



Accelerating the Re-Configuration of Digital Front-End Solutions for Cellular Base Stations

Using a library of pre-verified IP blocks, you can implement various TD-SCDMA digital front-end solutions.

by Tim James
Principal Engineer
Multiple Access Communications Limited
tim.james@macltd.com

David Kenyon
Managing Director
Multiple Access Communications Limited
david.kenyon@macltd.com

Sam Chalmers
Principal Engineer
Multiple Access Communications Limited
sam.chalmers@macltd.com

The advent of software-defined radio (SDR) enables wireless base station (BS) equipment designers to support a wide variety of configurations and radio frequency (RF) standards using generic hardware platforms. The realization of such systems has many potential benefits: development costs spread across multiple product lines; new and customized product variations brought to market more rapidly; and existing products upgraded in response to evolving RF standards.

Using the latest FPGA, DSP, and converter technology, hardware designers can develop flexible platforms capable of supporting a wide range of product configurations. At first this appears to greatly reduce the design task, but closer inspection reveals only a shift in the balance between hardware and firmware; the reduction in



hardware design has been replaced by a greater need for firmware. Every product variant will require a different firmware build, with each incurring significant development effort.

Base stations for cellular radio networks come in a variety of shapes and sizes. At one end of the scale are single-carrier, single-antenna devices for use indoors or in areas of low subscriber density. At the other end

of the scale are BSs that deploy perhaps six carriers per sector and use adaptive antennas, with as many as eight elements, to provide enhanced performance. What they all have in common is the requirement to transpose signals between the RF and base-band domains. This transposition is the role of a digital front-end (DFE).

Consequently, there is a need for software tools that allow for the accelerated

development of new firmware designs. You should be able to concentrate on the high-level system architecture and not have to worry about the low-level design and implementation of the core signal path building blocks. Obviously, you cannot overlook the detailed design of the filters and mixers. However, it should be possible to complete that task once and encapsulate the results so that you can construct other

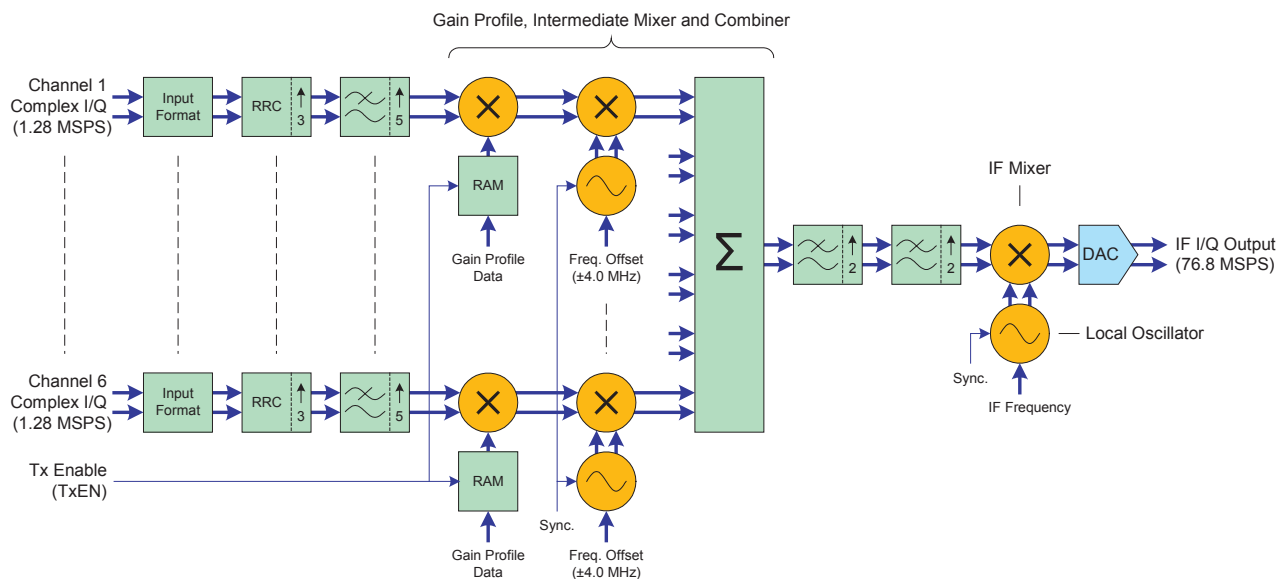


Figure 1 – Six-channel DUC signal path

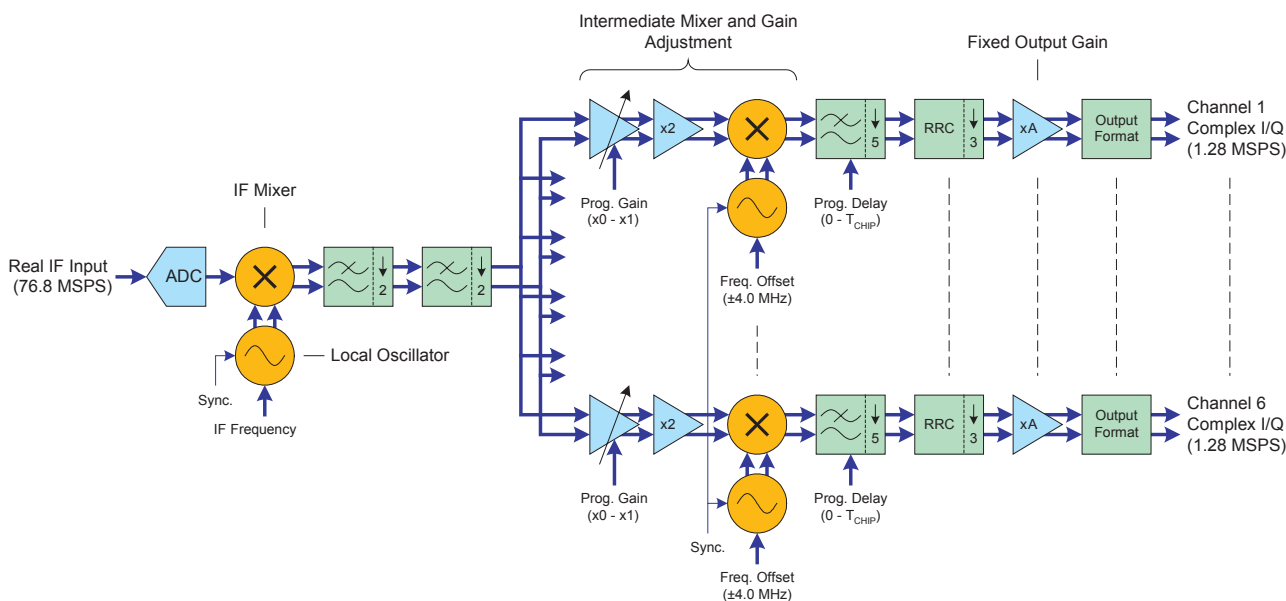


Figure 2 – Six-channel DDC signal path

designs at the system level, free from redesigning detailed signal processing functions for every product variant.

In this article, we'll show how you can use a library of pre-verified intellectual property (IP) building blocks developed for use with Xilinx® System Generator for DSP to rapidly assemble a range of DFEs for different BS configurations with confidence that all conformance criteria will be met. The realization of this goal has the potential to deliver huge savings in design time when measured over the lifetime of a cellular BS product.

Coping with Complexity

The function of the BS DFE can be split in two halves. Up-conversion (that is, the downlink transmit path), is handled by the digital up-converter (DUC). The function of the DUC is shown in detail in Figure 1. Down-conversion (the uplink receive path) is handled by the digital down-converter (DDC), which is shown in Figure 2. These figures show the design of a DFE developed for use with the time division-synchronous code division multiple access (TD-SCDMA) system, part of the 3G cellular radio standard.

In addition to performing the frequency translation and pulse-shaping operations, the DFE shown supports the implementation of burst gain profiling on the downlink and programmable channel gain and signal path delay on the uplink. Although this article focuses on the TD-SCDMA DFE, DFEs for other radio systems would have a similar function and structure.

At the system level, there is an obvious, regular structure to the design. It is at this level that a BS designer would ideally wish to work, for at the lower levels complexity increases significantly. Moreover, as readers familiar with designing DSP systems will be aware, each of the blocks shown in Figures 1 and 2 can be implemented in a variety of ways. To make optimum use of the FPGA resources available, the low-level blocks will typically employ resource sharing; using the highest possible clock frequency will make the most efficient use of key FPGA resources such as DSP48s and block RAMs. (The TD-SCDMA DFE library

discussed in this article is optimized for use with a 307.2 MHz system clock.)

Such techniques require careful consideration to ensure that the blocks will meet the required timing constraints when the FPGA is built. Furthermore, there is considerable complexity in the scheduling of the data paths to ensure that the desired signal processing functions are implemented correctly. It is this complexity that we seek to hide once the low-level blocks have been designed and their operation verified.

For any given DFE configuration, it is easy to see how the design details can be hidden by the design hierarchy; System Generator inherently provides the mechanisms to do this. But what we require is a way for BS designers to re-configure the DFE without having to delve into the details of working with the low-level blocks shown in Figures 1 and 2. One approach might be for the library designer to pre-

configure top-level blocks corresponding to each of the expected configurations. But this simply transfers the complexity from the BS designer to the library designer and does not provide true flexibility, as the supported configurations will be fixed when the library is designed.

The solution that we have adopted in the design of the TD-SCDMA library is to build top-level blocks that each support one antenna and as many as six carriers. When less than six carriers are required, the unused channels can be simply terminated; internal mechanisms will ensure that the unused logic is removed during the build process. This approach delivers ease of use while ensuring that the final design makes efficient use of FPGA resources.

Creating Designs Using the DFE Library

The use of the TD-SCDMA DFE library is simplified by the fact that most DFE func-

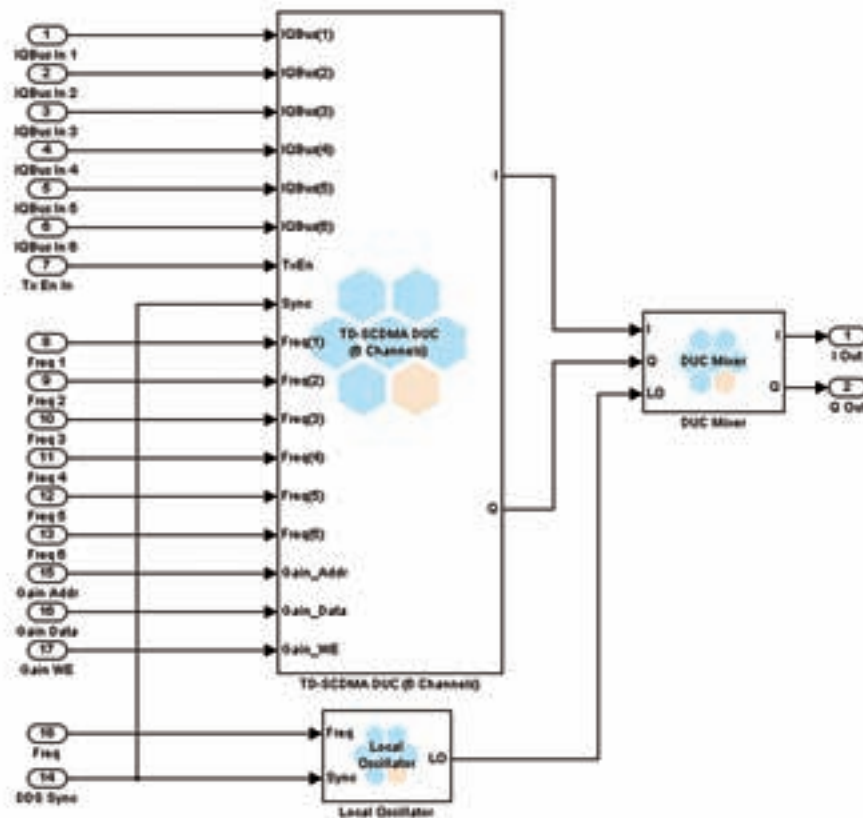


Figure 3 – Example six-channel DUC for a single antenna

The DUC and DDC solutions implemented by the TD-SCDMA DFE library have been designed to comfortably exceed all of the relevant requirements of the TD-SCDMA radio technology.

tionality is implemented in just two blocks (a six-channel DUC and a six-channel DDC), with additional blocks to allow different intermediate frequency (IF) mixing options. This simplifies the task of adding data interfaces to the design.

A six-channel DUC constructed using the DFE library blocks is shown in Figure 3. Most of the signal processing is performed within the six-channel “TD-SCDMA DUC (6 Channels)” block. The “Local Oscillator” and “DUC Mixer” library blocks are added to allow the composite output of the DUC to be translated from zero to a more practical IF. This subsystem generates the output for a single antenna. Multi-element antenna systems can be supported simply by replicating this once for each element.

Supporting an arbitrary number of antennas is relatively straightforward. But what about designs that require fewer than six carriers? To use a full six-carrier design in such circumstances, although a perfectly valid solution, would lead to an unnecessary large FPGA design. In the worst case, this might prevent the use of a smaller device and greatly increase the cost of the solution.

Solutions that require you to manually remove any unnecessary logic or that involve the implementation and supply of a complete set of pre-defined variants clearly defeat the purpose of the library, which is to distance the user from needing to understand the design’s intricacies. Instead, as mentioned

in the preceding section, the DFE library has additional features that help the downstream design tools optimize away unused logic at build time. Thus, you can simply terminate unused inputs using a library block provided for that purpose and tie unused control inputs to constant values. Now, although this design is constructed using the six-channel DUC subsystem, all of the logic, block RAM, and DSP48s dedicated to the unused channels will be removed at build time. With the DDC, unused channels are optimized away simply by terminating the unused outputs (using standard “Terminator” Simulink blocks) and tying unused control ports to constant values. As with the DUC, you can support multiple antennas simply by duplicating the single-antenna design.

Performance

It is important that the flexibility desired is not achieved at the expense of reduced RF performance. The DUC and DDC solutions implemented by the TD-SCDMA DFE library have been designed to comfortably exceed all of the relevant requirements of the TD-SCDMA radio technology. In most cases the margin is many tens of dBs, to allow as much headroom as possible for degradation by the analog RF stages of the signal path.

The logic resources utilized by the six-channel TD-SCDMA DUC and DDC subsystems are given in Table 1, which includes estimates for both six- and three-


channel configurations. We find that the resource requirements do not scale linearly with the number of active channels, since some parts of the signal path are common regardless of the number of channels used (as shown in Figures 1 and 2).

Conclusion

The flexibility of FPGAs solves the hardware designer’s dilemma of how to provide a platform to cater for a range of product variants, but in doing so transfers the problem to the FPGA firmware designer. In a cellular base station, the product variants share many common elements; with careful design this can be exploited to simplify the firmware designer’s task.

Using a TD-SCDMA DFE library developed for use with System Generator, we have shown how a library of pre-verified IP may be developed in a way that clearly separates the system-level design from that of the detailed DSP tasks. This combination of system-level design tools and IP can greatly reduce design time, hiding the detailed complexity of low-level implementation and allowing you to concentrate on the product architecture.

Furthermore, providing a clear separation between system-level and DSP-level designs facilitates the migration of designs to new FPGA devices, engendering similar advantages that object-oriented programming brings to software design.

The DFE library for TD-SCDMA was developed to target Virtex™-4 FPGAs on behalf of Xilinx by Multiple Access Communications Limited, a consultancy company based in Southampton, UK (www.macltd.com). The library is provided with example designs for hardware co-simulation and a working full-speed demonstration design targeted at the Virtex-4 variant of the Xilinx XtremeDSP™ development platform. 

Library Module	Configuration	Slices	Block RAM	DSP48s
Six-channel DUC	All Channels Used	905	17	19
Six-channel DUC	Three Channels Used	623	11	13
Six-channel DDC	All Channels Used	802	15	20
Six-channel DDC	Three Channels Used	479	9	12

Table 1 – Resource estimates for six-channel DUC and DDC TD-SCDMA DFE library blocks when built with six channels and three channels, respectively.