

# Software-Defined Radio: The New Architectural Paradigm

Reduce system power and cost with a shared resources SoC.

by Manuel Uhm  
Senior Marketing Manager  
Xilinx, Inc.  
[manuel.uhm@xilinx.com](mailto:manuel.uhm@xilinx.com)

Software-defined radios (SDRs) have already become a reality in the defense industry through programs such as the Joint Tactical Radio System (JTRS). Because SDRs are already being deployed, why would a new architectural paradigm be of interest? The reason is simple: the power consumption and cost of first-generation SDRs is generally too high for widespread deployment.

Despite Moore's Law, incremental process improvements are not sufficient to reach the power and cost restrictions for small-form-factor, handheld, and manpack radios, such as those required for JTRS Cluster 5. In addition, architectural changes are required to reach the strenuous requirements.

Using partially reconfigurable platform FPGAs as an SDR system-on-chip (SoC) addresses both of these issues by decreasing the number of DSP components in an

SDR while still providing the necessary functionality. In this article, I'll discuss how to apply this revolutionary technology to an SDR black-side modem.

## Architectural Improvement #1: SCA-Enabled SDR SoC

Current SDR modem architectures, such as that of JTRS Cluster 1, utilize a discrete general-purpose processor (GPP) to manage the application and control infrastructure, known as the Software Communications Architecture (SCA) Operating Environment (OE). The GPP is coupled with an FPGA for channelization and wideband waveform processing and a DSP for narrowband waveform processing.

The SCA OE comprises a POSIX-compliant real-time operating system (RTOS) for scheduling and memory protection, a CORBA (Common Object Request Broker Architecture) ORB for message passing, and an SCA Core Framework (CF) for loading and tearing down waveforms.

With the availability of platform FPGAs such as Xilinx® Virtex™-II Pro

and Virtex-4 FX devices that incorporate a hard-core PowerPC™ 405 GPP, it is now possible to run the entire SCA OE in the same FPGA that is required in the modem for channelization and wideband waveform processing. Because the 405 has a memory management unit (MMU), it is possible to run a full RTOS and ensure memory protection, unlike a soft-core GPP. This results in an SCA-enabled SDR SoC.

I/O accounts for the most power consumption in an SDR modem; thus it is important to reduce the I/O count as much as possible. Removing the discrete GPP from the modem does just this, thereby having the primary benefit of reducing the power consumption of the modem. Furthermore, the 405 hard-core is power-efficient, providing 600 DMIPS at 0.4W.

Secondary benefits include potentially lower cost by the removal of the discrete GPP and a reduced signal processing footprint, which can result in a smaller form factor or better thermal dissipation through superior board layout.

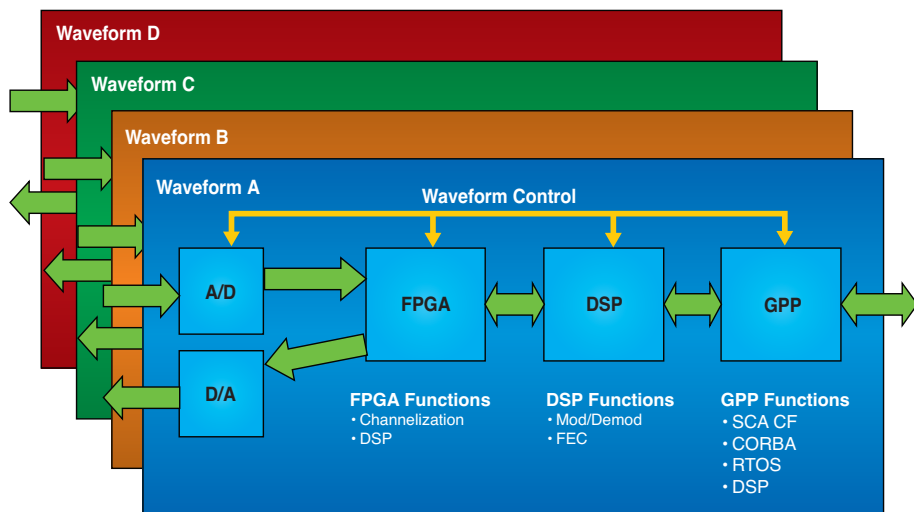


Figure 1 – Current SCA-enabled SDR modems use a dedicated set of signal processing hardware for each channel. The more channels the SDR must support, the more hardware it contains. This has a direct impact on power consumption and cost.

## Architectural Improvement #2: Shared Resources

The current architecture for implementing an SDR modem is known as a dedicated resources model. It is called dedicated resources because a set of processing resources is dedicated to a radio channel (where each channel is capable of running a waveform, such as the Single Channel Ground and Airborne Radio System [SINCGARS]). In this case, the processing resources consist of an A/D, D/A, FPGA, DSP, and GPP. To implement an N-channel radio, N sets of processing resources are required. This is illustrated in Figure 1 for a four-channel SDR modem supporting an SCA CF.

From a functional perspective, this architecture is sufficient for radios with less power and size-constrained environments, such as Cluster 1 vehicular radios. However, it is an inefficient usage of the available processing resources, resulting in excess power consumption and cost. For example, the signal processing parts for all channels of the radio must be selected for the worst-case scenario; the processing resources must be able to support the largest waveform (WNW, the Wideband Networking Waveform) such that if only a small narrowband waveform like SINCGARS is instantiated, most of that channel's processing resources are not utilized. This has a significant impact on driving up

the cost of the modem. Obviously, the problem gets exacerbated as you scale the model further. JTRS AMF (the JTRS Cluster for Airborne, Maritime, and Fixed installations) requires some radios to support eight channels. This also has an impact on power consumption.

A more efficient architecture for an SDR modem is referred to as a shared resources model. Unlike a dedicated resources model, this architecture offers the capability to support multiple waveforms across a single set of processing resources, allowing for much more efficient usage of the resources. The number of waveforms that can be support-

ed is a function of the size of the waveform and the size of the available processing resources. Figure 2 illustrates how a multi-channel SCA-enabled SDR modem could be implemented using this architecture. In this instance, the signal processing part count has decreased from 20 to 4 components. Hence, implementation of these architectural advantages can result in a production cost and power consumption that is two to three times lower than the dedicated resources model.

You will also notice that the FPGA is capable of doing all the heavy digital signal processing. The embedded GPP is also a natural fit for the light signal processing, such as synchronization loop control, as well as the upper protocol layers such as link and network layers.

It is also worth noting that both architectures illustrated here are using 100% commercially available components. In the shared resources model, the SoC FPGA can be a mid- to large-sized Virtex-II Pro or Virtex-4 FX FPGA with an embedded PowerPC core.

## Partial Reconfiguration: The Enabling Technology

The technology that enables the shared resources model is partial reconfiguration of the FPGA. Partial reconfiguration enables an application or component, such as a waveform or waveform component, to

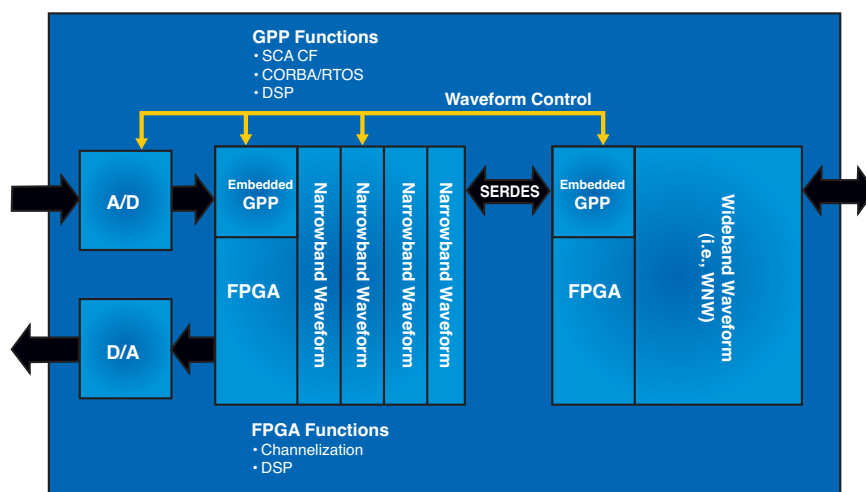


Figure 2 – A multi-channel SCA-enabled SDR modem architecture using a shared resources model. In this example, 5 channels supporting 1 wideband waveform and 4 narrowband waveforms have been implemented, while reducing the signal processing part count from 20 to 4 components compared to a dedicated resources model.

be dynamically loaded or unloaded in a portion of the device while other portions are either being used by other applications or going unused. This allows support for multiple independent applications concurrently in a single FPGA, which is somewhat analogous to dynamic task switching of a GPP. Without this capability, it would be necessary to reconfigure the entire FPGA to support a different application, which would result in the loss of all previous applications.

For example, if an FPGA was configured to support a SATURN comm link, it would have to be fully reconfigured to support an HAVEQUICK comm link, therefore resulting in the loss of the SATURN link regardless of how much leftover logic there was in the FPGA. Clearly this is unacceptable for a radio. Furthermore, partial reconfiguration enables an adaptive waveform to be supported in a smaller FPGA because the FPGA can dynamically load and switch between waveform components, rather than having all possible waveform components loaded at runtime.

Three basic elements are required to support partial reconfiguration in an FPGA:

1. An FPGA that inherently supports partial reconfiguration, such as the Xilinx Virtex family. The Virtex family is frame-reconfigurable, meaning that individual frames of logic within the device can be dynamically reconfigured independent of the rest of the frames. In Virtex-II Pro devices, a frame consists of a column, while in Virtex-4 FPGAs, a frame consists of a 16 x 1 configurable logic block (CLB) "tile."
2. At least a basic controller must be available to dynamically manage the reconfiguration of the FPGA. This could be an embedded GPP, a soft-core GPP (such as the Xilinx MicroBlaze™ processor core), or an external GPP con-

nected to the FPGA. In the shared resources model example, the same embedded GPP that is running the SCA OE is also managing the partial reconfiguration of the FPGAs.

3. Partial reconfiguration software development tools that support the development of applications, restricted to boundaries complying with the hardware architecture of the FPGA.

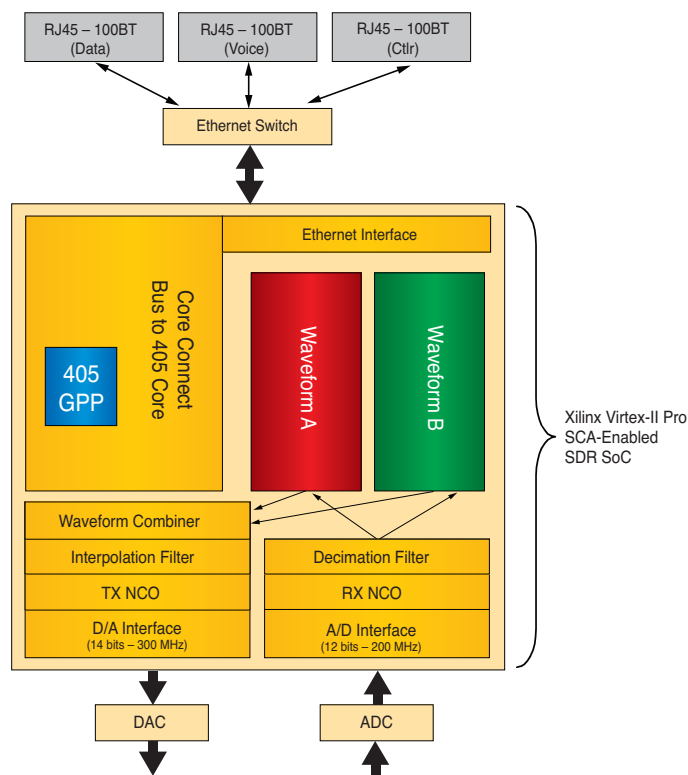


Figure 3 – Floorplan of a Xilinx Virtex-II Pro-based SCA-enabled SDR SoC supporting shared resources through partial reconfiguration. The yellow areas represent static infrastructure and do not change from waveform to waveform, whereas the rest of the FPGA is available for partially reconfigured applications.

Although elements #1 and #2 have been around for some time, it is only recently that software development tools enabling partial reconfiguration design have become available. Standard tools from Xilinx are available by request for Virtex-II Pro today and for Virtex-4 in the fourth quarter of 2005.

### Implementing the Architecture

The shared resources model using an SCA-enabled SoC has been proven to work today in a COTS (commercial-off-the-shelf) Xilinx Virtex-II Pro-based SDR modem from ISR Technologies. The demonstration system uses two modems,

each supporting two independent applications: a narrowband, 256 Kbps waveform supporting a comm link between two VoIP phones; and a wideband, 1,024 Kbps waveform supporting a streaming video link between two laptops.

Using a COTS SCA CF from the Communications Research Centre, the video link can be instantiated and torn down while maintaining the comm link – and vice versa.

More details on the demonstration can be found in the December 2004 JTRS JPO Technology Awareness Bulletin, published by the JTRS Joint Program Office at [http://jtrs.army.mil/sections/technicalinformation/fset\\_technical.html](http://jtrs.army.mil/sections/technicalinformation/fset_technical.html).

Figure 3 illustrates the floorplan of the Virtex-II Pro device in each of the modems. The yellow areas represent static infrastructure, as they do not change regardless of the waveforms being supported. This includes the digital down and up converter, internal shared buses (the Core Connect bus for the embedded PowerPC 405) and the interfaces to external devices, such as the A/D and D/A. The two applications (waveform A and B) run independently in the partially reconfigurable region in the right-hand side of the device. If necessary, a larger waveform or more smaller waveforms could run in the same space.

### Conclusion

Power consumption and cost are issues that are preventing widespread deployment of SDRs today, particularly in size, weight, power, and cost-constrained environments. Architectures that incorporate SCA-enabled SDR SoCs and a shared resources model can help to address these issues by providing the most efficient SDR modem implementation, thereby driving down the power and cost. ●●●