

COTS Journal

Software Radio Modems Enter the SoC Era

By: Jean Belzile, Chief Technology Officer, ISR Technologies, Steve Bernier, Project Leader, Communications Research Centre, Manuel Uhm, DSP Marketing Manager, Xilinx

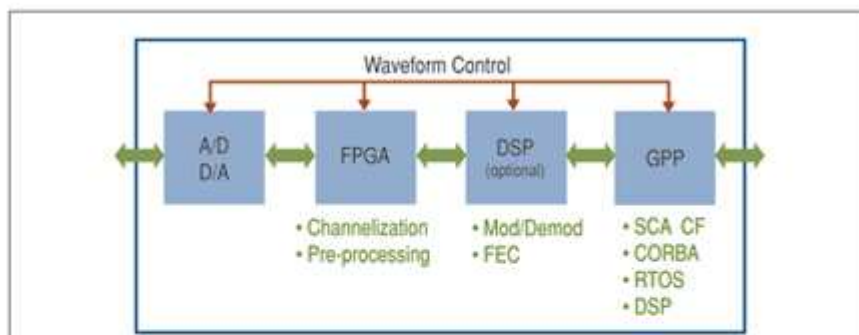
The powerful embedded processing capabilities of today's FPGAs make it possible to squeeze an SCA software radio into a single SoC.

Many software defined radios today, especially military radios, use the Software Communications Architecture Core Framework (SCA CF) as the application framework to manage the radio, its resources and the waveforms running on the radio. Until recently, a discrete general-purpose processor was needed to instantiate the SCA on the modem. This would typically supplement an FPGA, which would be used for pre-processing, and a digital signal processor, which would be used for modem and forward error correction (FEC) processing.

The technology is now available to create SCA-enabled software defined radio (SDR) Systems-on-Chips (SoCs) thanks to the availability of FPGAs with embedded general-purpose processors. This approach can result in significant power and cost savings through integration, as well as freeing up board space for other components or better thermal management.

SoC vs. Discrete Processor

Figure 1 illustrates the current model for implementation of an SCA-enabled SDR modem. An A/D receives a signal at intermediate frequency (IF). After digitization, an FPGA is used for channelization and pre-processing functions, such as digital down conversion. Baseband data is then passed to a DSP or general-purpose processor for demodulation and forward error correction (FEC). In the meantime, the general-purpose processor is controlling the modem and its active waveforms via the SCA CF. The SCA CF leverages the standards-based functionality of a CORBA Object Request Broker (ORB) for message passing, and a POSIX-compliant real-time operating system (RTOS) for scheduling.



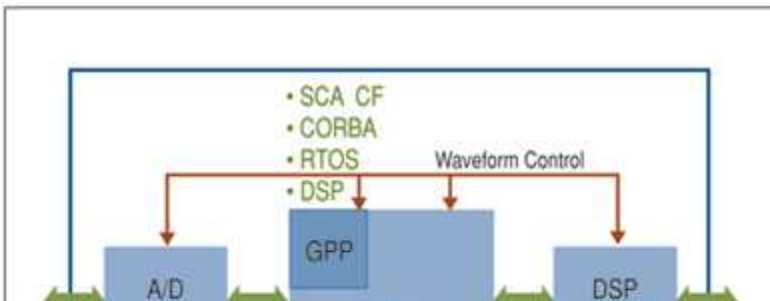
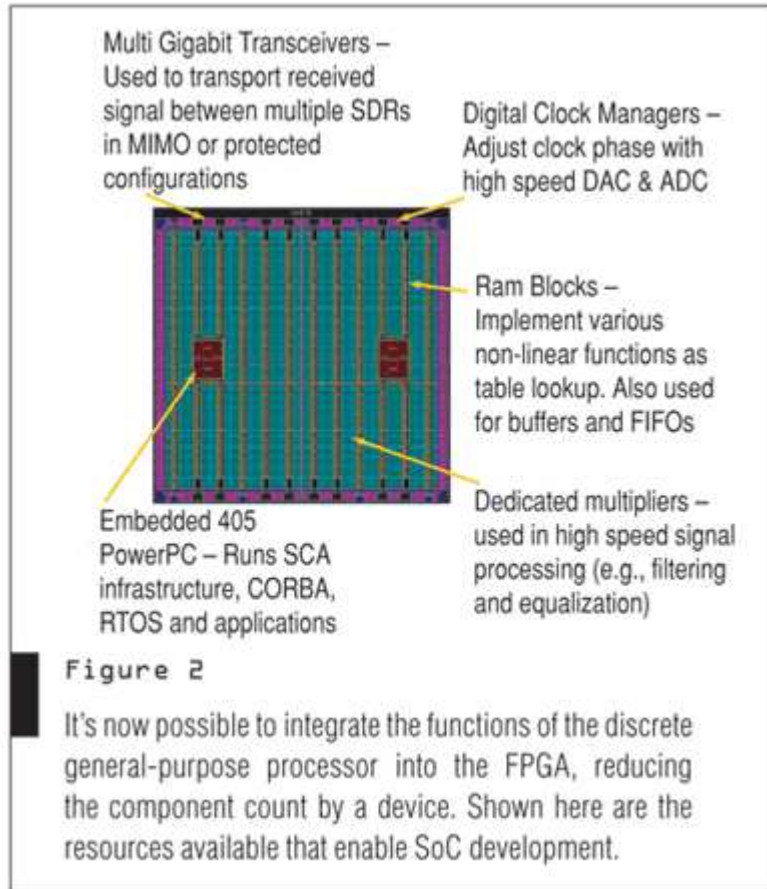
The model shown in Figure 1 uses two or three discrete processing devices per channel in an SDR modem. However, current technology allows for a different approach to the problem using FPGAs with embedded general-purpose processor cores, an example of which is the Xilinx Virtex-II Pro family of FPGAs with one or two embedded IBM 405 PowerPC cores. Each core provides 600 DMIPS of performance at 400 MHz. Hence, it is now possible to integrate the functions of the discrete general-purpose processor into the FPGA, reducing the component count by a device. Figure 2 shows the resources available that enable SoC development. Using such an SoC approach results in the

Figure 1

Shown here is the current model for implementation of an SCA-enabled SDR modem. An A/D receives a signal at IF. After digitization, an FPGA is used for channelization and pre-processing functions, such as digital down conversion. Baseband data is then passed to a DSP or general-purpose processor for demodulation and FEC. The processor controls the modem and its active waveforms via the SCA CF.

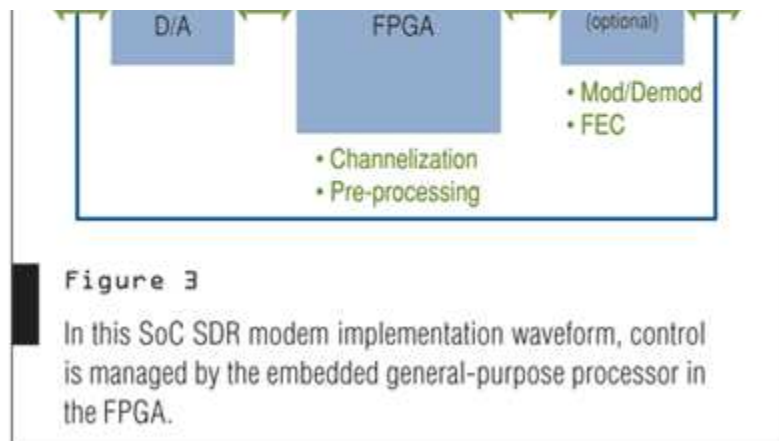
implementation illustrated in Figure 3, where waveform control is managed by the embedded general-purpose processor in the FPGA.

Benefits of FPGA



Integration

Lowering power consumption and cost are the major benefits of integrating the general-purpose processor functionality into an FPGA. An additional benefit is freeing up board space that can be used for other critical components, or to improve the board layout for thermal flow. These are major issues that impact the rate of deployment of SDRs. Clearly, cost is a factor in both SDR development and deployment. In the world of commercial telecommunications, a one percent decrease in the cost of terminals or base stations is considered significant.



Power consumption is a critical parameter, especially for radios that depend on batteries. Lower power consumption results in handheld radios with longer standby and talk times. In the military realm that translates as longer missions and easier logistics. Even radio infrastructure deployed on fixed, airborne or shipborne sites typically have strict power budgets that must be maintained. Commonly used general-purpose processors in current SDRs include the IBM 4xx and Motorola MPC85xx PowerPC families. These devices typically consume 2 - 6.4W depending on frequency, peripheral utilization and so on.

In contrast, usage of the 405 core at 400 MHz in a Virtex-II Pro consumes less than 0.4W. This is a result of reduced supply voltage, less capacitance due to a smaller geometry and not having to drive external board traces. While peripherals need to be added to the 405 core increasing the total power consumption, this is still substantially

lower than using discrete general-purpose processors.

Furthermore, eliminating the discrete general-purpose processor can result in savings of over one hundred dollars. While this may not seem like much at first glance, consider the savings in an N-channel radio where the architecture from Figure 1 is being repeated N times. Consider also that the reduction in board space translates to supporting far more channels in far less space.

An SCA-Enabled SDR Modem

An SCA-enabled SDR typically is comprised of the following elements:

Hardware: In order to be power-efficient, a mix of heterogeneous processing capabilities is required. For example, designers depend on the availability of general-purpose processor-type functionality for most management, infrastructure and intelligent algorithms. They also want power-efficient processors for digital signal processing algorithms. In general, DSPs or general-purpose processors are used for low to medium rate signal processing where C coding is possible. Finally, they rely on dedicated logic for high-performance signal processing that extends beyond the capabilities of a DSP or general-purpose processor.

RTOS: An RTOS is required for strict interrupt response and scheduling, as well as memory protection. Multiple waveforms will need to co-exist in an SDR without being corrupted or corrupting another waveform.

Middleware: CORBA frees the designer from messaging constraints and elevates the level of the code to the core functionality by abstracting the communication mechanisms between processes.

SCA CF: The Core Framework provides a number of basic infrastructure services to the application providers and facilitates the installation and removal of waveforms. It can even go as far as providing a mechanism for validating the platform's ability to run a waveform.

For its part, ISR Technologies has developed an SCA-enabled SDR modem known as a Software Defined Indoor Unit (SD-IDU). The focus of the platform is on

medium to high capacity waveforms, but lower capacities are certainly feasible. Based on a Xilinx Virtex-II Pro FPGA, the embedded 405 provides the processing horse power required. The SCA operating environment is based on the INTEGRITY RTOS from Green Hills. The middleware is represented by Objective Interface System's ORBexpress. Finally, the SCA CF Version 2.2 is supplied by CRC. This combination of software and hardware was selected for performance, flexibility, growth capability, evolution and optimization of development and production costs.

The SD-IDU contains all of the multiplexer, controller and modem functions. Depending on the waveform, typical power utilization can range from 2 - 9.5W. This platform has the ability to generate and receive waveforms of up to 50 MHz of bandwidth. The bit rates supported can easily reach over 200 Mbits/s while the choice of modulation, coding and filtering is only limited by the computational power of the platform. At a sustained rate of over 20G multiplies per second, these processing capabilities are considerable.

Design and Development Considerations

Clearly there are many considerations that have to be taken into account when crafting an SCA-enabled SDR SoC Architecture. These divide somewhat neatly into SoC, BSP, RTOS, CORBA ORB and SCA CF areas of concerns. When choosing an SoC implementation designers need to look at performance, availability of high-speed I/O and whether there's enough logic left in the SoC after implementing the various interfaces and internal infrastructure to do waveform processing.

The key recommendation that can be made regarding the creation of a BSP for an SoC is simply the availability of industry standard devices. A widely available CPU core is the first key component. Secondly, peripheral cores that emulate the look and feel of standard ICs are also very important. For example, Xilinx and IBM have a suite of cores and an architecture that is supported by a number of vendors. This greatly reduces the risk associated with the creation of a BSP from scratch, and also considerably reduces the "time to boot" and hence, time-to-market of the system.

In an SCA-enabled architecture, the RTOS must support a very specific set of POSIX calls. The RTOS should also provide a standard TCP/IP stack on which an ORB can run. A clean architecture, an efficient and powerful memory/resource management scheme, documentation, and excellent tools and support are also important, but ultimately in a real-time system it should come down to interrupt response time, task switching and other key parameters.

The ORB has to support the minimum CORBA specification. Important middleware aspects to consider in real-time systems are call efficiency between processes within the same processor and between different processors. Memory footprint is also an important consideration.

Typically the porting of the DomainManager, ApplicationFactory, Application and all the support/utility code is mostly ORB related as they only use simple C++ features. In contrast, an ExecutableDevice generally utilizes advanced C++ features offered by an RTOS. Therefore, porting of an ExecutableDevice from a discrete processor to an embedded processor can be more difficult. Using a POSIX-compliant RTOS makes this process a lot easier.

SCA in Action

CRC ported their SCA CF from their platform to ISR Technologies' SD-IDU. CRC's platform used a Pentium processor running Linux and the ACE TAO CORBA ORB to instantiate the SCA CF. From this environment, the SCA CF was ported to the embedded 405 processor of the Virtex-II Pro running Green Hills Software's INTEGRITY and ORBexpress. In order to do the port, some changes were necessary to the SCA CF and the BSP to support more portable ORB features for independence from the RTOS and ORB. The development of the above guidelines was the result of this effort. Implementation of the changes took

two man-days of effort. After that minimal effort, the SCA CF was recompiled and successfully ran on the embedded 405, resulting in the creation of an SCA-enabled SDR SoC.

Simply put, the technology exists today to create SCA-enabled SDR SoCs that enables a new model for developing SDRs. This model is cheaper to implement and consumes less power than current models. Nevertheless, careful consideration still needs to be taken when implementing such an architecture. Fortunately, the hardware abstraction enabled by CORBA and POSIX assists tremendously in easing the migration from previous models to the SoC model. Following the guidelines outlined here will enable a smooth transition to the SCA-enabled SDR SoC model.

Xilinx
San Jose, CA.
(408) 559-7778.
[www.xilinx.com].

© 2009 RTC Group, Inc., 905 Calle Amanecer, Suite 250, San Clemente, CA 92673