AN INVESTIGATION INTO THE APPLICATION
OF
BLOCK PROCESSING TECHNIQUES
FOR THE
GLOBAL POSITIONING SYSTEM

A Dissertation Presented to
The Faculty of the
Fritz J. and Dolores H. Russ
College of Engineering and Technology
Ohio University

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

by
Maarten Uijt de Haag

August, 1999
The work in this dissertation would not have been possible without the help and support of the many people I have worked with over the last couple of years. In particular I would like to acknowledge those people who have directly supported my research.

First, I would like to thank my advisor, Dr. Frank van Graas. Frank did not only give me excellent guidance throughout this research, but also provided me with an enormous insight in the magical world of the Global Positioning System. His efforts to stimulate one’s use of creativity, his infinite patience, and his sense of humor has inspired both my academic and personal life.

To the members of my dissertation committee who provided me with a thorough and challenging review, I offer my sincere thanks: in particular to Dr. Michael S. Braasch, Dr. Mehmet Celenk, and Dr. Jeffrey C. Dill of the School of Electrical Engineering and Computer Science, and to Dr. Roger W. Rollins of the Physics Department.

I would also like to thank Christopher Snyder and Jonathan Sayre of the Ohio University Avionics Engineering Center technical staff for their support during various data collections. Dr. Dan Aloi and Dr. David W. Diggle are thanked for their motivating words throughout the whole process and even though all students at the Avionics Engineering Center have always been a source of inspiration, I especially want to thank Gang Feng for the many stimulating discussions about block processors which served as a basis for this work.
Finally, I would like to thank my parents, Jos and Yvonne, and my sister, Heleen, for always being there for me. They have supported me in every step I took, even when that meant living thousands of miles apart.
TABLE OF CONTENTS

LIST OF TABLES ........................................................................................................ vi
LIST OF ILLUSTRATIONS ...................................................................................... vii
LIST OF ACRONYMS .............................................................................................. xii
GENERAL NOTATION .............................................................................................. xv

1. INTRODUCTION ................................................................................................. 1

2. THE GLOBAL POSITIONING SYSTEM ............................................................... 5
   2.1 Background ................................................................................................... 5
   2.2 GPS Observables ......................................................................................... 11
   2.3 GPS Model and Error Sources ................................................................... 13

3. GPS SEQUENTIAL PROCESSOR ...................................................................... 16
   3.1 Background .................................................................................................. 16
   3.2 Tracking Functions ..................................................................................... 18
       3.2.1 Carrier Tracking .................................................................................. 25
       3.2.2 Code-Phase Tracking ......................................................................... 32
   3.3 Signal Acquisition ....................................................................................... 35
   3.4 Advantages and Disadvantages of Sequential Processing of GPS .......... 38

4. TRANSMITTER/RECEIVER DYNAMICS ............................................................ 40
   4.1 Background .................................................................................................. 40
   4.2 The Doppler Effect ...................................................................................... 40

5. OSCILLATOR EFFECTS ...................................................................................... 44
   5.1 Background .................................................................................................. 44
   5.2 Oscillator Measures .................................................................................... 45
   5.4 A/D Conversion and the Sample Doppler Effect ........................................ 56
       5.4.1 Pre-sampling Local Oscillator .............................................................. 57
       5.4.2 Analog-to-Digital Conversion ............................................................... 58
       5.4.3 Post-sampling Local Oscillator ............................................................. 62

6. RADIO FREQUENCY INTERFERENCE ............................................................... 63
   6.1 Background .................................................................................................. 63
   6.2 Interference Suppression and Detection ..................................................... 67
   6.3 Transform and Joint Time-Frequency Techniques ................................ .... 69
7. BLOCK PROCESSING TECHNIQUES ................................................. 74
   7.1 Background ............................................................... 74
   7.3 Frequency-Domain Techniques ......................................... 78
   7.4 Time-Frequency Domain techniques .................................... 83
   7.5 Summary ................................................................. 89

8. GPS BLOCK PROCESSOR - CASE STUDIES ................................. 92
   8.1 Background ............................................................... 92
   8.1 Case I: Detection of GPS Signals at Low CN0 ....................... 94
   8.2 Case II: High Dynamic Tracking ....................................... 118
   8.3 Case III: Interference Suppression .................................... 131

9. SUMMARY, CONCLUSIONS AND RECOMMENDATIONS ....................... 144

10. REFERENCES ................................................................. 147

11. APPENDICES ........................................................................ 158
   Appendix A. GPS Software Signal Simulator .............................. 158
   Appendix B. The Effect of the Sample Rate Offset in Bandpass Sampling ... 170

ABSTRACT ........................................................................... 174
LIST OF TABLES

Table 3.1   DLL Discriminator Functions ............................................. 34
Table 5.1   Time and Frequency Measures of the Frequency and Phase .......... 51
LIST OF ILLUSTRATIONS

Figure 2.1 Two-Dimensional Position Determination without Uncertainty .......... 6
Figure 2.2 Two-Dimensional Position Determination with Uncertainty .......... 7
Figure 2.3 C/A Code Autocorrelation Function ........................................... 10
Figure 2.4 Portion of the C/A Code Line Spectrum ........................................ 10
Figure 2.5 Signal Power Spectral Density (PSD) ............................................ 11
Figure 2.6 Propagation Time Determination .................................................. 12
Figure 2.7 The GPS Measurement Process .................................................... 13
Figure 3.1 Digital GPS Receiver Block Diagram ........................................... 17
Figure 3.2 General Tracking Loop Model .................................................... 19
Figure 3.3 Phase Plane Portrait for a Second Order Phase Locked Loop .......... 22
Figure 3.4 Linear Mode Tracking Loop Model (Laplace Transformed) ............ 23
Figure 3.5 Down-Conversion and Correlation (Matched Filtering) ................. 24
Figure 3.6 Carrier Tracking Loop Block Diagram ........................................ 27
Figure 3.7 PLL Phase Error ($B_n = 5$ Hz, frequency offset: $12$ Hz) ............ 29
Figure 3.8 Estimated Phase-Plane Portrait from FLL and PLL Outputs .......... 29
Figure 3.9 FLL Frequency Error (FLL-PLL Implementation) .......................... 30
Figure 3.10 PLL Phase Error (FLL-PLL Implementation) ............................... 30
Figure 3.11 Occurrence of Cycle Slips During Pull-In .................................. 31
Figure 3.12 Code Tracking Loop Block Diagram .......................................... 33
Figure 3.13 Two-Dimensional Acquisition Search Grid .................................... 37
Figure 4.1  Transmitted C/A Code Line Spectrum ............................... 43
Figure 4.2  Received C/A Code Line Spectrum ............................... 43
Figure 5.1  Phase Noise Characteristic in Time-Domain ...................... 47
Figure 5.2  Phase Noise Characteristic in Time-Domain for the Allan Variance . 49
Figure 5.3  Phase Noise for Typical Oscillators ................................. 52
Figure 5.4  General Continuous-Time Oscillator Model ......................... 54
Figure 5.5  Bandpass-Sampling Software Radio Setup .......................... 57
Figure 5.6  Effect of a Sample Rate Offset ..................................... 61
Figure 6.1  Time-Frequency Representation of WBI ............................. 65
Figure 6.2  Time-Frequency Representation of NBI ............................. 65
Figure 6.3  Time-Frequency Representation of WBPI ........................... 66
Figure 6.4  Time-Frequency Representation of NBPI ........................... 66
Figure 6.5  Time/Frequency Excision Scheme Number 1 ....................... 70
Figure 6.6  Time/Frequency Excision Scheme Number 2 ....................... 70
Figure 6.7  Time/Frequency Excision Scheme Number 3 ....................... 70
Figure 6.8  Interference Suppression using the STFT .......................... 72
Figure 6.9  Interference Suppression using the Wavelet Transform .......... 72
Figure 7.1  Division of Time-series in Blocks ................................ 77
Figure 7.2  Block Addition Techniques ......................................... 77
Figure 7.3  Block Inspection Techniques ....................................... 79
Figure 7.4  Time vs. Frequency Tradeoff in the FFT .......................... 80
Figure 7.5  Transform Domain Correlator ..................................... 82
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.6</td>
<td>Massive Correlator Bank</td>
<td>82</td>
</tr>
<tr>
<td>7.7</td>
<td>Short-Time Fourier Transform Time-Frequency Tiling</td>
<td>86</td>
</tr>
<tr>
<td>7.8</td>
<td>Discrete Wavelet Transform Time-Frequency Tiling</td>
<td>86</td>
</tr>
<tr>
<td>7.9</td>
<td>Heisenberg Box</td>
<td>88</td>
</tr>
<tr>
<td>7.10</td>
<td>Filter Bank Analysis Element</td>
<td>90</td>
</tr>
<tr>
<td>7.11</td>
<td>Filter Bank Synthesis Element</td>
<td>90</td>
</tr>
<tr>
<td>8.1</td>
<td>Software Receiver Front End test Setup</td>
<td>93</td>
</tr>
<tr>
<td>8.2</td>
<td>Basic Element (BE) of the GPS Block Processor</td>
<td>97</td>
</tr>
<tr>
<td>8.3</td>
<td>Block Adder (BA) Element of the GPS Block Processor</td>
<td>99</td>
</tr>
<tr>
<td>8.4</td>
<td>Fading due to the Residual Carrier Frequency</td>
<td>103</td>
</tr>
<tr>
<td>8.5</td>
<td>Block Addition in the Absence of Code Doppler</td>
<td>105</td>
</tr>
<tr>
<td>8.6</td>
<td>Block Addition in the Presence of Code Doppler</td>
<td>105</td>
</tr>
<tr>
<td>8.7</td>
<td>Illustration of the Code Doppler Effect</td>
<td>106</td>
</tr>
<tr>
<td>8.8</td>
<td>TLM and HOW Word Structures</td>
<td>108</td>
</tr>
<tr>
<td>8.9</td>
<td>Logic Implementation for One Carrier Frequency</td>
<td>110</td>
</tr>
<tr>
<td>8.10</td>
<td>Navigation Correlation Characteristic</td>
<td>112</td>
</tr>
<tr>
<td>8.11</td>
<td>Navigation Correlation Characteristic</td>
<td>112</td>
</tr>
<tr>
<td>8.12</td>
<td>C/A Code Cross-Correlation</td>
<td>115</td>
</tr>
<tr>
<td>8.13</td>
<td>C/A Code Cross-Correlation</td>
<td>115</td>
</tr>
<tr>
<td>8.14</td>
<td>Navigation Cross-Correlation</td>
<td>116</td>
</tr>
<tr>
<td>8.15</td>
<td>C/A Code Cross-Correlation</td>
<td>116</td>
</tr>
<tr>
<td>8.16</td>
<td>Navigation Cross-Correlation</td>
<td>117</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>8.17</td>
<td>C/A Code Cross-Correlation</td>
<td>117</td>
</tr>
<tr>
<td>8.18</td>
<td>GPS Time-Frequency Analysis Block Diagram</td>
<td>119</td>
</tr>
<tr>
<td>8.19</td>
<td>Dynamic Tracking Zoom Function (Coarse and Fine)</td>
<td>120</td>
</tr>
<tr>
<td>8.20</td>
<td>Acquisition Grid for Satellite 7</td>
<td>123</td>
</tr>
<tr>
<td>8.21</td>
<td>Zoomed In Acquisition Grid</td>
<td>123</td>
</tr>
<tr>
<td>8.22</td>
<td>Real Cross-Correlation Peak Precession over 1 Second</td>
<td>124</td>
</tr>
<tr>
<td>8.23</td>
<td>Imaginary Cross-Correlation Peak Precession over 1 Second</td>
<td>124</td>
</tr>
<tr>
<td>8.24</td>
<td>Peak Amplitude Variation as a Function of Time</td>
<td>125</td>
</tr>
<tr>
<td>8.25</td>
<td>Code-Phase Variation as a Function of Time</td>
<td>126</td>
</tr>
<tr>
<td>8.26</td>
<td>Time -Frequency Tiling for Satellite 7</td>
<td>128</td>
</tr>
<tr>
<td>8.27</td>
<td>T-F Tiling for Real Squared Cross-Correlation Peak Variation</td>
<td>128</td>
</tr>
<tr>
<td>8.28</td>
<td>Ridge Algorithm Results for Real Correlation</td>
<td>129</td>
</tr>
<tr>
<td>8.29</td>
<td>Ridge Algorithm Results for Real Squared Correlation</td>
<td>129</td>
</tr>
<tr>
<td>8.30</td>
<td>T-F Tiling for a GPS Signal with 400 Hz/s Doppler Change</td>
<td>130</td>
</tr>
<tr>
<td>8.31</td>
<td>T-F Tiling of a Squared GPS Signal with 400 Hz/s Doppler</td>
<td>130</td>
</tr>
<tr>
<td>8.32</td>
<td>Acquisition Grid for Satellite 17 without NBI</td>
<td>133</td>
</tr>
<tr>
<td>8.33</td>
<td>Cross-Correlation Peak for Satellite 17 without NBI</td>
<td>134</td>
</tr>
<tr>
<td>8.34</td>
<td>Acquisition Grid for Satellite 17 with NBI ( 2 Hz )</td>
<td>135</td>
</tr>
<tr>
<td>8.35</td>
<td>Fast Fourier Transform of Block 1 of Satellite 17 with NBI</td>
<td>136</td>
</tr>
<tr>
<td>8.36</td>
<td>FFT of Block 1 after Nulling the NBI Source</td>
<td>136</td>
</tr>
<tr>
<td>8.37</td>
<td>Acquisition Grid for Satellite 17 after NBI Excision</td>
<td>137</td>
</tr>
<tr>
<td>8.38</td>
<td>Cross-Correlation Peak of Satellite 17 after NBI Excision</td>
<td>137</td>
</tr>
</tbody>
</table>
Figure 8.39  Error between Interference Free and NBI  

Figure 8.40  Same as Figure 8.39 but now with NBI at 138kHz  

Figure 8.41  ADC Output in the Presence of Pulsed Interference  

Figure 8.42  Acquisition Grid in the Presence of Chirp-like Pulse Interference  

Figure 8.43  T-F Behavior of Signal plus Chirp-like Pulsed Interference  

Figure 8.44  ADC Output after Excision Operation using the DWT  

Figure 8.45  Residual Error after Pulsed Interference Excision using the DWT  

Figure 8.46  Acquisition Grid after Interference Excision using the DWT  

Figure A.3  Software Signal Simulator: PSD of s(t)  

Figure A.4  PSD of the Signal after Bandpass Filtering (x_f(t))  

Figure A.5  PSD of the Signal after Down-conversion (x_2(t))  

Figure A.6  PSD of the Signal after Lowpass Filtering (x_3(t))  

Figure A.7  PSD of the Signal after Decimation (x_4(t))  

Figure A.8  PSD Signal after Amplitude Quantization (x_5(t))  

Figure A.9  Signal PSD Comparison for Sample Rates of 40 Msp and 5 Msp  

Figure A.10  PSD of Output of Signal plus Noise  

Figure A.11  Output of the Quantizer  

Figure B.1  Bandpass Sampling  

Figure B.2  Comparison of Bandpass and Regular Sampling
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADC</td>
<td>Analog-to-Digital Converter</td>
</tr>
<tr>
<td>ASIC</td>
<td>Application Specific Integrated Circuit</td>
</tr>
<tr>
<td>BPF</td>
<td>Bandpass Filter</td>
</tr>
<tr>
<td>BPSK</td>
<td>Binary Phase Shift Keying</td>
</tr>
<tr>
<td>C/A</td>
<td>Coarse/Acquisition</td>
</tr>
<tr>
<td>CDMA</td>
<td>Code Division Multiple Access</td>
</tr>
<tr>
<td>$C/N_0$</td>
<td>Carrier-to-Noise Ratio</td>
</tr>
<tr>
<td>CPU</td>
<td>Central Processing Unit</td>
</tr>
<tr>
<td>CW</td>
<td>Continuous Wave</td>
</tr>
<tr>
<td>DFT</td>
<td>Discrete Fourier Transform</td>
</tr>
<tr>
<td>DMF</td>
<td>Digital Matched Filter</td>
</tr>
<tr>
<td>DWT</td>
<td>Discrete Wavelet Transform</td>
</tr>
<tr>
<td>FFT</td>
<td>Fast Fourier Transform</td>
</tr>
<tr>
<td>FPGA</td>
<td>Field Programmable Gate Array</td>
</tr>
<tr>
<td>FTF</td>
<td>Fundamental Time Frame</td>
</tr>
<tr>
<td>GNSS</td>
<td>Global Navigation Satellite System</td>
</tr>
<tr>
<td>GPS</td>
<td>Global Positioning System</td>
</tr>
<tr>
<td>ID</td>
<td>Integrated Doppler</td>
</tr>
<tr>
<td>IF</td>
<td>Intermediate Frequency</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Definition</td>
</tr>
<tr>
<td>--------------</td>
<td>------------</td>
</tr>
<tr>
<td>ISI</td>
<td>Inter Symbol Interference</td>
</tr>
<tr>
<td>LPF</td>
<td>Lowpass Filter</td>
</tr>
<tr>
<td>NBI</td>
<td>Narrow-Band Interference</td>
</tr>
<tr>
<td>NBPI</td>
<td>Narrow-Band Pulsed Interference</td>
</tr>
<tr>
<td>NEB</td>
<td>Noise Equivalent Bandwidth</td>
</tr>
<tr>
<td>OCXO</td>
<td>Oven Controlled Oscillator</td>
</tr>
<tr>
<td>PR</td>
<td>Pseudorange</td>
</tr>
<tr>
<td>PRN</td>
<td>Pseudorandom Noise</td>
</tr>
<tr>
<td>PSD</td>
<td>Power Spectral Density</td>
</tr>
<tr>
<td>RF</td>
<td>Radio Frequency</td>
</tr>
<tr>
<td>RFI</td>
<td>Radio Frequency Interference</td>
</tr>
<tr>
<td>rms</td>
<td>Root Mean Square</td>
</tr>
<tr>
<td>RX</td>
<td>Receiver</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal-to-Noise Ratio</td>
</tr>
<tr>
<td>STFT</td>
<td>Short-Time Fourier Transform</td>
</tr>
<tr>
<td>SV</td>
<td>Space vehicle (satellite)</td>
</tr>
<tr>
<td>TCXO</td>
<td>Temperature Compensated Oscillator</td>
</tr>
<tr>
<td>TOR</td>
<td>Time-of-Reception</td>
</tr>
<tr>
<td>TOT</td>
<td>Time-of-Transmission</td>
</tr>
<tr>
<td>TX</td>
<td>Transmitter</td>
</tr>
<tr>
<td>UTC</td>
<td>Universal Time, Coordinated</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>VCO</td>
<td>Voltage Controlled Oscillator</td>
</tr>
<tr>
<td>WBI</td>
<td>Wide-band Interference</td>
</tr>
<tr>
<td>WBPI</td>
<td>Wide-band Pulsed Interference</td>
</tr>
</tbody>
</table>
GENERAL NOTATION

General Notation Issues:

- $x$ a scalar
- $x^i$ a scalar to the power $i$
- $x(t)$ a scalar function of time
- $x[n]$ a discrete-time scalar function
- $\bar{x}$ the average value of an ensemble
- $X$ a scalar
- $X(t)$ a scalar function of time
- $\mathbf{x}$ a vector (dimension specified in the text)
- $x_i$ the $i^{th}$ element of the vector
- $x_i$ a scalar specified by capital ‘I’ (for example $x_{RX}$)
- $\mathbf{x}(t)$ a vector function of time (dimension specified in the text)
- $\mathbf{X}$ a matrix
- $x_{ij}$ the element of the matrix at row $i$ and column $j$
- $C_{jk}$ a transformation matrix from the $j$-frame to the $k$-frame

Transforms:

- $\mathcal{F}(\cdot)$ is the Continuous-Time Fourier Transform of $(\cdot)$
- $\mathcal{L}(\cdot)$ is the Laplace Transform of $(\cdot)$
- $DFT(\cdot)$ is the Discrete Fourier Transform of $(\cdot)$
\[ DWT(\cdot) \quad \text{is the Discrete Wavelet Transform of } (\cdot) \]

\[ FFT(\cdot) \quad \text{is the Fast Fourier Transform of } (\cdot) \]

\[ IFFT(\cdot) \quad \text{is the Inverse Fast Fourier Transform of } (\cdot) \]

\[ STFT(\cdot) \quad \text{is the Short-Time Fourier Transform of } (\cdot) \]

**Operators:**

\[ E\{\cdot\} \quad \text{is the expected value of its argument} \]

\[ \langle \cdot \rangle \quad \text{is the infinite time-average of its argument} \]
1. INTRODUCTION

The Global Positioning System, or GPS, has enabled a wealth of possible applications and research opportunities, both driven by GPS' global three-dimensional position, velocity and timing services. Originally intended as a U.S. government Department of Defense (DoD) system, GPS' civilian market has shown a steady growth due to applications in areas such as aircraft navigation and attitude determination systems, surveying, and person and vehicle positioning on land, water, and in space.

GPS performance is highly dependent on the user's environment. Use of conventional GPS receivers in high radio frequency (RF) interference environments, urban canyons, or inside buildings may leave users with unreliable navigation service. That user may experience degraded performance, or even loss-of-signal and thus navigation capability. Traditionally, integration with other navigational aids, such as Inertial Measurement Units (IMU's), wheel sensors, and maps was pursued to provide user positioning during GPS signal outages. Although integration with other sensors results in a better reliability, higher availability, and better integrity than the individual systems, another approach would be to improve the GPS sensor performance by changing the receiver processing methods.

Currently, digital GPS receivers are based on sequential processing; the signal is processed on a sample by sample basis. The basic elements of a sequential receiver, the phase-locked loop (PLL), the frequency-locked loop (FLL), the delay-locked loop (DLL), and the correlator have been used in communication systems for decades. Consequently, dedicated analog and digital components as well as application-specific integrated circuits
ASICs), are commercially available at relatively low-cost. It is not only the cost, but also the inherent loss of information, the non-linearity of the tracking loops, and the noise versus tracking error tradeoff that make it worthwhile to investigate alternatives to the sequential receiver implementation.

The migration from fully hardware to highly software-based receivers, has made it possible to investigate alternatives to sequential processing. The software radio concept applied to global navigation satellite systems (GNSS) was addressed in (Akos, 1997). The proposed GNSS software radio uses bandpass sampling (Vaughan, Scott, & White, 1991) in order to minimize the number of RF components. Once the measurements are available in a software receiver, one logical alternative to sequential processing is the application of digital signal processing techniques to blocks of data. These techniques are referred to as block processing techniques.

Block processing techniques have been extensively studied and used for applications in the fields of communications, multimedia systems, and computer vision systems (Chellappa, et al., 1998). Examples are the implementation of modulation, coding, synchronization, and interference rejection schemes. However, fundamental differences exist between the application of block processing techniques to satellite navigation systems and the application in communication systems. For a navigation system, it is the temporal location of signal features that is of importance, whereas a communication system’s main goal is the optimal extraction of data.

The organization of this dissertation is as follows. After discussion of the GPS navigation principles and signal characteristics in Chapter 2, conventional sequential
receivers will be addressed in Chapter 3. This dissertation focuses on the observation of code phase, carrier phase, and carrier frequency. The behavior of these parameters is a function of time, and requires insight in the main mechanisms behind this relationship. Chapter 4 addresses the effects of relative user-satellite motion onto the carrier frequency and code chipping rate. Phase and frequency variations attributed to the local oscillators are discussed in Chapter 5. Chapter 6 deals with radio frequency interference (RFI). A classification of RFI will be given based on its time-frequency behavior, and both detection and suppression techniques will be discussed. An overview of existing block processing techniques will be provided in Chapter 7. Application of these techniques to GPS, will also be addressed and illustrated.

To illustrate the effectiveness of block processing, three case studies were performed; tracking of GPS at very low signal-to-noise ratios, sole use of GPS under high dynamics, and narrow-band radio-frequency interference suppression. The first part of Chapter 8 addresses tracking of GPS in environments where the signal-to-noise ratio is very low. This scenario is not unusual if one considers tracking of GPS inside buildings, or in an environment with significant wide-band interference. GPS signals inside buildings, for example, are typically 30 dB weaker than the signals received outside the building. Tracking of GPS under high dynamics usually requires the aid of other navigation sensors, such as IMUs, because the receiver tracking loops are unable to maintain frequency lock. This problem of observability of large changes in frequency may be resolved by examining the time-frequency behavior of the received signal. Lastly, time-frequency techniques for interference suppression will be discussed as applied to GPS.
The computational intensity of block processing techniques should no longer be considered a drawback, due to the availability of high-speed processing elements. Field-Programmable Gate Arrays (FPGAs) (Goslin, 1995), for example, may be able to perform complex block-processing algorithms in the very near future. Block processing techniques can enable a user to track GPS, and thus provide user positioning, under circumstances in which a conventional sequential receiver would not work.
2. THE GLOBAL POSITIONING SYSTEM

2.1 Background

The Global Positioning System (GPS) is a radio-navigation aid that provides a world-wide three-dimensional (3-D) position, velocity, as well as precise timing capability. GPS is based on tri-lateration; the user's position in a 3-D reference frame is determined by measuring the range with respect to at least three satellites. The user will then be located at the intersection of the three spheres whose radii and origins are given by the ranges and the positions of the satellites, respectively. This concept is illustrated in Figure 2.1 for the two-dimensional (2-D) case. Knowledge of both the range to and the position of a transmitter in 2-D space, for example transmitter number 1, limits the user's position to a point on a circle with radius $r_1$. Adding another range measurement, $r_2$, the user must be located on the intersection of two circles given by locations A and B. Addition of a third measurement allows for the removal of this ambiguity, see Figure 2.1, and pinpoints the user's position to be A. In 3-D space the circles are replaced by spheres. In that case a minimum of 3 spheres or 3 range measurements would be required to limit the number of possible locations to two or less. The remaining ambiguity is generally not resolved by adding another measurement, but by the realization that one of the possible locations is in the vicinity of the earth surface, whereas the other is not. In GPS, the collection of transmitters is referred to as the space segment and consists of 24 satellites in six orbital planes.
Figure 2.1 Two-Dimensional Position Determination without Uncertainty

The range to a satellite is determined by the measured propagation time of the GPS signal from the satellite to the user times the speed of light in vacuum. Deviation of the user clock from the satellite clocks introduces an error in the determination of a correct propagation time of the signal transmitted by the satellite. Due to this clock offset and the presence of other sources of error, the range measurement made by the user is commonly referred to as a pseudorange measurement rather than a range measurement. The effect of the pseudorange errors on the position estimate is illustrated in Figure 2.2, where $e_1$, $e_2$, and $e_3$ are the uncertainties in the three ranges, respectively. The pseudorange errors introduce uncertainty in the position estimate in the sense that the three circles no longer intersect in one point; instead there is now a region of uncertainty given by the dark-colored area. Furthermore, the clock uncertainty adds a fourth unknown to the problem and, therefore, a minimum of four satellites needs to be tracked by the user to obtain a valid position.
Tetra-lateration would therefore be a more appropriate description of the basic principle of operation \((x, y, z; t)\).

The propagation time is computed as the difference between the measured time-of-transmission and time-of-reception of the GPS signal. The GPS receiver extracts the time-of-reception from its local oscillator and the time-of-transmission from the modulation on the carrier of the incoming signal. This process will be discussed in detail in Section 2.2. The GPS signal structure is defined in the GPS Interface Control Document (ICD-GPS-200, 1991) and described in (Kaplan, 1996 and Parkinson & Spilker, 1996):

\[
\begin{align*}
S_{L_1}(t) &= A_1 P_1(t) D(t) \sin(\omega_{L_1} t) + \sqrt{2} A_1 G_1(t) D(t) \cos(\omega_{L_1} t) \\
S_{L_2}(t) &= A_2 P_2(t) D(t) \sin(\omega_{L_2} t)
\end{align*}
\]

where \(\omega_{L_1}\) is the L1 carrier angular frequency \((\omega_{L_1} = (2\pi) \cdot 1.575.42 \cdot 10^6 \text{ rad/s})\)
\( \omega_{L2} \) is the L2 carrier angular frequency (\( \omega_{L2} = (2\pi) \cdot 1.227.60 \cdot 10^6 \text{ rad/s} \))

\( P_i(t) \) is the P spreading code for the \( i^{th} \) satellite (SV)

\( G_i(t) \) is the C/A spreading code for the \( i^{th} \) SV

\( D(t) \) is the SV navigation data for the \( i^{th} \) SV

\( A_1, A_2 \) are the signal amplitudes

Though not reflected in Equation 2.1, the navigation data and the spreading (Gold) codes are Binary Phase Shift Keying (BPSK) modulated onto the carrier. The navigation data is modulated onto the carrier at a 50 bps rate, or with data bit period, \( T_d = 0.02s \). The Gold code is a pseudo random noise (PRN) code that repeats every 1ms (\( T_G = 1ms \)), and consists of 1,023 chips (chipping rate of \( R_c = 1.023 \cdot 10^6 \)). All GPS satellites share the same carrier frequency by means of Code Division Multiple Access (CDMA) (Dixon, 1994 and Kaplan, 1996). In GPS-CDMA each satellite has its unique spreading or Gold codes \( G_i \) and \( P_i \).

Because the availability of \( P_i \) is not guaranteed for civilian use (ICD-GPS-200, 1991), the research in this dissertation is focused on C/A code and carrier measurements. The portion of the transmitted signal used in the scope of this dissertation is thus:

\[
s_i(t) = A G_i(t) D(t) \cos(\omega_{L2} t)
\]  

(2.2)

The power spectral density (PSD) function of \( s_i(t) \) can be found by taking the Fourier transform of the autocorrelation function of \( s_i(t) \), \( R_{s_i}(\tau) \), according to the Wiener-Khinchine theorem (Ziemer, 1995). The signal in Equation 2.2, \( s_i(t) \), is a periodic signal, and therefore, its autocorrelation function is given by the expression for an autocorrelation function of a
The deterministic power signal as given in (Ziemer, 1995). Its period is given by the period of the C/A code, \( T_c = 1 \text{ ms} \). Because \( T_G << T_c \), the effect of the navigation data will be neglected for the estimate of the power spectral density function \( D(t)D(t+\tau)=1 \). \( R_s(t) \) can now be related to the autocorrelation function, \( R_{G_i}(\tau) \), of the spreading code, \( G_i \):

\[
R_s(t) = \frac{A^2}{T_c} \int_{-T_c}^{T_c} G_i(t)G_i(t+\tau)\sin(\omega_{L_i}t)\sin(\omega_{L_i}t+\omega_{L_i}\tau)dt \\
= \frac{A^2}{T_c} \int_{-T_c}^{T_c} G_i(t)G_i(t+\tau)[\sin(\omega_{L_i}\tau)+\sin(2\omega_{L_i}t+\omega_{L_i}\tau)]dt \\
= A^2 \sin(\omega_{L_i}\tau)R_{G_i}(\tau)
\]

and the PSD of \( s(t) \) is thus:

\[
S_s(f) = \mathcal{F}\{R_s(\tau)\} \\
= \frac{A^2}{2j} S_{G_i}(f) \ast \left[ \delta(f-f_{L_i}) - \delta(f+f_{L_i}) \right]
\]

where \( S_{G_i}(f) = \mathcal{F}\{R_{G_i}(\tau)\} \).

Equation 2.4 relates the PSD of the signal to the PSD of the spreading code autocorrelation function. The autocorrelation function of the periodic Gold code, \( G_i \), is illustrated in Figure 2.3. The PSD of this Gold code is the line spectrum given in Figure 2.4.
Figure 2.3 C/A Code Autocorrelation Function

Figure 2.4 Portion of the C/A Code Line Spectrum

(Kaplan, 1996) with a squared 'sinc' function as an envelope. The C/A lines in Figure 2.4 represent the ratio between the power in the spectral line and the total power in the signal spectrum, $P_s$. The signal PSD is shown in Figure 2.5. It is important to note that the signal line-spectrum is based on infinite time and periodicity.
2.2 GPS Observables

In general, the GPS observables are referred to as the pseudorange (PR), integrated Doppler (ID), and time output by the receiver. However, these quantities are just the result of the true observables, or natural measurements of a GPS receiver (Ward, 1995); the time-of-transmission (TOT) and the time-of-reception (TOR) of the GPS signal. TOR may be computed based on the output of the carrier-tracking loop or the output of the code-tracking loop. In the scope of this work, it is these quantities that are considered the observables. Pseudorange and Integrated Doppler may be computed from the TOT and TOR.

Time-of-transmission and time-of-reception measurements are determined as follows. At specific time epochs (see Figure 2.6), which are referred to as the fundamental time frames (FTF’s) in this dissertation, a receiver determines the time-of-transmission of the current point in the received GPS signal time-series by means of multi resolution (Kaplan 1996); a low resolution but unambiguous time estimate is determined by the Z-count, whereas a high resolution but ambiguous time estimate is determined by the accumulated code chips (basis for the pseudorange measurement) or carrier cycles (basis for the integrated
Doppler measurement). The Z-count is a counter with a 1.5-second resolution that has been modulated onto the GPS signal as a part of the navigation message, $D(t)$. This counter is reset every Sunday at 0:00 hrs GPS time.

![Figure 2.6 Propagation Time Determination](image)

Under error-free conditions the resolution of the measurements would be driven by the accuracy with which the accumulated code chips and carrier cycles can be determined. The GPS measurement process can thus be summarized by the block diagram in Figure 2.7.

In a conventional receiver, code chips and carrier cycles are accumulated after synchronization of internal code and carrier copies with the incoming signal. The elements used for synchronization are code and carrier tracking loops, respectively (Best, 1997).
Figure 2.7 The GPS Measurement Process

When using tracking loops it is necessary to start with good estimates of the carrier frequency (within ±500 Hz) and code offset (within ±1 chip) to operate within the tracking loops’ pull-in ranges (Akos, 1997 and Best, 1997). Obtaining these estimates is referred to as the acquisition process. Chapter 3 will present these sequential techniques in greater detail, and will address the short-comings of sequential processing techniques.

2.3 GPS Model and Error Sources

The measured GPS signal propagation time deviates from the actual propagation time due to a variety of error sources. Deviation of the satellite and receiver clocks from
Universal Time, Coordinated (UTC) corrupts the times of transmission and reception of the GPS signal directly. The GPS signal experiences delays as it propagates through the atmosphere. Further, multipath, Selective Availability (SA), and hardware effects may delay or advance the estimated time-of-transmission measurement (Kaplan, 1996), and thermal noise adds TOT and TOR jitter. Due to the emphasis on the application of digital signal processing techniques to extract the natural measurements, an additive description of the total error budgets as described in (Kaplan, 1996) would not be useful, instead a signal description would be more appropriate. To limit the scope of this dissertation, effects due to propagation through the atmosphere, multipath, hardware effects, and Selective Availability are not included in the signal model used. The following mathematical expression will be referred to in further analysis:

\[ z_i(t) = A_i(t) G_i(t-t_i) D(t) \cos[\omega_c(t-t_i) + \phi_r(t)] + n(t) \]  

(2.5)

where $A(t)$ is the time-dependent amplitude  

$t_i$ is the code-phase plus code ambiguity  

$\phi_r$ is the residual phase variation  

$n$ is the additive white Gaussian noise  

$\omega_c$ is the carrier frequency

$A_i$, $t_i$, and $\phi_r$ are the GPS parameters that need to be estimated while minimizing errors due to the additive white Gaussian noise. The research in this dissertation will focus
on the effects of user dynamics (Chapter 4), receiver reference oscillator variations (Chapter 5), and RF interference sources (Chapter 6). The dynamics will be subdivided in dynamics due to the satellites' orbits and user motion. All three effects directly determine the temporal behavior of $A_t$, $t_r$, $\varphi_r$, and the noise.

The received C/A code GPS signal strength is specified to be a minimum of -160 dBW (ICD-GPS-200, 1991). The noise floor is well approximated by $N = kTB$, where $k$ is Boltzmann's constant ($1.3805 \times 10^{-23} \, J \, K^{-1}$), $T$ is the temperature in Kelvin ($K$), and $B$ is the input bandwidth in $Hz$. Because the noise power is dependent on the input bandwidth, the signal-to-noise ratio, $SNR$, is directly related to the bandwidth as well. The carrier-to-noise ratio, $C/N_0$, is a signal-to-noise ratio indication which is not dependent on the bandwidth, (Braasch & van Graas, 1991), and is given by $C/N_0 = S/kT$. The $C/N_0$ is expressed in dB-Hz. Given the minimum receiver GPS signal strength, and a temperature equal to $T = 290 \, K$, the minimum received $C/N_0$ is equal to 44 dB-Hz.
3. GPS SEQUENTIAL PROCESSOR

3.1 Background

This chapter describes the conventional GPS sequential processor. The term GPS sequential processor is used instead of the more-often used term ‘GPS receiver’ to distinguish between the use of block processing and sequential techniques to obtain the GPS measurements. Basic knowledge of the principles behind code and carrier tracking loops as well as the acquisition process will give more insight into the advantages and disadvantages of conventional sequential processing.

Figure 3.1 shows a block diagram of a digital GPS receiver. The signal-in-space is received by an antenna, conditioned by a radio-frequency (RF) network, down-converted to intermediate frequency (IF), and sampled at the analog-to-digital converter (ADC). After sampling, the digital IF signal is processed by a number of receiver channels. The RF network consists of a preamplifier and a prefilter. Prefiltering may be necessary to remove out-of-band RF interference in order to avoid the generation of in-band interference due to non-linearities in the down-conversion stage. The implementation of the receiver channels may be digital hardware, or software in the case of a software receiver (Akos, 1997). It is in the receiver channels, where acquisition, code, and carrier tracking takes place. A receiver processor will take the results of the N channels to compute the pseudoranges, integrated Doppler, and time measurements. Figure 3.1 also shows the presence of an automatic gain control (AGC) element. The AGC amplifies or attenuates the down-converted signal to guarantee optimal coverage of the dynamic range of the ADC.
Figure 3.1 Digital GPS Receiver Block Diagram

Copyright © 1994-1995 The Institute of Navigation)
The receiver channels perform the signal acquisition and tracking functions. The process that detects the presence of a certain PRN code and finds coarse estimates of the carrier frequency and code-phase, is referred to as acquisition, and will be discussed in Section 3.3. The code- and carrier-synchronization is addressed in Section 3.2. The discussion of the tracking loop precedes that of the acquisition because the tracking loop's correlator is also used in acquiring the GPS signals. Finally, Section 3.4 will present the major limitations of the sequential processing methods.

3.2 Tracking Functions

As stated in Chapter 2, fine time-of-transmission resolution is obtained from the code- and carrier phase measurements. These measurements are obtained by synchronizing the incoming signal with a locally-generated C/A code and carrier. Synchronization is established using tracking loops. A block diagram of a general tracking loop is shown in Figure 3.2. The phase of the tracking loop's input is estimated (detected), filtered to reduce the noise, and fed back to the phase detector using a voltage-controlled oscillator (VCO). The feedback mechanism is used to reduce the error to zero.

In GPS, the tracking loop design is optimized with respect to the noise, multipath, and dynamic tracking performance. Multipath reduction as a design criterion is an entire subject by itself, and will not be part of the present discussion. Noise and dynamic tracking error, however, are the major considerations in this dissertation. The difficulty associated with the noise and dynamic tracking performance, is the tradeoff between both performances as a function of the tracking loop bandwidth.
The wider the tracking loop, the better the dynamic tracking performance, but the noisier the measurement, and visa versa (Braasch, 1994-1995).

Noise performance of a tracking loop is determined by the tracking loop's bandwidth, the amount of predetection integration, and the carrier-to-noise ratio. The variance of the measurement is linearly related to the bandwidth of the tracking loop (Van Dierendonck, Fenton & Ford, 1992). A convenient way to express the noise performance of a tracking loop is by means of the noise equivalent bandwidth (NEB), $B_n$, given by (Cooper & McGillem, 1986) assuming $|H_{max}| \neq 0$:

$$B_n = \frac{1}{|H_{max}|^2} \int_0^\infty |H(f)|^2 df$$

(3.1)

where $H(f)$ is the closed loop transfer function of the tracking loop

$|H_{max}|^2$ is the maximum magnitude of $H(f)$
The larger the NEB, the more sensitive the tracking loop is to noise, and thus the noisier the measurement will be. Typical values for the NEB are between 5 and 18 Hz for phase-locked loops and between 0.05 and 1 Hz for delay-locked loops.

The dynamic behavior and response of the tracking loop is related to the change in the loop's input. Two major effects contributing to the input dynamics are the user motion and oscillator variations. These effects will be discussed in chapters 4 and 5, respectively.

The frequency range of operation is defined typically by four ranges, the hold-in range, the pull-out range, the pull-in range, and the lock-in range (Best, 1997 and Gardner, 1979). The *hold-in range* ($\pm \Delta \omega_{hf}$ around the reference frequency) is defined as the frequency range in which the tracking loop can maintain phase tracking; a frequency offset that exceeds this range causes the output of the tracking loop's phase detector to become a beat note rather than a constant value. The *pull-out range* ($\pm \Delta \omega_{po}$) is defined as the frequency range in which the tracking loop can dynamically maintain phase tracking. In contrary to the hold-in range definition, the pull-out range definition is based on dynamics; the occurrence of a frequency step larger than the pull-out range may cause the tracking loop to slip cycles or even lose lock. The *pull-in range* ($\pm \Delta \omega_{pi}$) is the range within which a tracking loop always becomes locked. Acquisition of lock may occur after slipping cycles for a while, and tends to be slow and often unreliable. In general, higher order loops, have a faster pull-in time (Gardner, 1979). The *lock-in range* ($\pm \Delta \omega_{l}$) is the frequency range within which a tracking loop locks within one single-beat note (1 Hz) between the reference frequency and output frequency; the tracking loop may lock up with a phase transient, but no cycle slipping will occur. This process is also called the self-acquisition of a tracking loop. In general,
The tracking loop's non-linear behavior is often explained and investigated using a phase-plane portrait (Viterbi, 1966 and Blanchard, 1976). The phase-plane portrait is generated by numerically solving a second-order differential equation for a variety of initial conditions. These solutions form trajectories in a two-dimensional case, as shown in Figure 3.3. In Figure 3.3, the phase error and its derivative are on the x and y-axis, respectively. Ideally, the phase-rate error is zero, which occurs when a trajectory intersects the x-axis. These points, the singular points, can either be stable (an equilibrium) or unstable (a saddle point). The trajectory that terminates on a saddle point is called the separatrix; the lines indicated by points A in Figure 3.3. If a trajectory lies outside the separatrices, the tracking loop will slip cycles (Gardner, 1979) until the trajectory itself turns into a separatrix. The frequency-step limit associated with the separatrices, is the pull-out range defined earlier. Below these frequency steps, or within the separatrices, the loop will eventually lock. The spiral trajectories indicate the lock-in process.

If the tracking loop is within the lock-in range, a linearized model may be used to model the tracking loop's performance. This linear model is illustrated in Figure 3.4. The phase detector is approximated by subtracting the estimated phase from the signal's phase. The general expression for the closed-loop transfer function and the phase error transfer function in the Laplace domain are given by $H(s)$ and $H_{e}(s)$, respectively (Ziemer, 1985):

$$H(s) = \frac{\Phi(s)}{\Theta(s)} = \frac{K_vK_dF(s)}{s + K_vK_dF(s)}, \quad H_{e}(s) = \frac{s}{s + K_vK_dF(s)} \quad (3.2 \text{ a-b})$$
Figure 3.3 Phase Plane Portrait for a Second Order Phase Locked Loop

(Reprinted from Blanchard, 1976 with permission of John Wiley & Sons, Inc., Furnished through the courtesy of the Jet Propulsion Laboratory, California Institute of Technology, Copyright © 1976 by John Wiley & Sons, Inc.)
where \( K_d \) and \( K_v \) are the discriminator and VCO gain respectively, and \( F(s) \) is the loop filter transfer function. Equation 3.2a shows that the order of the tracking loop is one order higher than the loop filter order. Typically, the tracking loop is first, second, or third order. The choice of the loop order depends on the desired performance and the implementation complexity. Equation 3.2a and b furthermore show that the loop performance depends on the detector gain, the NCO gain, and the loop filter architecture.

![Figure 3.4 Linear Mode Tracking Loop Model (Laplace Transformed)](image)

The dynamic tracking error of a locked-in tracking loop, is determined by its steady-state behavior to changes in phase or frequency. The steady-state performance of the tracking loop is discussed in many references (Gardner, 1979), (Ziemer, 1985), (Chie & Lindsey, 1984). In general, the steady-state behavior is given by:

\[
\lim_{t \to \infty} \psi(t) = \lim_{s \to 0} sH_e(s)\Phi(s) \tag{3.3}
\]

where \( \psi(t) \) is the tracking error and \( \Phi(s) \) is the Laplace transform of the input phase \( \varphi(t) \).
Evaluating Equation 3.3 for a first, a second, and a third order tracking loop, and for a variety of input phase functions (Ziemer, 1985), yields that the steady-state response of a first order loop, a second order loop, and a third order loop is linearly related to the line-of-sight velocity stress \( \frac{dR}{dt} \), the acceleration stress \( \frac{d^2R}{dt^2} \), and the jerk stress \( \frac{d^3R}{dt^3} \), respectively (Kaplan, 1996).

In sequential GPS processors the phase detector consists of two parts: code/carrier wipe-off and the discriminator operation. The former removes the code-phase and carrier-phase based on the tracking loop’s feedback, whereas the latter estimates or detects the code- or carrier-phase error. The basic element for code- and carrier wipe-off is the correlator, illustrated in Figure 3.5. The tracking loop input is first down-converted to baseband and then correlated with a locally-generated copy of the PRN code. The correlator integration time is referred to as the pre-detection integration time or \( T_{PDI} \). As a result of this integration the update rate of the loop is much smaller than the sampling rate. Such an operation is commonly referred to as down-sampling. The update rate of the tracking loop is often

\[
 y(t) = \int_{T_{PDI}}^{t} \cos(\omega_{io}t) \, dt
\]

Figure 3.5 Down-Conversion and Correlation (Matched Filtering)
specified by the post-detection bandwidth, \( B_I = 1/T_{PD1} \). Longer pre-detection integration times are necessary for low \( C/N_0 \) signals, whereas a pre-detection integration time as short as 1 ms may be used for strong signals (high \( C/N_0 \)). The ideal continuous integrator in Figure 3.5 is approximated by a linear time-invariant filter in a digital implementation. The most-often used digital implementation is the accumulate and dump operation, A&D. Like the ideal continuous integrator, the A&D has a frequency-response that has a low-pass characteristic. This low-pass filter (LPF) function is also used to remove the double frequency term that results from the down-conversion stages right before correlation. The digital implementation of the correlator is referred to as the digital matched filter (DMF). The reduction in bandwidth due to the correlator results in a gain which is referred to as the processing gain in spread-spectrum systems.

### 3.2.1 Carrier Tracking

Generally, two types of carrier tracking are used; carrier-frequency tracking and carrier-phase tracking. The Frequency Lock loop (FLL) is used to track the carrier-frequency, whereas a Phase Lock Loop (PLL) is used to track carrier-phase. The FLL will perform better under dynamic stress and RF interference due to its higher tracking threshold (Ward, 1998). However, a PLL provides better measurement accuracy than the FLL. The presence of large frequency uncertainties during the initial pull-in process in combination with a high accuracy requirement, may require the presence of both a FLL and a PLL; the FLL will initiate the search for the right frequency until the frequency estimate is within the PLL’s pull-in range, after which the PLL takes over. Another possible implementation is
described in (Ward, 1998) where a FLL is used in combination with a FLL-assisted-PLL (FPLL) to obtain both good measurement accuracy and a large pull-in range.

Figure 3.6 shows the block diagram for both the PLL and FLL implementations. The digital input to the carrier-phase tracking function is in-phase and quadrature down-converted to baseband, correlated with a prompt copy of the C/A code (correlator function), and input to the actual phase detector or discriminator function. The difference between the FLL and PLL is the implementation of their respective discriminators and loop-filters. The carrier-frequency being the derivative of the carrier-phase, a FLL should include one more integration in the feedback than the PLL. Possible discriminators for the FLL and the PLL are $D_{PLL}$ and $D_{FLL}$:

\[ D_{PLL} = \arctan\left( \frac{Q_p}{I_p} \right), \quad D_{FLL} = \frac{\arctan2(cross,dot)}{(t_2-t_1)360} \]  \hspace{1cm} (3.4)

where

- $dot$ equals \( I_p(t_1) \cdot I_p(t_2) + Q_p(t_1) \cdot Q_p(t_2) \)
- $cross$ equals \( I_p(t_1) \cdot Q_p(t_2) - Q_p(t_2) \cdot Q_p(t_1) \).
- $I_p(t)$ is the in-phase output of the correlator
- $Q_p(t)$ is the quadrature output of the correlator

These discriminators are maximum likelihood estimators (MLE's) and have a linear output phase and frequency error, respectively.

The tracking thresholds of the FLL and PLL are defined as the thresholds above
Figure 3.6 Carrier Tracking Loop Block Diagram
which an input error causes the tracking loop to lose lock. In case of the FLL, this threshold equals the lock-in range. Rule of thumb values for the FLL and PLL thresholds are given by 1/12 of the pre-detection bandwidth for the frequency jitter (FLL), and 15 degrees for the phase jitter (PLL) (Ward, 1998). The frequency and phase jitter are a combined result of the noise, the oscillator, and the dynamics (Kaplan, 1996).

To verify the above results, GPS data were collected with the setup described in Chapter 8, and code- and carrier-tracking loops were implemented in the ‘C’ programming language. The results for a variety of tests on satellite 7 are shown in Figures 3.7 through 3.11. First, a second order PLL with $B_n = 5 \text{ Hz}$ was implemented. The transfer function of the loop filter is shown below:

$$F(s) = \frac{\omega_n^2 + 2\zeta \omega_n s}{s}$$  \hspace{1cm} (3.5)

where $\zeta$ is the damping of the tracking loop ($= \frac{1}{2\sqrt{2}}$)

$\omega_n$ is the tracking loop natural frequency ($= 8\zeta B_n/(1 + 4\zeta^2)$)

The software receiver was already code locked. Figure 3.7 shows the output of the PLL, for an initial frequency offset of 12 Hz. After an initial transient, the PLL locks in. Frequency offsets larger than this 12 Hz resulted in occurrence of cycle slips, and the inability of the PLL to lock-in within 4 seconds. For this implementation, this implies that pull-out range is approximately 12 Hz. Figure 3.8 shows the phase-plane portrait. One can
Figure 3.7 PLL Phase Error ($B_n = 5\text{Hz}$, frequency offset: $12\text{Hz}$)

Figure 3.8 Estimated Phase-Plane Portrait from FLL and PLL Outputs
Figure 3.9 FLL Frequency Error (FLL-PLL Implementation)

Figure 3.10 PLL Phase Error (FLL-PLL Implementation)
clearly observe the spiral trajectory which is so common for lock-in (see also Figure 3.3).

Next, a combination of a FLL with a PLL was implemented. The FLL was used to perform initial pull-in (to within 8 Hz), after which the PLL took over. The PLL used was the same as the one described above. The FLL was of first order with $B_n = 100\, \text{Hz}$. The initial frequency offset was 20 Hz. While the PLL alone was not able to lock-in within the 4 seconds, the combination of a FLL and a PLL, succeeds locking-in after about 1.5 seconds. Figures 3.9 and 3.10 show the frequency and phase errors, respectively. Until lock-in, many cycle slips occur which is clear from Figure 3.11.

Figure 3.11 Occurrence of Cycle Slips During Pull-In
3.2.2 Code-Phase Tracking

Sequential GPS processors measure the code-phase by synchronizing a locally-generated C/A code with the incoming signal. The tracking loop used for this synchronization is the delay locked loop (DLL), which is illustrated in Figure 3.12. The digital input to the code-phase tracking function is in-phase and quadrature down-converted to baseband, correlated with advanced, prompt, and delayed copies of the C/A code, and input to the actual phase detector or discriminator function. The rest of the tracking loop’s operation is analogous to the description in the previous section.

There are a few available discriminator functions for the DLL (see Table 3.1). Generally, they are divided into coherent and non-coherent DLL’s. In contrary to the non-coherent DLL, the coherent DLL only needs the in-phase component and does not use squaring operations. The draw back is that the coherent DLL requires accurate phase synchronization. This phase information is obtained from the output of the PLL.

Information about the frequency and carrier phase is required to remove the carrier from the incoming signal. This becomes clear when deriving an expression for the in-phase and quadrature correlator outputs in the presence of a frequency error, $\omega_e$, and a phase error, $\varphi_e$:

\[
I(\tau) = \int_{T_{PDI}} s(t)\cos([\omega_f+\omega_e]t+\varphi_e)dt = A R_G(\tau) \text{sinc}(f_eT_{PDI}) \cos(\pi f_eT_{PDI}+\varphi_e)
\]
\[
Q(\tau) = \int_{T_{PDI}} s(t)\sin([\omega_f+\omega_e]t+\varphi_e)dt = A R_G(\tau) \text{sinc}(f_eT_{PDI}) \sin(\pi f_eT_{PDI}+\varphi_e)
\]

(3.6)
Figure 3.12 Code Tracking Loop Block Diagram
<table>
<thead>
<tr>
<th>Algorithm ($D_{\text{DLL}}$)</th>
<th>Nomenclature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coherent $(I_E - I_L) \text{sign}(I_P)$</td>
<td></td>
</tr>
<tr>
<td>Non-coherent $(I_E - I_L) I_p + (Q_E - Q_L) Q_p$</td>
<td>Dot product</td>
</tr>
<tr>
<td>$(I_E^2 + Q_E^2) - (I_L^2 + Q_L^2)$</td>
<td>Early-Late Power</td>
</tr>
<tr>
<td>$\sqrt{(I_E^2 + Q_E^2)} - \sqrt{(I_L^2 + Q_L^2)}$</td>
<td>Early-Late Envelope</td>
</tr>
</tbody>
</table>

$I_E$, $I_L$, $I_p$ are accumulated in-phase outputs of the early, late, and prompt correlators, respectively

$Q_E$, $Q_L$, $Q_p$ are accumulated quadrature outputs of the early, late, and prompt correlators, respectively
where $T_{PDI}$ is the pre-detection integration period. The output of the correlators is equivalent to the output of a matched filter. For this reason, this part of a digital receiver is referred to as a Digital Matched Filer (DMF). At the DMF, an inherent down-conversion takes place. Because the code period equals 1ms, a PDI of at least that length is required. As a result, the update rate after the correlator is smaller than or equal to 1,000 Hz. With sample rates as large as 5 Msp, the correlators function as a down-sampling stage, $M = 5,000$ (Strang, 1997). Weaker signals require even longer PDI's. For a typical PDI of 0.02 seconds, the down-sampling factor equals, $M = 10,000$. In this case, the update rate of the tracking loops equal 50 Hz.

The code-rate (1.023 MHz) is three orders of magnitude smaller than the carrier frequency (1,575.42 MHz). Consequently, the code-rate Doppler shift is three orders of magnitude smaller than the carrier-frequency Doppler shift, because the magnitude of the Doppler shift is a linear function of the frequency (see Chapter 4). Due to the lower code-rate Doppler shift, the DLL may be of order 1 and the NEB of a code-tracking loop is typically much smaller than the NEB of the PLL and FLL; $B_n$ ranges from 0.05 Hz to 1 Hz.

3.3 Signal Acquisition

Prior to the pull-in process of the code and carrier tracking loops, coarse estimates of the code-phase and carrier-frequency must be determined. This process is referred to as acquisition of the GPS signal. Acquisition of the GPS signal consists of the estimation of three parameters; the CDMA PRN code, the code-phase estimate, and the carrier frequency. This estimation process is commonly referred to as a search process.
While in acquisition mode, each channel of the digital receiver shown in Figure 3.1 may perform a two-dimensional search for one SV; carrier-frequency versus code-phase. Figure 3.13 shows the two-dimensional search space. The code-phase axis range is 1,023 chips, and the carrier frequency range is typically ±10 kHz. The resolution of the carrier-frequency and code-phase estimates are determined by the search bin-widths along the x- and y-axes of the two-dimensional search grid. The code-phase bin-width depends on the resolution required for the DLL to ‘pull in’. In general a half-chip bin-width is sufficient. The carrier-frequency bin-width depends on the signal strength. For weaker signals, the receiver channel needs to average over a longer period and thus requires a smaller bin-width. For the same Doppler range this implies a heavier computational burden and, thus, a longer time-to-acquisition. Typical carrier-frequency bin-widths are 667 Hz for a 1 ms integration time and 67 Hz for a 20 ms integration time (Kaplan, 1996). Three search methods can be used to search the 2-D grid, the sequential code-phase search, the parallel frequency space search, and the parallel code-phase search.

The sequential search is the most commonly used technique (Rappaport & Grieco, 1984). For each PRN, the output of a DMF for each bin in Figure 3.13 is inspected. The bin in which the DMF output gives the strongest response and exceeds a preset threshold will identify the coarse frequency-offset and code-phase. In the presence of noise, this process is a detection process based on the statistical properties of the noise. (Jovanovic, 1988) goes into more detail on the mathematics behind the serial search. Inspection of each bin of the search grid makes the serial search a computationally intensive method (Akos, 1997).
Figure 3.13 Two-Dimensional Acquisition Search Grid
The parallel frequency space search (Cheng, et al., 1990) improves the computational efficiency significantly by limiting the search to the code-phase axis. The incoming signal is correlated with a locally generated code without down-conversion to baseband. Taking the Fourier transform of the output of this correlator, and inspecting its magnitude squared, provides a measure of the presence of a signal in any of the frequency bins.

The most efficient acquisition method is described by (van Nee & Coenen, 1992); the parallel code-phase search. With the parallel frequency space search, the search was limited to the code-phase axis. With a code-phase bin-width of 0.5 chip, that process would still require 2046 bins to be searched. It would be more efficient to search the frequency-offset axis. One would need to inspect only 20 bins for a typical bin-width of 500 Hz and a frequency range of ±10 kHz. The described method is based on the realization that the output of a digital correlator, \( R[m] \), may be given by the inverse Fourier transform of the multiplication of the Fourier transforms of the incoming signal and local code, respectively, or:

\[
R[m] = \sum_{n=0}^{M-1} s[n] \cdot G_L[n+m] = s[n](*)G_L[-n] = IFFF\{FFT\{s[n]\}\cdot FFT\{G_L[n]\}\} \quad (3.7)
\]

The efficiency of this algorithm is based on the computational efficiency with which a FFT can be performed in modern processors.

### 3.4 Advantages and Disadvantages of Sequential Processing of GPS

Sequential processing techniques are widely used for code and carrier acquisition and synchronization. Consequently, these techniques are well-understood, and dedicated analog
and digital components as well as designs in ASICs are commercially available at a relatively low-cost.

The main disadvantage of sequential processing is its inherent loss-of-information due to down-sampling of the signal at the correlators and the causality restriction on the sequential implementation. Down-sampling of the discrete-time signal from sample rate to the tracking loop update rate, results in a deteriorated observability of phase, frequency, and frequency change. Furthermore, in a high dynamic environment, specification of long pre-detection integration times may result in a mismatch between desired and actual discriminator output. This effect may be verified by substituting Equation 3.6 in the DLL and PLL discriminator functions of Table 3.1 and Equation 3.4, respectively.

Non-linearity of the tracking loops, as illustrated by Figure 3.3, limits the tracking loop’s flexibility and dynamic range of operation, as discussed in Section 3.2. Careful design of loop parameters is necessary depending on the receiver’s application. Therefore, more robust receivers may be complex and thus expensive. Furthermore, change in phase or frequency may result in cycle slips or even loss of lock. In these circumstances re-acquisition and lock-in of the signal is required, which results in a temporary loss of an SV, and thus a degraded navigation service.

Lastly, the correlator and tracking loop filters may introduce undesired limitations and tradeoffs. Random walk behavior due to integration of thermal noise and random oscillator variations, may put limits on the time constants of the filters involved when output drift needs to be avoided. Furthermore, tradeoffs between noise reduction and dynamic tracking capability directly drive the tracking loop filter design.
4. TRANSMITTER/RECEIVER DYNAMICS

4.1 Background

Motion of a user receiver with respect to a transmitter in inertial space gives rise to the observation of a shift in the frequency components of the GPS signal that is being received. This so-called Doppler shift is linearly related to the magnitude of the relative velocity, or speed, of the transmitter and receiver (Alonso & Finn, 1983). This relative velocity is directly related to the rate at which the line-of-sight between receiver and transmitter is changing, and is therefore referred to as the line-of-sight velocity, \( v_{\text{los}} \). The observed frequency, \( f'_i \), of every spectral component (sine wave), \( f_i \), of the transmitted GPS signal is mathematically given by:

\[
f'_i = f_i \left(1 - \frac{v_{\text{los}}}{c}\right), \quad \text{or} \quad \Delta f_i = -\frac{v_{\text{los}}}{c} f_i
\]  

where \( c \) is the speed of light in vacuum and \( \Delta f_i \) is the Doppler shift for frequency \( f_i \).

The following paragraph will discuss the effect of the line-of-sight motion on the signal model of Chapter 2 is illustrated in Section 4.2.

4.2 The Doppler Effect

Section 4.1 showed that there is a relation between the line-of-sight velocity and the Doppler shift of a spectral component. Another way to interpret Equation 4.1 is that the line-
of-sight velocity scales the frequency axis in the signal’s frequency domain by a factor $\gamma$, hereafter referred to as the inertial Doppler scale factor. This scale factor equals $\left(1 - \frac{V_{los}}{c}\right)$; if $\gamma > 1$ the frequency axis is linearly expanded, if $\gamma < 1$ the frequency axis is compressed.

To characterize the impact of the Doppler effect on a received GPS signal, the GPS signal of Equation 2.2 needs to be rewritten as an expression containing pure spectral components. Fortunately, the Gold code, $G_f(t)$, is a periodic signal and may be Fourier expanded according to (Papoulis, 1980):

$$G_f(t) = \sum_{k=0}^{\infty} a_k \cos(\omega_G kt) + b_k \sin(\omega_G kt) \quad (4.2)$$

where $\omega_G = \frac{2\pi}{T_G}$ is the inter-line bandwidth, $T_G = 1 \text{ ms}$ is the duration of one Gold code and $a_k$ and $b_k$ are shown below:

$$a_k = \frac{2}{T_G} \int_0^{T_G} G_f(t) \cos\left(\frac{2\pi k t}{T_G}\right) \, dt, \quad b_k = \frac{2}{T_G} \int_0^{T_G} G_f(t) \sin\left(\frac{2\pi k t}{T_G}\right) \, dt \quad (4.3)$$

Substituting Equation 4.2 into Equation 2.2, neglecting the effect of the data bits, $D(t)$, and rewriting the trigonometric function, yields a periodic expression for the GPS signal:
\[ s(t) = AD \left\{ \sum_{k=0}^{\infty} a_k \cos(\omega_G kt) + b_k \sin(\omega_G kt) \right\} \sin(\omega_{LL} t) \]

\[ = AD \sum_{k=0}^{\infty} a_k \cos([\omega_{LL} - \omega_G k]t) + b_k \sin([\omega_{LL} - \omega_G k]t) \]

\[ + AD \sum_{k=0}^{\infty} a_k \cos([\omega_{LL} + \omega_G k]t) - b_k \sin([\omega_{LL} + \omega_G k]t) \]

Equation 4.4 shows that the GPS signal's spectrum consists of an infinite number of spectral lines, each 1,000 Hz apart. The Doppler scale factor may now be applied to the spectral lines, resulting in an expression for the observed signal, \( s'(t) \):

\[ s'(t) = AD \sum_{k=0}^{\infty} a_k \cos(\gamma^* [\omega_{LL} - \omega_G k]t) + b_k \sin(\gamma^* [\omega_{LL} - \omega_G k]t) \]

\[ + AD \sum_{k=0}^{\infty} a_k \cos(\gamma^* [\omega_{LL} + \omega_G k]t) - b_k \sin(\gamma^* [\omega_{LL} + \omega_G k]t) \]

Equation 4.5 indicates the scaling of the carrier frequency. However, as a result of the frequency scaling operation the bandwidth between the spectral lines of the GPS PSD, is scaled also. This effect is illustrated in Figures 4.1 and 4.2. Figure 4.1 shows the original PSD as derived in Chapter 2. Figure 4.2 shows PSD of the received signal with the apparent frequency scaling. Analogous, the observed code rate is given by \( R_c' = \gamma^* R_c \). Typical numbers for the Doppler on the carrier frequency are within a frequency range of -10,000 and 10,000 Hz.
Figure 4.1 Transmitted C/A Code Line Spectrum

Figure 4.2 Received C/A Code Line Spectrum
5. OSCILLATOR EFFECTS

5.1 Background

Precise location of a signal in the time or frequency domain requires the use of precise or stable reference oscillators in the signal down-conversion and sample stages of a GPS receiver. Furthermore, a precise local oscillator or clock may be necessary to obtain an accurate time-of-reception estimate. In general, the local oscillator errors on the time-of-reception estimate are common to the GPS receiver channels and may be estimated by adding the clock error as an additional unknown in the least squares algorithm (Kaplan, 1996) or estimating it using a Kalman filter (Brown & Hwang, 1998). However, oscillator behavior may limit the flexibility in code- and carrier tracking loop designs, and block processing algorithms. Therefore, oscillator quality measures need to be defined, and the impact of the oscillator variations on the stages of down-conversion of the observed signal need to be determined.

Section 5.2 of this chapter will discuss measures that have been developed to quantify the oscillator behavior and quality. Next, a model will be proposed that may be used to simulate the noise-like character of an oscillator. This model will be used in the GPS signal generator to evaluate the response of block processing techniques to oscillator variations. Finally, the effects of oscillator variations in the stages of down-conversion and sampling will be quantified in Section 5.4. The impact of local oscillator phase and frequency noise on the down-conversion and sampling stages will be discussed also.
5.2 Oscillator Measures

Oscillator frequency stability can be measured in the time and frequency domain (Allan et al., 1974 and Barnes et al. 1970). Both methods are based on a basic oscillator output model given by a voltage $v(t)$:

$$
v(t) = [v_0 + \varepsilon_0(t)] \sin(\omega_0 t + \varphi(t))
$$

(5.1)

where

$\nu(t)$ is the frequency standard output

$v_0$ is the normal voltage output

$\varepsilon_0(t)$ is the amplitude variation as a function of time

$\omega_0$ is the nominal oscillator output ($= 2\pi f_0$)

$\varphi(t)$ is the phase variation as a function of time

In the scope of this dissertation, the variations in the amplitude are considered to be small compared to the phase variations because of the small amplitudes of the received GPS signals. Given Equation 5.1, the instantaneous phase and frequency of the oscillator source is equal to $\Phi(t) = \omega_0 t + \varphi(t)$ radians and $\Xi(t) = f_0 + \dot{\varphi}(t)/2\pi$ Hz, respectively. Define the instantaneous fractional phase deviation as $x(t) = \varphi(t)/\omega_0$ and the instantaneous fractional frequency deviation from the nominal frequency of the oscillator as $y(t) = dx(t)/dt = \dot{\varphi}(t)/\omega_0$. Then, an average value of the frequency deviation can be defined (Allan, et al., 1974) as follows:
where $\tau$ is the averaging period of the instantaneous fractional frequency and $t_{k+1} = t_k + T$. $T$ is the repetition interval of the measurements of duration $\tau$. Figure 5.1 shows the interpretation of this integral; the averaged instantaneous frequency is given by the area underneath $\varphi(t)$ between two consecutive $\tau$, normalized to the nominal carrier frequency.

In the example illustrated in Figure 5.1, the repetition interval is equal to $8\tau$ and, consequently, there exists a dead time of $7\tau$ between the measurements.

The frequency measure is given by the power spectral densities (PSD) of $y(t)$ and $x(t)$ given by $S_y(f)$ and $S_x(f)$, respectively. A model for these spectral densities will be defined in the next paragraph. Furthermore, the power spectral density function of the instantaneous fractional phase can be related to the power spectral density of the fractional instantaneous frequency by $S_y(f) = (1/j\omega)(1/j\omega)^*S_y(f) = 1/(2\pi f)^2S_y(f)$ (Allan, 1974).

The general expression for the time-domain measure is given in (Barnes, et al., 1970, and Allan, et al., 1974):

\[
<\sigma_y^2(N,T,\tau)> = \left\langle \frac{1}{N-1} \sum_{n=1}^{N} (\bar{y}_n - \frac{1}{N} \sum_{k=1}^{N} \bar{y}_k)^2 \right\rangle
\]  

Equation 5.3 can be interpreted as the infinite time average of the sample variance of $\bar{y}_k$.

The value of this time average with $N = 2$ and $T = \tau$ ($t_k + \tau = t_k + T = t_{k+1}$) has been
Figure 5.1 Phase Noise Characteristic in Time-Domain
accepted as the preferred measure of frequency stability in the time domain (Barnes, et al., 1970) and may be written as:

$$\sigma_\delta^2(\tau) = \frac{\left(\bar{y}_{k+1} - \bar{y}_k\right)^2}{2} \quad (5.4)$$

The expression in Equation 5.4 is referred to as the Allan variance (Allan, 1966); the time-average of the two-sample variance, see Figure 5.2. Choosing $T = \tau$ results in the absence of a dead time between measurements as was the case in Figure 5.1. An approximation of this Allan variance is given by (Barnes, et al., 1970):

$$\sigma_\delta^2(\tau) \approx \frac{1}{m} \sum_{k=1}^{m} \frac{(\bar{y}_{k+1} - \bar{y}_k)^2}{2} \quad (5.5)$$

The previous paragraph discussed the measures used to define the frequency stability in frequency and time domains. A model that has been found useful for the fractional frequency $y(t)$ (Barnes, et al, 1970 and Allan, et al., 1974) is a summation of five noise processes $z_i(t)$:

$$y(t) = z_{-2}(t) + z_{-1}(t) + z_0(t) + z_1(t) + z_2(t) \quad (5.6)$$

with frequency domain representation:

$$S_y(f) = h_{-2}f^{-2} + h_{-1}f^{-1} + h_0 + h_1f + h_2f^2 \quad (5.6)$$
Figure 5.2 Phase Noise Characteristic in Time-Domain for the Allan Variance

\[ (N = 2, \; T = \tau) \]
where \( f \) is the frequency

- \( h_2 f^{-2} \) is the random frequency walk (\( h_2 \) is a constant)
- \( h_1 f^{-1} \) is the flicker frequency noise (\( h_1 \) is a constant)
- \( h_0 \) is the random phase walk or frequency noise (\( h_0 \) is a constant)
- \( h_1 f^1 \) is the flicker phase noise (\( h_1 \) is a constant)
- \( h_2 f^2 \) is the white phase noise (\( h_2 \) is a constant)

Using \( S_x(f) = 1/(2\pi f)^2 S_x(f) \), the phase noise spectral density may be expressed as:

\[
S_x(f) = h_2 f^{-4} + h_1 f^{-3} + h_0 f^{-2} + h_1 f^{-1} + h_2
\]  

(5.7)

Table 5.1 shows the Allan variance corresponding to the 5 noise sources for both phase and frequency noise.

Figure 5.3 shows the typical phase noise spectra for a variety of oscillators. Both the temperature-compensated crystal oscillator (TCXO) and the oven-controlled crystal oscillator (OCXO) are oscillators used in GPS receivers. These crystals have a short term frequency stability of approximately \(1.10^{-6}\) and \(2.10^{-11}\), respectively, and a center frequency of 20.473 MHz. The Rubidium oscillator data came from a specification sheet of the Frequency Electronics, Inc. 10 MHz oscillator. Specified frequency stability is on the order of \(1.4\cdot10^{-11}/\sqrt{\tau}\). An oscillator with excellent low-frequency phase noise is the PTS 040
Table 5.1 Time and Frequency Measures of the Frequency and Phase Stabilities

(Allan, 1974)

<table>
<thead>
<tr>
<th></th>
<th>$S_x(f)$</th>
<th>$S_y(f)$</th>
<th>$\sigma_x^2(\tau)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Frequency Walk</td>
<td>$\frac{h_{-2}}{f^2}$</td>
<td>$\frac{h_{-2}}{(2\pi)^2 f^4}$</td>
<td>$\frac{h_{-2} (2\pi)^2</td>
</tr>
<tr>
<td>Flicker Frequency Noise</td>
<td>$\frac{h_{-1}}{f}$</td>
<td>$\frac{h_{-1}}{(2\pi)^2 f^3}$</td>
<td>$h_{-1} \cdot 2\ln 2$</td>
</tr>
<tr>
<td>Random Phase Walk</td>
<td>$h_0$</td>
<td>$\frac{h_0}{(2\pi)^2 f^2}$</td>
<td>$\frac{h_0}{2}</td>
</tr>
<tr>
<td>Flicker Phase Noise</td>
<td>$h_1</td>
<td>f</td>
<td>$</td>
</tr>
<tr>
<td>White Phase Noise</td>
<td>$h_2 f^2$</td>
<td>$\frac{h_2}{(2\pi)^2}$</td>
<td>$h_2 \frac{3 f_0}{(2\pi)^2 \tau^2}$</td>
</tr>
</tbody>
</table>
Figure 5.3 Phase Noise for Typical Oscillators ($S_x(f)$)
5.3 Oscillator Models

A large number of oscillator models is available in the literature. Most models are based on modeling the random frequency walk, the flicker frequency noise, and the random phase walk. The general implementation is illustrated in Figure 5.4 and is based on integrating white Gaussian noise. Such processes are known as Gauss-Markov processes (Gelb, 1974, Brown & Hwang, 1992, Brown, 1984, Mandelbrot & van Ness, 1968). Implementation of white and random walk frequency noise is straightforward. Models specifically for Kalman filter applications have been investigated in (van Dierendonck, McGraw & Brown, 1984, Brown & Hwang, 1992), whereas (Denaro, 1983, Uijt de Haag, 1997) describe receiver clock error models in a similar fashion for GPS signal model applications. Modeling flicker frequency noise is more complicated because it involves fractional integration (Barnes and Allan, 1966). This fractional integration is clear from the \( 1/s^{1/2} \) term in Figure 5.4. Because of its complexity a variety of models for flicker frequency noise have been investigated. A summary of models is given in (Barnes, 1976). One of the models, which is easy to implement, is a model that uses Auto-Regressive Integrated Moving Average (ARIMA) processes to describe the fractional Brownian noise (Barnes & Jarvis, 1971, Barnes, 1976). Another approach is to neglect the flicker noise, and compensate for its absence by increasing the random frequency and random phase walk (Brown & Hwang, 1992).

The continuous time oscillator model is rewritten to allow a discrete-time
Figure 5.4 General Continuous-Time Oscillator Model
implementation. First, the continuous-time integrator is replaced by a digital boxcar integrator:

\[
\frac{1}{s} = \frac{T_s}{z-1}
\]  

Neglecting the flicker noise and white phase noise, and substituting the continuous-time integrators in Figure 5.4, by their digital equivalents, yields the following state-space model:

\[
\begin{bmatrix}
  x[n+1] \\
  y[n+1]
\end{bmatrix} =
\begin{bmatrix}
  1 & T \\
  0 & 1
\end{bmatrix}
\begin{bmatrix}
  x[n] \\
  y[n]
\end{bmatrix}
+ 
\begin{bmatrix}
  0 \\
  T
\end{bmatrix}
w_{-2}[n] 
+ 
\begin{bmatrix}
  T \\
  0
\end{bmatrix}
w_0[n]
\]

\[
= A x[n] + B_1 w_{-2}[n] + B_2 w_0[n]
\]

where \( T \) is the sample time.

\( x[n] \) is the state vector

\( A \) is the state transition matrix

\( B_1, B_2 \) are the input coupling matrices

\( w_{-2}[n], w_0[n] \) are white, Gaussian noise sequences with zero-mean and variance \( \sigma^2_{w_{-2}} \) and \( \sigma^2_{w_0} \), respectively.

The covariance matrix, \( X[n] \), of state vector, \( x[n] \) may be calculated iteratively (Jacquot, 1981), as follows:

\[
X[n+1] = A X[n] A^T + B_1 \sigma^2_{w_{-2}} B_1^T + B_2 \sigma^2_{w_0} B_2^T
\]  

(5.9)
An advantage of a state-space model, such as Equation 5.8, is its capability to be converted from a high rate to a low rate model (McGraw & Braasch, 1998).

The covariance matrix given in Equation 5.9 approximates the results of algebraically evaluating the closed form integrals. (Brown & Hwang, 1992) derive the continuous time result as a function of the total integration time, \( \Delta t \):

\[
X(\Delta t) = \begin{bmatrix}
\sigma_{w_2}^2 \Delta t + \sigma_{w_2}^2 \frac{\Delta t^3}{2} & \sigma_{w_2}^2 \Delta t^2 \\
\sigma_{w_2}^2 \Delta t & \sigma_{w_2}^2 
\end{bmatrix}
\]  

(5.10)

Good approximations for \( \sigma_{w_2}^2 \) and \( \sigma_{w_0}^2 \) are \( 2\pi^2 h_2 \) and \( h_0/2 \), respectively. The covariance analysis shows that the oscillator error is expected to grow as a function of time. The error has therefore a strong drift-like character.

5.4 A/D Conversion and the Sample Doppler Effect

GPS receiver processing requires down-conversion and sampling of the signals. Figure 5.5 shows a block diagram of standard down-conversion and sampling stages for a bandpass sampling setup (Akos, 1997). The incoming signal \( s(t) \) is first mixed with a local oscillator, \( s_{lo}(t) \), to obtain a signal at an intermediate frequency (IF). This local oscillator will be referred to as the pre-sample local oscillator. Next, the signal is filtered and sampled at a sampling rate that corresponds to a sample period of \( T_s \). The result is a discrete-time signal, indicated by the square brackets. Mixing of the sampled signal to baseband is
performed by a second local oscillator that is referred to as the post-sampling local oscillator, $s_{lo2}[n]$. The final step is the despreading of the signal by multiplication with a locally generated PRN code. At all the mixing stages, errors may be introduced due to oscillator instabilities in the locally generated signals. The next paragraphs will discuss how the oscillator stability effects the signal observed by the GPS signal processor.

5.4.1 Pre-sampling Local Oscillator

Let the local oscillator signal, $s_{lo1}(t)$, be given by $\cos(\omega_{lo1}t + \varphi_{lo1}(t))$ where $\varphi_{lo1}(t)$ is the phase noise as discussed in the previous sections. For reasons of simplicity, $\varphi_{lo1}(t)$ will initially be given by $\varphi_{lo1}(t) = \epsilon_{lo1}t + \phi_{0lo1}$, a frequency- and phase-offset. Multiplication of this signal with the observed GPS signal (ideal mixer), which was derived in Chapter 4, yields:

$$x(t) = \frac{1}{2} A G_f(t) D(t) \{ \cos([\gamma \omega_{li} - \omega_{lo1} - \epsilon_{lo1}]t - \varphi_{0lo1}) + \cos([\gamma \omega_{li} + \omega_{lo1} + \epsilon_{lo1}]t + \varphi_{0lo1}) \} \quad (5.11)$$
Under the assumption that the bandpass filter of Figure 5.5 completely removes the double frequency term and that still most of the signal’s energy has passed through the bandpass filter (typically 95% for $B = 2.5$ MHz), the following expression for $x_2(t)$ is found:

$$x_2(t) = \frac{1}{2}AG_i(t)D(t)\cos((\gamma \omega_{L,i} - \omega_{l_0,l} - \epsilon_{l_0})t - \phi_{l_0,l})$$ (5.12)

As expected, the mixing operation just offsets the incoming signal’s spectrum by the frequency of the local oscillator, $\omega_{l_0,l}$. In this dissertation this operation is referred to as a frequency translation. Note that the entire filtered GPS spectrum has been translated, but that the code chipping rate remains unchanged. This differs from the effect of the user-satellite motion on the signal.

If the random oscillator variations described in the previous section are taken into account, and added to the oscillator phase variation model, Equation 5.12 may be rewritten as follows:

$$x_2(t) = \frac{1}{2}AG_i(t)D(t)\cos((\gamma \omega_{L,i} - \omega_{l_0,l} - \epsilon_{l_0})t - \phi_{l_0,l} - \phi_{l_0}(t))$$ (5.13)

5.4.2 Analog-to-Digital Conversion

Discrete-time signals are analog signals that are sampled along the time-axis, whereas discrete-amplitude signals are digitized along the amplitude axis. To obtain a digital version of an analog signal, it needs to be digitized along both the time and the amplitude axes. The discussion in this section is limited to the discrete-time signals. Discretization in amplitude
will add noise to the signal in the form of quantization errors. The signal-to-quantization-noise ratio is inverse proportional to the number of quantization levels or bits. Although discretization in amplitude is an important design criterion in a digital implementation, it will not be considered in this dissertation.

To obtain the discrete-time version, $x_2[n]$, of $x_2(t)$ the signal is sampled at a sampling rate equal to $f_s$ samples/second (sps). A continuous-time expression for the sampled signal can be given by (Papoulis, 1977 and 1980):

$$x_2(t_{(t-nT_s)}) = \frac{1}{2} AG_i(t) D(t) \cos([\gamma \omega_L - \omega_{lo} - \varepsilon_{lo}] t - \varphi_{lo}) \delta(t-nT_s)$$  \hspace{1cm} (5.14)

If there were no variations in the sample rate, Equation 5.14 could easily be converted to its discrete-time equivalent. In the presence of sample-rate variations, however, one needs to determine the frequency or frequencies of the received signal as observed by the receiver’s logic. Assume a sample period offset, $T_e$, then the sample period is given by $T'_s = T_s + T_e$. The impact of this offset on a received sine wave is illustrated in Figure 5.6 for a negative $T_e$. The following analysis is similar for a positive $T_e$. When sampled at a rate of $f_s = 1/T_s$ every period is sampled by 14 samples (Figure 5.6 a). The presence of a 50% error in the sample rate, or $T'_s = 2T_s$ results in the observation of 7 samples per sine period. Under the observer’s assumption that $T'_s = T_s$, the observed sine period equals $T'_{SINE} = (7/14)T_{SINE}$. In general, the observed frequency in the presence of a sample rate error can be expressed by:
\[ f_i' = f_i \left( \frac{T_s'}{T_s} \right) = f_i \left( 1 + \frac{T_e}{T_s} \right), \text{ or } \Delta f_i = \frac{T_e f_i}{T_s} \]  

(5.15)

When the actual sample rate is higher than the specified sample rate, the receiver’s logic will observe a frequency that is lower than the actual frequency. Comparison of Equations 5.15 and 3.1 shows a strong similarity between the effect of a sample period offset on the observed frequency and the Doppler effect. For this reason the effect of the sample-period offset will be referred to as the sample Doppler. Like the inertial Doppler effect, the sample Doppler effect can be achieved by scaling the frequency axis of the signal by a scale-factor:

\[ \beta = \left( 1 + \frac{T_e}{T_s} \right) = \frac{f_s}{f_s'} \]  

(5.16)

For the case of bandpass sampling the observed frequency is both scaled and translated. The frequency scaling and translation factors are given by \( \beta \) and \( \alpha \), respectively. The derivation of these parameters is given in appendix B and \( \beta \) and \( \alpha \) may be expressed as follows:

\[ f_i = \beta f_i + \frac{T_e}{T_c^2} M_{fix} = \beta f_i + \alpha \]  

(5.17)

where \( M_{fix} = \| f_i / f_s \| \).
Figure 5.6 Effect of a Sample Rate Offset
Using Equation 5.14, a discrete-time expression for $x_2(t)$ can be derived:

$$x_2[n] = \frac{1}{2} A G_i^{ii}[n] D^{ii}[n] \cos(\omega_{obs} nT_s - \varphi_{lo})$$

(5.18)

where:

$$\omega_{obs} = \beta [\gamma \omega_{li} - \omega_{lo} - \varepsilon_{lo}] + \alpha$$

(5.19)

and $G_i^{ii}(t)$ is the C/A code as observed by the signal processor’s logic. The new chipping rate is given by:

$$R_{obs} = \beta \cdot \gamma \cdot R_c$$

(5.20)

5.4.3 Post-sampling Local Oscillator

Post-sampling mixing is performed by the signal processor unit in software. As a result, the post sampling oscillator will not have frequency or phase variations that are unknown to the observer. The same observation applies to the locally-generated PRN code.
6. RADIO FREQUENCY INTERFERENCE

6.1 Background

Radio frequency interference (RFI) is one of the major concerns in any radio-based navigation aid. Even though GPS, as a spread-spectrum system, is designed to withstand a substantial amount of interference, it has proven to be vulnerable to RFI. (Moelker, 1998) and (Ward, 1994-1995) describe a variety of events in which the GPS performance was degraded by RF interference.

Often, interference is divided into intentional and unintentional interferences. Unintentional interference occurs even though the GPS band is protected by the Federal Communication Commission (FCC). Examples are TV harmonics, microwave links, pulsed interference from radar, spurious emissions from commercial electronics, satellite communications systems, accidental transmission of signals in the wrong band, etc.

Intentional interference is, in general, thought of as interference generated during a state of war, but may include spoofing and jamming by terrorists, military and civil frequency transmission tests or even transmissions by hackers.

Another way of specifying interference is by the band the RF interference source is located in; in-band specifies the sources that are in the GPS band, all other sources are referred to as out-of-band interference. Examples of out-of-band interference are the earlier-mentioned satellite communications systems, whereas spoofing, intentional generation of GPS-like signals, is considered in-band interference. Another in-band interference source is GPS itself. Being a CDMA system, GPS discerns between the signals originated from
different satellites by modulating them with different spreading codes (Chapter 2), which results in approximately 23 dB isolation between the two signals. Difference in Doppler frequency further increases this isolation. However, if the received signal of two satellites have equal Doppler, two effects may occur; fading and pseudorange errors. (de Bruijn, van Graas & Uijt de Haag, 1999) have shown that this form of self-interference occurs quite often and can last for tens of minutes depending on the receiver design. The pseudorange error effect was already noted by (van Nee, 1995), and looks like multipath to the other satellite. This effect becomes more noticeable when there is a power difference between two satellites such that the isolation is less than 23 dB. This may be due to the antenna pattern, satellite transmit power differences, and atmospheric attenuation differences. The fading effect is based on the destructive and constructive interference of the signals’ carriers and is the worst when the received signals from two satellites are equal.

In the context of this dissertation, the RF interference distinction is made based on their spectral and temporal behavior. Globally, three categories are defined; wide-band interference (WBI), narrow-band interference (NBI), and pulsed interference. Pulsed interference may be both wide or narrow-band. Figures 6.1 through 6.4 show the time-frequency (T-F) behavior of wide-band (WBI), narrow-band (NBI), wide-band pulsed (WBPI), and narrow-band pulsed RF interference (NBPI). The figures show the frequency versus the time. Wide-band interference covers a wide frequency band and remains present for a long period of time. Narrow-band interference is well-located along the frequency axis, but not along the time-axis, covering a long period of time. The pulsed interference in Figures 6.3 and 6.4 is characterized by a duty cycle along the time axis. Typically this duty-
Figure 6.1 Time-Frequency Representation of WBI

Figure 6.2 Time-Frequency Representation of NBI
Figure 6.3 Time-Frequency Representation of WBPI

Figure 6.4 Time-Frequency Representation of NBPI
cycle is very low, but the pulse energy is very high.

The energy of an interference source is directly related to the time-frequency coverage in Figures 6.1 through 6.4. The more time-frequency area a specific interference source covers the more power the source requires. For example, using the same amount of power, it will be more difficult to jam a GPS user at a specific instance in time using wide-band interference, than it would be with a narrow-band interferer.

6.2 Interference Suppression and Detection

Because of the severe consequences the effect of RF interference may have on the GPS performance and integrity, it is necessary to detect or even mitigate or suppress RF interference. Interference suppression and detection techniques are divided into four categories based on the interference discussion in (Moelker, 1998); amplitude domain, spatial domain, polarization domain, and time and/or frequency domain.

Amplitude domain techniques are based on the amplitude distribution of the GPS signal, the noise, and the interference source. Knowledge of these distributions may be used to detect or even mitigate the effects of interference. One method to detect RF interference is located at the AGC of the GPS receiver (see Figure 3.1). Under interference-free conditions the GPS signal is below the noise floor at the input of the AGC. The presence of an interference source will change the rms level of the AGC’s input. In other words, the AGC detects a change in the amplitude characteristics of its input signal. The advantage of this detector is that it works even when the receiver processor is not tracking the GPS signals. Using the ADC in an adaptive way can furthermore make the receiver more robust to
interference by changing the ADC threshold to the point where the GPS signal can be tracked by the receiver.

Spatial techniques are based on steering a null of the gain pattern of an antenna towards the external jammer (Ward, 1994-1995). These techniques require an antenna with a controllable gain pattern, or the use of multiple antennas in combination with the necessary estimation techniques (Moelker, 1998). Polarization domain techniques are based on the different polarization characteristics of the desired signal and the interference source.

The last technique is the time and/or frequency domain filtering. Temporal filters estimate the RF interference source and subtract it from the incoming signal. In case of narrow-band interference (NBI) one can think of adaptive linear predictors (Ketchum, 1982) and non-linear notch filters (Laster & Reed, 1997), whereas wide-band interference (WBI) may be mitigated by smoothing or Wiener filters. Another approach would be to transform the signal to another domain and perform the interference suppression in this new domain. The next section will address these techniques in more detail. The conventional sequential GPS receiver has some temporal filtering capability built-in, in form of the tracking loop bandwidth (Parkinson & Spilker, 1996). Making the tracking loop narrower will suppress both narrow and wide-band interference. The disadvantage is a deteriorated dynamic tracking performance (see Chapter 3). For this purpose, GPS receivers are often integrated with Inertial Measurement Units, IMUs, (Phillips & Schmidt, 1996); the IMU aids the carrier-tracking loop, so that its NEB can be made very narrow. Although the tracking bandwidth removes much interference, narrow-band continuous wave (CW) signals may leak through the filter when they coincide with one of the spectral lines of the GPS signal C/A

All previously discussed approaches have each their advantages and disadvantages. Therefore, joint use of any variety of techniques may be used to optimize interference suppression for a particular application.

6.3 Transform and Joint Time-Frequency Techniques

Transformation to another domain may improve the interference source’s observability, and simplify excision schemes. The simplest method is pure frequency domain interference excision. This method is illustrated in Figure 6.5. The input signal is transformed to the frequency domain by the Fast Fourier Transform where the interference is detected and excised. After removal of the interference, the signal is transformed back to the time-domain (Milstein, 1980 and Milstein, 1983), where normal processing techniques, such as correlation in the GPS receiver, may be applied. The major disadvantage of the technique is the impact of the excision function on the desired signal; excision results in the removal of desired signal energy (Arkansa & Smith, 1996). For this reason, the use of pure frequency domain excision is limited to NBI. Figures 6.6. and 6.7 give alternatives for the block diagram in Figure 6.5. Correlation can be performed more efficiently in frequency domain (see next chapter), but distortion of the incoming signal by the exciser may deteriorate the correlation properties of the locally generated PRN code. Performing an identical excision function on the locally generated code may improve correlation.

An improvement in interference suppression capability may be achieved by looking at the time and frequency domains simultaneously (Amin, 1997). One method is the use of
Figure 6.5 Time/Frequency Excision Scheme Number 1

Figure 6.6 Time/Frequency Excision Scheme Number 2

Figure 6.7 Time/Frequency Excision Scheme Number 3
spectral estimators to drive adaptive time-domain filters. Another approach is to apply time-frequency domain transformations, such as the Gabor or Short Time Fourier Transform (SFTF) (Amin, Ouyang, & Lindsey, 1998), the wavelet transform, or wavelet packet transforms. Such interference suppression schemes are already widely used in the fields of speech processing, radar and sonar, but are currently gaining more popularity in the spread-spectrum communication community (Giannakis, 1999). Figures 6.8 and 6.9 illustrate the use of the STFT and the dyadic wavelet transform for interference suppression. STFT and wavelet transforms can be represented by a bank of filters and result in coefficients that represent the signal strength with a certain tiling of the time-frequency plane. Using the transform function, more or less, as a two-dimension filter, the regions contaminated with RF interference, such as the dashed regions, may be excised or assigned weighting coefficients, such that the inverse transform generates a signal with suppressed interference (Amin, 1997, and Hlawatsch & Kozek, 1994). The choice of transform is dependent on the desired layout of the time-frequency tiling based on the frequency band and time interval in which the interference source resides.

In dynamic environments RF interference may appear and disappear as a function of time due to the user's motion. The interference is referred to as non-stationary. In these cases, it would be preferable to change the time-frequency tiling to a configuration that minimizes the removal of desired signal energy. Such schemes include the wavelet packets transform (Coifman & Wickerhauser, 1992) and multiple over-determined tiling (Krongold, Kramer, Ramchandran, & Jones, 1997).
Figure 6.8 Interference Suppression using the STFT

Figure 6.9 Interference Suppression using the Wavelet Transform
Although joint time-frequency techniques may have attractive capabilities, the time and frequency domain of a signal are very much related. More dissimilar joint techniques, such as joint space and time domain interference suppression, may be more applicable for many problems (Moelker, 1998).
7. BLOCK PROCESSING TECHNIQUES

7.1 Background

Chapter 3 addressed the use of sequential processing techniques for GPS tracking and acquisition. Shortcomings, such as the inherent loss-of-information due to down-sampling, limited flexibility, a noise vs. dynamic tradeoff, and the non-linear behavior of the tracking loops, has made it worth while looking for alternatives. Availability of software receivers and continuous increase in processor speed have furthermore made it possible to investigate and test alternatives. One alternative is the use of block processing techniques.

Within the context of this dissertation, the definition of block processing is as follows:

*Any digital signal processing technique that operates on a block of samples*

Although general, this definition emphasizes operation on blocks of samples as opposed to operation on a sample by sample basis as used in sequential processing. The blocks of samples are generated by the front end and ADC of the software receiver, converted from a serial stream to a parallel stream, and stored in local memory. Examples of block processing, to acquire GPS signals, were introduced in chapter 3; the parallel code and frequency search algorithms are based on taking the Fast Fourier Transform (FFT) of a block of samples after down-conversion and de-spreading, respectively.

Within the context of this dissertation block processing techniques are subdivided
into three classes; time domain techniques, frequency domain techniques, and joint time-frequency domain techniques. The distinction is made based on the signal representation, referred to as the signal domain. Frequency domain methods require the use of a Discrete Fourier Transform (DFT) or its computationally more efficient equivalent, the Fast Fourier Transform (FFT). Joint time-frequency analysis transforms the signal to a domain in which the signal’s behavior in both time and frequency may be observed. Sections 7.2 through 7.3 will address these various techniques in more detail.

An important issue that arises from the application of block processing techniques, is the choice of the block length. The distinction is made between the size of the blocks stored in memory, $M$, and the size of the blocks that are operated on, $N$. These block sizes do not necessarily have to agree; one may store one second of data for average purposes, but only operate on blocks of 1 millisecond. Operation on longer blocks of data may result in a noise reduction due to the averaging effect, but makes it difficult to estimate frequency with a high resolution under dynamically challenging conditions. Shorter data blocks will enable the tracking of dynamics, but result in a poor frequency resolution when estimating exact frequency. Parallel investigation of the signals at different block lengths, may be a solution to all these tradeoffs. It will however require more computational power, and thus additional cost.

7.2 Time-Domain Techniques

Time domain techniques are those techniques that do not require the signal to be transformed to another domain, or signal representation. The incoming time-series is divided
into blocks of size $N$, (Figure 7.1) using a serial to parallel converter. Next, code-phase, carrier-frequency, and carrier-phase are estimated using these blocks. The availability of blocks of data does not limit the operation to causal filters. Block processors may well apply non-causal filters, such as zero-phase filters (Kormylo & Jain, 1974) allowing the design of filters which do not introduce phase delays that could have caused ranging errors in a causal implementation. The drawback of non-causal processing is the inherent delay of the output.

This dissertation focuses on block addition and block inspection time-domain methods. Both methods may be implemented in an iterative way. Alternative methods include the iterative conventional method described by (Gang Feng, 1999). This method applies the carrier and code tracking function from chapter 3 to blocks of samples, and introduces feedback by means of iteration. The advantage of this method is a better observability of the correlation peak form or characteristics.

The main goal of block addition techniques is to obtain extra processing gain by averaging the signal over a long period of time. Averaging is achieved by adding blocks of samples. This technique can be seen as the block processing equivalent of the accumulate and dump function in sequential processing. The principle behind block addition techniques is illustrated in Figure 7.2. An operator $f(\cdot)$ is defined and used on $M$ consecutive blocks. The result is added and used to estimate the desired parameters. The methods flexibility is defined by the content of operators $g(\cdot)$ and $f(\cdot)$. Chapter 8 will discuss an example of a block addition technique to track GPS signals in those environments where GPS signal levels are well below the nominal signal to noise ratio.
Figure 7.1 Division of Time-series in Blocks

\[ T_N \]

\[ 1 \quad 2 \quad 3 \quad \cdots \quad M-2 \quad M-1 \quad M \]

\[ \leftarrow N \rightarrow \]

\[ f(1) \]
\[ f(2) \]
\[ \vdots \]
\[ f(M-1) \]
\[ f(M) \]
\[ g(\overbrace{x}) \]

Figure 7.2 Block Addition Techniques
Block inspection methods are used in those cases where no extra processing gain is needed and the inherent low-pass filter characteristic of a block addition technique is not desired. A possible application for block inspection techniques would be the estimation of the dynamic behavior of the GPS signal. Block inspection is illustrated in Figure 7.3. Operator \( g(\cdot) \) and \( f(\cdot) \) can be defined based on the parameters that need estimation.

7.3 Frequency-Domain Techniques

Digital signals are transformed to the discrete-frequency domain by means of the Discrete Fourier Transform (DFT). Transforming the signal to the frequency domain may lower the computational burden of an operation or improve the signal's spectral observability. The DFT owes its popularity to the Fast Fourier Transform (FFT), a computationally efficient implementation of the DFT (Elliott & Ramamohan, 1982), and has been used for a variety of applications, especially in the field of spectral estimation (Marple, 1987, and Kay, 1988).

The DFT assumes the signal to be periodic, with a minimum periodicity equal to the block size (Oppenheim & Schafer, 1989). The frequency range of the DFT is fully determined by the Nyquist criterion. When operated on a block of samples, the DFT’s minimum resolution, \( r_{FFT} \), and frequency range are given by:

\[
 r_{FFT} = \frac{1}{\Delta t}, \quad f[k] \in (-f_s/2, f_s/2) \quad (7.1)
\]

where \( \Delta t \) is the block size in seconds and \( f_s \) is the sampling frequency. Equation 7.1 can be
interpreted as follows: the spectral lines range from minus half the sample rate to plus half
the sample rate and are separated by $r_{FFT}$ Hz. To obtain a better frequency resolution a
longer block-size is required. To extend the frequency range, the sampling rate needs to be
increased.

One of the major problems with the DFT is the time versus frequency tradeoff. Better
frequency resolution is achieved by choosing a larger block size. However, in the presence
of frequency changes, performing an FFT on longer block sizes, obscures the instantaneous
frequency behavior. The FFT results in the observation of a frequency band. This effect is
illustrated in Figure 7.4. For this reason, GPS does not allow an accurate observation of
instantaneous frequency using the FFT in its current implementation; the GPS signal can go
through large dynamics due to transmitter-receiver motion and oscillator effects. In the
literature, this time versus frequency tradeoff is indicated by Heisenberg's uncertainty
principle which states that the product of the variance around a point in time and the variance
around a point in frequency has a lower limit (Gabor, 1946). The next section will address
this principle in more detail.
(Coenen & van Nee, 1991) and (Davenport, 1991) noted that the correlation function can be expressed as a circular convolution. Circular convolution of two signals can be implemented efficiently by transforming both discrete-time signals to the frequency domain using a DFT, multiplying the transformed signals, and inverse DFT the result (Oppenheim & Schafer, 1989). In case of a GPS correlator, this operation can be expressed as follows:

\[
R[m] = \sum_{n=0}^{M-1} s[n] \cdot G_{Li}[n+m] \\
= s[n](\ast)G_{Li}[-n] \\
= IFFF\{FFT\{s[n]\}\ast FFT^\dagger\{G_{Li}[n]\}\} \\
\]

(7.2)

Figure 7.4 Time vs. Frequency Tradeoff in the FFT
The result of this efficient circular convolution implementation is the whole cross-correlation function sampled at the ADC sample rate, $f_s$. Figure 7.5 shows a block diagram of this transform domain correlator implementation based on FFT’s. For comparison, a conventional implementation is illustrated in Figure 7.6. The A&D operators are accumulate and dump operations. If $M$ is a power of two, both methods can be compared as follows: the implementation in Figure 7.5 requires about $\frac{3}{2} M \log_2 \frac{M}{2} + M$ complex multiplications and $3M \log_2 M$ complex additions whereas the implementation in Figure 7.6 requires $M^2$ complex multiplications and $M(M-1)$ complex additions. For $M = 2,048$ ($f_s = 2.048 \text{ Msps}$), the conventional correlator would use 4,194,304 multiplications and 4,192,256 additions, whereas the vector correlator would only require 35,840 ($0.8\%$ with respect to the conventional method) multiplications and 67,584 ($1.6\%$) additions. Mixed radix implementations of the FFT and IFFT algorithms may be necessary when $M$ is not a power of two. These implementations usually require more complex multiplications and additions than the radix-2 implementation (Elliott and Rao, 1982). The mixed radix FFT, which is used for the implementations in Chapter 8, uses 0.1203 seconds of computation time (benchmark CPU: 50MHz 486DX) for a 2,048 point FFT (radix-2) and 0.1294 seconds for a 2,000 point FFT (radix 5 and 2). A 5,000 point mixed radix FFT (radix 5 and 2) takes 0.3300 seconds. Implementation in ASICs and FPGAs significantly reduces the processing time.
Figure 7.5 Transform Domain Correlator

Figure 7.6 Massive Correlator Bank
The code-phase resolution at the output of the vector correlator is determined by the sample time, $T_s$. This resolution could be improved by interpolation methods, such as zero-padding and low-pass filtering (Strang & Nguyen, 1997), or fractional delay filters (Laakso, Välimäki, Karjalainen & Laine, 1996 and Coenen, 1998). Both methods have the disadvantage that temporal information is lost, because the filtering operation removes sharpness of the correlation peak. An alternative method uses the relationship $G_j(t-\delta t) \rightarrow \mathcal{F}\{G_j(t)\}e^{-j\omega \delta t}$. This method would be equivalent to adding extra correlators to the massive correlator bank and decreasing the spacing between local Gold code samples.

Thus far, the whole cross-correlation peak was estimated using a vector correlator. Time-of-transmission, however, is determined by the temporal location of the cross-correlation peak. For processing purposes the samples of the cross-correlation function that do not lay on the cross-correlation peak are not of interest and can be neglected, in other words, after detection of a cross-correlation peak, the vector correlator can zoom in on the peak samples. This way, the same hardware can be applied to a smaller temporal region, and result in a higher temporal resolution. This mechanism is similar to the feedback mechanism in the code-tracking loop.

7.4 Time-Frequency Domain techniques

As previously stated, the DFT does not provide good time and frequency resolution simultaneously. This section will deal with transformations that have better joint time-frequency characteristics. The theory of time-frequency analysis, although intuitively simple, is heavily built on mathematical concepts. The discussion here will be limited to the basics.
For more in-depth theory, the author refers to a variety of texts, such as (Mallat, 1998), (Vetterli, 1995), (Strang & Nguyen, 1997), (Qian & Chen, 1996), and (Burrus, Gopinath, & Guo, 1998).

In general, transformation to a new domain means projecting the current signal or signal block on a set of basis functions, \( \varphi_k[l] \):

\[
X[k] = T\{x[n]\} = \langle \varphi_k[l], x[l] \rangle = \sum_{l \in \mathbb{Z}} \varphi_k[l] x[l]
\]

where \( T\{\cdot\} \) is the transform operator and \( \langle \cdot, \cdot \rangle \) is the inner product of the signal \( x[l] \) and \( \varphi_k[l] \).

For example, in case of the discrete Fourier series, the Fourier coefficients, \( X[k] \), are given by the inner product of \( x[l] \) and \( \varphi_k[l] = (1/N)e^{j(2\pi nk/N)} \). In other words, the signal is projected onto sine and cosine basis functions; the signal is decomposed into a sum of sines and cosines with discrete frequencies based on the values of 'k'.

The short-time Fourier transform (STFT) and discrete wavelet transform (DWT) are two examples of basis function decompositions. Their respective basis functions are shown below:

\[
\text{STFT: } \varphi_{m,k}[l] = g[l-m]e^{-j2\pi k l/N}
\]

\[
\text{DWT: } \varphi_{m,k}[l] = \frac{1}{\sqrt{a^k}} \psi \left( \frac{l-m}{a^k} \right)
\]

where \( g[l] \) is a window function, such as a Gaussian or a rectangular window (Mallat,
1998), $\Psi[l]$ is a function referred to as the mother wavelet (Burrus, Gopinath & Guo, 1998), and $a$ is a scalar which is commonly assigned the value 2. Basis functions may have more than one index, as illustrated by Equation 7.4 a and b. The use of two indices with the SFTF and DWT indicates their two-dimensional time-frequency character.

The STFT is a windowed Fourier transform; the block of samples is multiplied by a window function and transformed to frequency domain using an FFT (Zelniker & Taylor, 1994). By moving the window over block of samples for a number of shift values, $m$, time-frequency behavior can be investigated. The DWT basis function not only shifts a window over the block samples, but also scales the window function. The resulting time-frequency tilings are shown in Figures 7.7 and 7.8, respectively. The STFT has a uniform time-frequency tile layout, whereas the DWT tiling layout has tiles with a large temporal support and narrow frequency support for low frequencies, and visa versa.

The capability of a basis function to observe time and frequency simultaneously is limited by the uncertainty principle (Gabor, 1946):

$$\sigma^2_t \sigma^2_\omega \geq \frac{1}{4}$$

(7.5)

where $\sigma^2_t$ and $\sigma^2_\omega$ are the variances around specific points $u$ and $\xi$ on the time and frequency axes, respectively, given by:
Figure 7.7 Short-Time Fourier Transform Time-Frequency Tiling

Figure 7.8 Discrete Wavelet Transform Time-Frequency Tiling
\[ \sigma_t^2 = \frac{1}{\| \varphi(t) \|^2} \int_{-\infty}^{\infty} (t-u)^2 |\varphi(t)|^2 dt \]  

(7.6)

and:

\[ \sigma_\omega^2 = \frac{1}{2\pi \| \varphi(t) \|^2} \int_{-\infty}^{\infty} (\omega-\xi)^2 |\mathcal{F}\{ \varphi(t) \}|^2 d\omega \]  

(7.7)

where \( \varphi(t) \) is an energy basis function. The time frequency localization of a basis function, defined by Equation 7.5, may be illustrated by a so-called Heisenberg box with dimensions \( \sigma_t^2 \) and \( \sigma_\omega^2 \). (Mallat, 1998). An example of a Heisenberg box and its dimensions is shown in Figure 7.9. The uncertainty principle states that the area of the Heisenberg box has a minimum area. Other examples of Heisenberg boxes are the time-frequency tiles in Figures 7.7 and 7.8. The uncertainty principle, furthermore, explains the time-frequency tradeoff problem with the DFT. The DFT basis functions, which are sines and cosines, have an infinite time localization. To satisfy the uncertainty principle the frequency localization must be infinite small or given by a Dirac pulse.

Rewriting the inner product of Equation 7.3 as a convolution, it can easily be shown that transformations can be implemented by banks of filters, where the filter transfer functions are directly related to the basis functions. Depending on the basis function, the filter bank may also include decimators and mixers. One class of filter banks is based on the
Figure 7.9 Heisenberg Box
basic element shown in Figures 7.10 and 7.11. \( H_0 \) and \( H_1 \) are low- and high-pass filters, respectively. Resolution in time- and frequency domain is controlled by the decimators (Oppenheim & Schafer, 1989 and Strang & Nguyen, 1997). The analysis element or a composite of analysis elements is used for transforming the signal to the new domain. The transformation coefficients are then given by the values, \( y_i[n] \). Inverse transformation is performed by the synthesis filter bank. To obtain a signal, \( \hat{x}[n] \), that is equal to the input signal, \( x[n] \), the filters, \( H_0, H_1, F_0 \), and \( F_1 \) must satisfy perfect reconstruction criteria (Strang & Nguyen, 1997). Examples of the use of analysis filters, for the STFT and the dyadic \((a = 2)\) DWT, are given in Figures 7.7 and 7.8.

Joint time-frequency domain techniques may be used as the operator function in the block addition and inspection methods of Section 7.2 for observations of the dynamic behavior of the GPS signal as a result of user-transmitter motion or local oscillator. Another application was discussed in Chapter 6, where interference was excised from the time-frequency plane.

### 7.5 Summary

Applying techniques to blocks of data opens a whole world of possible digital signal processing. Block addition techniques may be used to get a large gain in these environments where a GPS user is experiencing low signal-to-noise ratios. Block inspection techniques may be used where the signal power is nominal, but where other conditions such as a high dynamic environment denies the user GPS capability.

The largest drawback of block processing is its inherent delay of the processing
Figure 7.10 Filter Bank Analysis Element

Figure 7.11 Filter Bank Synthesis Element
result. However, integration of GPS block processors with an IMU may overcome this limitation by coasting during the block processing latency (Uijt de Haag, 1999a). This integration requires insight in the short term behavior of the IMU. Another disadvantage is the computational intensity of the methods. Developments of fast processors and logic, such as FPGAs, may overcome this problem in the near future.
8. GPS BLOCK PROCESSOR - CASE STUDIES

8.1 Background

This chapter discusses several examples of the application of block-processing techniques to GPS. First, tracking GPS signals at low carrier-to-noise ratios will be addressed. Then, a method is developed for observation of the time-frequency behavior of the GPS signal. Finally examples are given of narrow-band and pulsed interference suppression using frequency excision methods.

To test the proposed block processing concepts, GPS time series were generated using both a software signal simulator and real data. The GPS software signal simulator provides one or two satellites, simultaneously, for a variety of signal levels. A more detailed description of this simulator can be found in Appendix A. Simulating a GPS signal in software allows the user to put in a variety of Doppler profiles as well as RF interference sources.

To obtain data from both real GPS satellites and a GPS satellite simulator output, the test setup in Figure 8.1 was used. In this setup the GPS signal is down-converted to an intermediate frequency of 21.25 MHz and sampled at 5.0 megasamples per second (5.0 Msps), simultaneously, using bandpass sampling (Akos, 1997). The extensive RF network shown in Figure 8.1 is necessary to suppress out-of-band signals, but will be replaced by a simpler network in future implementations. The input of the receiver front may be connected to a GPS antenna or a GPS satellite simulator.
Figure 8.1 Software Receiver Front End test Setup
8.1 Case I: Detection of GPS Signals at Low CN0

There are many circumstances in which the desired GPS signal has relative low signal strength. In these cases the GPS signal is said to have a low carrier-to-noise ratio or $C/N_0$. Some examples are tracking of GPS in an urban environment or an environment with a high wide-band RF interference level, indoor tracking, and tracking under foliage. In an urban area GPS performance may be deteriorated or completely lost due to limited satellite visibility. Under these circumstances, traditional receivers loose lock regularly, requiring re-acquisition of the lost satellite. Usually, urban GPS is augmented with wheel-sensors or databases to obtain the required availability of positioning. Inside buildings the signal strength is typically 30 dB less (Moeglein & Krasner, 1998) than outside a building due to the attenuation of the building’s superstructure. Because conventional receivers do not acquire signals beyond a 5-10 dB attenuation, alternatives need to be sought. (Parkinson & Spilker, 1996) indicate that the attenuation due to foliage is dependent on the satellite’s elevation and can be as high as 27 dB. Even for trees without foliage the attenuation is roughly only 35% less than the attenuation of a tree in full foliage. Thus, deep forest survey applications or expeditions need to come up with alternative measuring methods to receive signals at the desired strengths, such as antennas on extension poles. Even for mobile users driving along trees, fades as deep as 25 dB may occur. These fades could cause the receiver to loose lock.

Block processing techniques may provide the gain necessary to raise the signal to useful levels by means of averaging. An example of such a technique is the SnapTrack server-aided GPS system architecture and DSP software-based receiver (Moeglein &
The disadvantage of this technique is that it requires an extensive differential infrastructure to transmit estimates of the satellite’s Doppler. This section describes an alternative method based on the block addition principle described in the previous chapter. The algorithm is based on averaging the signal over a long period of time and the availability of a priori navigation information, time estimates, and frequency estimates.

The model of the received GPS signal used for the mathematical formulation given in this dissertation is:

$$z[k] = s[k] + n[k] = AG_f[k]d_L[k]\cos(\omega_L k T_s + \phi_L) + n[k]$$  \hspace{1cm} (8.1)

where $G_f[k]$ is the Pseudo Random Noise (PRN) code (at chipping rate $R_c$), $d_L[k]$ is the navigation data, $\omega_L$ is the intermediate frequency, $\phi_L$ is the fractional carrier phase, and $n[k]$ is Additive White Gaussian Noise (AWGN). $k$ is the time index and $T_s$ is the sample period. A block of $N$ samples is represented by a vector $z$, whose elements are $z[k]$. In the proposed algorithm the sample frequency was chosen to be 5.0 Msps and the minimum block was set to one code period or $N = 5,000$ samples. If the Doppler shift equals 0 Hz, every code-period (= 1 ms) contains 5,000 samples. Presence of frequency variations due to the user motion and local oscillators causes the number of samples per code-period to be a non-integer number. It is the block processor’s function to estimate $G_f[k]$, $d_L[k]$, $\omega_L$, and $\phi_L$ in Equation 8.1, while minimizing the error due to the AWGN, $n[k]$. Partial a priori knowledge of the navigation data bits and an estimate of the Doppler frequency from, for example, an Inertial Measurement Unit (IMU) (Uijt de Haag, 1999a-b), limits the estimation
process to the fractional carrier phase and the code-phase of $G_j[k]$.

The basic element that is used to estimate the code-phase of $G_j[k]$, is the transform domain correlator which was discussed in Chapter 7. Correlation of the incoming signal with a locally generated pseudo random noise (PRN) code may be expressed by a circular convolution in time domain. A computational effective method of implementing the circular convolution is via the frequency domain or Fast Fourier Transform (FFT):

$$R[m] = \text{IFFT}\{\text{FFT}\{s[k]\} \cdot \text{FFT}^*\{G_j[k]\}\}$$

The summation in Equation 8.2 may be compared to the cross-correlator in the traditional receiver. In this case the pre-detection integration time would be one code-period or $N$ samples. In a traditional receiver the correlator is followed by a discriminator and a feedback loop to obtain an accurate estimate of the code-phase (Chapter 3). When the transform domain correlator is used, the result is an $N$-dimensional vector, $R$, containing the cross-correlation function sampled at the sampling rate of 5 MHz. To obtain accurate estimates of the code-phase, interpolators or tracking schemes need to be designed. A block processing approach to such a tracking scheme is given by (Feng, 1999).

* A priori* knowledge of the intermediate frequency and the navigation data bits, as discussed in the previous section, may be used to generate the desired cross-correlation of the incoming signal with a locally generated copy of that signal for one block of samples. This process forms the basic element of the proposed algorithm and is illustrated in Figure 8.2.
Figure 8.2 Basic Element (BE) of the GPS Block Processor
The received signal block, \( z \), is in-phase and quadrature down-converted by a locally generated carrier, based on the frequency and fractional carrier-phase estimates. The resulting signal block, \( r \), is then cross-correlated with a locally generated PRN code using the transform domain correlator (TD correlator) of Chapter 7. Finally, the cross-correlation function is multiplied by the estimated data-bit to remove the sensitivity to the navigation data bit sign. This enables the processor to integrate across data bit boundaries.

To obtain signal gain, the cross-correlation results of \( M \) consecutive blocks are added as shown in Figure 8.3. Addition of the \( M \) resulting cross-correlation functions may result in a noise average, and thus a relative noise reduction. The signal gain, thus obtained, may enable detection of GPS signals under low carrier-to-noise ratio conditions.

The principle of operation, shown in Figure 8.3, can mathematically be illustrated as follows. Down conversion of the received signal of Equation 8.1 yields the following discrete-time signal, which includes narrow-band Gaussian Noise:

\[
\begin{align*}
    r[k] &= s[k] + n[k] \\
    &= AG_j[k]D[k]\cos(\omega_e kT_s + \varphi_e) + \\
    &\quad n_i[k]\cos(\omega_e kT_s + \varphi_e) + n_q[k]\sin(\omega_e kT_s + \varphi_e) \\
\end{align*}
\]

(8.3)

where \( \omega_e \) is the residual frequency \( (\omega_e = \omega_L - \hat{\omega}_L) \) and \( \varphi_e \) is the residual phase \( (\varphi_e = \varphi_L - \hat{\varphi}_L) \) after down-conversion. The block of samples, \( r[k] \), is given by \( N \times 1 \) vector \( r \). Under the assumptions that all navigation data bits are known, the Doppler effect on the carrier is constant, and the effect of the Doppler effect on the code-rate can be neglected, the cross-correlation of \( r[k] \) with the locally generated pseudo-random noise code, \( G_i[k] \), for
Figure 8.3 Block Adder (BA) Element of the GPS Block Processor
the $j^{th}$ SV, may be expressed as follows:

$$R[m] = A \sum_{k=0}^{N-1} G_j[k]G_{l,j}[k+m]\cos(\omega_\epsilon kT_s + \phi_\epsilon) +$$

$$\sum_{k=0}^{N-1} n_j[k]G_{l,j}[k+m]\cos(\omega_\epsilon kT_s + \phi_\epsilon) +$$

$$\sum_{k=0}^{N-1} n_q[k]G_{l,j}[k+m]\sin(\omega_\epsilon kT_s + \phi_\epsilon)$$

$$= R_s[m] + R_n[m]$$

(8.4)

$R[m]$ equal the elements of the $R$ vector at the output of the basic processing element in Figure 8.2. The cross-correlation function consists of a signal ($R_s[m]$) and noise ($R_n[m]$) term. Isolating the signal term in Equation 8.4, and realizing that $G_{l,j}[k+IN] = G_{l,j}[k]$ due to periodicity, addition of the $M$ down-converted and cross-correlated blocks may be described as follows:

$$R^a[m] = \sum_{k=0}^{M-1} R_s[m+kM] + \sum_{k=0}^{M-1} R_n[m+kM]$$

$$= R^a_s[m] + R^a_n[m]$$

(8.5)

where

$$R^a_s[m] = A \sum_{k=0}^{N-1} G_j[k]G_{l,j}[k+m]\cos(\omega_\epsilon kT_s + \phi_\epsilon) +$$

$$A \sum_{k=0}^{N-1} G_j[k]G_{l,j}[k+m]\cos(\omega_\epsilon (k+N)T_s + \phi_\epsilon) +$$

$$\vdots$$

$$A \sum_{k=0}^{N-1} G_j[k]G_{l,j}[k+m]\cos(\omega_\epsilon (k+(M-1)N)T_s + \phi_\epsilon)$$

(8.6)
If it is furthermore assumed that the change in carrier frequency over one code period is negligible, or $T_G \ll 1/f_c$, the C/A code cross-correlation function can be isolated from the cosine sum term, and Equation 8.6 may be simplified as follows:

$$R^a_s[m] = AR^a_C[m] \sum_{k=0}^{M-1} \cos(\omega_c kT_G + \phi_c)$$

For a zero residual frequency and phase, the summation in Equation 8.7 will equal $M$, and a full addition gain of $10\log_{10} M$ is obtained. For certain values of the residual frequency, $\omega_c$, however, the summation in Equation 8.7 goes to zero, resulting in a total signal power loss. This phenomenon is referred to as destructive self-interference. A relationship between the block addition gain and the residual carrier frequency and phase may be obtained by expanding the summation in Equation 8.7 using trigonometric identities. Approximation of the summations by their continuous-time integrations, then leaves:

$$\sum_{l=0}^{M-1} \cos(\omega_c lT_G + \phi_c) = \cos \phi_c \sum_{l=0}^{M-1} \cos(\omega_c lT_G) - \sin \phi_c \sum_{l=0}^{M-1} \sin(\omega_c lT_G)$$

$$= M \cos \phi_c \int_0^{(M-1)T_G} \cos(\omega_c t)dt - M \sin \phi_c \int_0^{(M-1)T_G} \sin(\omega_c t)dt$$

Evaluating the integrals in Equation 8.8 and substituting the result in Equation 8.7, yields the following expression for the cross-correlation after adding $M$ blocks:
The impact of Equation 8.9 is illustrated in Figure 8.4. The destructive interference is illustrated by the occurrence of nulls at frequencies equal to a multiple of the inverse of the block-size, \(1/M T_G\) Hz. So, longer average times necessitate more precise estimates of the carrier-frequency provided by the IMU. Under stationary conditions on the surface of the earth, the change in carrier Doppler frequency does not exceed 1 Hz per second. In those cases where large Doppler changes are expected, however, integration with IMU’s or other motion sensors will be necessary.

The impact of a change in the code-chipping rate due to the Doppler effect on the described block addition technique is twofold; loss of cross-correlation power due to a discrepancy between the chipping rate of the local and incoming C/A code and limitation of the block addition gain due to the migration of the correlation peak with respect to time. The former effect is small for realistic values of the carrier Doppler \((-10,000\,\text{Hz} < f_d < 10,000\,\text{Hz})\) and is given by the following equations:

\[
\begin{align*}
&\gamma > 1 : \Delta E_{\text{loss}} = \frac{(1-1/\gamma)}{1.023} \sum_{k=0}^{1.023} k (d_k d_{k+1} - 1) \% \\
&\gamma < 1 : \Delta E_{\text{loss}} = \frac{(1/\gamma - 1)}{1.023} \sum_{k=0}^{1.023} (k-1)(d_k d_{k-1} - 1) \% 
\end{align*}
\]  

\((8.10)\)
Figure 8.4 Fading due to the Residual Carrier Frequency
where $\gamma$ is the Doppler scale factor and $g_k$ is the $k^{th}$ bit of the C/A code. For example, a Doppler frequency of 1,000 Hz results in a power loss, due to chipping rate mis-match, equal to 0.21%. Note, that the sample Doppler, described in Section 5.4.2, can be introduced in Equation 8.10, also, by replacing $\gamma$ by $\beta \gamma$.

Due to the Doppler effect, the effective code period is given by $(1/\gamma) T_c$. $\gamma$ is the Doppler scale factor defined in Chapter 4. Consequently, the temporal location of the cross-correlation peak within each of the $M$ consecutive blocks will change as a function of time. Thus, addition of multiple blocks will result in a limited desired gain. This effect is illustrated in Figures 8.5 and 8.6. Under Doppler free conditions, as is illustrated in Figure 8.5, addition of the cross-correlation peaks results in a maximum gain, $G = 10 \log_{10} M$. The presence of a Doppler shift causes the peaks to migrate along the locally defined time-axis and addition will result in a trapezoidal function as shown in Figure 8.6. This problem may be solved by using fractional delay (FD) filters (Laakso, Välimäki, Karjalainen, 1996 and Coenen, 1998) to delay or advance the cross-correlation peak. In these cases, approximate knowledge of the Doppler is required. Figure 8.3 shows the location of the FD filters within the BA element. To illustrate the effect of cross-correlation peak migration, the processing element of Figure 8.3 was applied to real data with a $C/N_0$ of 45 dB-Hz for $M = 1,000$ consecutive blocks of 5,000 samples (= 1 second). The results are shown in Figure 8.7. One can clearly see that the peaks seem to migrate along the code-phase axis, due to a decreased duration of the code-period ($T_g/\gamma$ with $\gamma > 1$).
Figure 8.5 Block Addition in the Absence of Code Doppler

Figure 8.6 Block Addition in the Presence of Code Doppler
Figure 8.7 Illustration of the Code Doppler Effect
The other assumption made in the derivation of Equation 8.5 is a priori knowledge of the navigation data bits. In the traditional receiver, lack of knowledge of the navigation data bit signs and transitions, limits the pre-detection integration time to the duration of one data bit duration or 20 ms, and requires the receiver to synchronize with the data bits. The proposed method utilizes the fact that the navigation data bits are not entirely unknown. This is illustrated in Figure 8.8 where the hand over word (HOW) and telemetry word (TLM) of the navigation data are shown. The HOW and TLM words are received once every 6 seconds, and, under normal operation, eight preamble bits, three subframe identification bits, and three flag bits are known. Dependency on the last bit of the previous word as specified by the parity check may be neglected if one realizes that for the block-addition method to work, the sign of the total does not matter. An additional 17 bits may be known if an a priori estimate of time is available from a local clock, such as the CPU clock. Knowledge of the time within 3 seconds makes it possible to estimate the truncated Z-count, which 17 bits form the first 17 bits of the HOW.

To use the a priori navigation information optimally, it needs to be correlated with the navigation data modulated on to the correlation peaks. This operation will increase the complexity of the block addition technique. A disadvantage of cross-correlation with the navigation data is the data's poor cross-correlation properties. However, cross-correlation may still provide the necessary information: the C/A code cross-correlation peak.
<table>
<thead>
<tr>
<th>Preamble</th>
<th>TLM Message</th>
<th>Parity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 0 0 0 0 1 0 1 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 8.8 TLM and HOW Word Structures

TOW Truncated Z-count | Flags | Subframe ID | Parity | Zeros
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

30 Bit TLM Word

30 Bit HOW Word
Figure 8.9 shows the final implementation of the block processor. Blocks are read into a buffer and rotated into a shift register consisting of \( 'M' \) \( N \)-size block buffers. After each shift operation the processing element of Figure 8.3 is executed and the cross-correlation peak is inspected by looking at the square of the norm of the complex cross-correlation output. When one of the cross-correlation samples exceeds a preset threshold value, a signal is possibly detected. At that point the \textit{a priori} navigation information lines up with the navigation data bits of the received data.

Neglected so far are the noise terms in Equation 8.6. Block addition of these noise terms yields:

\[
R_n^a[m] = \sum_{k=0}^{N-1} G_{l_j}[k+m] \sum_{l=0}^{M-1} \left\{ n_j[k+lN] \cos(\omega k T_s + \varphi) + n_q[k+lN] \sin(\omega k T_s + \varphi) \right\}
\]  

(8.11)

Equation 8.11 can not easily be rewritten in the same form as Equation 8.6. However, for \( f_\epsilon = 0 \) and \( \varphi_\epsilon = 0 \), Equation 8.11 reduces to:

\[
R_n^a[m] = \sum_{k=0}^{N-1} G_{l_j}[k+m] \sum_{l=0}^{M-1} n_j[k+lN]
\]

\[
= \sum_{k=0}^{N-1} G_{l_j}[k+m] \hat{n}_i[k]
\]

(8.12)

where \( \hat{n}_i[k] \) is Gaussian with \( \bar{\sigma}_i = \sqrt{M} \sigma_i \) and \( \bar{\mu}_i = \mu_i \) (Papoulis, 1991). Note, that the samples \( R_n^a[m] \) are correlated among each other according to:
Figure 8.9 Logic Implementation for One Carrier Frequency
Another neglected effect that will need further attention is the presence of oscillator variations described in Chapter 5. Oscillator variations cause the residual frequency and phase to be a function of time as can be seen from Equation 8.11. Consequently, the noise component of the received signal is no longer stationary, which may cause the block addition technique to be less efficient due to a loss in signal gain. This non-stationarity will, furthermore, complicate the statistical analysis of the block addition technique.

To illustrate how the proposed block addition technique performs, a variety of experiments were performed. First, the software simulator described in appendix A was used to generate a GPS signal for satellite 17 with additional Gaussian noise and a 900 Hz Doppler shift. A priori, 16 navigation data bits were known. This a priori knowledge results in a signal gain equal to \( G = 10 \log_{10}(16 \times 20) = 25 \, \text{dB} \). Figures 8.10 and 8.11 show the outputs of the GPS block processor of Figure 8.9 for simulated signal strengths of 44 and 22 dB-Hz, respectively. Therefore, Figure 8.10 and 8.11 show the maximum of the cross-correlation function, \( R^a \), for all specified frequency offset - code-period combinations.
Figure 8.10 Navigation Correlation Characteristic as a Function of Frequency and Code Period (CN0 = 44 dB-Hz)

Figure 8.11 Navigation Correlation Characteristic as a Function of Frequency and Code Period (CN0 = 22 dB-Hz)
The x-axis shows the frequency offset, $\hat{f}_l$, from the IF, and the y-axis shows the number of block or code-period clock cycles used by the shift register in Figure 8.9. This 3 dimensional plot will be referred to as the navigation cross-correlation. After 121 code-periods a strong response can be observed. This response corresponds to the alignment of the local a priori navigation data bits and the received navigation data bits modulated onto the cross-correlation peaks. One may also observe numerous local cross-correlation maxima. These maxima correspond to partial cross-correlation and constructive self-interference. Figures 8.12 and 8.13 show the C/A code cross-correlation functions which correspond to the strongest maximum cross-correlation response over all code-period shifts for each of the predefined frequency offsets.

A similar experiment was performed on a simulated satellite with a $C/N_0$ of 17 dB-Hz. At this $C/N_0$, a priori knowledge of 40 bits was necessary to achieve the necessary cross-correlation peak amplification. 40 navigation data bits will provide $G = 10 \log_{10}(40 \times 20) = 29 dB$ gain. Figure 8.14 shows the resulting navigation cross-correlation characteristic and Figure 8.15 shows the output of the block processor after each cycle of the processor’s shift register. As can be seen, even at a signal level this low the block processor successfully detects the signal.

The setup of Figure 8.1 was also connected to a GPS signal simulator. For satellite 6 with a $C/N_0$ in the high forties and a Doppler of about -750 Hz, the navigation and C/A cross-correlation graphs are shown in Figures 8.16 and 8.17, respectively. The 8 preamble bits were chosen as the a priori known data bits. The expected ideal cross-correlation gain would therefore be $G = 10 \log_{10}(8 \times 20) = 22 dB$. The data set duration was chosen to be
6 seconds and, therefore, included at least one preamble. This preamble can be clearly observed in the Figure 8.16. No fractional delay filters were used to compensate for code Doppler. The detection performance could be improved when these filters would be applied.
Figure 8.12 C/A Code Cross-Correlation as a Function of Frequency and Code Period  

(CN0 = 44 dB-Hz)

Figure 8.13 C/A Code Cross-Correlation as a Function of Frequency and Code Period  

(CN0 = 22 dB-Hz)
Figure 8.14 Navigation Cross-Correlation as a Function of Frequency Offset and Code Periods for (CN0 = 17 dB-Hz)

Figure 8.15 C/A Code Cross-Correlation as a Function of Frequency Offset and Code Period (CN0 = 17 dB-Hz)
Figure 8.16 Navigation Cross-Correlation as a Function of Frequency Offset and Code Periods for Averaging over One Preamble

Figure 8.17 C/A Code Cross-Correlation as a Function of Frequency Offset and Code Periods for Averaging over One Preamble
8.2 Case II: High Dynamic Tracking

The dynamic behavior of the GPS signal due to line-of-sight velocity and oscillator effects, is reflected in a change of the carrier frequency and code-rate. Estimating the frequency content has been subject to many applications, such as speech-processing and sonar. Section 7.4 discussed the observation of joint time-frequency behavior of the signal. In this section a block inspection method is described that utilizes both the Short-Time Fourier Transform (STFT) and interpolation techniques to obtain insight in the dynamic behavior of the GPS signal. These methods enable the user to observe changes in frequency that conventional sequential methods could not achieve.

The proposed method is illustrated in Figure 8.18. An initial frequency and code-phase estimate is obtained from the parallel code-phase search method described in Section 3.3. This phase will be referred to as the coarse zoom function. If a signal is detected, the algorithm zooms in on the detected signal by increasing the frequency resolution to 50 Hz bins. Both coarse and fine zoom functions can be described by the block diagram in Figure 8.19. This figure shows a bank of in-phase and quadrature mixers in combination with vector correlators. \( \omega_c \) is the center frequency of the filter bank, \( n \) determines the discrete range of frequencies, and \( \omega_\Delta \) specifies the frequency bin-widths or resolution. In case of the coarse zoom function, \( \omega_c \) equals the IF of the digitized GPS signal, \( n \) equals 8, and \( \omega_\Delta \) equals 500 Hz. Consequently, the frequency range covered by the coarse zoom function is \(-4,000 \, \text{Hz} < f_d < 4,000 \, \text{Hz}\). A refined frequency estimate can be obtained by setting \( \omega_c \) to the coarse estimate, increasing \( n \) to 20, and decreasing \( \omega_\Delta \) to 50 Hz. The result of the fine
Figure 8.18 GPS Time-Frequency Analysis Block Diagram
Figure 8.19 Dynamic Tracking Zoom Function (Coarse and Fine)
zoom is a frequency estimate within ±50 Hz and a code-phase estimate within ±1 sample. Using this frequency estimate, the correlation function can be computed for any number, $M$, of code-periods. The described implementation limits $M$ to 1,000 or 1 second. If the resultant cross-correlation were continuous, the expression for each of the $M$ cross-correlation peaks could be derived from Equation 3.5:

\[
\begin{align*}
I_{\text{peak}}[k] &= R_0[k] \cos(2\pi f_c k T_I + \phi_0') \text{sinc}(f_c T_{PDI}) \\
Q_{\text{peak}}[k] &= R_0[k] \sin(2\pi f_c k T_I + \phi_0') \text{sinc}(f_c T_{PDI})
\end{align*}
\]

(8.13)

where $k = 0, \ldots, N-1$, $T_I = T_G = 1 \text{ ms}$ is the integration period, and $f_c$ is the offset from the true carrier frequency. The nomenclature, $T_I$, is used instead of $T_{PDI}$ to distinguish between the integration time used for sequential processing and the integration time used for the described block inspection method.

The cross-correlation peak value is estimated by choosing the maximum value of the cross-correlation function for each code-period. Thus, two new time series can be generated, both sampled at 1,000 Hz; cross-correlation amplitude variation versus time and code-phase estimates as a function of time. This method of deriving cross-correlation peak amplitude/time series is very sensitive to noise, oscillator jitter, and the natural residual frequency (given by frequency $f_c$), and results in time series with much spike-like noise.

The residual frequency may be estimated from the amplitude time-series, whereas the code-rate may be determined from the code-phase time series. The amplitude variation is input to a STFT algorithm using a ‘Gaussian’ window function and a window separation of
one sample. Performing a ridge extraction algorithm on the result (Delprat, et al., 1992), and using a polynomial fit on the resulting ridge, gives a good estimate of the change in frequency over time. Change in code-rate is estimated directly from the code-phase time series by performing a polynomial fit. The presence of spike-like noise, however, deteriorates the outcome of the polynomial fit. To remove the jitter, a median filter was applied. Median filters outperform low-pass filters for noise with a spike-like behavior (Schalkoff, 1989).

The algorithm was tested on real data collected with the test setup in Figure 8.1. Satellite 7 with an indicated signal strength of about 46 dB-Hz was selected. Figure 8.20 shows the acquisition grid; the initial estimates of the carrier frequency and code-phase are 2,500 Hz and 441 samples respectively. Next, the algorithm zooms in on the correlation peak with a 50 Hz resolution, the results are given in Figure 8.21. One can clearly observe the effect of the ‘sinc’ function in Equation 8.13. Nulls appear for \( f_c T_{PDI} = \pm 1 \Rightarrow f_c = \pm 1,000 \text{ Hz} \). The refined carrier-frequency estimate is given by 2,350 Hz. Using this estimate to down-convert and cross-correlate all 1,000 code-periods, Figures 8.22 and 8.23 can be generated for the in-phase and quadrature components, respectively. These figures clearly illustrate the variation of the cross-correlation peak amplitude and the progression of the cross-correlation peak along the code-phase axis due to the Doppler and oscillator effects on the code-rate.

Plots of the cross-correlation amplitude variation and code-phase as a function of time are shown in Figures 8.24 and 8.25. The described spike-like jitter is apparent in Figure 8.21. Applying the SFTF to the amplitude variation series, results in the observation of the
Figure 8.20 Acquisition Grid for Satellite 7

Figure 8.21 Zoomed In Acquisition Grid
Figure 8.22 Real Cross-Correlation Peak Precession over 1 Second

Figure 8.23 Imaginary Cross-Correlation Peak Precession over 1 Second
Figure 8.24 Peak Amplitude Variation as a Function of Time
Figure 8.25 Code-Phase Variation as a Function of Time
change in frequency over time. Figure 8.26 shows the direct results. The 180 degrees phase
shifts of the navigation bits are clearly present. Squaring the amplitude variation series and
repeating the SFTF yields Figure 8.27. The navigation bit transitions are no longer visible,
but the carrier frequency offset is doubled. When finally applying ridge extracting
algorithms to the SFTF (figures 8.28 and 8.29), a more accurate estimate of the carrier-
frequency can be obtained. The Doppler shift is estimated to be equal to 2,387.5 Hz,
compared to a 2,521.84 Hz Doppler shift given by the Novatel sequential GPS receiver.

The code-phase time series can be used to estimate the code-rate due to Doppler and
oscillator effects. The results of median filtering and a polynomial fit is an estimated
residual code-rate of 18.4 sps (initial code-phase: 441.387). The Doppler shift found above,
would have resulted in a code-rate difference of 7.5 samples per second. The discrepancy
can be blamed on the presence of oscillator variation (see Chapter 5).

The described method is robust in the sense that large changes in frequency can be
observed. This makes the method suitable for high dynamic tracking. An example was
implemented using the software GPS simulator (Appendix A); Satellite 17 was given a 400
Hz/s Doppler change. The results are given in figures 8.30 and 8.31. The algorithm was able
to visualize the frequency change. Due to the limited number of available samples per
second (1,000) a maximum frequency of 500 Hz can be observed. Consequently, both Figure
8.30 and Figure 8.31 show folding of the signal. To extend the frequency range, the $M$
cross-correlation functions, the peak detection, and the STFT could be repeated for other
center frequencies, $f_c$. 
Figure 8.26 Time-Frequency Tiling for Satellite 7

Figure 8.27 T-F Tiling for Real Squared Cross-Correlation Peak Variation
Figure 8.28 Ridge Algorithm Results for Real Correlation Amplitude Variations

Figure 8.29 Ridge Algorithm Results for Real Squared Correlation Amplitude Variations
Figure 8.30 T-F Tiling for a GPS Signal with 400 Hz/s Doppler Change

Figure 8.31 T-F Tiling of a Squared GPS Signal with 400 Hz/s Doppler
A drawback of the proposed algorithm is the nominal signal strength requirement. A block addition technique, such as described in Section 8.2 could be used to add gain. Consequently, less samples per second would be available, and the frequency range would be reduced. To obtain the same frequency range as before, the hardware could be increased. Other interesting applications of the described method are Doppler crossover detection and oscillator offset and drift estimation.

8.3 Case III: Interference Suppression

Chapter 6 described a variety of methods to detect and suppress RF interference. This section will give two cases in which transform domain techniques were utilized to excise narrow-band and pulsed interference. The general block diagram for transform domain interference suppression techniques is shown in Figure 6.5. The choice of transform depends on the present interference. The DFT will perform sufficiently well suppressing narrow-band interference, but will result in unnecessary signal loss in the presence of pulsed interference, because this form of interference is well localized in time and frequency, simultaneously.

The GPS software signal simulator of Appendix A was used to generate GPS signals with narrow-band interference. A sine wave was added to the satellite signal with an amplitude equal to half the dynamic range of the ADC. The resultant signal was transformed to the frequency domain using an FFT, where the sine wave was detected. Next, the FFT coefficients that exceeded a pre-defined threshold were set to zero and the signal was transformed back to time domain. The threshold level can be based on the noise-floor and
the dynamic range of the ADC. The results of the frequency excision scheme were compared to the results with satellite 17 without NBI. Figures 8.32 through 8.38 show the results. In the presence of CW interference the acquisition search grid does not show any visible spikes due to the spreading of the sine wave's energy over all code-frequency bins. Conversion to frequency domain (Figure 8.35) clearly shows spikes at the CW frequency. Setting these excessive FFT coefficients to zero, inverse transforming the result, and performing a parallel code-phase search yields an acquisition grid that shows the cross-correlation peak at its original location. To compare the cross-correlation peak with the interference-free peak, the difference is shown in Figure 8.39. The error is a result of deteriorated cross-correlation properties of the GPS signal, because the excision function not only removes interference but also GPS spectral lines (Chapter 2). The worst-case scenario would be a narrow-band interference source at the C/A code's strongest spectral line. For satellite 17 this line is located at the IF plus 139 kHz. Repeating discussed procedure for a CW source at this frequency, results in an error (Figure 8.40) significantly larger than the error in Figure 8.39.

In the discussed example, the amplitude level of the NBI source was relatively small. When this level exceeds the dynamic range of the ADC, it will be impossible to suppress NBI using frequency domain techniques without removing most of the desired signal; most of the desired signal is clipped due to ADC hard limiting. The NBI can, however, still be detected and located in frequency using frequency domain techniques. An alternative detection scheme would be to look at the signal amplitude statistics, or use an adaptive ADC.
Figure 8.32 Acquisition Grid for Satellite 17 without NBI
Figure 8.33 Cross-Correlation Peak for Satellite 17 without NBI
Figure 8.34 Acquisition Grid for Satellite 17 with NBI (2 Hz)
Figure 8.35 Fast Fourier Transform of Block 1 of Satellite 17 with NBI

Figure 8.36 FFT of Block 1 after Nulling the NBI Source
Figure 8.37 Acquisition Grid for Satellite 17 after NBI Excision

Figure 8.38 Cross-Correlation Peak of Satellite 17 after NBI Excision
Figure 8.39 Error between Interference Free and NBI Contaminated Cross-Correlation Peak (after excision)

Figure 8.40 Same as Figure 8.39 but now with NBI at 138kHz
To detect and excise low strength pulsed interference, a joint time-frequency transform was chosen. Again, a signal was generated and pulsed inference with a pulse width of 41 μs was added. Furthermore, the pulse carrier frequency was chirp-like (Mallat, 1998); the frequency changes from $f_{IF}$ to $f_{IF} + 20$ Hz. The signal was transformed using Daubechies wavelets as basis functions. These wavelets and their associated filters are described in (Strang & Nguyen, 1997).

The results are illustrated in Figures 8.41 through 8.46. Figure 8.43 shows the time frequency localization of the interference source. After excision some pulse interference is still present as can be seen in Figure 8.46. Figure 8.43 illustrates the power of time-frequency excision, as compared to frequency domain excision, for the case of pulsed interference. Both frequency and time domain techniques would have removed significantly more power than the discussed method.

Normally, the pulsed interference pulse contains much more energy than used in the example. In these cases adaptive ADC techniques, such as the one described in Chapter 6, could be used in combination with time-frequency analysis.
Figure 8.41 ADC Output in the Presence of Pulsed Interference

Figure 8.42 Acquisition Grid in the Presence of Chirp-like Pulse Interference
\[ \log_2 \left( \frac{f}{2441} \right) \]

Figure 8.43 T-F Behavior of Signal plus Chirp-like Pulsed Interference
Figure 8.44 ADC Output after Excision Operation using the DWT

Figure 8.45 Residual Error after Pulsed Interference Excision using the DWT
Figure 8.46 Acquisition Grid after Interference Excision using the DWT
9. SUMMARY, CONCLUSIONS AND RECOMMENDATIONS

This dissertation has shown that block processing techniques are a viable alternative to sequential techniques. Sequential techniques were investigated and it was shown that they suffer from an inherent loss of information, limited flexibility, non-linear behavior, a causality restriction, and a noise versus dynamic tracking error tradeoff. A sequential receiver was implemented to illustrate above effects. A general block processing approach was introduced, and three case studies were performed on both simulated and real GPS data.

The basic observables of GPS were defined, and the effects of receiver-transmitter dynamics on the carrier frequency and code-rate were discussed. An overview of oscillator measures was given, and a discrete-time oscillator model was derived. The effects of oscillator variations on the different stages of a software receiver were quantified.

Block processing techniques can be divided into time-domain, frequency-domain, and time-frequency domain techniques. In general, a mixture of these techniques will have to be applied to achieve the required goal. Frequency domain techniques are not well-qualified for observation of the signal frequency behavior of long block sizes. However, efficient implementations of the correlator may be implemented using the FFT. To observe the dynamic behavior of the signal frequency time-frequency domain techniques are more suitable. The time domain discussion was restricted to block addition and block inspection techniques.

The case studies in this dissertation have shown that block processing enables GPS signal detection and estimation in environments where conventional sequential receivers
provide unreliable navigation service. Block addition techniques may provide extra signal gain, necessary to raise the post-correlation signal level above the noise floor. Thus, users are enabled to track GPS in urban environments, inside buildings, and under foliage. Described block addition technique requires *a priori* knowledge about the time and Doppler effect. The block addition technique is suitable for integration with low-cost IMUs. Block inspection techniques may be used to perform high dynamic tracking and RFI suppression. Combination of these techniques with block addition schemes could extend illustrated applications to signals at lower power levels.

The computational power required to perform block processing is no longer considered to be a limitation based on the developments of fast processors and logic, such as FPGAs. Near future implementations could augment software receiver implementations of conventional sequential receivers with described block processing techniques. This would reduce the need for re-acquisition in cases where the sequential receiver loses lock due to high dynamics and low signal levels. The inherent latency, introduced by block processing, may be resolved by integration of the GPS software receiver with IMUs, sequential receivers, or other navigation aids.

As an outgrowth of the discussed investigation in the application of block processing techniques, the core technology involved with GPS software receiver techniques was advanced. More insight was gained in the behavior of the sequential receiver, and its restrictions. Further, advanced techniques in the field of communications, computer vision, and digital signal processing were introduced in the observation of the GPS basic observables. Block processing equivalents of the basic GPS software receiver were
investigated.

It is the author's belief that this dissertation has only begun to uncover the potential of block processing techniques in the field of spread-spectrum radio navigation. In an era of ever growing computational power, handling of large blocks of data or information will enable the GPS receiver designer to design more robust and efficient algorithms based on block processing. It is recommended that further research should include the investigation of GPS / IMU integration using block processing techniques, and the application of block processing techniques to IMUs. Other recommended areas of investigation are the extension of proposed detection schemes to accurate measurement schemes using interpolators, discriminator schemes, or even time-frequency techniques, the combination of adaptive ADC and joint-time frequency analysis for RFI suppression, and the implementation of the above schemes in hardware.
REFERENCES


Contractor Report (NAS2-11339), December.


Appendix A. GPS Software Signal Simulator

A GPS software signal simulator was implemented in the “C” programming language to generate a well controlled GPS signal for the development and evaluation of sequential and block processing techniques. The simulator’s goal is to generate GPS signals, including additive Gaussian noise, sampled at 5 MHz with a minimum amount of distortion due to aliasing. The aliasing distortion requirement was set to be less or equal to one percent of the signal power.

A block diagram of the simulator is shown in Figure A.1. A signal is generated according to Equation 2.2 at a sample rate of 40 Msps. White Gaussian noise with predefined standard deviation is added, and the result is bandpass filtered. The bandpass filter is a 10th order Butterworth bandpass filter with a 3-dB bandwidth of 2.5 MHz and a center frequency, $f_c$, equal to 10 MHz. Next, the signal and noise are down-converted to an intermediate frequency of 1.25 MHz and lowpass filtered by a 5th order Butterworth LPF. To obtain a signal at 5 Msps, the 40 Msps signal is decimated by a factor of eight. Finally, the discrete-time signal is amplitude-quantized to obtain a digital signal. The number of quantization levels is user-defined.

The PSDs of the signal at the various stages of the signal simulator is illustrated in Figure A.2. Sampling at 40 Msps results in the capture of 99% (10 sidelobes) of the signal power. Only 1% of the signal power will be folded back into the desired
Figure A.1 Block Diagram of the GPS Software Signal Simulator
Figure A.2 PSDs of Signal at Signal Simulator Stages
signal spectrum, thus satisfying our set criterion. The decimation operation provides the desired 5 Msp signal.

The chosen filters are all infinite impulse response filters (IIRs). Consequently, the magnitude and phase response are not linear phase, and the transients are infinite length. In a future implementation the chosen filters could be replaced by finite impulse response filters (FIRs). Although, in general, these filters require a higher filter order to obtain a 3-dB bandwidth similar to an equivalent IIR filter, and introduce larger filter delays, they have finite transients and are linear phase.

The standard deviation of the generated noise samples is given by the desired $C/N_0$. The expression for the $C/N_0$ was derived in Chapter 2:

$$C/N_0 = \frac{S}{kT}$$  \hspace{1cm} (A.1)

The signal power, $S$, is given by $A^2/2$, where $A$ is the signal's amplitude. Sampling at a rate equal to $f_s$, limits the maximum bandwidth to $f_s/2$ (Nyquist). The variance of the noise is thus given by:

$$\sigma^2 = N_0 \frac{f_s}{2} = kT \frac{f_s}{2}$$  \hspace{1cm} (A.2)

The signal simulator allows a user to specify $S$, and $T$. For $S = -160 \text{ dBW}$, $T = 290K$, and a sample rate, $f_s$, equal to 40 Msp, the $C/N_0$ equals $44 \text{ dB-Hz}$ and the noise variance is $8.0 \cdot 10^{-14} \text{ V}^2$.

A uniform amplitude quantizer is used in the simulation. The number of bits, $b$, can
be set by the user and is, in general, equal to 8 or 12. The amplitude quantizer has $2^b$ levels. The input range of the quantizer can be set by the user.

A simulation was run with $b = 12$ bits, and an input range equal to four times the standard deviation of the input noise. Figures A3 through A.8 show the power spectrum density of the signal as it goes through the different stages of the simulator. The simulator outputs may be compared to Figure A.2. The signals were also generated by direct sampling a carrier signal at 1.25 MHz at 5 Msps. In that case, 10% of the signal would fold back into the desired band. This effect is illustrated in figure A.9. The y-axis in this example must be ignored; the signals have been given an offset, purposely. The simulator output signal has nulls, which are deeper than the direct generated signal, due to the reduction in aliasing.

Finally, the signal plus noise was generated. Figure A.10 shows the PSD of the combined signal. Note that the spectrum is nearly constant over the frequency supported range, indicating that the signal is entirely below the noise floor. The signal itself is illustrated in figure A.11. The variance of the resulting signal is approximately 90. The discrepancy between the set dynamic range and the illustrated signal, results from a 15 dB loss of noise in the filtering and down-conversion operations.
Figure A.3 Software Signal Simulator: PSD of $s(t)$
Figure A.4 PSD of the Signal after Bandpass Filtering ($x_f(t)$)
Figure A.5 PSD of the Signal after Down-conversion (x_2(t))
Figure A.6 PSD of the Signal after Lowpass Filtering (x_j(t))
Figure A.7 PSD of the Signal after Decimation ($x_d(t)$)

Figure A.8 PSD Signal after Amplitude Quantization ($x_q(t)$)
Figure A.9 Signal PSD Comparison for Sample Rates of 40 Msp and 5 Msp
Figure A.10 PSD of Output of Signal plus Noise

Figure A.11 Output of the Quantizer
Appendix B. The Effect of the Sample Rate Offset in Bandpass Sampling

The test setup, which is described in Chapter 8, uses the principle of bandpass sampling to obtain samples at 5 Msps. An extensive discussion on bandpass sampling can be found in (Vaughan, Scott & White, 1991). Its application to GPS and GLONASS is described in (Akos, 1997). Bandpass sampling uses intentional under-sampling to perform down-conversion; the signal is moved to the desired frequency location by intentional aliasing. Figure B.1 illustrates the concept of bandpass sampling. After filtering the desired signal with a high order bandpass filter, the signal can be sampled at a frequency smaller than the center frequency and aliased toward the desired frequency location. Note that the Nyquist criterion still holds; The sampling rate must be at least twice the desired bandwidth, $B_d$.

Let $f_c$ be the center frequency of the desired band, and $f_s$ the sampling rate, then the intermediate frequency is given by (Akos, 1997):

$$\text{if } \left| \frac{f_c}{f_s/2} \right| \text{ is even, } f_{if} = \text{rem}(f_c, f_s)$$

$$\text{odd, } f_{if} = f_s - \text{rem}(f_c, f_s)$$

(B.1)

where $\lfloor \cdot \rfloor$ is the floor operator and $\text{rem}(\cdot)$ is the remainder operator. Define $M_{fix} = \lfloor f_c/f_s \rfloor$ and $M_{fix}$ is even, as in used test setup, then $f_{if}$ may be written as:

$$f_{if} = f_c - M_{fix}f_s$$

(B.2)
Figure B.1 Bandpass Sampling
Now, assume there exists a sample period offset, $T_s' = T_s + T_e$, or $f_s' = f_s / \beta$ (see Section 5.4.2), then the intermediate frequency is given by:

$$f_{\text{if}} = f_c - M_{\text{fix}} f_s'$$

(B.3)

under the assumption that $M_{\text{fix}}$ does not change, or $|f_c/f_s'| = |f_c/f_s|.$

To obtain the intermediate frequency, as observed by the receiver logic, Equation 5.15 may be applied to Equation B.3:

$$f_{\text{if}}' = \beta f_{\text{if}}'$$

$$= \beta f_c' - \beta M_{\text{fix}} f_s'$$

$$= \beta f_c' - M_{\text{fix}} f_s$$

(B.4)

Substituting Equation B.2 into equation B.4, and evaluating the expression yields Equation 5.17:

$$f_{\text{if}}' = \beta f_{\text{if}}' + \frac{T_e}{T_s^2} M_{\text{fix}} = \beta f_{\text{if}}' + \alpha$$

(B.5)

Figure B.2 illustrates the effect of the sample Doppler on a bandpass sampled signal and a regular sampled signal. The impact on the bandpass sampled signal is one order of magnitude larger for all given error percentages.
Figure B.2 Comparison of Bandpass and Regular Sampling