

Standardizing Transceiver APIs for Software Defined and Cognitive Radio

This article introduces the key features of the transceiver API facility, underlining the main benefits of the approach, and presents the SDR Forum Transceiver Sub-system Interfaces Task Force and the associated standardization stakeholders.

By Eric Nicollet and Lee Pucker

Over the past several years, numerous specifications have been developed defining the interface between the RF front-end and the baseband processing sub-systems in advanced wireless products. These interfaces include the Reference Point 3 specification developed under the open base station architecture initiative (OBSAI), the DigRF interface specification under development by the mobile industry processor interface (MIPI) alliance, the VITA radio transport draft standard developed by the VITA standards organization as VITA 49, and the common public radio interface (CPRI). The reason these specifications were developed is simple: standardization of these interfaces allows an ecosystem of vendors to form interoperable RF and baseband processing modules. As more and more vendors enter the market with these standards-based modular products, the modules will become commoditized, allowing wireless original equipment manufacturers (OEMs) to choose equivalent modules from a range of competing vendors, reducing their overall production and production costs.

The ongoing proliferation of software-defined radio (SDR) technologies in advanced wireless systems, allowing some or all of

the radio's physical layer (PHY) processing to be defined in software and HDL code, has created a need for a further level of standardization in the interface between these two subsystems—standardization of an application-programming interface (API) that would provide a layer of hardware abstraction from the underlying RF front-end and baseband processing connections, including support for both signal transport and control. Toward that end, the SDR Forum has been working with other organizations to define a model for a transceiver API that, when standardized, will provide significant benefit to the wireless community by facilitating the insertion of new or updated physical-layer waveform applications and air interface standards while the radio is in operation and, at the same time, improving the portability of these applications and standards from radio to radio. This article introduces the work of the SDR Forum Transceiver Subsystem Interface Task Force in defining this “Transceiver Facility,” discussing a process they are following in developing the API in detail, and discussing the next steps toward final standardization.

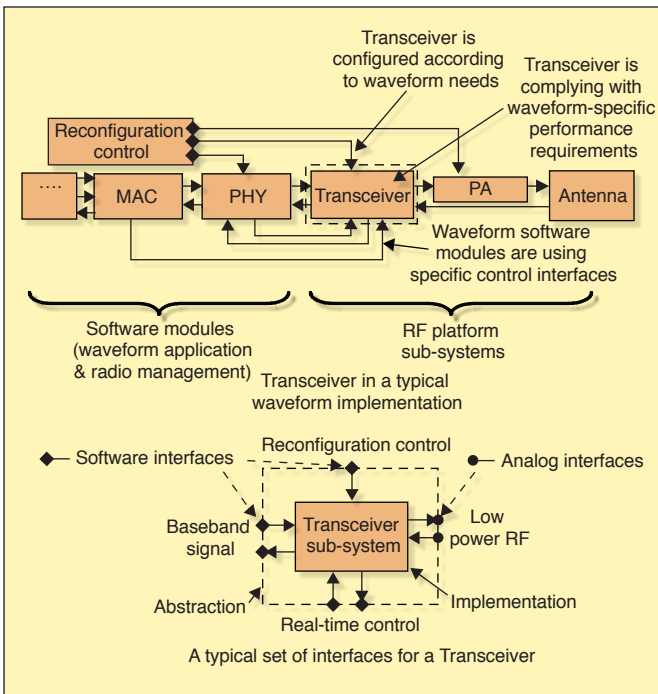


Figure 1. Notional view of the Transceiver sub-system within a radio set's layered architecture.

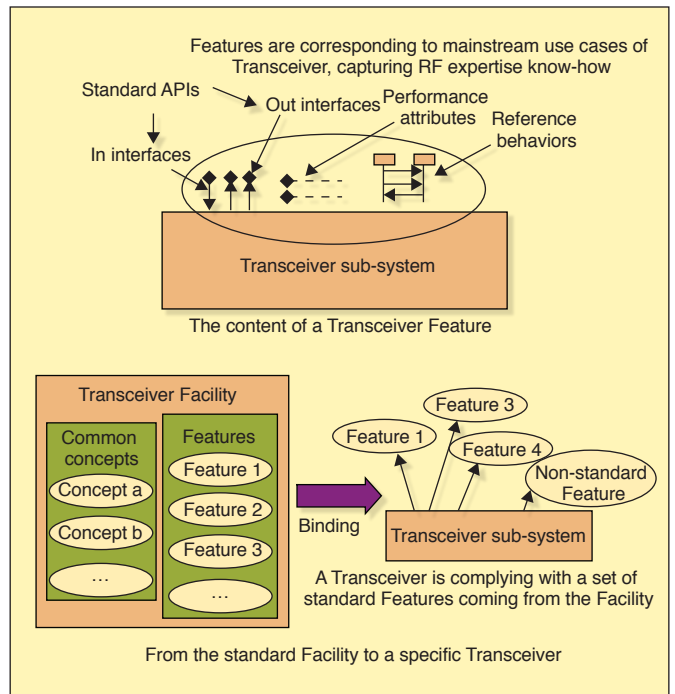


Figure 2. A Transceiver is complying with a set of standard features coming from the Facility.

Background

The term “transceiver” is used here to encapsulate the entire set of hardware and software components within a radio set necessary to convert a low-power RF signal to digital baseband on the receive side, and reciprocally to convert digital baseband signals to low-power RF on the transmit side. Any management and control features necessary to support the transceiver usage are also considered as part of the transceiver sub-system (Figure 1). The transceiver subsystem provides intermediary processing between the radio’s baseband or modem-processing subsystem, and the other subsystems contained with the “RF front-end” such as the power amplifier, the co-site mitigation, and the antennas.

While numerous specifications and standards have been developed over the past several years defining the lower level interfaces between transceiver subsystem and the rest of the radio set, there has been a recognized need within the reconfigurable radio community to pursue the definition of an API to support this interface at the system level.

The reason for this interface is that this software-level frontier should match the industrial frontier already identified as one key reference point between baseband and transceiver stages. Setting the frontier “closer to the RF” presents a number of drawbacks, including introducing dependencies on implementation-dependant choices in the radio set design, which lowers the portability of the developed software. Setting the frontier at some higher level also presents drawbacks in that it prevents significant parts of the waveform application software from taking advantage of the platform abstraction.

The process

The initial work on the transceiver API was done within the scope of the French Department of Defense (DoD) “Software Radio Study,” (ended 2006) where a first level of definition was issued in order to define a common transceiver abstraction. This study used two transceiver platforms that were manufactured following two entirely different hardware designs by THALES and Raytheon. However both platforms supported the same transceiver API, and this level of transceiver abstraction was flexible enough to support the needs of two different waveforms: FM3TR and STANAG 4285. When implemented on the two platforms, these waveforms were demonstrated to be interoperable, across themselves and with legacy radios.

The work on the transceiver API was then progressed within the European End-to-End Reconfigurability Phase II (E²R II) Project.^[1,2] Under the auspices of the European Union “6th” Framework Program, this program had as a goal to define standardized system architectures that “provide common platforms and associated execution environments for multiple air interfaces, protocols and applications.” The E²R contribution has been to leverage the previous program results in order to meet the specific industrial requirements of the transceiver abstraction endeavour, in satisfying the needs of multiple waveforms and highly different possible platforms. The key breakthrough to square of the circle has been the introduction of the transceiver facility concept, which yielded to the creation of the task force at SDR Forum level.

This work is now being refined in the SDR Forum under the Transceiver Subsystem Interface Task Force. The objective of this group is to mature the specification to ensure that it supports the needs of a broad base of radio architectures in the commercial, civil and defense communications domains. A key objective of the group is to evalu-

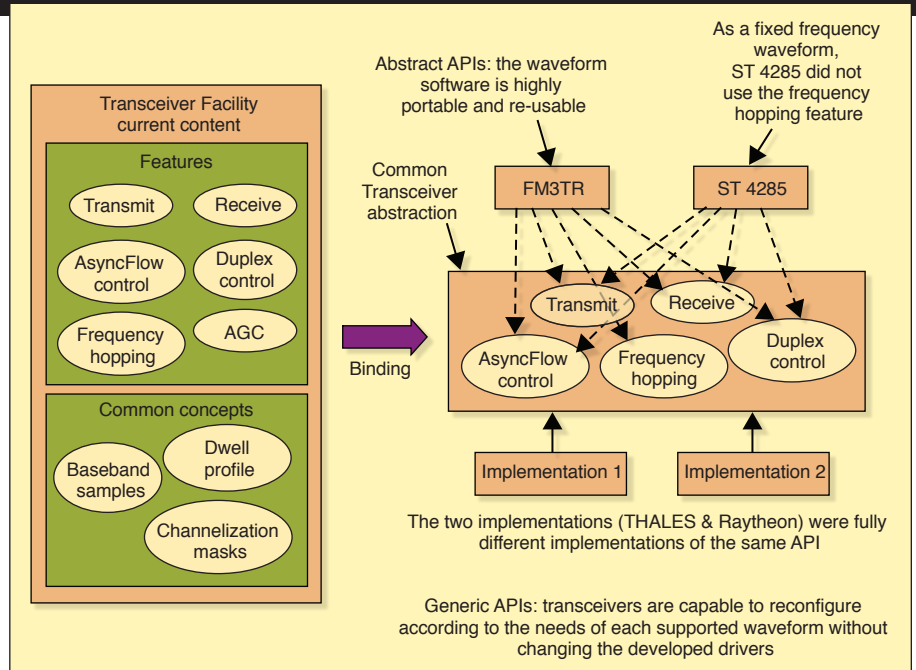


Figure 3. The current content of the Facility and the demonstrated features.

ate the proposed transceiver facility against the various lower-level transceiver interface specifications and standards that have been developed for the commercial and defense wireless markets over the past several years to ensure that they can be supported by the resulting API. This evaluation is expected to include the Reference Point 3 specification developed under the OBSAI, the DigRF Interface specification under development by the MIPI Alliance, the VITA Radio Transport draft standard developed by the VITA Standards Organization as VITA 49, and the CPRI. Several additional developments are also envisioned, including providing extensions to better support spectrum-aware cognitive radio or smart radio systems.

The organization

The transceiver facility defines reference content for APIs to access a specific transceiver implementation from the baseband software.^[3] The facility is composed of “features,” with each feature corresponding to an elementary prime capability that may be implemented in a transceiver sub-system (Figure 2). The use of features in this manner allows the transceiver facility to easily scale to the needs

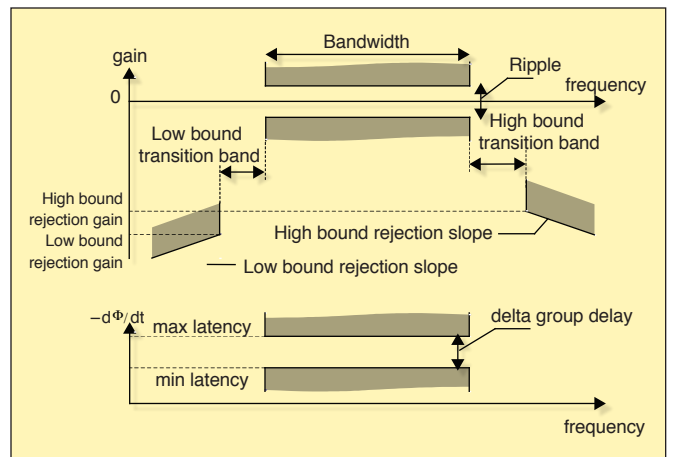


Figure 4. Common concept “ChannelMask” provides a simple and sufficient view of the expected signal channelization, using a transfer function mask.

of a specific transceiver implementation. Each proposed feature includes an extensive set of APIs and performance attributes that enable the radio developer to manage the various transceiver interfaces and characterize their performance. For instance, the feature “transmit” proposes a reference function to send the packets of baseband samples toward the transceiver, called *pushBBSamplesTx*, with associated performance attributes such as the expected sample rate and size of the samples packet.

The features are based on a certain number of “common concepts,” which are captured independently from the features themselves. Common Concepts have no content directly usable as API definition, but capture key radio-domain concepts on top of which the features are built. For instance, the notion of baseband signal power level is defined as a common concept that is used by the “receive” feature to characterize the nominal output level and the “AGC” feature to characterize signal level ranges for the AGC.

The transceiver development activity is built upon solid methodological foundations using the model-driven architecture (MDA) approach.^[4] The essential objective of this approach is to enable the portability and/or reusability of architectural models across different platforms. In order to achieve this, the transceiver facility must first be defined independent of any technology or implementation, with the resulting model referred to as a platform-independent model or PIM. Once the PIM is complete, the various functionalities defined by the model can then be mapped to a specific platform, creating what is referred to as a platform-specific model (PSM).

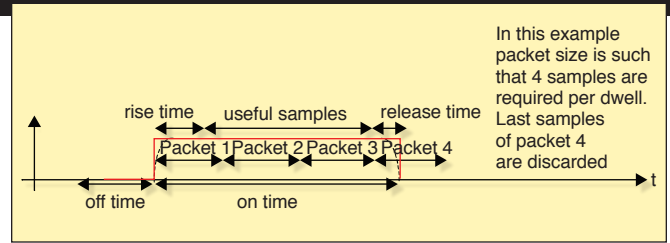


Figure 5. Common concept “Dwell Profile” provides for characterization of frequency-hopping, real-time control mechanisms.

Usage of the facility

The transceiver facility can be used from either a waveform characterization viewpoint or a transceiver characterization viewpoint. From a waveform characterization viewpoint, each feature of the transceiver Facility provides a certain number of functions and performance attributes that are given specific waveform-dependent values. The subset of features necessary to support a given waveform is identified, and the values of the associated attributes are defined. For instance, for FM3TR one can state that for the transmit feature, the sampling rate shall be equal to 100 kilosamples per second (ksps), which is a value specific to the FM3TR waveform definition. The organization of the facility into features avoids the need to mandate unnecessary APIs for a given waveform, so the transceiver facility does not lead to overspecification.

From a SDR platform characterization viewpoint, the transceiver supports a number of waveform configuration states, which can be defined by referencing the waveform-specific characterizations that used the facility as exposed beforehand. Additionally, some features that would not be specific to a certain waveform can be captured

thanks to direct usage of the facility in setting boundaries on the implementation, saying for instance that the carrier frequency may range from 2 MHz to 2 GHz independent from any other setting.

A flexible SDR transceiver is capable of accommodating the needs of different waveforms and switching between the configuration states needed for each supported waveform. A dedicated feature must harness the associated reconfiguration needs, with the corresponding control functions proposed to the reconfiguration software. It is then sending the suitable reconfiguration commands to the transceiver through this management API, which can be product-specific or standard. The facility proposes the standard "Device" interface (from SCA and OMG) as one possible solution.

Current content of the transceiver facility

The current contents of the transceiver facility are illustrated in Figure 3. This content presents capabilities that were sufficient at the time of the initial program, and acts as a starting point for follow-on development of the final transceiver facility. Features and common concepts included in the initial facility are as follows:

- Feature "transmit" expresses the transmission capabilities of the transceiver. It specifies the interface PacketTransmit in case of packet transmission of the baseband samples, and mandates usage of a transmission channelization mask.

- Feature "receive" expresses the reception capabilities of the transceiver. It specifies the interface PacketReceive in case of packet reception of the baseband samples, and mandates usage of a reception channelization mask.

- Feature "HalfDuplexControl" enables control of the duplex state of the transceiver. The interface DuplexControl and associated state machine are specified in this feature.

- Feature "FrequencyHoppingControl" enables control of the frequency hopping of the transceiver. The interfaces DwellSetting, DwellTuning and FrequencyFeeding are defined in this feature.

- AsynchronousTxFlowControl. This feature enables an asynchronous exchange mode in transmission between a transmitting physical layer resource and the transceiver. It defines the interface flowSignaling that enables the high/low watermark information to be given by the transceiver to the transmitting software.

- RxAdaptiveGainControl. This feature enables specification of the expected adaptive gain control behavior of the transceiver. It defines the associated classes and optional real-time control interfaces.

- TxAdaptiveLevelControl feature enables to specify the expected adaptive level control behavior of the transceiver. It defines the associated classes and necessary real-time control interfaces.

- Common concept "BaseBandSamples" provides for characterization of the baseband sample flow carried at the data path interface level. The concepts BBSamplesStream (stream general features), BBSamples (a buffer of consecutive samples) and BBSample (the elementary sample) are defined in this concept.

- Common concept "ChannelMask" provides a simple and sufficient view of the expected signal channelization, using a transfer function mask (Figure 4). The structures ChannelMask, GroupDelayMask and SpectrumMask are defined in this concept.

- Common concept "Dwell Profile" provides for characterization of frequency-hopping real-time control mechanisms (Figure 5). The classes DwellProfile, FrequencySet and LevelSet are defined in this package.

Next steps

The SDR Forum believes that the transceiver API, as the very place where software defines the radio, is of critical importance in promoting the success of next-generation radio technologies. As such, the SDR Forum Transceiver Subsystem Interface Task

Force plans to aggressively work to complete the evolution of the transceiver facility and promote it for SDR Forum balloting during the 2008 calendar year. Once the balloting is completed, the transceiver API will be an approved SDR Forum specification, and available for use by the larger community. In addition, the SDR Forum plans to submit the approved transceiver facility to the OMG for follow-on standardization through its process. This will be done as a revised submission in response to the Digital IF Request for Proposal issues by the OMG's software-based communications domain task force (SBC-DTF).^{1,5,6} The SDR Forum, in support of its members has been closely collaborating with the OMG through a joint process that is evolving the SDR Forum Smart Antenna API in a similar manner. The Transceiver Subsystem Interface Task Force will also explore the qualification or certification of transceiver facility compliant implementations, and will discuss the addition of transceiver facility toolboxes with various tools vendors to facilitate implementation.

References

1. "End-to-End Reconfigurability Phase II Integrated Project," <http://www.e2r2.motlabs.com/>.
2. E. Nicollet, "E²R commercial Reconfigurable Radio Equipments: from advanced research to proof-of-concept and Standardization," SDR Forum Technical Conference, Orlando, November 2006.
3. E. Nicollet, "The Transceiver Facility Platform Independent Model (PIM): Definition, Content And Usage," SDR Forum Technical Conference, Orlando, November 2006.
4. J. Miller and J. Mukerji (Editors), "MDA Guide Version 1.0," June 2003, <http://www.omg.org/docs/omg/03-06-01.pdf>.
5. OMG Software-Based Communications DTF, "Request for Proposal PIM and PSM for Digital Intermediate Frequency Interface," OMG Document, sbc/2004-08-15, 2004.
6. T. Demirbilek and E. Nicollet, "Joint Digital IF Initial Submission," OMG Document, sbc/05-03-01, March 21, 2005.

ABOUT THE AUTHORS

Eric Nicollet is the embedded SDR architecture specialist with THALES Communications, France. He is practicing SDR-related topics for DSPs since early 1998. He is currently involved in several European initiatives and programs on SDR, in all defense (SCORED/ESSOR), safety-of-life (WINTSEC) and commercial (E²R/E3) domains. He holds an MSc from the National Superior School of Electronics and Applications, Paris.

Lee Pucker is the consulting chief executive officer of the SDR Forum. Pucker has more than 20 years of experience in wireless communications and signal-processing markets and holds a BSEE from the University of Illinois, and an MSc from The Johns Hopkins University.