bps navigation data and a pseudorandom code called P code that has a period of one-week and chipping rate of 10.23 MHz.

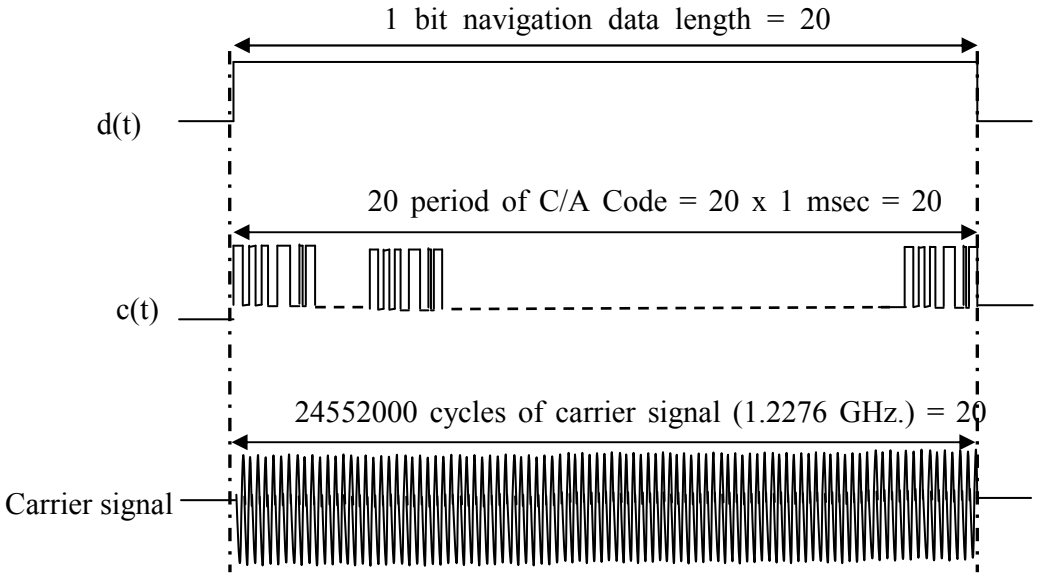

Figure 13. GPS signal timings

The mathematical representation of the L_2 signal is ([9]):

$$s(t) = \sqrt{2P_Q}\, d(t)\, p(t) \sin(2\pi f_{L_2} t + \theta) \qquad (2.2)$$

where

P_Q : Signal power

$d(t)$: 50 bps navigation data

$p(t)$: Military-P (10.23 MHz) pseudo codes

f_{L_2} : L_2 carrier frequency (1227.60MHz)

θ : Oscillator drift component

In the next years, L_5 signal at 1176.45 MHz is considered to be activated. Bandwidth of the L_5 signal is defined as 24 MHz and signal power is -154 dB. Besides, new codes are planned to be used such as M (military) code and C (civil but different from C/A code) code in L_2 signal (ION GNSS Conference 2004).

When it comes to the power levels of the GPS signal, minimum power levels of the signals must satisfy the values presented in Table 1 ([8]).

Table 1. Minimum power levels of the GPS signals

Frequency	P code	C/A code
L_1	-133dBm	-130 dBm
L_2	-136 dBm	Not present

Actually, these power levels are so weak that they can not be observed easily. This is the result of the fact that the noise is stronger than the signal. Low power level of the GPS signal is convenient for jamming. Figure 14 ([24]) shows the thermal noise floor and the L_1 signal level.

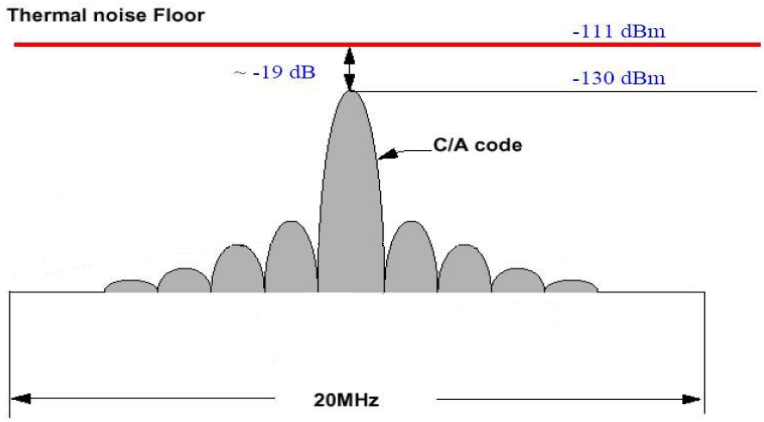

Figure 14. L_1 GPS signal and the noise levels

2.1.3. GPS Signal Content

Up to this part, L_1 GPS signal structure is considered in terms of modulation techniques, power levels, etc. In this part, GPS signal is analyzed according to its content according to definitions in [10]. Since the PRN code and

the carrier signal do not contain any information, only the navigation data bit content will be discussed.

As mentioned before, GPS navigation data are transmitted through 50 Hz bit stream. The whole GPS data structure is shown in Figure 15. GPS data bits are addressed to word, subframe, frame and page structures.

Word is the smallest element of the GPS data, which includes 30 bits. All words contain the navigation data bits from 1 to 24. Left most 6 bits are reserved for the parity check algorithm. Parity check algorithm is used to correct the polarity of the data bits. Subframe of the GPS data consists of 30 words.

Every frame of the navigation data are not the same since the forth and the fifth frames contain the almanac data of all the GPS satellites. Almanac data replaced through 25 frames successively. A superframe consists of 25 frames and it contains the necessary information including almanac data of all the GPS satellites.

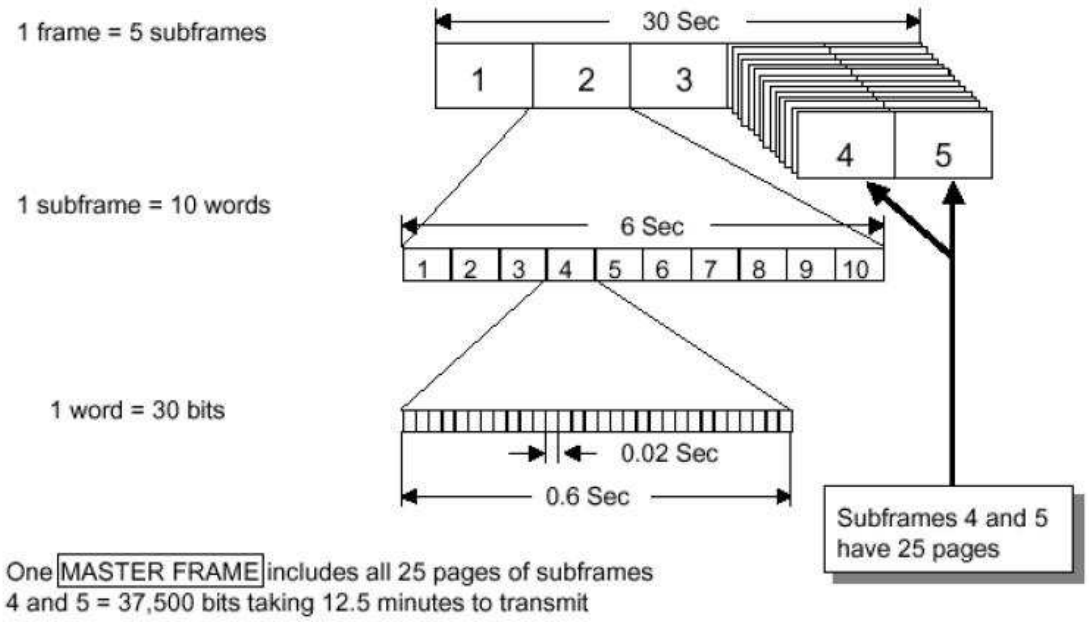

Figure 15. GPS Data structure

2.2. GPS Receiver

GPS receiver determines its position by processing the GPS signals and the navigation information. GPS receiver consists of the sections as shown in Figure 1.

2.2.1. GPS Antenna

The GPS antenna receives signals either at L_2 (1575.42MHz) or L_1 (1227.60MHz) and the characteristic bandwidth of both L_1 and L_2 signals is 20 MHz.

The major task of a GPS antenna is covering a wide spatial angle to collect the maximum number of signals according to [8] and [6]. Besides, antenna picks the reflected signals from the environment of the receiver. Some of the specially designed antennas have the ability of eliminating jamming and interference signals.

GPS antenna conducts the electromagnetic signals to the receiver stages by converting the signals to the electrical currents. GPS antenna and receiver may be united or separated. In a separated configuration, antenna is connected to the receiver by a cable. Since long cable may cause data loss, antenna cable should be as short as possible. Depending on the frequency operation of the receiver, antenna may operate on dual frequency L_1 and L_2. Such antennas get rid of signal reflections by the help of “ground plane” and “choke ring” joints. The most common antenna types are micro strip, dipole and helix as [14] is emphasized.

2.2.2. RF stage (front end)

The mission of the receiver front-end is to filter the GPS signal and to amplify it according to [8] and [25]. The incoming signal contains channel noise and it may be interfered with more power signals that have center frequency close to the GPS signals. In addition to this, GPS signal has so weak power that the signal should be amplified for processing. Typically front-end procures 35-55 dB of gain to the GPS signal. Band-pass filters prevent the front-end to amplify

the powerful interference. Nevertheless, actual interference filtering is achieved in the intermediate-frequency (IF) stage.

Intermediate-Frequency Stages

The purpose of the IF-Stages is to amplify and to decrease the center frequency of the GPS signal in one or more steps. Bandpass filters are utilized for the down-conversion since the bandwidth of the signal shall not be affected. As the signal has lower center frequency, sampling of the signal becomes less complex.

Analog to Digital (A/D) Converter

The modern GPS receivers apply digital signal processing to extract the navigation data and measure the relevant information such as Doppler frequency, pseudorange, etc. For this purpose, IF signal in 2-20 MHz range should be sampled for further processing.

2.2.3. DSP module

Digital signal processing module of the GPS receiver is the main portion of the receiver that first acquires the signal and than provides the measurements and the navigation data bits. Figure 16 shows the signal acquisition, signal tracking and frame extraction section flow diagram. DSP part consists of three major sections:

1. Signal Acquisition: In signal acquisition, satellite signal is searched. Besides, one can have an idea of raw measurements of the code and the carrier phase in the signal acquisition. Because satellite signal can only be acquired if and only if the indefiniteness on the PRN code phase and carrier frequency are determined.

2. Signal Tracking: Fine pseudorange and Doppler frequency measurements are obtained by processing the GPS signal in the code and carrier tracking part. In addition to this mission, tracking parts control the existence of the satellite signal in the received signal. If the signal is

missed, signal acquisition needed to operate again in order to catch the signal. Code and the carrier tracking loop outputs provide the feedback to the other loops. Navigation bits are observed from the output of the carrier tracking loop.

3. Bit and Frame Extraction: Frame synchronization process is required in order to convert the navigation bit streams into the relevant navigation parameters. Finding the subframe places in the bit streams and the parity check algorithm are the main functions of the frame extraction part.

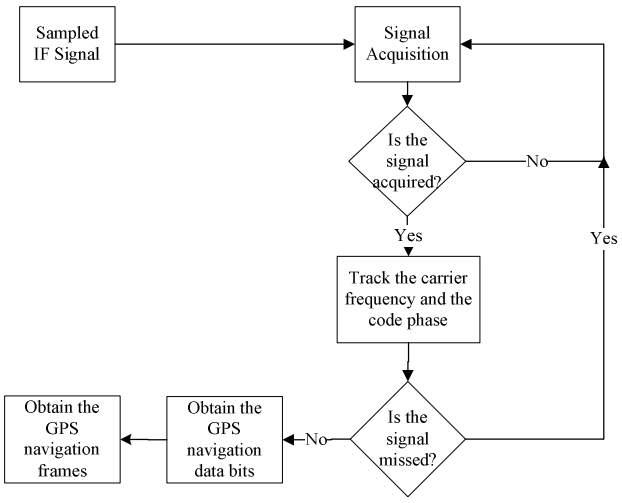

Figure 16. Signal acquisition and tracking flow diagram

2.2.4. Navigation Algorithms

Navigation algorithms are used for the user position and velocity by using the navigational parameters obtained in the bit extraction and the measurements from the tracking loops. Pseudorange measurement calculated by the receiver has some errors such as satellite clock correction, atmospheric delays, etc. Thus, these errors are eliminated in order to obtain precise pseudorange as much as possible. Navigation algorithms calculate the satellite position by using the ephemeris parameters. Kalman filter or least square estimation algorithm

calculates the position and the velocity by direct use of the satellite positions, pseudoranges and Doppler frequencies. User position, velocity, receiver clock bias and receiver clock drift are the outputs of the navigation algorithms. Figure 17 is the flow diagram of the navigation algorithms.

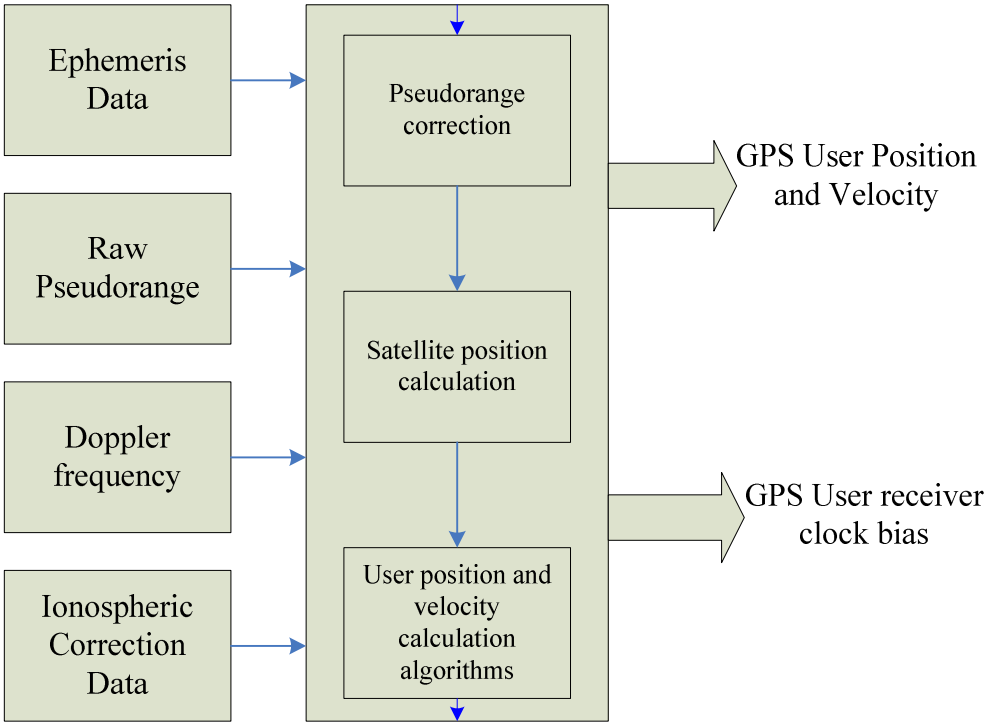

Figure 17. GPS Navigation Algorithms

CHAPTER 3

DSP MODULE

The functionality of the digital signal processing module of the GPS receiver is to extract the navigation data and the GPS measurements which are the pseudoranges and the Doppler frequencies. The GPS navigation message is a low frequency signal (50 bits per second) while the C/A code and the carrier are high frequency signals. In order to receive GPS signal, these two high frequency components of the GPS signal must be acquired and tracked. Therefore, signal acquisition and signal tracking are two-dimensional processes. Signal acquisition part provides the raw measurements of the satellites whereas signal tracking finds out the fine measurements and the navigation data bits. Bit and frame extraction is the process to find out the relevant information in the navigation data and to form the exact measurements. Signal acquisition and tracking are implemented in the software GPS receiver. Signal processing module is implemented by using two methods, which are conventional, and BAAS approach. In this chapter, these methods are discussed in detail.

3.1 Operation of the signal processing module

Signal processing module consists of signal acquisition, signal tracking, bit and frame synchronization as detailed in the following sections.

3.1.1 Signal Acquisition

The purpose of acquisition process is to detect and acquire the GPS signal. In general, signal acquisition searches all 32 GPS satellites signals in the received GPS signal. As it is discussed in [8], if a receiver knows its approximate position and time, it can estimate the possible satellite elevation angles from the current GPS almanac data. Almanac data might be obtained either from the GPS signal or even from the internet sites offline. This extra information increases the speed of the acquisition process. Hot start of GPS receivers is the operation of uploading data into the receiver as soon as it is opened for navigation.

In order to achieve acquisition of satellite signals simultaneously, all or possible GPS satellites are searched in parallel channels. Signal acquisition implementation approach of this thesis consists of two methods: the conventional acquisition and the acquisition by circular convolution. Both of the signal acquisition methods are based on multiply-accumulate operations as discussed in [25]. However, conventional and circular convolution acquisitions process the signal in different domains. These methods will be presented in the following parts.

Before starting the methods, multiply-accumulate operation should be discussed first. From [17], the multiply-accumulation operation in signal acquisition is the calculation of the correlation between the incoming signal and the locally generated reference signal, i.e.,

$$z(n) = \sum_{m=0}^{N-1} x(m)y(m+n) \tag{3.1}$$

where *z(n)* is the correlation of the received signal *x(n)* and the locally generated reference signal *y(n)*. It should be considered as the multiplication of the code signal and the carrier signal at an acquisition step. So, *z(n)* is the correlation of all code offsets at one frequency bin. Signal acquisition requires having *z(n)* larger than a pre-defined threshold.

3.1.2 Signal Tracking

As GPS satellites move through their orbits, GPS signal measurements such as Doppler frequency and the pseudorange vary from time to time. Signal acquisition cannot be done since it is a computationally expensive approach during whole operation of GPS receivers. Hence, tracking is fine-tuning of the GPS satellite signal.

Signal tracking is also required for accurate derivation of the code delay and the carrier frequency estimate. Consequently, tracking process consists of two main closed loop systems, which are called code tracking loop and carrier tracking loop. Both of the tracking loops can operate independently with feedbacks to each other.

The purpose of C/A code tracking loop is the synchronization of the locally generated C/A code with the GPS signal. This synchronization will end with the following two results as [9] indicates:

1. Signal despreading will be achieved as soon as the C/A code is fully cleared from the signal. Navigation data is not reachable without code synchronization. C\A code in the transmitted signal becomes shifted owing to the transmission from satellite to the receiver. The shift in the code imparts information to measure the pseudorange. C/A code is a pseudo-random code that is from family of the codes known as gold codes. PRN code has values of -1 and 1. If we are able to generate the replica code in correct series, we obtain (-1) x (-1) = (+1) x (+1) = 1 for the code correlation. Otherwise, noise is obtained for the correlator output since the code does not match.

2. Fine range measurement can only be obtained from the code loop. Since the GPS signal is below the noise level, high gain code loop will provide precise measurement by considering the synchronized reference code carefully. In order to find out the correct sequence of the code and the fine range measurement, three version of the C\A code of the satellite are generated by the receiver, which are early, punctual and late with at least

0.5 chip delay between each other. Then, they are multiplied by the GPS signal. This correlator structure determines which code version is correlated most. Interval in the correlator should be a multiple of the period of the C\A code (i.e., 1ms-4ms) in order to observe the correlation of the generated codes. Signal is tracked according to the error that is defined by the early correlation minus late correlation. If the error is positive, we can conclude that early code is more correlated than the late code. For the next step, early code becomes punctual code. By delaying under such error control, the correct timing of the C\A code is obtained. If the position of the C\A code is found correctly, the correlator of the punctual code output gives the maximum output and error becomes very close to 0.

Carrier frequency of the GPS signal is affected by the Doppler effect. Doppler effect causes frequency shift in the carrier frequency which is because of the large satellite velocities. The received signal can have large Doppler shifts as much as ±5kHz with 1Hz/second for stationary or dynamic receiver. Doppler shift should be calculated sensitively to provide precise velocity measurements. Determining the Doppler frequency precisely will present the following advantages as [9] indicates:

1. Precise velocity measurement is provided via phase rate. Doppler frequency is directly related to the relative velocity difference of the user and the satellite. Satellite average velocities are estimated from the rate of change in position. Therefore, user velocity can be estimated by the Doppler frequency measurement via least squares estimation or Kalman filter just like the position determination.

2. Doppler frequency is designated to generate the local reference carrier signal. Locally generated reference signal is the most critical element for the detection of the BPSK modulated navigation data.

3. GPS navigation data is observed through the phase change of the demodulated GPS signal. This is actually the final point to reach for the digital signal processing section of a GPS receiver.

In this thesis, two main methods for tracking are considered. These are the conventional signal tracking loop approach and the software specialized block adjustment of synchronizing signal (BAAS) approach.

3.1.3 Bit and Frame Synchronization

In code tracking, C/A code beginning point is known precisely. However, it is not known where the navigation bit data begins. Hence, there exists bit offset ambiguity and bit synchronization solves this ambiguity as in [25].

Frame synchronization is required in order to extract the navigation parameters from the navigation bits. Subframes 1, 2, 3, 4 and 5 are found in the bit stream. Preamble bits are placed in the beginning of a subframe. Therefore, subframes are found by searching the preamble bits. Furthermore, parity check algorithm is applied to test bits polarities in a word.

3.2 Conventional Signal Processing

Conventional signal processing is composed of conventional acquisition loop and conventional tracking loops.

3.2.1 Conventional Signal Acquisition Method

Conventional signal acquisition performs two-dimensional carrier frequency and C\A code search. C\A code and the carrier frequency acquisition go on until the correlator output gives maximum, i.e. until the presence of the signal is guaranteed.

Conventional approach calculates the correlation of a specified satellite signal with the received GPS signal. This is a serial search that tests every possible combination of parameters until the signal is acquired. Correlation value is called as acquisition metric and it is calculated for each candidate carrier

frequency for every shifted code. One code shift magnitude is 0.5 chip. Since the whole code length is 1023, candidate code phases are 2046 length. Conventional acquisition method must search over a frequency range of ±10 kHz to cover all of the expected Doppler frequency range for even high-speed aircrafts.

Let us assume that at any step of a signal acquisition, the code shift value is m and the searched carrier frequency is f_n. The ranges are m = 1:0.5:1023 and f_n = IF frequency + n.500Hz; n = -10: 1: +10. Consequently, acquisition metric will be a matrix with a dimension of 21 x 2046. Acquisition metric value is obtained from the integration of the signal multiplied with carrier signal at f_n and code shifted by m during one code period of 1 millisecond. Thus, any phase shift of the carrier signal becomes unimportant.

$$acquisition\_metric(f_n, m) = \int_{T=1\ ms} s(t)c(t-m)\cos(2\pi f_n t)dt \qquad (3.2)$$

where:

acquisition_metric(f_n,m) : acquisition metric at f_n frequency and m shifted code,

s(t) : GPS signal,

c(t-m) : m shifted C/A pseudo code,

f_n : carrier frequency.

Figure 18, which is given in [17], illustrates the computation of the *acquisition_metric(fn,m)*. Block diagram of the conventional acquisition loop is presented in Figure 19. In-phase and the quadrature components are obtained in order to decide whether the satellite is acquired or not.

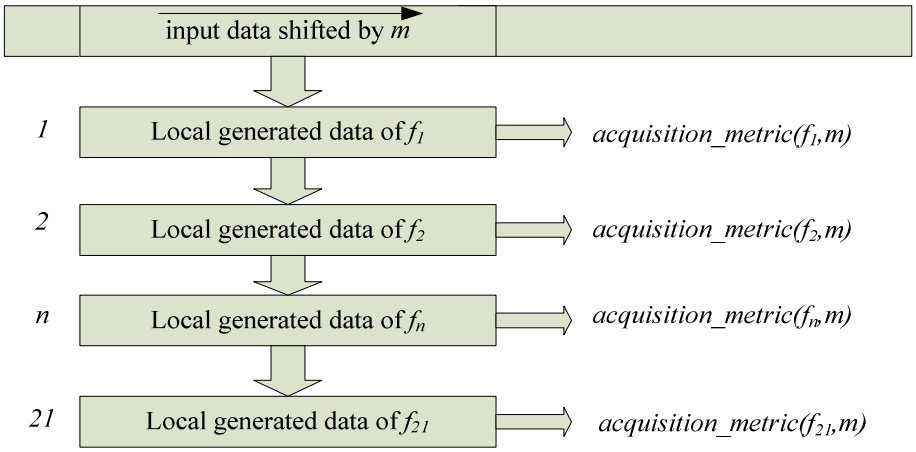

Figure 18. Conventional acquisition process

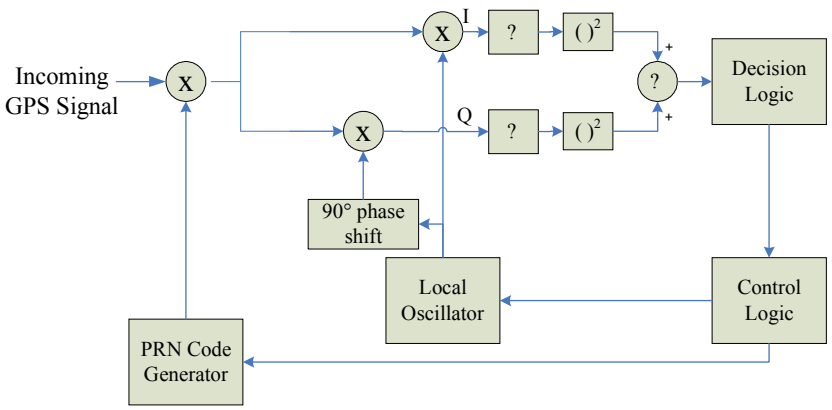

Figure 19. Conventional acquisition loop

Consequently, carrier frequency is obtained within 500 Hz steps. However, this frequency resolution is not appropriate for tracking loops. Therefore, confirmation search is needed. Frequency resolution is increased to 25 Hz by

applying the acquisition between 500 Hz below and 500 Hz above the acquired frequency.

Conventional signal acquisition software program follows the steps given below:

Step 1. Generate the local PRN codes of the candidate satellites with desired length, $c(t)$.

Step 2. Set the intermediate frequency as f_n.

Step 3. Generate the in-phase and the quadrature local reference signal by multiplying the code signal and the carrier signal:

$$\text{In-phase reference signal} = c(t)\sin(2\pi f_n t)$$

$$\text{Quadrature reference signal} = c(t)\cos(2\pi f_n t)$$

Step 4. Shift the incoming signal $s(t)$ by m ; $s(t+m)$.

Step 5. Multiply the shifted signal with the local reference signals and obtain $s_{in-phase} = s(t+m)c(t)\sin(2\pi f_n t)$ and $s_{quadrature} = s(t+m)c(t)\cos(2\pi f_n t)$.

Step 6. Take the squares of the in-phase and the quadrature components and add them. This is the acquisition metric value $acquisition\_metric(f_n, m)$ for that frequency bin and that code offset:

$$acquisition\_metric(f_n, m) = \left(s_{in-phase}\right)^2 + \left(s_{quadrature}\right)^2$$

Step 7. Return to step 4 until 1 millisecond length is scanned.

Step 8. Return to step 2 and cover all of the frequency bins.

Step 9. Apply the confirmation search with 25 Hz frequency steps.

Obtained $acquisition\_metric$ matrix is investigated and the candidate code shift and Doppler frequency values are determined. However, acquisition might be stopped if the magnitude of the $acquisition\_metric$ is larger than a pre-determined threshold.

Conventional acquisition method is the slowest method in GPS receiver design. Nevertheless, it is the most popular and well-known method since this method can be utilized in hardware or ASIC as pointed in [18].

3.2.2 Conventional Signal Tracking

Conventional approach for GPS signal tracking is based on the basic phased-locked loop concept. As known before, GPS signal is BPSK signal. It consists of two components of the carrier and the C/A code. On the other hand, conventional phase locked loop accepts a continuous wave signal as the input is actually inconvenient for GPS signal tracking. Hence, two loops are to be constructed. One phase locked loop for carrier signal tracking and the other one is the code tracking that is actually a delay locked loop. These two loops are coupled as shown in Figure 20.

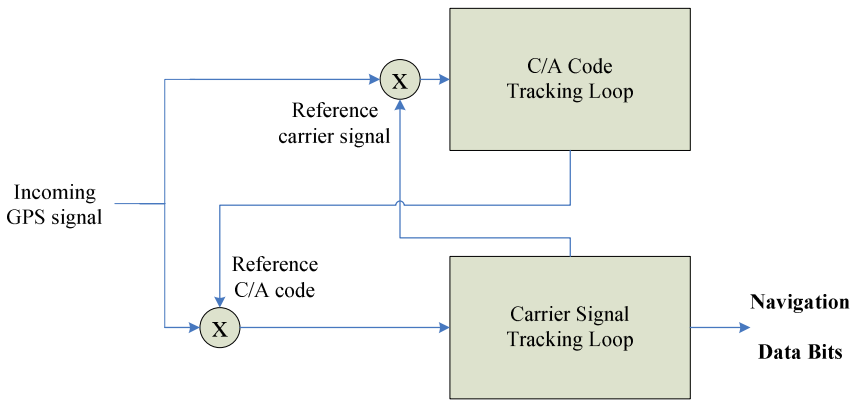

Figure 20. Combination of the code and the carrier tracking loops

3.2.2.1 Phase Locked Loop

As in [8] and [25], discussion of the tracking loops in GPS receivers should begin by investigating the generic tracking loop which is actually a phase lock loop. Figure 21 is the block diagram of a PLL in Laplace domain.

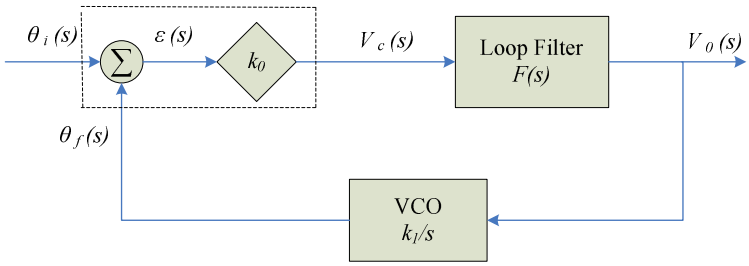

Figure 21. Basic PLL diagram

Error signal $\varepsilon(s)$ is obtained from the discriminator by using the input and the reference signal. Discriminator is illustrated by the dotted box in Figure 21. In this PLL configuration, discriminator implements the subtraction and the amplification operation of the input and the reference signals. Error signal is passed through a low-pass filter. Filter output is run through the voltage-controlled oscillator to produce the reference signal for the next step of the operation. Since the main goal of the PLL is determining the phase of the input signal as precisely as possible, we want to have $V_0(s)$ as close to zero as possible. In $V_0(s) = 0$ case, generated reference signal phase is equal to the input signal phase.

3.2.2.2 C/A Code Tracking Loop

Figure 22 is the block diagram of the implemented GPS code delay lock loop (DLL). Delay lock loop is similar to the generic phase locked loop structure of Figure 21. DLL discriminator aims to give an idea about the magnitude of the

delay according to the correlation of the generated code and the incoming GPS signals code.

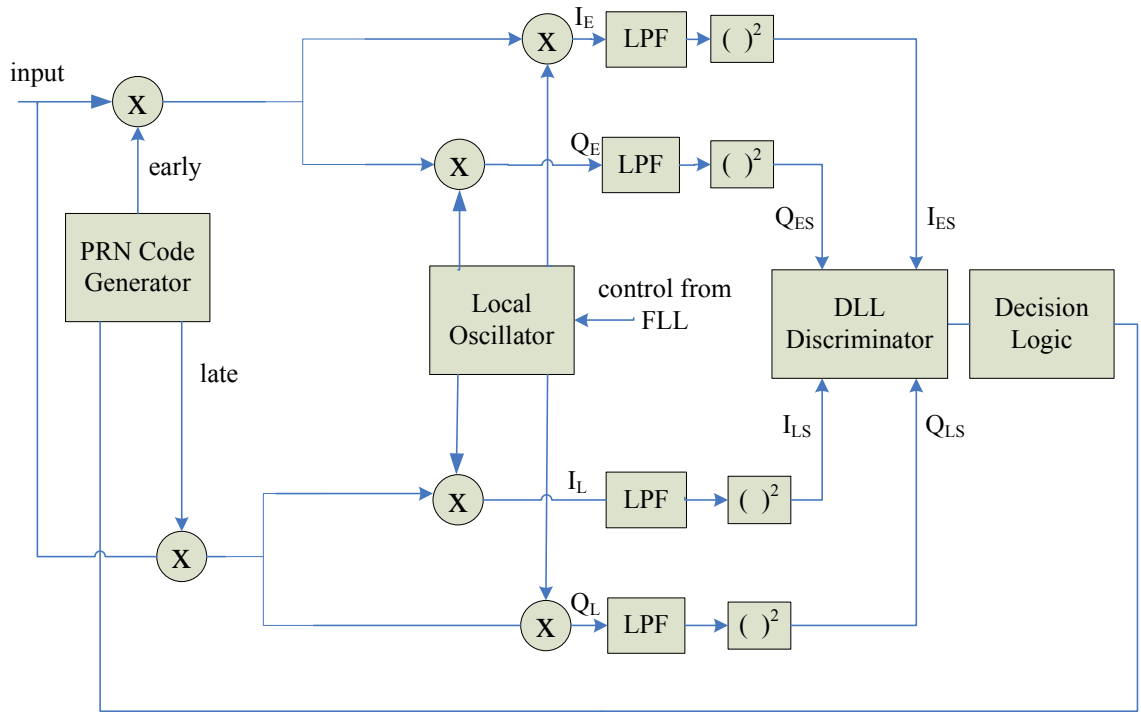

Figure 22. Conventional code tracking

In Figure 22, early and the late C/A codes are used. DLL discriminator compares the correlation value of the early and the late channels. Decision logic feeds the code generator to lock the code in the input signal. Decision logic will use a lock detector to decide whether the DLL is locked or not. If I_{PS} and Q_{PS} are the in-phase and the quadrature components for the prompt code, lock detector will be:

$$A_{PS} = \sqrt{I_{PS}^2 + Q_{PS}^2}$$

If A_{PS} is smaller than a pre-determined threshold, it means that satellite signal has been lost and the signal acquisition should be re-processed as shown in Figure 16.

Main goal of the DLL in code tracking is equalization of the energies of the early and the late signals as soon as possible. This is because; late and early code correlations are almost equal at the correct synchronization of the reference prompt code and the incoming code.

Time interval between the prompt and the early/late codes is a critical parameter of the DLL. Let us assume that $R_e(x)$ is the correlation function of the early code and $R_l(x)$ is the correlation function of the late code where x is the delay in terms of code chip. Idealized difference of $R_e(x)$ and $R_l(x)$ is illustrated in Figure 23. As observed from the figure, time interval should not be larger than 0.5 chip since the resolution is lost after 0.5 chip delay.

Table 2 ([3]) summarizes the typical delay locked loop discriminators. Pre-determined thresholds are used to decide to lead or lag the current prompt code for precise signal tracking. $I_{ES}, Q_{ES}, I_{LS}, Q_{LS}$ are as shown in Figure 22. I_{PS} and Q_{PS} are the in-phase and the quadrature components when we use the prompt code as the local code in Figure 22. These discriminators are chosen according to the application. Computational complexity and signal to noise ratio are the critical points in discriminator selection.

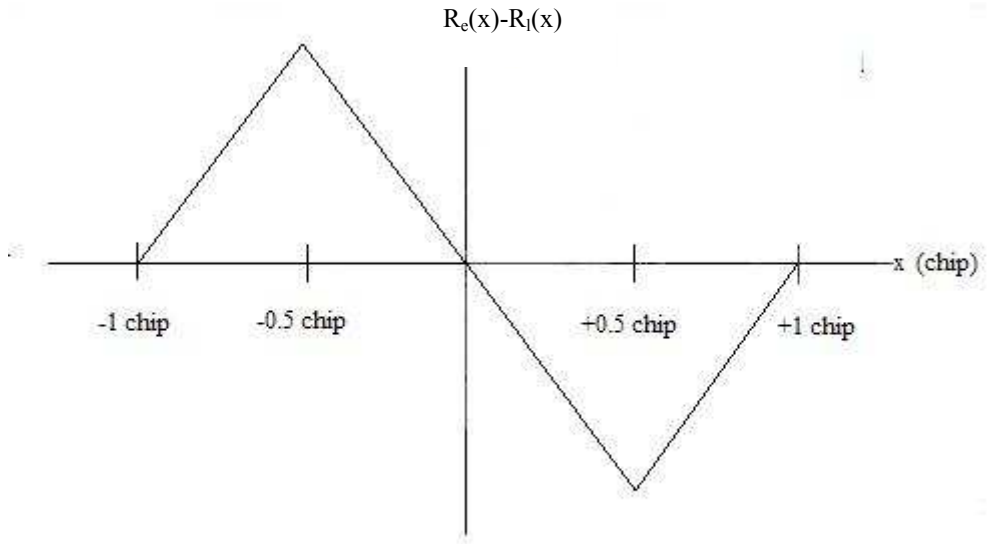

Figure 23. Correlation function difference of the early and the late codes

Table 2. Common Delay Locked Loop Discriminators

Discriminator	**Algorithm**
Dot product power	$\sum(I_{ES}-I_{LS})I_{PS}+\sum(Q_{ES}-Q_{LS})Q_{PS}$
Early minus late power	$\sum(I_{ES}^2+Q_{ES}^2)-\sum(I_{LS}^2+Q_{LS}^2)$
Early minus late envelope	$\sum\sqrt{(I_{ES}^2+Q_{ES}^2)}-\sum\sqrt{(I_{LS}^2+Q_{LS}^2)}$
Normalized early minus late envelope	$\frac{\sum\sqrt{(I_{ES}^2+Q_{ES}^2)}-\sum\sqrt{(I_{LS}^2+Q_{LS}^2)}}{\sum\sqrt{(I_{ES}^2+Q_{ES}^2)}+\sum\sqrt{(I_{LS}^2+Q_{LS}^2)}}$

3.2.2.3 Carrier Frequency Tracking Loop

Costas PLL accomplishes tracking of the carrier frequency. However, in our study we implement a frequency locked loop (FLL). FLL is more convenient for software processing since direct frequency update is possible to track the carrier signal. Furthermore, the capture range of a FLL is typically much larger than that of a PLL as [6] describes. A FLL produces frequency error to track the input carrier, unlike a PLL, which uses phase error. In this section, both of the phase and the frequency locked loops are discussed in detail.

3.2.2.3.1 Costas Loop

A Costas-type or equivalent PLL is used to prevent the phase loss. Frequency search keeps on until the numerically controlled oscillator (NCO) is locked to the phase of the incoming GPS signal, the measured phase rate will typically be in the range of ±5 kHz due to the signal Doppler shift. Figure 24 is the typical GPS carrier tracking loop which has the PLL structure as in Figure 21.

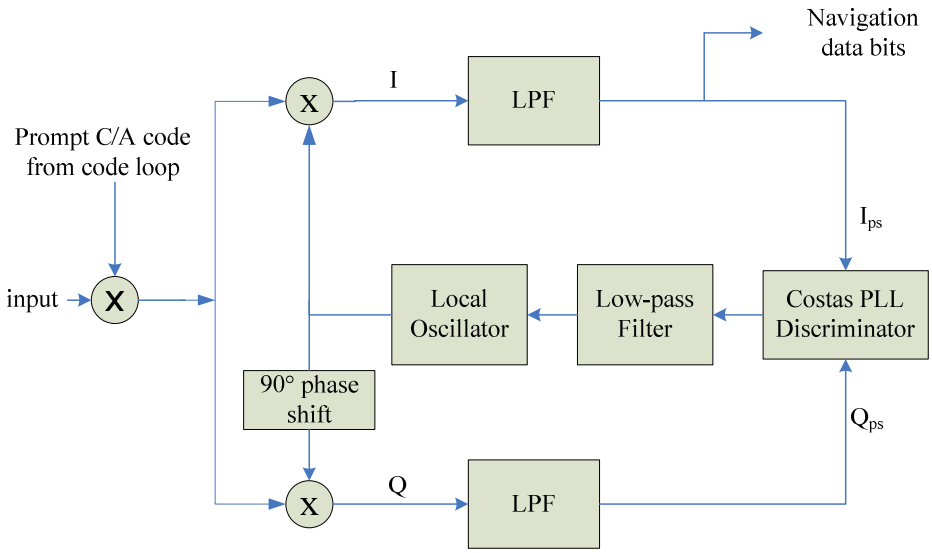

Figure 24. Conventional carrier tracking by using Costas PLL

50-Hz navigation data demodulation is obtained directly from the I channel in the Costas loop. Costas loops are insensitive to 180º phase shift due to the navigation bit transition, which provides safe operation if the pre-detection integration time includes bit phase reversal case. Discriminator of the Costas loop calculates the phase error between the I_{ps} and Q_{ps} . I_{ps} and Q_{ps} are derived by using *s(t)* in the equation (2.2):

$$I_{ps} = s(t)\cos(2\pi f_{local}t)c(t) \tag{3.3}$$

$$Q_{ps} = s(t)\sin(2\pi f_{local}t)c(t) \tag{3.4}$$

where:

$s(t)$: Incoming GPS signal,

$c(t)$: C/A code,

f_{local} : local reference signal frequency.

By substituting the equation (2.1) in (3.3) and (3.4), we obtain:

$$I_{ps} = \sqrt{2P_I}\,d(t)c(t)\cos(2\pi f_{L_1}t)\cos(2\pi f_{local}t)c(t) + \sqrt{2P_Q}\,d(t)p(t)\sin(2\pi f_{L_1}t+\theta)\cos(2\pi f_{local}t)c(t) \tag{3.5}$$

$$Q_{ps} = \sqrt{2P_I}\,d(t)c(t)\cos(2\pi f_{L_1}t)\sin(2\pi f_{local}t)c(t) + \sqrt{2P_Q}\,d(t)p(t)\sin(2\pi f_{L_1}t+\theta)\sin(2\pi f_{local}t)c(t) \tag{3.6}$$

Multiplication $c(t)p(t)$ is at the level of noise since P code is a longer pseudo random code than the C/A code. If correct synchronization of the C/A code is assumed, i.e. $|c(t)|^2 = 1$, the equations (3.5) and (3.6) become:

$$I_{ps} = \sqrt{2P_I}\,d(t)\cos(2\pi f_{L_1}t)\cos(2\pi f_{local}t) \tag{3.7}$$

$$Q_{ps} = \sqrt{2P_I}\,d(t)\cos(2\pi f_{L_1})\sin(2\pi f_{local}t) \tag{3.8}$$

In ideal lock of the Costas loop, locally generated signal is pulled up to the incoming signal frequency, i.e. $f_{local} = f_{L_1}$

$$I_{ps} = \sqrt{2P_I}\, d(t)\cos^2(2\pi f_{local} t) \tag{3.9}$$

$$Q_{ps} = \frac{\sqrt{2P_I}}{2} d(t)\sin(2\pi 2 f_{local}) \tag{3.10}$$

If we compute the averaged value of the equations (3.9) and (3.10) over $(-\pi, +\pi)$, we have $Q_{ps} = 0$ and I_{ps} maximum. Thus, phase error between I_{ps} and Q_{ps} is zero as expected.

According to the above discussion, Costas PLL tries to make I_{ps} maximum whereas Q_{ps} is tried to be pulled down its minimum during the carrier frequency trace in the tracking process. Phasor diagrams of I_{ps} and Q_{ps} are presented in Figure 25. *A* is the vector sum of I_{ps} and Q_{ps} and it will flip to *–A* (by 180º*)* each time the data bit changes sign. In Costas PLL operation, *A* tends to align with the *I* axis in order to have Q_{ps} minimum and I_{ps} maximum. Therefore, phase error can be zero as desired.

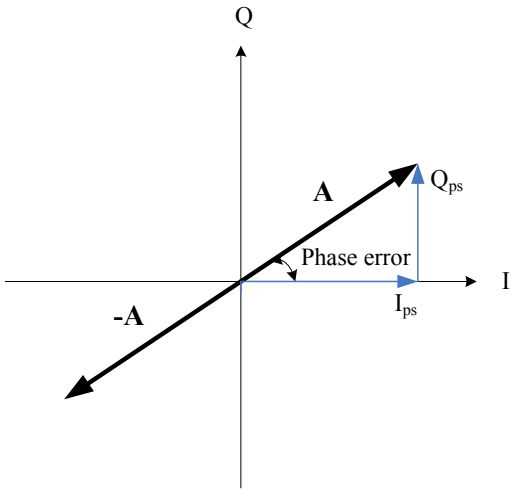

Figure 25. Phasor diagram for Costas PLL

Some features of a Costas loop are discriminators expressed in Table 3 ([3]). In general, two-quadrant arctangent discriminator is selected since it is optimal.

Table 3. Common Costas Loop Discriminators

Discriminator algorithm	**Output Phase Error**	**Characteristic**
$Sign(I_{ps}) \cdot Q_{ps}$	$\sin \Theta$	Near optimal at high SNR
$I_{ps} \cdot Q_{ps}$	$\sin 2\Theta$	Near optimal at low SNR
Q_{ps} / I_{ps}	$\tan \Theta$	Suboptimal, but good at high and low SNR
$arctangent(Q_{ps} / I_{ps})$	Θ	Two-quadrant arctangent. Optimal at high and low SNR

3.2.2.3.2 Frequency Locked Loop (FLL)

As in [3], frequency locked loop is nothing but a phase lock loop which uses frequency error to track the incoming GPS carrier signal. FLL replicates the approximate frequency of the carrier signal. In Figure 26, FLL discriminator outputs the frequency error between the input signal and the reference signal.

In frequency pulling, I_{ps} and Q_{ps} are calculated at t_1 and t_2 and denoted by I_{ps1}, I_{ps2}, Q_{ps1} and Q_{ps2}. Time difference of t_2 - t_1 is called as pre-detection interval and can be either 5 or 10 milliseconds. Therefore, pre-detection bandwidth may be 200 Hz or 100 Hz. Phasor diagram of the vectors is illustrated in Figure 27, in [3]. Phasor A (vector sum of I and Q) remains constant and rotates at a rate directly proportional to the frequency error when the loop is in frequency lock.

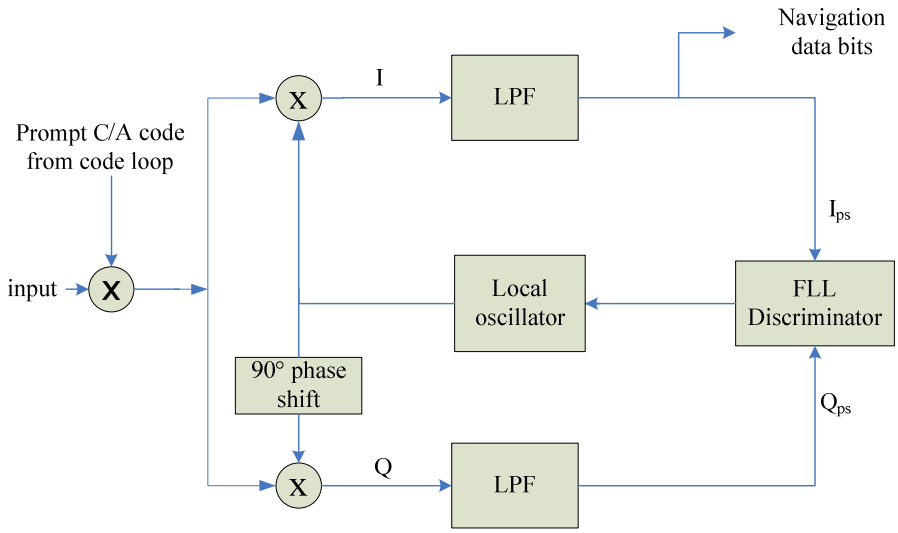

Figure 26. Conventional carrier tracking by using FLL

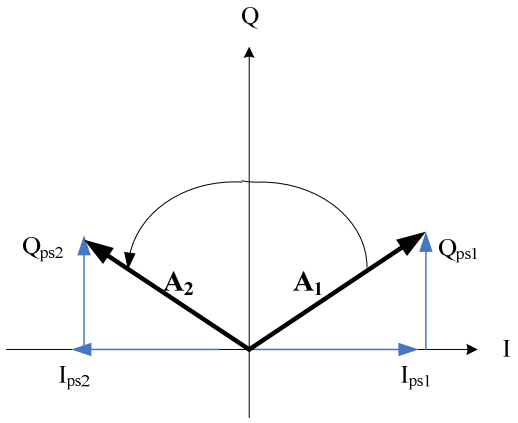

Figure 27. Phasor diagram for FLL

True frequency error is computed by the phase difference of the A_1 and A_2 divided by the pre-detection interval. Cross and dot products of the A_1 and A_2 can be used to compute phase difference. FLL discriminator produces this frequency error. Table 4 in [3] contains the commonly used FLL discriminators and their properties.

Table 4. Common frequency locked loop discriminators

Discriminator algorithm	Output Frequency Error	Characteristic
Sign(dot) . cross / (t_2-t_1)	*sin [2$(\Theta_2-\Theta_1)$] / (t_2-t_1)*	Near optimal at high SNR
cross / (t_2-t_1)	*sin [$(\Theta_2-\Theta_1)$] / (t_2-t_1)*	Near optimal at low SNR
arctangent(cross/dot) / (t_2-t_1)	*$(\Theta_2-\Theta_1)$ / 360(t_2-t_1)*	Optimal four quadrant arctangent

where:

(I_{psi}, Q_{psi}) : prompt in-phase and quadrature components at time t_i,

dot : $I_{ps1} . I_{ps2} + Q_{ps1} . Q_{ps2}$,

cross : $I_{ps1} . Q_{ps2} - I_{ps2} . Q_{ps1}$,

Θ_1 : Phase difference of the I_{ps1} and Q_{ps1},

Θ_2 : Phase difference of the I_{ps2} and Q_{ps2}.

Calculated output frequency error is used for updating the carrier signal. Thus, carrier tracking is performed.

3.3 Block Adjustment of Synchronizing Signal (BAAS) Signal Processing

BAAS signal processing is used for the software GPS receiver. Signal acquisition is achieved by circular correlation. After the signal is acquired, BAAS tracking approach traces the signal in frequency domain.

3.3.1 Acquisition by Circular Correlation

As stated in [8] and [19], circular correlation acquisition method is suitable for software receiver approach. Circular correlation is actually based on circular

convolution, which is periodic convolution process. Hence, it is convenient to concentrate on circular correlation concepts.

In a discrete time system, the output is found from the input either in time domain through convolution or in frequency domain by a multiplication through Fourier transforms. If the system input is *x(n)* and the impulse response of the system is *h(n)*, output *y(n)* is found as $y(n) = \sum_{m=0}^{N-1} x(m)h(n-m)$ in time domain and $Y(k) = X(k)H(k)$ in frequency domain.

The acquisition algorithm uses the circular correlation, which is different from the circular convolution. A correlation between *x(n)* and *h(n)* is written as $y(n) = \sum_{m=0}^{N-1} x(m)h(n+m)$ in time domain. In order to reach the frequency domain representation of correlation, following equation derivation is done:

$$Y(k) = \sum_{n=0}^{N-1}\sum_{m=0}^{N-1} x(m)h(n+m)e^{(-j2\pi kn)/N} = \sum_{n=0}^{N-1} x(m)\left[\sum_{m=0}^{N-1} h(n+m)e^{(-j2\pi k(n+m))/N}\right]e^{(-j2\pi k(-m))/N}$$

$$Y(k) = H(k)\sum_{n=0}^{N-1} x(m)e^{(j2\pi km)/N} = H(k)\left[\sum_{n=0}^{N-1} x^*(m)e^{(-j2\pi km)/N}\right]^* = H(k)X_*(k)$$

where $X_*(k)$ is the Fourier transform of the $x^*(n)$ (conjugate of the $x(n)$). Note that, if the signal *x(n)* is real than the conjugate of the signal is equal to itself, i.e. $x^*(n) = x(n)$ and so we have $X_*(k) = X^*(k)$. Hence, the magnitude of $Y(k)$ can be derived as:

$$\left|Y(k)\right| = \left|X^*(k)H(k)\right| = \left|H^*(k)X(k)\right|$$

This result is directly used to find the correlation in frequency domain. Frequency domain operation completes all the shift values of *h(n)* or *x(n)*. This property reduces the computational complexity of the acquisition process.

Consequently, the correlation (acquisition metric value) of the received signal and local reference signal can be calculated as:

$$acquisition\_metric(f_n,:) = IFFT[FFT[s(n)] * CONJ(FFT[c(n)e^{-j2\pi f_n/N}])] \quad (3.11)$$

acquisition_metric(f_n,:) : acquisition metric at f_n for all possible code shifts *m*,

s(n) : GPS signal,

$c(n)e^{-j2\pi f_n/N}$: Local reference signal, which is obtained by the multiplication of C/A pseudo code and carrier signal including both of the in-phase and the quadrature components.

f_n : carrier frequency bin,

FFT : Fast Fourier Transform,

IFFT : Inverse FFT operation,

CONJ : Conjugate operation.

Circular correlation method provides high-speed acquisition. Three FFT operations in equation (3.11) complete the PRN code offset search at one frequency bin. Thus, only frequency is changed successively over the range -10 kHz to 10 kHz with 1 kHz steps. Figure 28 illustrates the computation of the *acquisition_metric(f_n,m)*. Block diagram of the circular correlation method is presented in Figure 29 given in [18].

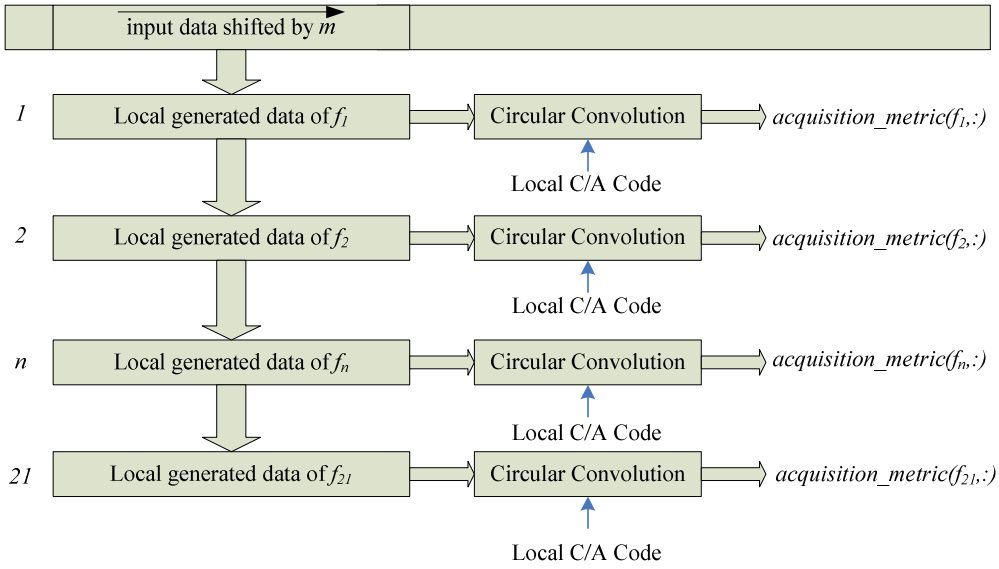

Figure 28. Circular correlation process

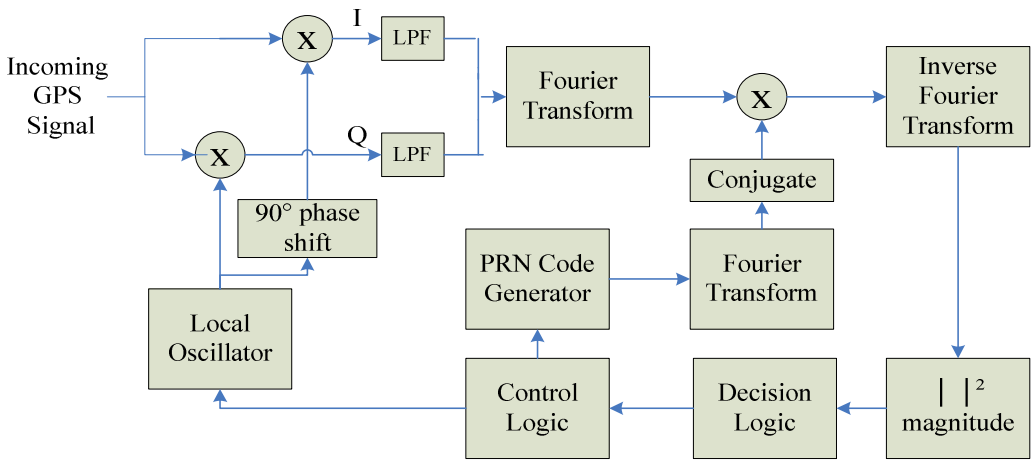

Figure 29. Circular correlation acquisition loop

Signal acquisition by circular convolution software program follows the steps given below:

Step 1. Obtain the FFT of the 1 millisecond duration of the incoming GPS signal, *s(n)*. Transform the input to the frequency domain. Hence, we have *FFT{s(n)}=S(k)*. Next, take the complex conjugate of *S(k)* and obtain $S^*(k)$.

Step 2. Set the carrier frequency as f_n

Step 3. Generate the local code by multiplying C/A code of the satellite-*c(n)* and the complex RF carrier signal at the searched frequency f_n, i.e. $l(n) = c(n)e^{-j2\pi f_n / N}$.

Step 4. Take the FFT of the locally generated signal $l(n)$ and obtain $L(k)$.

Step 5. Perform circular correlation in frequency domain by multiplication of $S^*(k)$ and $L(k)$. Thus, $R(k) = S^*(k)L(k)$.

Step 6. To find the absolute value of $|r(n)|$ take the inverse FFT of $R(k)$.

Step 7. The maximum of $|r(n)|$ in the n^{th} location of the frequency f_n gives the candidate beginning of C/A code in the input data.

$$acquisition\_metric(f_n, m) = \max\{|r(n)|\}\,|_{n=m}$$

Step 8. Return to step 2 until 21 frequency steps from -10 kHz to 10 kHz are covered. Frequency bin size is 1 kHz.

Step 9. Find the maximum of the $acquisition\_metric(f_n, m)$ in order to determine the code beginning and carrier frequency.

3.3.2 Fine Frequency Estimation

Since we increase the efficiency by using circular correlation, it is convenient that fine frequency estimation algorithm is also performed according to [8] and [19]. By fine frequency estimation, frequency resolution is decreased to a few tens of Hertz from 1 kHz resolution of the search algorithms. Increased frequency resolution would increase the sensitivity of the tracking loops.

Fine frequency estimation is based on phase relation of the signal. Incoming signal becomes a continuous wave signal when the C/A code is stripped. Let us assume that the maximum frequency component of *s(n)* occurs at time t_m. So, $S_m(k)$ is maximum with *k* denoting the frequency component. Phase of $S_m(k)$ can be found from the equation below:

$$\theta_m(k) = \tan^{-1}\left(\frac{\mathrm{Im}(S_m(k))}{\mathrm{Re}(S_m(k))}\right) \quad (3.12)$$

Let us assume that a short time after time t_m at time t_n, $S_n(k)$ is the maximum. Phase of the DFT component of $S_n(k)$, $\theta_n(k)$ can be obtained as in (3.12)

Fine frequency is found by comparing the phases $\theta_m(k)$ and $\theta_n(k)$ of the DFT component of the input signal from which C/A code is stripped. Therefore, fine frequency estimation is achieved by:

$$f_{fine} = \frac{\theta_n(k) - \theta_m(k)}{2\pi(t_n - t_m)} \quad (3.13)$$

$$f_{fine} = \frac{\Delta\theta}{2\pi(t_n - t_m)} \quad (3.14)$$

where $\Delta\theta$ is the phase difference. It should be in the range of $[0, 2\pi]$. Frequency ambiguity and phase shift due to the navigation data should be eliminated from the phase difference in the equation (3.14). Below discussion explains the corrections to be applied to the phase difference.

1. Frequency ambiguity elimination: Let us assume that the DFT component at the frequency *k* is *S(k)*. However, input signal frequency may not be exactly *k*. This case is called as frequency ambiguity. If the input frequency falls into between *S(k-1)* and *S(k)*, this ambiguity should be eliminated for efficient use of the equation (3.12). Amplitude comparison scheme is one of the ambiguity

elimination approaches. According to this approach, *S(k-1)* and *S(k)* should be compared. In case *S(k-1)* is larger than *S(k)* in magnitude, the input frequency falls into between *S(k-1)* and *S(k)*. Hence, input signal frequency is searched again in the range of [*k-1*, *k*]. Afterwards, corrected frequency component is used to calculate the phase, which will be used in the equation (3.14).

2. Phase shift due to the navigation data elimination: Navigation data is observed in the carrier signal as a 180-degree phase shift between two consecutive data. Therefore, this navigation data phase shift should be taken into account in the equation (3.14). In navigation data change, phase difference would also include π phase shift. To prevent this condition, the maximum frequency uncertainty must be less than 250 Hz since 250 Hz corresponds to $\pm\pi/2$ phase difference as explained in [8]:

$$250Hz = \frac{\Delta\theta}{2\pi(1ms)} \Rightarrow \Delta\theta = 0.5\pi = \frac{\pi}{2}$$

So, the maximum frequency difference of ±200 Hz is convenient to be selected. Hence, the maximum phase difference is $2\pi/5$ as derived below:

$$200Hz = \frac{\Delta\theta}{2\pi(1ms)} \Rightarrow \Delta\theta = 0.4\pi = \frac{2\pi}{5}$$

If a $\pm\pi$ phase shift due to the navigation data occurs, phase angle becomes $\Delta\theta = \pm\frac{2\pi}{5}\,\mu\,\pi = \pm\frac{3\pi}{5}$. Hence, if the phase difference is greater than $2\pi/5$, π can be subtracted from the result to keep the phase difference less than $2\pi/5$. When the noise is taken into account, $2\pi/5$ threshold is extended to $2.3\pi/5$ as appointed in [8].

The steps for circular convolution signal acquisition including fine frequency estimation can be outlined as follows:

Step 1. Execute signal acquisition by circular convolution on one period of C/A code. Therefore, C/A code beginning point and the carrier frequency within 1 kHz resolution is obtained.

Step 2. If the highest frequency component is *S(k)*, find out the two DFT values *S(k-1)* (400 Hz lower than *S(k)*) and *S(k+1)* (400 Hz larger than *S(k)*). Maximum output from the set {*S(k-1),S(k),S(k+1)*} is the new *S(k)*. If the input signal is in the middle of the two adjacent *k* values, the input signal is 400 Hz / 2= 200 Hz from both of the *k* values. By this arrangement, we set the maximum frequency difference as 200 Hz.

Step 3. In order to find out the fine frequency, consider 5 ms long consecutive data starting from the beginning of the C/A code.

Step 4. Obtain all the $S_n(k)$ values for all $n = 1, \ldots, 5$. Next, find the phase difference angle values as $\Delta\theta_i = \theta_{n+1} - \theta_n$ for all n. Note that $i = 1, 2, 3, 4$.

Step 5. Phase difference angle $\Delta\theta_i$ should be kept below $2.3\pi/5$ in order to eliminate the frequency ambiguity as explained before. If this constraint is not met, 2π can be added or subtracted from $\Delta\theta_i$. If the constraint is still not met, phase shift due to the navigation data elimination should be performed. So, π is added or subtracted from $\Delta\theta_i$.

Step 6. There will be 4 sets of $\{\Delta\theta_1, \Delta\theta_2, \Delta\theta_3, \Delta\theta_4\}$ and the average value of the set will be the phase difference, i.e. $\Delta\theta = \frac{\Delta\theta_1 + \Delta\theta_2 + \Delta\theta_3 + \Delta\theta_4}{4}$. Fine frequency is calculated according to the equation (3.14) and that will be added to the coarse frequency obtained from the circular correlation.

3.3.3 BAAS Tracking

BAAS (block adjustment of synchronizing signal) approach to track the GPS signal aims to build a software algorithm for the civil L_1 signal frequency. In this approach, which is presented in [8] and [19], one frequency component of the discrete Fourier transform (DFT) is used to determine the carrier signal frequency and the C/A code delay. This is actually the same as the conventional FLL and DLL operation. However, BAAS tracking has less computational complexity than the conventional tracking. BAAS tracking traces the GPS signal in frequency domain. On the other hand FLL and DLL operate in time domain.

3.3.3.1 BAAS Carrier Tracking Algorithm

In BAAS carrier tracking algorithm, consecutive phase measurements are used to track the carrier frequency. Fine frequency component of the incoming GPS signal is computed according to equation (3.14). Hence, carrier tracking algorithm uses this equation as a frequency update and aims to trace the carrier signal in the received signal. Carrier tracking consists of two steps:

Step 1. Generate the local signal. The local signal contains two sets of locally generated data: the C/A code of the satellite and the complex carrier signal. C/A code is generated for one time and it is stored and used repetitively. The complex carrier signal must be generated at the correct frequency and the phase should be continuous. Decoding process depends only on the difference in phase between consecutive phase measurements as in the equation (3.14).

Step 2. Adjust the carrier frequency. Frequency adjustment is not achieved for every millisecond but for every 10 milliseconds. Fine frequency is calculated by:

$$f_{fine} = \frac{\delta\theta}{2\pi(t_n - t_m)} \tag{3.15}$$

In this equation $\delta\theta$ is the phase angle of the two consecutive milliseconds of data. After 10 milliseconds, we have 9 phase angle

values. These phase angles are corrected according to the Step 5 of the circular correlation acquisition process. After the phase angles are corrected, their average value is taken as the phase difference to be used in fine frequency estimation in equation (3.15) by having $(t_n - t_m) = 1ms$. Once the carrier signal frequency is updated according to the fine frequency value, it will be used for the next set 10 milliseconds of the GPS data.

3.3.3.2 BAAS Code Tracking Algorithm

C/A code algorithm is to find the shift value relative to the generated C/A code, which is actually the fine pseudorange measurement. Algorithm is applied as fallows:

Step 1. Generate early, prompt, and late local signals. Each of these signals is the multiplication of the code and the complex carrier signal. For example, prompt signal is the product of the prompt C/A code and the complex carrier signal. Late signal can be easily obtained by shifting the prompt signal two samples right. Early signal is obtained by shifting prompt signal by two samples left. Since the magnitude of the DFT components of the local signals are considered, phase of the carrier does not matter for code tracking.

Step 2. Adjust the code for every 10 milliseconds. Ratio below determines the adjustment on the code:

$$R = \frac{A_{late}}{A_{early}} \tag{3.16}$$

where A_{early} and A_{late} are the DFT magnitude of the early and the late signals, respectively. If the ratio R is above the code threshold, late signal is more correlated than the early signal. Conversely, if the ratio is below the lower code threshold or above the upper code threshold, early signal is more correlated than the late signal.

Code threshold determination: In order to determine the upper and the lower code threshold, an approximation is taken into account. Approximation of the auto-correlation function of the C/A code is shown in Figure 30.

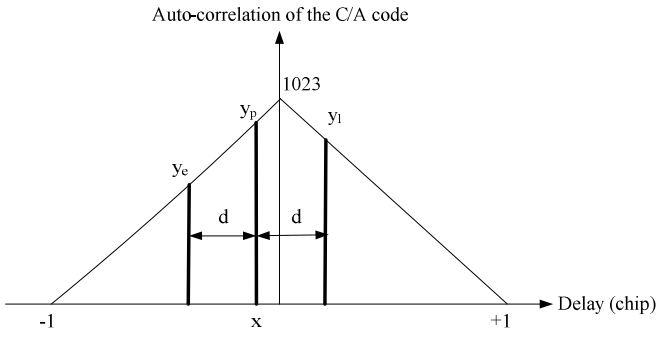

Figure 30. Correlation lag in unit of chip (case 1)

As shown in Figure 30, the maximum correlation is assumed 1023 and the minimum correlation is zero. y_p, y_e and y_l in the figure are computed as the following:

$$\begin{aligned} y_p &= 1023(1-|x|) \\ y_l &= 1023(1-|x+d|) \\ y_e &= 1023(1-|x-d|) \end{aligned} \tag{3.17}$$

Where d is the distance between the prompt and the early/late C/A code and x is the fine time resolution in code delay. Fine time resolution is important to calculate the pseudorange precisely. In the figure, x is negative and the absolute values are determined as below:

$$\begin{aligned} y_p &= 1023(1+x) \\ y_l &= 1023(1-x-d) \\ y_e &= 1023(1+x-d) \end{aligned}$$

$$r = \frac{y_l}{y_e} = \frac{1-x-d}{1+x-d} \Rightarrow x = \frac{(1-r)(1-d)}{1+r}.$$

According to the equation above, one can find the lower and the higher threshold for *R* in (3.16). Let us assume that the sampling frequency of the signal is f_s and $d = \frac{(2/f_s)}{4}$. Thus, half the sampling time, which is $1/(f_s/2)$, actually shows the range of the value of *x*. So, $x \in \left(-\frac{2}{f_s}, \frac{2}{f_s}\right)$.

Code threshold is obtained as the following equations by substituting the maximum and the minimum values of x:

$$r_{high} = \frac{(y_l)_{\max}}{(y_e)_{\max}} = \frac{1-x_{\min}-d}{1+x_{\min}-d} = \frac{1+\frac{2}{f_s}-\frac{2}{4f_s}}{1-\frac{2}{f_s}-\frac{2}{4f_s}} = \frac{1+\frac{6}{4f_s}}{1-\frac{10}{4f_s}}$$

$$r_{low} = \frac{(y_l)_{\min}}{(y_e)_{\min}} = \frac{1-x_{\max}-d}{1+x_{\max}-d} = \frac{1-\frac{2}{f_s}-\frac{2}{4f_s}}{1+\frac{2}{f_s}-\frac{2}{4f_s}} = \frac{1-\frac{10}{4f_s}}{1+\frac{6}{4f_s}} = \frac{1}{r_{high}} \quad (3.18)$$

It is important that 1 in the equation (3.18) stands for one chip duration that is 1 milliseconds/ 1023 = 955.7 nanoseconds.

More on fine time resolution *x* : In *x* computation, situations in Figure 31 and Figure 32 should be taken into account. Equations (3.19) and (3.20) show the *x* computation in such cases. Fine resolution of pseudorange should be recorded for every 10 milliseconds.

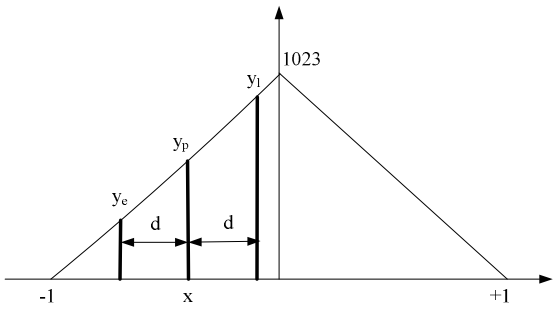

Figure 31. Correlation lag in unit of chip (case 2)

$$\begin{aligned} y_p &= 1023(1+x) \\ y_l &= 1023(1+x+d) \\ y_e &= 1023(1+x-d) \end{aligned} \Rightarrow r = \frac{y_l}{y_e} = \frac{1+x+d}{1+x-d} \Rightarrow$$

$$x = d\frac{r+1}{r-1} - 1 \qquad (3.19)$$

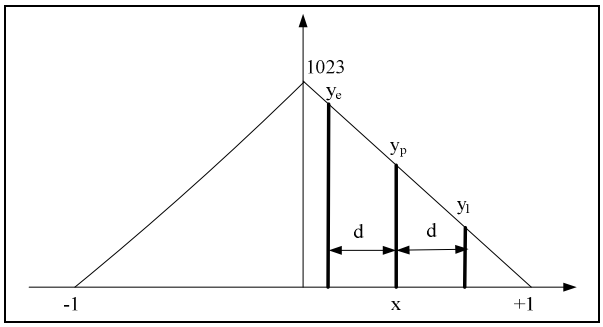

Figure 32. Correlation lag in unit of chip (case 3)

$$\begin{aligned} y_p &= 1023(1-x) \\ y_l &= 1023(1-x-d) \\ y_e &= 1023(1-x+d) \end{aligned} \Rightarrow r = \frac{y_l}{y_e} = \frac{1-x-d}{1-x+d} \Rightarrow x = 1 + d\frac{r+1}{r-1} \qquad (3.20)$$

Step 3. A_{prompt} is the DFT magnitude of the prompt signal. It is used as the lock indicator. If A_{prompt} is below a pre-determined threshold,

acquisition process should be started as explained in Figure 16. Above the threshold tracking goes on.

Example: In this example, values used in implemented BAAS algorithm are computed. If the sampling frequency of the GPS signal is $f_s = 16.035MHz$ half of the sampling time is $2/f_s = 2/16.035MHz$. So, $x = \pm 1/f_s$ and $d = 2/f_s$. After normalizing x to the chip time, one can find the worst mismatched condition $y_p \approx 1023(1 - x/977.5) = 990.36$ from equation (3.17). It is actually $20\log(990.36/1023) =$ -0.2816 dB. Therefore, the worst situation is that the correlation peak is approximately 0.3dB less than the ideal case.

CHAPTER 4

GPS POSITIONING

GPS is an open-air satellite based navigation system that provides precise positioning and timing. Precise navigation is obtained by processing the navigation data broadcasted from satellite to user by spread spectrum modulation technique. When the navigation data and GPS measurements are obtained, following main steps must be accomplished for positioning:

1) Correcting the raw pseudorange - the distance from the satellite to the user - as precise as possible,
2) Computing the satellite position precisely,
3) Determining the PVT (position-velocity-time) information.

In this chapter, all of the three navigation algorithm steps are explored in detail.

4.1 GPS Navigation Data Structure and Content

GPS navigation data is transmitted in 50 bps bit rate. Navigation bits are addressed into word, subframe, frame and superframe structures as presented in Table 5.

Table 5. GPS Data Structures

Structure	Bit number	Time duration
1 word	30 bits	600 milliseconds
1 subframe	10 words (300 bits)	6 seconds
1 frame	5 subframes (1500 bits)	30 seconds
1 superframe	25 frames (1500 x 25 bits)	12.5 minutes

Word Structure: Word is the smallest structure that contains 30 bits. Navigation bits are placed between 1st and 24th bits, remaining 6 bits are used for parity check. Parity check algorithm is to correct the polarity of the bits in the word.

Subframe Structure: Navigation data subframe structure is presented in two groups as common and uncommon properties:

a) Common Properties of Subframes

The first and the second words of each subframe are called TLM (telemetry) and HOW (handover) word. TLM and HOW bit streams are expressed in Table 6.

Table 6. TLM and HOW words structures

TLM word	8 bit Preamble : '100010 11'	16 reserved bits			6 parity check bits	Total: 30 bits
HOW word	17 bit TOW	Bit 18: (anti-spoof) Bit 19: (alert bit)	(20.,21.,22). bits=Subframe id Ex.:(011)→(3rd subframe)	23. and 24. bits are reserved	6 parity check bits	Total: 30 bits

Notes:

1. TOW (time of week) information includes GPS time in 6 seconds resolution.
2. TOW obtained in one subframe is the time of week data of the next subframe. In order to find the TOW of a subframe, 6 seconds should be subtracted from the TOW in the current HOW word since one subframe duration is 6 seconds.
3. If bit 18 is 1, then URA (user range accuracy) is not healthy.
4. If bit 19 is 1, then anti-spoof mode is open.

b) Uncommon Properties of Subframes

Table 7 shows the contents of each subframe in GPS navigation data.

Table 7. GPS Navigation Subframes

Subframe	Content
1	Satellite clock correction data
2	Ephemeris
3	Ephemeris
4	Almanac
5	Almanac

Navigation parameters in the subframe 1, 2 and 3 are presented in Table 8, Table 9 and Table 10. These parameters should be converted from binary to decimal. 2's complement form is used for some cases during this conversion.

Frame Structure

Frame consists of subframes 1, 2, 3, 4 and 5. Subframes 1, 2 and 3 have the same format from frame to frame. However, subframe 4 and 5 have 25 pages or different sets of data and contain the almanac. Almanac contains the approximate satellite ephemeris, clock correction and space vehicle status for all of the satellites.

Table 8. Subframe-1 navigation parameters

Parameter	Bit number	Scale Factor
T_{gd} (estimated group delay differential)	8*	2^{-31}
t_{oc} (reference time for clock correction)	16	2^{4}
a_{f2} (satellite clock correction)	8*	2^{-55}
a_{f1} (satellite clock correction)	16*	2^{-43}
a_{f0} (satellite clock correction)	22*	2^{-31}
TOW (time of week)	17	x6-6

*2's complement format and sign bit is MSB.

Table 9. Subframe-2 navigation parameters

Parameter	Bit number	Scale Factor
Δ_n (mean motion difference from computed value)	16*	2^{-43}**
M_o (mean anomaly at reference time)	32*	2^{-31}**
e_o (eccentricity of the orbit)	32	2^{-33}
$\sqrt{A}$ (square root of the semi-major axis)	32	2^{-19}
t_{oe} (reference time for ephemeris)	16	2^{4}
C_{rs} (amplitude of the sine harmonic correction term to the orbit radius)	16	2^{-5}
C_{uc} (amplitude of the cosine harmonic correction term to the argument latitude)	16*	2^{-29}
C_{us} (amplitude of the sine harmonic correction term to the argument latitude)	16*	2^{-29}

*2's complement format and sign bit is MSB.

**parameter unit is semicircle.

Table 10. Subframe-3 navigation parameters

Parameter	Bit number	Scale Factor
C_{ic} (amplitude of the cosine harmonic correction term to the angle of inclination)	16*	2^{-29}
C_{is} (amplitude of the sine harmonic correction term to the angle of inclination)	16*	2^{-29}
C_{rc} (amplitude of the cosine harmonic correction term to the orbit radius)	16*	2^{-5}
Ω_0 (longitude of ascending node of orbit plane at reference time)	32*	2^{-31}**
i_o (inclination angle at reference time)	32*	2^{-31}**
ω (argument of perigee)	32*	2^{-31}**
$\dot{\Omega}$ (rate of right ascension)	24*	2^{-43}**
IDOT (rate of inclination angle)	14*	2^{-43}**

*2's complement format and sign bit is MSB.

**parameter unit is semicircle.

Superframe Structure

Superframe, consisting of 25 frames includes all relevant navigation data for a satellite including ephemeris parameters, satellite clock correction parameters and the almanac.

4.2 GPS Measurements

GPS positioning and velocity designation is possible if and only if the following main information is provided by at least 4 satellites:

1. Precise satellite positions that have been tracked by the receiver,

2. Satellites corrected pseudoranges,

3. Satellites Doppler frequency.

In order to obtain the first two main information time of transmission, which is time at the satellite when the GPS signal is sent, is very important. Hence, time of transmission will be discussed first. Afterwards, pseudorange correction and satellite position calculation will be explained.

4.2.1 Time of transmission (TOT) of GPS signal from the satellite

Theoretically, time of transmission can be considered as a combination of two components as follows:

t : desired transmission time,

TOW : time of week data in the first subframe

$$t = TOW\ component + Fine\ tracking\ component$$

TOW is the time of week data. It is decoded from the binary navigation message from the satellite. On the other hand, tracking component of the TOT is obtained from the tracking unit of the receiver, which is actually the C/A code shift value to synchronize the generated code by the receiver with the GPS signal.

All GPS satellites broadcast GPS signals at the same time except from satellite clock errors. However, since the range between the satellites and the user changes from satellite to satellite, the antenna receives GPS signals at different times. Signal transit time is the signal transmission time duration from satellite to user and it is (pseudorange) / (speed of light). In order to designate the user position, it is convenient to choose the receiver time -t_r- as a reference to determine the relevant timings. Therefore, time of transmission is obtained by the subtraction of the signal transit time from the receiver time. Each satellite will have different transmission time due to the different signal transit time. Let us assume that time of transmission time found by selecting the receiver time reference is represented as t_c and defined as transmission time corrected by transit time. Time used for satellite positioning algorithms-represented as t- is designated by correcting t_c by using the satellite clock correction parameters. Figure 33 illustrates the timing relations.

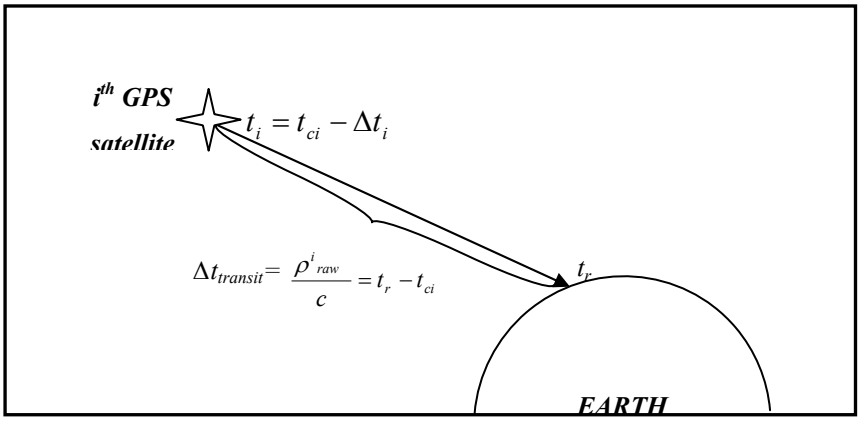

Figure 33. Timing representations of the GPS signal

If GPS receiver tracks n number of satellites, where $i=1,2,...,n$, transmission time is for the i^{th} satellite:

$$t_{ci} = t_r - \frac{\rho^i{}_{raw}}{c}$$

$$t_i = t_{ci} - \Delta t_i \qquad (4.1)$$

t_{ci} : Transmission time corrected by transit time of the i^{th} satellite,

t_i : Transmission time,

Δt_i : Satellite clock correction,

$\rho^i{}_{raw}$: Raw pseudorange provided by the receiver,

c : GPS speed of light, 299792458 m/second,

t_r : Receiver time that the GPS signal is received.

Although the GPS satellites have very sensitive atomic clocks, GPS control segment should correct the satellite clocks simultaneously with the clock-correction parameters. Satellite adds these clock-correction parameters into the ephemeris data. So, the receiver can calculate the clock-correction. Satellite

clock correction is described in [10]. Satellite clock correction Δt in equation (4.1) is calculated by using the parameters in the first subframe, which are presented in Table 8, according to the algorithm below:

Step 1. Let us define t_k by using the time of ephemeris and the transmission time corrected by transit time: $t_k = t_c - t_{oe}$. Here, time t_k shows whether start of the GPS week and end of the GPS week are overlapped or not. That's why following correction should be applied to t_k :

$$\text{If } t_k > 302400 \text{ then we have } t_k = t_k - 604800 \text{ or}$$
$$t_c = t_c - 604800$$

$$\text{If } t_k < -302400 \text{ then we have } t_k = t_k + 604800 \text{ or}$$
$$t_c = t_c + 604800$$

Step 2. Mean motion n is calculated by using the equation $n = \sqrt{\frac{\mu}{A^3}} + \Delta n$. In this equation μ=3.986005 x 10^{-10} second/$m^{1/2}$.

Step 3. Mean anomaly M: $M = M_0 + n(t_c - t_{oe})$.

Step 4. Kepler's equation for the eccentric anomaly E is $E = M + e_0 \sin E$. This equation is solved by iteration.

Step 5. Relativistic correction Δt_r varies as the sine of the satellite eccentric anomaly as follows: $\Delta t_r = Fe_0\sqrt{A}\sin E$. In this equation F is a constant and it is -4.442807633 x 10^{-10} second/m^2.

Step 6. Finally, satellite clock correction term Δt is approximated by a polynomial below:

$$\Delta t = a_{f0} + a_{f1}(t_c - t_{oc}) + a_{f2}(t_c - t_{oc})^2 + \Delta t_r - T_{gd} \quad (4.2)$$

As a result of this algorithm, transmission time is corrected by satellite clock correction. Therefore, obtained time correction in (4.2) is put in the equation (4.1). Transmission time *t* is derived and it is called GPS time. GPS time is a critical parameter to be used in satellite position equations. Because, successful GPS positioning is possible if the satellite positions are calculated at the time of the signal reception by the antenna.

4.2.2 Satellite Position Calculation

Satellite position computation uses the GPS time *t* and the ephemeris parameters as inputs. Satellite positions in Earth centered-Earth fixed (ECEF) coordinate system are computed by using the ephemeris model equations in [10].

Step 1. First, true anomaly *v* is found by using $\nu_1 = \arccos(\frac{\cos E - e_s}{1 - e_s^2 \cos E})$ and $\nu_2 = \arcsin(\frac{\sqrt{1-e_s^2}\sin E}{1 - e_s \cos E})$. So, $\nu = \nu_1 sign(\nu_2)$ is obtained. Note that E and e_s are obtained from GPS ephemeris data.

Step 2. Argument of latitude ϕ is the sum of the true anomaly (ν) and the argument of perigee: $\phi = \nu + \omega$.

Step 3. Argument of latitude ϕ, radius *r* and inclination *i* are corrected according to their second harmonic perturbations:

$$\delta\phi = C_{us}\sin 2\phi + C_{uc}\cos 2\phi \text{ and than } \phi \Rightarrow \phi + \delta\phi$$

$$\delta r = C_{rs}\sin 2\phi + C_{rc}\cos 2\phi \text{ and than } r \Rightarrow r + \delta r$$

$$\delta i = C_{is}\sin 2\phi + C_{ic}\cos 2\phi \text{ and than } i \Rightarrow i + \delta i + idot(t - t_{oe})$$

Step 4. Corrected longitude of ascending node is $\Omega_{er} = \Omega_0 + \dot{\Omega}(t - t_{oe}) + \Omega_e$. Ω_e is the Earth rotation rate and it is equal to 7.2921151467x10^{-5} radian/second.

Step 5. At the final point of the satellite position, x, y and z positions are calculated in ECEF coordinates:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} r\cos\Omega_{er}\cos\phi - r\sin\Omega_{er}\cos i\sin\phi \\ r\sin\Omega_{er}\cos\phi + r\cos\Omega_{er}\cos i\sin\phi \\ r\sin i\sin\phi \end{bmatrix} \tag{4.3}$$

Step 6. Even though the satellite position is calculated according to the equation (4.3), satellite coordinates should be adjusted by the signal transit time. ECEF coordinate system depends on time strictly, since the axes of the coordinate are fixed to the Earth and they rotate as the Earth is rotating. For this reason, satellite position should be adjusted according to the ECEF coordinates at the time of the signal reception. This adjustment is a feedback to the algorithm by updating the corrected longitude of ascending node Ω_{er}. Signal transit time $\Delta t_{transit}$ is calculated by the raw pseudorange and the speed of light as $\Delta t_{transit} = \frac{\rho_{raw}}{c}$. Next, Ω_{er} is improved by:

$$\Omega_{er} = \Omega_{er} - \Omega_e \Delta t_{transit}$$

where Ω_e is the Earth rotation rate and it is equal to 7.2921151467x10^{-5} radian/second.

4.2.3 Pseudorange

In GPS receivers, the distance of the antenna from the satellite is needed for positioning. Distance measurement should be achieved as precise as possible since the GPS signal propagates at speed of light. Thus, even minor errors cause big range errors. The distance in positioning is called as pseudorange. The word "pseudo" is used since this measurement includes errors such as satellite clock, atmospheric and receiver clock errors. On the other hand, there are error models to estimate the magnitudes of the errors. Raw pseudorange measurement is made more precise by the satellite clock and the atmospheric correction algorithms.

Following discussion is about the raw and the corrected pseudorange measurements. Raw pseudorange ρ_{raw} is defined as

$$\rho_{raw} = (t_r - t_c)c$$

where:

t_c : Signal transmission time corrected by signal transit time,

c : GPS speed of light, 299792458 m/second,

t_r : Receiver time.

Corrected pseudorange ρ is represented as

$$\rho = (t_r - t).c - \varepsilon_{ionosphere} - \varepsilon_{troposphere} \quad (4.4)$$

where:

t : Corrected signal transmission time (seconds);

$\varepsilon_{ionosphere}$: Ionospheric correction (meter);

$\varepsilon_{troposphere}$: Tropospheric correction (meter).

By substituting the equation (4.1) into the equation (4.4), we will have:

$$\rho = \left[t_r - (t_c - \Delta t)\right]c - \varepsilon_{ionosphere} - \varepsilon_{troposphere}$$

$$\rho = (t_r - t_c)c + c\Delta t - \varepsilon_{ionosphere} - \varepsilon_{troposphere} \quad (4.5)$$

In the equation (4.5), ρ_{raw} is observed. Therefore, raw pseudorange is converted into the corrected pseudorange, which is

$$\rho = \rho_{ham} + c\Delta t - \varepsilon_{troposfer} - \varepsilon_{iyonosfer} \quad (4.6)$$

Satellite clock correction $c\Delta t$ is calculated as explained in section 4.2.1. Atmospheric corrections are described in the next section.

4.2.3.1 Atmospheric Delay Correction

Atmosphere surrounding the Earth causes delays in the pseudorange measurement since atmosphere layers decrease the speed of the GPS signal during the transmission from satellite to the user. Precise positioning needs to eliminate this delay in the pseudorange. In GPS applications, atmosphere is modeled as consisting of ionosphere and troposphere.

4.2.3.1.1 Ionospheric Model

Ionosphere layer of the atmosphere is comprised of free ions inside. Collisions between the ions and the GPS signal cause delay in the pseudorange significantly. As stated in [10], ionospheric model is used to calculate the ionospheric delay in the pseudorange. The root mean square (rms) position error caused by the ionospheric delay can be reduced by at least 50 percent as indicated in [6].

Almanac data of the GPS signal includes the atmospheric correction parameters in almanac data to be used in ionospheric model called ionospheric parameters. Moreover, for the ionospheric correction elevation and the azimuth angles of the satellite with respect to the user should be known, too. Elevation and azimuth angles are illustrated in Figure 34. Zenith angle is the angle between the north vector and the vector from user to satellite called line of sight vector. Elevation angle is nothing but 90˚ minus zenith angle. When it comes to azimuth angle, it is computed from the east and north components of line of sight vector.

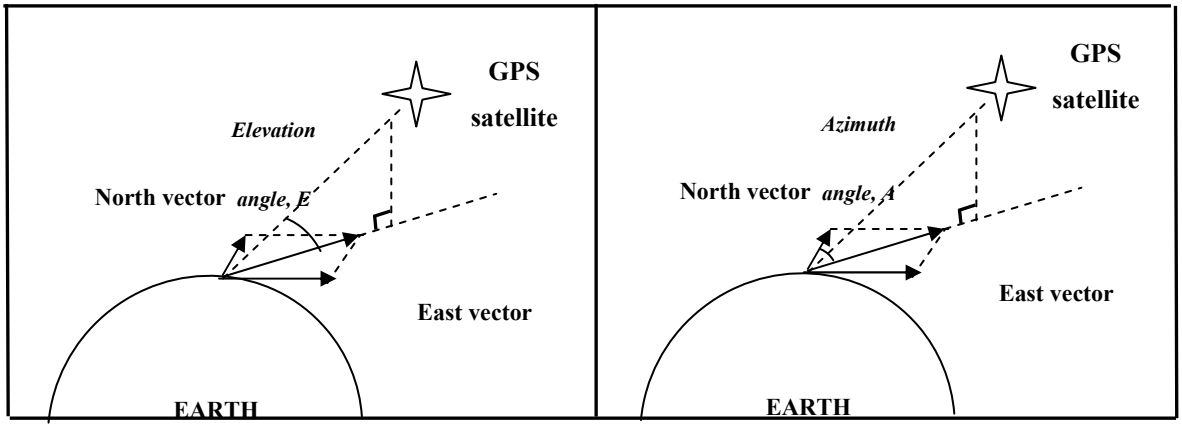

Figure 34. Elevation and Azimuth angles

In order to use the ionospheric model, receiver should generate the following terms:

A: Azimuth angle (semi-circle)

E: Elevation angle(semi-circle)

ϕ_u, λ_u: user geodetic latitude and longitude (semi-circle)

t: GPS time.

GPS satellite broadcasts the ionospheric correction parameters and they are represented as:

α_0, α_1, α_2, α_3 = The coefficients of cubic equation representing the amplitude of the vertical delay.

β_0, β_1, β_2, β_3 = The coefficients of cubic equation representing the period of the model.

Ionospheric algorithm is presented below by using the receiver generated terms and the ionospheric correction parameters:

Step 1. Compute the necessary terms as below:

x: Phase(radian)

F: Obliquity factor (unit less)

t_y: Local time (second)

ϕ_m: Geomagnetic latitude of the Earth projection of the ionospheric intersection point, mean ionospheric height assumed 350 km (semi-circles)

λ_i: Geodetic longitude of the Earth projection of the ionospheric intersection point, (semi-circles)

ϕ_i: Geodetic latitude of the Earth projection of the ionospheric intersection point, (semi-circles)

φ : Earth central angle between user positions Earth projection of the ionospheric intersection point, (semi-circles)

$$\varphi = 0.0137/(E+0.11)-0.022$$

$$\phi_i = \phi_u + \varphi \cos(A)$$

If $\phi_i > 0.416$ then $\phi_i = 0.416$.

If $\phi_i < -0.416$ then $\phi_i = -0.416$.

$$\lambda_i = \lambda_u + \varphi \sin(A)/\cos(\phi_i)$$

$$\phi_m = \phi_i + 0.064\cos(\lambda_i - 1.617)$$

$$t_y = (4.32e+4)\lambda_i + t;$$

If $t_y >= 86400$ then $t_y = t_y - 86400$

If $t_y < 0$ then $t_y = t_y + 86400$

Step 2. Following final terms should be calculated in order to reach ionospheric delay T_{iono} in seconds:

$F=1.0 + 16.0*(0.53-E)^3$

$$PER= \begin{cases} \sum_{n=0}^{3} \beta_n \phi m^n & , PER >= 72000 \\ 72000 & , PER < 72000 \end{cases}$$

$x=2\pi (t_y-50400)/PER;$

$$AMP=\begin{cases} \sum_{n=0}^{3} \alpha_n \phi_m^n & , AMP >= 0 \\ 0 & , AMP < 0 \end{cases}$$

If $|x| < 1.57$ than $T_{iono}=F(5e^{-9}+ AMP(1-x^2/2+x^4/24)$.

If $|x| \geq 1.57$ than $T_{iono}=F(5e^{-9})$.

Step 3. Consequently, ionospheric correction in meters in the equation (4.6) is obtained as the following equation:

$$\varepsilon_{iyonosfer} = T_{iono}\, x\, c \text{ (}c\text{: 299792458 meter/second)}$$

4.2.3.1.2 Tropospheric Model

Troposphere is the closest layer to the Earth and contains neutral particles. Troposphere delay is less important than the ionosphere delay in magnitude. There are many models for tropospheric delay computation. In this thesis, we use a simplified model in [6]. Simplified model uses elevation angle of the satellite only, i.e.

$$\varepsilon_{troposfer} = \frac{2.47}{\sin(E) + 0.0121} \text{ (meters)}$$

4.3 GPS Equations

GPS positioning is achieved by using the corrected pseudoranges and the satellite positions. ECEF position of the user is desired to be solved. However, an additional unknown should be solved since GPS receivers do not generally have sensitive atomic clocks. Therefore, receiver cannot provide accurate timing like

satellites can do. Receivers have timing bias, which is observed in the pseudorange measurement. This bias value is the other unknown with the ECEF position of the receiver x, y, z. Receiver bias value is unique at a time that does not change with different satellites.

As a result, we need to have four equations which can be provided by the tracking at least four satellites. Inputs of the position and the velocity computation algorithms are listed below:

If n is the number of satellites that the user has tracked and $n \geq 4$, $i = 1, 2, \ldots, n$:

a. Satellite position in ECEF coordinate system: i^{th} satellite position is the set (x_s, y_s, z_s), which is computed by the ephemeris model .

b. Corrected pseudorange between the satellite and the user: GPS signal transmission time for the i^{th} satellite is t_i and the pseudorange of that satellite is ρ_i.

c. Doppler frequency of the satellite: Doppler frequency is necessary for velocity calculation. f_i is the i^{th} satellite Doppler frequency.

There exist eight unknowns, which are user position (x_u, y_u, z_u), velocity position (v_x, v_y, v_z), receiver clock bias b_u and user clock drift f_d.

4.3.1 Pseudorange

Pseudorange is nothing but the analytical difference between the user and the satellite positions in 3 dimensional space. Therefore pseudorange equation is

$$\rho_i = \sqrt{(x_{si} - x_u)^2 + (y_{si} - y_u)^2 + (z_{si} - z_u)^2} + b_u \tag{4.7}$$

where

(x_u, y_u, z_u) : GPS receiver position in ECEF coordinate system,

(x_s, y_s, z_s) : GPS satellite position in ECEF,

b_u : Receiver clock bias in meters,

ρ_i : Pseudorange between the satellite and the receiver,

c : speed light (299792458 m/sec),

i : satellite number and $i=1,2,...,n$ if the receiver tracks n number of satellites signal.

4.3.2 Doppler frequency

Doppler frequency shift of a satellite is observed from the carrier tracking loop as mentioned in Chapter 3. Doppler frequency includes nothing but the relative velocities of the user and the satellite. That is why Doppler frequency is used as a measurement for the user velocity. However, satellite velocity is also needed in the computations. Thus, the first thing to do is to determine the space vehicle velocity through x, y and z axes. Satellite average velocity is directly calculated from the position change of the satellite divided by the passed time. This is possible by differentiating the calculated space vehicle position according to the ephemeris model.

The equation relating the measurements of the Doppler shift to the user velocity is given in [9] as

$$f_{doppler} = \frac{1}{\lambda}(v.u_i - v_i.u_i) + f_b \tag{4.8}$$

where

$f_{doppler}$: Doppler shift (Hz)

f_d : Receiver error drift (Hz)

λ : Wave-length of the GPS signal, which is c (speed of light) / f (L_1 or L_2)

v : Receiver velocity v_x, v_y, v_z

v_i : i^{th} satellite velocity v_{ix}, v_{iy}, v_{iz}

u_i : Normalized line of sight vector from the satellite to the receiver which can be found according to the equation (4.7): $u_i = \left[\frac{(x_{si} - x_u)}{\rho_i - b_u} \quad \frac{(y_{si} - y_u)}{\rho_i - b_u} \quad \frac{(z_{si} - z_u)}{\rho_i - b_u} \right]^T$.

Hence, equation (4.8) becomes

$$\lambda f_{doppler} = (v - v_i).u_i + \lambda f_b$$

$$\lambda f_{doppler} = \begin{bmatrix} v_x - v_{ix} & v_y - v_{iy} & v_z - v_{iz} \end{bmatrix} \begin{bmatrix} \frac{(x_{si} - x_u)}{\rho_i - b_u} \\ \frac{(y_{si} - y_u)}{\rho_i - b_u} \\ \frac{(z_{si} - z_u)}{\rho_i - b_u} \end{bmatrix}_i + \lambda f_b$$

$$\lambda f_{doppler} = \frac{(x_{si} - x_u)}{\rho_i - b_u}(v_x - v_{ix}) + \frac{(y_{si} - y_u)}{\rho_i - b_u}(v_y - v_{iy}) + \frac{(z_{si} - z_u)}{\rho_i - b_u}(v_z - v_{iz}) + f_d \quad (4.9)$$

In the equation (4.9), $\lambda f_{doppler}$ is nothing but the deltarange measurement of the satellite, which is the rate of change of the pseudorange represented by $\dot{\rho}_i$ referred to the equation (4.7).

4.4 GPS Position and Velocity Estimation Methods

Mainly two types of position estimation method will be considered. First one is the (LSE) least squares estimation and the second one is Kalman filter approach.

4.4.1 Least Squares Estimation

By recalling the equation (4.7), there exist 4 unknowns in pseudorange which are *(x_u, y_u, z_u, b_u)*. Thus, receiver needs at least 4 satellite signal processing in order to have 4 equations. If the derivative of both sides of the equation (4.7) is taken:

$$\delta\rho_i = \frac{(x_{si} - x_u)\delta x_u + (y_{si} - y_u)\delta y_u + (z_{si} - z_u)\delta z_u}{\sqrt{(x_{si} - x_u)^2 + (y_{si} - y_u)^2 + (z_{si} - z_u)^2}} + \delta b_u$$

$$\delta\rho_i = \frac{(x_{si} - x_u)\delta x_u + (y_{si} - y_u)\delta y_u + (z_{si} - z_u)\delta z_u}{\rho_i - b_u} + \delta b_u$$

$$\delta\rho_i = \alpha_{i1}\delta x_u + \alpha_{i2}\delta y_u + \alpha_{i3}\delta z_u + \delta b_u \quad (4.10)$$

where: $\alpha_{i1} = \frac{(x_{si} - x_u)}{\rho_i - b_u}$, $\alpha_{i2} = \frac{(y_{si} - y_u)}{\rho_i - b_u}$, $\alpha_{i3} = \frac{(z_{si} - z_u)}{\rho_i - b_u}$

In order to obtain velocity information, we should take the derivative of the deltarange, which is shown in the equation (4.9) as

$$\dot{\delta\rho}_i = \frac{(x_{si} - x_u)}{\rho_i - b_u}\dot{\delta x}_u + \frac{(y_{si} - y_u)}{\rho_i - b_u}\dot{\delta y}_u + \frac{(z_{si} - z_u)}{\rho_i - b_u}\dot{\delta z}_u + \dot{\delta b}_u$$

$$\dot{\delta\rho}_i = \alpha_{i5}\dot{\delta x}_u + \alpha_{i6}\dot{\delta y}_u + \alpha_{i7}\dot{\delta z}_u + \dot{\delta b}_u \quad (4.11)$$

where: $\alpha_{i5} = \frac{(x_{si} - x_u)}{\rho_i - b_u}$, $\alpha_{i6} = \frac{(y_{si} - y_u)}{\rho_i - b_u}$, $\alpha_{i7} = \frac{(z_{si} - z_u)}{\rho_i - b_u}$.

By combining the equations (4.10) and (4.11), we will have the below expression:

$$\underbrace{\begin{bmatrix} \delta\rho_1 \\ \delta\rho_2 \\ \cdot \\ \cdot \\ \cdot \\ \delta\rho_n \\ \dot{\delta\rho}_1 \\ \dot{\delta\rho}_2 \\ \cdot \\ \cdot \\ \cdot \\ \dot{\delta\rho}_n \end{bmatrix}}_{\bar{\rho}} = \underbrace{\begin{bmatrix} \alpha_{11} & \alpha_{12} & \alpha_{13} & 1 & 0 & 0 & 0 & 0 \\ \alpha_{21} & \alpha_{22} & \alpha_{23} & 1 & 0 & 0 & 0 & 0 \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ \alpha_{n1} & \alpha_{n2} & \alpha_{n3} & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \alpha_{15} & \alpha_{16} & \alpha_{17} & 1 \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ 0 & 0 & 0 & 0 & \alpha_{n5} & \alpha_{n6} & \alpha_{n7} & 1 \end{bmatrix}}_{\alpha} \underbrace{\begin{bmatrix} \delta x_u \\ \delta y_u \\ \delta z_u \\ \delta b_u \\ \dot{\delta x}_u \\ \dot{\delta y}_u \\ \dot{\delta z}_u \\ \dot{\delta b}_u \end{bmatrix}}_{\delta}$$

(4.11)

$$\bar{\rho} = \alpha\delta$$

$$\delta = \alpha^{-1}\bar{\rho}$$

Since α in the equation (4.11) is not a square matrix, pseudo inverse is applied:

$$\delta = (\alpha^T\alpha)^{-1}\alpha^T\bar{\rho} \qquad (4.12)$$

In the light of this information, LSE method steps are given as follows:

Step 1. Choose initial values of the position, velocity and the user clock bias and drift. These initial values can be taken as

$$x_i = \begin{bmatrix} x_i & y_i & z_i & b_i & v_{xi} & v_{yi} & v_{zi} & d_i \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Step 2. Calculate $\bar{\rho}$ as

$$\delta\rho_i = \rho^i_{corrected} - \rho_i$$

$$\dot{\delta\rho}_i = \lambda f_i - \dot{\rho}_i$$

$$\bar{\rho} = \begin{bmatrix} \delta\rho_1 & \delta\rho_2 & . & . & . & \delta\rho_n & \delta\dot{\rho}_1 & \delta\dot{\rho}_2 & . & . & . & . & \delta\dot{\rho}_n \end{bmatrix}^T \quad (4.13)$$

where:

ρ_i : Estimated pseudorange by using equation (4.7),

$\dot{\rho}_i$: Estimated deltarange by using equation (4.9),

$\rho^i_{corrected}$: Calculated and corrected pseudorange from the navigation data at time i,

λf_i : Doppler shift frequency times the wavelength at time i,

Step 3. Calculate the α matrix according to the equation (4.11).

Step 4. Obtain δ vector as shown in the equation (4.12)

Step 5. Calculate the value of $A = \sqrt{\delta x_i^2 + \delta y_i^2 + \delta z_i^2}$. A is threshold for LSE steps.

Step 6. Update the unknown parameters by $\delta = \begin{bmatrix} \delta x_i & \delta y_i & \delta z_i & \delta b_i & \delta v_{ix} & \delta v_{iy} & \delta v_{iz} & \delta d_i \end{bmatrix}^T$ as:

$$\begin{aligned} x_{i+1} &= x_i + \delta x_i & v_{i+1\ x} &= v_{ix} + \delta v_{ix} \\ y_{i+1} &= y_i + \delta y_i & v_{i+1\ y} &= v_{iy} + \delta v_{iy} \\ z_{i+1} &= z_i + \delta z_i & v_{i+1\ z} &= v_{iz} + \delta v_{iz} \\ b_{i+1} &= b_i + \delta b_i & d_{i+1} &= d_i + \delta d_i \end{aligned}$$

Step 7. Continue from Step 2 to Step 6 until the value of A is smaller than the pre-determined threshold. (During the iteration A decreases)

4.4.2 Kalman Filter

In linear system estimation literature, Kalman state estimator or filter is used for finding the best estimate of the states by using the measurements of the system, as emphasized in [1]. Beside the state estimation, error covariance matrix is also propagated according to the modeled system noise and uncertainty about

the estimated state. This error covariance update is the main difference of the Kalman filter and the least squares estimation. Kalman filter tries to decrease the state estimation error by using the previous filter step information.

GPS equations are non-linear such as the pseudorange equation (4.7). Nevertheless, Kalman filtering is possible by discrete extended Kalman filter structure, which is actually a non-linear Kalman filter as stated in [9].

4.4.2.1 Discrete Extended Kalman Filter (EKF)

States of the implemented EKF in GPS positioning are the user position, receiver clock bias, user velocity and receiver clock drift.

As in [1], any linear stochastic system is represented by

$$x_{k+1} = Ax_k + Bu_k + Gw_k$$
$$y_k = Cx_k + Hv_k$$

x_k is the state, u_k is the input and w_k is the process noise, y_k is the output and v_k is the measurement noise at time *k*. *A*, *B*, *G*, *C*, *H* are matrices of appropriate dimension.

System representation of the GPS positioning should be defined in order to construct the Kalman filter easily. Note that u_k and *B* are zero for the GPS system representation. Therefore, we have

$$x_{k+1} = Ax_k + Gw_k$$
$$y_k = C_k x_k + Hv_k \quad (4.14)$$

where x_k is the state vector which includes the user position x_{uk}, y_{uk}, z_{uk}, the receiver bias b_{uk}, the user velocity $v_{xuk}, v_{yuk}, v_{zuk}$, the receiver drift d_{uk} at time *k*

$$\Rightarrow x_k = \begin{bmatrix} x_{uk} & y_{uk} & z_{uk} & b_{uk} & v_{xuk} & v_{yuk} & v_{zuk} & d_{uk} \end{bmatrix}^T,$$

y_k is measurement matrix and it is exactly equal to $\bar{\rho}$ defined in equation (4.13)

C_k is exactly equal to the α matrix in (4.11) at time *k*, i.e., $C_k = \alpha$

H is equal to the identity matrix I i.e. $H = I$

v_k is the measurement noise with covariance matrix as R_k at time *k*

w_k is the process noise as $w_k = \begin{bmatrix} a_x & a_y & a_z & a_b \end{bmatrix}^T$ where a_x, a_y and a_z are the acceleration information of the receiver. Covariance matrix of w_k is Q_1.

A is as

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & \Delta t & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.15}$$

G is as

$$G = \begin{bmatrix} \frac{1}{2}\Delta t^2 & 0 & 0 & 0 \\ 0 & \frac{1}{2}\Delta t^2 & 0 & 0 \\ 0 & 0 & \frac{1}{2}\Delta t^2 & 0 \\ 0 & 0 & 0 & \frac{1}{2}\Delta t^2 \\ \Delta t & 0 & 0 & 0 \\ 0 & \Delta t & 0 & 0 \\ 0 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & \Delta t \end{bmatrix}.$$

Covariance matrix of measurement noise and covariance matrix of Gw_k should be found in order to complete the system representation in (4.13):

- Let us assume that covariance matrix of Gw_k is represented as Q and it is $Q = E\{(Gw_k)(Gw_k)^T\} = E\{Gw_k w_k^T G^T\} = GE\{w_k w_k^T\}G^T$. By substituting the process noise covariance matrix Q_1, we have

$$Q = GQ_1G^T \tag{4.16}$$

- Covariance matrix of measurement noise is R as appointed before.

After defining all the parameters in system representation, discrete extended Kalman filter can be constructed. By [9] and [2], steps of the filter are as follows:

Step 1. Define the state of the EKF as the receiver position, receiver clock bias, receiver velocity and clock drift as below:

$$x_k = \begin{bmatrix} x_{uk} & y_{uk} & z_{uk} & b_{uk} & v_{xuk} & v_{yuk} & v_{zuk} & d_{uk} \end{bmatrix}$$

Step 2. Initialize the state as $x_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$

Step 3. As in [6], set the process noise covariance matrix Q_1 as:

$$Q_1 = \begin{bmatrix} 0.333 & 0 & 0 & 0 \\ 0 & 0.333 & 0 & 0 \\ 0 & 0 & 0.333 & 0 \\ 0 & 0 & 0 & 0.083 \end{bmatrix}$$

Step 4. Compute the covariance matrix Q by using equation (4.16). Note that Δt is the time interval between two successive GPS data.

Step 5. As in [6], initialize the error covariance matrix is as below:

$$P_0 = \begin{bmatrix} (5000)^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & (5000)^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & (5000)^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & (100)^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & (1000)^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & (1000)^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & (1000)^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & (299792458 \text{ x } 1e-6)^2 \end{bmatrix}$$

Step 6. As in [6], set the measurement noise covariance matrix R_k as:

$$R_k = \begin{bmatrix} 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & . & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & . & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.0625 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & . & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & . & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0625 \end{bmatrix}$$

Note that the size of the measurement noise covariance matrix is $2n$ x 8, where n is the number of satellites.

Step 7. Evaluate matrix C_k at time k:

$$C_k = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \alpha_{13} & 1 & 0 & 0 & 0 & 0 \\ \alpha_{21} & \alpha_{22} & \alpha_{23} & 1 & 0 & 0 & 0 & 0 \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ \alpha_{n1} & \alpha_{n2} & \alpha_{n3} & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \alpha_{15} & \alpha_{16} & \alpha_{17} & 1 \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ 0 & 0 & 0 & 0 & \alpha_{n5} & \alpha_{n6} & \alpha_{n7} & 1 \end{bmatrix}$$

where

$$\alpha_{i1} = \alpha_{i5} = \frac{(x_{si} - x_{uk})}{\rho_i - b_u}, \alpha_{i2} = \alpha_{i6} = \frac{(y_{si} - y_{uk})}{\rho_i - b_u}, \alpha_{i3} = \alpha_{i7} = \frac{(z_{si} - z_{uk})}{\rho_i - b_u}.$$

Step 8. Compute the Kalman gain matrix by using the a priori error covariance matrix P_k, measurement connection matrix C_k and the measurement noise R_k:

$$K_k = P_k C_k^T [C_k P_k C_k + R_k]^{-1}$$

Step 9. Next, state update must be applied as follows:

$$x_k = x_k + K_k [y_k - C_k x_k]$$

where y_k is the measurement vector which is calculated according to equation (4.13).

Step 10. Update the covariance matrix as below:

$$P_k = [I - K_k C_k] P_k [I - K_k C_k]^T + K_k R_k K_k^T$$

Step 11. Covariance matrix is propagated for the next time *k+1*:

$$P_{k+1} = A P_k A^T + Q$$

where *A* is as in equation (4.15).

Step 12. State estimate is also propagated for the next time *k+1*: $x_{k+1} = A x_k$

Step 13. Increase *k* by 1 and return to Step 7 until $k = k_{final}$ is reached.

CHAPTER 5

SIMULATION RESULTS

Software receiver implementation has been explained in the previous chapters. This chapter includes the results of the software GPS receiver. By referring to Figure 2, simulation results will be presented in two main sections as signal processing module and navigation algorithm results.

5.1 Software Signal Processing Module Results

Signal processing module block diagram of the implemented software GPS receiver is shown in Figure 35.

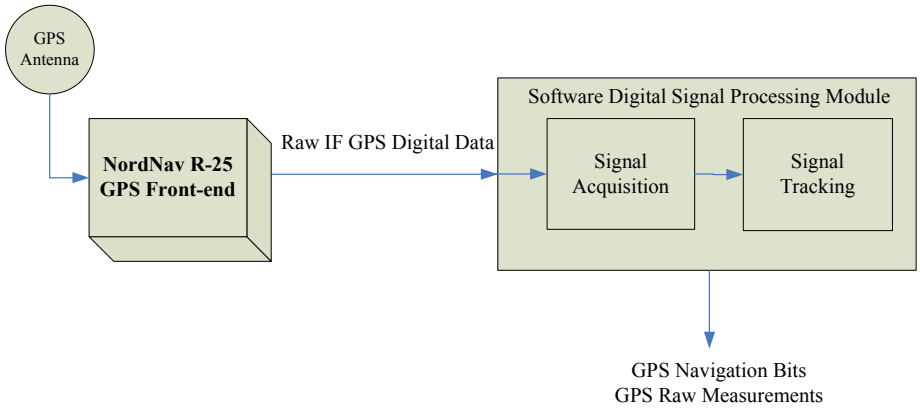

Figure 35. DSP module set-up of the implemented GPS receiver

In the implemented signal processing module results, the specifications of the input raw GPS IF digital data will be presented, first. Next, software C/A code generator will be explained. Furthermore, conventional signal processing

method, BAAS signal processing method results and the comparison of two methods will be presented.

5.1.1 Raw GPS IF Signal

Software signal processing module results consist of signal acquisition and signal tracking results. It is driven by raw GPS IF signal samples collected from the NordNav^{TM} GPS front-end and it is provided by Dr.Dennis Akos, Colorado University. Specifications of the raw GPS IF samples are presented below:

- Center frequency is decreased to 3.99 MHz by GPS front-end. Therefore, intermediate frequency is 3.99 MHz.
- Sampling frequency is 16.035 MHz.
- Satellites elevation and azimuth angles at the time of data collection are found from the almanac data by Dr. Dennis Akos. Candidate satellites information is presented in Table 11.

Table 11. Candidate satellites information

Satellite PRN #	Elevation Angle	Azimuth Angle
05	4.6°	31.7°
07	18.1°	326.3°
09	10.8°	4.9°
11	64.3°	216.2°
14	50.4°	80.5°
19	18.0°	195.2°
20	27.1°	245.1°
22	7.7°	76.4°
25	9.9°	123.0°
28	8.1°	278.6°

- In order to illustrate the raw GPS IF samples, Figure 36 is presented to show the first 1000 samples of the GPS signal in time domain. Furthermore, Figure 37 shows the power spectral density of the IF signal.

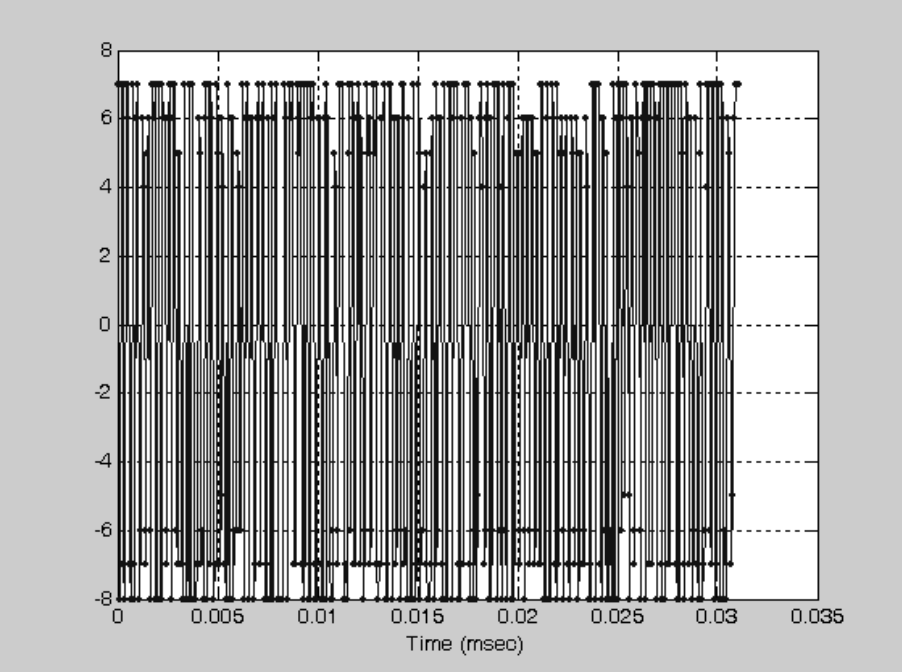

Figure 36. A portion of the time domain GPS signal (first 1000 samples)

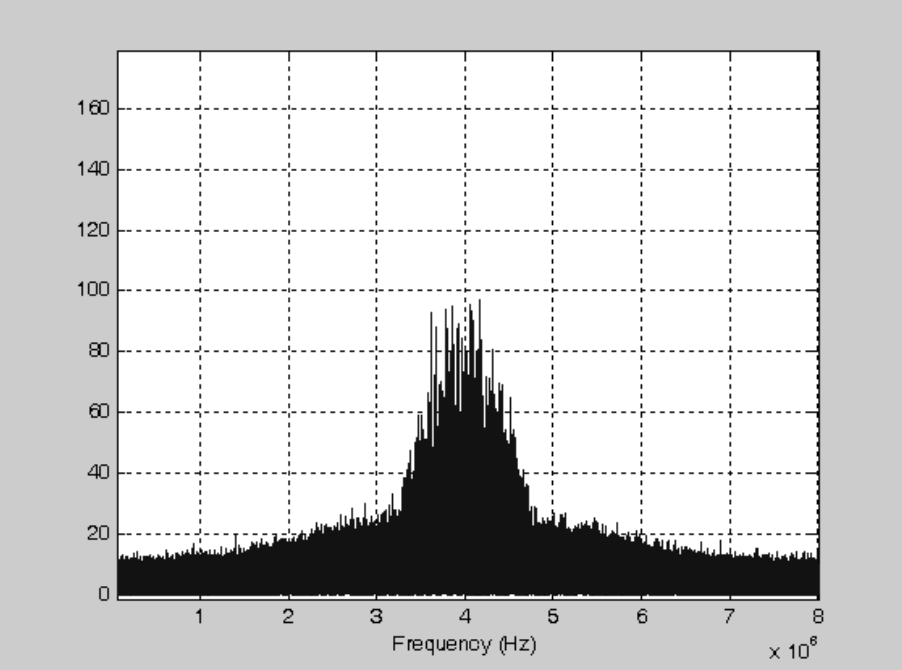

Figure 37. Power spectral density of the IF signal

5.1.2 Software C/A code generator

C/A code generator is the critical element in GPS receivers. It should generate C/A code bit stream at any sampling frequency, at any desired shift and at any length. Hence, implemented code generator is the function in Matlab as:

$$[\ prnf\ ] = C\_A\_code(prnid,\ length,\ shift\_chip, f_s)$$

where

prnf : C/A code output

prnid : Satellite PRN id (1-32)

length : Code sequence length

shift_chip : Shift value in terms of chip (note that 1 chip = 1 milliseconds / 1023 = 977.5 nanoseconds)

f_s : Sampling frequency

As an example of a C/A code signal, Figure 38 presents 0.1 millisecond portion of the 31st satellite code with sampling frequency 1.023 MHz that is the actual GPS clock rate.

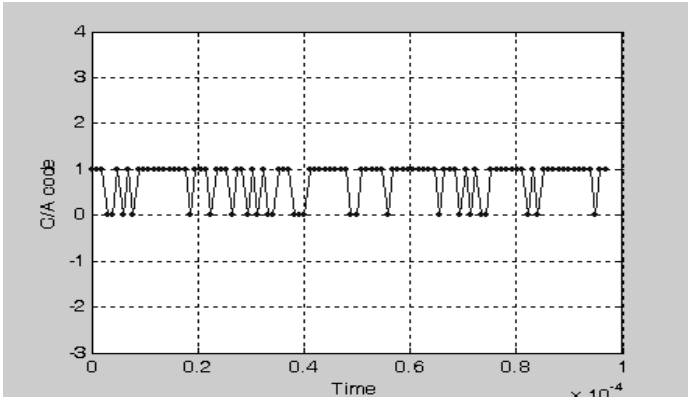

Figure 38. 0.1 milliseconds duration of the 31st satellite C/A code

Auto-correlation and the cross-correlation functions of the C/A code are the base in CDMA technique. Hence, auto-correlation function of the 19th

satellite code and the cross-correlation function of the 19th and the 31st satellite are presented in Figure 39 and Figure 40, respectively. Note that the maximum auto-correlation is obtained as soon as the codes are synchronized. The cross-correlation of the different satellites can never be as large as the auto-correlation function maximum value. Therefore, GPS receiver can extract a satellite signal from the carrier signal. All GPS satellite signals are transmitted through the same carrier frequency.

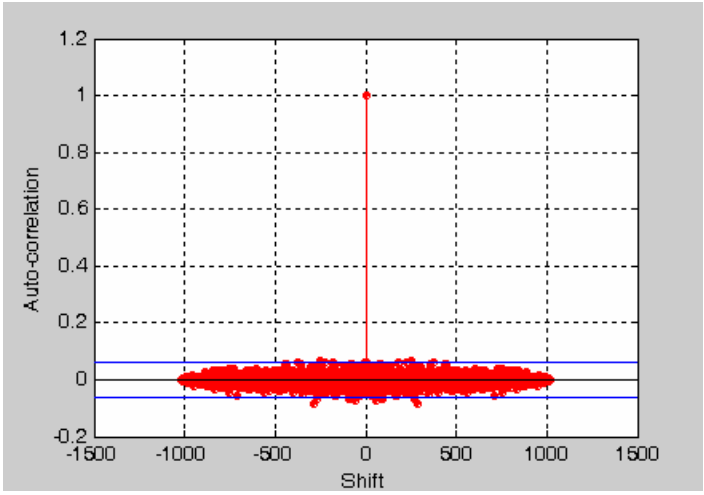

Figure 39. Auto-correlation of the 19th satellite C/A code

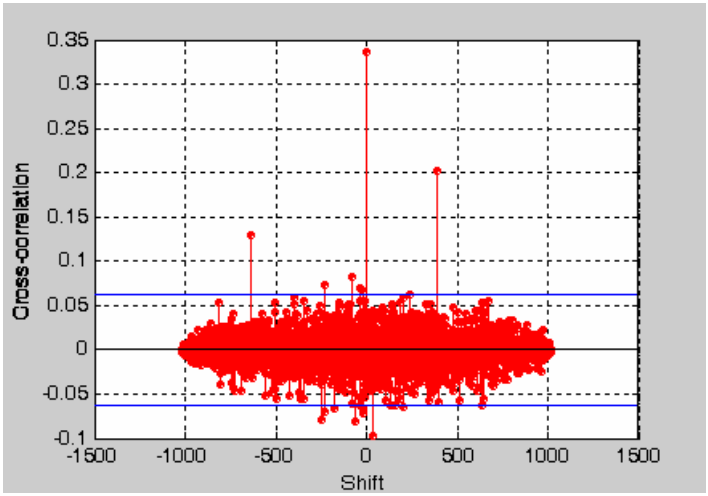

Figure 40. Cross-correlation of the 19th and 31st satellite C/A codes

5.1.3 Conventional Signal Processing Results

According to Table 11, most probable satellites in the signal are 11^{th} and 20^{th} PRN numbered satellites since they have high elevation angles. On the other hand, satellite with PRN id 15 is not expected to be found since it is not in the sky at the time of data collection. Therefore, PRN 11, 20 and 15 will be processed in order to present the conventional signal processing performance.

In the conventional signal acquisition, C/A code is searched over all possible values for code shift and carrier frequency signal. This is an exhaustive signal acquisition process. Carrier frequency and C/A code delay are the main outputs of signal acquisition. Furthermore, acquisition process duration is presented by using *tic* and *toc* commands of Matlab in order to comment on the methods speed. Used PC is Pentium 4(R) CPU 2.00 GHz and has 512 MB of RAM.

Conventional acquisition parameters are selected as below:

- Code acquisition resolution is 0.0638 chip
- Carrier acquisition resolution is 25 Hz.

When it comes to conventional tracking loops, C/A code is tracked by DLL and carrier signal is tracked by FLL. During the tracking process, threshold is used to control whether the signal is missed or not. If the satellite signal is missed, acquisition must be re-processed in order to acquire the signal again. Conventional tracking parameters are selected as below:

- Code tracking loop is designed by using DLL with normalized early minus late envelope discriminator as defined in Table 2. Pre-detection integration time for DLL is 10 milliseconds. Code delay is updated at every 10-millisecond intervals.
- Code tracking resolution is 0.1276 chip.

- Carrier tracking loop is designed by using FLL with two-quadrant arctangent discriminator as defined in Table 3.

- Carrier frequency is tracked by the FLL discriminator output. Pre-detection integration time for FLL is 10 milliseconds. Therefore, carrier signal is updated at every 10 milliseconds intervals.

- Signal presence is controlled by comparing the signal correlation with a pre-determined threshold. Threshold is selected to be 600000. Correlation below this threshold shows that the signal is missed and it is needed to be acquired.

Tracking process traces the C/A code, carrier signal, and GPS navigation bits are observed from carrier tracking loop as mentioned before. Therefore, navigation bits of the satellite, C/A code delay and Doppler frequency change in the input GPS signal will be presented in the tracking results.

Conventional signal processing results for PRN 11

PRN 11 has elevation angle of 64.3º. Thus, satellite is expected to be found.

Conventional acquisition of the satellite PRN 11

In Figure 41, obvious peak is observed in the acquisition result. So, PRN 11 is acquired. Acquisition metric value is normalized to show the strength of the maximum peak. Figure 43 is code delay search at correct carrier signal frequency. Figure 42 is carrier frequency search at correct code delay.

Table 12 includes the acquisition outputs. According to this table C/A code of PRN 11 is shifted 0.60704 milliseconds because of the transmission from satellite to the user. If we multiply this delay with the speed of light, we can obtain the fine component of the pseudorange as shown in the table. Doppler frequency of PRN 11 is the frequency deviation from the IF frequency. Furthermore, process duration shows the speed of the conventional acquisition.

As presented in the table, conventional acquisition speed is too low and it may cause operation time lag especially in real-time processing.

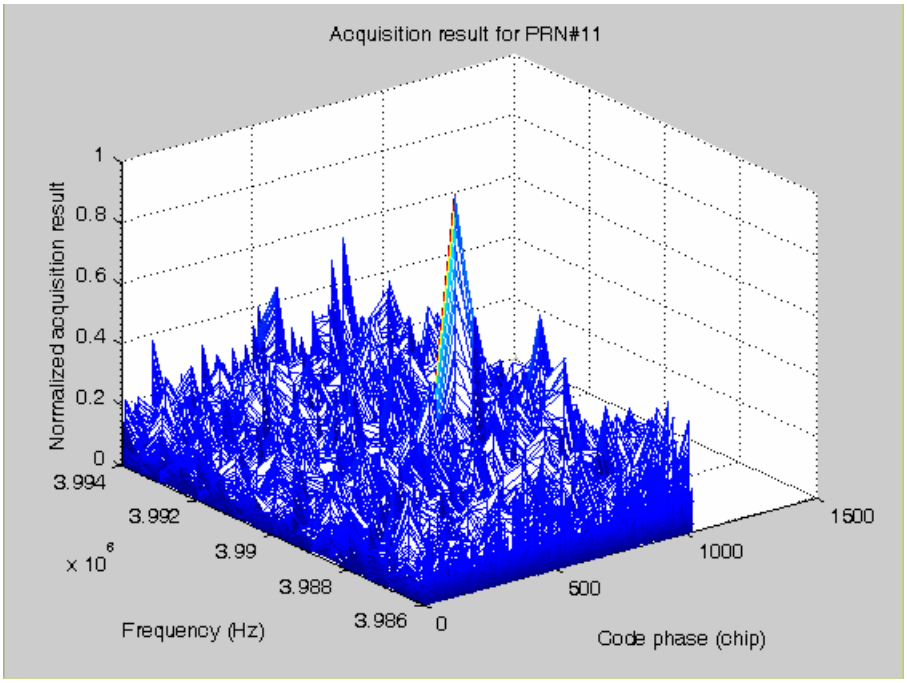

Figure 41. Acquisition metric for the satellite with PRN 11

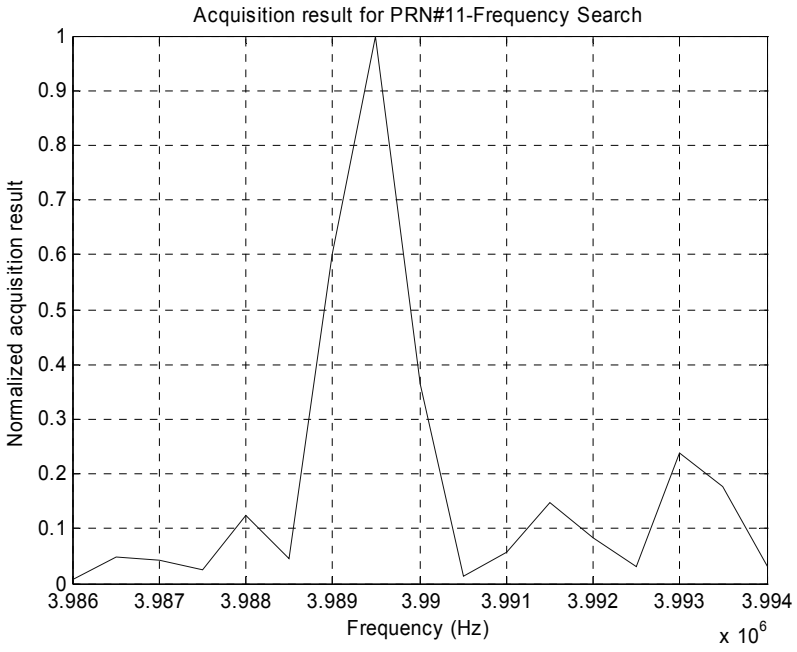

Figure 42. Frequency search of PRN 11

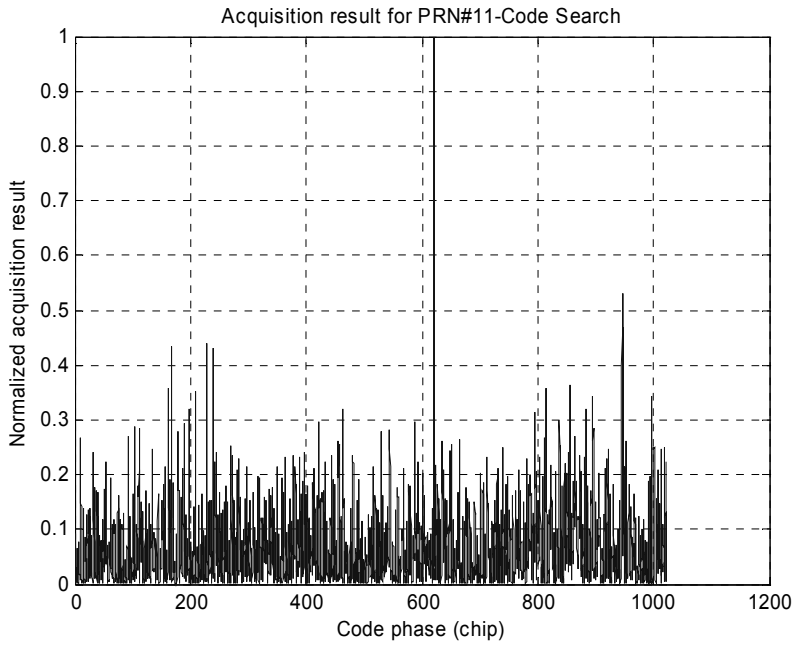

Figure 43. Code search of PRN 11

Table 12. Acquisition results of the conventional algorithm for PRN id 11

Code shift value	621 chip = 620.88 / 1.023 MHz = 0.60704 millisecond
Sample that the code begins	9734
Fine pseudorange	181986.0137 meters
IF frequency	3989525 Hz.
Doppler frequency	475 Hz.
Conventional acquisition process duration	657.86 seconds

Conventional tracking of the satellite PRN number 11

250 milliseconds portion of navigation bits of the PRN 11 is presented in Figure 44. Figure 45 is Doppler frequency change during carrier tracking. On the other hand, Figure 46 is C/A code delay change during code tracking.

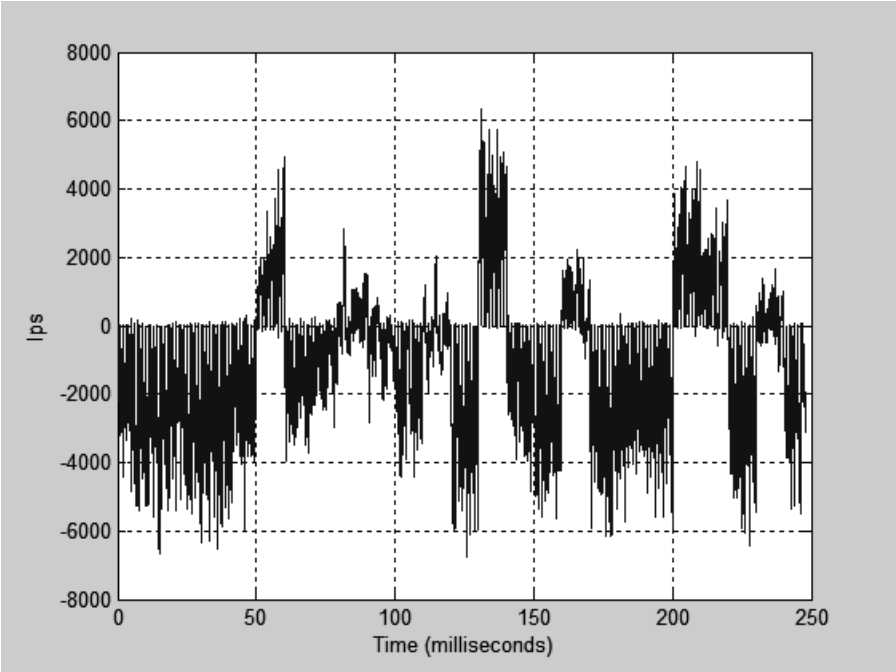

Figure 44. Navigation bits, output of the carrier tracking loop for PRN 11

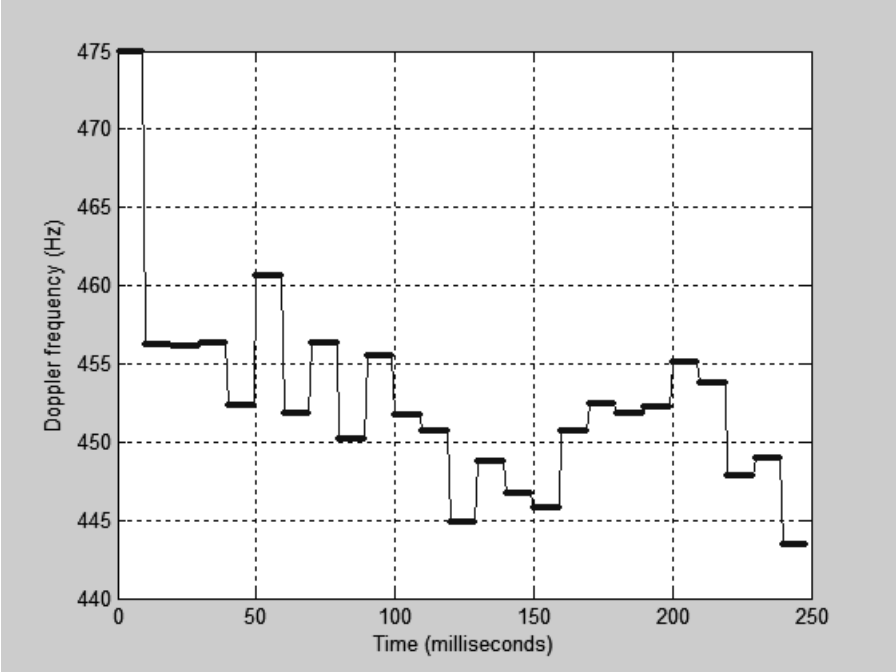

Figure 45. Doppler frequency obtained from carrier tracking for PRN 11

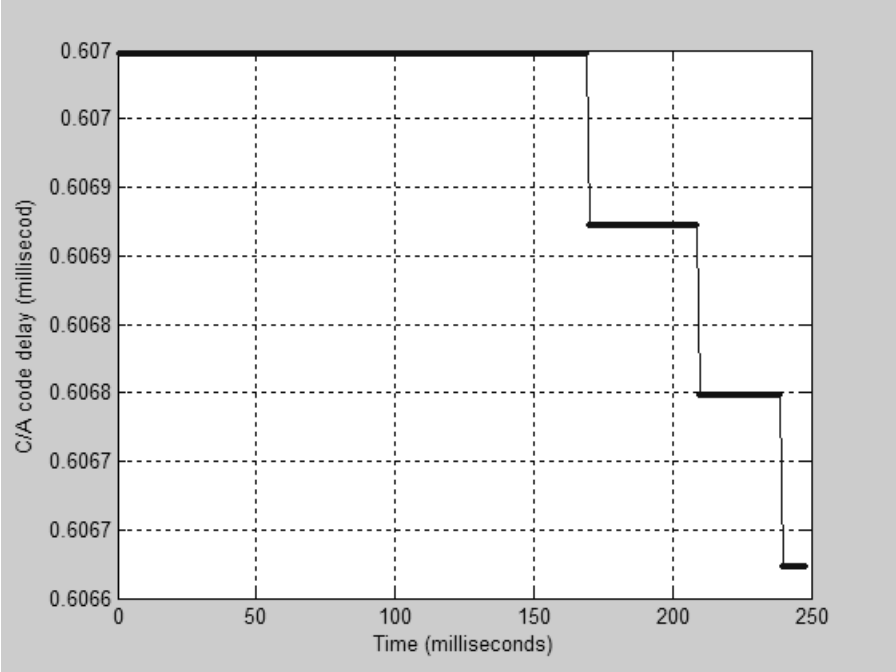

Figure 46. C/A code delay obtained from code tracking for PRN 11

Conventional signal processing results for PRN 20

PRN 20 has elevation angle of 27.1º. Thus, satellite is expected to be found. Furthermore, it can be weaker than PRN 11 since PRN 20 has elevation angle lower than the PRN 11 elevation angle.

Conventional acquisition of the satellite PRN 20

In Figure 47, obvious peak is observed in the acquisition result. So, PRN 20 is acquired. Acquisition metric value is normalized to show the strength of the maximum peak. Figure 49 is code delay search at correct carrier signal frequency. Figure 48 is carrier frequency search at correct code delay. Table 13 includes the acquisition outputs. According to this table C/A code of PRN 20 is shifted 0.733 milliseconds because of the transmission from satellite to the user. If we multiply this delay with the speed of light, we can obtain the fine component of the pseudorange as shown in the table. Doppler frequency of PRN 20 is the frequency deviation from the IF frequency. Furthermore, process duration shows the speed of the conventional acquisition. As presented in the

table, conventional acquisition speed is too low and it may cause operation time lag especially in real-time processing.

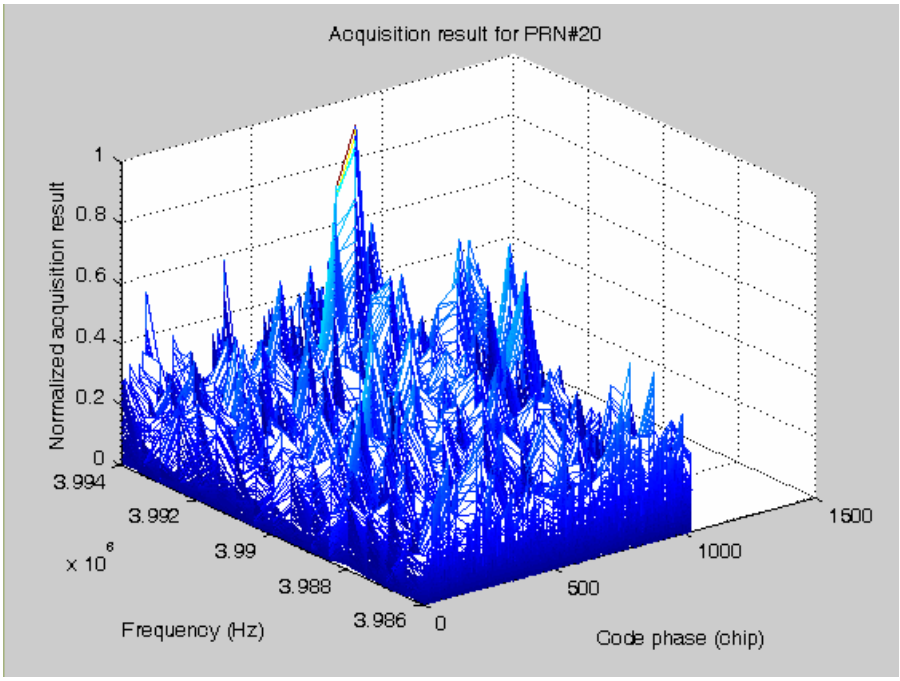

Figure 47. Acquisition metric for the satellite with PRN 20

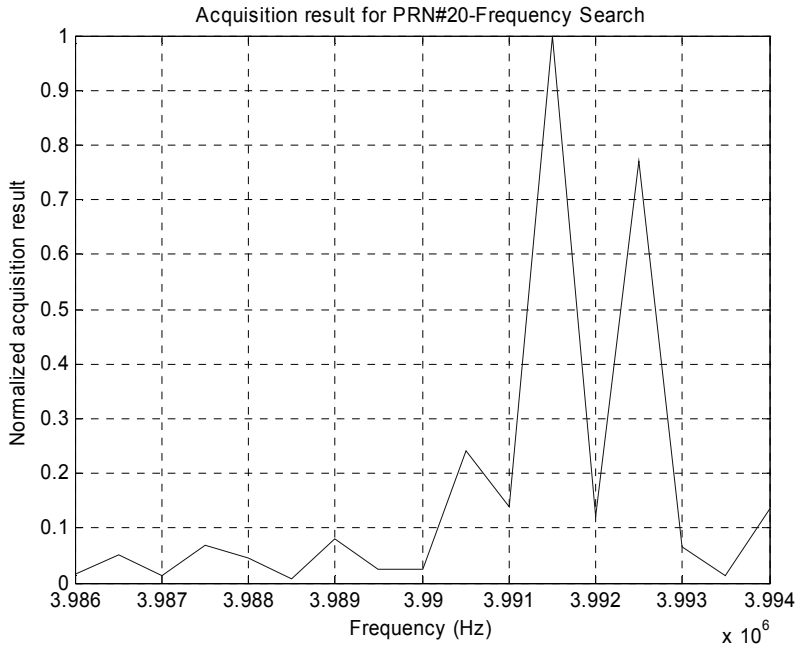

Figure 48. Frequency search of PRN 20

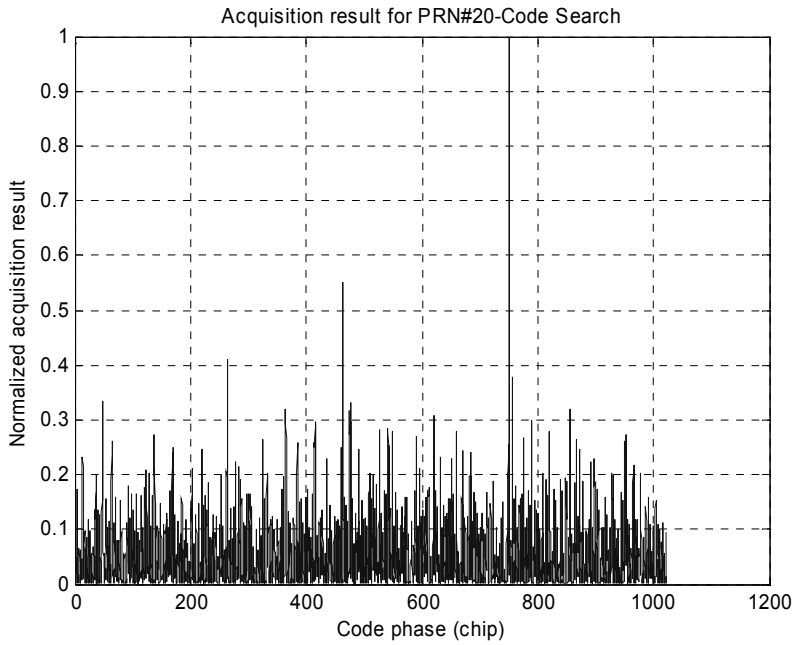

Figure 49. Code search of PRN 20

Table 13. Acquisition results of the conventional algorithm for PRN id 20

Code shift value	749.945 chip = 749.945 / 1.023 MHz = 0.733 millisecond
Fine pseudorange	219747.87 meters
Sample that the code begins	11756
IF frequency	3993195 Hz.
Doppler frequency	-3195 Hz.
Conventional acquisition process duration	665.93 seconds

Conventional tracking of the satellite PRN number 20

250 milliseconds portion of navigation bits of the PRN 20 is presented in Figure 50. Figure 51 is Doppler frequency change during carrier tracking. On the other hand, Figure 52 is C/A code delay change during code tracking.

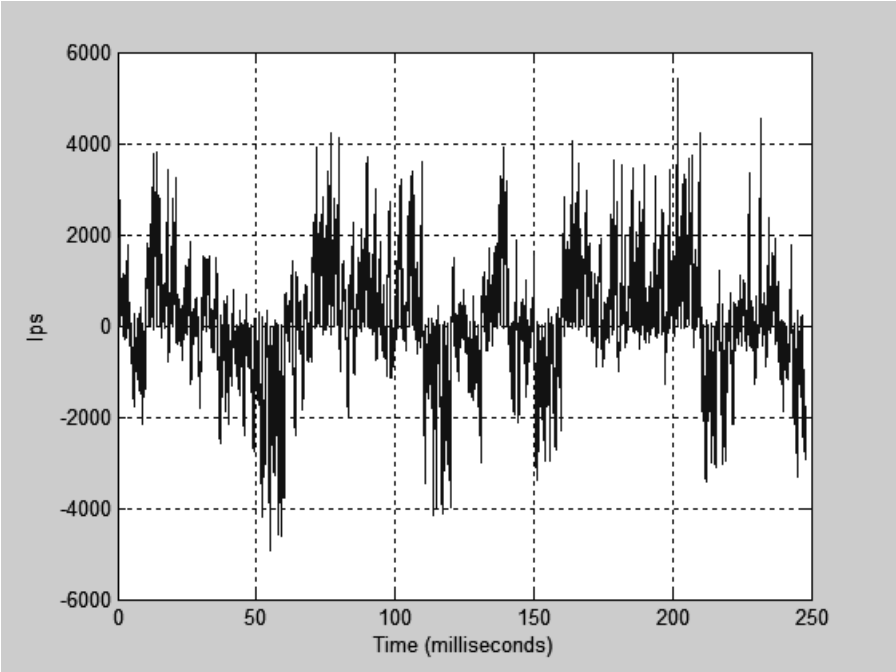

Figure 50. Navigation bits, output of the carrier tracking loop for PRN 20

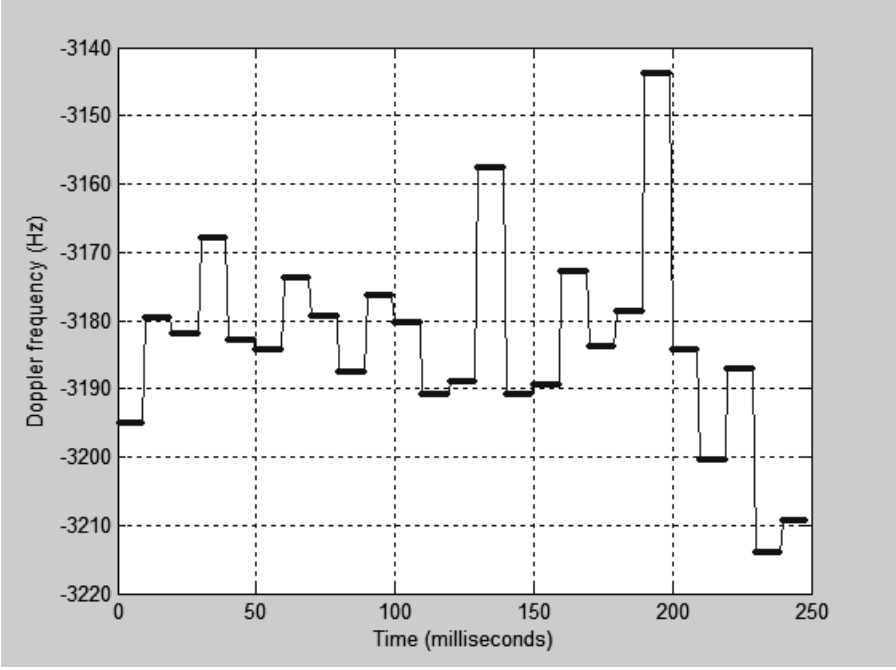

Figure 51. Doppler frequency obtained from carrier tracking for PRN 20

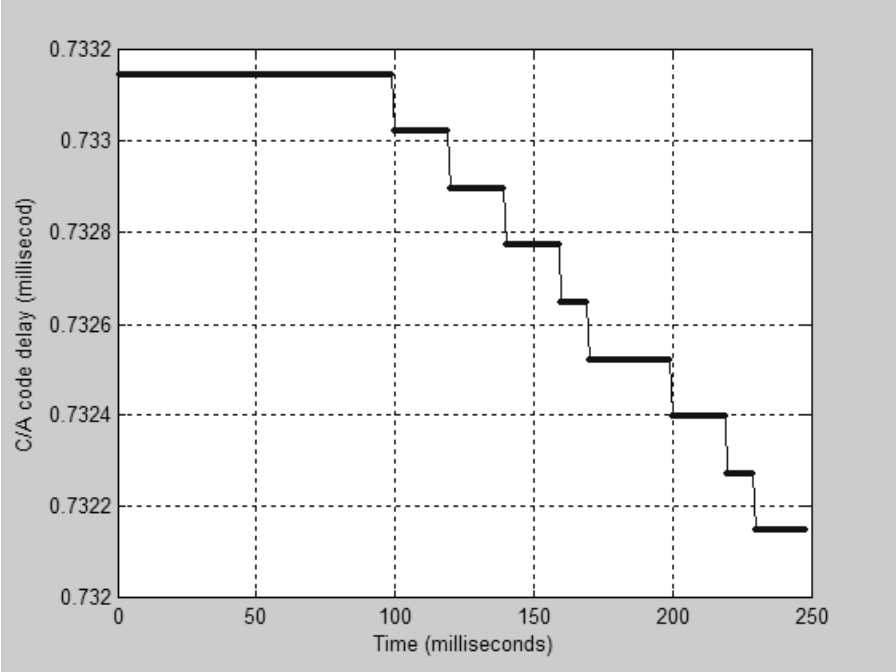

Figure 52. C/A code delay obtained from code tracking for PRN 20

Conventional signal processing results for PRN 15

PRN 15 is not expected to be found in the signal. Figure 53 verifies this expectation. There is not any obvious peak in the result of PRN 15. So, satellite PRN 15 can not be detected from the signal. PRN 15 is an unallocated PRN number at the time of the data collection. Therefore, tracking process is not applied.

5.1.4 BAAS Signal Processing Results

According to the Table 11, two of the most probable satellites in the signal are 11^{th} and 20^{th} PRN numbered satellites since they have the highest elevation angles. On the other hand, satellite with PRN id 15 is not expected to be found since it is not in the sky at the time of data collection. Therefore, PRN 11, 20 and 15 will be processed in order to present BAAS signal processing performance. Satellite numbers are selected same with the satellites in the conventional signal processing in order to compare the performance of the two methods.

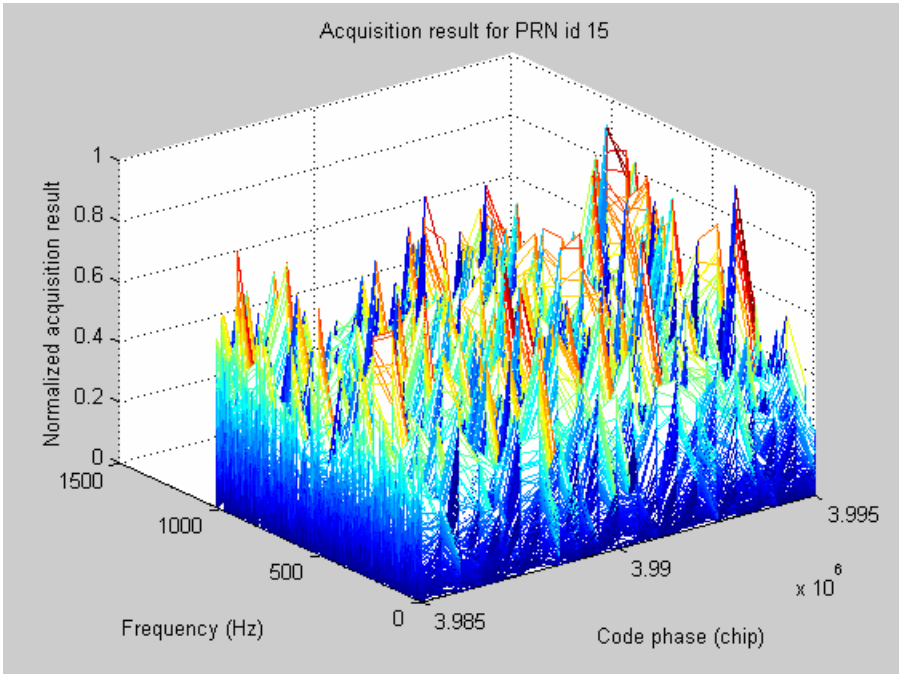

Figure 53. Acquisition metric for the satellite PRN 15

In BAAS signal processing, acquisition is provide by circular correlation. C/A code and carrier frequency is searched in frequency domain. Carrier frequency and C/A code delay are the main outputs of the acquisition. Fine frequency estimation is also applied to the acquisition in order to provide accurate Doppler frequency estimation. Furthermore, acquisition process duration is presented by using *tic* and *toc* commands of Matlab in order to comment on the methods speed. Used PC is Pentium 4(R) CPU 2.00 GHz and has 512 MB of RAM.

Circular correlation acquisition parameters are selected as below:

- Code acquisition resolution is 0.0638 chip
- Carrier acquisition resolution is within a few Hz.

When it comes to BAAS tracking, once the C/A code phase and the fine carrier frequency are established, tracking can proceed. The tracking contains two parts: carrier tracking and C/A code tracking. Carrier tracking is achieved by tracing the phase of the input signal. C/A code is tracked by FFT values of the early, late and prompt signals. During the tracking process, threshold is used to control whether the signal is missed or not. If the satellite signal is missed, acquisition must be re-processed in order to acquire the signal again. BAAS tracking parameters are selected below:

- Code tracking is designed by. Code delay is updated at every 10 millisecond intervals.

- Code tracking resolution is 0.0638 chip.

- Carrier signal frequency is updated by the fine frequency obtained from the phase difference of the two adjacent samples taken at 10 milliseconds intervals. Hence, carrier frequency is updated every 10 milliseconds.

  In BAAS tracking, navigation bits are observed from the raw phase measurement of the input GPS signal as well as the carrier tracking output. However, raw phase measurement should be corrected according to Figure 55. π phase shift stands for navigation bit change in the input signal.

- Signal presence is controlled by comparing the DFT amplitude of the prompt signal with a pre-determined threshold. Threshold is selected to be 2500. If the prompt signal DFT amplitude is below this threshold shows that the signal is missed and needed to be acquired.

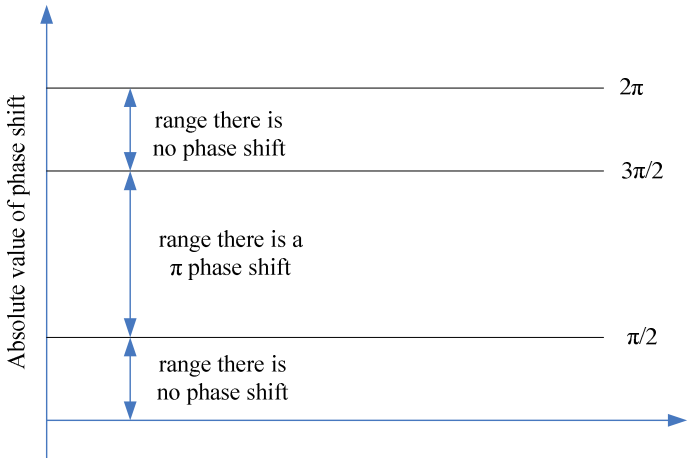

Figure 54. Thresholds of a phase shift

BAAS signal processing results for PRN 11

PRN 11 has elevation angle of 64.3º. Thus, satellite is expected to be found.

Signal acquisition by circular correlation of satellite PRN id 11

C/A code search and carrier frequency search are presented in the figures below. Obvious peak is observed as expected. Table 14 includes the acquisition outputs. According to this table C/A code of PRN 11 is shifted 0.6072 milliseconds because of the transmission from satellite to the user. If we multiply this delay with the speed of light, we can obtain the fine component of the pseudorange as shown in the table. Doppler frequency of PRN 11 is the frequency deviation from the IF frequency. Furthermore, process duration shows the speed of the circular correlation acquisition.

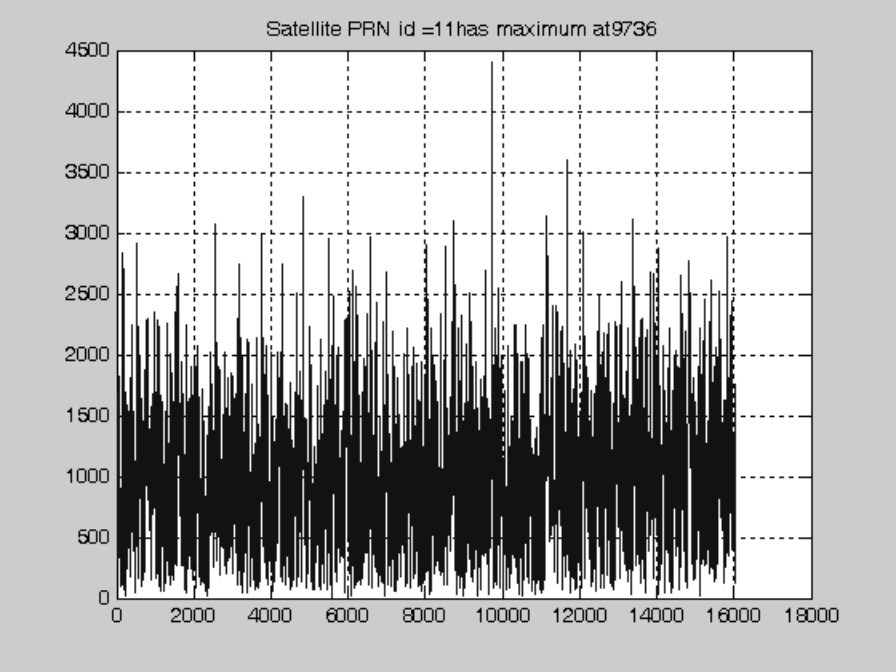

Figure 55. Code search of PRN 11, C/A code step vs. correlation value

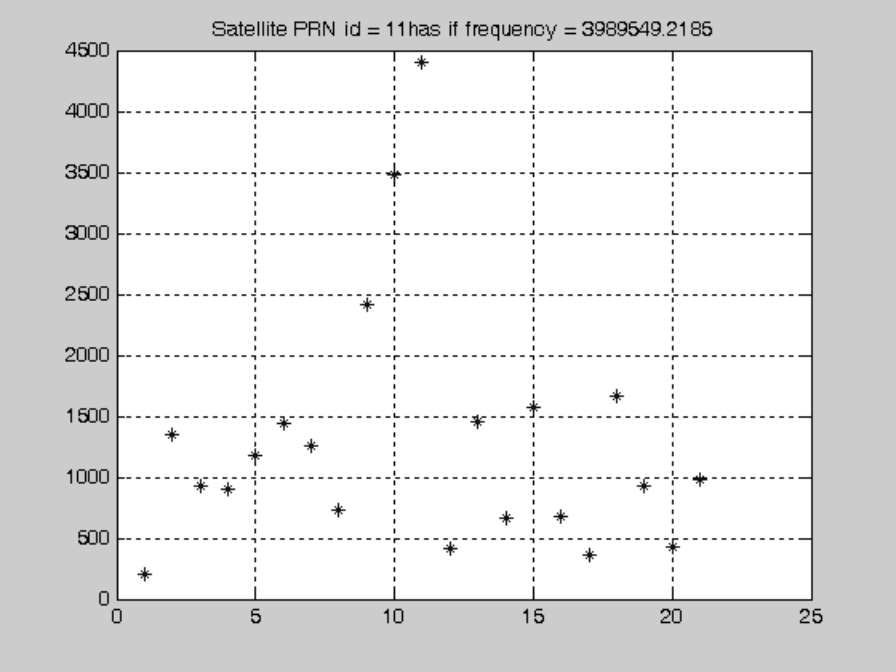

Figure 56. Frequency search of PRN 11, frequency step vs. correlation value

Table 14. Acquisition results of the circular correlation algorithm for PRN id 11

Code shift value	621.1368 chip = 621.1368 / 1.023 MHz = 0.6072 millisecond
C/A code beginning	9736
Fine pseudorange	182033.98 meters
IF frequency	3989549.2185 Hz.
Doppler frequency	450.7815 Hz.
BAAS acquisition process duration	20.19 seconds

BAAS signal tracking for satellite PRN id 11

250 milliseconds portion of navigation bits of the PRN 11 are presented in Figure 57. Figure 58 is the raw phase difference of the satellite signal. Figure 59 is the corrected phase shift of the input signal. Note that these three figures are consistent with each other since all of them show the navigation bits. Unexpected peaks in Figure 59 will be eliminated in bit synchronization section of the signal processing module.

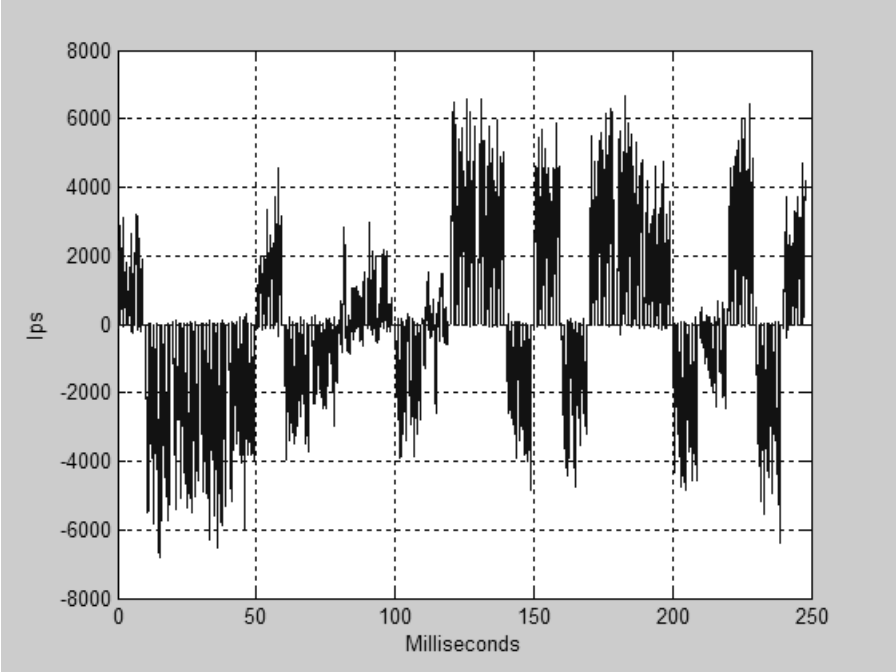

Figure 57. Prompt channel output, i.e. navigation bits of PRN 11

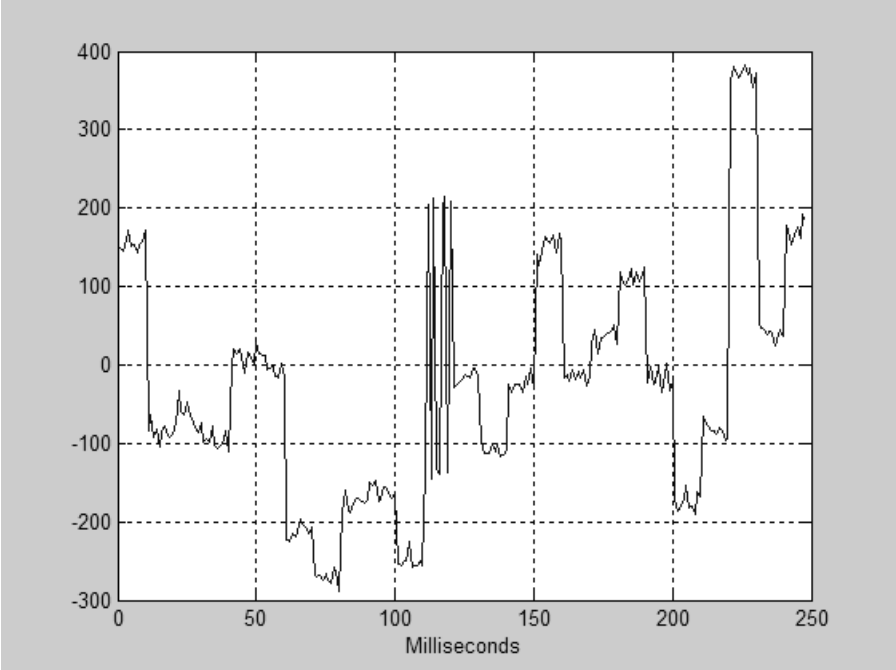

Figure 58. Raw phase of the input signal for satellite PRN 11

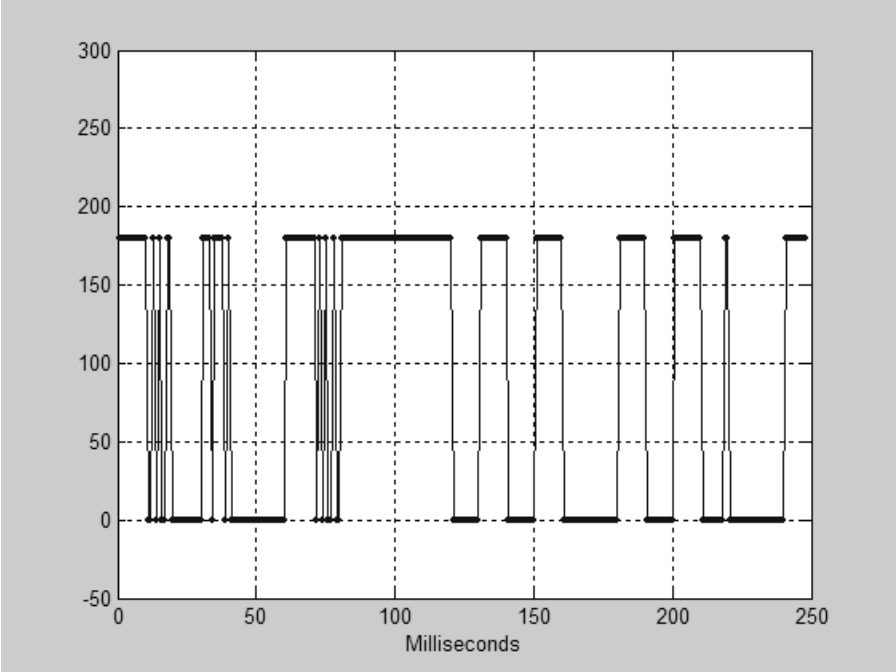

Figure 59. Phase of the input signal for satellite PRN 11

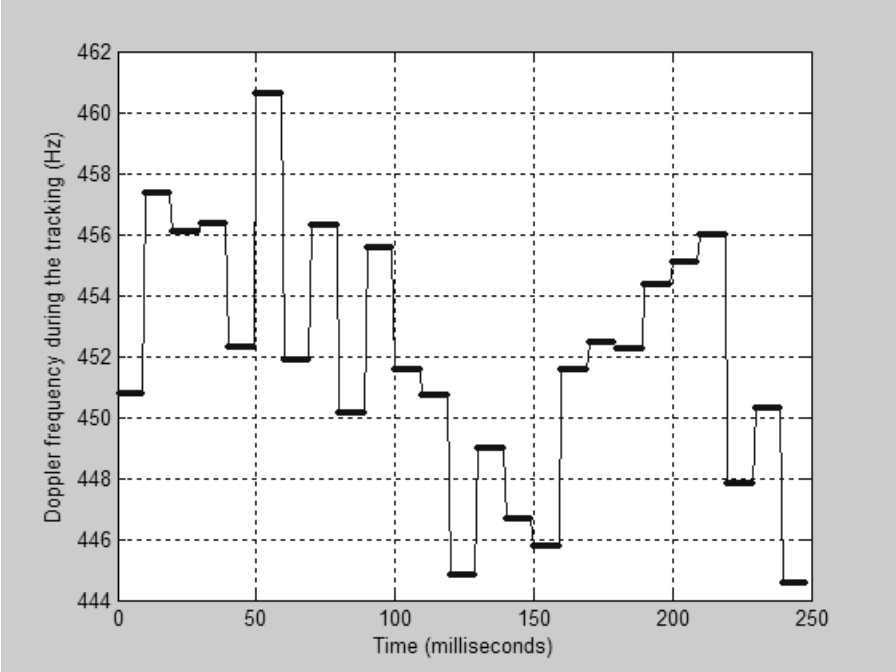

Figure 60. Doppler frequency of PRN 11 during tracking

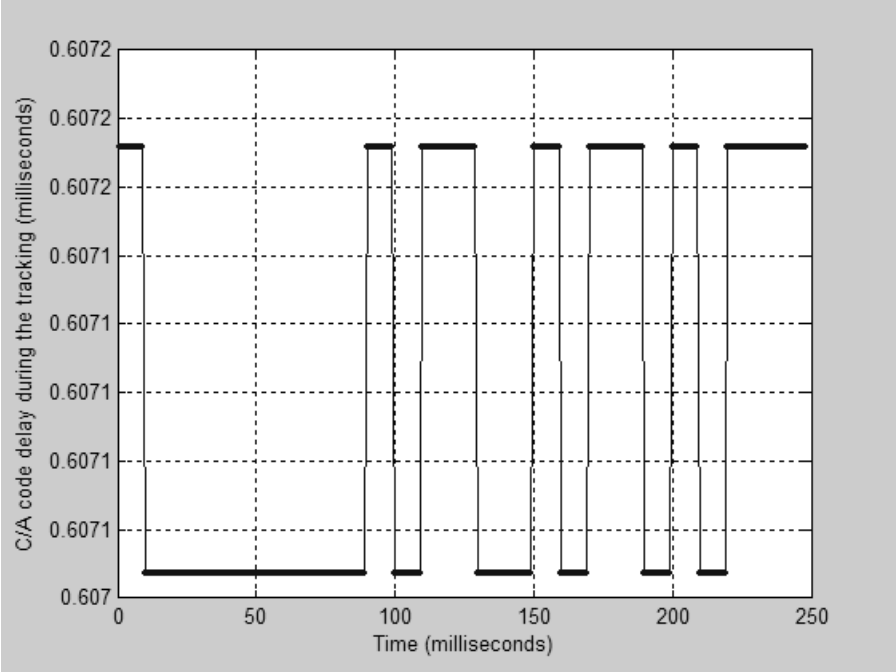

Figure 61. C/A code delay of PRN 11 during tracking

BAAS signal processing results for PRN 20

Signal acquisition by circular correlation of satellite PRN 20

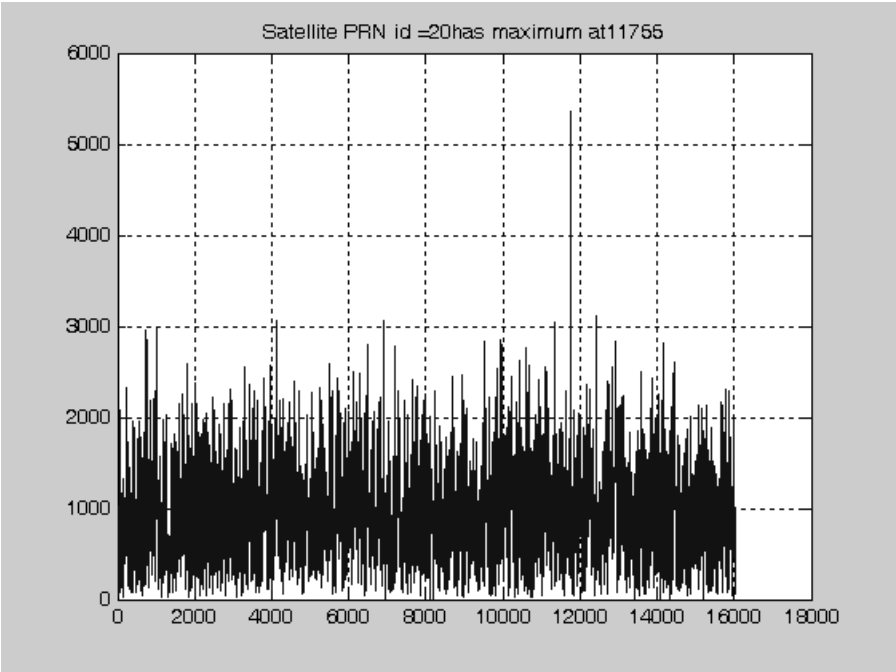

Figure 62. Code search of PRN 20, C/A code step vs. correlation value

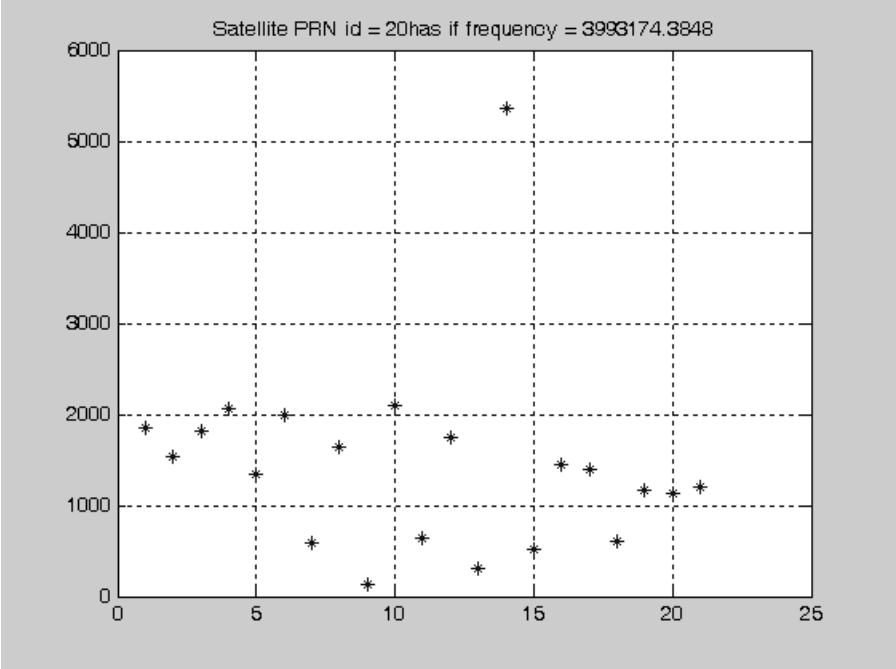

Figure 63. Frequency search of PRN 20, frequency step vs. correlation value

Table 15. Acquisition results of the circular correlation algorithm for PRN 20

Code shift value	749.945 chip = 749.945 / 1.023 MHz = 0.733 msec
Sample that the code begins	11755
Fine pseudorange	219747.87 meters
IF frequency	3993174.3848 Hz.
Doppler frequency	-3174.384 Hz.
BAAS acquisition process duration	19.4 seconds

BAAS signal tracking for satellite PRN 20

250 milliseconds portion of navigation bits of the PRN 20 are presented in Figure 64. Figure 65 is the raw phase difference of the satellite signal. Figure 66 is the corrected phase shift of the input signal. Note that these three figures are consistent with each other since all of them shows the navigation bits. Unexpected peaks in Figure 66 will be eliminated in bit synchronization section of the signal processing module.

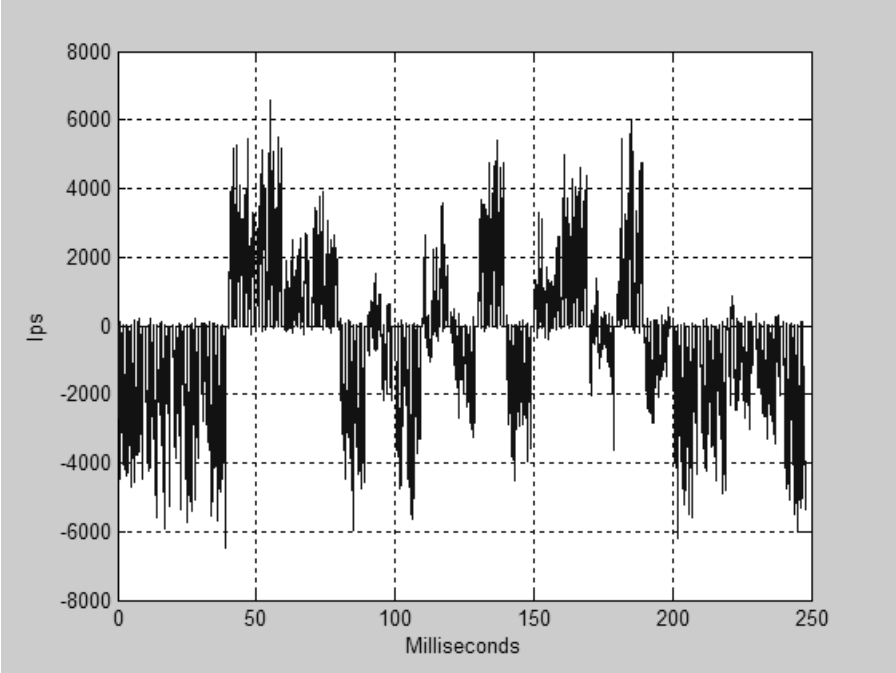

Figure 64. Prompt channel output, i.e. navigation bits of PRN 20

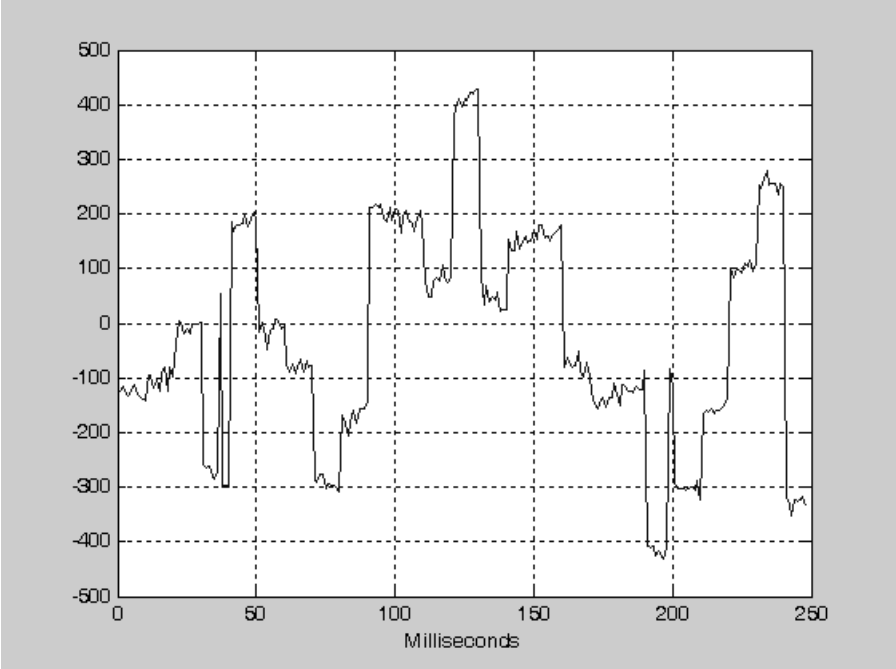

Figure 65. Raw phase of the input signal for satellite PRN 20

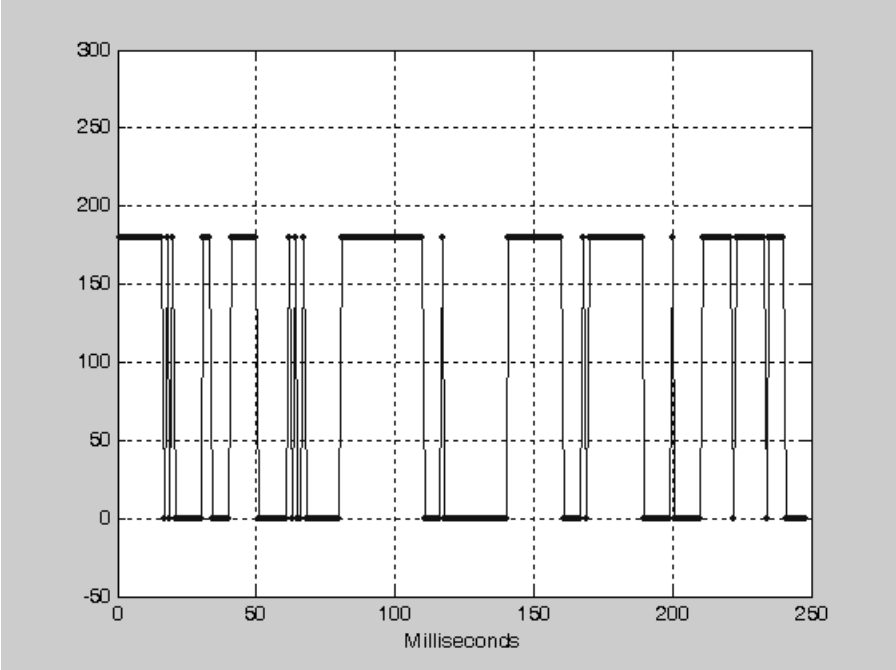

Figure 66. Phase of the input signal for satellite PRN 20

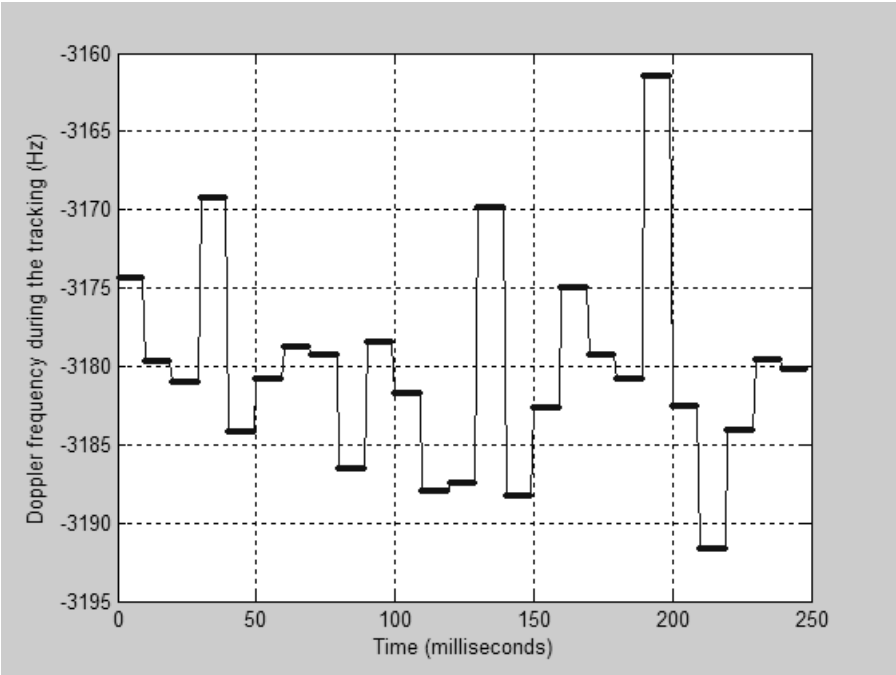

Figure 67. Doppler frequency of PRN 20

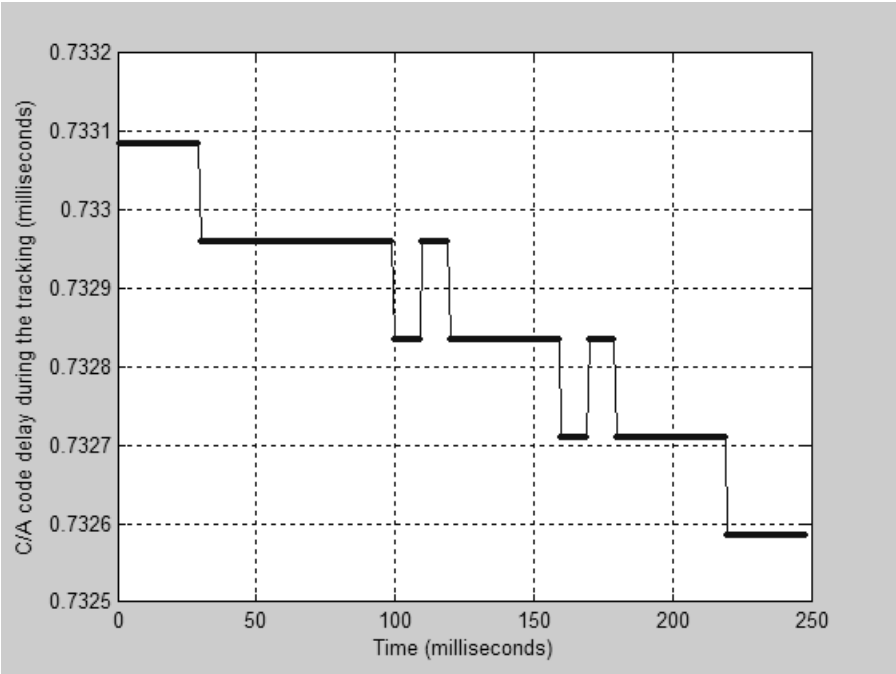

Figure 68. C/A code delay of PRN 20

5.1.5 Comparison of conventional and BAAS signal processing methods

Conventional and BAAS signal processing methods are compared according to their process speed and the outputs.

Acquisition methods are compared according to their speed, Doppler frequency outputs and computational complexity. Acquisition speed is a critical parameter for a GPS specification. Since conventional acquisition is an exhaustive serial search, it spends much more time than the circular correlation acquisition. Circular correlation process is applied in frequency-domain. Circular correlation method is faster that the conventional methods since has less computational complexity.

Furthermore, circular correlation acquisition outputs Doppler frequency accurately when we compare it with the conventional acquisition Doppler output. Circular correlation acquisition provides fine frequency estimation with a few Hertz resolutions. On the other hand, conventional acquisition obtains 25 Hz fine frequency resolution. C/A code acquisition resolution is same for both of the methods; same results are obtained for C/A code beginning points.

When it comes to the tracking performance comparison, tracking speeds of the two tracking methods are not different significantly. However, Doppler frequency outputs are different. Doppler frequency search starts with different values but they reach to almost same value in the two types of tracking. Conventional tracking starts with 475 Hz Doppler frequency and conventional tracking ends with 444 Hz Doppler frequency after 250 milliseconds of operation. BAAS tracking starts with 451 Hz Doppler frequency and conventional tracking ends with almost 444 Hz Doppler frequency after 250 milliseconds of operation.

Main objective of tracking is to obtain navigation bits. GPS navigation bits are almost same for both of the methods. Some deviations in navigation bits are observed between the conventional and BAAS outputs. However, almost same navigation bits are obtained.

5.2 Software Navigation Algorithm Results

Software navigation algorithms are driven by the real raw data obtained from a commercial GPS receiver. Results are compared by the outputs of the real commercial GPS receiver for a moving platform. However, stationary platform results are compared with precisely known position.

Program performance will be presented for dynamic case and stationary case.

5.2.1 Dynamic case

In dynamic experiment, GPS receiver Ashtech^{TM} is placed in a vehicle and path in ODTU was fallowed. Figure 69 illustrates the experiment set-up and the block diagram of the software navigation algorithm. Table 16 summarizes the raw data messages that have been collected from the GPS receiver for the software navigation algorithm inputs.

Table 16. Ashtech G12 Board GPS receiver Messages

Message	Content
SNV	Clock correction and ephemeris parameters
PBN	Raw pseudorange and Doppler frequency
CT1	Calculated receiver position
ION	Ionospheric correction data

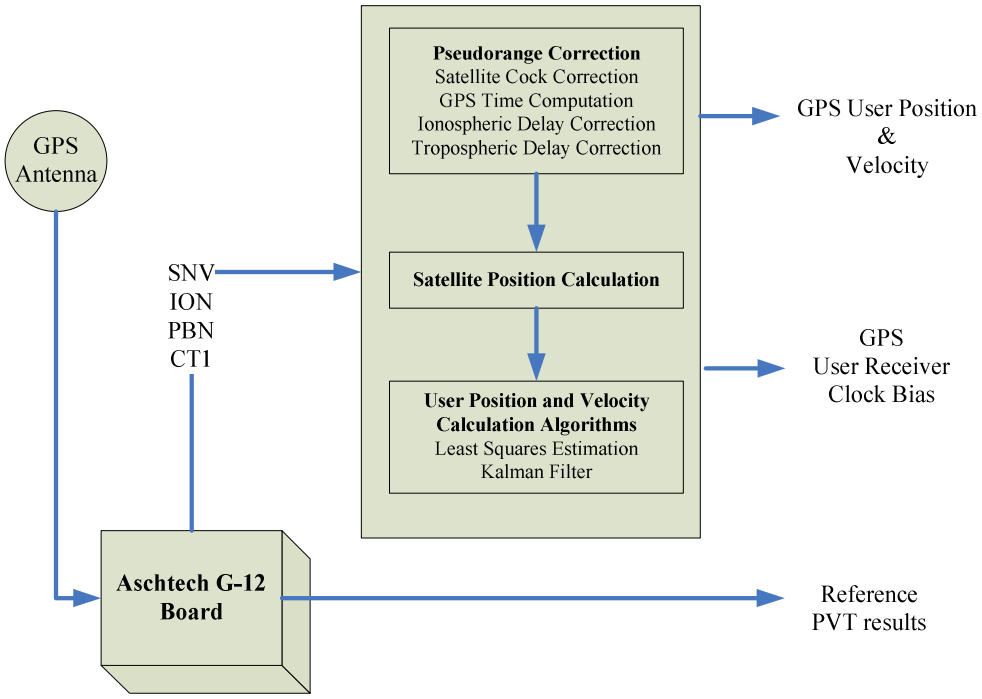

Figure 69. Dynamic experiment set-up

5.2.1.1 Pseudorange correction

It is meaningful to give an example of pseudorange errors magnitude before presenting the dynamic case result. In Table 17, calculated angles, atmospheric corrections and the satellite clock corrections are presented for the satellites in view at a time in the path. For ionospheric and tropospheric effects, it is obvious that correction magnitude is maximum for the minimum elevation angled satellite. It is also clear that the satellite clock correction may be any value since it is special for a satellite.

Table 17. Pseudorange errors

PRN ID	Elevation angle (degree)	Azimuth angle (degree)	Ionospheric correction(m)	Tropospheric correction(m)	Satellite clock correction(m)
18	53	48	8.21	3.05	2248.13
17	18	79	15.46	7.54	2068.31
14	29	15	12.87	4.93	-6872.99
31	38	-5	10.03	3.94	31695.89
23	49	46	8.53	3.18	7136.52
15	52	60	8.34	3.09	28285.68
3	66	-72	7.24	2.65	36006.49

5.2.1.2 Path experiment results

Receiver position and the velocity are computed in ECEF coordinate system. Error functions are evaluated by choosing the Ashtech™ GPS results as reference. Results will be given in two subsections. PVT computations will be achieved by LSE and Kalman filter method by using the same input raw data files.

5.2.1.2.1 LSE results

Least squares estimation method is used for position, velocity, receiver clock bas and receiver clock drift. Three dimensional path results are presented in Figure 70. Figure 71 is the x-y-z coordinate errors. Error is calculated by comparing Ashtech™ GPS position output and the calculated output. Furthermore, mean value of the x-y-z errors are illustrated in Figure 71, too.

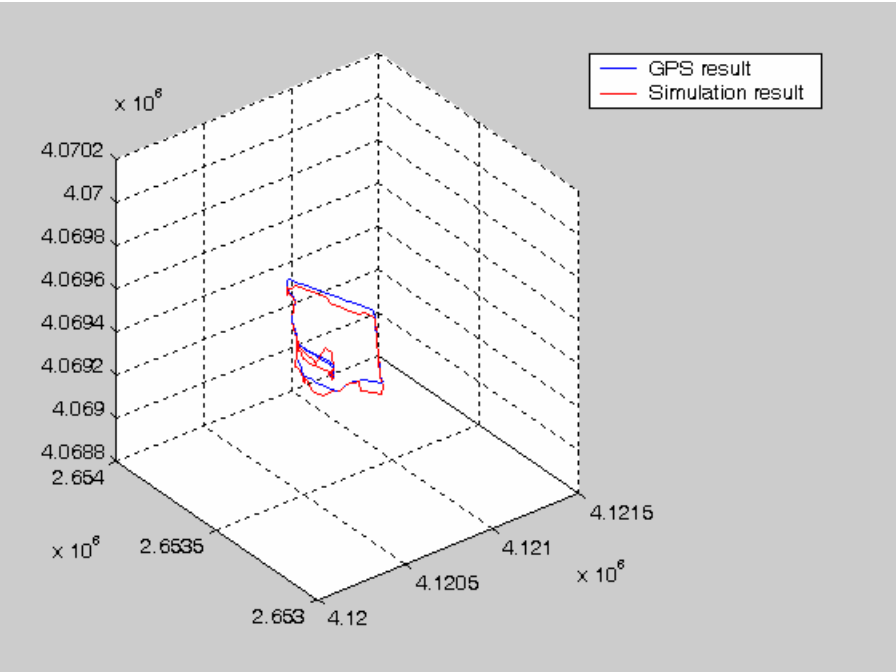

Figure 70. Three dimensional X-Y-Z path results (meters)

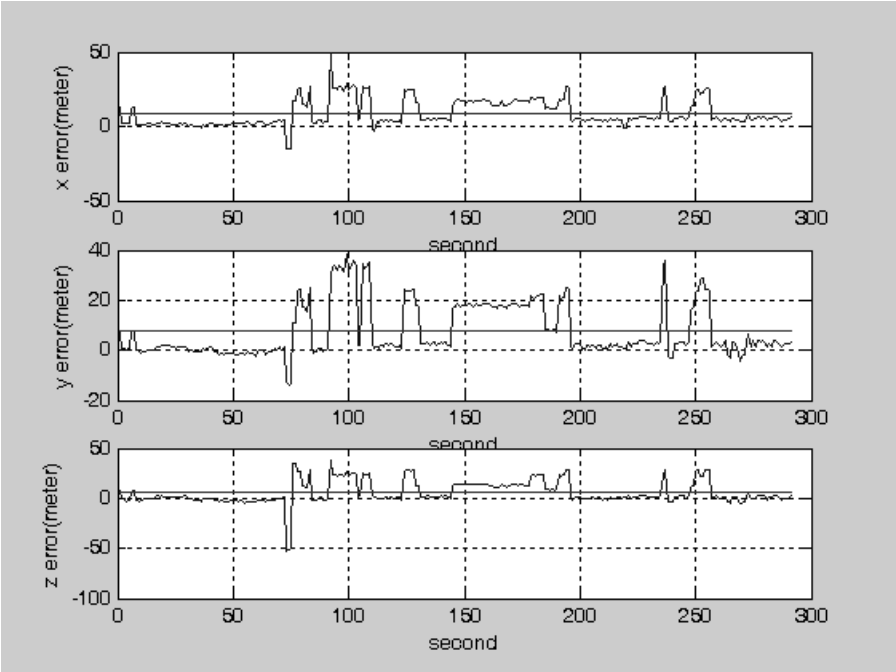

Figure 71. X-Y-Z position errors

Table 18 includes the maximum, minimum and mean value of the errors in Figure 71.

Table 18. LSE method errors

	X (m)	Y (m)	Z (m)
Maximum value of the error	49.312	38.18	37.934
Minimum value of the error	0.0037224	0.00715	0.0000463
Mean value of the error	8.56	7.606	5.56

Figure 72 is presented to show the velocity error of the LSE results. Reference velocity information is obtained by differentiating the position obtained from the Ashtech™ GPS receiver. From the figure, two peaks are observed at about 75th milliseconds since new satellites enter or exit at that time of the path experiment.

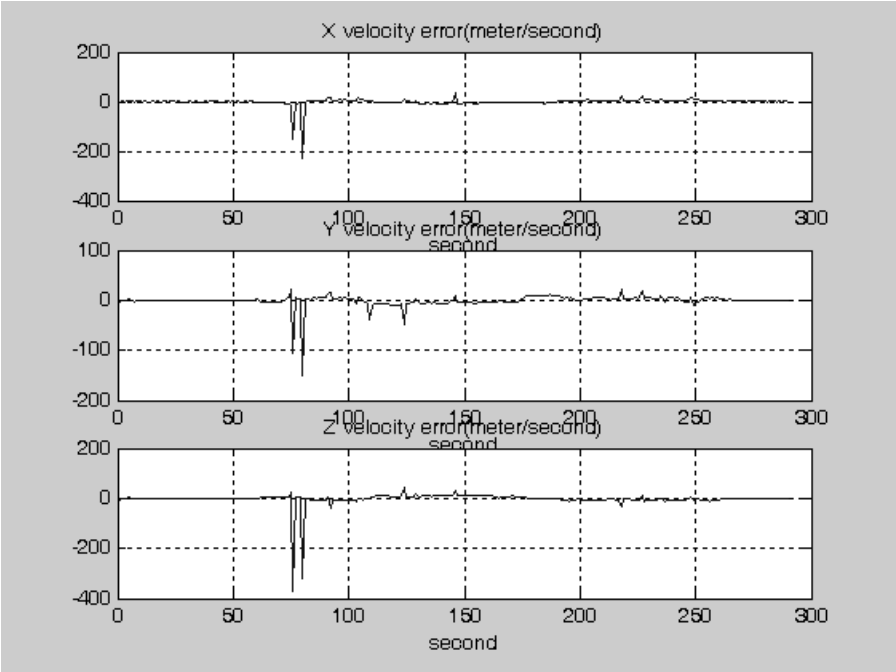

Figure 72. X-Y-Z velocity errors

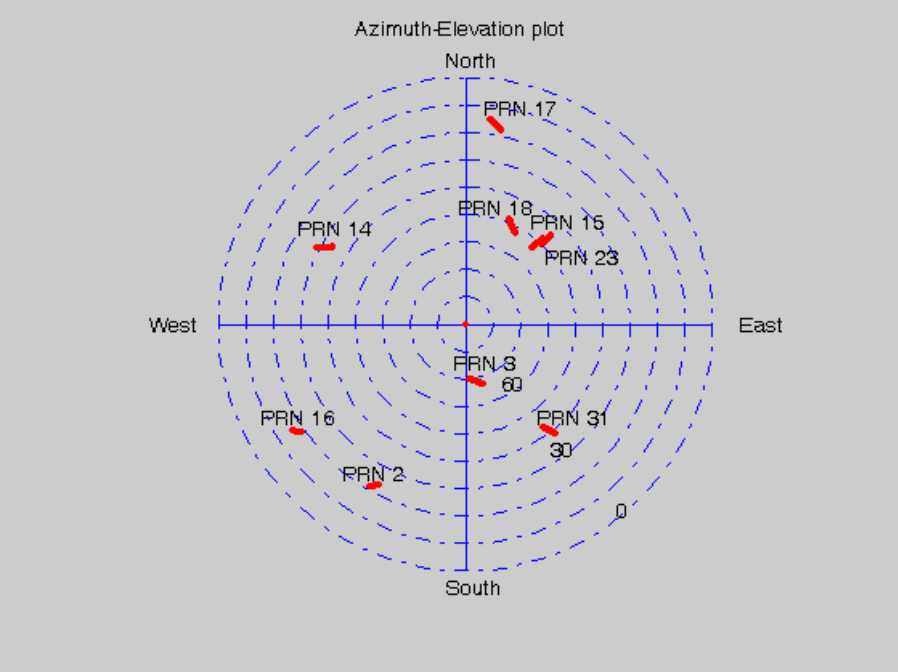

Figure 73. Azimuth and elevation plot

Figure 73 is the azimuth and elevation plot and it is obtained from the satellite positions. Center point of the graph is the zenith in the sky.

Figure 74 is the calculated receiver clock bias. Note that receiver clock bias is given in meters, which is actually receiver bias in seconds multiplied by speed of light. Receiver clock bias calculated at a time is directly observed in the pseudoranges of the satellites that are tracked at that time.

Figure 75 is the calculated receiver clock drift. Similar to the velocity error graphics, peaks are observed at about 75^{th} seconds since new satellites enter or exit at that time of the path experiment.

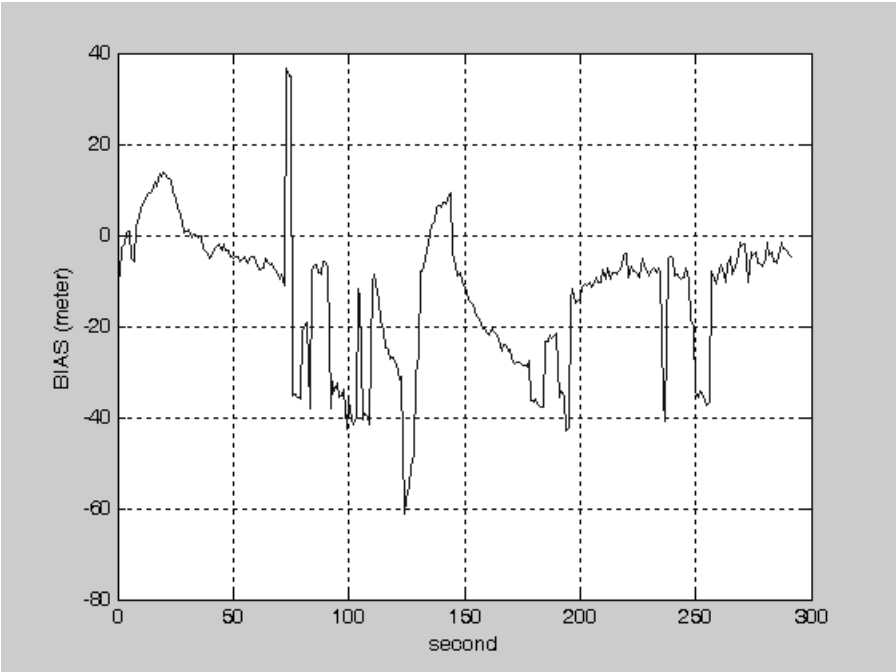

Figure 74. Calculated bias of the receiver clock

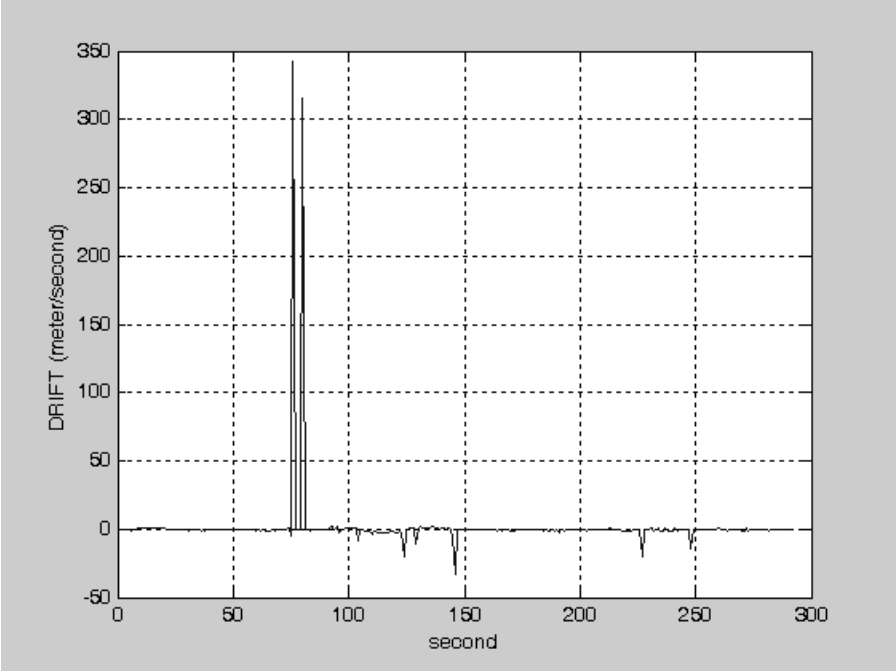

Figure 75. Calculated drift of the receiver clock

5.2.1.2.2 Kalman Filter results

Kalman filter is used for position, velocity, receiver clock bas and receiver clock drift. Three dimensional path results are presented in Figure 76. Figure 77 is the x-y-z coordinate errors and Figure 78 is the mean square position error. GPS position outputs and the calculated position outputs are shown in Figure 79, Figure 80 and Figure 81 in order to show the consistency of the position results. Position error is calculated by comparing Ashtech™ GPS position output and Kalman filter output. Furthermore, mean value of the x-y-z errors are illustrated in Figure 77, too. Table 19 includes the maximum, minimum and mean value of the errors in Figure 77.

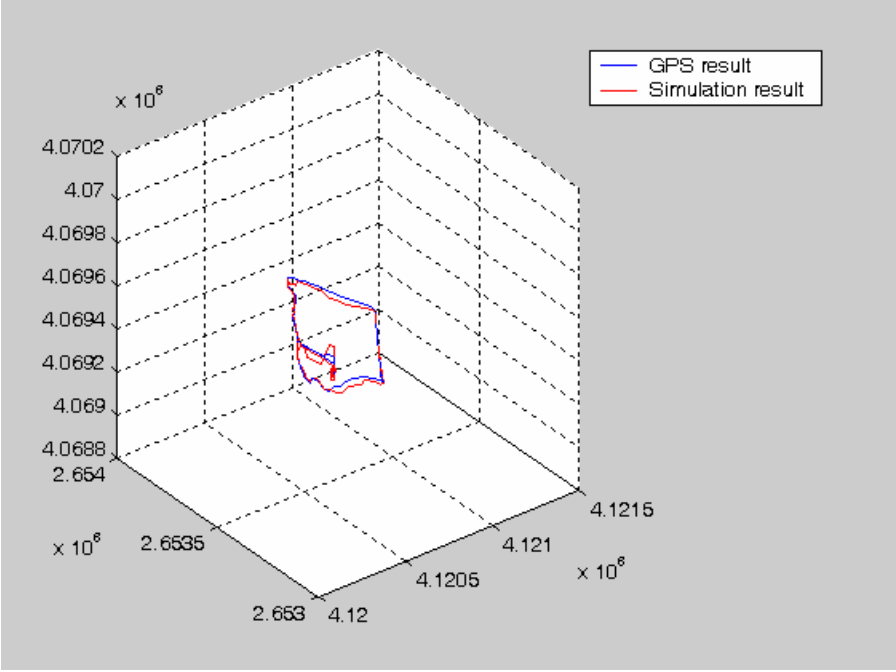

Figure 76. Three dimensional X-Y-Z path results (meters)

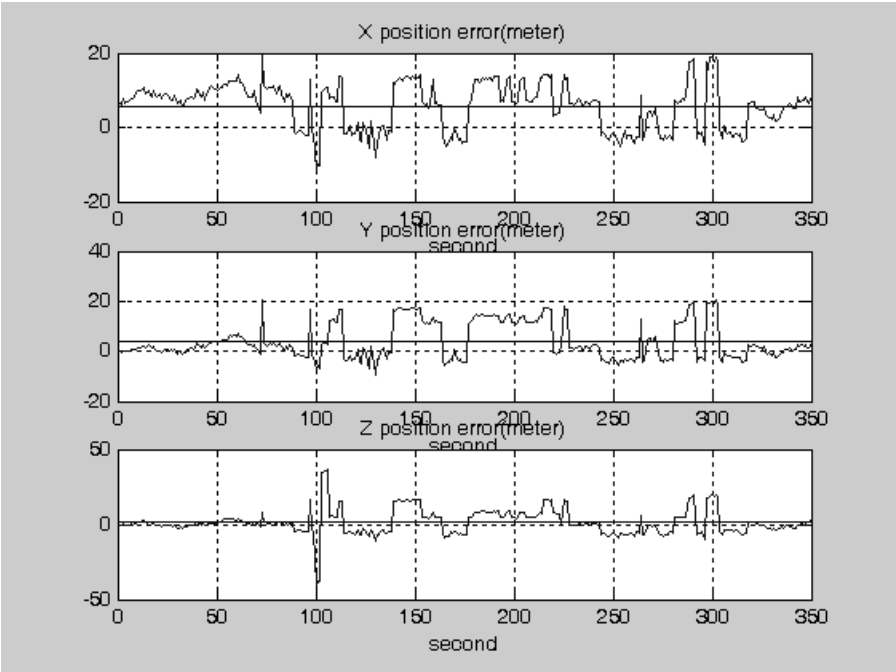

Figure 77. X-Y-Z position errors

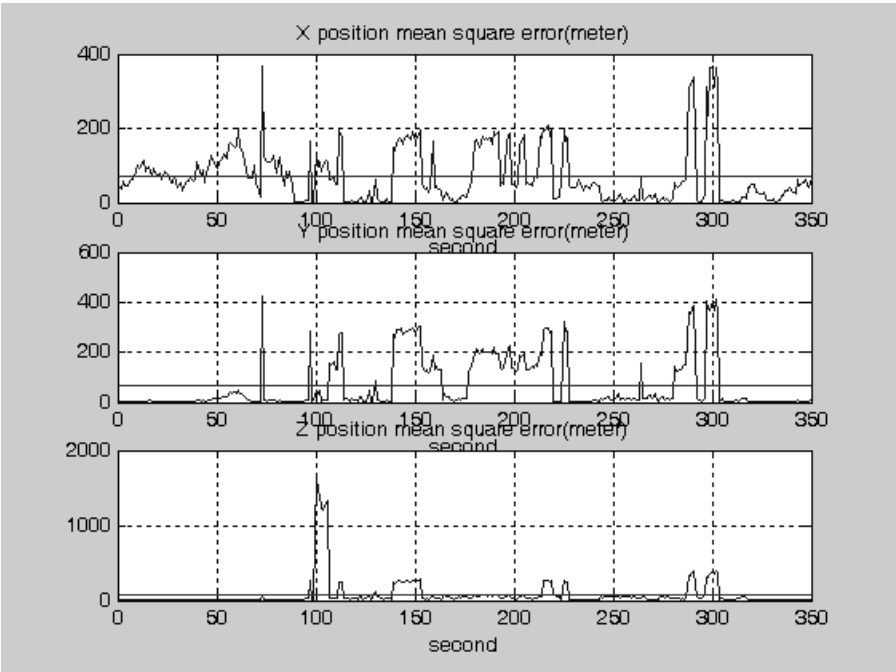

Figure 78. X-Y-Z position mean square errors

Table 19. Kalman Filter method errors

	X (m)	Y (m)	Z (m)
Maximum value of the error	19.19	20.64	36.38
Minimum value of the error	0.1418	0.0174	0.00099
Mean value of the error	5.8	4.021	1.37

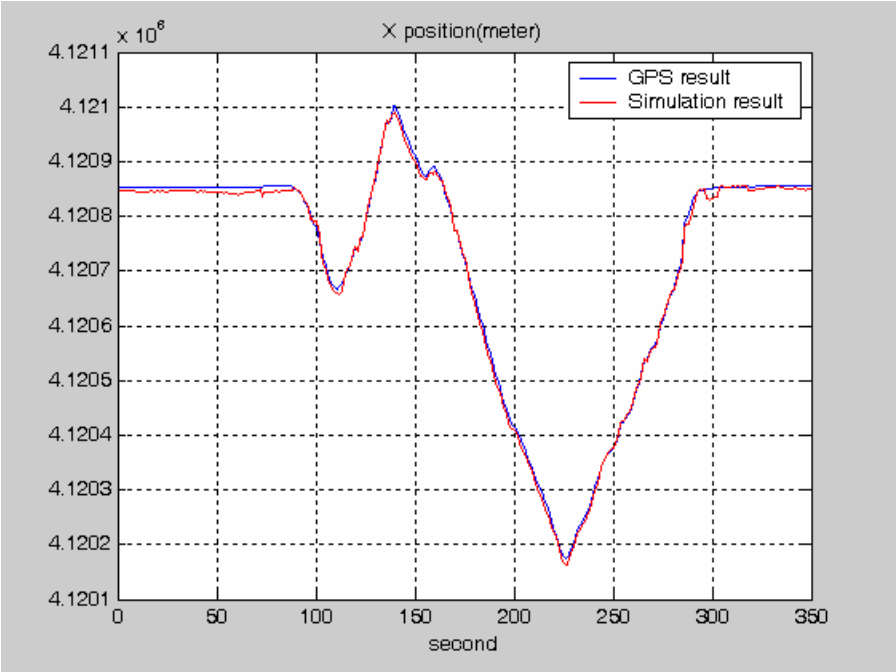

Figure 79. X position results

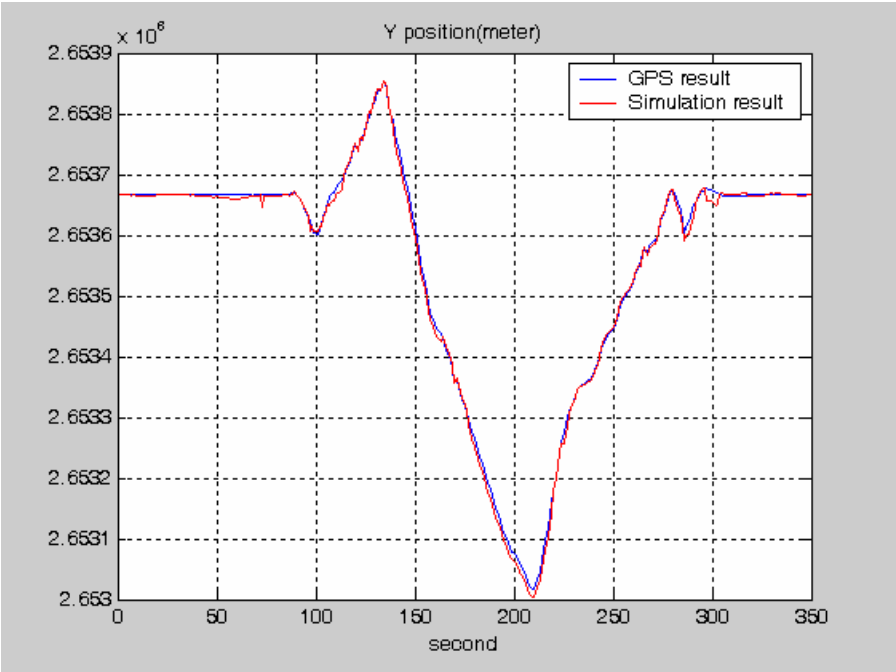

Figure 80. Y position results

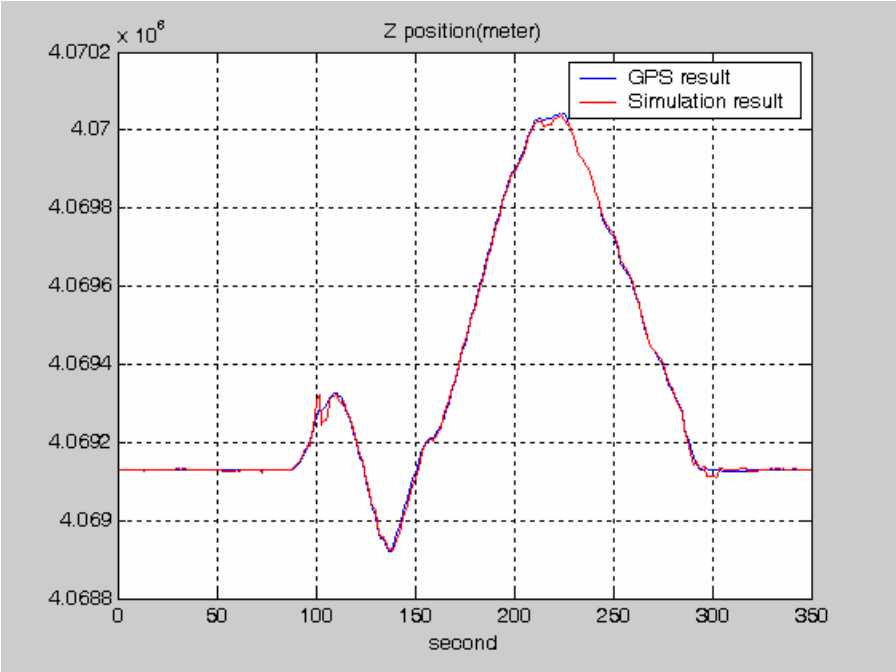

Figure 81. Z position results

Figure 82 is presented to show the velocity error of Kalman filter results. Reference velocity information is obtained by differentiating the position obtained from the Ashtech™ GPS receiver. From the figure, two peaks are observed at about 75th milliseconds since new satellites enter or exit at that time of the path experiment as observed in LSE results.

Figure 83 is the azimuth and elevation plot and it is obtained from the satellite positions. Center point of the graph is the zenith in the sky.

Figure 85 is the calculated receiver clock bias. Note that receiver clock bias is given in meters, which is actually receiver bias in seconds multiplied by speed of light. Receiver clock bias calculated at a time is directly observed in the pseudoranges of the satellites that are tracked at that time.

Figure 85 is the calculated receiver clock drift. Similar to the velocity error graphics, peaks are observed at about 75th seconds since new satellites enter or exit at that time of the path experiment.

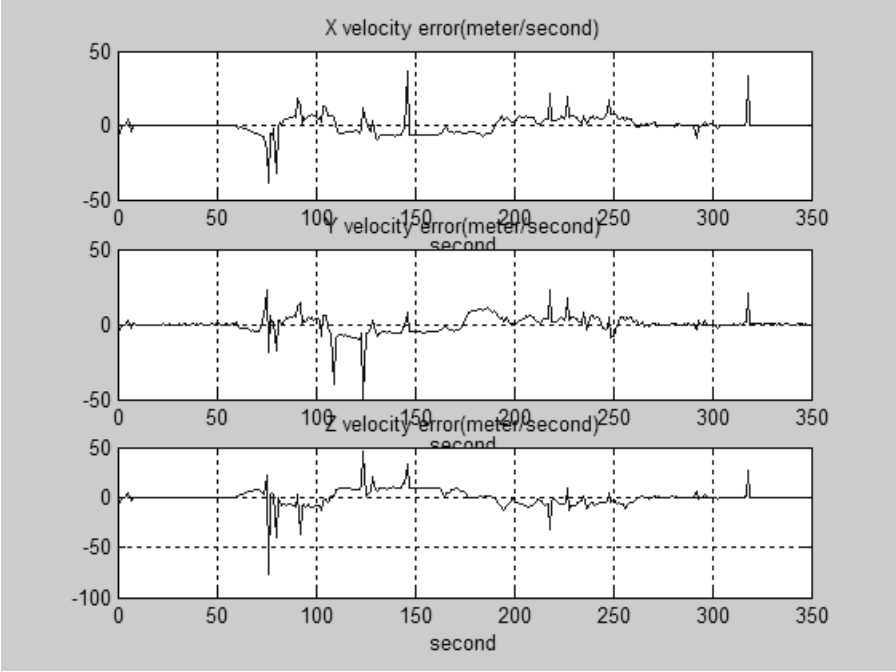

Figure 82. X-Y-Z velocity errors

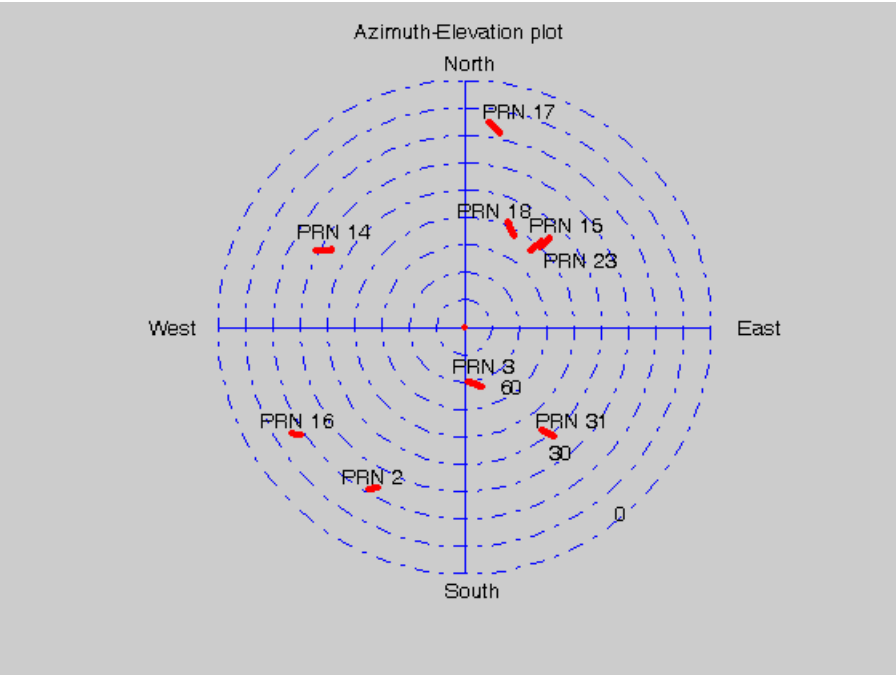

Figure 83. Azimuth-elevation plot

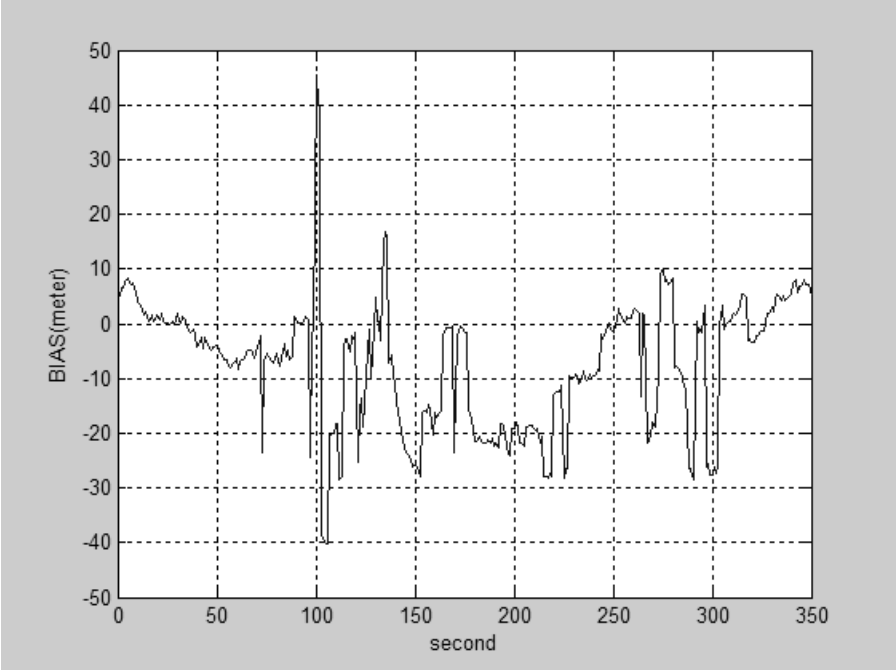

Figure 84. Calculated bias of the receiver clock

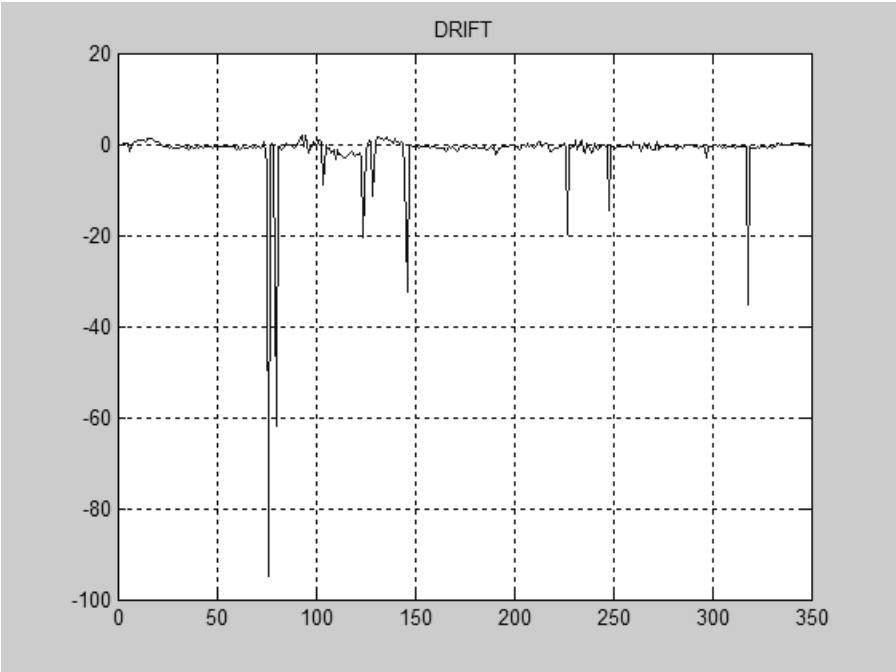

Figure 85. Calculated drift of the receiver clock

5.2.1.2.3 Kalman filter and LSE methods comparison

In order to compare the LSE and the Kalman Filter methods, we should compare the Table 18 and Table 19. Kalman filter results with less error and smoother position outputs in dynamic case. However, LSE method has less computational complexity than Kalman filter.

5.2.2 Stationary case

In stationary experiment, our navigation algorithm results are compared with respect to a precisely known position. Trimble™ GPS receiver antenna is located at a precisely known position. Antenna is located at the precisely known point in METU. By this experiment both of the implemented software receiver and the commercial Trimble™ GPS receivers performance are examined. Experiment set-up is illustrated in Figure 86. In PVT computation, LSE method is used.

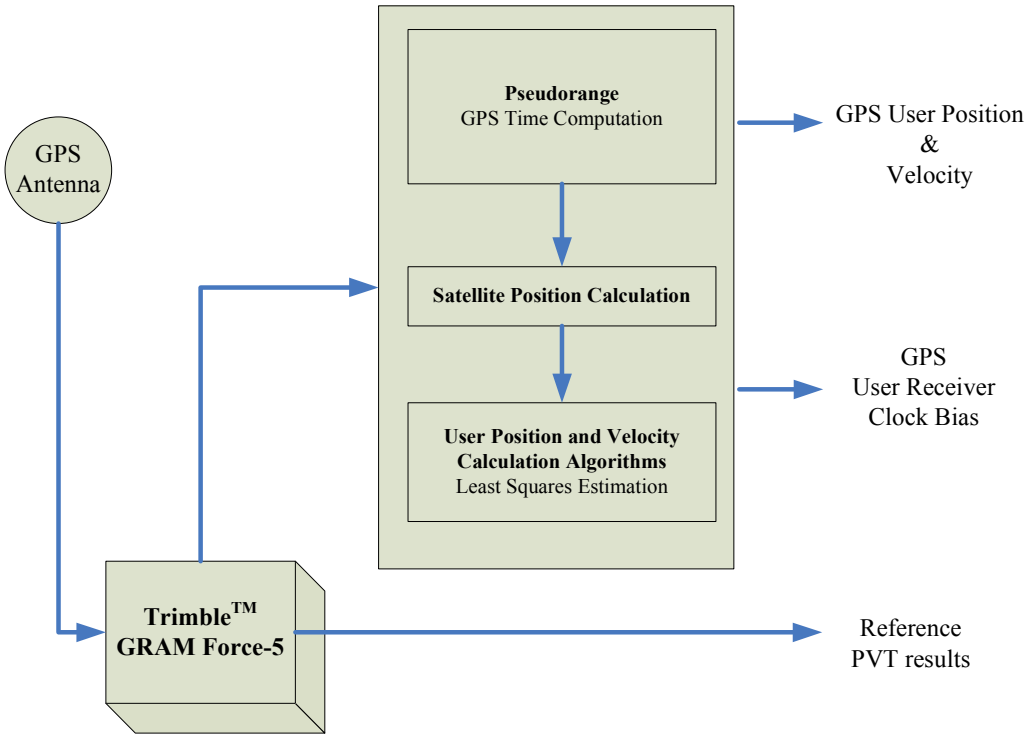

Figure 86. Stationary experiment set-up

Precisely known position in ECEF coordinate is: X = 4120867.043 meters, Y = 2653678.999 meters and Z = 4069126.699 meters. Software navigation algorithm and Trimble™ GPS error functions are computed. Figure 87, Figure 88 and Figure 89 are x-y-z errors of the software navigation algorithm. Figure 90, Figure 91 and Figure 92 are x-y-z errors of the Trimble™ outputs.

From the error functions, one can observe that Trimble™ GPS produces smoother results than the implemented navigation section. Table 20 summarizes the raw data messages that have been collected from the GPS receiver for the software navigation algorithm inputs.

Table 20. Trimble GPS receiver Messages

Message	Content
LOS	Raw pseudorange and Doppler frequency
Time Mark Message	Calculated receiver position by Trimble
Ephemeris-1 & Ephemeris-2	Clock correction and ephemeris parameters

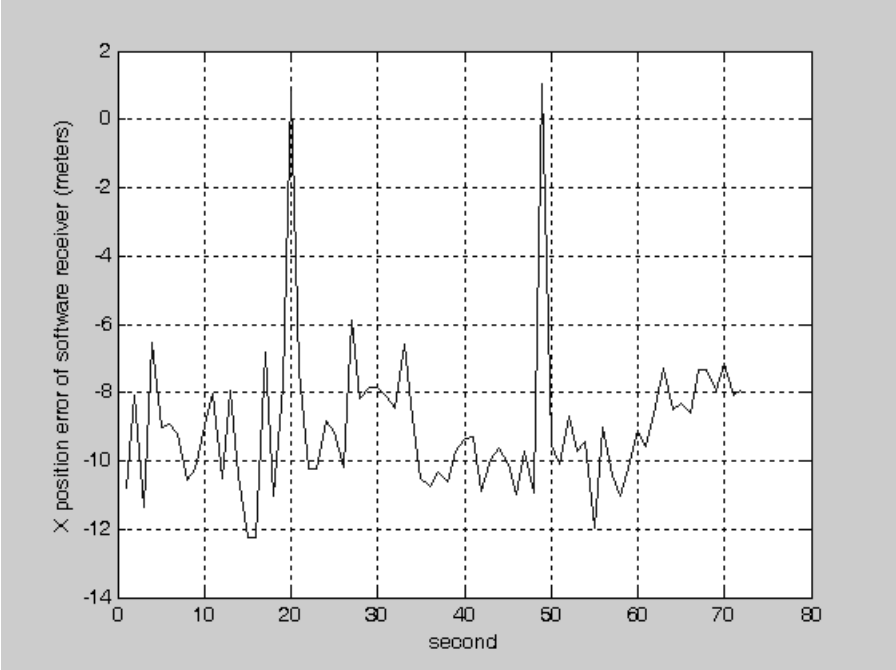

Figure 87. X position error of the software receiver

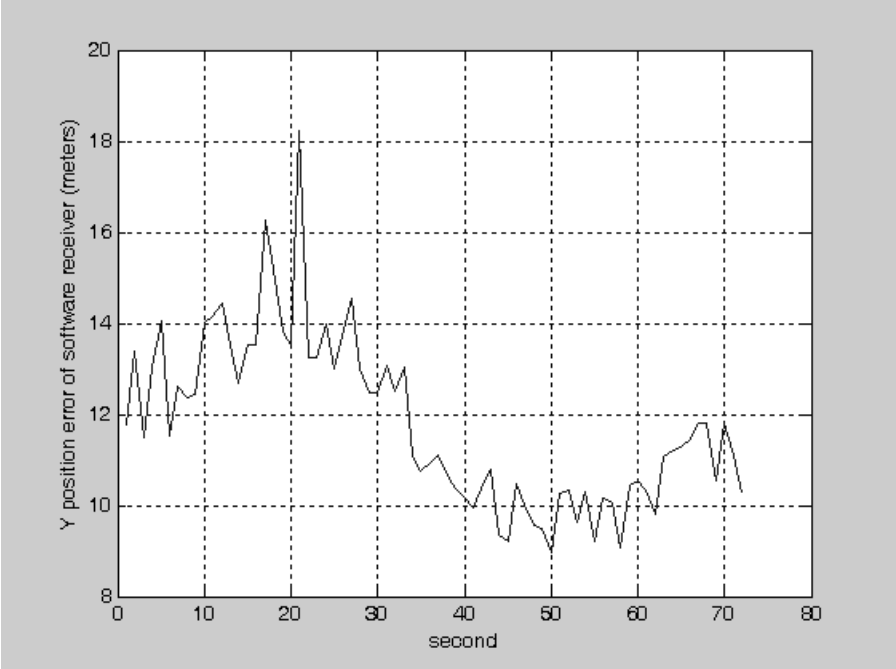

Figure 88. Y position error of the software receiver

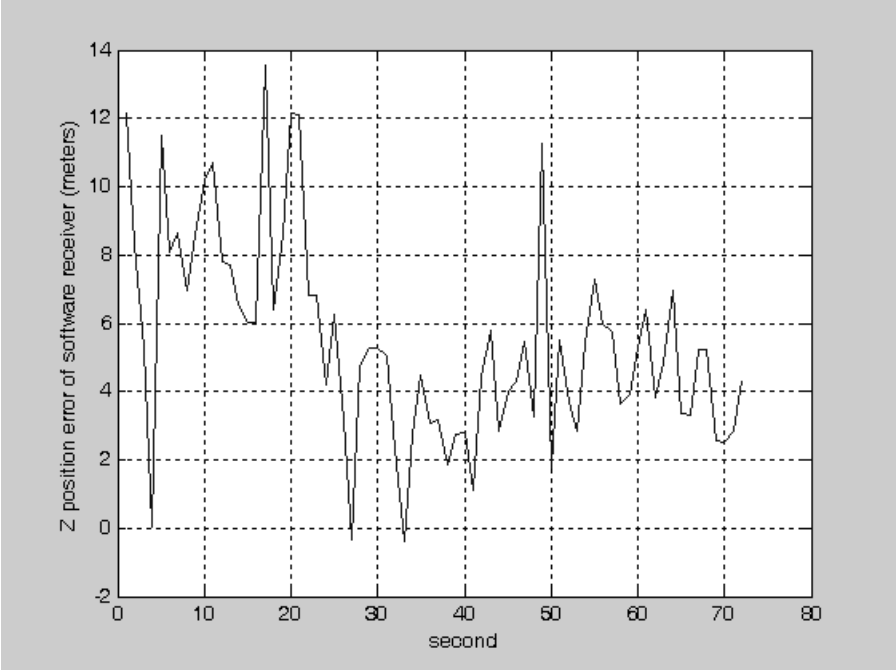

Figure 89. Z position error of the software receiver

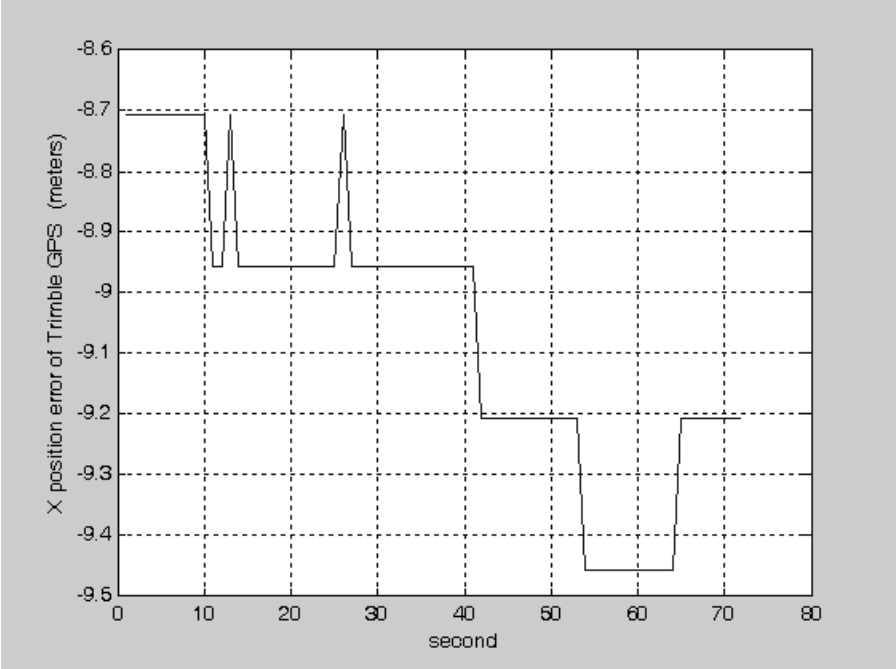

Figure 90. X position error of the Trimble GPS

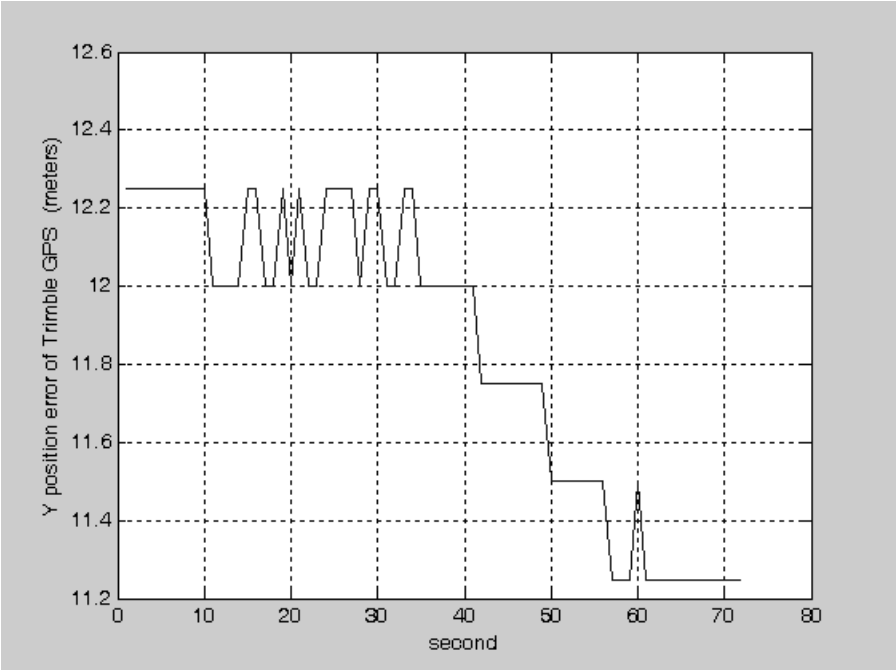

Figure 91. Y position error of the Trimble GPS

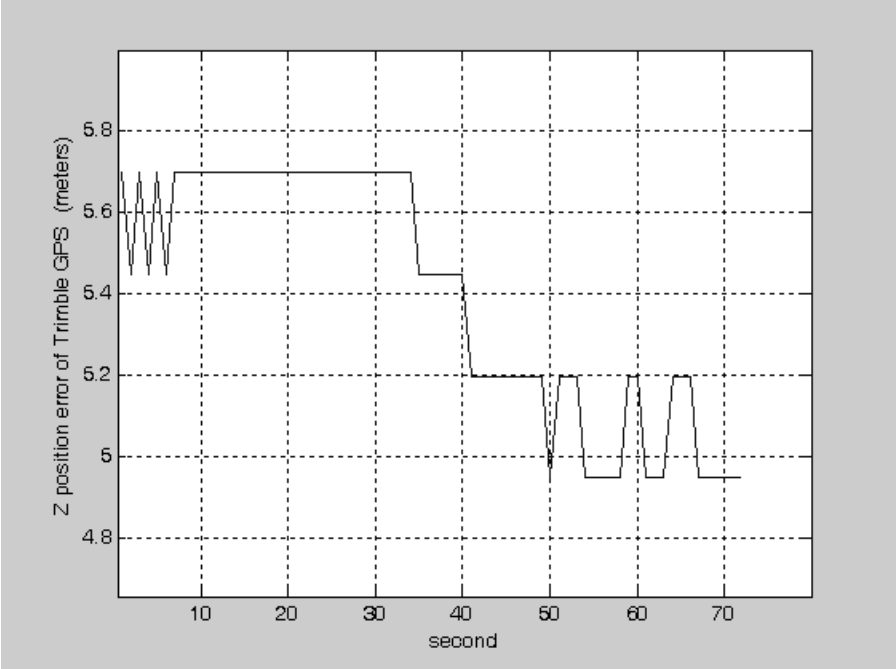

Figure 92. Z position error of the Trimble GPS

CHAPTER 6

CONCLUSION

In this chapter, we will make an overall assessment of the thesis work.

In this thesis, a software GPS receiver implementation is done in Matlab environment. Before the implementation, we have made detailed investigation as summarized below:

- GPS system structure including the system segments, signal construction, signal content and general information on GPS receivers are presented in detail.

- Digital signal processing module of the receiver, which consists of signal acquisition, signal tracking, bit and frame synchronization, is detailed. Different types of the signal processing are surveyed, which are the conventional and BAAS methods.

- Navigation algorithms, which include timing corrections, satellite position calculation and PVT (position-velocity-time) computation, are explained.

In signal processing section implementation, we use a pre-recorded raw GPS IF data. Since this data length is insufficient for a complete navigation process, the input of the navigation block is fed from the commercial GPS receivers, which are recorded in different environments. Therefore, we have implemented a software GPS receiver in two sections, which are signal processing section and navigation section.

1. Signal processing section: The purpose of the signal processing section is acquiring and tracking satellite signals in the received GPS signal. We have implemented two types of signal processing methods, which are the conventional and BAAS signal processing as explained below:

- Conventional signal processing is composed of conventional signal acquisition and conventional signal tracking. Conventional signal acquisition is actually a serial search. Serial search sequentially tests every possible combination of C/A code delay and Doppler frequency. When the signal is acquired, serial search is stopped and tracking is started. Conventional tracking is implemented by using the FLL (frequency locked loop) for carrier tracking and DLL (delay locked loop) for code tracking. FLL operates to lock the locally generated reference carrier frequency to the input GPS signal frequency. FLL discriminator outputs the frequency error of the local carrier signal. In the same way, DLL operates to lock the locally generated C/A code to the input GPS signal code. Inputs of the DLL discriminator are early and late code correlations. DLL discriminator outputs the delay between the reference code and the input GPS signal code.

- BAAS signal processing is composed of circular correlation acquisition and BAAS tracking. Signal acquisition uses circular convolution through use of the Fast Fourier transform (FFT) to determine the carrier frequency and code phase for each satellites. C/A code tracking is maintained by shifting the input signal, on a sample by sample basis. Using this synchronized code, one component of a discrete Fourier transform (DFT) is calculated in an early, late and prompt channel. The carrier frequency is tracked from the phase of the DFT in the prompt channel and the navigation data can be recovered.

Implemented signal processing is driven by the real GPS signal samples. Conventional signal acquisition method is a slow algorithm when it is compared with the circular correlation acquisition method. Moreover, circular correlation method outputs more accurate Doppler frequency output since it performs fine

frequency estimation. Frequency resolution of the circular correlation method can be chosen within a few Hertz. On the other hand, frequency resolution of the conventional method cannot be chosen as small as the circular correlation method since the computational complexity of this method is very high. BAAS tracking and conventional tracking performs almost identical. However, BAAS tracking has less computational complexity than the conventional tracking since BAAS tracking operates in frequency domain.

2. Navigation section: Navigation algorithm is used for PVT (position-velocity-time) computation. The position of the user can be found from measured distances from this point to some known satellite positions. The distance between the satellite and the user measurements called pseudorange is based on the transmission and receipt times of the signals. Pseudorange is biased due to offset between the satellite and receiver clocks. This receiver bias value is also an unknown like the position of the user. On the other hand, deltarange (calculated from Doppler frequency) and satellite average velocities (computed by differentiating the satellite positions) are necessary for user velocity determination and the corresponding receiver clock drift. Hence, unknowns of the receiver are the user position, receiver clock bias, user velocity and receiver clock drift. In order to find these unknowns satellite position computation is achieved first. Next, atmospheric delays and satellite clock errors are eliminated from pseudorange measurements. Finally, either Kalman filter or LSE method is used for PVT computation. Inputs of the PVT computation are the corrected pseudoranges, deltaranges, satellite positions and satellite average velocities.

In the navigation algorithms, PVT computation is achieved by using navigation data and raw GPS measurements of commercial GPS receivers. We have tested dynamic and stationary cases. In the dynamic case, commercial GPS receiver is placed on a vehicle. PVT outputs of the commercial GPS and the implemented navigation algorithm results are compared. It is observed that Kalman filter gives more reliable results than the LSE for a moving platform. Therefore, Kalman filter is preferred for a moving platform. In the stationary case, commercial GPS receiver is placed at a precisely known point. In such

case, Kalman filter and LSE give same results. Therefore, LSE is preferred for a stationary platform since it has less computational complexity. Since the position is known precisely, errors of the commercial GPS receiver and implemented navigation algorithm are found.

This study has provided several benefits as listed below:

- This study is a source to implement a software GPS receiver.
- Hardware GPS receiver can also be designed by using this study as a base.
- Any enhancements in GPS can be easily adapted to the implemented SGR since it is flexible enough.
- New generation GNSS (Global Navigation Satellite Systems) receivers can be implemented by the experience obtained from this study. Galileo is a new GNSS and it is being designed by the Europeans.
- Integrated systems such as GPS/INS (inertial navigation system) integration can be developed by using the implemented SGR as a tool.
- Different scenarios for integrated systems and GPS studies can be realized by using this implemented SGR as a tool.

There are available commercial software GPS receivers by several companies such as Accord Software & Systems, NordNav Technologies, etc. Among the commercial software GPS receivers, Matlab GPS signal simulation toolbox is the most similar one to our study since we have implemented a SGR in Matlab, too. GPS signal simulation toolbox includes functions to generate simulated SGR scenarios, model the SGR tracking and navigation capability and perform analysis on recorded SGR data sets. The toolbox costs $24000. Therefore, our SGR has also commercial value.

Future work on the implemented software GPS receiver can be arranged as follows:

- Bit and frame synchronization in signal processing module can be achieved after providing GPS IF signal long enough. Hence, PVT output can be obtained from the GPS IF signal.

- Real-time processing can be considered especially in different and faster software environments.

Consequently, implemented GPS software receiver sections, benefits provided by this study, the commercial importance and future work are mentioned. This is the first complete thesis work about software GPS receiver implementation in Turkey. Furthermore, this work leads us to have GPS receiver technology in our expose and this thesis can be a source for satellite navigation systems.

REFERENCES

[1] P. R. Kumar, Pravin Varaiya, *"Stochastic Systems-Estimation, Identification and Adaptive Control"*, Prentice-Hall, Inc., 1986

[2] Robert Grover Brown, Patrick Y.C. Hwang, "*Introduction to Random Signals and Applied Kalman Filtering"*, A Wiley Interscience Publication, 1992, Second Edition, Canada

[3] Elliott D. Kaplan, *"Understanding GPS Principles and Applications"*, Artech House, Inc., 1996

[4] Roger L.Peterson, Rodger E.Ziemer, David E. Borth, "*Introduction to Spread Spectrum Communications"*, Prentice-Hall,Inc. 1996

[5] Naseer A. Choudhry, "*Introduction to GPS, Seminar at METU*", March 18-22, 1996

[6] Bradford W. Parkinson, James J. Spilker Jr., *"Global Positioning System: Theory and Applications"*, Volume 1, American Institute of Aeronautics and Astronautics, Inc., 1996

[7] Scott I. Snyder, "Integration of Satellite Navigation Systems (GPS) with Inertial Navigation Systems", September 15-17, 1997

[8] James B. Y. Tsui, *"Fundamentals of Global Positioning System Receivers, a software approach"*, John Wiley & Sons, Inc., 2000

[9] Mohinder S. Grewal, Lawrence R. Weill, Angus P. Andrews, *"Global Positioning Systems, Inertial Navigation, and Integration"*, John Wiley & Sons,Inc., 2000

[10] *GPS Interface Control Document*, , ICD-GPS-200C-004,12.April.2000

[11] Dennis M. Akos, M. S. Braasch, *"Global Navigation Satellite System Software Radio Receiver Implementation"*, John Wiley & Sons, Inc., 2000

[12] Simon Haykin, *"Communication Systems"*, John Wiley & Sons, Inc., 2001

[13] *"The ION Red Book CD-ROM, Volumes 1-6, Global Positioning System-GPS and Its Augmentation Systems"*, The Institute of Navigation, Inc., 2002

[14] "*Harita Genel Komutanlığı Eğitimi Notları*", June 2003, TÜBİTAK-SAGE

[15] Dr-Ing. Bernd Eissfeller, *"GPS & GNSS Receiver Technology and Trends"*, Navtech Seminars&GPS Supply,Inc., ION GNSS 2004 Tutorial Course 710

[16] Dr. M. Elizabeth Cannon, *"High Accuracy GPS Positioning Techniques & Applications I"*, Navtech Seminars&GPS Supply,Inc., ION GNSS 2004 Tutorial Course 610A

[17] C. Ma, G. Lachapelle, M. E. Canon, *"Implementation of a Software GPS Receiver"*, ION GNSS Conference 2004

[18] Dennis M. Akos, *"Software Receiver Architectures & Design for GPS/GNSS"*, Navtech Seminars&GPS Supply,Inc., ION GNSS 2004 Tutorial Course 550

[19] James B. Y. Tsui, M. H. Stockmaster, Dennis M. Akos, *"Block Adjustment of Synchronizing Signal (BAAS) for GPS Receiver Signal Processing"*

[**20]** David Akopian and Jari Syrjarinne, "*A Network Aided Iterated LS Method for GPS Positioning and Time Recovery Without Navigation Message Decoding"*, IEEE, Research and Technology Access Nokia Mobile Phones.

[**21]** M. S. Braasch, A. J. Van Dierendonck, *"GPS Receivers Architectures and Measurements"*, Invited Paper

[**22]** J. Mell, "*Spread Spectrum (introduction)*", Sirius Communication, Rotselaar, Belgium

[23] NATO Publication, *"Navstar GPS User Equipment"*, Navtech Seminars & Navtech Book and Software Store, Inc., from U.S. Government document

[24] http://www.sss-mag.com/ss.html

[25] Kent Krumvieda, DFC; Premal Madhani, CCAR; Chad Cloman, DFC; Eric Olson, DFC; Dr. John Thomas, DFC; Dr. Penina Axelrad, CCAR; Dr. Wolfgang Kober, DFC, *"A Complete IF Software GPS Receiver: A Tutorial about the Details"*

[26] http://www.navsys.com/Products/sgr.html

[27] Ravindra Babu, *"Web Based Resources on Software GPS Receivers"*, The University of New South Wales, Sydney, Australia