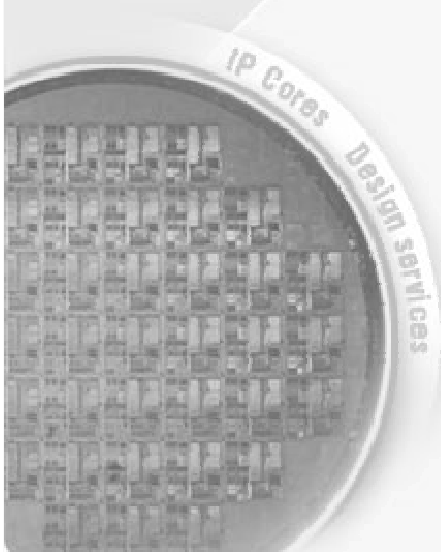




Straightforward IP Integration with IP-XACT RTL-TLM Switching

Demo done within the scope of SPRINT
FP6 European Project using Magillem tool

Ireneusz Sobański



Straightforward IP integration with IP-XACT RTL-TLM switching

- ▶ Platform-independent tests verifying integration of Evatronix USB OTG
- ▶ Abstraction level switch using IP-XACT 1.4 design configuration and interface abstractors
- ▶ Usage of UTMI+ Verification IP controlled by a set of registers mapped to the system bus

IP-XACT benefits from IP provider point of view

- ▶ Standard way of IP packaging and description of multilevel designs
 - Bus definition, bus interfaces
 - definition of standard interfaces at various abstraction levels
 - binding of the component to the bus
 - Memory map
 - consistent description of the register set
 - useful for software development
 - Design and design configuration
 - assembly of RTL-TLM hierarchical designs
 - Component generators
 - configuration, compilation, simulation, checks
 - easy to use
 - simplified definition of legal configuration values
 - File set
 - and others
- ▶ Simplified IP integration
 - IP_XACT description can contain all the information required for automatic assembly of a SoC
 - Manual work reduced to minimum
 - Easy usage of bus monitors and other verification components

RTL/TLM switching

▶ TLM

- TLM emerging as a real standard in ESL modeling
- Usage of third-party TLM require well defined interfaces
 - The need for consistent definition
- Replacement of transactional models by corresponding RTL components during SoC integration
 - Demand for an effortless switching mechanism

▶ IP-XACT 1.4 benefits

- *spirit:abstractionDefinition* allows standard definition of interfaces at different levels of abstraction
- *spirit:designConfiguration* automates switching without any additional work for IP integrator site
 - Insertion of abstractors
 - Selection of desired component view
 - Customization of parameters

Verification software

- ▶ The need for a fast validation of system components during SoC building process (especially difficult when 'unknown' IP components are integrated)
 - Correct work of interfaces (glue logic is transparent)
 - Correct mapping of registers (address decoder functionality)
 - Visibility of interrupts on target system (interrupt controller)
 - Verification of basic functionality of the integrated IP (e.g. USB packets reception)
- ▶ Requirements for such software
 - C based / highly portable
 - applicable various microprocessors models or directly with TLM
 - All input/output functions should be verified
 - If possible, component configuration should be verified
- ▶ IP-XACT 1.4 benefits
 - Registers access tests easily generated from IP-XACT description
 - More functional tests (like DMA transfers or interrupt functions) provided with the IP
 - Packaged in IP-XACT
 - Configured from the IP-XACT
 - Can take advantage of VIPs packaged in IP_XACT and easily connectable to the SoC system (e.g. through the system bus)

Presented example

▶ IP package deliverables

- USB OTG models
 - RTL – real commercial IP
 - TLM – fast simulation model for software development
- Interface abstractors
 - Translates RTL interfaces to TLM levels
 - Inserted automatically
- USB verification IP
 - Fully controlled by system bus interface
- Verification software
 - Platform independent
 - Generation based on the component IP-XACT memory map

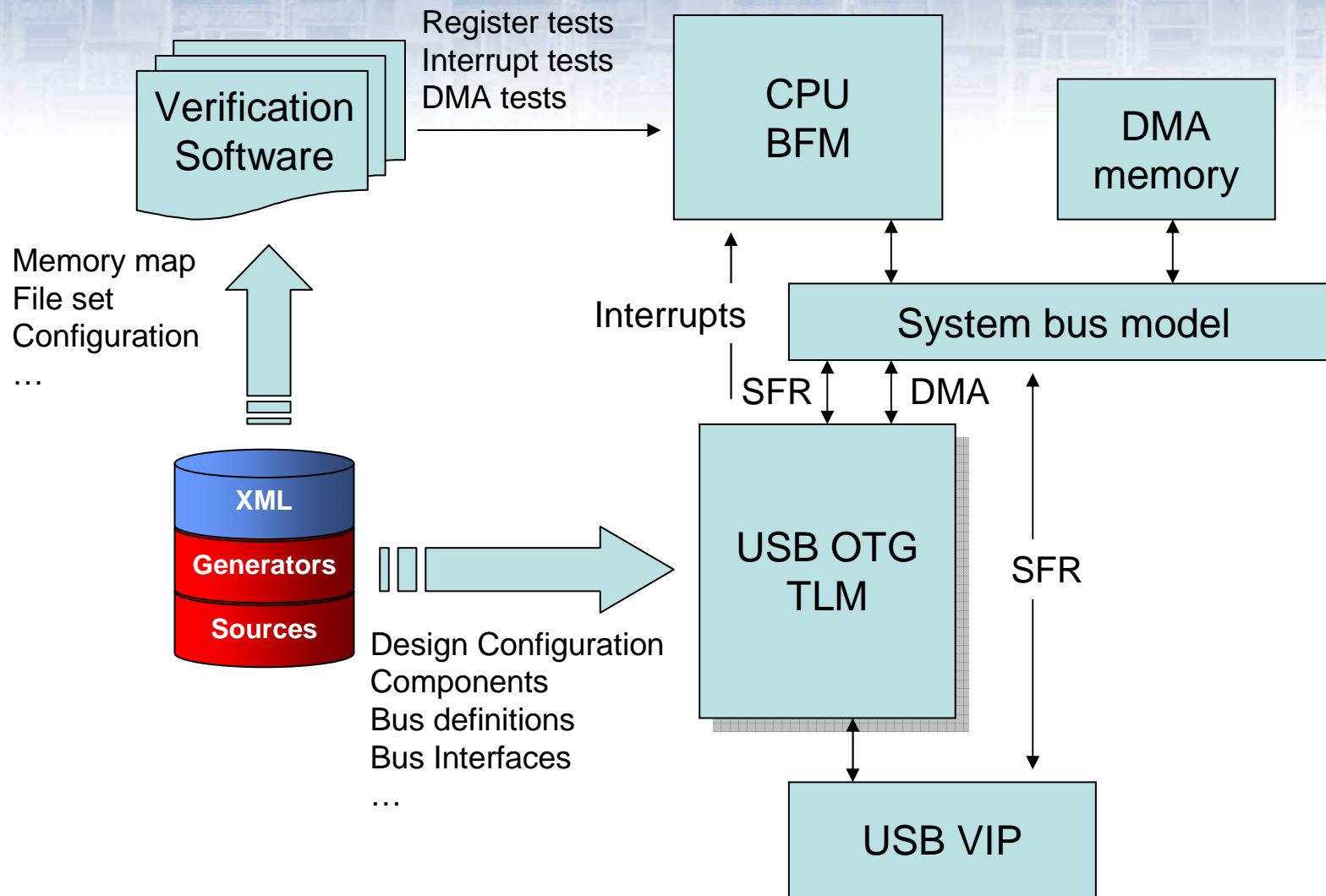
▶ Simulation environment

- DMA memory
- system bus model
- CPU model
 - connects to the system bus
 - runs verification software
 - provides interrupt routines

Presented example stages

- ▶ Instantiation of the USB-OTG component
 - configuration using IP_XACT generator
- ▶ Assembly of the TLM design
 - with an instance of TLM model
 - with an instance of RTL model
 - TLM-RTL switch using 2 views of the TLM model (one design with two design configurations)
 - TLM
 - hierarchical design containing RTL component and abstractors
- ▶ Assembly of the MIX design
 - with an instance of RTL model
 - RTL-TLM binding using abstractors
- ▶ TLM/MIX design generator chain
 - compilation
 - simulation

TLM block diagram



Mix block diagram

