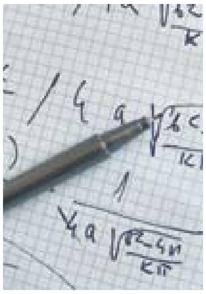

Mapping waveforms to systems: What would a wideband networking waveform system require?

By Kevin Maier

When building a Joint Tactical Radio System (JTRS) modem that must support a wide variety of disparate waveforms, it can be very challenging to assess the overall system requirements. A pragmatic process exists that can be employed to evaluate the requirements of a wideband waveform and to map these requirements to a hardware platform.

The steps recommended in this process include:
1. *Creating a representative waveform with the most demanding attributes*
2. *Performing a functional analysis of the target radio architecture*
3. *Mapping the functional analysis to specific processors*
4. *Determining the data flow, bandwidth, and latency requirements*
5. *Assessing requirements against hardware platforms*

The data presented herein is intended to be instructive in nature and is not exhaustive.

From a functional perspective, most modern tactical military waveforms can be classified into three broad representative waveform classes that all emphasize different parameters, and a rigorous platform evaluation can be done for a representative waveform in each class:

- Narrowband slow hopping waveforms (< 4k hops per second and <1 Mbps data rate)
- Narrowband fast hopping waveforms (> 4k hops per second and <1Mbps data rate)
- Wideband networking waveforms

While wideband networking-type waveforms are typically the most resource-intensive and demanding of all three classes, in reality, the first two classes should also be assessed if the JTRS modem is to support them. However, the following platform evaluation and information is limited to the examination of a representative waveform in the wideband networking class only.

Step 1: Creating a representative waveform

A wideband networking waveform may have a broad variety of parameters. For instance, the Joint Tactical Radio System Wideband Networking Waveform (WNW) has four distinct signals in space to fulfill its operational requirements: Bandwidth Efficient Advanced Modulation, Orthogonal Frequency Division Multiplexing, Anti-jam, and Low Probability of Intercept. For this illustration, we will create a waveform representative of the most demanding parts of the WNW subcategories. Our representative waveform would enable end users to exchange wideband data on the battlefield or, with systems providing reach-back capability such as the WIN-T program, through the Global Information Grid (GIG). Other examples of this waveform type include the Tactical Data Rate waveform supported by the TRC-4000 terminal from Thales, and the AN/GRC-245 HCLOS terminal from Ultra Electronics[1].

Many of the parameters associated with JTRS and other waveforms are not publicly available; however, many of these waveforms are similar to the IEEE 802 series of networking waveforms (802.11, 802.16). As such, we will model the waveform based on publicly available information and supplement that information with details from 802.x waveforms. This hybrid structure will not specifically meet with the requirements of either 802.11 or WNW, but will be sufficient to drive the requirements of our modem architecture and help to evaluate against a hardware platform. The parameters for this representative waveform are presented in Table 1.

Parameter	Value
Channel Location in Spectrum	225-2000 MHz
Channel Spacing	30 MHz
Modulation	OFDM, DQPSK
INFOSEC	N/A
TRANSEC	None
Hops Per Second	None
OFDM Symbol Length	80 Samples
OFDM Guard Interval	16 Samples
OFDM FFT Window Length	64 Samples
Baseband Sample Rate	30 Msps
Error Correction for Payload	Rate ½ Turbo Code
Uncoded data bits per OFDM symbol	48
Signaling (bps)	18 M
Packet Structure	10 Training Sequence Bursts, 1 long OFDM Synchronization Symbol, 1 SIGNAL Header Symbol, 1 to N Data Symbol
Access Structure	Carrier Sense Multiple Access
Acknowledgement Time	10 µs
Duplex	Full

Table 1

Our representative waveform is connection-oriented, meaning that for each packet transmitted, an acknowledgement must be sent indicating whether the packet was successfully received or not. This acknowledgement is based on a test of the Cyclic Redundancy Check (CRC) embedded in the data. The CRC response must be transmitted 10 µs following the receipt of the last OFDM symbol, similar to the acknowledgement time for 802.11g. A hard requirement such as acknowledgement time will help drive the latency requirements of our target system.

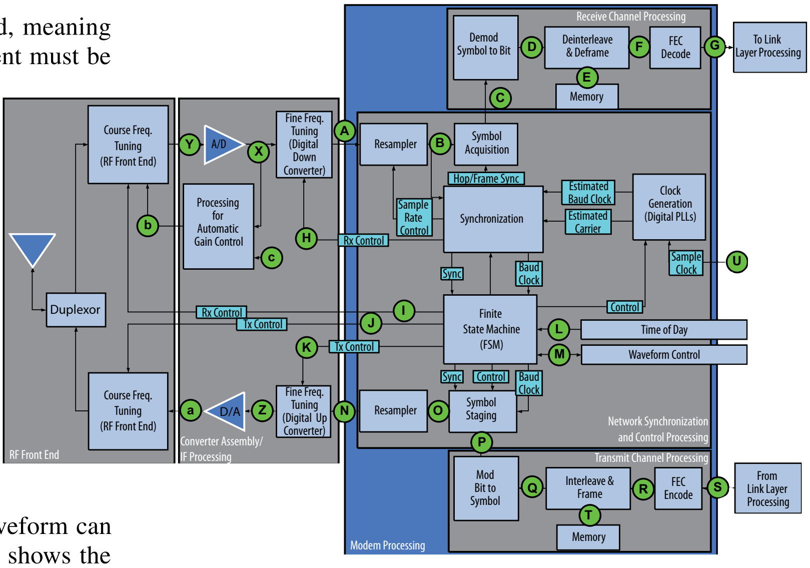

Figure 1

Step 2: Performing a functional analysis

After creating a representative waveform, the next step in the platform evaluation exercise is to construct a functional block description of the target radio architecture. The wideband representative waveform can be deployed in a system similar to Figure 1, which shows the black side (unclassified) of the modem that partitions the modem functionality into six arbitrary blocks:

- RF front end – Provides the antenna interface and first stage of down conversion, accepts and distributes IF data to the next functional block
- IF processing – Performs analog-to-digital conversion and up/down conversion of the digital data
- Network synchronization and control processing – Converts data to/from symbols, contains the Finite State Machine (FSM) that controls the radio
- Receive channel processing – Performs demodulation, deinterleaving and deframing, and FEC decoding

– Provides higher layer data protocol processing and interfaces with the red-side modem processing

- Transmit channel processing – Performs modulation, interleaving and framing of data, as well as FEC encoding

In a supporting analysis, the exact data rate, latency, signaling, and pin count (where appropriate) are determined for particular data flows (indicated by letters) as appropriate to the representative waveform. A partial analysis of the receive path is shown in Table 2.

Path ID	Path Description	Bandwidth, Latency & Determinism Notes
A	**DDC to Resampler**	**The baseband sample rate is 30 Msps, x 2 to allow for synchronization, x 4 bytes per sample (I & Q), which gives a rate of 240 MBps into the resampler**
B	**Resampler to Symbol Acquisition & Synchronization**	**The resampler will take the data rate down by a factor of 2, to 120 MBps**
C	**Symbol Acquisition to Demod**	**Assumption: the training sequence and frequency offset estimation symbols are generally irrelevant in the calculation of these rates.** **30 Mbaud x 48/80 (packet overhead factor) x 4 bytes per sample (2 bytes per complex sample) = 72 MBps**
D	**Demod to Deinterleave and Deframe**	**2 samples per bit and 2 bytes per sample = 36 Mbps**
E	**Deinterleave to Memory**	**2 x data rate = 72 MBps assuming 8-bit addressable regions. Could be up to 4x this rate if we only have 32 bit addressable regions**
F	**Deinterleave to FEC**	**36 Mbps**
G	**FEC to Link Layer Processing**	**Rate ½ encoding gives 18 Mbps**
H	**DDC RX Control from FSM**	**The fine control requires adjustment inside of the guard band of a symbol – or 20% of a symbol length = 0.533 µSec**

Table 2

Step 3: Mapping to specific processors

After our representative waveform has been created and mapped to a functional block diagram of the modem, the results need to be allocated to actual physical processing elements. Notice at this point a target platform has still not been identified. The functional blocks should be assigned to specific processor types to estimate resource utilization that will help determine part sizing, leading to appropriate hardware platform architectures.

In general, high-speed or computationally intensive algorithms are mapped to FPGAs, with other algorithms mapped to either a DSP or a General Purpose Processor (GPP), depending on the anticipated power utilization and requirements for code portability. Higher layer back-end processing (link layer, network layer) is typically performed exclusively on a GPP.

In the case of this representative waveform, the functional processing blocks described above are mapped into all three different categories: DSPs, FPGAs, and GPPs. The functional blocks are partitioned as shown in Figure 2.

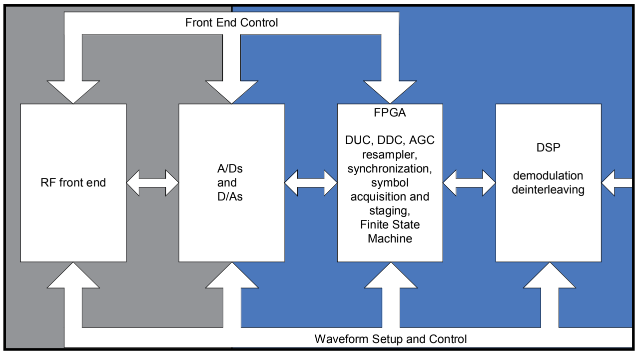

Figure 2

Once the waveform algorithms are partitioned across different processors, it is time to start our resource utilization assessment. Utilization estimates vary, but significant information is available in the public domain (for example, the Xilinx website provides FPGA utilization estimates). Much information about MIPS processing requirements of waveform components is also publicly available from the US government.

A detailed analysis of each functional processing element must be performed to create an accurate resource utilization estimate. For instance, our detailed receive side resampler analysis might be described as follows:

Rx resampler: Resampling is required to ensure that 4 samples per baud are input to the symbol acquisition and synchronization blocks. The resampling architecture follows that of the GC3011A resampler. The input rate to this resampler will nominally be 60 Msps, and the output rate will be 30 Msps. It is anticipated that the interpolation filter will be implemented using a Farrow filter, following the method prescribed by Harris and Dick[2]. This implements a 256-tap interpolation filter as five 8-tap filter stages with four MAC stages, thus the equivalent performance for the GC3011 is estimated to require sixteen 15-tap filter stages and 15 MAC stages. The 15-tap FIR filters can be implemented using a Distributed Arithmetic (DA) approach with a total estimated resource utilization of less than 16 x 125 x 2.25 = 4500 slices[3].

Total estimated resource utilization would include 15 multiply and add blocks, and 1 block RAM in a Xilinx Virtex-II or Virtex-4 device. (Note that in Virtex-II, the adder would have to be implemented in gates). Also note that since the original filter architecture was based on XC4000 series FPGA technology, additional savings could be achieved using Virtex-II or Virtex-4 devices. As such, adding an additional 20 percent logic for overhead glue logic and allowing for floor planning risk puts total FPGA resource utilization at less than 5400 logic cells, 15 multiply and accumulate blocks, and 1 block RAM. Other detailed analysis leads to the summary presented in Table 3.

The detailed breakdown in Table 3 leads us to target a platform that has a single 600 MHz TMS320C6416 DSP, a GPP capable of at least 1000 MIPS, and an FPGA with approximately 32k logic cells and 3 x 18k block RAMs. In a size-, weight-, and power-constrained environment such as tactical MILCOM, the target platforms should encompass devices that do not have much excess capacity beyond a planned risk margin. For instance, one would not employ a Motorola AltiVec engine to field this waveform because of the high power consumption and general overcapacity of that particular processor.

Component Name	Proposed Processing Technology	Estimated Resource Utilization
Rx Resampler	FPGA	5400 Logic Cells, 15 MAC Blocks, 1 Block RAM
OFDM Symbol Synchronization	FPGA	3197 Logic Cells
OFDM Symbol Acquisition	FPGA	1590 Logic Cells, 2 MAC Blocks, 1 Block RAM
OFDM Symbol to Bit	FPGA	10744 Logic Cells
FSM	FPGA	5580 Logic Cells
Deinterleave/ Decode/ Interleave/ Encode	TMS320C6416	<100%
Link Network Layer Processing	GPP	< 1000 MIPS
Symbol Staging	FPGA	Minimal
Tx Resampler	FPGA	5400 Logic Cells, 15 MAC Blocks, 1 Block RAM

Table 3

There exist several modern processors that can fulfill the GPP requirements for target MIPS and target power consumption; for example, 1000 MIPS would require ~80 percent of a MPC 8541 at 533 MHz and run at approximately four watts. To fulfill the FPGA requirements, we could look at the two most recent Xilinx FPGA architectures; our waveform would require approximately 70 percent of a Xilinx Virtex-II Pro V2P40 or 53 percent of a Virtex-4 LX60. The Virtex-4 runs at approximately 50 percent of the power of a Virtex-II Pro and would be a suitable choice for our modem platform.

Step 4: Determining data flow, bandwidth, and latency

Once the functional blocks have been allocated to individual devices, the data flow requirements between our target devices can be summarized as shown in Table 4. This analysis will help determine if an interprocessor link requires channelized data flow and also what amount of bandwidth and latency is required for the link. The bracketed letters refer to the data flows identified in Figures 1 and 2.

Step 5: Assessing requirements against hardware platforms

Performing all of the above stages of analysis will generate a set of hard requirements for a target platform. One can then choose to build the hardware from scratch or utilize existing Commercial Off-The-Shelf (COTS) boards and modules. The latter option is usually desirable if there are schedule or resource limitations. A first and easy test to see if specific COTS equipment is suitable is to use a checklist approach. Gather product information (typically available on a datasheet) and compare how well the product meets the overall system and data flow requirements. (Note that COTS designers marketing tactical MILCOM targeted systems should perform the same analysis to ensure that their systems can handle the target waveform requirements.)

Path	Channels	Aggregate Bandwidth	Aggregate Latency
FPGA to RF	Data: None	N/A	N/A
	Control: (I,J)	Low (under 100 kB/s)	All paths must be deterministic and bounded (~1-10 millisecond is appropriate).
RF to FPGA	None	N/A	N/A
FPGA to IF	Data: (A)	240 MBps	.5 µs
	Control: None	N/A	N/A
IF to FPGA	Data: (N)	240 MBps	.5 µs
	Control: (H K), U	Low	All paths must be deterministic. Channels H and K must be fixed latency
FPGA to DSP	(D)	36 Mbps	Low
DSP to FPGA	(Q)	36 Mbps	Low
FPGA to GPP	(M)	Low	Low
GPP to FPGA	(M)	Low	Low
DSP to GPP	Data: (G)	18 Mbps	Low
	Control: none	N/A	N/A
GPP to DSP	Data: (S)	18 Mbps	Low
	Control: none	N/A	N/A

Table 4

The final step in assessing any platform is to actually build a representative waveform and port it to the target hardware. An example is the recent implementation of the 802.11g waveform on Spectrum's off-the-shelf SDR-3000 platform. This waveform requires the same bandwidth and latency performance as military wideband waveforms and as such demonstrates the SDR-3000's suitability for wideband waveform implementations.

The process described herein was employed during the development of Spectrum Signal Processing's newest series of Software Defined Radio (SDR) products. These SDR products have been designed from the ground up to support tactical MILCOM applications; consequently, it was critical to validate that the modem architecture could support all of the JTRS waveforms in all of the key parameters: processor types and capacity, interprocessor bandwidth and latency, and external connectivity.

References:

1. Pucker, L. "Paving Paths to Software Defined Radio," *Communication Systems Design Magazine*, June 2001.
2. Dick, Chris and Harris, Fred, "FPGA Interpolators Using Polynomial Filters," The 8th International Conference on Signal Processing Applications and Technology, Toronto, Canada, Sept. 13-16, 1998.
3. Xilinx Logicore Distributed Arithmetic FIR Filter, v9.0, Product Datasheet DS240 May 21, 2004.

***Kevin Maier** has been a systems engineer at Spectrum Signal Processing Inc. for eight years, working on hardware, software, and embedded systems. He earned a BAS (honors) degree in systems engineering from Simon Fraser University.*

To learn more, contact:
Spectrum Signal Processing Inc.
#300-2700 Production Way
Burnaby, BC V5A 4X1
Canada
E-mail: sales@spectrumsignal.com
Website: www.spectrumsignal.com