

Architecture and Benefits of an Advanced GNSS Software Receiver

Mark G. Petovello, Cillian O'Driscoll, Gérard Lachapelle, Daniele Borio and Hasan Murtaza

Position, Location And Navigation (PLAN) Group

<http://plan.geomatics.ucalgary.ca>

Department of Geomatics Engineering

Schulich School of Engineering

University of Calgary

INTRODUCTION

Software-based GNSS receivers have been receiving considerable attention in the past several years. Not only do such receivers provide an excellent research tool for investigating and improving GNSS receiver performance in a wide range of conditions, they are also gradually becoming commercially viable, with some companies having already released products to the market (Morton 2007, Scott 2007, Fastrax 2008). The above advantages are further highlighted by the proliferation of new systems and signals. In contrast, the primary drawback of software receivers is the computational requirements needed to implement the receiver in the first place. In particular, with GNSS sampling rates generally exceeding 4 Msps (samples per second), processing requirements are indeed extreme, especially for the receiver's signal processing operations.

The major objective of a software receiver is therefore to efficiently implement the high rate computations while maintaining the desired flexibility inherent in a software-based approach. Unfortunately, these two objectives are generally at odds with an improvement in one aspect often occurring at the expense of the other. Traditional "hardware-based" GNSS receivers can be viewed as an extreme example of this where the most computationally intense processing is performed using very efficient hardware (i.e., application specific integrated circuits, or ASICs) which is inherently inflexible.

This paper discusses the general design, implementation and testing of a software-based GNSS receiver that addresses the above challenges. The software – GSNRx™ (GNSS Software Navigation Receiver) – was developed in C++ and is flexible enough to allow for a wide range of configurations involving different processors, receiver architectures, and acquisition and tracking strategies. With this in mind, the objectives of the paper are two-fold; first, to describe and rationalize the general architecture of the software, and second, to show some sample results obtained with the receiver.

The paper begins with a general overview of the software receiver architecture and its corresponding benefits in terms of processing efficiency and algorithm flexibility. The paper discusses how different processors can be incorporated into the receiver and the benefits realized. For example, the receiver can be configured to use a "pure software" approach, or, if available, any other co-processors such as an FPGA (Field Programmable Gate Array) or a GPU (Graphics Processing Unit). The latter is discussed in detail, as this represents a novel implementation for software receivers. It is also demonstrated how the choice of processor can be optimized by making use of any suitable instruction sets available on Intel processors.

Following the description of the software, some sample results will be presented that demonstrate the software's capability.

REFERENCES

Fastrax (2008) *Smart Positioning with Fastrax Software GPS Receiver*, Fastrax Ltd. 2008.

Morton, J. (2007) *Expert Advice: Software Defines Future, GPS World System Design and Test News*, GPS World, Retrieved January 7, 2008, from <http://sidt.gpsworld.com/gpssid/Expert+Advice+%26+Leadership+Talks/Expert-Advice-mdash-Software-Defines-Future/ArticleStandard/Article/detail/445464?contextCategoryId=35358&searchString=software%20receiver>.

Scott, L. (2007) *Directions 2008: Software-Defined Radio Role to Grow*, GPS World System Design and Test News, Retrieved January 7, 2008, from <http://sidt.gpsworld.com/gpssid/Receiver+Design/Directions-2008-Software-Defined-Radio-Role-to-Gro/ArticleStandard/Article/detail/476704>