A Software Development and Validation Framework for SDR Platforms

Jeroen.Declerck@imec.be
Outline

- IMEC SDR Platform
- Problem Statement
- Framework (XMSF) Implementation
  - XMSS server
  - Graphical logger
  - IMEC SDR
  - TCP/IP connection
- Framework (XMSF) Simulation Results
  - 802.11a PHY
  - 802.11 MAC
  - Platform Exploration
- Conclusion
IMEC SDR Platform: Functional View

Reconfigurable Analog Front End

Digital Baseband Platform = BEAR

- DFE
- BBE
- OMD

- FE I/F Synchro AGC
- Encoding (De)Interl. (De)Mapping (I)FFT
- CRC FEC Scrambling

174MHz – 6GHz
802.11a/b/g/j/n
802.15
802.16

UMTS-TDD/FDD
HSDPA
3GPP-LTE
DAB/DBM/DVB-H

IMEC SDR Platform: BEAR Chip

- ARM
  - MAC
  - S/W PHY
  - HAL
  - RTE

- RAM

- DMA1

- DMA2

- IRC

- Timer

- ITCM
- DTCM

- AMBA

- FECE
  - ASIC Forward Error Correction, (de)scrambling CRC

- FECE
  - ASIC Forward Error Correction, (de)scrambling CRC

- BBE1
  - ADRES Processor Compiled C-code (De)Modulation Encoding

- BBE2
  - ADRES Processor Compiled C-code (De)Modulation Encoding

- DFE
  - ASIP synchronization AGC Analog FE I/F

- Host I/F
Problem Statement: Intention

- Simulation of native PHY/MAC software code on a specific SDR platform (IMEC SDR)
- Visualize the impact on the hardware components (DMA, timer, BBE, DFE, OMD...)
- Existing solution (see next): TLM
Problem Statement: TLM

TLM model of IMEC SDR

- Byte accurate
- Bus simulations
- Sw/Hw cosimulations
- Full Hw validation
- Runs native code

- No channel
- No multiple terminals
- No TCP/IP
- No full MAC/PHY s/w validation possible
- 2 minutes simulation / packet

Proceeding of the SDR 08 Technical Conference and Product Exposition. Copyright © 2008 SDR Forum. All Rights Reserved
Problem Statement: Requirements

- Development/ Simulation/ Validation platform for system code (SDR PHY/MAC control software)
- Multiple terminals connected together
- Simulate native compiled PHY/MAC code
- Fast enough for network simulations

XMSF
(cross layer –XL- MAC Simulation Framework)
XMSF: XMSS

XL MAC Simulation Server (XMSS)

- Time space model
- Ether bandwidth model
- Log reflector

XMSS: Server process that accepts connections from clients

- Add Events to a global timeline
- Callback when event elapses

- Perfect Air Interface Model
- Add complex channel models through dll libraries

- Centralized logging point
- Graphical analyzer using logs

Client 1

Client 2

Client n
XMSF: Implementation of PHY/MAC

Simulation of the native PHY/MAC code using the XMSF

802.11 DCF MAC (Time Critical Functions)

802.11a Software PHY

BBE Abstraction layer
DFE Abstraction layer
DMA Abstraction layer
Timer Abstraction layer
OMD Abstraction layer

XMSF Interface (API)

HAL

Simulation Server (XMSS)

Time space model
Ether bandwidth model
Log reflector

Proceeding of the SDR 08 Technical Conference and Product Exposition. Copyright © 2008 SDR Forum. All Rights Reserved
XMSF: Integration with TCP/IP
XMSF Simulation Results: Platform Exploration

- 802.11 DCF MAC
- Transmit/receive
- Preamble/dat a processing
- Synchronize timing (at end)
- Task Graph Based Timing Simulation
- Redundant model of environment timing
- Simulation Server API
XMSF: Other Features

- Coupling between CoWare ConvergenSC tools (SystemC/VHDL co-simulator), the XMSS server and graphical analyzer.
- Synchronize the XMSS server with the Spooks streaming video server
Future Work

- GUI for channel model
- 802.11n (MIMO) PHY
- 3GPP-LTE PHY/MAC
- Handover
- Scheduler for multiple streams on SDR