

TOOLKIT

DESIGNER'S TOOLKIT

Software-Defined Radios Make Their Move



BY TIM OWEN

Recent Advances Highlight The Technologies And The Solutions That Are Now Available For SDR Processing Systems.

THE term “software-defined radio” refers to the use of software-programmable hardware to provide flexible radio solutions. It is conventionally abbreviated as SDR (SEE SIDEBAR). The concept behind the technology is that it will provide software control of radio functionality. Traditional radio designs are constructed of fixed analog or digital components. Such designs also are custom built for each application. By comparison, SDRs offer an inherent flexibility. It serves as the main incentive to engage in an SDR methodology. It also makes it possible for an SDR approach to be applied to many radio-based applications.

Fundamentally, software defines the radio functionality. The use of a uniform hardware platform is therefore possible across multiple applications. To transform radio function and operation, SDRs also allow on-the-fly dynamic hardware reprogrammability. With a layered software structure and the adoption of hardware and software industry standards, it is possible to provide varying degrees of abstraction from the underlying hardware. This simplifies porting of the software to future hardware—a feature of particular importance to the defense industry, where systems often require multiple upgrades. The system lifecycle may exceed the availability and capability of hardware technologies.

Currently, SDR embraces a

varied scope of radio applications. It is becoming increasingly evident that it can be applied to systems that differ in the number of antennae, channels, and processors. For example, a high-performance direction-finding system may have eight receiving antennae, while requiring 64 digital downconversion (DDC) channels and 40 processors. A tactical military radio, on the other hand, may only require two processors and one each of the antenna, DDC channel, and digital upconversion (DUC) channel. To facilitate the use of a uniform software/hardware approach, it is therefore critical that both the hardware and software can be easily scaled with minimal limitations. This is especially true for the data interconnects between the processing devices.

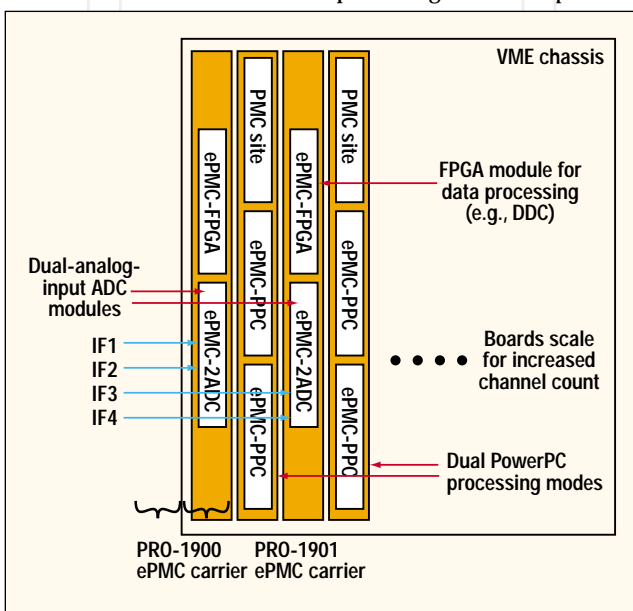
Toward that end, this article examines issues, available technologies, and solutions for typical SDR processing systems.

Thanks to the emergence of SDR, commercial-off-the-shelf (COTS) systems structured on standard backplanes, FPGAs, and processors have become a cost-effective implementation. With the reprogrammability of the SDR, it is no longer necessary to discard the entire system hardware as the system evolves technologically. System re-engineering is a more software-oriented task. In practice, however, it is likely that certain elements of the hardware (e.g., ADCs and DACs) will need to be replaced and upgraded to meet improved system specifications.

To ensure the disposal of the minimum amount of hardware at a system upgrade, the system must be constructed in both a modular and scalable fashion. In this way, only the necessary elements will require upgrading. This modularity and scalability also make the hardware applicable to a wider range of radio applications. In practice, this means a scalable data-inter-

connect mechanism is important. If the interconnects support a standard data-transfer protocol like TCP/IP, software porting to a new platform will be easier. Of course, the new hardware must support the same protocol.

New solutions have been developed to address the issue of SDR processing systems. One such solution hails from Spectrum Signal Processing. Known as the Solano Communications IC, this chip connects devices in a system by providing point-to-point data channels between the devices via low-voltage differential signaling (LVDS). It provides a total of four such dedicated data links, called *quicComm* links.



1. Shown here is an example of a HCDR implementation for a quad-IF input-receiver application.

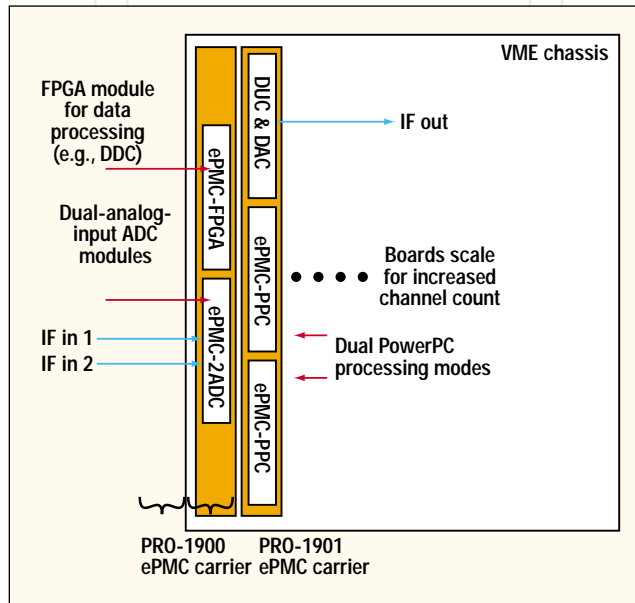
DESIGNER'S TOOLKIT

[SDR TECHNOLOGY]

Each of these links is capable of up to 200-Mbps full-duplex communication. The chip also has an internal direct-memory-access (DMA) engine to relieve the processor of the data-movement operation.

Solano provides each device with routable, high-speed, low-latency data paths to other system devices. It interfaces to each *quicComm* link as a simple memory location (FIFO). The chip can be used with processors, as well as with FPGAs, ADCs, and DACs with minimal, if any, interface logic. As a result, the processing heart of the radio system can be constructed with a common interconnect fabric. This possibility simplifies the overall system software by enabling the employment of a uniform software library. And it offers the potential to build heterogeneous processing architectures. Just use different processors, making sure that each one has a Solano interface.

A modular approach to SDR system construction can further build upon the 'point-to-point' data-link concept. The links can be routed off modules using extra connectors. They could even be routed off of a PMC by a connector that still complies with the PMC specification. This approach offers the enhanced capability to support direct data links to the module motherboard. It also provides an alternative data path to the standard PCI bus interface. The enhanced



2. The HCDR implementation illustrated here corresponds to a transceiver application.

PMC or ePMC module forms a flexible and scalable modular concept on which to design and construct systems.

An example of such a modular concept is illustrated in Spectrum's range of VME-based *flexComm* HCDR systems. These systems provide an ideal architecture for wireless applications. The system is assembled using carrier cards that support ePMC modules. The carrier card exists as either a single- (PRO-1900) or dual-slot (PRO-1900 and PRO-1901) VME64x board. It offers easily scalable modular I/O and processing by supporting up to five module sites. Four of these sites are interconnected with *quicComm* links, which are also available at the VME front panel and P0 backplane connector.

The result is data connectivity between boards within the chassis and even between chassis. The PRO-1900 has an embedded PowerQUICC processor for ePMC module control. It also hosts VxWorks for standalone operation. An optional RACE++ interface can be used as an alternative data path or for connection to legacy equipment. The flexibility of such an approach is shown in FIGURES 1 and 2. Both systems are constructed using the same hardware building blocks or modules.

The momentum of SDR in defense applications has been spearheaded by the United States Government's JTRS program. The program requirement calls for "affordable, high-capacity tactical radios that meet the bandwidth needs of various echelons. Therefore, a software-programmable and hardware-configurable digital radio system is required to provide increased interoperability, flexibility, and adaptability to support the varied mission requirements of the warfighters."

As part of the JTRS program, a specification for JTRS radios has been written. It is the Software Communications Architecture (SCA). To achieve adaptability and flexibility and be able to migrate radio waveforms across hardware platforms, technologies need to be adopted and supported by hardware. Examples include bus platforms, bus protocols, and high-level software. These technologies

GLOSSARY OF TERMS

The following is a glossary of the terms referenced in this article:

- ADC - analog-to-digital converter
 - COTS - commercial off-the-shelf
 - DAC - digital-to-analog converter
 - DDC - digital downconverter
 - DUC - digital upconverter
 - ePMC - Enhanced PCI Mezzanine Card
 - FPGA - field-programmable gate array
 - JTRS - Joint Tactical Radio System
 - LVDS - low-voltage differential signal
 - PCI - Peripheral Computer Interconnect
 - PMC - PCI Mezzanine Card
 - RTOS - real-time operating system
 - SCA - software communications architecture
 - SDR - software-defined radio
- For more information pertaining to software-defined-radio technology, check out the following Web sites:
- www.jtrs.saalt.army.mil/
This site provides general information on the Joint Tactical Radio System.
 - www.spectrumsignal.com/Products/solano.pdf
Go to this site to download a copy of the Solano IC datasheet.
 - www.sdrforum.org
The official SDR Forum Web site offers information into ongoing work on SDR technology, research activities, joint developments, and new products.

DESIGNER'S TOOLKIT

[SDR TECHNOLOGY]

must be based on industry standards.

It is important for the hardware to incorporate such standards. It can then support the high-level software required by the SCA (i.e., POSIX-compliant RTOS and CORBA). Engineers are already finding that the newest system architectures push the hardware bus bandwidth limits, especially when the bus is shared by multiple devices. Consequently, emerging standards now propose switched-packet backplanes, such as PICMG 2.16, RapidIO, and Serial RapidIO. Offering the point-to-point data transfer of high-bandwidth data leaves the standard backplane to deal with functions that require less bandwidth.

The engineer faces difficult decisions when selecting software-development tools. High-level tools improve development time and alleviate upgrading effort. Yet they do so at the expense of a more inefficient software implementation. Low-level software, on the other hand, tends to be closely coupled to the hardware. But it requires more intensive software development and upgrade effort. The SCA defines a high-level, layered software-stack architecture based on industry standards. It provides a high degree of abstraction from the hardware and hence the flexibility and portability prescribed by the JTRS program.

In SDRs that do not need SCA compli-

ance, the software developer might not want to be constrained to programming at a high software level. In such instances, it is desirable to have a layered software methodology. The software level can then be chosen. Ideally, the software can be designed using a mix of both high-level (e.g., RTOS and CORBA) and low-level software (e.g., optimized libraries).

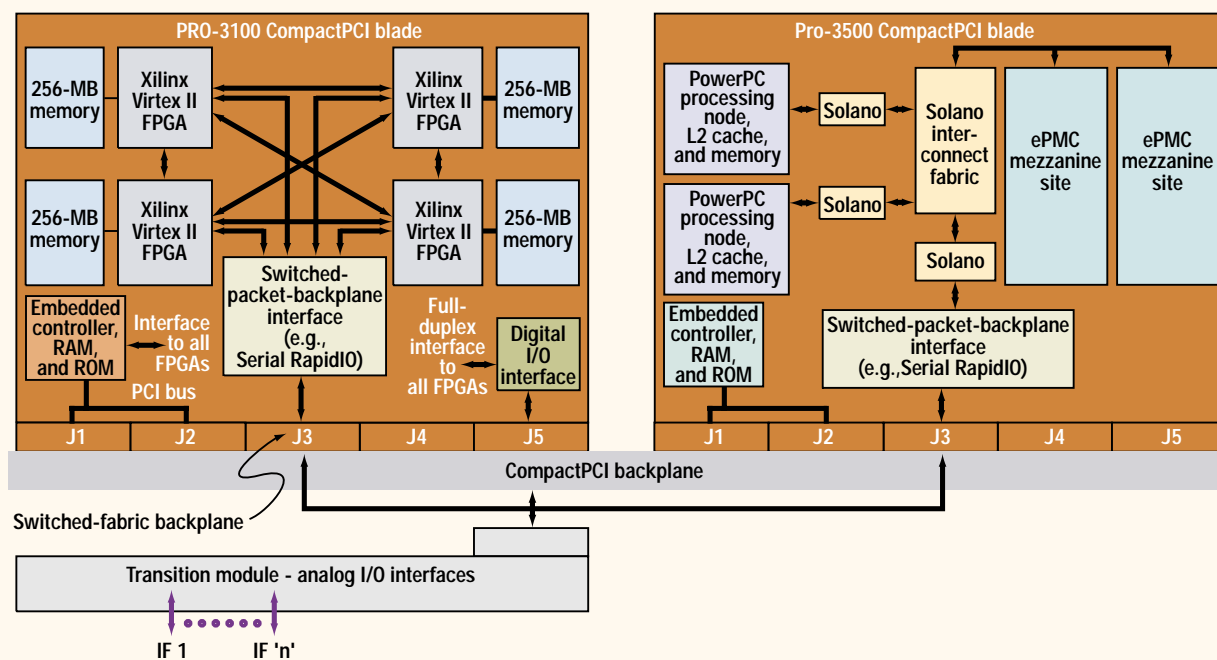
The software programming of processors offers the more traditional algorithm implementation. But in recent years, the FPGA has advanced significantly in speed, size, and gate density. In fact, it is now feasible to implement processor cores and DSP-based algorithms within a single FPGA. Algorithms that can be partitioned to process multiple signals in parallel can be more efficiently implemented in an FPGA device. Compared to implementation in a processor, such partitioning enables the parallel calculation of many more MACs in an FPGA device.

Certain FPGAs have specific internal logic to implement efficient DSP algorithms, such as the Xilinx Virtex-II and Virtex-PRO families. Realistically, all of the required processing of an SDR system can thus be accomplished in FPGA devices. By using a combination of off-the-shelf and custom FPGA logic, this approach offers the ultimate flexibility. A total FPGA processing solution also provides on-the-fly reconfigurability of

the system processing. At the same time, it preserves the hardware investment through the system lifecycle. Certain algorithms are still more efficiently realized in software on a processor, however. Subsequently, a SDR implementation uses a blend of processors and FPGAs.

Certain defense systems specify high-availability (99.999% uptime) and high-reliability requirements. High-availability systems are specified and employed in the commercial communications marketplace as well. These requirements can be applied to defense applications if they offer genuine advantages. A high-availability requirement dictates that hot-swap capability and redundant hardware are implemented to minimize system downtime. As a result, the SDR system needs to be based on the CompactPCI form factor—assuming the use of a standard form factor and backplane.

The CompactPCI specification provides for a transition module that connects to the rear of the CompactPCI backplane. It connects to the main CompactPCI board or 'blade' via the CompactPCI user-defined connectors. This transition-module printed-circuit board (PCB) can be used to mount I/O devices. Typically, this I/O hardware will vary for different SDR applications. In contrast, the FPGA and processing boards can remain common across a



3. Shown here is a detailed representation of a high-density, commercial-off-the-shelf (COTS), heterogeneous software-defined-radio (SDR) platform.

DESIGNER'S TOOLKIT

[SDR TECHNOLOGY]

wider range of applications. To provide versatility and scalability, the FPGAs and processing must be partitioned separately. Each can then be scaled independently to meet system requirements.

An example is Spectrum Signal Processing's SDR-3000 systems. Each system boasts a scalable, high-density heterogeneous platform of FPGAs and PowerPC processors that can be used for SDR applications. The PRO-3100 is a CompactPCI FPGA-based processing engine. It possesses four user-programmable, Xilinx Virtex-II FPGAs for the processing of high data rates. In a typical SDR system, this can accomplish the multi-channel DDC and DUC functions (FIG. 3). High-performance buses based on industry standards, such as Ethernet and Serial RapidIO, provide the high data throughput needed to meet the board's FPGA processing capability. An embedded PowerPC controller is present to host control software for the board resources.

The PRO-3500 is a PowerPC MPC7410-based CompactPCI board. It operates as a baseband-processing engine and has two embedded PowerPCs. It can support further processors by adding

modules to two ePMC sites that support *quicComm* links. These sites support ePMC modules and standard PMC modules for third-party interface support.

The high-bandwidth combination of *flexFabric* and *quicComm* provide the high data I/O rates required to meet the processing capabilities of the PowerPCs. The *flexFabric* is Serial RapidIO-based for a point-to-point connection between PRO-3x00 boards at up to 320 Mbps. The *quicComm* links serve as the on-board inter-device connection. A 405GP embedded controller also is provided for host control software.

Each board supports standards-based data interconnect. For instance, Serial RapidIO and *quicComm* links can support the TCP/IP protocol. The boards also house processors that support the use of a POSIX-compliant RTOS. Subsequently, the SCA software stack can be supported (including CORBA). SDR-3000 platforms can then be used for JTRS radio implementations.

SDR methodologies provide an excellent platform for the development, maintenance, and evolution of radio systems. Modular COTS platforms address the

hardware and key software building blocks. The platforms allow the system developer to focus his or her efforts on the radio waveforms. Yet the use of a uniform hardware interconnect and the correct combination of high-level and lower-level software remains critical. Genuine upgrade and time-to-market benefits can only be realized through the right combination of these parts. It also is fundamental for the data interconnect to have the bandwidth to support the system processing density.

As a technology, SDR is quickly becoming established. It is being increasingly embraced in the wireless defense market. In North America, this move is driven by JTRS. Yet the adoption of SDR should become more widespread. It is being bolstered by industry groups like the SDR Forum. Its members are working to promote the interest, development, and adoption of SDR as a standard. ■

Tim Owen, *Field Applications Engineer*,
Spectrum Signal Processing Ltd., 200-2700
Production Way, Burnaby, B.C. V5A 4X1; (604)
421-5422, FAX: (604) 421-1764, www.spectrum-signal.com.



**Wireless
Systems Design**
www.wsdmag.com

Your gateway site to

Planet EE
Penton Electronics Group