

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.

For more information visit www.intechopen.com



System-Level Modelling of Heterogeneous System-on-Chip Architectures Targeting Multi-Standard Wireless Applications

Ali Ahmadinia¹, Balal Ahmad² and Tughrul Arslan²
¹*Glasgow Caledonian University,* ²*University of Edinburgh*
United Kingdom

1. Introduction

In this chapter, we propose a novel framework for system-level modelling of heterogeneous SoC (System-on-Chip) architectures with emphasis on reconfigurable components targeting multi-standard wireless applications including WiMAX (Worldwide Interoperability for Microwave Access) standard (IEEE 802.16e, 2005). System-level modelling for system-on-chip and hardware-software co-design has been gaining importance due to rising complexity of such systems, continual advances in semiconductor technology, and the productivity gap in design of complex systems.

The IEEE 802.16 standard, WiMAX technology, is a wireless broadband access standard which helps make the vision of pervasive connectivity a reality. WiMAX technology can provide up to tens of Mbps symmetric bandwidth over many kilometres. This gives WiMAX a significant advantage over other alternatives like Wi-Fi and DSL. The physical layer of WiMAX includes both downlink (channel coding and IFFT) and uplink (FFT, modulation and decoding) data processing. WiMAX standard is rapidly proving itself as a technology that will play a key role in fixed broadband wireless metropolitan area networks. The scalable architecture, high data throughput and low cost deployment make Mobile WiMAX a leading solution for wireless broadband services.

The physical layer of WiMAX provides the wireless means of transmitting raw bits among devices. At present, the WiMAX standard has been implemented on DSP or ASIC technologies such as the WiMAX solution based on the Freescale MSC8 126 DSP (Freescale, 2005) and Intel NetStructure WiMAX Baseband Card (Intel, 2006). These solutions cannot satisfy all the features required by current and future embedded systems, including higher performance, higher power efficiency and higher flexibility. Filling the gap between the high flexibility of DSPs and the high performance of ASICs, reconfigurable architectures offer an alternative solution for embedded systems. This chapter extends our work in (Ahmadinia et al., 2007) and presents specially tailored high performance and low power architectures, based on newly emerging custom reconfigurable cores and communication centric platforms, for the WiMAX physical layer. To address this, we have proposed a system level framework that enables automatic generation and optimization of heterogeneous SoC

architectures based on communication centric platforms. The WiMAX 802.16-2004 physical layer was ported on this platform.

In WiMAX physical layer, apart from the usual functions such as randomization, forward error correction (FEC), interleaving, and mapping to QPSK and QAM symbols, the standard also specifies optional multiple antenna techniques. This includes space time coding (STC), beam forming using adaptive antennas schemes, and multiple input multiple output (MIMO) techniques which achieve higher data rates. The OFDM modulation/demodulation is usually implemented by performing FFT and inverse FFT on the data signal.

Consequently, FFT and Viterbi are considered in this work are modelled in SystemC as hardware accelerators which are compute intensive modules in WiMAX receiver computing chain. The WiMAX application runs as a software core on the ARM7 (ARM, 1999) processor. A Reconfigurable FFT IP and a Viterbi decoder as hardware accelerators are integrated into the WiMAX chain, to validate their performance by co-simulating modules from different levels of abstraction.

2. Related Work

In (Rissa et al., 2002), a modelling technique for reconfigurable systems is proposed based on OCAPI-x1 – a modelling language similar to SystemC. A set of processes is used to model a run time reconfigurable process which is supervised by a HardWare Scheduler (HWS). In (Noguera & Badia, 2003), a modelling methodology for reconfigurable architectures based on discrete event systems is presented. In an object-oriented manner discrete event classes and objects form a reconfigurable system, which introduces all necessary modelling capabilities to cover run time reconfiguration. In (Enzler et al., 2003), a co-simulation environment is unveiled enabling performance estimation for a dedicated architecture, which couples a CPU together with a coarse grained, multi-context reconfigurable unit (RU). Performance evaluation is based on a cycle-accurate simulation including execution and reconfiguration times. The CPU behaviour is modelled by an extension of a CPU simulator, while the RU itself is implemented in VHDL. The approach addresses the impact of multi-context RUs and focuses on the coupling of a multi-context device and a controlling CPU. In (Qu et al., 2005), a method is proposed to model dynamically reconfigurable hardware in SystemC. For this purpose, a special component is introduced called dynamically reconfigurable fabric (DRCF), which includes a configuration scheduler, an input splitter, and an associated configuration memory. After analyzing the SystemC modules, the DRCF code is constructed using a template. During this process, the reconfigurable functionality is identified and integrated into different configurations within a DRCF. The DRCF modification enables modelling of context switching and bus traffic. A system-level framework called, Perfecto, for rapid explorations of different reconfiguration alternatives and performance evaluation is presented in (Liao & Hsiung, 2005).

Through additional information, e.g. task execution time or configuration time, Perfecto is able to estimate aspects like total execution time and overall area usage. In (Brito et al., 2007) a technique is proposed for dimensioning dynamic and partially reconfigurable FPGA area by using a SystemC approach.

Unlike above approaches, our methodology is applied to a real world application, namely WiMAX. Furthermore, our modelling captures the reconfigurable features of cores, not just

instantiating different components. Also, our models have been verified in a hybrid simulation framework.

3. System-Level Reconfigurable Core Modelling

System-level modelling for system-on-chip and hardware and software co-design has been gaining importance due to rising complexity of such systems, continual advances in semiconductor technology, and the productivity gap in design of complex systems.

In the past few years, the introduction of many transaction level languages such as SystemC and SpecC has been seen (Open SystemC Initiative, 2008). Transaction level modelling (TLM) is a top-down approach to system design widely adopted in the verification phase of the design. In TLM, the system is first described at a high level of abstraction where communication details are hidden, and then the description is refined with the necessary details (Cai & Gajski, 2003).

The key concept of TLM is the separation of functionality definitions from communication details, by using the concept of channel. A channel allows communication through methods calling. Then, two functional components, interconnected with a channel, communicate calling the methods exported by the channel. In this way, the communication detail is hidden in the channel definition. SystemC enables TLM through *sc_interface*, *sc_channel*, and *sc_port*. A channel interface can be derived from *sc_interface*, then the *sc_interface* can be implemented in a *sc_channel*. At this point, a *sc_module* can communicate through a *sc* channel, using a *sc* port connected to the *sc* channel.

WiMAX is one of crucial wireless standards which requires SoC implementation platforms. In general, WiMAX as a whole needs reconfigurable architecture, as the system should adjust a significant number of parameters based on the volume of data, data rate, type of modulation, number of channels, etc.

In the proposed framework, the hardware component has been described, using the SystemC description language, as a C++ class derived from the *sc* module class. Due to the fact that no *sc_module* object, neither any objects derived from this class, can be initialized after the beginning of the simulation process, it is not possible to model a module reconfiguration process as a new *sc* module initialization during the simulation phase. The approach followed in the framework to model the system reconfiguration using SystemC is based on the fact that the functionalities of a *sc_module* could be implemented as *sc_methods/sc_threads*. Therefore by using one of them as a wrapper to call a function pointer, we are allowed to change the functionality of modules, by varying only this function pointer.

Different to other approaches, that implement the complete reconfigurable area or the reconfigurable modules as buses, the reconfigurable modules remain in their original scope and the surrounding design must be changed only a little. Obviously the topology of a static design remains basically unchanged if some parts are made reconfigurable. The straightforward way to achieve this is to intercept the communication between static and reconfigurable modules at the channels, which interconnect those parts. The current configuration is simulated by forwarding channel events only to the currently active modules by using dynamic switches. Additionally, all channel accesses of inactive modules are suppressed. If the modules are either sensitive to certain events or make use of blocking channel accesses, such a system will behave like a reconfigurable design during simulation.

Still this is not a completely satisfying solution, since for every channel type a new switch has to be built. This leads to the concept of a reconfigurable interface (*sc_rec_interface*), which provides a framework for building these switches.

A *sc_rec_interface* is a special switch, designed to connect a static channel with a port of a reconfigurable module, as shown in Figure 1. During the simulation, accesses to the corresponding port from within the reconfigurable module are then forwarded to the static channel. Additionally, any required events, the reconfigurable module is listening to (via sensitivity or dynamic *wait()* statements), is forwarded from the static channel to the module. Multiple reconfigurable modules can be bound to a single *sc_rec_interface*. During the simulation, the actual reconfiguration operations change the data-flow through the interface depending on the reconfiguration state of the connected modules. If all ports of a reconfigurable module are equipped with *sc_rec_interface*, no port can be triggered from the outside, if the module is inactive (not configured).

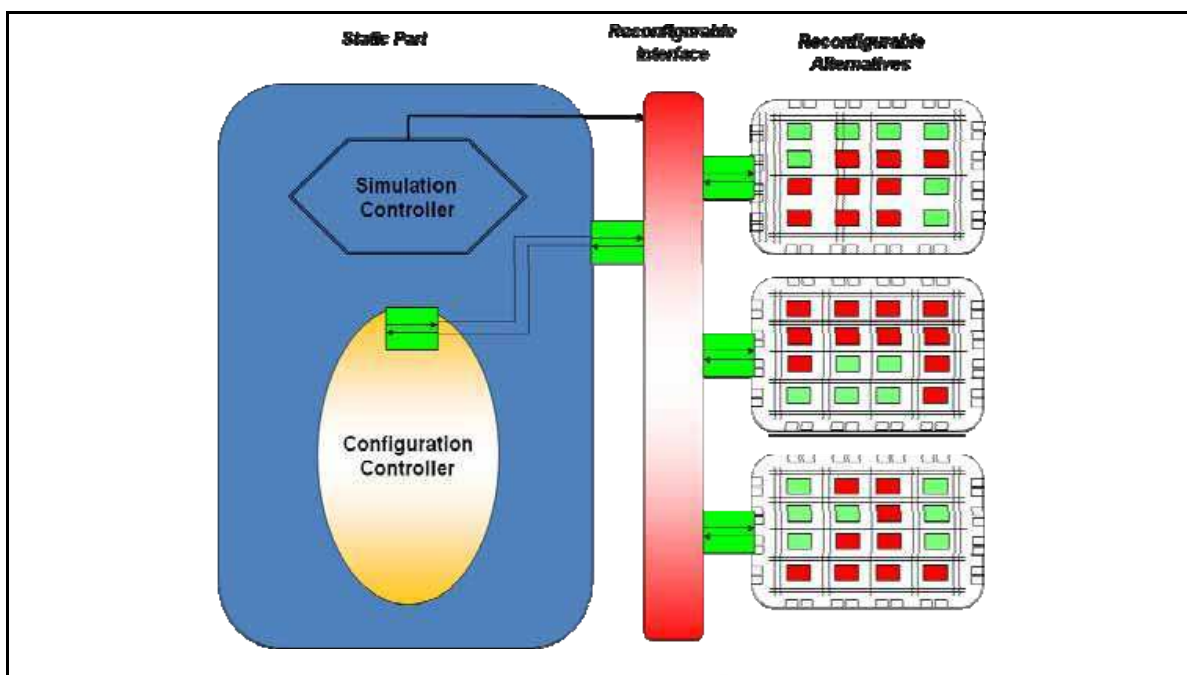


Fig. 1. A TLM-based Reconfigurable Design

Therefore, no outbound traffic should occur any longer, since the module's processes are no longer triggered. Nevertheless, technically it is possible that a module keeps on triggering itself (for instance by a member of type *sc_clock*). In this case outbound traffic is suppressed and a warning is reported to the designer.

For example, in the following code a manual generation of a reconfigurable module is shown. Module *Smod_reconfig* is derived from *reconfig* module and static module *Smod*. The constructor starts the finite state setup to machine that takes care of the reconfiguration state of the module and calls *reconfig_reset* the module at start-up. In member function *reconfig_setup* *a* is registered to be reset to zero and *b* to be preserved during reconfiguration. Modelling of configuration times is split into loading and activation delay. Integrating a reconfigurable interface into a design is done analogously to the integration of a channel. Here the usage of a FIFO reconfigurable interface is shown.


```
class Smod_reconfig:
public reconfig_module,
public Smod {
protected:
inline void reconfig_setup();
public:
Smod_reconfig( sc_module_name _name ):
Smod(_name)
{
reconfig_init();
// Initialize reconfiguration
// module functionality
reconfig_setup();
// Setting up reconfiguration
}
};

virtual inline
void Smod_reconfig::reconfig_setup() {
reconfig_reset<int>(a,0);
// reset a to 0
reconfig_store<sc_signal<int > > (b);
// store b
set_config_time(sc_time(25,SC_MS));
// module needs 25 ms to configure
set_exec_time((sc_time(1.5,SC_MS));
// and 2 ms to execute
set_removal_time((sc_time(2,SC_MS));
// storing variables
}

Smod_reconfig module1, module2, module3;
// instantiate reconfigurable modules
sc_fifo< int > fifo;
// instantiate (static) FiFo
sc_rec_interface<sc_fifo_out<int>>
rec_int1;
// instantiate reconfigurable interface
// for sc_fifo_out<int> port
rec_int1.static_port( fifo );
// connect the static channel
rec_int1.bind( module1.out );
// bind reconfigurable modules' ports
// to interface
rec_int1.bind( module2.in);
rec_int1.bind( module3.in);
```

4. WiMAX Application

To evaluate the proposed methodology and SystemC reconfigurable cores, WiMAX is chosen as a real application for the SoC design. WiMAX technology is standard widely known as worldwide wireless networking standard, the technology that addresses the interoperability issues proposed by the IEEE 802.16.

The WiMAX technology is globally adopted and it has unlimited demand in the new markets for wireless chips. In this work, the receiver section of the WiMAX system is the area of interest. The main constraint in the receiver section is the performance. There are upcoming trends to commercialize the technology, it evidently deserves attention while determining infrastructure of the receiver sections. The system designers are looking into the type of tools that would help them to design the WiMAX system (WiMAX Forum, 2008). The typical WiMAX receiver measurements includes measurements of sensitivity, the maximum input level, adjacent-channel and alternate-channel rejection, reference timing accuracy, and SS uplink transmit time tracking accuracy (Heegard & Stephen, 1999). As the receiver section of the WiMAX system suffers from the challenges with performance, it is essential to obtain optimal system performance. This can be achieved by obtaining optimization in the individual modules. One such module dealt in the following sections is the de-interleaver module of eth WiMAX receiver section (Larsen, 1973).

The WiMAX system is capable of working on the voice, data and video applications. There could be various applications of the WiMAX, could be used for VoIP, radio, IPTV, Video Teleconferencing, Displays, Interactivity, social networking, public works, sensor networks and event-based media (WiMAX Forum, 2008).

Figure 2 shows block diagram of a simple WiMAX system. The orthogonal frequency division multiplexing (OFDM) is one of the modules in the WiMAX system which supports many modulation schemes. The OFDM modulation/demodulation is usually implemented by performing FFT and inverse FFT on the data signal.

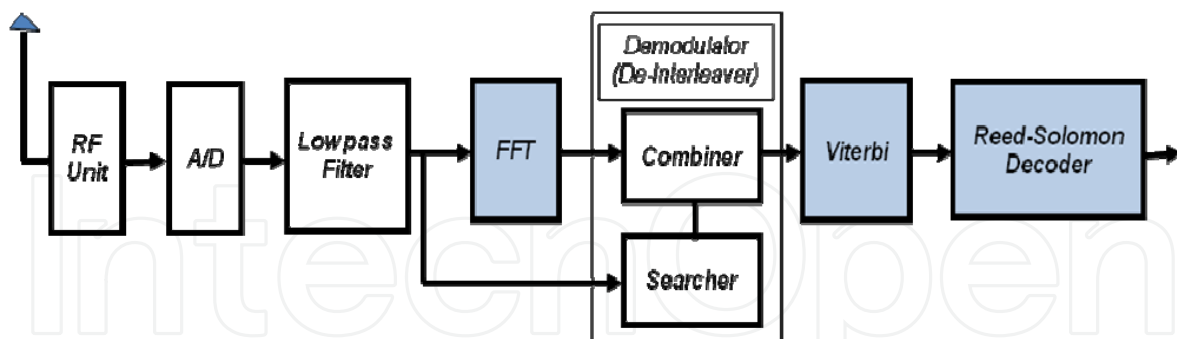


Fig. 2. Block Diagram of WiMAX Receiver

As a result, FFT and Viterbi are considered in this work as case studies which are computational intensive modules in WiMAX computing chain. A Reconfigurable FFT core and a Viterbi decoder as hardware accelerators are integrated into the WiMAX chain, to validate their performance by co-simulating modules from different levels of abstraction.

5. Reconfigurable FFT Implementation in SystemC

Discrete Fourier Transform (DFT) is one of the most widely used algorithms in digital signal processing for spectral analysis and for filter implementations. It operates on an N point sequence of numbers $x(n)$. However, the DFT has limitation in speed for larger transform sizes. As a result, Fast Fourier Transform (FFT) is used, which is an efficient algorithm for computation of the DFT.

The implemented FFT block is based on a radix-2 algorithm and can be reconfigured for different FFT sizes from 16-points to 2048-points. The input data is read as a signed 16-bit fixed-point number with 10 fractional bits. Twiddle factors and output values use the same representation. Internally, computation is performed with fixed-point arithmetic. The input samples and output transformations are externally inferred as 16-bit integers. A fixed-point functional version of the FFT was developed to be compatible with the original synthesizable FFT core to verify the results, working at the highest level of abstraction.

Figure 3 shows block diagram of implemented reconfigurable FFT. The details of each block are discussed below:

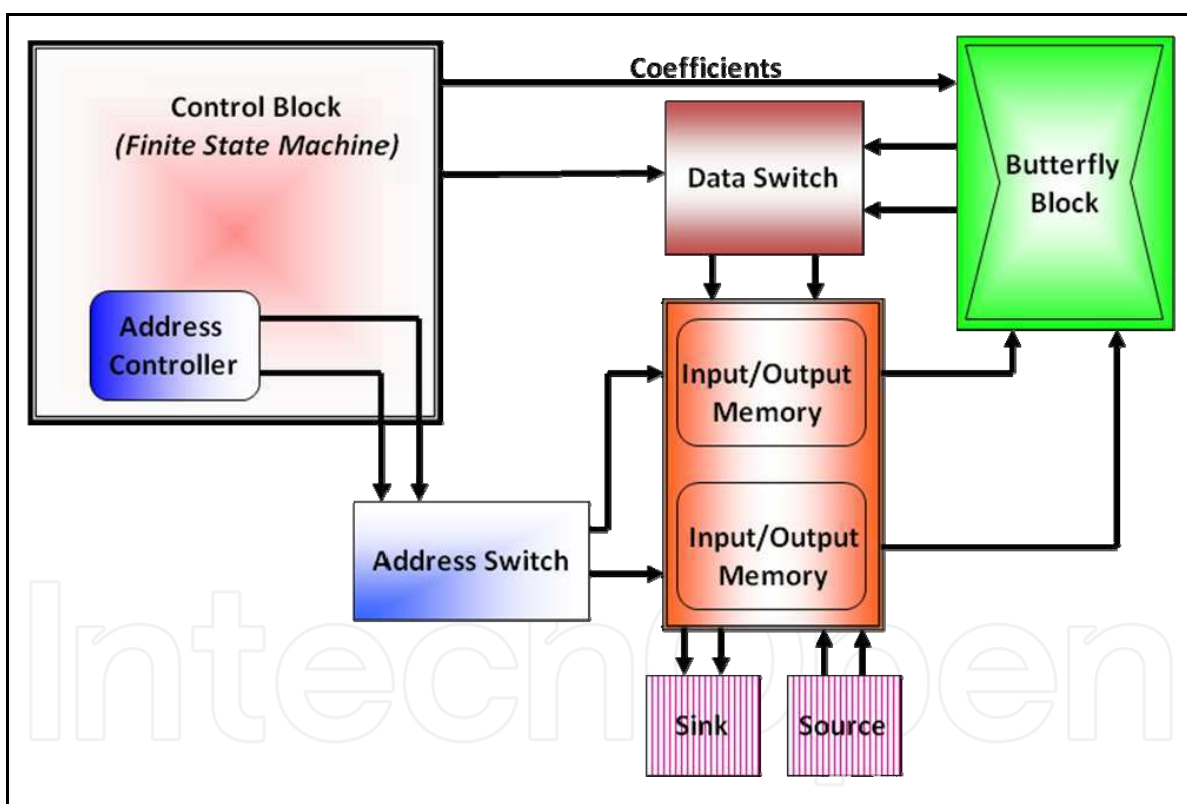


Fig. 3. Block Diagram of Implemented Reconfigurable FFT Processor

- Control Block:** This is the main control unit of the FFT fabric. Two different counters are used for butterfly operations. First one is used to count the number of butterfly operations in each pass and then the second one is used to count the number of passes. It generates the inputs for Address Controller, the control signals for Data Memories and the configuration bits for data and address switching.

- **Butterfly Block:** This butterfly block has two parallel inputs and two outputs. Each butterfly computation consists of a number of additions, subtractions and multiplications.
- **Data Memory Interface:** As shown in Figure 3, this architecture has two data memories. Each divided into sixty-four 8x32-bit dual port memories. Therefore, each data memory has a capacity of 512x32-bit.
- **Address Controller:** This block is designed to generate addresses for data memories. Different from the address blocks in conventional fixed-point FFT fabrics, these reconfigurable ones can generate addresses for various FFT sizes. In (Ma & Wanhammar, 2000), an efficient coefficient address generation method was proposed by Cohen. According to his work, for a $N = 2n$ point FFT, at the i th butterfly calculation in the j th pass, $W_k^0 = (\exp(2\pi/N))^{k^2}$ is the coefficient needed. The k can be obtained from i by masking out its $(n-1-j)$ least significant bits.
- **Data and Address Switching:** The Data Switch receives the outputs of the butterfly module and routes them into the right data memory. For all the FFTs ranging from 16-2048 points, the address switch input width is 10 bits. Address switching works in parallel with data switching in order to allocate the right addresses for the data which are being processed through the data switch block.

6. Reconfigurable Viterbi Implementation in SystemC

All communication systems involve three main subsystems: the transmitter, the channel and the receiver. The channel medium in general attenuates the signal and as a result the noise level of the channel and the noise of the receiver will cause errors at the receiver. Here comes the task of the engineers to design communication systems with the least possible errors while satisfying the design constraints, such as allowable transmitted energy, allowable signal bandwidth and cost. To optimize the performance of the communication systems, channel coding is used. The term channel coding refers to the techniques performed on the information bearing digital signal, which requires the use of redundancy to detect and possibly correct errors. The main techniques of channel coding are: automatic repeat request (ARQ) and forward error correction (FEC).

One of the most efficient FEC techniques is the convolution encoding and Viterbi decoding. Convolution codes operate on serial data, one or few bits at a time, and they are usually described by the code rate and the constraint length. The code rate, k/n , is expressed as a ratio of the number of bits into the convolution encoder (k) to the number of channel symbols output by the convolution encoder (n), which gives a bandwidth expansion of n/k . The constraint length parameter is the length of the convolution encoder.

The implemented reconfigurable Viterbi module can decode for different constraint lengths ($K=3$, $K=5$, and $K=7$). It would be clocked with external clock and at the beginning of the burst sequence decoder would be synchronized with the input sequence (i.e. reset to the beginning known state). Two main modes of operation are implemented: normal and self-test modes. In normal mode, decoder is initialized to the known state, with *Synchronizer* pulse signal asserted while *Self Test* is de-asserted, one cycle prior to the beginning of data sequence. Eleven cycles after the *Synchronizer* pulse was asserted, *Data Valid* becomes active signalling that first valid bits have appeared at the *Decoded Bit* output of the decoder. In self-

test mode, the decoded sequence is compared with delayed original bits from pseudo random generator and in case of an error, the *Fail* is asserted.

The entire Viterbi decoder unit can be divided into three distinct functional units: Add Compare Select unit, Survivor Path unit, and Controller (See Figure 4). First, the operation of each unit will be explained, and then focus on the interface between the units.

- Add, Compare, and Select Unit (ACS):** consists of 64 single-state ACS units and branch metric unit (BMU). BMU generates the Hamming distance metrics for the input encoded sequence, arriving from controller. The inputs of each single-state ACS unit are path metrics from the two states in the previous time (trellis) stage, and the two branch metrics each for one state to state transition. The single-state ACS unit adds corresponding branch metrics to path metrics to obtain the total path metrics for the resulting state and selects the smaller of the two total path metrics. The need for explicit metric rescaling is avoided by using implicit metric rescaling. In this way, the new path metric for the given state is calculated and latched in order to be fed back to the inputs of single-state ACS units in the next cycle (trellis stage). All single-state ACS units are interconnected in shuffle-exchange fashion.

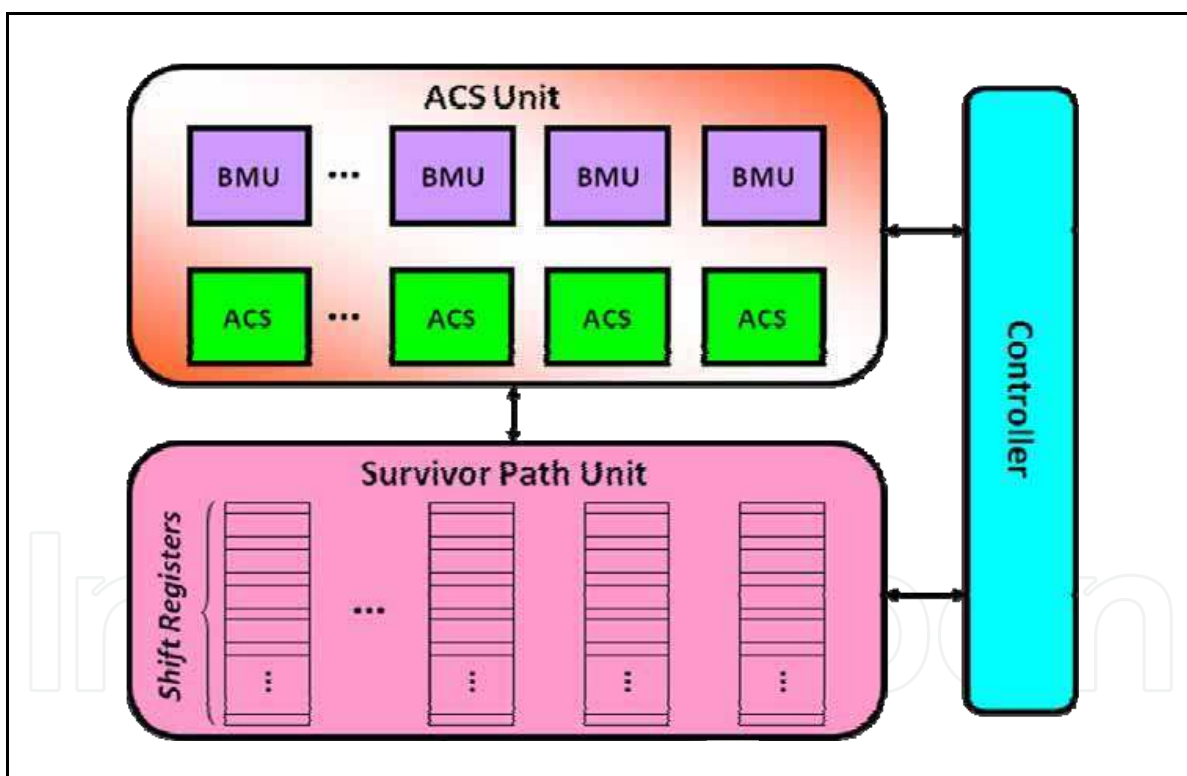


Fig. 4. Block Diagram of Implemented Reconfigurable Viterbi Decoder

- Survivor Path Unit (SPU):** consists of a matrix of cells placed and interconnected according to the trellis diagram. Each cell consists of a latch cell and a combined latch multiplexing cell. Each row of cells corresponds to one of the states in the trellis diagram, and at each time instance holds the complete decoded history of the survivor path ending in that particular state.

As the top registers of the survivor paths change states at each new time instance, the complete history shifts, horizontally for one cell and vertically to the row corresponding to the new state of the survivor path. Each cell in a row has the same select input, which is the decision bit from the ACS unit for the corresponding state. Two data inputs are from the cells in the rows corresponding to two possible previous states in trellis diagram. The first column of cells has fixed inputs which correspond to the decoded bits for the given state and type of convolution code used. The last column contains the decoded bit. Ideally, all bits in decoded column would be the same, since all the paths converge to the common trunk, but sometimes that does not happen, and bits are not the same, thus raising the question of which of them to choose as the final result. The majority function is implemented to select the most represented value in the column, and thus minimize the decoding error.

- **Controller Unit:** consists of the self-test module, delay line, mode selection logic and majority function logic (See Figure 5). Self-test module consists of a convolution encoder and a pseudo-random generator. Mode selection logic selects between external encoded inputs and internally generated self test inputs, based on the value of the Self Test signal. Majority function logic selects the most represented value in decoded column, and outputs the decoded bit. Shift register is used both as a delay line, in self test mode, and as a counter for asserting the Data Valid signal.

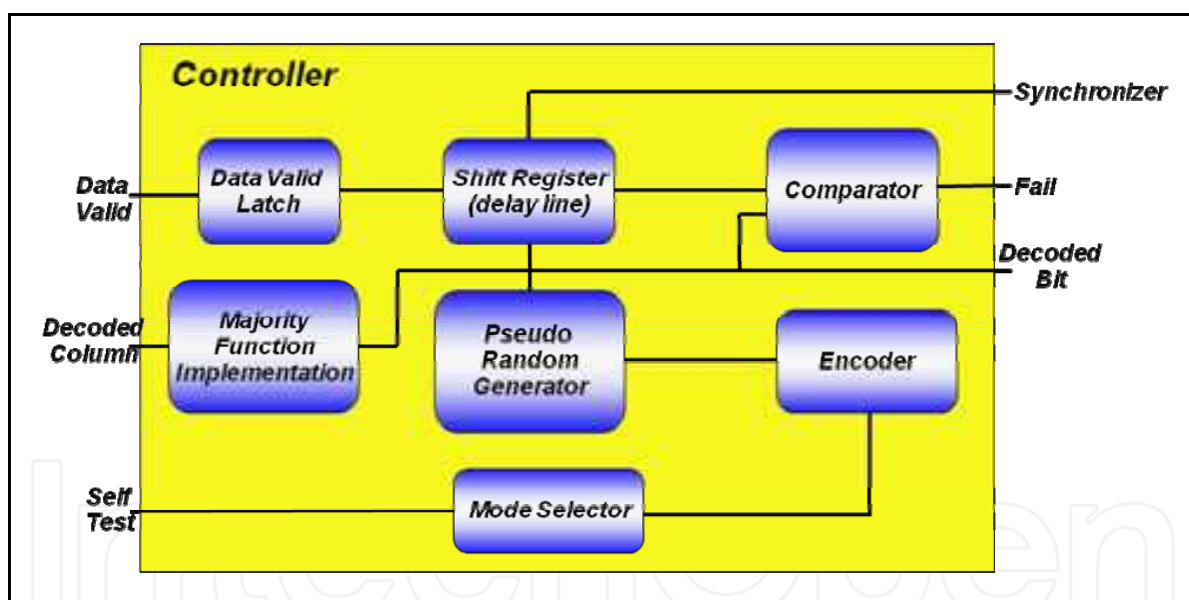


Fig. 5. Architecture of Controller in Viterbi Decoder

The implementation of the classical state machine controller is avoided, since it would just increase the number of latches used in the controller. Controller unit outputs Select Initial signal which loads the ACS units' path metric registers with the known initial metrics. Branch metric signals generated in controller are inputs to the corresponding single-state ACS units. Controller inputs are the decoded column signals from the survivor path unit. The decision bit outputs of the ACS units are wired to the inputs of the survivor path unit.

7. Result Analysis

A new design evaluation framework was constructed. This involved plugging the developed SystemC modules of the FFT and Viterbi into an industrial C-based implementation of the complete WiMAX telecommunication standard. Functionality evaluation was performed through the complete model. This is demonstrated as received bit rate performance of the overall receiver. This is shown in Figure 6, which demonstrates similar profiles between exact C-based model of the WiMAX for a range of FFT sizes: 128-2048 points. Since, in Mobile WiMAX (IEEE 802.16e, 2005), the FFT size of OFDM part is scalable from 128 to 2,048. As shown in Figure 6, the WiMAX bit rate is accelerated by plugging in SystemC cores, which gives more speed up for smaller FFT sizes due to decreasing FFT computation time.

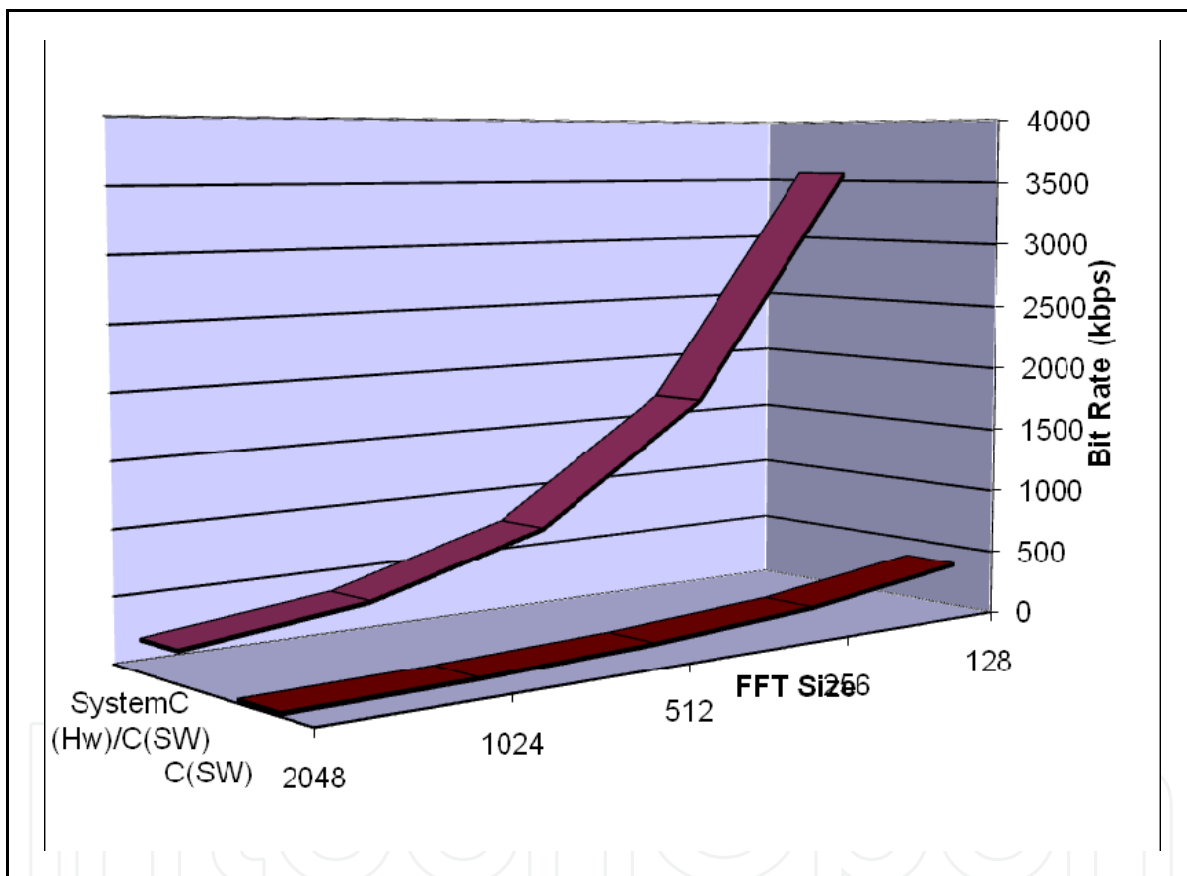


Fig. 6. Impact of reconfigurable SystemC cores on WiMAX Performance

To investigate power and area characteristics of the main blocks of WiMAX receiver, and their impact on the overall performance of system, we have used our RTL-version of these blocks. Figure 7 illustrates the change in power and area as FFT size increases. We have also investigated the impact of different code rates of Viterbi decoder compatible to (IEEE 802.16e, 2005), ranging from 12 to 78. It showed that the code rate has nearly no influence on area and power figures of the system, because by changing the code rate, the structure of Viterbi decoder remains the same, just one more code word has to be added in the same clock cycle.

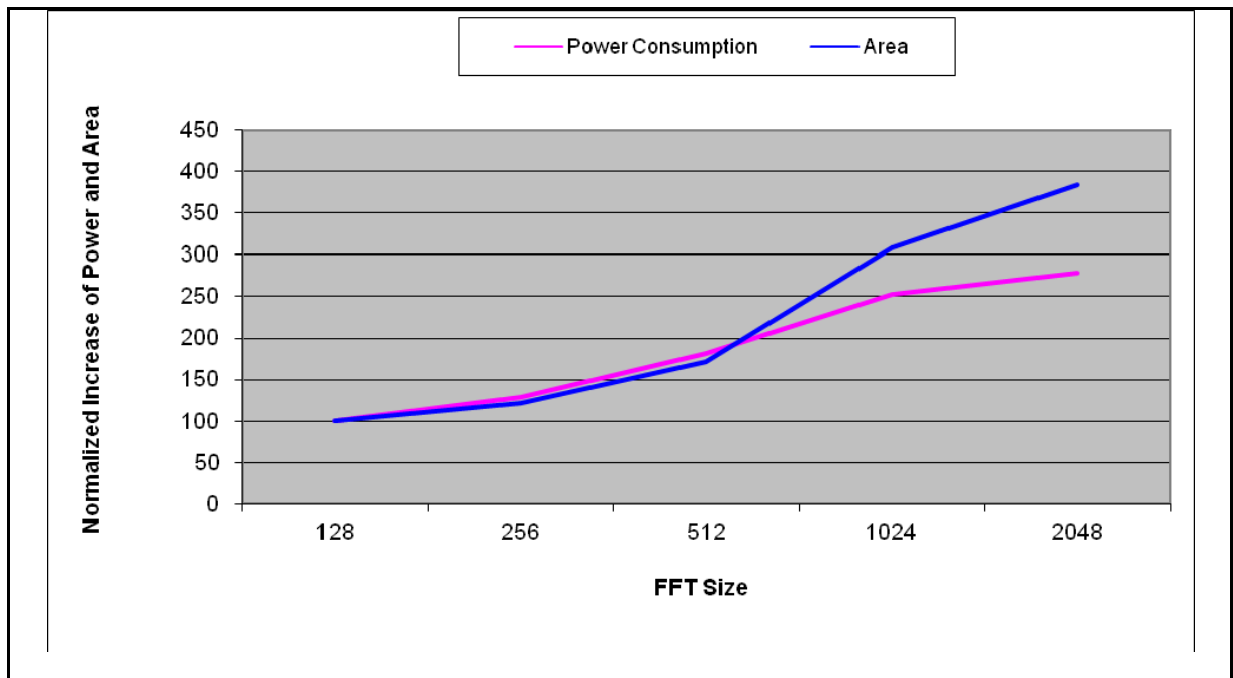


Fig. 7. Impact of reconfigurable FFT on Power and Area of WiMAX

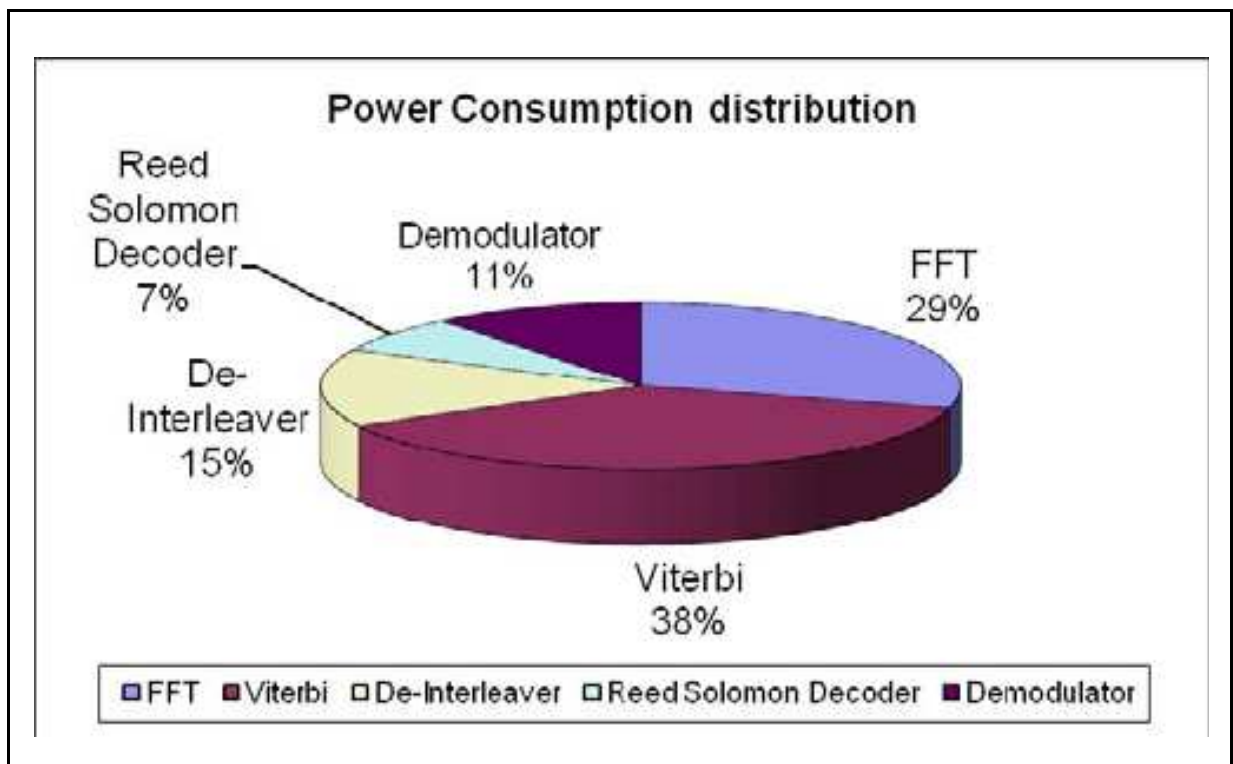


Fig. 8. Power Analysis of WiMAX Blocks

The power and area distribution of each module in WiMAX is demonstrated in Figure 8 and Fig. 9. Gate level simulation is carried out and Synopsys design compiler is used to calculate power and area results. A 0.13 micron technology library is used for ASIC synthesis. The

power consumption of Viterbi decoder is the highest among other blocks of the WiMAX receiver, and FFT gives the largest area overhead compared to others.

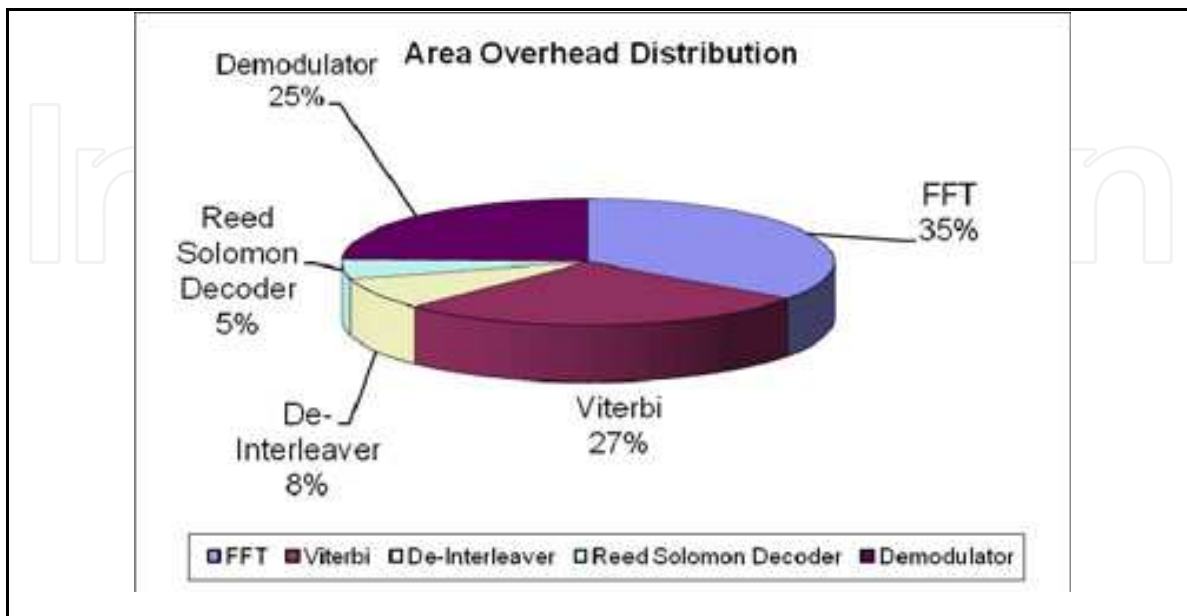


Fig. 9. Area Distribution of Blocks in WiMAX

8. Conclusion

In this paper, a new multi-context representation of system level modelling is proposed in order to capture the reconfigurable features within different embedded reconfigurable cores. A hybrid SystemC-HDL co-simulation framework was presented in order to demonstrate interoperability of the designed reconfigurable hardware block. As a case study, we applied our modelling strategy in order to design a complete WiMAX compliant receiver where two new reconfigurable fabrics are employed for the implementation of the performance bottlenecks within these standards, namely FFT and Viterbi processing tasks. Such a receiver needs to reconfigure the parameters and these tasks in order to balance performance against issues such as channel condition, spread delay, etc. We showed how our modelling approach can be utilised to capture complete system functionality and provide performance and power consumption figures for system design.

These reconfigurable SystemC models and their power and area attributes will be used in library of an SoC design and generation tool, for automatic parameterizable embedded core integration, functional verification at the system level, and optimization in terms of power and area.

9. References

- Ahmadinia A., Ahmad B. & Arslan T., "System-level Modelling and Analysis of Embedded Reconfigurable Cores for Wireless Systems", In *Proceedings of International Conference on Field-Programmable Logic and Applications (FPL)*. 27-29 August 2007. Amsterdam, Netherlands, pp. 757-760.
- ARM. ARM7 TDMI Datasheet. ARM. 1999. <http://www.arm.com>.
- Brito A., Kuehnle M., Huebner M., Becker J. & Melcher. U., Modelling and simulation of dynamic and partially reconfigurable systems using SystemC. In *Proceedings of ISVLSI*, pages 35-40, 2007.
- Cai L. & Gajski. D., Transaction level modeling: an overview. In *Proceedings of the CODES+ISSS conference*, pages 19-24, 2003.
- Enzler R., Plessl C. & Platzner M., Co-Simulation of a Hybrid Multi-Context Architecture. In *Proceedings of Engineering of Reconfigurable Systems and Algorithms Conference*, pages 174-180, 2003.
- Freescale Semiconductor. Implementing WiMAX PHY Processing Using the Freescale MSC8126 Multi-Core DSP. Freescale Semiconductor. 2005. <http://www.freescale.com>.
- Heegard, C. & Stephen B. W., *Turbo Coding*, Boston, Kluwer Academic Publishers, 1999.
- IEEE 802.16e: Air interface for fixed and mobile broadband wireless access systems. 2005. <http://standards.ieee.org/getieee802/802.16.html>
- Intel Corporation. Intel NetStructure WiMAX Baseband Card Product Brief. Intel Corporation. <http://www.intel.com>.
- Larsen K. J., Short convolutional codes with maximal free distances for rate 1/2, 1/3 and 1/4, *IEEE Trans. Inf. Theory*, pp. 371-372, May 1973.
- Liao C. & Hsiung. P., A SystemC-based Performance Evaluation Framework for Dynamically Reconfigurable SoC. In *Proceedings of the VLSI Design / CAD Symposium*, pages 444-447, Aug. 2005.
- Ma Y. & Wanhammar L., A hardware efficient control of memory addressing for high-performance FFT processors. In *IEEE Transactions on Signal Processing*, 48(3):917-921, 2000.
- Noguera J. & Badia R., System-level power performance trade-offs in task scheduling for dynamically reconfigurable architectures. In *Proceedings of the CASES conference*, pages 73-83, 2003.
- Open SystemC Initiative, 2008, www.systemc.org.
- Qu Y., Tiensyrja K. & Soininen J.-P., SystemC-based Design Methodology for Reconfigurable System-on-Chip. In *Proceedings of the 8th Euromicro Conference on Digital System Design*, pages 364-371, Los Alamitos, CA, USA, 2005.
- Rissa T., Vasilko M., & Niittylahti. J., System-Level Modelling and Implementation Technique for Run-Time Reconfigurable Systems. In *Proceedings of FCCM Symposium*, pages 295-296, Los Alamitos, CA, USA, 2002.
- WiMAX Forum, 2008, www.wimaxforum.org.



WIMAX New Developments

Edited by Upena D Dalal and Y P Kosta

ISBN 978-953-7619-53-4

Hard cover, 442 pages

Publisher InTech

Published online 01, December, 2009

Published in print edition December, 2009

WiMAX (Worldwide Interoperability for Microwave Access) is a wireless broadband access network named by industry group called the WiMAX forum formed in June 2001. It is Wireless MAN with IEEE 802.16 family standards. Loosely, WiMAX is a standardized wireless version of Ethernet that enables the last mile, intended primarily as an alternative to wire technologies (such as Cable Modems, DSL and T1/E1 links) to provide broadband access to customer premises. Mission of the WiMAX forum is to promote and certify compatibility and interoperability of broadband wireless products. This book touches most of the above issues in form of 22 individuals' papers containing research work in WiMAX domain in particular. WiMAX has two important standards/usage models: a fixed usage model IEEE 802.16-2004 for Fixed Wireless Broadband Access (FWBA) and a portable usage model IEEE 802.16e-2005, which is mainly concentrated on Mobile Wireless Broadband Access (MWBA). Both are released as standards and amendments are available in form of drafts. Higher data rate transmissions (@ 100 Mbps) are achieved in IEEE 802.16-2004 WiMAX through LOS communications which incorporate a stationary transmitter and receiver but IEEE 802.16e supporting NLOS communication is much complicated and little less bit rate is achieved. 2-11 GHz licensed band is the range of frequencies with TDD and FDD supports. The book will provide a wide horizon to visualize the WiMAX technology and its developments leading towards 4G systems. It will provide a good platform to the researchers with clues to the innovative ideas in WiMAX domain. I wish all the best to the authors and readers of this book in their successful research of WiMAX technology.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Ali Ahmadinia, Balal Ahmad and Tughrul Arslan (2009). System-Level Modelling of Heterogeneous System-on-Chip Architectures Targeting Multi-Standard Wireless Applications, WIMAX New Developments, Upena D Dalal and Y P Kosta (Ed.), ISBN: 978-953-7619-53-4, InTech, Available from:

<http://www.intechopen.com/books/wimax-new-developments/system-level-modelling-of-heterogeneous-system-on-chip-architectures-targeting-multi-standard-wirele>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元

www.intechopen.com

Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen

IntechOpen

© 2009 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen