SystemC AMS extension –
alignment with SystemC-TLM

Contribution to AMS-TLM Workshop
January 2008

Martin Barnasconi
Requirements

- Interface and communication: SystemC TLM 2.0 compliant
  - using PVInitiator_port and PVTarget port
  - support interrupt based synchronization (using sc_core::sc_out port)

- Register interface based on the (CoWare) SCML memory model
  - register interface splits communication in architecture context from functional modeling
  - note: internal register communication/callbacks not part of TLM standard

- Compatibility and Interoperability with digital functional models and subsystems

- Approach: Wrap SystemC AMS subsystem into ‘TLM shell’
  - embed functional model in architecture block
  - two options
    - option 1: Plain SystemC FIFO interface
    - option 2: communicate directly to SystemC AMS extension / TDF solver
SoC Architecture high-level view

not available!

HW assist
CPU
Memory

DSP
Memory
UART
GPIO

DSP bus
adapter
adapter

system bus
peripheral bus

TLM model

analog mixed-signal part

‘digital’ part

SystemC
DE

anno 2006

NXP
SoC Architecture high-level view

TLM model

SystemC
AMS

TDF

HW assist
CPU
Memory

DSP

adapter

DSP bus

adapter

system bus

peripheral bus

Tx/Rx

analog
mixed-signal
part

‘digital’ part

UART

GPIO

anno 2007/8
SoC Architecture high-level view

SystemC AMS

TDF

HW assist

CPU

Memory

SystemC

DE

DSP bus

adapter

system bus

peripheral bus

Tx/Rx

DSP

Memory

UART

GPIO

TLM model

‘digital’ part

analog mixed-signal part

signal level interface:

‘de2tdf’

‘tdf2de’

anno 2007/8
SoC Architecture interface - option 1
SoC Architecture interface - option 1

signal level interface
d e2 tdf “de2tdf”
tdf2de “tdf2de”
SoC Architecture interface - option 2
SoC Architecture interface - option 2

TLM model

DSP

Memory

SystemC

DE

PV Target Port

Interrupt Port

functional model

register interface

"TDF queues"

Tx/Rx

SystemC AMS

TDF

transaction level interface: ‘tlm2tdf’
‘tdf2tlm’
Conclusions

- Compatibility and interoperability
  - using communication, protocol and payload as proposed by TLM WG
  - using SCML register interface similar to other sub-systems

- Embedding/wrapping of functional models
  - valid for AMS and digital subsystems
  - two approaches possible
    1. Based on standard SystemC FIFO and communication
    2. New register communication in SystemC AMS extensions (e.g. “TDF queues”)

- Evaluation of pros and cons for both options
  - Impact on standardization for both AMS and TLM
  - future AMS/TLM (in)dependency
  - Impact on simulation performance, flexibility, compatibility and interoperability