



Session 5.1

Formal Verification of AHB interfaces

Maurizio Spadari
Nemerix



Agenda

- **Company & personal overview**
- **Main project features**
- **AMBA bus features**
- **Verification challenge & strategy**
- **AMBA verification strategy**
- **Benefits of the approach**
- **Conclusion**



Company overview

- **Nemerix (<http://www.nemerix.com>) is a provider of GPS solutions**
 - *Analog RF receivers*
 - *Digital base-band chip*
 - *Processors*
- **Nemerix's core competence is development of low-power integrated circuits**



Personal overview

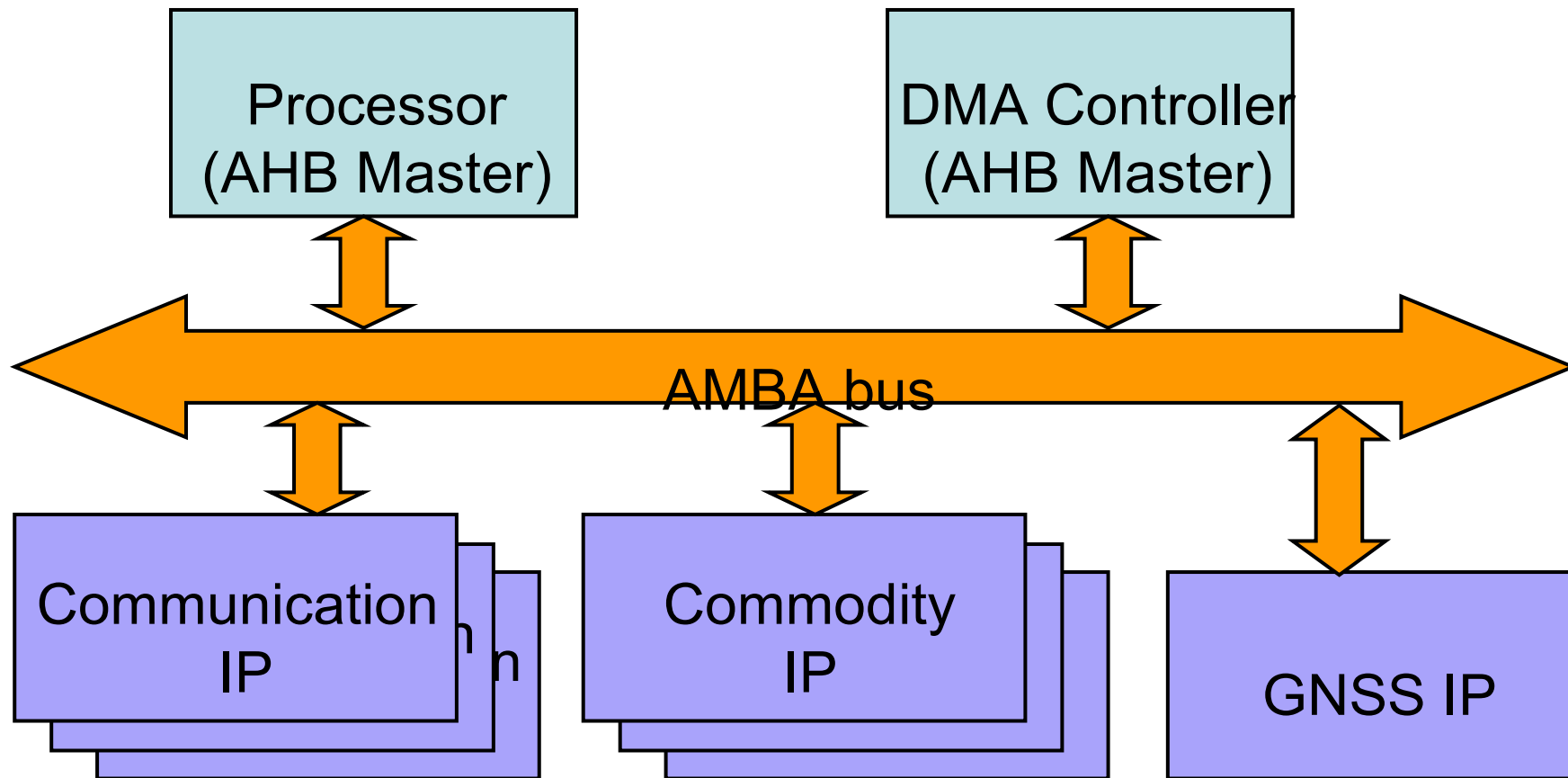
- **10+ year of experience in digital design & verification at various companies (LSI-Logic, ST-Microelectronics)**
- **Currently managing design verification and verification methodologies at Nemerix**



Design overview

- **Next generation base-band processor**
- **Proprietary IP for GNSS signal processing**
- **3rd-party IP processor**
- **3rd-party communication peripherals**
- **Commodity IP developed in-house**
- **Modules connected via AMBA interface**

Base-band processor





AMBA bus structures

- **Multi-master system (processor + DMA controller)**
- **AHB-Lite interface for slave peripherals**
- **Requirement for AMBA bus**
 - *Robustness*
 - *Low power implementation*
 - *No routing overhead*
 - *Reduced toggling activity*



AMBA bus features

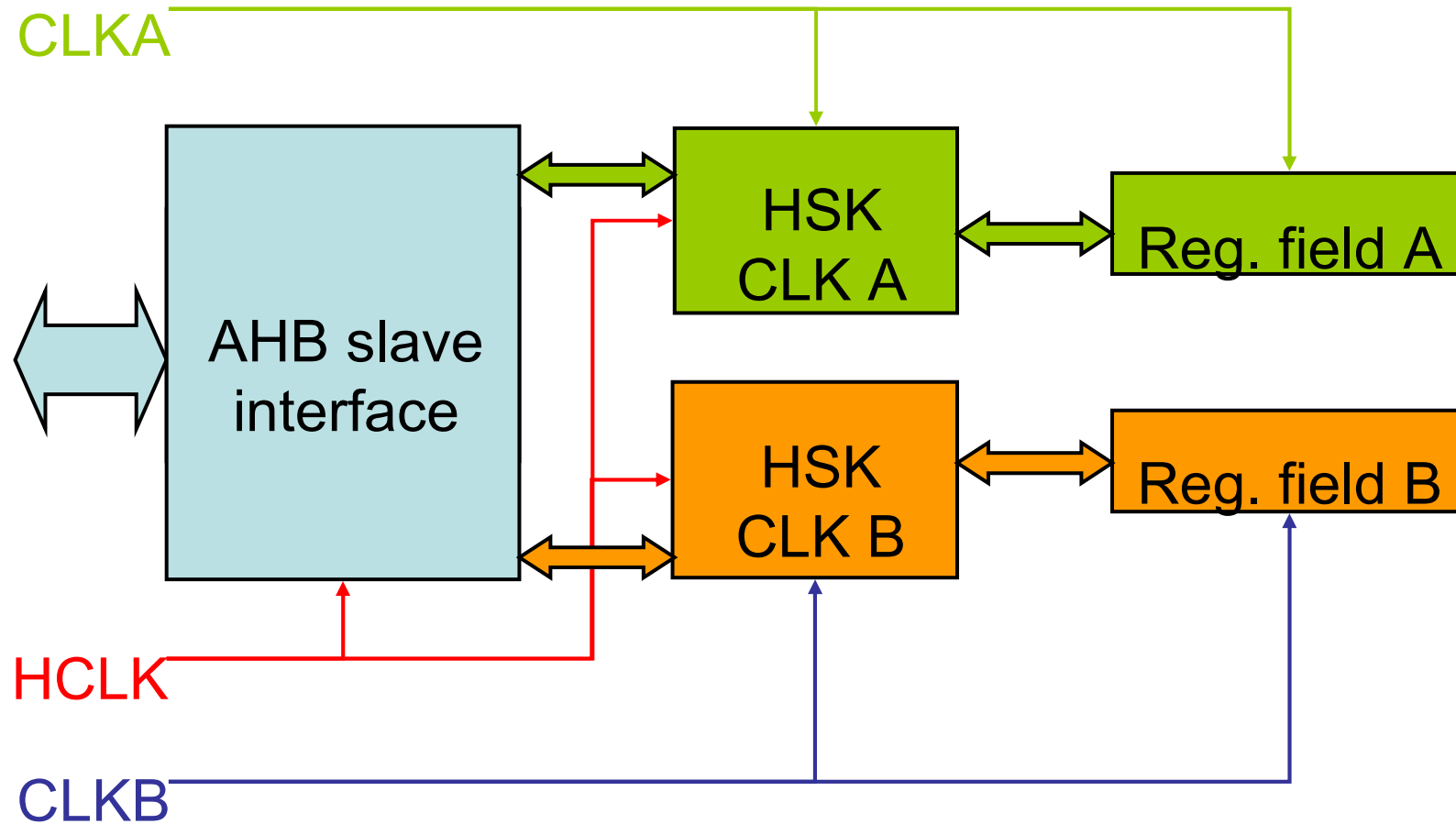
- **Read-Data from slave are ORed together**
 - *Peripherals should keep HRDATA bus low when not addresses*
- **Multi-clock domain system**
 - *Source/Destination register may be in general on a clock domain different than HCLK*
 - *Fields of the same register might be in clock domains different from each other*



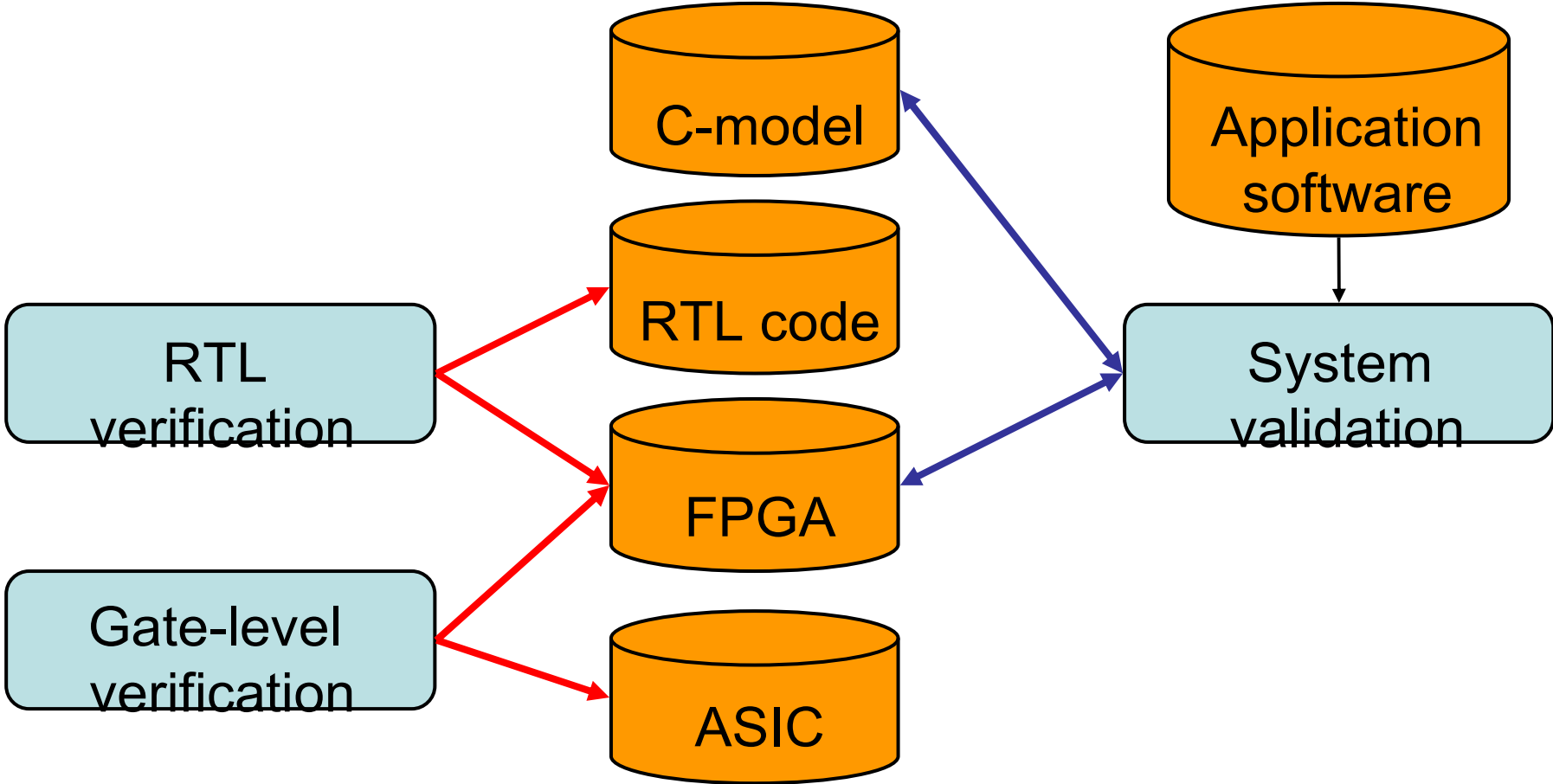
AMBA bus implementation

- **For write/read operation re-timing and hand-shake might be required**
 - *For read operation value is returned on HRDATA only when hand-shake has completed*
 - *For write operation peripherals assert HREADY only when data has been transferred to destination register*
- **Since hand-shake circuitry has a penalty in term of area and timing it should be used only when necessary**
- **For time critical operation where hand-shake is required a read-ahead mechanism is implemented**

Register handshake



Design flow





Verification challenges

- Final “green-light” is given by system team that checks the SW drivers achieve the expected performances on the HW.
- GPS signal processing requires long integration (especially when acquiring “low” signals)
- System must be checked with real-life signals
- An FPGA implementation is mandatory
- Architectural exploration is also made at FPGA level
 - *This implies several architectural modifications also in AMAB bus structure*

AMBA design & verification requirements

- **The above approach demands frequent changes on the AMBA structure:**
 - *Register changes/additional/removal*
 - *Address map changes to improve efficiency in register access*
- **A design methodology is required to quickly implement and verify changes**
 - *Automatic generation of register structure*
 - *Automated verification methodology*



AMBA register generation

- A Python script (**RegGen**) has been devised to generate AHB infrastructure at peripheral level:
 - *AHB-Slave Lite interface*
 - *Hand-shake mechanism for register fields requiring it*
- **Script structure**
 - *GUI interface*
 - *Generation of Verilog HDL code*
 - *Generation of documentation (HTML file)*



RegGen script : user interface

- **Via GUI user can**

- *Define register map*
- *At register level define each field and its relevant clock domain*
- *Place a short field and register description (for documentation purposes)*

- **Deliverables**

- *Documentation : HTML file*
- *Generation of Verilog HDL code*
- *Python script for re-loading session (useful for modifying an existing register configuration)*

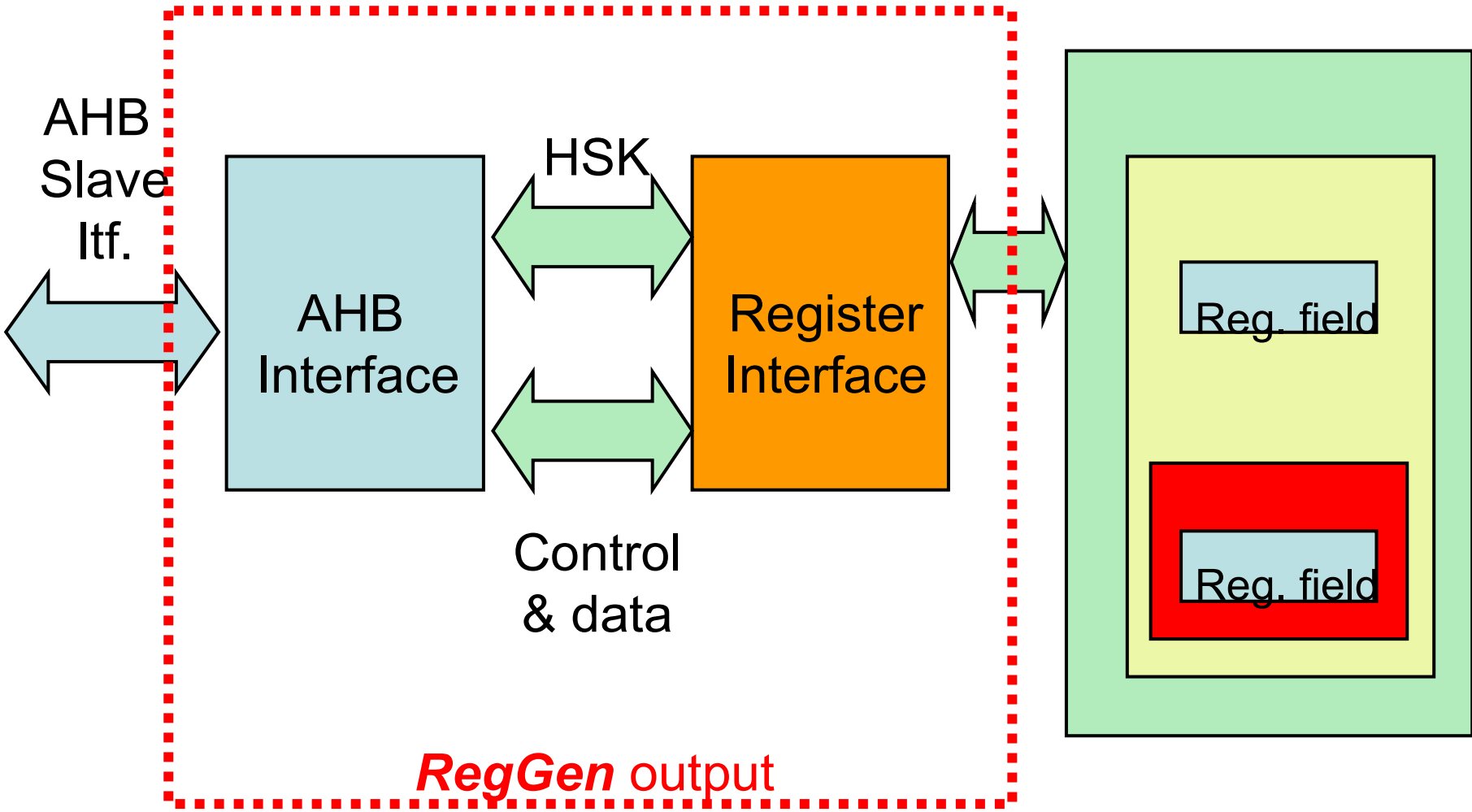
- **RegGen** script is not a pre-defined product but a living script: new feature are added continuously

RegGen script : RTL deliverables

● **RegGen** generates following deliverables:

- *AMBA interface control logic*
 - *Handles AHB interface signal*
 - *Generate the clock hand-shake signals*
- *Register interface*
 - *Handles the register hand-shake signals*
 - *Generate read/write signals to physical register (located in design hierarchy)*

RegGen RTL structure



RegGen script : RTL deliverables

- **Interface control logic block is relatively independent from actual register configuration**
 - *Variable parts of this blocks are number of clock domains and specific block features like read-ahead buffer*
- **Register interface block is very implementation specific since it handles hand-shake signals for bit fields of registers being instantiated**



Formal verification on AMBA

- **Formal verification (FV) methodology is well suited to validate current AMBA structure.**
- **FV explores all possible scenarios and interactions between AMBA operations.**
- **AMBA VIP allows to check that slave responses are consistent with AMBA.**
- **VIP checks are supplemented with additional checks that ensure that project specific features (hand-shake, look-ahead buffer) operate correctly.**



Modular approach

- **RegGen** script allows a modular approach to validation.
- **out-of-context verification**
 - *When a new feature or a modification is done on script, script itself is validated by generating a simple but exhaustive test-case*
- **in-context verification**
 - *When register file is modified at module-level the new module is validated via IFV*



Out-of-context verification

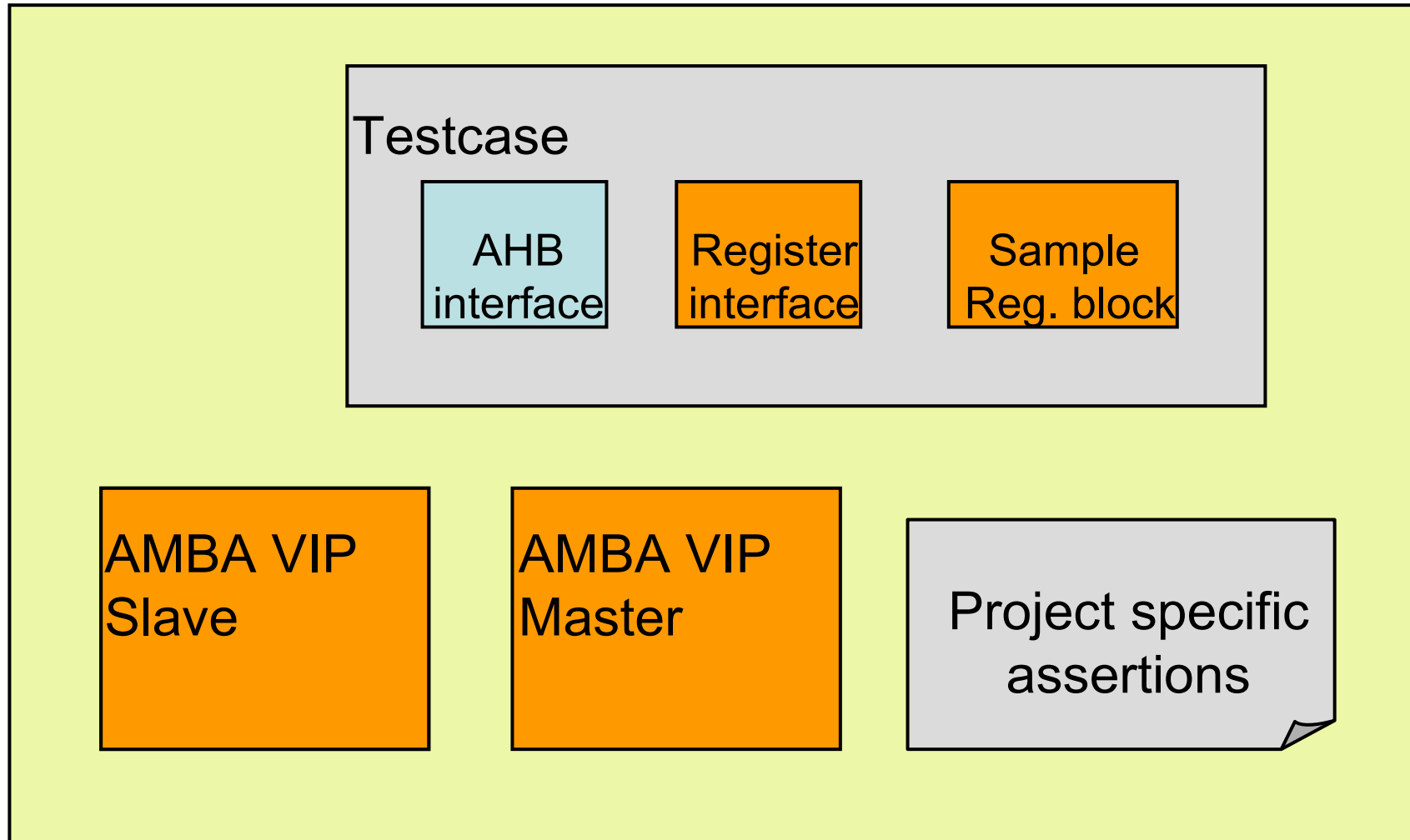
- Purpose of **out-of-context** verification is to ensure all supported features operate correctly
- Test-case used for validation should be exhaustive in the sense that contains all features supported by the script
- Verification is done with a combination of AMBA VIP together with additional checks
 - *AMBA VIP ensures that module is compliant to AMBA specification*
 - *Additional checks ensure compliancy to project-specific requirements*



Project specific requirements

- Project specific requirements are captured in a verification plan and address project-specific features of the AMBA bus
- Example of project-specific features:
 - **Valid address**: An access to an unmapped area (inside the slave address space) should return an error response on HRESP bus
 - **Register write** : A write operation should update register content.
 - **Register read** : A read operation should return register content.
 - **Hand-shake** : In a read/write operation the correct hand-shake signals should activate

Out-of-context test-bench



Advantage of out-of-context verification

- **A simple test-case allows faster turn-around time**
- **Limited number of assertions required to validate design**
 - *Easy to add specific assertions to debug/analyze in-depth the FSM and control block parts*
- **Due to module structure out-of-context verification will stress mostly the control part which is less implementation specific**



In-context validation

- **In-context validation is applied to the AHB peripheral**
- **It ensures that register file structure is consistent with specification**
 - *Unused areas are correctly defined*
 - *Address map is correct : a write access places correct data in the physical register*
 - *Each register is mapped to correct clock domain*

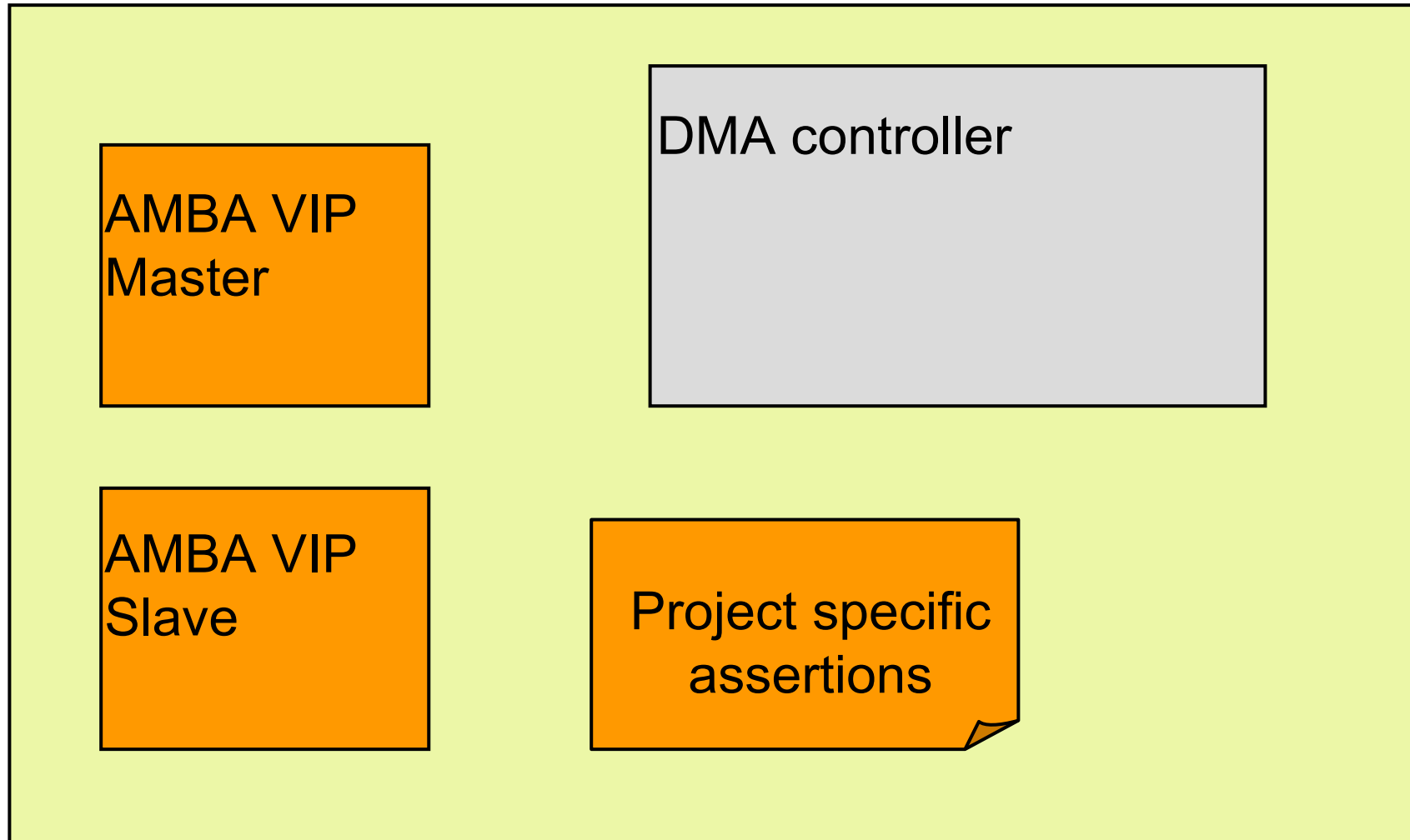
In-context test-bench: case 1

Slave peripheral

AMBA VIP
Slave

Project specific
assertions

In-context test-bench case 2

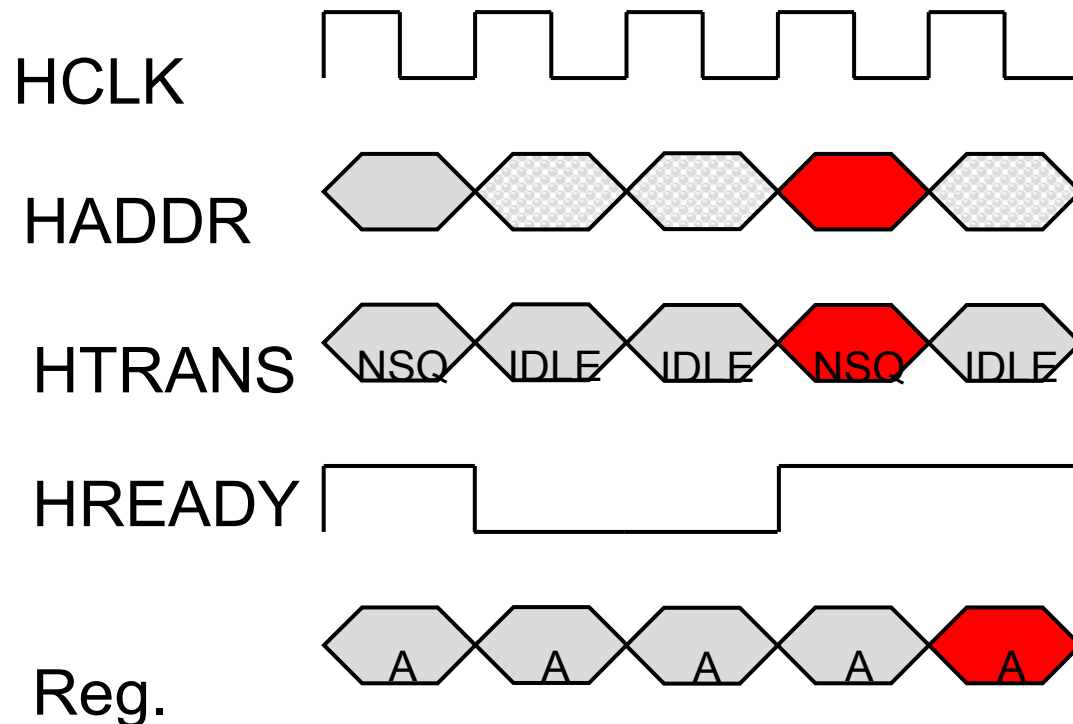




Advantage of in-context validation

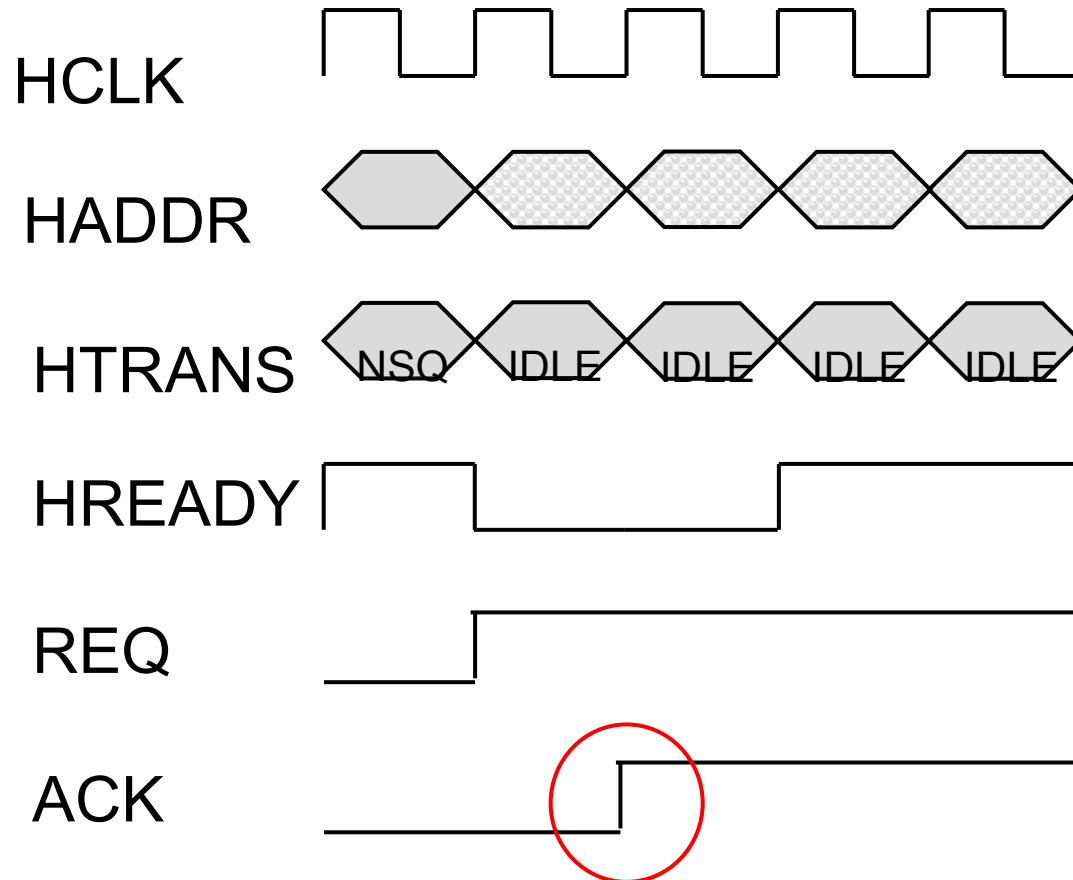
- **Most of control logic already validated in **out-of-context** verification**
 - *Less likelihood to find a bug there*
- **Larger number of assertion required but most of them are similar and can be generated via a Perl script**
- **Using the regression script itself to generate assertions is not recommended since an error on input files will be reflected in both HDL and assertions**

Example of bugs found (1)



- Bug occurred on access marked in red
- A register access is done on the same cycle hand-shake completes and HREADY is released
- AMBA protocol is not violated however register is not updated

Example of bugs found (2)



- **HREADY** is released when handshake completes, i.e. when acknowledge is asserted
- If acknowledge does not occur AMBA bus is held indefinitely
- Need to check that this never occurs

- **Modular approach reduces time spent on AMBA validation without compromising on accuracy and completeness**
- **RegGen script is fully reusable for future projects**
- **In previous projects SpecMan had been used to validate AMBA bus**
 - *Once the test has been put in place several investigation has been done to ensure that all possible corner cases are covered: no guarantee that this goal is reached*
 - *IFV approach is opposite: we check property on all possible cases and then we restrict search space by ruling out false cases. Faster method and more comprehensive.*



Conclusions

- **VIP approach is a time-saving approach**
 - *It allows to quickly verify standard bus & interfaces*
- **A re-usable methodology is essential to capitalize effort spent on assertion generation**