



The commercial delivery of the first generation of Software-Defined Radios has shown that the technology is viable and scalable. In order to push into new markets and domains, Software-Defined Radio must meet the challenges of security, safety, and a smaller footprint.

As the first wave of Software-Defined Radios (SDRs) built for compliance with the U.S. Military's Joint Tactical Radio System (JTRS) Software Communications Architecture (SCA) becomes available, radio manufacturers continue to increase their use of COTS tools. Their goal is to reduce development and deployment costs while enabling more design options throughout the engineering process. Commercial companies are looking to leverage this substantial investment in technology and tools, in order to extend SCA-based SDRs to new domains and to new types of devices.

Various groups see SDR as a solution to these needs. The military needs *smart radios* that can flexibly work in whatever country they are deployed, since they may be interacting with local forces on different networks. Cell phone makers need to consolidate the multimode radios they are building into their handsets and provide bug fixes with downloaded software. Public safety officials need a way to enable interagency communications problems during a crisis.

Reprinted from Military Embedded Systems Summer 2006

The next advancements in Software-Defined Radio

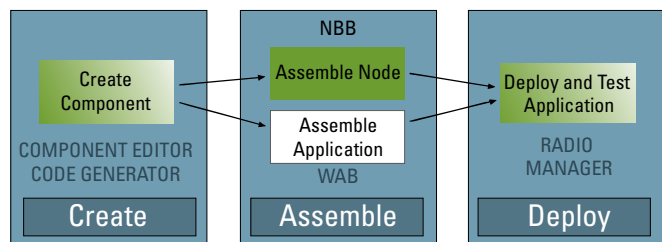
By Joseph M. Jacob

SDR defined

What is Software-Defined Radio?

In an effort to improve upon the flexibility, usability, and extensibility of radios, a variety of companies have been working for almost 20 years to create radios where the core functionality is implemented in software rather than hardware. A Software-Defined Radio (SDR) is a radio in which 100 percent of the modulation and demodulation is defined in software instead of being hardwired into the electronics. This means that the frequency band, performance, and functionality can be upgraded with a simple software download and update. In essence, an SDR is a radio that is substantially defined in software, with a physical layer behavior that can be significantly altered through changes to its software. SDR provides an efficient and comparatively inexpensive solution to the problem of building multimode, multiband, multifunctional wireless devices. An SDR is capable of being reconfigured to operate with different waveforms and protocols through dynamic loading of new waveforms and protocols. A waveform can contain a number of different parts: It may not be just *AM* vs. *FM* but also have security and safety characteristics built into the waveform itself. The figure below, courtesy of the Communications Research Centre of Canada, shows the steps in the Software-Defined Radio life cycle.

SDR Development Lifecycle



Software Communications Architecture

In order to achieve these goals, SDR technology needs to be further advanced to satisfy security, safety-critical, and footprint issues.

Security as a priority

Ensuring that radio communications are secure is one of the highest priorities for both military and commercial radio markets. In order for an SDR to be effective, the radio must implement robust security. In the military, security of communications on the battlefield is

simply a basic requirement. Without it, the radio is useless, even harmful.

Radio often provides the only means of communication in high-threat military environments. Unfortunately, military personnel have not yet been able to trust that radios will be effective at keeping multiple levels of classified and unclassified transmissions separate. They need to have confidence that secret communications on one channel intended for U.S. forces only will not bleed into

Copyright 2006

unclassified channels, or be compromised by hostile third parties. It is important to note that SDRs transmit not only voice but also data. Voice transmission becomes only a small part of the overall usage of SDRs.

Both in the military and commercial markets, wireless transmission of information will continue to grow. Whether browsing on the Web, transmitting video, sending private financial information, or simply sending E-mails, the data component usage of SDR is the most significant component, and the one that often requires the highest security levels.

For public safety, security of communications during a time of crisis is essential to ensuring that public responders can get their job done. In the commercial world, handheld cell phones or radios will be used to do a variety of different tasks, including speaking with friends, sending text or video messages to colleagues, and communicating with the bank to engage in financial transactions. Consumer confidence in these devices will depend in large part on the level of security that they provide. If that device is accepting E-mail and instant messages while transmitting credit card and bank information, consumers will want assurance that subversive code contained in an E-mail or IM will not have any effect on their private financial information.

Virtually all Software-Defined Radios use a Real-Time Operating System (RTOS) as the core operating system for the radio. A remarkable amount of work has been done during the past five years in collaboration with the U.S. Air Force Research Laboratory and the National Security Agency to create highly secure versions of some of these RTOSs. The result of this effort is the Multiple Independent Levels of Security (MILS) architecture (www.mils.us). The MILS versions of the RTOSs are called *separation kernels*. Currently, three different RTOS vendors have publicly announced separation kernels: Green Hills, LynuxWorks, and Wind River. Although the initial development and deployment of these separation kernels is

for defense applications, there is growing demand for these high-assurance MILS separation kernels throughout commercial domains as well.

The core security functionality of the MILS architecture is to keep data separate and to control information flows on a highly secure basis. The core foundational communications middleware for MILS, the Partitioning Communication System (PCS), ensures that only authenticated and authorized parties are allowed to exchange data, that their communications are secure and unbreakable, and that communications of data from different domains are kept separate over just one communications channel.

These separation kernels, as well as implementations of the PCS, are being certified to at least an EAL 6+ level under the Common Criteria, an internationally accepted security standard. This will be the first time that true COTS software is certified to such high levels. Both military and commercial radios will be able to use these validated, highly secure COTS components to build highly secure radios.

Table 1 shows the different common criteria evaluation levels and the rigor with which software must be developed and analyzed at each level.

MILS is a perfect match for SDR because it ensures a high level of security while enabling modularity of new capabilities, provisioning of bandwidth and channels, and backwards compatibility with legacy radios. It also supports dynamic intra- and inter-network routing of data transparent to the radio operator. A number of different military SDR programs around the world are beginning development of radios using the MILS architecture to guarantee highly robust security.

Safety-critical issues and requirements

Aircraft and avionics manufacturers are trying to achieve size, weight, and power savings by using multichannel, multiband Software-Defined Radios instead of separate radios for each waveform application. The software used in these radios will require certification by the Federal Aviation Administration in accordance with *DO-178B* at a level

Common criteria evaluation levels

Common criteria level	EAL description
EAL 1	Functionally tested
EAL 2	Structurally tested
EAL 3	Methodically tested and checked
EAL 4	Methodically designed, tested, and reviewed
EAL 5	Semiformally designed and tested – Must ensure resistance to penetration attacks with a moderate potential. Covert channel analysis and modular design are required.
EAL 6	Semiformally verified, designed, and tested – Must ensure resistance to penetration attacks with a high potential. The search for covert channels must be systemic. Development environment and configuration management controls are further strengthened.
EAL 7	Formally verified, designed, and tested – The formal model is supplemented by a formal presentation of the functional specifications and high-level design showing correspondence. Evidence of developer <i>white box</i> testing and complete independent confirmation of developer test results are required. Complexity of the design must be minimized.

Table 1

dictated by the impact to flight safety in the event of the radio's failure.

Although a great deal of the initial work for safety-critical SDRs is under the rubric of avionics systems, the results of this work apply to other safety-critical areas as well. These areas include medical devices, industrial process control, robotics, and so on, where communications via wireless devices are necessary and the flexibility of SDR is very useful, but where failure could lead to significant damage or even loss of life.

The FAA's Advisory Circular AC20-115B, produced by the Radio Technical Commission for Aeronautics (www.rtca.org), established DO-178B as the accepted means of certifying all new aviation software. The targeted DO-178B certification levels are either A, B, C, D, or E. Correspondingly, these DO-178B levels describe the consequences of a potential failure of the software: catastrophic, hazardous-severe, major, minor, or no effect. Table 2 shows the different DO-178B certification levels and the type of consequence if there is a failure at that level.

The generally accepted standard is that a normal aviation radio would be certified under Level C. However, if the SDR functionality will be implemented in a flight-critical component (for example, an instrument landing system), then the certification could be as high as Level A.

There are significant issues for certifying SDR for safety-critical applications. The primary issue is that the fundamental benefit of SDR, which is its flexibility in dynamically updating and changing the system (including waveforms), rubs against traditional safety-critical principles of maintaining static configurations. For example, the ability to dynamically load and unload a waveform in flight runs counter to the traditional safety-critical principle of having a static configuration for the radio so that new possible threats to the radio's integrity are not introduced in-flight. Rather than try to shoehorn SDR into the traditional DO-178B analysis, the Avionics Special Interest Group (SIG) of the SDR Forum is taking a somewhat different approach. As discussed below, the Avionics SIG is working with accreditation agencies to demonstrate how modern software tools and techniques can ensure that certain types of dynamic behavior can meet and exceed the strictest levels of safety-critical scrutiny.

There is currently an effort underway by U.S. and European aviation certification authorities (together with RTCA) to revise DO-178B. This project, referred to as *Special Committee 205*, aims to create DO-178C, which will revise and update DO-178B. The Avionics SIG is working with SC-205 as well as with the entities responsible for the Integrated Modular Avionics certification issues. The focus of this effort is to keep the flexible and

dynamic benefits of SDR, and to use modern software development tools and techniques to achieve the high level of safety-critical protection required for avionics. For example, code coverage tools, static code analyzers, and other new software testing tools can create a high level of assurance that flexible, dynamic object-oriented code meets Level A safety-critical protection. Likewise, the aforementioned high-assurance security techniques can potentially allow, for the first time, dynamic behavior previously prohibited. For example, loading and unloading waveforms in flight might now be allowed if a particular waveform has a high-assurance, FAA-digitally signed authorization certificate allowing it to be implemented dynamically into an SDR in-flight.

The immediate benefit of this work will be SDRs that can be used in military avionics where they might have a DO-178B/C requirement because of their use of civilian airspace, as well as in commercial avionics where the flexibility of SDR provides increased functionality for the aircraft. Work in this area will continue throughout 2006 and 2007 to create a set of standards that will allow SDRs to be certified to high levels of safety-critical standards.

Reducing the footprint and improving the performance of SCA radios

In order to ensure that SDRs can fit into smaller and smaller devices to extend their usability into new domains, there are a number of different efforts underway to reduce the size and improve the performance of SCA-based SDR. Smaller code size results in lower power requirements, since the device needs less processing power and less memory in order to achieve its functions. Two of the most significant ongoing efforts are SCA-Lite, and CORBA ORBs in an FPGA.

Adopting an SCA-based SDR architecture provides several benefits; among them are software re-use, well-tested COTS tools, field upgradeability, common hardware, and software platforms to reduce production cost. At the same

DO-178B levels

Level	Effect of anomalous behavior
A	Catastrophic failure condition for the aircraft (ex: aircraft crash)
B	Hazardous/severe failure condition for the aircraft (ex: several persons could be injured)
C	Major failure condition for the aircraft (ex: flight management system could be down and the pilot would have to complete it manually)
D	Minor failure condition for the aircraft (ex: some pilot-ground communications could have to be done manually)
E	No effect on aircraft operation or pilot workload (ex: entertainment features may be down)

Table 2

time, some developers of very small form factor devices have raised concerns about the cost of full SCA compliance in terms of size, cost, and power. A number of companies interested in SDR have expressed concern that the current versions of the SCA may not fit into the very small form factors they plan on using for their Software-Defined Radios. At the SDR Forum (www.sdrforum.org), the leading organization for builders and users of Software-Defined Radios with more than 100 organizational members, there is work underway to rethink certain aspects of the SCA. The goal is to reduce and modify the required SCA components for systems that require a significantly smaller footprint than standard radios. The result will be to implement an *SCA-Lite* core framework for very small form factor commercial applications while preserving core functionalities.

At the same time, regardless of which version of the SCA they are using, radio builders are finding that they can never have too much processing power. As waveforms become larger and more complex, many radio builders are running into a performance wall as they try to manage multiple large waveforms at the same time. Software vendors have responded by providing technologies to increase performance. One of the most important technologies is to implement elements of an ORB directly into IP blocks on an FPGA. This can increase performance 10 to 100 times for selected functions. Radio builders can use a standard ORB for the general purpose processor and

ORB IP blocks for selected functionality to significantly improve overall throughput of the radio. This will allow larger waveforms to be used in radios without overloading the processing power of a standard general purpose processor or DSP.

The next generation of SDR

As the SCA matures and radio builders acquire experience in developing and deploying SCA-based radios, they are already thinking about the next generation of SDR devices. In order to meet the performance requirements for increased waveform utilization while ensuring lower development and deployment costs, these radios will need to utilize the security, safety-critical, and footprint/power enhancements being developed and provided by COTS component vendors.



Joseph M. Jacob is Senior Vice President, Sales and Marketing, at Objective Interface Systems, Inc., and is co-chair of the Avionics

Special Interest Group at the SDR Forum. Since joining Objective Interface in 2003, Joe leads the company's sales, marketing, business development, and product management teams. Prior to joining Objective Interface, his

professional experience included leading the international business development and strategy function for America Online. He holds a B.A. from the University of Illinois at Urbana-Champaign with a double major in Economics and Political Science. He also holds a J.D. degree from Harvard Law School.

To learn more, contact Joseph at:

Objective Interface Systems, Inc.

13873 Park Center Road, Suite 360
Herndon, VA, 20171-3247
Tel: 703-295-6500
E-mail: info@ois.com

History of SDR

In the '90s, efforts began to intensify to find standards-based approaches to SDR. The U.S. military, through the JTRS program, worked with its allies to establish the internationally endorsed open Software Communications Architecture (SCA). This standard uses Common Object Request Broker Architecture (CORBA) on POSIX operating systems as a framework for the various software modules that define the radio's behavior.

As with any new, disruptive technology, the development of powerful SDRs has had its fair share of ups and downs. In the past few years, the technology has matured to the point where solid SCA-based SDR implementations in small form factors are now becoming available.

For example, Thales Communications recently announced the first SCA-certified radio for the U.S. military, and several more are to follow shortly in the pipeline.

In fact, a number of turnkey development platforms are now available, providing an integrated COTS development platform – hardware, operating system, ORB, SCA core framework, and SDR development tools – that allows researchers and developers to begin building waveforms immediately. This substantially reduces the time and risk of developing a working small form factor SDR. In order for SDR to thrive and prosper, it needs to adapt to users' demands for safe, secure, smaller, and more efficient implementations.