Software Receiver Processing for Deep Space Telemetry Applications

Norman Lay,* Mark Lyubarev,* Andre Tkacenko,* Meera Srinivasan,* Kenneth Andrews,* Susan Finley,† Charles Goodhart,† and Robert Navarro †

Recently, much effort has been placed toward the development of the Reconfigurable Wideband Ground Receiver (RWGR): a variable-data-rate, reprogrammable receiver, whose technologies are intended for infusion into the Deep Space Network. A significant thrust of that effort has been focused on the development of field-programmable gate array (FPGA)–based algorithms for processing high-rate waveforms up to 640 Mbps. In this article, we describe the development of software receiver algorithms used to perform telemetry demodulation of low- to medium-data-rate signals.

I. Introduction

Telemetry processing for current and future NASA missions is expected to span a large range of data rates from less than 10 bits per second (bps) to many hundreds of Mbps. The entire receiver processing needed for higher-data-rate applications is currently the province of hardware-based digital signal processing. These functions are executed in reconfigurable hardware to provide a modular and straightforward avenue for enhancements, upgrades, and simple reprogramming for alternate applications. Details of the wideband telemetry processing architecture, algorithms, and performance are the subject of a companion article in this issue [1]. In contrast, the software receiver only utilizes the field-programmable gate array (FPGA) hardware to perform the digital downconversion and sample rate reduction processing. Sampled outputs are passed along to a suite of software algorithms to perform signal-dependent synchronization and matched filter processing.

The actual hardware for the Reconfigurable Wideband Ground Receiver (RWGR) is directly based on a processing platform from a family of ground-based instruments fielded within the Deep Space Network (DSN) for radio science and navigation applications. These include variants of the Radio Science Receiver (RSR) and Wideband Very Long Baseline Interferometry (VLBI) Science Receiver (WVSR)[1,2] [2]. It was envisioned that the process of demonstration and infusion would be simplified through the selection of a common hardware base for developing enhanced telemetry processing capabilities.

* Communication Architectures and Research Section.
† Tracking Systems and Applications Section.
[1] http://psdg.jpl.nasa.gov/wiki/Radio_Science_Receivers (internal JPL website).
[2] http://psdg.jpl.nasa.gov/wiki/WVSR_Phase_II (internal JPL website).

The research described in this publication was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

The outline for this article is as follows. Section II describes receiver architectures at the system and subsystem level. Section III describes the software processing algorithms. Section IV provides details of the receiver testing. In Section V, an example of rapid development for customized processing is described, and concluding remarks are provided in Section VI.

II. Receiver Architectures

The software algorithms reside as a construct within the overall RWGR system architecture depicted in Figure 1. Note that the portable radio science receiver (PRSR) [2] may equivalently substitute as a compact, single-channel version of the hardware processing platform shown in this figure. On the right-hand side of Figure 1, DSN antennas produce one or more intermediate frequency (IF) signals as inputs to the reprogrammable receiver hardware. For wideband telemetry, all receiver processing takes place within the FPGA. For low- to medium-rate signals allocated to software processing, the IF signal is digitized, downconverted, digitally filtered, and decimated prior to being streamed to a separate general-purpose processor that performs demodulation, synchronization, and data detection. The FPGA firmware and corresponding software (operating on the monitor and control [M&C] processor) functions that perform the real-time streaming and capture of these samples is available both from the native PRSR open-loop recording function as well as a processing mode of the RWGR itself. In both cases, the real-time streaming application transfers the open-loop samples to a storage device — either directly interfaced to a host processor or across a local area network (LAN) to another designated computer. The software processing application is able to retrieve those samples as they are being recorded and perform the processing from yet another computer, if so desired. For localized LAN connectivity with commonly deployed gigabit Ethernet (GbE), streaming sample rates into the megasamples per second (Msps) range are easily achievable. In an eventual deployment, it is envisioned that development work would be performed at a centralized location such as JPL and that the software receiver processing element would be locally staged (i.e., at a DSN

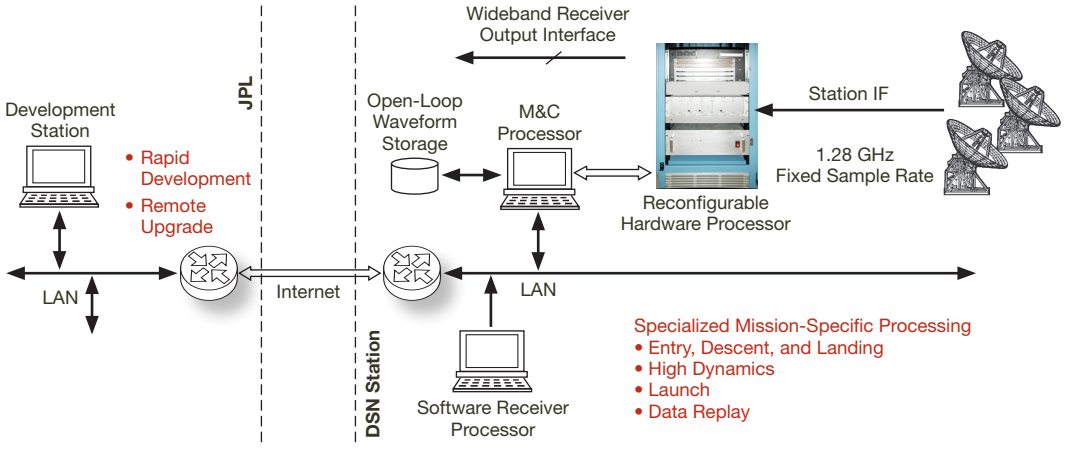

Figure 1. Reconfigurable receiver system block diagram.

station) to maximize real-time throughput connectivity between the hardware platform and the final telemetry processing stages.

The subsystem block diagram of Figure 2 illustrates the interface point within the FPGA processing stages of the wideband telemetry receiver design from which the downconverted, complex baseband samples are retrieved. Lowering of the sample rate is achieved through the filter-decimate mechanism contained on the first FPGA, followed by fine interpolation and/or programmable integrate-and-dump processing located on the receiver processing core. The capture random-access memory (RAM) circled in the figure utilizes circular buffer pointers and an overflow flag mechanism to communicate with the software process that is retrieving the samples and typically recording to a file for storage and remote access.

III. Software Processing Algorithms

The algorithms used to process low- to medium-data-rate telemetry waveforms address a larger set of downlink telemetry signaling formats than the wideband, suppressed carrier modulations processed exclusively by the reconfigurable FPGA hardware. These formats include traditional binary phase-shift keying (BPSK) and quadrature phase-shift keying (QPSK) as well as BPSK modulated upon a residual carrier. The residual carrier options include the ability to process with or without subcarrier modulations. For block-based forward error correction (FEC) codes (e.g., turbo or Reed-Solomon [RS]), the software also provides the ability to detect the frame synchronization marker in order to align the detected, soft-decision symbols in a known fashion as required by the FEC decoder.

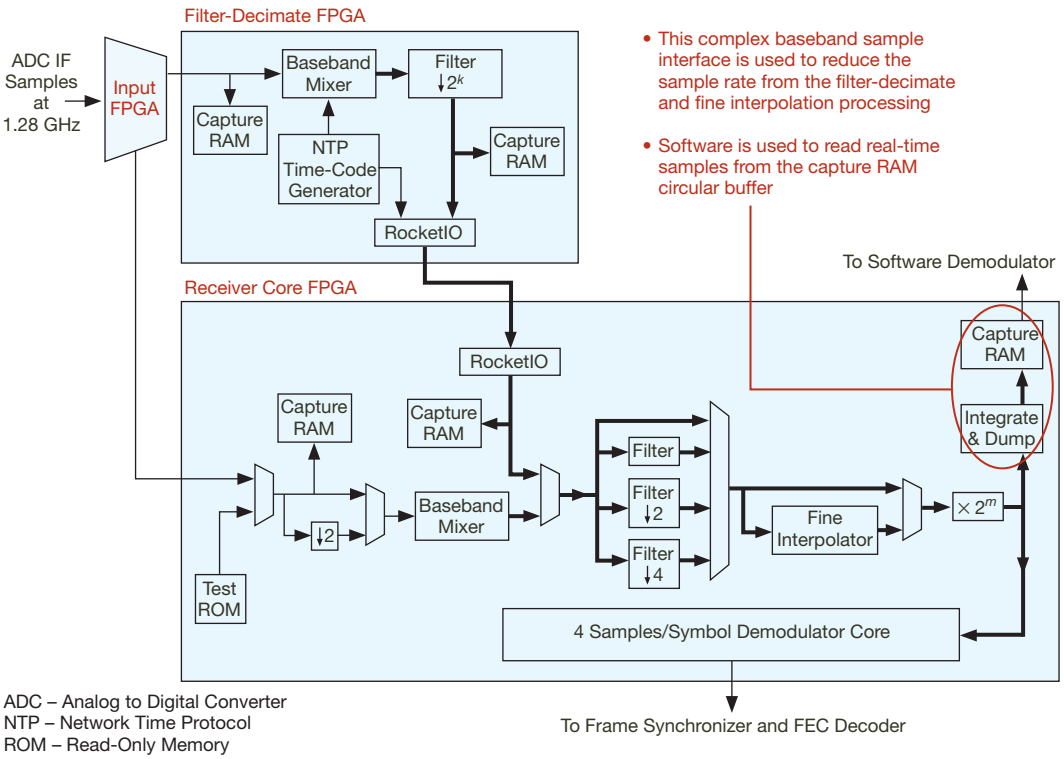

Figure 2. Reconfigurable receiver FPGA processing.

In Figure 3, the processing stages of the block-based demodulator operating upon a residual carrier and subcarrier modulated waveform are shown. The receiver processing makes liberal use of multirate processing to perform sample rate conversion between stages in order to maintain efficient processing (i.e., ensuring that the demodulator is not performing computations on a highly oversampled data record). The initial step utilizes a fast Fourier transform (FFT) to perform spectral analysis for estimating the frequency of the residual carrier. Once the location is identified, an open-loop downconversion and filtering operation to extract only the carrier is used to lower the sample rate prior to tracking by a digital phase-locked loop (PLL). The estimated phase is then reinterpolated up to the original input sample rate and used to derotate the incoming signal and extract the subcarrier waveform as a real signal. The subcarrier signal is open-loop downconverted by the nominal subcarrier frequency and then squared to strip the binary data modulation, which generates a tracking tone at twice the frequency of the residual offset. This tone undergoes filtering and downsampling to reduce the subcarrier loop processing rate and is then tracked by a second PLL. The subcarrier phase estimates are reinterpolated to the original subcarrier signal bandwidth and used to coherently demodulate the signal. Finally, the signal undergoes another rational sample rate conversion to produce close to an integer number of samples per symbol. The resulting sample stream is matched filtered, and processed by the symbol timing recovery algorithm. The timing recovery uses a delay-and-multiply nonlinearity to generate an unmodulated tone at the exact symbol rate. This tone is bandpass filtered for noise rejection, and used to generate the sample instances for the matched filter output.

IV. Testing

The initial testing of the algorithms described in Section III used a combination of modeled as well as recorded downlink telemetry data. The processing blocks of Figure 3 were tested on an individual basis to confirm independent operation and then sequentially integrated to ensure each of the elements were operating in concert. Examples of recorded data records that were successfully tested through to FEC decoding are enumerated in Table 1. These

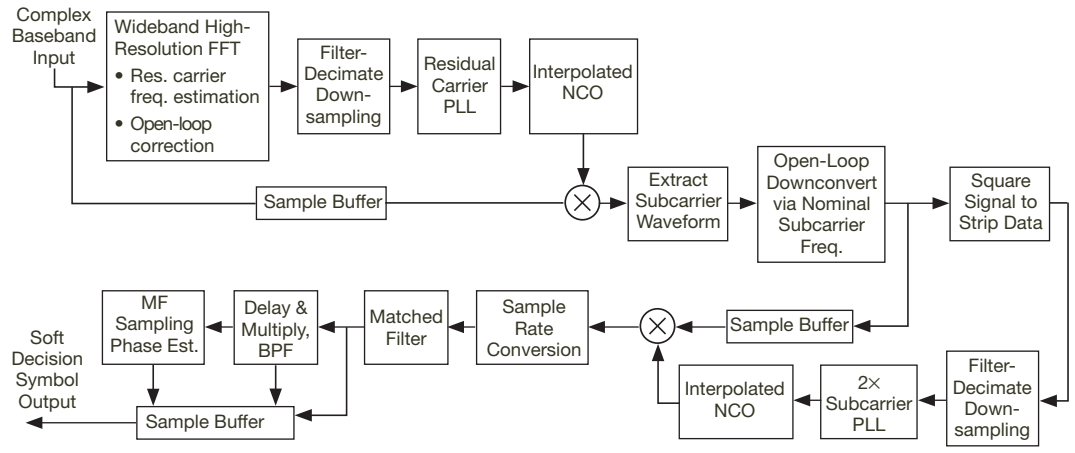

Figure 3. Block-based software demodulator processing flow diagram.

Table 1. Downlink telemetry testing examples, CY2009.

Mission	DOY	Symbol Rate	Modulation	Coding	Sampling Bandwidth
Kepler launch	66	300 sps	Residual carrier, 25 kHz subcarrier, BPSK	Turbo 1/3	100 kHz
Mars Odyssey	29–30	85320 sps	Residual carrier, 375 kHz subcarrier, BPSK	Convolutional (15, 1/6)	2 MHz
Mars Express	14	524288 sps	Residual carrier, bi-phase	Convolutional (7,1/2) + RS	2 MHz
Voyager 1	59	320 sps	Residual carrier, 22.5 kHz subcarrier, BPSK	Convolutional (7,1/2) + RS	100 kHz
Voyager 2	59	320 sps	Residual carrier, 22.5 kHz subcarrier, BPSK	Convolutional (7,1/2) + RS	100 kHz

tests provide a robust variety of actual downlink telemetry waveforms and demonstrated demodulation of these data sets serves as a good measure of the consistent functionality of the software receiver processing.

Successful operation of the receiver is assessed using several measures. In Figure 4, representative status displays plotting intermediate processing results are employed to evaluate proper operation. These displays include frequency and time domain plots that include but are not limited to the following: FFT estimation of the residual carrier, nonlinear generation of tones for tracking the subcarrier and symbol timing, symbol timing phase estimation, time domain comparisons of the residual and subcarrier estimates versus the signal-derived waveforms, and correlation processing to identify frame synchronization marker peaks. These results provide both qualitative as well as quantitative confirmation that the stages of receiver processing are executed properly.

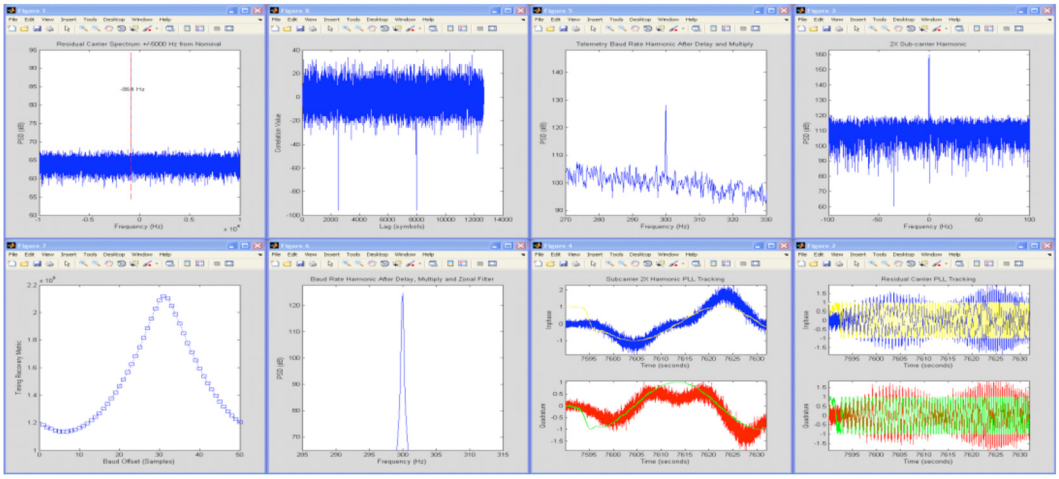

Figure 4. Representative status displays from the software demodulator.

The final measure for confirming successful receiver operation extends the processing into the FEC decoding domain. As previously noted, block-based FEC signal formats utilize known data sequences for achieving frame synchronization to aid in decoder symbol alignment. Figure 5 illustrates the results when correlating the detected soft-decision symbols with the known 96-bit attached synchronization marker (ASM) present in the Mars Reconnaissance Orbiter (MRO) downlink when the rate 1/3 turbo code is employed. Thresholding the correlation peaks provides the locations of the ASM. For all processed data, the measurement of the interpeak distances corresponded exactly to the 13434 QPSK symbols per codeword, indicating tracking of the downlink symbol timing. Utilizing the known ASM data, the raw bit error rate was also measured and corresponded to an uncoded bit signal-to-noise ratio (SNR), E_b/N_0, of approximately 0 dB. Furthermore, all turbo code frames were detected without error after processing by a software decoder. The estimated information bit SNR ranged from 4.5 to 4.7 dB.

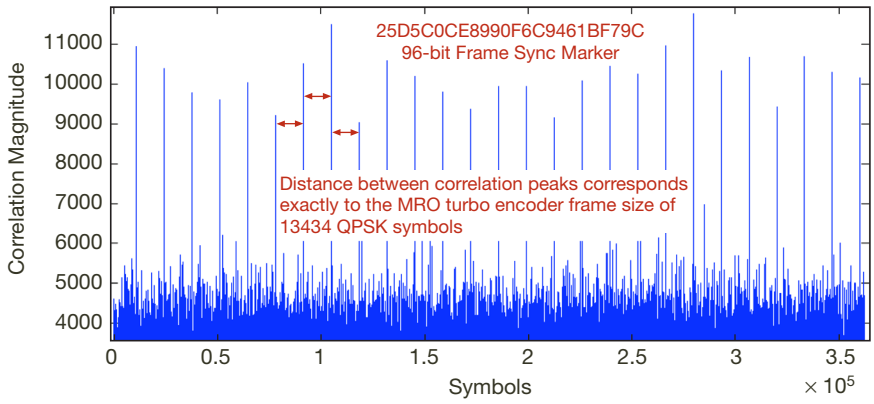

Figure 5. Detection of FEC frame synch markers.

V. Specialized Processing

In order to illustrate the value and flexibility of software-based processing, we describe a rapid, customized development of specialized receiver processing techniques to address the problem of co-channel interference present in deep-space mission downlinks. These conditions have, in the past, arisen owing to the unexpected longevity of missions and subsequent reassignment of downlink operating frequencies to later missions sent to explore the same planetary destination. While operational deconfliction via scheduling of telemetry passes resulted in a pragmatic solution to a critical issue, it also produces a relative data return inefficiency. Furthermore, with the proliferation of future missions, the ability to quickly address such unanticipated impairments will be critical for future capabilities.

Under a small technology development initiative executed over the course of a few months,[3] the authors investigated the performance of algorithms to detect and coherently remove strong telemetry interference from a weaker time- and frequency-coincident signal.

[3] N. Lay, R. Navarro, M. Srinivasan, A. Tkacenko, and K. Andrews, "Simultaneous Reception of Interference Limited Downlinks," *2009 Research and Technology Development Program Annual Report*, Jet Propulsion Laboratory, Pasadena, California, http://rtd.jpl.nasa.gov/ (internal JPL website).

A block diagram of the processing steps used to demonstrate this concept is shown in Figure 6. Pre- and post-interference removal results shown in the spectral domain are shown in Figure 7. Although these techniques were tested on synthesized co-channel interference data sets derived from recorded downlink signals, the algorithms are capable of operating in real time for sufficiently narrowband data sets.

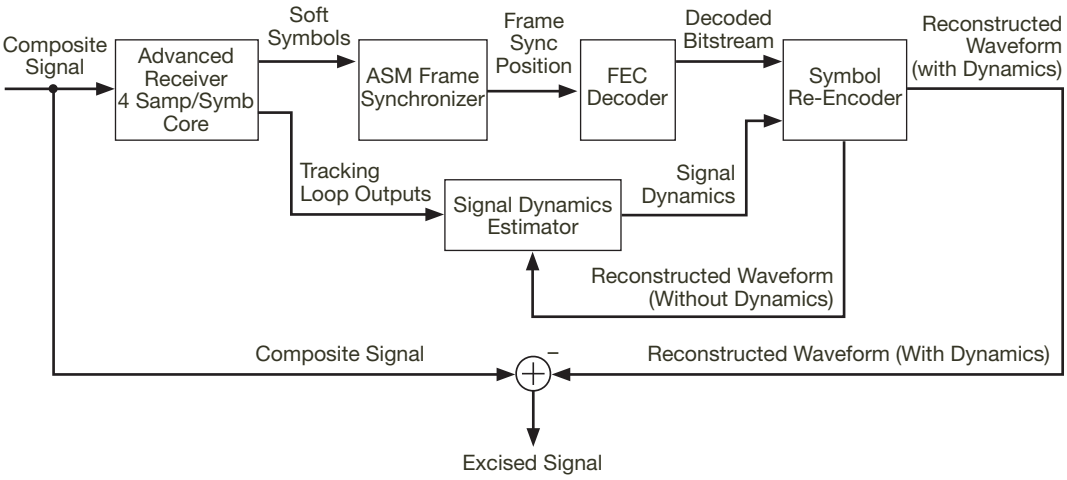

Figure 6. Interference detection and excision processing.

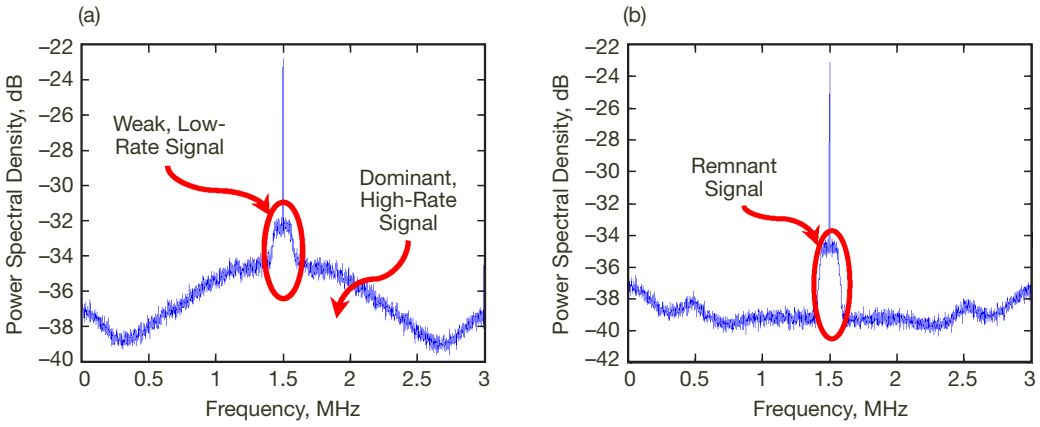

Figure 7. Wideband view of composite signal spectrum (a) prior to and (b) after dominant interference signal estimation and removal.

VI. Concluding Remarks

The software receiver algorithms developed for real-time telemetry processing of low- to medium-data-rate signals from deep-space missions provide a flexible and cost-effective means to augment the DSN's capabilities through the use of fielded and/or soon-to-be fielded hardware platforms coupled with commercial off-the-shelf general-purpose computers. Furthermore, by pursuing such a development and infusion path, the DSN is afforded the opportunity to leverage the benefits of ever-increasing processing capabilities available in commercial computing technology that have routinely shown tremendous growth over time.

References

[1] K. Andrews, A. Argueta, N. Lay, M. Lyubarev, E. Sigman, M. Srinivasan, and A. Tkacenko, "Reconfigurable Wideband Ground Receiver Hardware Description and Laboratory Performance," *The Interplanetary Network Progress Report*, vol. 42-180, Jet Propulsion Laboratory, Pasadena, California, pp. 1–30, February 15, 2010. http://ipnpr.jpl.nasa.gov/progress_report/42-180/180D.pdf

[2] S. Rogstad, R. Navarro, S. Finley, C. Goodhart, R. Proctor, and S. Asmar, "The Portable Radio Science Receiver (RSR)," *The Interplanetary Network Progress Report*, vol. 42-178, Jet Propulsion Laboratory, Pasadena, California, pp. 1–7, August 15, 2009. http://ipnpr.jpl.nasa.gov/progress_report/42-178/178B.pdf