

In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, pages 1510–1513, Dallas, Texas, March 2010.

FPGA-BASED DESIGN AND IMPLEMENTATION OF THE 3GPP-LTE PHYSICAL LAYER USING PARAMETERIZED SYNCHRONOUS DATAFLOW TECHNIQUES

Hojin Kee and Shuvra S. Bhattacharyya

Dept. of Electrical and Computer Engineering,
University of Maryland,
College Park, MD 20742, USA
{hjkee, ssb}@umd.edu

Ian Wong and Yong Rao

National Instruments Corporation
Austin, TX 78759, USA
{Ian.Wong, Yong.Rao}@ni.com

ABSTRACT

Synchronous dataflow (SDF) is an ubiquitous dataflow model of computation that has been studied extensively for efficient simulation and software synthesis of DSP applications. In recent years, parameterized SDF (PSDF) has evolved as a useful framework for modeling SDF graphs in which arbitrary parameters can be changed dynamically. However, the potential to enable efficient hardware synthesis has been treated relatively sparsely in the literature for SDF and even more so for the newer, more general PSDF model. This paper investigates efficient FPGA-based design and implementation of the physical layer for 3GPP-Long Term Evolution (LTE), a next generation cellular standard. To capture the SDF behavior of the functional core of LTE along with higher level dynamics in the standard, we use a novel PSDF-based FPGA architecture framework. We implement our PSDF-based, LTE design framework using National Instrument's LabVIEW FPGA, a recently-introduced commercial platform for reconfigurable hardware implementation. We show that our framework can effectively model the dynamics of the LTE protocol, while also providing a synthesis framework for efficient FPGA implementation.

Index Terms— FPGA implementation, Dataflow modeling, 4G Communication systems, LTE.

1. INTRODUCTION

The ever increasing demand for richer applications and multimedia content in mobile devices has fueled the continuous evolution of wireless standards towards bringing higher data rates and lower latencies to the end user. The third-generation partnership project (3GPP) has responded to this by recently finalizing the latest cellular standard called *long-term evolution (LTE)* [1]. LTE promises data rates of up to 300 Mbps in the downlink, 150 Mbps in the uplink, spectrum flexibility from 1.4 to 20 MHz, and mobility support from stationary users all the way to high-speed train speeds with a graceful degradation of service. In order to meet these demanding re-

quirements, both base station and user equipment also require much higher complexity than ever before. In order to meet the ever tightening time-to-market requirements and resource constraints, the ability to quickly design, simulate, and prototype complex communication systems such as LTE is becoming more and more valuable to equipment vendors and network operators alike. The ability to input a design at an appropriate level of abstraction, and having the tools to make necessary trade-offs early in the design process are becoming more and more crucial in this rapidly evolving marketplace.

Synchronous dataflow (SDF) [2] has been used widely as an efficient model of computation (MOC) to analyze performance and resource requirements when implementing DSP algorithms on various kinds of target architectures (e.g., see [3, 4, 5]). The SDF model has been incorporated in many commercial tools for DSP system design, such as ADS from Agilent, Signal Processing Designer from CoWare, and System Studio from Synopsys. In SDF semantics, DSP applications are modeled by directed graphs in which vertices (*actors*) correspond to computational blocks, and edges represent the passage of data between blocks. SDF imposes the restriction that the number of data values (tokens) that is produced on each output edge is constant per actor execution (*firing*), and similarly, the number of tokens consumed per firing is constant for each actor/input-edge pair. Thus, SDF does not accommodate actors that can have dynamically varying token production and consumption rates. Such “dynamic dataflow” actors are employed in many modern DSP applications, including the LTE physical layer, and therefore, when developing such applications, we must explore models of computation that are more general than pure SDF.

Parameterized synchronous dataflow (PSDF) is a generalization of SDF that allows dynamically-changing production and consumption rates that are formulated in terms of changes to parameters of *parameterized SDF graphs* (PSDF graphs) [6]. A PSDF graph can be viewed as a parameterized family of graphs such that each instance in the family (i.e., each specific setting of the parameters) corresponds to an SDF graph. PSDF significantly improves upon the ex-

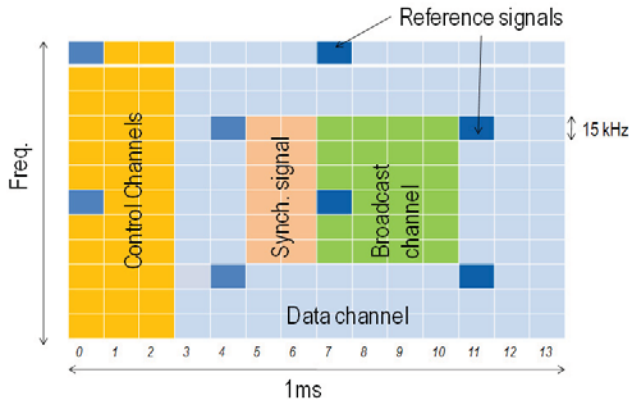


Fig. 1. Example LTE subframe showing multiplexing of various channels on a 2D time-frequency grid (not to scale).

pressive power of SDF while providing a framework in which many SDF analysis techniques can be naturally adapted into parameterized versions. For example, techniques for constructing efficient parameterized looped schedules have been developed for PSDF graphs [6]. These scheduling techniques can provide for efficient simulation or software synthesis from PSDF specifications.

In this paper, we apply PSDF to modeling the LTE physical layer protocol. A distinguishing aspect of our approach is that we develop a PSDF-based *hardware synthesis* framework for efficient utilization of parallel processing capabilities in FPGAs. In contrast, the parameterized looped schedules described above have been designed for single-processor, software-based implementations. Also, our work develops novel connections among model-based DSP system design, FPGA implementation, and next generation wireless communication systems, which lead to systematic, formally supported design methods for hardware implementation in this domain.

2. BACKGROUND

2.1. LTE downlink physical layer

The LTE downlink physical layer is based on the modulation and multiple access scheme called Orthogonal Frequency Division Multiple Access, or OFDMA subchannels [1]. OFDMA uses an inverse fast Fourier transform (IFFT) to divide a wideband channel into multiple narrowband. This creates a two-dimensional resource grid in frequency and time. In LTE, each element of this grid is called a *resource element*. This 2D grid allows multiplexing various physical channels, e.g., data and control channels, which could be intended for possibly multiple users. An example 1ms LTE subframe comprising 14 OFDMA symbols in the normal cyclic prefix mode is shown in Fig. 1. LTE can be configured for 6 different bandwidths, namely 1.4, 3, 5, 10, 15, and 20

MHz, but still maintain a constant 15 kHz subcarrier spacing. The LTE physical layer can also support multiple antenna transmission schemes, including transmit diversity, beamforming, and spatial multiplexing, but our paper primarily focuses on implementation for the single-antenna transmission mode.

2.2. Parameterized Synchronous Dataflow

Parameterized Synchronous Dataflow (PSDF) [6] extends the expressive power of SDF to manage DSP application dynamics in terms of run-time configuration of dataflow actor, edge, and subsystem parameters. A PSDF subsystem that is enabled for run-time configuration involves two separate “parameter configuration controllers,” which are referred to as the *init* and *subinit* graphs of the associated subsystem. These controllers provide two different levels of granularity in the run-time configuration processing — the *init* graph can form parameter configurations that are in general less restricted but also less frequent compared to the kinds of configurations that are allowed by the *subinit* graph.

The modeling discipline imposed by the *subinit* and *init* graphs in PSDF is designed to provide significant flexibility in how and when parameters are configured, while ensuring that configurations that affect the structure of subsystem schedules are allowed to occur only between iterations (in terms of SDF repetitions vectors) of the associated subsystems. This allows each subsystem to be viewed as a dynamically evolving sequence of SDF graphs whose SDF properties can change only at well-defined points in time (between SDF graph iterations). Such a structured view of dynamic dataflow graph execution is useful for efficient quasi-static scheduling [6].

3. PSDF MODEL OF LTE

3.1. LTE specification

Fig. 2 shows our PSDF model for a single-antenna LTE Base Station Modulator, which is the basis of our FPGA implementation. Each of the solid blocks corresponds to a PSDF actor whose production and consumption rate at its solid edges can change given the value of the parameters indicated by the dashed blocks communicated by the dashed edges. The data, control, and reference symbol generation blocks provide the QPSK, 16-, or 64-QAM symbols that are multiplexed via the Resource Element (RE) mapper. The RE mapper takes in different numbers of symbols s_1 , s_2 , and s_3 from the available input ports as a function of the number of control symbols ($N_{ctrl} \in \{1, 2, 3, 4\}$), subframe index ($Sfidx \in \{0, \dots, 9\}$), bandwidth configuration ($BW \in \{1.4, 3, 5, 10, 15, 20\}$), cyclic prefix mode ($CP_{mode} \in \{Normal, Extended\}$), and symbol index ($SymbIdx \in \{0, \dots, 13\}$). These symbols are multiplexed into $N_u \in \{72, 180, 300, 600, 900, 1200\}$ used subcarriers, which is a direct map from the bandwidth configuration BW . The Zero Pad block then takes in N_u symbols

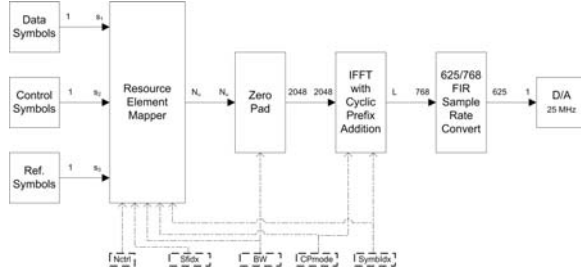


Fig. 2. PSDF Model for LTE BS Modulator.

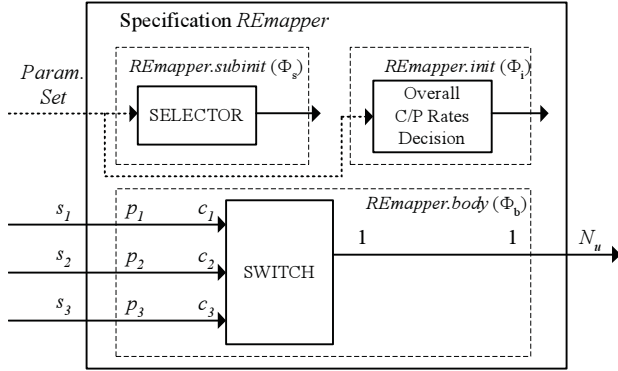


Fig. 3. PSDF specification of RE Mapper.

and appends zeros at the DC and edge subcarriers forming 2048 frequency domain complex values. The following block then performs a 2048-pt IFFT, and appends a cyclic prefix of length that is a function of the CP_{mode} and $SymbIdx$ parameters. The rate at the output of this block should be 30.72 Ms/s with a worst case bandwidth of 20 MHz, and so in order to interface to the 25 MHz D/A converter in our hardware platform, we require a fixed 625/768 FIR rational sampling rate converter.

3.2. PSDF Modeling Details

A PSDF specification for the RE mapper is shown in Fig. 3. Since there are different bandwidth configurations allowed, and each symbol of the LTE subframe is composed of different combinations of physical channel symbols (see Fig. 1), production and consumption rates in the RE mapper subsystem can be changed across OFDMA symbols, i.e., across the invocations of the RE mapping subsystem. Meanwhile, in order to multiplex the combination of physical channel types in each OFDMA symbol, the appropriate input edge is connected to the output edge for each resource element in the OFDMA symbol during each invocation of the RE mapper. We have likewise modeled the other processing blocks in the downlink LTE physical layer protocol, and have verified that PSDF has sufficient expressive power for describing the full functionality of our target LTE protocol.

PSDF specifications support *hierarchical* reconfigurable subsystem modeling structures in that a PSDF specification can be abstracted as a hierarchical PSDF actor, and embedded in a parent (higher level) PSDF graph. For example, a PSDF abstraction of the RE mapper in Fig. 2 is considered as a PSDF specification consisting of a *body* graph (Φ_b), *init* graph (Φ_i), and *subinit* graph (Φ_s), as shown in Fig. 3.

Before the invocation of the *RE Mapper* PSDF specification, the *init* graph receives the parameter set, determines the physical channel data combinations in the particular OFDMA symbol, and counts the number of REs allocated for each physical channel to determine production rates on input edges and consumption rates on output edges in the parent graph. During the invocation of the specification, the subinit graph determines production and consumption rates on internal edges in order to switch the input edge connected to an output edge depending on the value of the received remapping matrix data at run-time. Based on the distribution of active and inactive edges, the body graph, which implements the computational core of the subsystem, can produce a sequence of data corresponding to the OFDMA symbol index. Hence, in the architecture of our parameterized dataflow framework, the body graph models the main functional behavior of the RE mapper, while the init and subinit graphs provide two different levels of control based on the given, dynamically arriving parameter sets.

3.3. PSDF Execution Model

Each LTE subframe is composed of multiple OFDMA symbols, and each OFDMA symbol in our PSDF specification is processed after all actors in the graph are fired at the rate determined by the repetitions vector of the enclosing graph. Because PSDF semantics guarantees that any specific configuration of a PSDF graph is an SDF graph, and that such configurations can only be changed between SDF graph iterations, there is always a well-defined repetitions vector that governs the processing of a given OFDMA symbol. For details on fundamental relationships between SDF graphs and repetitions vectors, we refer the reader to [2].

When executing the LTE FPGA implementation, we apply a self-timed execution model, which means that each actor should be fired as soon as all of its input edges have sufficient data. When actors execute and communicate on dedicated resources (so that resource contention is not an issue), this type of execution generally enhances throughput by facilitating the exploitation of parallel processing capabilities on the target hardware. This type of distributed-control execution model also avoids hardware and run-time overhead due to the stronger synchronization requirements that are associated with centralized-control schedules.

FPGA targets allow dataflow actors to be assigned onto independent, dedicated processing units that are implemented by FPGA slices. In such a computing environment, signal

Table 1. FPGA resource utilization for LTE implementation.

<i>Occupied FPGA Slices</i>	5,244 out of 14,720 (35%)
<i>Number of BlockRAM</i>	96 out of 244 (39%)
<i>Number of DSP48Es</i>	54 out of 640 (8%)

processing throughput can be significantly increased due to the possibility for simultaneous firings of multiple actors. To ensure valid, distributed firing rule checking in our PSDF-based implementation framework, we model empty memory spaces on dataflow graph edges by adding feedback edges with appropriate numbers of initial tokens (based on the sizes of the corresponding buffers) in the execution model graph (an intermediate dataflow graph representation used to map the application into hardware), and enable actors for execution using principles of efficient self-timed execution [7].

Wiggers et al. have employed a similar *backpressure-driven*, self-timed execution model to implement cyclo-static dataflow (CSDF) graphs in multi-processor system-on-chip devices [8]. Our approach in this paper differs in its exploration of PSDF, which is a significantly more dynamic form of dataflow compared to SDF or CSDF, and its application to FPGA implementation.

4. LTE PROTOTYPE IMPLEMENTATION

As a proof-of-concept of our PSDF LTE model, we have designed and implemented from the top down an LTE real-time base station emulator prototype. The prototype is based on a PXI-express system with an embedded real-time controller PC running a real-time operating system, which handles the link control, higher-layer software, and communication with an optional host PC via TCP-IP. The PSDF LTE model is designed in LabVIEW FPGA, and implemented on the PXI-5641R, Intermediate Frequency (IF) Transceiver module, which includes a Xilinx Virtex-5 SX95T FPGA with integrated 2-input and 2-output IF ports. The IF signals are then modulated onto a radio frequency carrier using the PXI-5610 2.7 GHz RF upconverter, and looped-back to a PXI-5600 2.7 GHz RF downconverter, where the downconverted IF signal is fed back to the IF Transceiver for receiver processing. The base clock for our experiments with this system is 160 MHz. Synthesis results from the experiments are shown in Table 1.

As an illustrative example, we detail the implementation of the 625/768 sample rate conversion block of Fig. 2, which converts a 30.72MSPS LTE signal to the DAC at 25MSPS. In order to save hardware resources, we divide the filter into a cascade of two rational resampling stages, namely a 25/24 and a 25/32 stage. Using the LabVIEW Digital Filter Design Toolkit (DFDT), the individual floating point rational filters are designed and the fixed point behavior of the overall filter is simulated. We then use Xilinx's FIR compiler to imple-

ment the filter using the IP integration node from NI-Labs. This node uses an XCO or VHDL file as imports to build a simulation and implementation model compatible with LabVIEW FPGA.

5. CONCLUSION

We have presented a framework for the modeling and FPGA implementation of an LTE downlink physical layer using the parameterized synchronous dataflow (PSDF) model of computation. The results of our study and our associated prototype provide a concrete demonstration of PSDF-based design and implementation techniques for emerging wireless communication systems. Due to its formal properties, support for systematic scheduling and implementation techniques, and capabilities for efficient frame-based dynamic dataflow modeling, PSDF is promising as a semantic foundation for future design tools, and as an architectural foundation for digital system design methodologies in the domain of fourth generation wireless communication systems.

6. REFERENCES

- [1] 3G Americas, *The Mobile Broadband Evolution: 3GPP Release 8 and beyond*, Feb. 2009.
- [2] E. A. Lee and D. G. Messerschmitt, "Static scheduling of synchronous dataflow programs for digital signal processing," *IEEE Transactions on Computers*, February 1987.
- [3] C. Hsu, S. Ramasubbu, M. Ko, J. L. Pino, and S. S. Bhattacharyya, "Efficient simulation of critical synchronous dataflow graphs," in *Proceedings of the Design Automation Conference*, San Francisco, California, July 2006, pp. 893–898.
- [4] C. B. Robbins, *Autocoding Toolset Software Tools for Automatic Generation of Parallel Application Software*, Technical report, Management Communications and Control, Inc., 2002.
- [5] Jens Horstmannshoff and Heinrich Meyr, "Efficient building block based rtl code generation from synchronous data flow graphs," in *DAC '00: Proceedings of the 37th Annual Design Automation Conference*, New York, NY, USA, 2000, pp. 552–555, ACM.
- [6] B. Bhattacharya and S. S. Bhattacharyya, "Parameterized dataflow modeling for DSP systems," *IEEE Transactions on Signal Processing*, vol. 49, no. 10, pp. 2408–2421, October 2001.
- [7] S. Sriram and S. S. Bhattacharyya, *Embedded Multiprocessors: Scheduling and Synchronization*, CRC Press, second edition, 2009.
- [8] M. Wiggers, M. Bekooij, P. Jansen, and G. Smit, "Efficient computation of buffer capacities for cyclo-static real-time systems with back-pressure," in *Proceedings of the IEEE Real-Time and Embedded Technology and Applications Symposium*, 2007.