AN OPEN ARCHITECTURE SCA REFERENCE PLATFORM

David K. Murotake, Ph.D., SCA Technica, Inc. dmurotak@scatechnica.com
Phone +1-603-321-6536, Fax +1-603-222-2098
Address: PO Box 3168, Nashua NH 03061

ABSTRACT

Software defined radios (SDR) are capable of supporting a number of radio technologies (waveforms) using software downloads. For each waveform, a "reference waveform" is developed and published as a "standard" for deployment on different radio platforms. : At present, each "reference waveform" each has a unique hardware reference implementation and waveform architecture. Thus each waveform has a unique Hardware Abstraction Layer (HAL), Board Support Package (BSP) and Operating Environment (OE). Furthermore, each Reference Waveform may employ a unique waveform component architecture, with control and and dataflow connections being made in a unique fashion. This stymies attempts to create a unified, composable component architecture, or a Hardware Abstraction Layer (HAL), common to multiple waveforms. This paper explores the requirements and architecture of a proposed SCA 2.2.2 compliant Open Architecture Common Reference Platform, based on COTS commercial hardware and software components, which may be used to simplify the development and porting of future Reference Waveforms to SDR programs. Additionally, the paper examines the possibility of using the Common Interface Cryptologic Module (CICM) interface for the open architecture Type 1 Crypto Development Card.

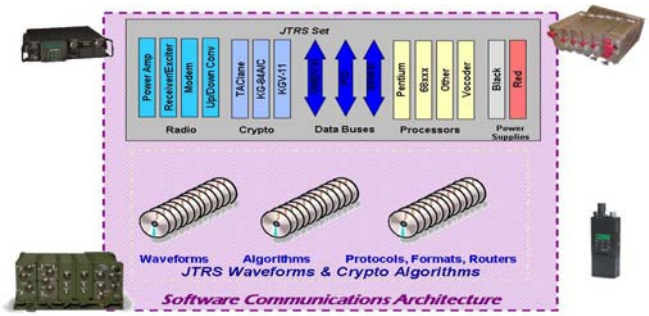

Figure 1. Joint Tactical Radio System software defined radio. (Figure: JTRS Public Distribution)

1. INTRODUCTION

One example of a family of SDR radio sets is the Joint Tactical Radio System (JTRS) (Figure 1). JTRS radio sets can support multiple radio technologies through changes to their software and firmware. This allows a single radio set to communicate on multiple radio networks, and also allows a family of JTRS radios (called Clusters) to be used to replace a variety of different radio types used on ground vehicles, aircraft, man-pack and hand-held form factors. Nine "reference waveforms" have been developed so far by JTRS Program for application to different types of radio platforms.

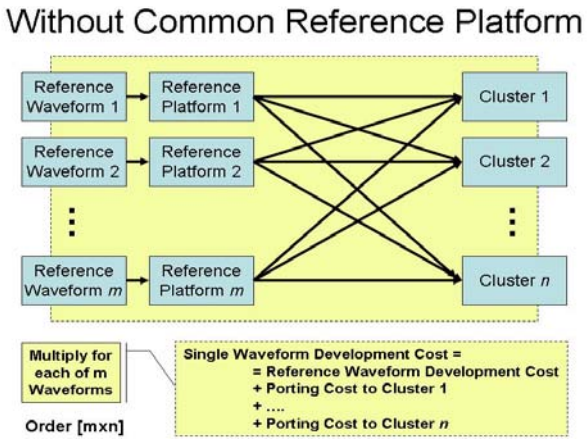

Figure 2. Waveform "porting" costs without a Common Reference Platform can grow by order [m X n}.

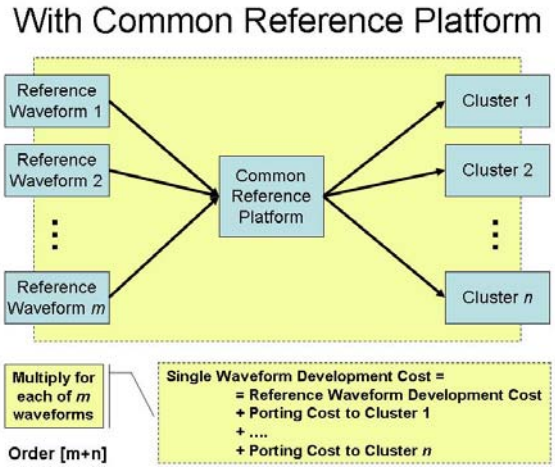

Figure 3. Waveform porting costs with a Common Reference Platform. Costs can be reduced to order of [m + n].

Without a common reference platform for hardware and waveform component architecture, the porting costs for waveforms can become more costly. For *m* waveforms and *n* platforms, the cost is an order of *m X n* (Figure 2) [1]. By contrast, when a common platform is utilized, the expense approaches order of *m* + *n* (Figure 3).

2. REQUIREMENTS

The principle objective of this section is to identify the key requirements and architecture for an SCA 2.2.2 compliant Common Reference Platform hardware (Figure 4).

Figure 4. An example of a Red/Black isolated radio platform, in which the protected "Red" side is isolated from the "Black" side by a cryptographic boundary.

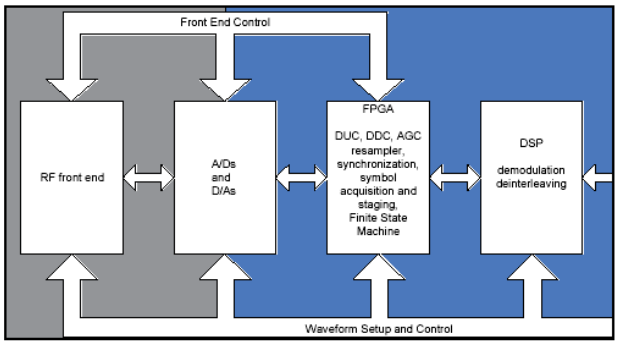

Figure 5. Black side control and data flow architecture. (Source: Maier, Spectrum Signal Processing)

Ideally, the "Reference Platform" should support the largest practical number of current reference waveforms as well as future reference waveforms.

Most modern military waveforms being implemented on SDRs today fall into four general categories:

- Slow-hopping, narrow band, half-duplex (<4000 hops/sec, < 1 Mbps). Payloads: analog or digital.
- Fast-hopping, narrow band, half-duplex (>4000 hops/sec, < 1 Mbps). Payloads: digital.
- Low probability of detection (LPD), pseudo-noise direct sequence spread spectrum, half- or full-duplex (~30 dB spread, < 1 Mbps). Payloads: digital.
- Wideband networking waveforms, half- or full-duplex, possibly employing mesh and mobile ad-hoc network (MANET) routing protocols. Payloads: digital.
- Commercial radio technology waveforms.

The hardware processing requirements of such waveforms have been discussed in the literature. [2, 3, 4]. Most industry analysts agree that for the current JTRS waveforms, the wideband networking waveform (WNW) is iconic of the DSP requirements for modern networking waveforms. Maier [2] estimates the requirements for a single channel WNW in (Table 1).

Table 1 Estimated black-side computing requirements for WNW. (Source: Maier, Spectrum Signal Processing)

Component Name	Proposed Processing Technology	Estimated Resource Utilization
Rx Resampler	FPGA	5400 Logic Cells, 15 MAC Blocks, 1 Block RAM
OFDM Symbol Synchronization	FPGA	3197 Logic Cells
OFDM Symbol Acquisition	FPGA	1590 Logic Cells, 2 MAC Blocks, 1 Block RAM
OFDM Symbol to Bit	FPGA	10744 Logic Cells
FSM	FPGA	5580 Logic Cells
Deinterleave/ Decode/ Interleave/ Encode	TMS320C6416	<100%
Link Network Layer Processing	GPP	< 1000 MIPS
Symbol Staging	FPGA	Minimal
Tx Resampler	FPGA	5400 Logic Cells, 15 MAC Blocks, 1 Block RAM

Not including the processing requirements of the operating system and object request broker (ORB), Sterkel [5] estimates the implementation processing requirements for a single WNW channel to be:

- TOTAL MIPS: 1600 - 2300
- TOTAL MBYTES: 500 – 800
- DSP MIPS: 250 – 350
- DSP MBYTES: 7.0 – 8.5
- FPGA LE: 70000 - 85000

Maier [2] concludes the processing requirements for a single WNW channel requires:80% of one MPC 8541 microprocessor at 533 MHz, and 53% of a Vurtex-4 LX60 FPGA. These computing resources can be easily supplied by COTS hardware and software.

Requirements for the Common Reference Platform include:

- The product shall comply with SCA 2.2.2, including public APIs and extensions. Note: SCA compliance testing is explained in Haskings [6] and JTeL [7].
- The product shall satisfy the Maier [2] and Sterkel [5] implementation requirements for WNW.
- The product shall be modular, scalable, using an open architecture. As a goal, the product shall be implemented in a small form factor such as 3U Compact PCI to best emulate Small Form Factor radio development environment.
- The product shall includes a modular, open-architecture Crypto Development Module employing a cryptographic chipset supported and approved by the JTRS, such as a General Dynamics Advanced INFOSEC Machine (AIM),

Harris Sierra II, or Raytheon Cornfield. As a goal, the product shall support the Common Interfaces to Cryptographic Modules (CICM) APIs. As a goal, the module shall be implemented in 3U CPCI.

- The product, including Crypto Development Module, shall have a high assurance architecture capable of defeating the typical wireless/network threats, addressing JTRS CP-295, "Exposed Black Side". See also Murotake and Martin [7].
- The common reference platform shall be OS, ORB, Core Framework, and Crypto Engine agnostic.
- The product shall be internationally exportable.

3. HARDWARE ARCHITECTURE

The proposed hardware architecture is based on COTS hardware and software to the greatest extent possible. Although 3U cPCI COTS hardware exists for much of the hardware requirement (Figure 6), one significant "gap" is the availability of a CICM and SCA compliant cryptographic development module.

Figure 6. COTS hardware such as the SDR-4000 can provide a good "starting point".(Source: Spectrum Signal Processing.)

An example configuration for a COTS based Common Reference Platform is shown in (Table 2).

Table 2. COTS hardware for Common Reference Platform.

Module	COTS Product
Testbed Chassis	SSPI Chassis 3U CPCI Air Cooled
RF/IF	COTS to be selected
A/D-D/A	SSPI XM3221 GPIO card
Black Baseband DSP	SSPI PRO4600 Signal Processing cards
Crypto Development Module	Not commercially available - to be developed
Red Baseband DSP	SSPI PRO4600 Signal Processing cards
Red Host	Personal Computer (Desktop or laptop)

4. CRYPTOGRAPHIC DEVELOPMENT MODULE

One module not currently available as COTS is a family of products providing a common open architecture interface to a variety of cryptographic technologies supported by JTRS algorithm development, such as the GD AIM and Harris Sierra II chipsets. A family of 3U CPCI modules (Figure 7) can be designed to interface with the 3U CPCI chassis and modules listed in Table 2. The cryptographic development modules differ from currently available cryptographic modules:

- The modules will employ a new standard called Common Interfaces to Cryptographic Modules (CICM) [8]
- The modules provide a common set of APIs and functional interfaces to different encryption chipsets from different manufacturers.

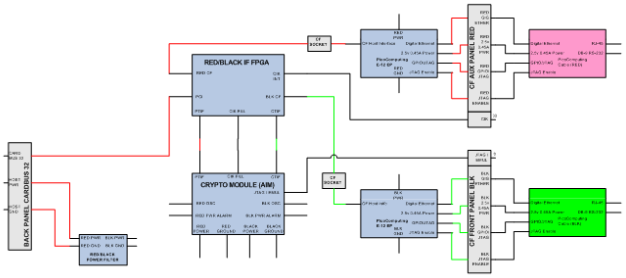

Figure 7. Common interfaces to cryptographic modules (CICM) APIs are employed on this concept for a Cryptographic Development Module for the SCA Common Reference Platform. The version employing the Advanced INFOSEC Machine (AIM) is shown in this figure.

The CICM is a draft Application Programming Interface (API) specification standardizing the management of and access to cryptographic services implemented by Type 1 and Type 2 cryptographic modules. CICM will enable applications to access reference platform cryptographic services in a common way regardless of the specific underlying encryption module. CICM allows application programmers to access cryptographic functions without knowledge of the details of the specific module that is providing the services. This is especially important for the SCA Common Reference Platform, since waveform developers may be developing applications for a variety of radio platforms without specific knowledge of underlying encryption chipset types. Although existing modules with similar functionality are built with differing, proprietary interfaces, there exist significant advantages to adhering to one common standard. The Common Reference Platform project will attempt to incorporate CICM- and SCA-compliant APIs and provide valuable feedback to the project sponsors.The CICM draft specification [8] defines a comprehensive set of functions necessary for module operation and management, regardless of manufacturer. These functions and their corresponding APIs comprise several classes:

- Module Management
- Algorithm Management
- Policy Management
- Channel Management

- Data Management
- Key Management
- Trust Anchor Management

5. SOFTWARE ARCHITECTURE

The Common Reference Platform complies with the JTRS Software Communcation Architecture (SCA) Version 2.2.2 [9]. This specification defines a comprehensive set of functions necessary for module operation and management, regardless of manufacturer. Figure 8 shows one view of the SCA - the architecture layer diagram.

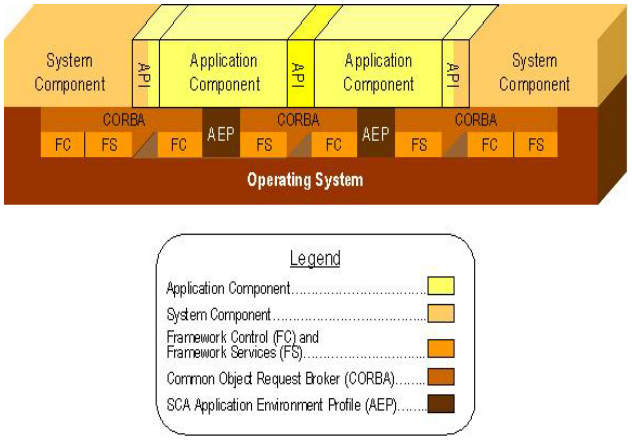

Figure 8. SCA Architecture Layers. (Source: SPAWAR)

Figure 9 shows another view of the SCA, more clearly identifying the interaction between applications, core framework, CORBA and the Application Environment Profile (AEP) compliant POSIX operating system. Figure 10 shows the procedural and message passing interfaces for the system, including the SCA and CICM compliant APIs.

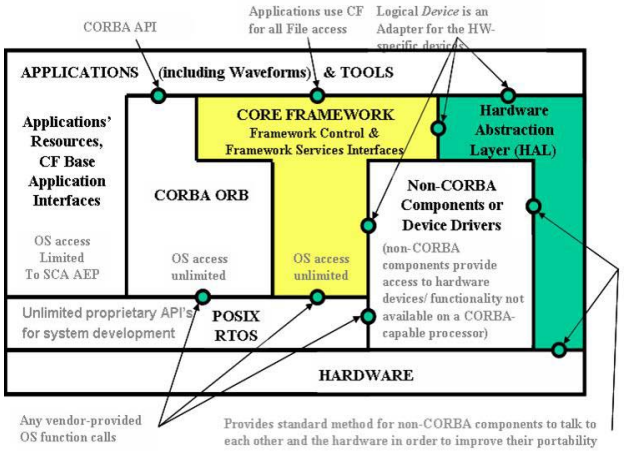

Figure 9. Architecture view showing interaction boundaries between SCA applications, core frameworks, operating systems, device drivers and hardware.

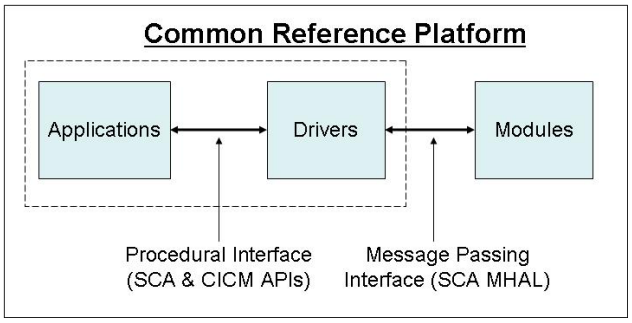

Figure 10. Common Reference Platform procedural and message passing interfaces between applications, drivers and modules.

Table 3. SCA 2.2.2 Published APIs (Source: JTRS)

SCA 2.2.2 API	Version	Date
AudioPortDevice	1.3.1.1	29-Mar-07
DeviceIO	1.0.1	29-Mar-07
DeviceIOControl	1.1.1	29-Mar-07
DeviceIOSignals	1.1.1	29-Mar-07
DeviceMessageControl	1.1.1	29-Mar-07
DevicePacket	1.1.1	29-Mar-07
DevicePacketSignals	1.2.1	29-Mar-07
DeviceSimplePacket	1.1.1	29-Mar-07
DeviceSimplePacketSignals	1.1.1	29-Mar-07
EthernetDevice	1.1.1	29-Mar-07
Packet	2.0.1	29-Mar-07
SerialPortDevice	2.0.1	29-Mar-07
VocoderService	1.1.1	29-Mar-07
JTRS_CORBA_Types	1.0.1	29-Mar-07

In addition to incorporating the SCA 2.2.2 public APIs (Table 3), the project should also study incorporation of CICM v0.3 APIs, and report back difficulties discovered in the development of the crypto APIs using the draft standard. The COTS software platform and tool components for the common reference platform should satisfy the objective of being platform and tools agnostic. A partial list of available COTS software platforms and tools useful for the reference platform have been assembled in Table 4.

Table 4. COTS software platform and tools components.

Platform	Tools	Component	Special Features
Operating System		Linux	Open source, POSIX AEP Compliant
Operating System		Green Hills Integrity	Real Time OS, Partitioning Kernel, MILS
Operating System		Wind River VX Work	Real Time OS, Partitioning Kernel, MILS
CORBA		ACE TAO	Open source
CORBA		OIS OrbExpress	Real Time, Minimum, MILS
CORBA		PrismTech eORB	Real Time, Minimum, MILS
	Core Framework	CRC SCARI II ++	SCA 2.2 compliant Core Framework(CF)
	Core Framework	Harris CF and dmTK	SCA 2.2.2 compliant CF, domain manager toolkit
	Programming Tools	Zeligsoft CE/CG	SCA 2.2.2 code modeling and generation
Board Support Package		quicComm	HAL with optional SCA 2.2.2 MHAL support

6. WAVFORM COMPONENT ARCHITECTURE

The Common Reference Platform also plans on deploying a common Waveform Component Architecture. The team will begin by surveying several reference waveforms provided to the JTRS Program in the past (Table 1), including the WNW and SRW networking waveforms. A composable, modular waveform component architecture (including both control and dataflow models) will be developed for future SCA 2.2.2 compliant waveforms (Figure 11) [10].

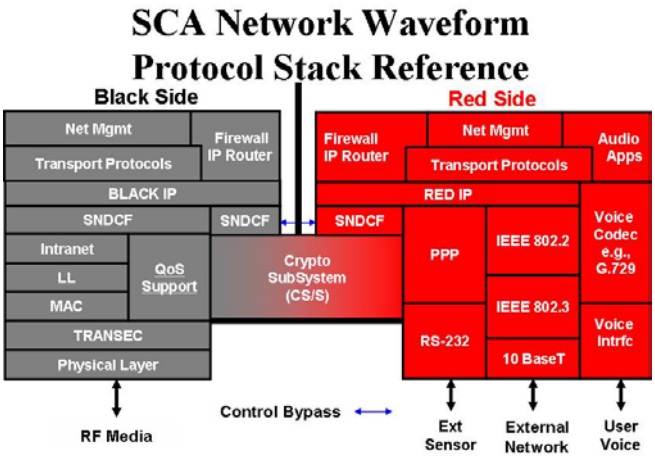

Figure 11. Networking waveform protocol stack reference model developed by SDR Forum Waveform Component Study Group. (Source: SDR Forum [10]).

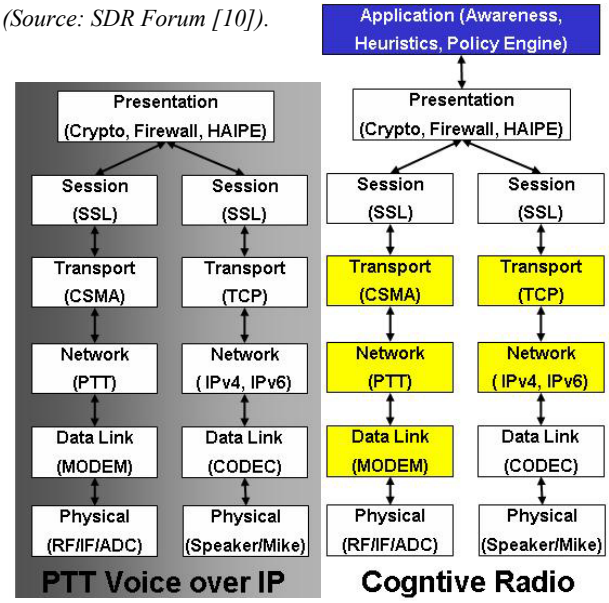

Figure 12. Push to talk (PTT) VoIP and Cognitive Radio use cases. (Source: SDR Forum [10])

The waveform model is intended to be useful in a number of waveform use cases [10] including push to talk (PTT) analog voice and voice over IP (VoIP) (Figure 12) and Cognitive Radio (Figure 12), as well as networking waveforms such as WNW and SRW.

7. SECURITY ARCHITECTURE

The vulnerability of wireless mobile devices and wireless networks "hacker" attacks has long been known to the commercial wireless community [11], and the software defined radio industry has been alerted to potentially serious compromises which can threaten the platform integrity of both mobile computing devices AND software defined radios [7].

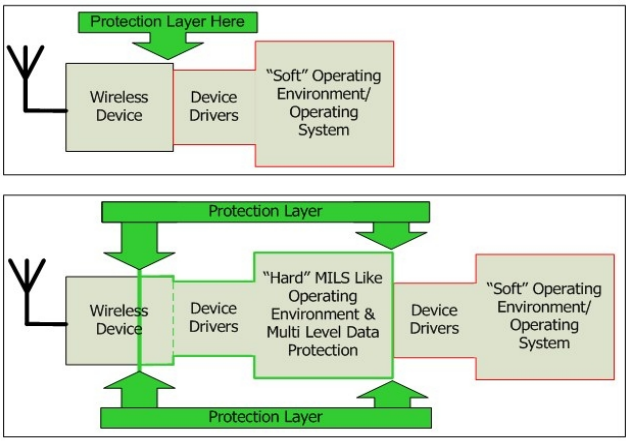

Figure 13. HAWCS™ architecture provides waveform developers with an approach for securing the SDR platform from wireless and network "hacking".

A potentially serious flaw in software defined radio security architecture has been discussed by Murotake and Martin [7], in which exposure of the network interface devices and their drivers to a "soft" operating system may seriously compromise the platform's integrity. The ability of the computing platform to be maliciously re-programmed or reconfigured, by-passing encryption, secure browsers and Virtual Private Networks through use of root kits/key logger Trojan horses and similar malicious software, may be easy to overlook. The patent-pending High Assurance Wireless Computing System (HAWCS™) security architecture employs specialized hardware, protected memory and kernel-mode techniques to isolate network interface devices and drivers from operating systems, files and applications using a defense-in-depth technique. It places security applications and component reconfiguration/programming channels within special partitions not easily reachable by network attackers, and accessible only by those with supervisor privileges. HAWCS™ can be used to correct vulnerabilities identified by JTRS CP295, "Exposed Black Side", and added to existing and future software defined radios and mobile consumer devices. The Common Reference Platform employs HAWCS™ (Figure 13) to

provide developers with a method of securing the computing platform when developing SCA compliant high assurance waveforms and networking software applications.

8. CONCLUSIONS

The development of a common reference platform for SCA compliant applications incorporating a Type-1 or Type-2 interface can begin by assembling COTS component hardware and software which has already achieved SCA 2.2 or SCA 2.2.2 compliance certification as a starting point. Remaining tasks in development of an open architecture platform, achievable by 1Q CY 2010 include:

- Integration, functional testing and certification pre-testing of a SCA 2.2.2 compliant "Black Side" development platform based on 3U CPCI COTS hardware and software. (Note: first platform is based on SSPI SDR-4000).
- Development of a new common interface prototype cryptographic development module using an exportable, software defined Type 1 encryptor.
- Integration of the new crypto development module in a simulated Red/Black SCA 2.2.2 compliant platform.
- Functional testing and certification pre-testing of the SCA 2.2.2 compliant reference platform, with integrated crypto development module and integrated HAWCS™ high assurance architecture.
- Open-architecture publication of common reference platform interfaces and APIs.

9. REFERENCES

[1] D. Murotake, "A future common reference platform for JTRS", SDR Forum SCA Workshop, Portland OR, 18 June 2007

[2] K. Maier, "Mapping waveforms to systems: what would a wideband networking waveform require?", *Military Embedded Systems*, October 2005

[3] D. Murotake, J. Oates & A. Fuchs, "Real-time implementation of a reconfigurable IMT-2000 base station channel modem", IEEE Communications Magazine, February 2000

[4] D. Murotake & S. Pearce, "Scalable architectures for computationally intensive software radio systems", Proceedings 27th General Assembly of the International Union of Radio Sciences (URSI'02), Maastricht, the Netherlands, August 2002

[5] T. Sterkel, "Joint tactical radio system (JTRS) wideband networking waveform (WNW)", Proceedings JTRS Industry Day, 2003

[6] R. Haskings, "COTS Software-defined radio and SCA compliance", Evaluation Engineering, August 2006

[7] D. Murotake & A. Martin, "A high assurance wireless computing system (HAWCS™) for software defined radio", Proceedings SDR Forum Technical Conference (SDR'06), Orlando Florida, November 2006.

[8] D. Lanz et al, "Common interface to cryptographic modules (CICM) Draft Version 0.3 – Interface standard to Type 1 and Type 2 cryptographic modules", MITRE MP070175, August 2007.

[9] Joint Tactical Radio System, "Software communications architecture specification – Final", Version 2.2.2, Space and Naval Warfare System Center, San Diego CA, 15 May 2006.

[10] D. Murotake and R. Leschhorn, "Report of Waveform Components Study Group to SCA Working Group", SDR Forum Technical Committee, Vancouver Canada, 22 June 2006.

[11] "Security Threats and Requirements; 3GPP TS 21.133 V4.1.0 (2001-12); 3rd Generation Partnership Project, Technical Specification Group Services and Systems Aspects"

AUTHOR BIOGRAPHY

Dr. David Murotake received his Bachelor and Master of Science degrees in electrical engineering and computer science from MIT, where he also received the Bachelor of Science degree in humanities and science majoring in English literature and creative writing. He later received his Ph.D. in Management of Technological Innovation from the MIT Sloan School. Dr. Murotake served a five-year tour of duty with the US Army as a Combat Engineer, Tactical Intelligence Officer, and Signals Intelligence Officer, beginning a 33 year career as an engineer and manager in the electronics industry. Before starting his own company, SCA Technica, he pursued a career as a systems architect, program manager and business developer at RCA, GE, Lockheed, Sanders and Mercury Computer Systems, Inc. He is a member of the SDR Forum Board of Directors.