A Reconfigurable Software Digital Radio Architecture for Electronic Signal Interception, Identification, Communication and Jamming

Software programmable radios, such as those planned for use in JTRS, are best implemented by a balanced architecture using CPUs, FPGAs and a high-speed interconnect such as RapidIO.

Frank Van Hooft, System Engineering Manager
and
Mohammad Darwish, Senior Technical Lead;
Spectrum Signal Processing

Since the early days of backpack AM radios, military planners have sought more efficient methods for wireless battlefield communications. This relentless trend has resulted in today's reality of *hundreds* of different signals across the electromagnetic spectrum, varying from simple narrowband modulations to highly complex secure wideband schemes. A large percentage of these signals are used for communications; however, applications such as range and direction finding, navigation, remote imaging and control also share the spectrum.

Traditional methods for dealing with all these signals are breaking down. Today, individual pieces of equipment are used for each signal type. A MILSTAR satellite radio is different from that of a UHF tactical radio, which is different again from an IFF (identification, friend or foe) transponder or a GPS receiver. This fragmentation results in the requirement for multiple pieces of equipment consuming large amounts of space, weight and power, greatly limiting the areas in which they can be applied.

This issue is significant in a tactical situation where the need may exist to work with perhaps a half-dozen signal types. But it becomes almost overwhelming on a strategic scale, because in a conflict situation the problem instantly multiplies in scope. Friendly signals from one's own and allied forces must be received, transmitted and secured from interference. This is a relatively predictable task. However, hostile signals must be located, identified, intercepted, decoded, range and direction found, and blurred or jammed as required. Moreover, these signals may span the range from simple VHF communications to gigahertz frequency-agile radar pulses. For an army to gain information superiority on the modern battlefield, every one of these signals must be processed and responded to in real time.

This becomes an impossible problem using traditional techniques. An entire AWACS loaded with racks of equipment would still be unable to cover all of the signals that may conceivably be used, as the number of signals of interest may range from several hundred to over one thousand. Clearly what is needed is a totally new way of dealing with these signals. It was for this reason that the Joint Tactical Radio System (JTRS) program was introduced. This program defines a standardized way of utilizing Software Defined Radio (SDR) architectures so these problems can efficiently be addressed. In so doing it has also become a driving force for the adoption of SDR in military wireless applications.

SDR Primer

The basic premise behind SDR is to use digital signal processing techniques in place of today's predominantly analog signal processing. By replacing the hardwired analog circuits with reprogrammable software, a "digital radio" is able to change its personality on-the-fly, switching from one modulation method to another in the time a processor takes to branch to a new instruction. A single piece of equipment becomes capable of handling a multitude of different signal types (waveforms), providing a drastic reduction in the space, weight and power requirements for the system. Additionally, by building upon Moore's Law in the semiconductor space, rapid improvements in performance and size can be reliably predicted for future generations of equipment.

An SDR radio's rapid reconfigurability offers other advantages. Ultra-fast hop waveforms can be widely utilized. Cryptographic keys can be updated in real time, from today's 4-hour update rate to several-minute or even faster rates in the future. Radios can even be updated remotely, while deployed, as circumstances demand without having to be removed from service and shipped to a maintenance facility.

Implementing a true wide-bandwidth digital radio is a significant challenge. Consider the basic block diagram in Figure 1. A wideband RF converter translates a portion of spectrum from some super-high frequency (for example, 3 GHz) down to a range suitable for the A/D and D/A converters to handle. The better the radio, the wider the bandwidth this portion of spectrum will be ("Analog IF" path in the diagram), and hence the faster the A/D and D/A converters must operate. The speed of these A/D and D/A converters thus dictates perhaps the most basic characteristic of the radio: its bandwidth, and the resulting data rate ("Digital IF" path in the diagram) is a critical factor in determining the requirements for the symbol and bit rate processing blocks.

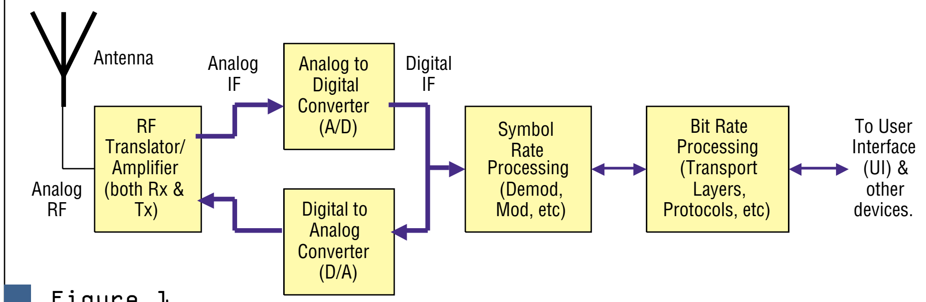

Figure 1

Basic SDR Block Diagram, showing the relationship between the analog and digital components in the radio.

	FPGA	Processor
Strengths:	➢ Highly parallel architecture permits multiple operations to be performed simultaneously. ➢ Embedded multipliers and RAMs enable extremely fast filters and synthesizers, good for many signal-processing operations. ➢ Programmable data path widths eases interfacing to and processing a wide variety of data types.	➢ Highly versatile; can implement an almost limitless number of applications. ➢ Embedded math logic makes for an efficient use of die resources. ➢ Excellent for decision making and branching. Well suited for implementing protocol stacks. ➢ Relatively low cost.
Weaknesses:	➢ Relatively inefficient for branching or decision making operations; these consume large numbers of gates. ➢ Large and fast devices are expensive. ➢ High-precision math operations difficult and may consume many resources.	➢ May be comparatively slow at simple fixed-point math operations, due to inherently serial processing nature. ➢ Only able to perform a low number of tasks at a single time (i.e. per clock cycle).

Table 1

Comparison of FPGA and processor technologies. In a software defined radio application, combining both types of processing elements yields a powerful, flexible architecture.

Consider a 14-bit A/D sampling at 105 MSPS and a 14-bit D/A outputting at 200 MSPS. This provides an instantaneous Nyquist bandwidth of 50 MHz on the receive side and 100 MHz on the transmit side. Zero padding each to 16-bits, these two components alone require an aggregate data rate of 610 Mbytes/s. Applications such as direction finding (DF) or beam steering require at least two of each; the aggregate data rate is now greater than 1.2 Gbytes/s. This is a minimum number. A very high-performance system bus is essential to reliably move data at such high sustained rates.

Processing this "firehose" of data is a non-trivial task. No single general-purpose DSP on the market today can process data at this rate on a sustained basis. Some specialized ASICs are capable of these data rates; however, their limited and fixed functionality rules them out when reconfigurability is required. An architecture is required that can partition this data out to multiple processing elements, then combine their ensuing results together for further down-stream processing. Processing elements must be selected based not only on their processing throughput, but also upon their capability for rapid (sub-2 second) reconfiguration.

SDR Reconfigurable Technologies

There are two applicable reconfigurable technologies available today: (1)

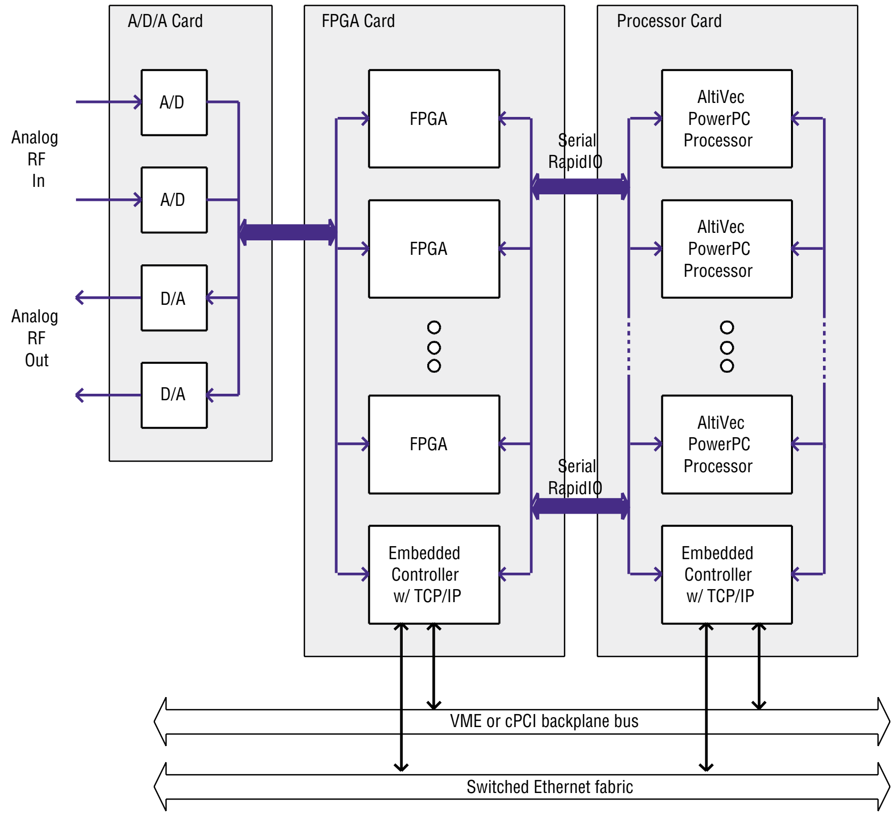

Figure 2

Example SDR Architecture demonstrating technology partitioning between FPGAs, CPUs and high-speed data interconnects.

Field Programmable Gate Arrays (FPGAs) and (2) processors, including DSPs as well as general-purpose processors. Both FPGAs and processors are truly reconfigurable. The FPGA can be reconfigured by reloading its firmware code. Depending upon the size and type of the FPGA, this can take anywhere from a few milliseconds to a few seconds.

On the other hand, a processor may be reconfigured by loading new code into its instruction memory, a process that is usually measured in milliseconds. In both cases the repository of the firmware and executable code is some kind of a general-purpose processor somewhere in the system, containing an operating system with a file system and the ability to access the FPGAs and signal processors at will. Also, this general-purpose processor will usually contain the interface to communicate outside of the signal processing sub-system, for example to a user-interface or monitoring station set up nearby.

Both have their strengths and weaknesses, and as with any technology each should be applied in such a way as to maximize its strengths while minimizing its weaknesses. Table 1 illustrates some of their major strengths and weaknesses.

It's interesting to note that many of the FPGA's strengths are complementary to the processor's weaknesses, and vice-versa. It is clear that a well-designed system combining both of these elements would have the possibility of leveraging both of their strengths.

The symbol rate processing block in Figure 1 in the receive path consists of elements such a digital down-converters, demodulators (for example QPSK demodulators, or rake receivers and correlators for CDMA), various filter types, and so forth. Similarly the symbol rate processing on the transmit path requires different filter sets, modulators such as PSK or CMDA PN (Psuedo-random Numerical) spreading-sequences. Many recent military waveforms occupy quite high bandwidths, requiring these elements to function at frequencies beyond 100 MSPS. Implementing this using DSP processors would require an entire array of them; this is neither space nor cost-effective. These functions are well suited for FPGA implementation.

Bit-rate processing consists of tasks such as bitstream mux'ing and demux'ing, sub-channel extraction and compilation, transport-level protocol handling, error detection and handling, encryption / decryption, plus symbol coding schemes such as constellation mapping, Viterbi or turbo encoding / decoding, and so on. Theoretically, all of these functions can be implemented on DSP processors. However some functions, such as Viterbi or turbo decoding, benefit greatly from hardware acceleration such as can be provided by the FPGAs. Others, such as the bit and error handling and protocol stack implementations, map very well to the processors. This makes it generally worthwhile to allocate the bit-rate processing tasks to the processors, and provide them with some FPGA processing assistance on an as-needed basis.

An example architecture, similar to Spectrum Signal Processing's SDR-3000 platform, is shown in Figure 2. This example shows four A/D converters and four D/A converters interfaced to a bank of FPGAs via a high-speed digital bus: 1.6 Gbytes/s bi-directional (3.2 Gbytes/s aggregate). The FPGAs are interconnected using fast 64-bit data paths. This permits full-rate "digital intermediate frequency" data to be routed to any FPGA, essential for high data rate operations such as Digital Up and Down-Converters. The FPGAs are then able to implement any of the symbol-rate processing tasks described earlier.

For modularity and flexibility, four AltiVec floating-point PowerPC processors can be located on a separate board. Placing them on a separate board makes it simple to vary the ratio of PowerPC processors to FPGAs as different applications require. These are the bit-rate processors, and they're connected to the FPGA board via a backplane serial RapidIO bus providing greater than 200 Mbytes/s bi-directional. This bandwidth not only allows for the pre-processed data from the FPGAs to be passed to the processors, but it permits sufficient headroom for extracted data to be routed back to the FPGA board for additional processing (such as turbo or Reed Solomon decoding) and the results returned to the processor board, if required. A flexible high-bandwidth interconnect scheme, such as serial RapidIO, permits multiple FPGA cards to interface to multiple processor cards, although Figure 2 only shows one of each, the modularity and expansion capabilities are obvious.

Each board contains its own embedded controller running the VxWorks operating system. These controllers are therefore capable of storing and dynamically loading code into the FPGAs or processors directly, as well as communicating with other elements of the system (such as the user terminal) via a standard Ethernet network utilizing the robust TCP/IP stack provided within VxWorks.

The flexibility of such an architecture makes it suitable for numerous applications. In perhaps the simplest application, the system can be used as a true SDR radio set, able to receive and transmit using pre-programmed frequencies and waveforms. Taking this a step further, the system can additionally implement features such as direction-finding and automatic modulation recognition, allowing it to determine the direction of a received signal, as well as determine what type of transmission it is.

If a signal is determined to be hostile the system may optionally transmit on that frequency, jamming it. If a friendly signal is being jammed, beam steering may be used to spatially isolate the friendly signal, greatly improving the signal-of-interest strength. It's the reconfigurability of the FPGA/CPU architecture that makes this possible. All of these options are programs loaded into the FPGAs and processors; they can be dynamically loaded or unloaded in moments.

Finally, the system is not constrained to only communication signals. IFF signals can be broadcast. Radar chirps can be received, and blurs transmitted in response. An SDR system such as shown in Figure 2 becomes limited only by its processing power and system bandwidths, both of which are substantial today and constantly improving.

The concepts and architectures presented herein provide a very high-performance and adaptable platform for Software Defined Radio applications. Replacing many of the traditional analog radio functions with high-performance digital processors makes possible a truly reconfigurable radio. The use of programmable processors and reconfigurable FPGAs bears this out. However, such a radio can only be practically achieved with a well thought-out balance of optimized processing elements and high-speed interconnect bandwidths.

Spectrum Signal Processing
Burnaby, BC, Canada.
(604) 421-5422.
[www.spectrumsignal.com].