APPLICABILITY OF THE JTRS SOFTWARE COMMUNICATIONS ARCHITECTURE IN ADVANCED MILSATCOM TERMINALS

Lee Pucker
Spectrum Signal Processing, Inc.
Burnaby, B.C, Canada

ABSTRACT

The JTRS Software Communications Architecture (SCA) provides important benefits in military satellite communications programs by allowing a reconfigurable MILSATCOM terminal to support multiple waveforms. Special treatment of the SCA is often required, however, in order to support the multiple user channel and high-bandwidth nature of advanced MILSATCOM waveforms. Such waveforms demand hardware providing a high degree of parallelism with low-latency connections between processing elements. This paper presents techniques for mapping the SCA onto such hardware in a manner that maintains the benefits of SCA while providing the required performance to support advanced waveforms.

INTRODUCTION

Next generation military satellite communications programs are beginning to follow the lead of the JTRS program in adopting the use of software defined radio technology. This will allow a family of reconfigurable satellite terminals, based on a common platform architecture, to replace the dozens of independent stove-pipe terminals currently deployed [1]. The benefits of this shift in paradigm include:

- Increased interoperability, with a single terminal now able to communicate over multiple satellite networks
- Simplified upgrade of existing waveforms, allowing new features and capabilities to be added without the wholesale replacement of the satellite terminal
- Reduced time to deployment for new waveforms
- Reduced maintenance, spares, configuration management and other logistic costs

A key element of the JTRS program is the Software Communications Architecture (SCA) developed by the Modular Software-programmable Radio Consortium (MSRC) under contract to the JTRS Program Office [2]. This architecture “objectizes” the radio structure, and defines a standard application framework (referred to as the Core Framework) for instantiating and connecting the waveform objects associated with each radio channel.

The use of the JTRS SCA in an advanced satellite communication terminal can often be a challenge, since the fundamental differences in the waveform structure utilized in advanced MILSATCOM networks can often lead to a modification in the way the terminal is architected over the establish model for JTRS radios. This paper will explore these differences in greater detail, and identify ways in which the SCA Core Framework can extend inside of the architecture of a military satellite communications terminal, allowing the SCA Core Framework to act as a standard application framework for advanced military satellite communications programs.

SCA CORE FRAMEWORK SUPPORT FOR PRIMITIVE MILSATCOM WAVEFORMS

Traditional military satellite communications systems, such as the UHF DAMA SATCOM Waveform supported by the JTRS program [3], typically employ a Frequency Division Multiple Access technique for accessing transponder bandwidth on the satellite payload. This technique uses a single channel per carrier (SCPC) for satellite communications, with channel assignment often based on advanced Demand Assignment Multiple Access (DAMA) protocols.

Software Defined Radio architectures supporting these types of waveforms are well documented [4, 5, 6, 7]. These architectures are summarized in Figure 1. In this model, an FPGA is used for the high speed “front-end” pre-processing, with either a GPP or DSP performing the baseband signal processing, depending on the size, weight, and power limitations imposed upon the modem sub-system. Depending on the overall radio architecture, the A/D and D/A converters may be considered a part of the modem or a part of the RF Transceiver sub-system. When a GPP is incorporated into the modem architecture, waveform link layer processing may be supported directly by the modem. Conversely, when a DSP is employed in lieu of a GPP, a black side “host” may be utilized to provide a common GPP outside of the

Copyright © 2003 IEEE. Reprinted from MILCOM 2003 - IEEE Military Communications Conference, no.1, October 2003, pp. 533 - 537. This material is posted here with permission of the IEEE. Such permission of the IEEE does not in any way imply IEEE endorsement of any of Spectrum Signal Processing's products or services. Internal or personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution must be obtained from the IEEE by writing to pubs-permissions@ieee.org. By choosing to view this document, you agree to all provisions of the copyright laws protecting it.

modem architecture that is shared across all modem channels.

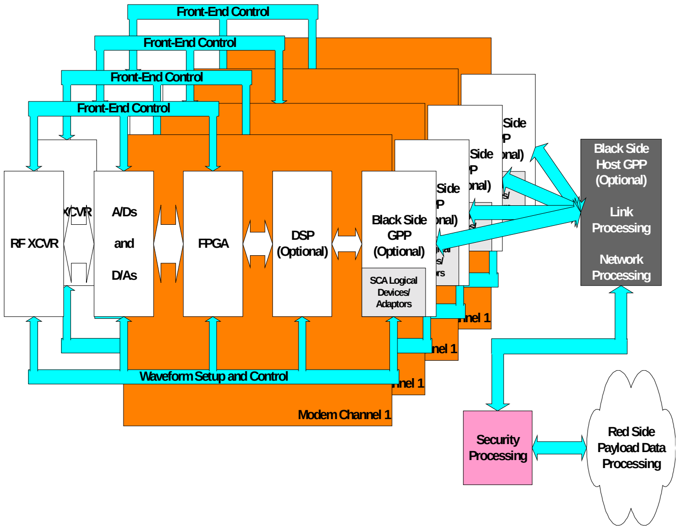

Figure 1: Typical Radio Hardware Architecture for a Multi-channel "System of Systems

Support for the SCA Core Framework within this type of architecture requires that each of the processing devices within the radio be exposed to the CORBA based “Logical Software Bus”. The CORBA bus is then used to setup and tear down software components on these devices, and to connect and control these software components through well-defined CORBA based interfaces. Processing devices, such as FPGAs and DSPs, that do not provide direct support for CORBA enabled communications, require a logical device adaptor that resides on a CORBA enabled processor and acts as a proxy for the non-CORBA enabled device within the confines of the Core Framework. These adaptors use a local application programming interface (API) that supplies library calls that are native to the modem platform to bridge communications between the non-CORBA enabled processing elements and the “Logical Software Bus”. A more detailed analysis of this the SCA Mapping for these radio architectures can be found in [8].

SCA CORE FRAMEWORK SUPPORT FOR ADVANCED MILSATCOM WAVEFORMS

Architectures supporting advanced military satellite communications waveforms differ from the model presented above in two key areas:

- Advanced military satellite communications networks often employ a complex multiple access scheme, including Time Division Multiplexing (TDM) and Time Division Multiple Access (TDMA), allowing many logical user channels to occupy a single RF carrier [9, 10]. While advanced JTRS waveforms, such as Link 16, may also employ these types of multiple access structures [11], the paradigm for tactical radios of this type is to support a only a single user channel per radio channel, whereas a military satellite communications terminals supporting strategic command and control communications typically require support for multiple simultaneous user channels per terminal [12]. For example, the SMART-T terminal supporting EHF communications services provides 16 simultaneous MDR capable channels [13]. As such, the channel based architecture typically employed in a JTRS radio may not be suitable for use in an advanced MILSATCOM terminals, since the front end processing within the terminal’s modem structure must support multiple backend channels.

- Military satellite communications networks often operate at much higher data rates, and subsequently much higher bandwidths, than a typical JTRS radio. For example, the MILSATCOM Advanced EHF Waveform supports payload data rates of 8 Mbps per user channel, with multiple user channels supported on each RF carrier [14]. Advanced communications systems of this type can support composite payload data rates of greater than 40 Mbps, requiring terminal architectures that support RF channel bandwidths often exceeding 100 MHz [15,16]. The front end processing of these types of systems often require “parallelization” of the processing algorithms to allow for a realizable modem architecture given the performance constraints of available processing devices. This differs substantially from typical JTRS radio architectures, where lower data rates allow for more channelized processing.

To illustrate these issues, consider a theoretical MILSATCOM radio architecture supporting 4 simultaneous user channels with a TDM multiple access scheme used for the downlink channels and a TDMA multiple access scheme used for the uplink. A functional block diagram for this type of system is shown in Figure 2. For this example, a data rate of 8.192 Mbps per channel has been selected, with 16 data channels and 1 synchronization channel per downlink carrier utilizing rate ½ convolutional coding. An analysis of the architecture in this example shows two key concepts that must be addressed by the SCA Core Framework utilized within the MILSATCOM terminal: Shared Resource

Processing and Array Processing. Each of these will be explored separately.

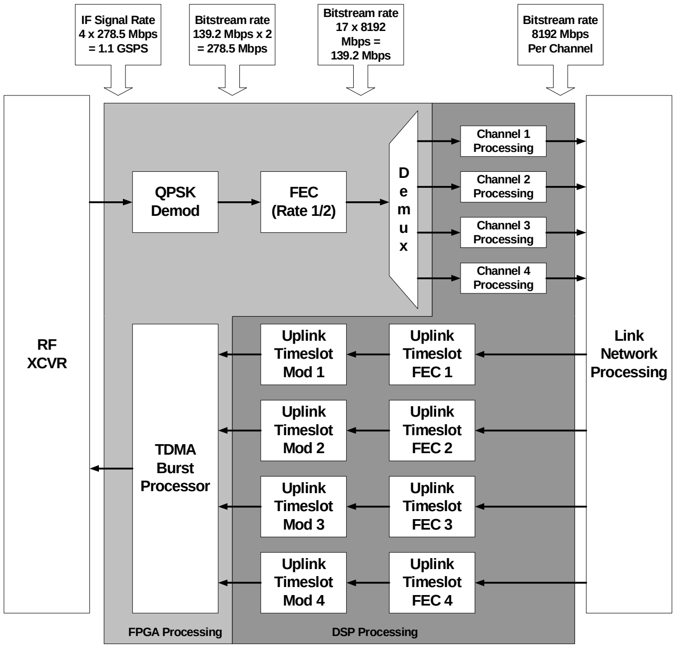

Figure 2: TDM/TDMA Modem Example

ARRAY PROCESSING

Pipelined signal processing for complex signals at the bandwidths required in many advanced MILSATCOM terminals cannot be directly supported by any single current generation, or even next generation, signal-processing device. As a result, front-end algorithms used in these types of systems are "parallelized", with data snapshot into memory allowing processing to occur simultaneously on separate sections of the signal data, as shown in Figure 3. Often, this will require the algorithm to be distributed across multiple signal processing elements, with high-speed low-latency communications maintained between the elements for data transfer, synchronization and control. Thus, data sampled at 1.1 GSPS in the example above could be buffered and distributed to an array of FPGA devices that would collectively perform the channelization processing necessary to extract or insert user channels, which operate at a much more manageable rate, from the IF interface.

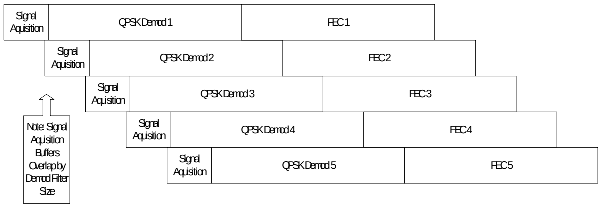

Figure 3: Sample Timing Requiring 5 Parallel Processes for Front End Processing of Sample TDM Signal

Support for array processing in this manner by the SCA Core Framework requires that the processing elements associated with waveform resource that has been "parallelized" be perceived as a single logical device. The SCA Core Framework can support this capability by instantiating a Logical Device that aggregates each of the valid processing elements within the array into a new Logical Device. The Device Manager associated with those devices will establish port connections between this new Cumulative Logical Device and the local Logical Devices, as appropriate, to allow the Cumulative Logical Device to act as a device facade on the "Logical Software Bus" (see Figure 4) by delegating its operations to one or both of its subordinate Devices. This technique allows the individual Logical Devices, as well as the Cumulative Logical Device, to be exposed to the Application Factory, which allows a great deal of flexibility in assigning device resources for other waveforms.

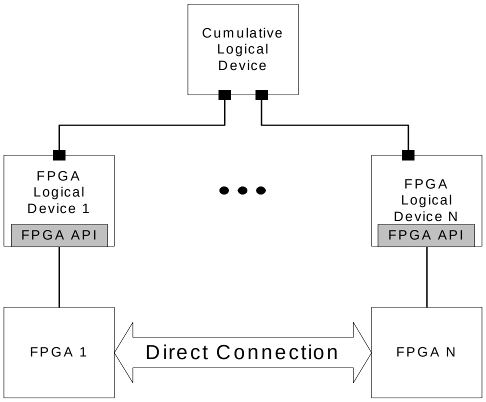

Figure 4: Core Framework support for Array Processing in the modem architecture

SHARED RESOURCE MODEM PROCESSING

The use of an advanced multiple access scheme in advanced military satellite communications requires front end modem processing that is shared across multiple user channels. These channels may be dynamically assigned, with individual channels set up and torn down as needed. In addition, the back end processing for each channel may differ significantly in each instantiation. For example, each channel on a common TDMA carrier may use a different modulation scheme. As a result, even though the front end processing resources can be created to support multiple simultaneous waveforms, these resources must be loaded and maintained outside of the normal waveform setup and tear down activity. Thus, as individual channel processes are set up or torn down based on current access requirements, they must attach or detach from this front end resource.

To support this capability within the confines of the SCA, a separate "application" is created allowing the instantiation of each front-end modem resource that will be shared by multiple backend waveforms. This application then acts as a resource for the other waveform applications. For example, a 10-channel channelizer image could be created for a given FPGA, and this image would be installed on the modem platform as a separate application (See Figure 5).The channelizer application then registers with the Naming Service, allowing independent channel processing applications to find and attach themselves to free channels available from the channelizer application. The channelizer application keeps track of attached channel processing applications, and will only allow itself to be torn down when no modem applications are attached to it. The down side of this model is that a single modem application can tie up an entire processing device.

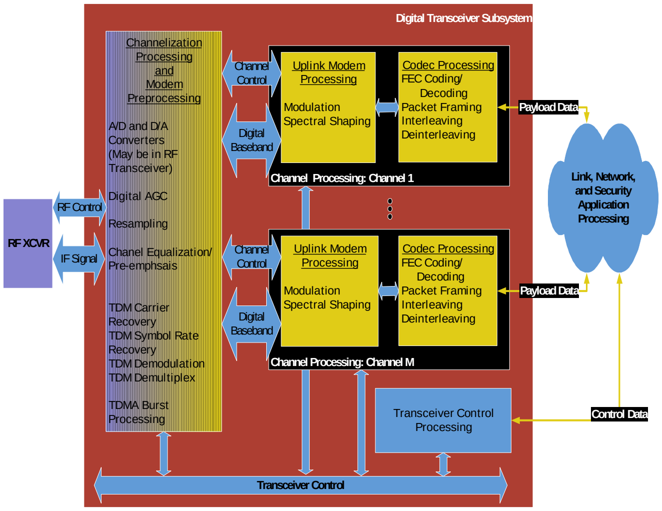

Figure 5: Common channelizer shared by multiple user channels in a TDM/TDMA Modem Architecture.

CONCLUSION

The SCA supports almost infinite variations on the modem architectures used in Software Defined Radio systems. As such, the SCA Core Framework is being widely considered for use as the standard application framework for a variety of military communications programs including WIN-T, FAB-T, and FCS. The approaches described above outline manners in which issues associated with how the SCA Core Framework interacts with the radio architecture in advanced military satellite communications terminals. This includes extending the SCA to support both array and shared resource modem processing. By applying these techniques the SCA Core Framework can act as the standard applications framework in these types of advanced military satellite communications programs.

REFERENCES

1. Deborah Wilson, "Software Defined Radio", Keynote Address at SDR Technical Conference, November 2002
2. SCA Version 2.2, http://jtrs.army.mil/pages/sections/technicalinformation/fset_technical_sca.html
3. John Bard and TJ Mears, "A CORBA Based DAMA Modem for the Digital Modular Radio (DMR)", http://www.spacecoastcomm.com/docs/334.PDF
4. Peter G. Cook and Wayne Bonser, "Architectural Overview of the SPEAKeasy System", *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 4, April 1999, pp. 650 - 661
5. Arthur Akhurst, "Scalability Aspects of the SRA", Software Defined Radio Forum Contribution, 25 May 2001, http://www.sdrforum.org/2001_docs.html
6. David Murotake, "JTRS SCA Platform Hardware Scalability", Software Defined Radio Forum Contribution, 13 Nov 2001, http://www.sdrforum.org/2001_docs.html
7. SCA Training Course Day 3 - Waveform Design http://jtrs.army.mil/pages/sections/technicalinformation/fset_technical_waveforms.html
8. Lee Pucker and Geoff Holt, " Extending the SCA Core Framework Inside the Modem Architecture of a Software Defined Radio", To Be Published
9. Jung Min Park, Uday R. Savagaonkar, Edwin K. P. Chong, H. J. Siegel and Steven D. Jones, "Efficient resource allocation for QoS channels in MF-TDMA satellite systems", MILCOM 2000 - IEEE Military Communications Conference, no. 1, October 2000, pp. 645 - 649
10. James Yoh, Charles C. Wang and Gary W. Goo, "Survey of Ka-band satellites for wideband communications", MILCOM 1999 - IEEE Military Communications Conference, no. 1, October 1999, pp. 120 - 125
11. Link-16 Summary, May 1997, http://prodevweb.prodev.usna.edu/SeaNav/NS40x/NS401_old/introduction/html/linksum.html
12. Richard A. Williams and Heywood I. Paul, "Military X and KA - bands media mix analysis for the wideband gapfiller satellite system", MILCOM 2000 - IEEE Military Communications Conference, no. 1, October 2000, pp. 1214 - 1218
13. Wade A. Leonard, "MILSTAR and DOD teleport", MILCOM 2000 - IEEE Military Communications Conference, no. 1, October 2000, pp. 1181 - 1186
14. John Mogor and John Wilson, "Wideband Communication Options for the E6-B", http://www.mitre.org/support/papers/tech_papers99_00/wilson_wideband/wilson_ideband.pdf

15. Michael A. Rupar and David R. Beering, “A Ka-band high data rate shipboard satellite terminal”, MILCOM 1999 - IEEE Military Communications Conference, no. 1, October 1999, pp. 1490 - 1494

16. Len P. Calabretta, Matthew Ryerse, and Lawrence Goeller, “ X-Band, Ka-Band, and Future MILSATCOM Requirements: and Outing in Trade Space”, MILCOM 1998 - IEEE Military Communications Conference, October 1998