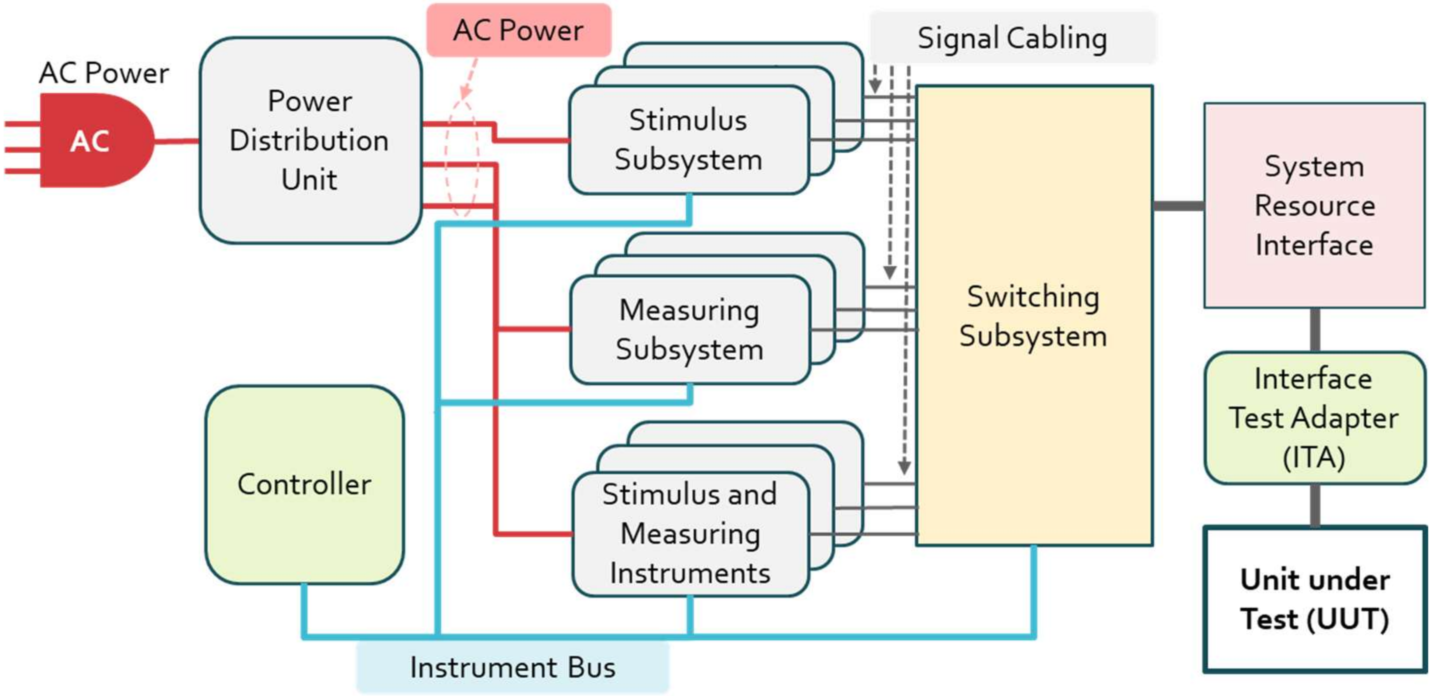
FPGA-Centric Design Concepts for Advanced Test and Measurement Systems

AmiES 2025

Peter Schulz

Context ATE

Automatic Test Equipment



Content

Motivation I

Synthetic Instruments

- Measurement Instruments
    - may become obsolete!
    - are expensive!
    - have limited functionality!

- Synthetic Instruments
    - are versatile computing platforms combined with measurement signals
    - can be configured to replace a class of measurement instruments
    - can mimic original measurement instruments' behavior
    - may be low-cost

Motivation II

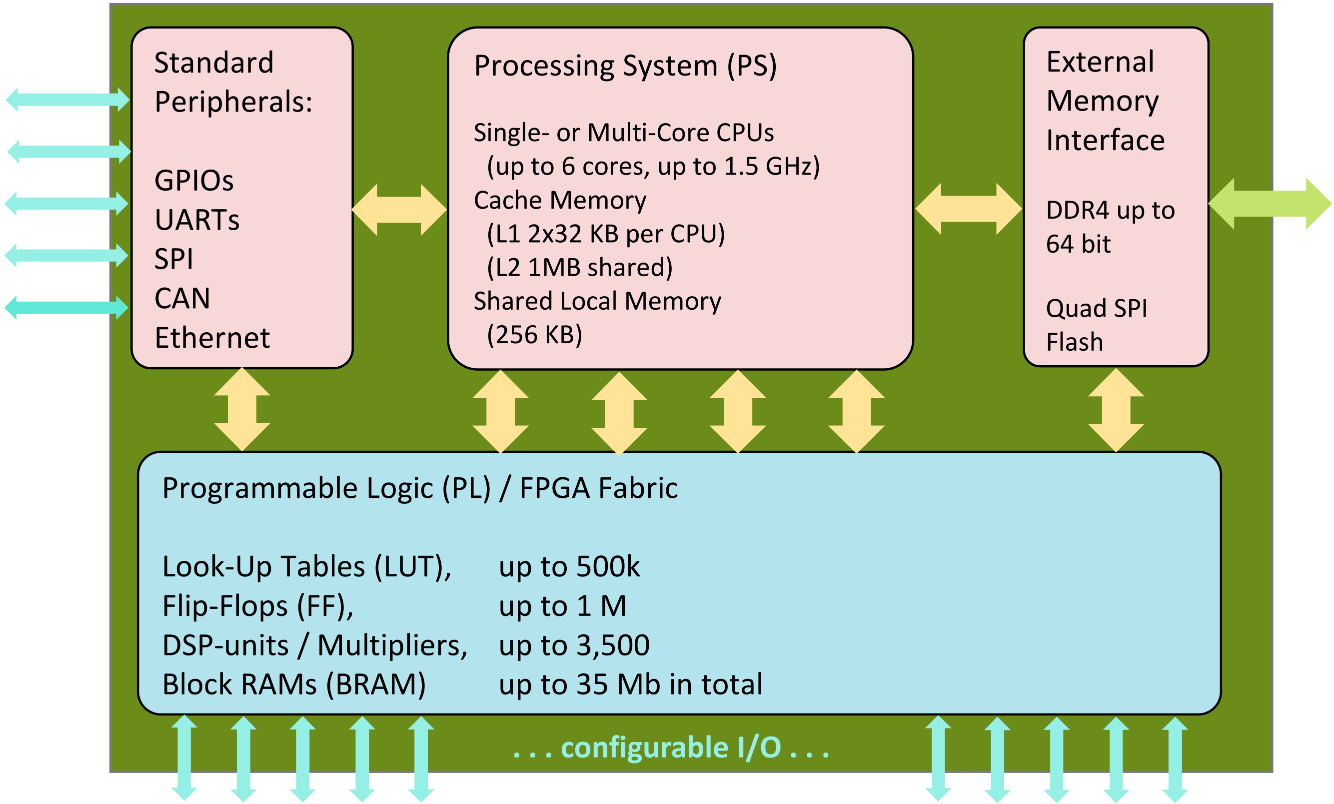
Why FPGAs?

- Why use Field Programmable Gate Arrays (FPGAs) rather than software?
  - Realtime
  - Concurrency
- System on Chip FPGA devices combine processing system (PS) with programmable logic (PL)
  - capable of running SW and HW on one platform.
- Reconfigurability / Dynamic Function Exchange (DFX)

Implementation Approach:

SoPC

- System on Programmable Chip (SoPC)
  - the right platform for execution of logic circuits, DSP algorithms and software on a single chip (plus external memory, if needed)



Standard Peripherals:

GPIOs
UARTs
SPI
CAN
Ethernet

Processing System (PS)

Single- or Multi-Core CPUs
(up to 6 cores, up to 1.5 GHz)
Cache Memory
(L1 2x32 KB per CPU)
(L2 1MB shared)
Shared Local Memory
(256 KB)

External Memory Interface

DDR4 up to 64 bit

Quad SPI Flash

Programmable Logic (PL) / FPGA Fabric

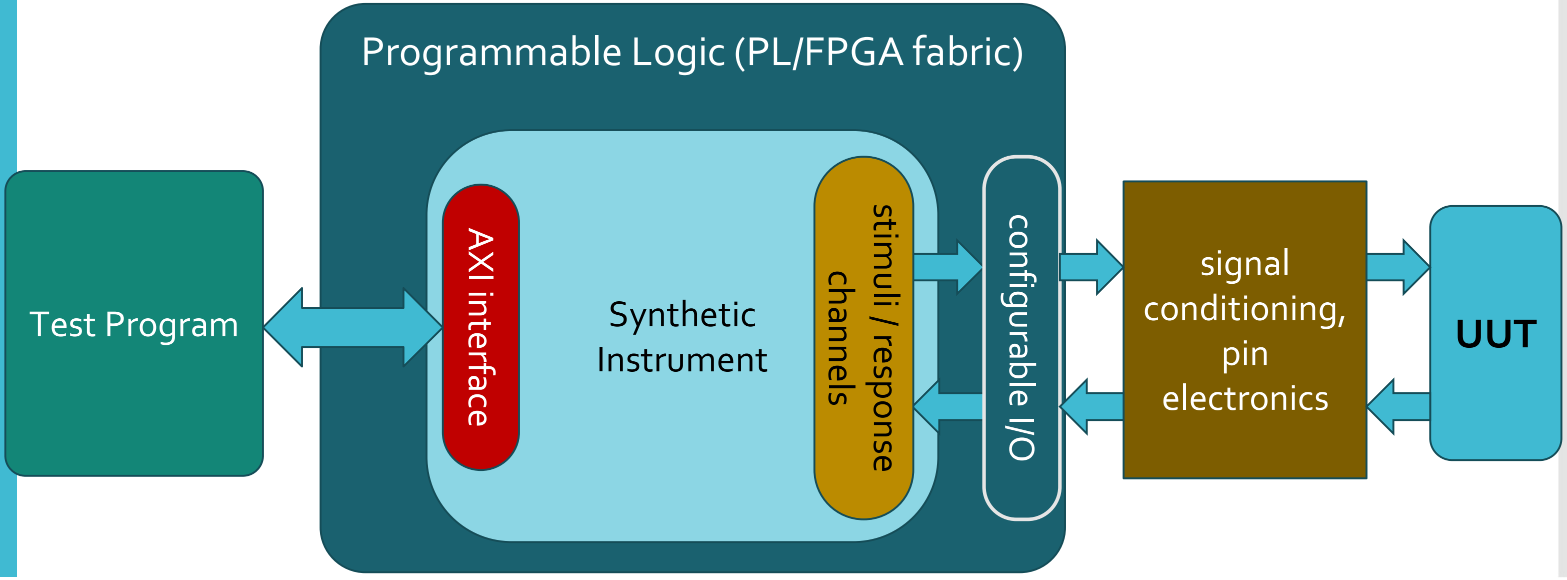
Look-Up Tables (LUT),	up to 500k
Flip-Flops (FF),	up to 1 M
DSP-units / Multipliers,	up to 3,500
Block RAMs (BRAM)	up to 35 Mb in total

. . . configurable I/O . . .

FPGA-based Synthetic Instruments

FPGA-based Synthetic Instrument

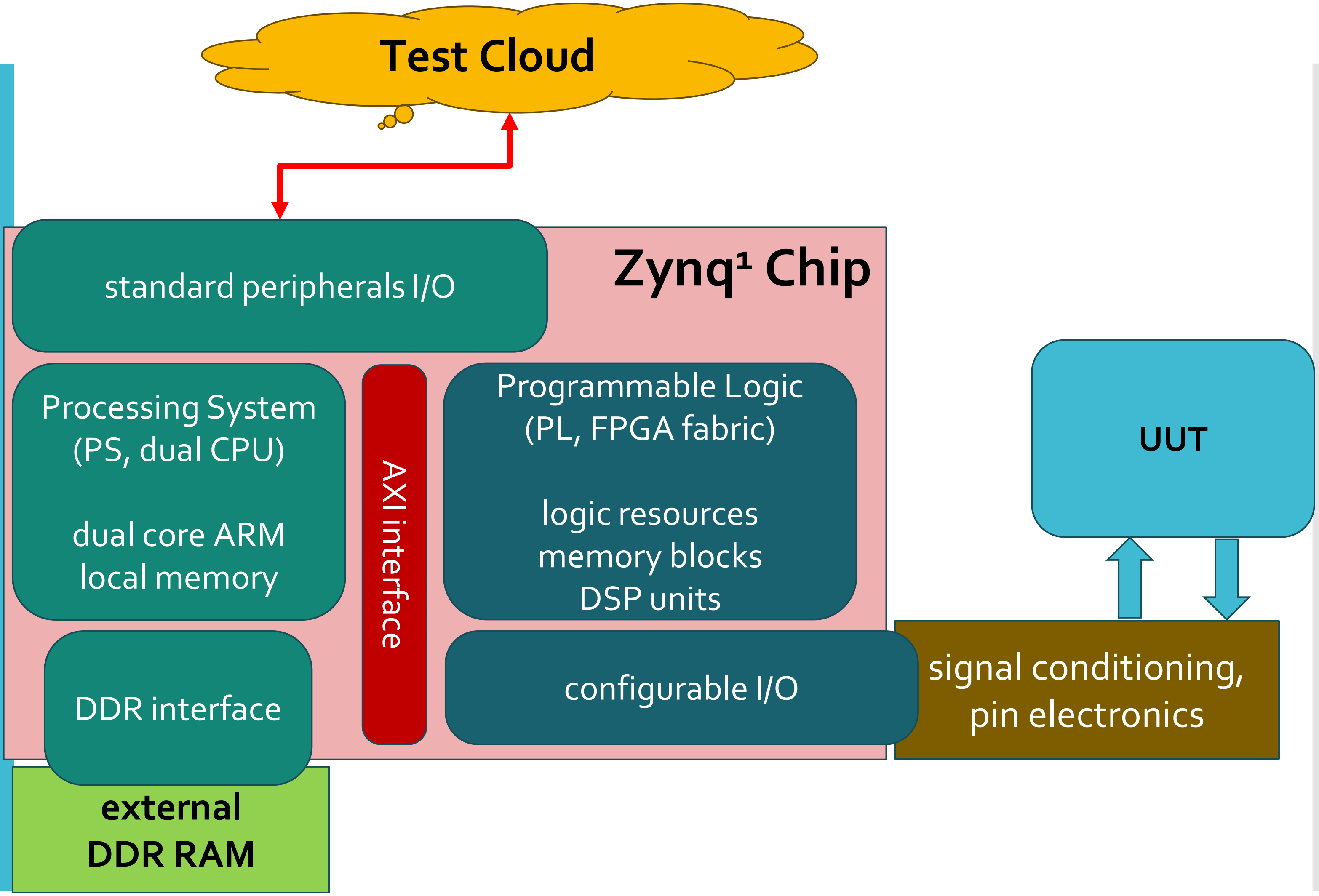
- FPGA-implemented VHDL or SystemVerilog model
- provides a register interface for control/status and data exchange (AXI)
- requires external physical layer for UUT-access (ITA, pin electronics)
- more signals possible if required (Trigger, Synchronization, etc.)



CPTS

Low-Cost Test System with Cloud Connectivity

Zynq[1]-based Test System in CPTS



Test Cloud

Zynq[1] Chip

standard peripherals I/O

Processing System
(PS, dual CPU)

dual core ARM
local memory

AXI interface

Programmable Logic
(PL, FPGA fabric)

logic resources
memory blocks
DSP units

DDR interface

configurable I/O

external DDR RAM

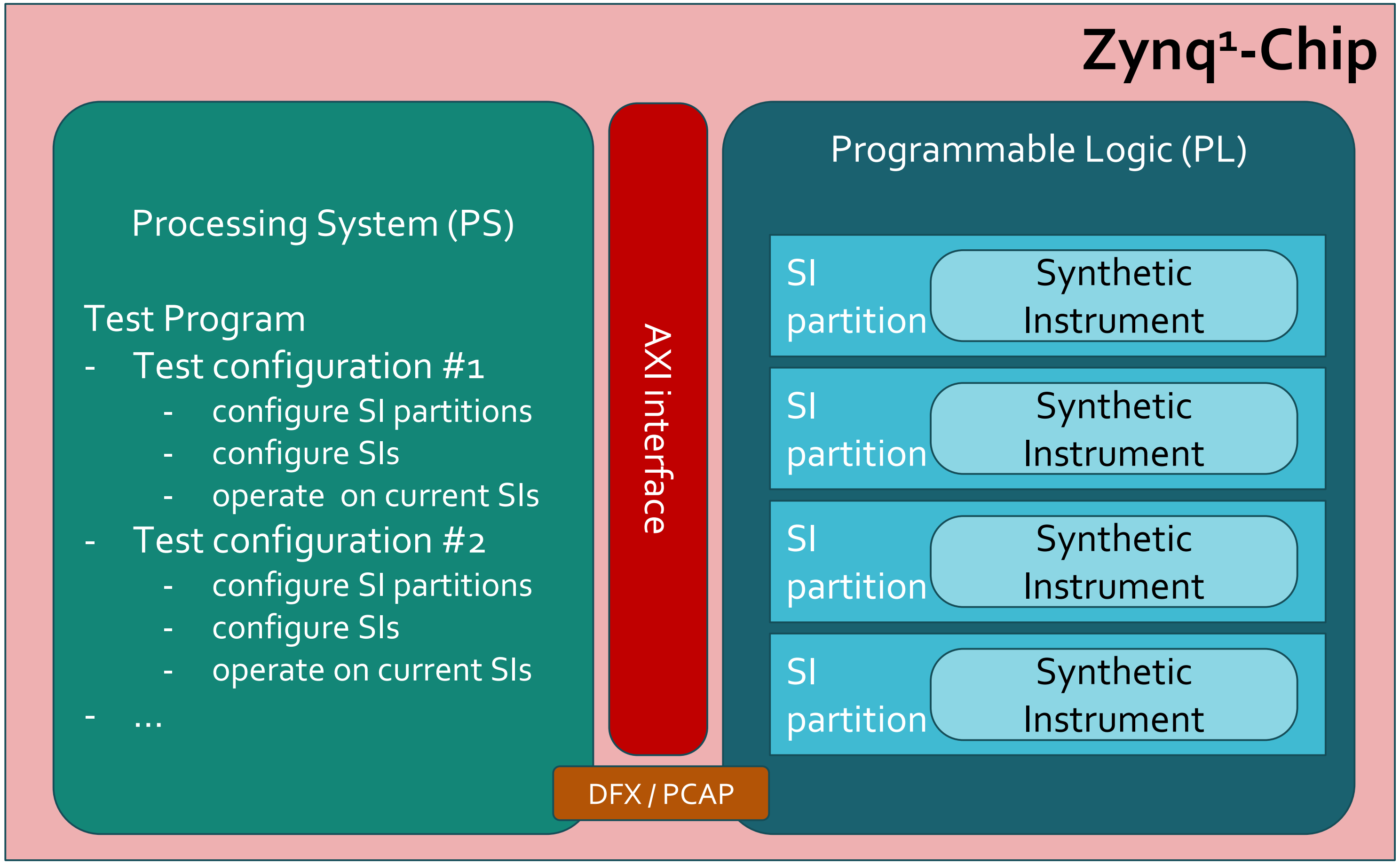
signal conditioning,
pin electronics

UUT

---

[1] Zynq is a trademark of AMD (Xilinx)

Allocation of Tasks in a Zynq-based Test System



Zynq[1]-Chip

Processing System (PS)

Test Program
- Test configuration #1
  - configure SI partitions
  - configure SIs
  - operate on current SIs
- Test configuration #2
  - configure SI partitions
  - configure SIs
  - operate on current SIs
- …

AXI interface

DFX / PCAP

Programmable Logic (PL)

SI partition — Synthetic Instrument

SI partition — Synthetic Instrument

SI partition — Synthetic Instrument

SI partition — Synthetic Instrument

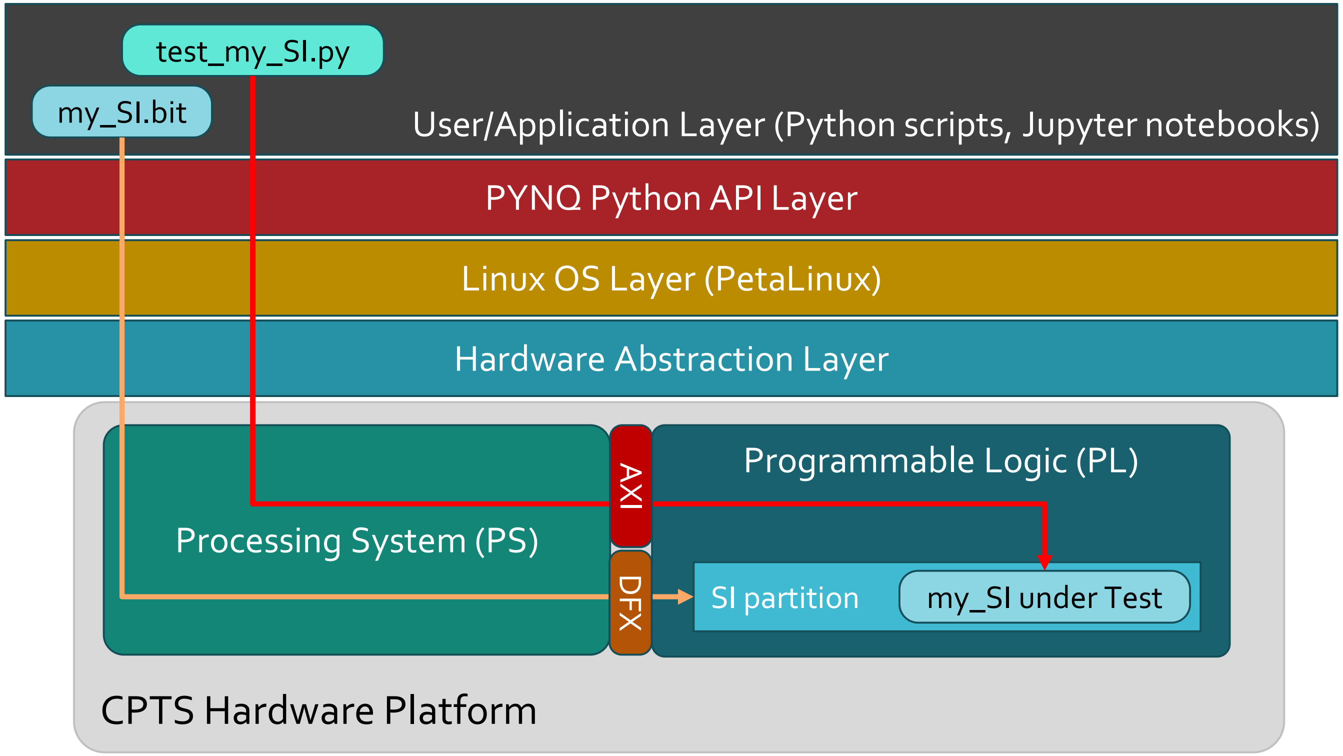
---

[1] Zynq is a trademark of AMD (Xilinx)

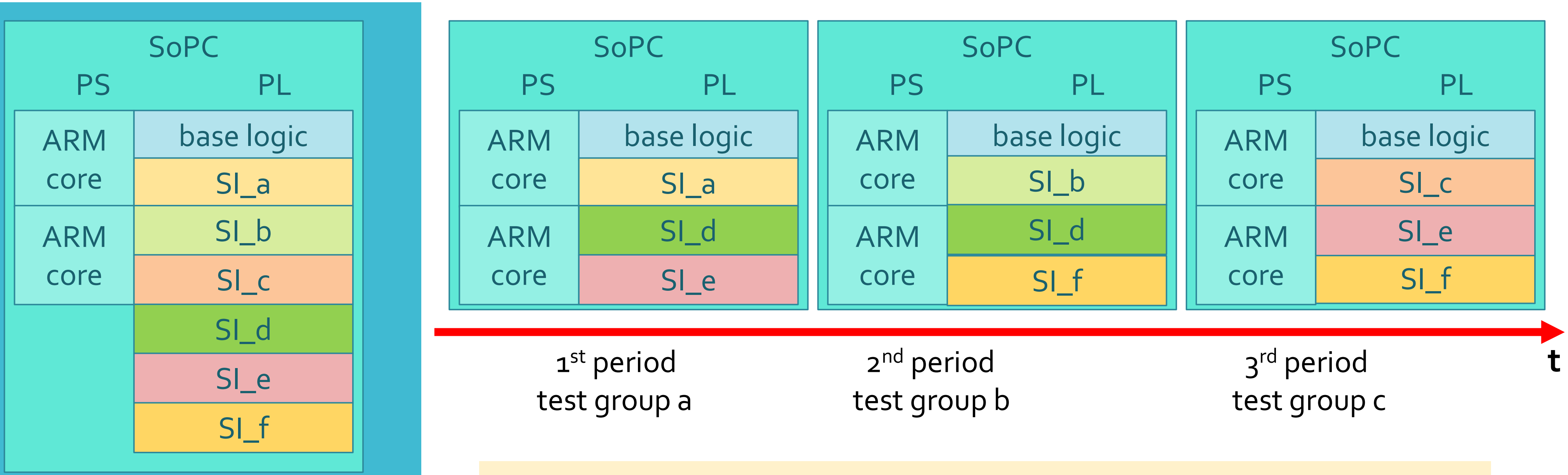
SI and *DFX* and *PYNQ*

- DFX: Runtime Hardware Reconfiguration
  - Selective reconfiguration of FPGA Regions
  - Static + Reconfigurable Partitions (RPs)
  - Multiple Reconfigurable Modules (RMs)
  - DFX features
    - No system shutdown required.
    - Results in Runtime functionality switching
- Partial Reconfiguration Support with PYNQ
  - PYNQ: "Python Productivity for Zynq"
  - Rapid Prototyping: Python APIs for DFX integrated with a Jupyter notebook
  - Increased Focus on Pipeline/Application development: Overlay-based HW abstraction for RM Management.
  - Max System uptime : Runtime Bitstream Loading
  - Overlay concept eases handling of Linux device driver for FPGA support.

SI management via PYNQ



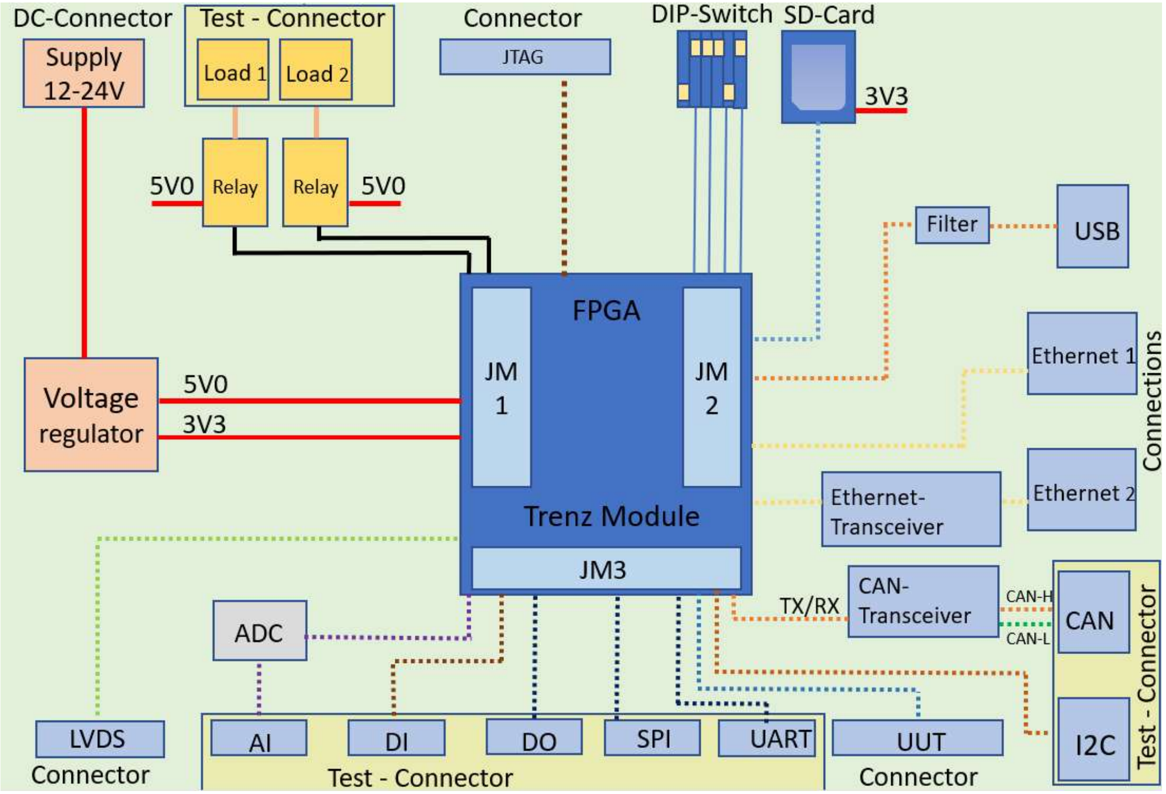
Hardware Multitasking

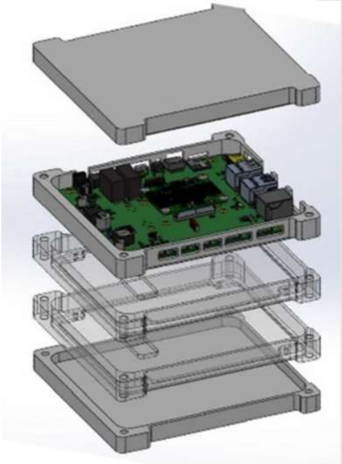


- <u>on the left:</u> static PL configuration with all synthetic instruments loaded permanently into a *big* PL
- <u>on the right:</u> *smaller* PL, **partly reconfigured (DFX)**
  - Intermediate reconfigurations from test group to test group

CPTS Hardware Development

Main Unit

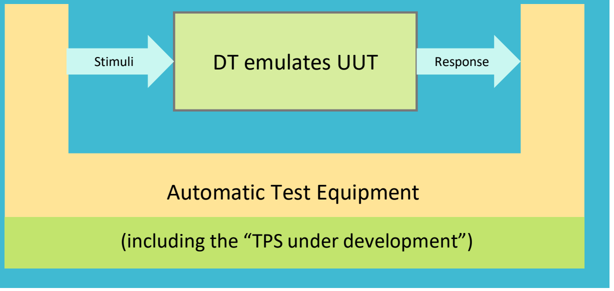




UUT-DT

mimicking a Unit under Test

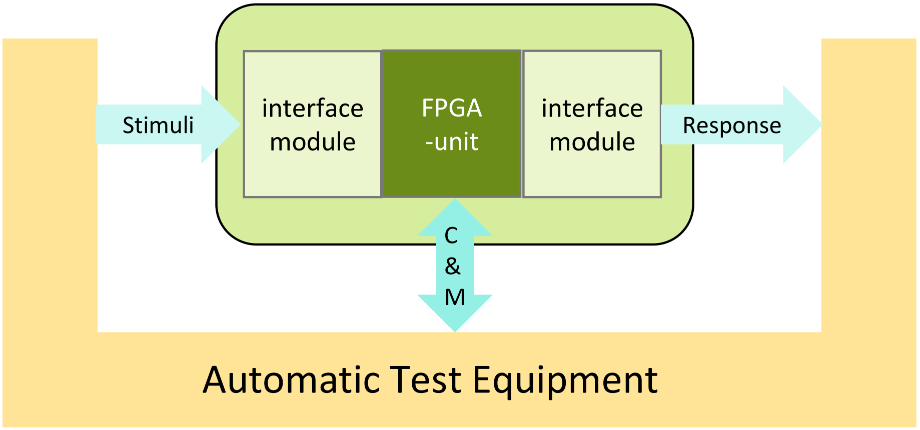
Motivation



- An Automatic Test System (ATS) shall find nearly all detectable faults of a unit under test (UUT) by means of an end-to-end test.
- The ATS should also support diagnosing to the root cause by fault isolation.

- A Digital Twin (DT) that simulates a UUT can support development of the Test Program Set (TPS).
  - Software implemented DTs are usual today but lack real-time behavior for UUTs of higher complexity.
  - Field Programmable Gate Arrays (FPGAs) offer a reasonable approach to implement "real-time-DTs".
  - Both approaches are reconfigurable and therefore can easily provide fault injection for TPS verification.

FPGA-based DT concept

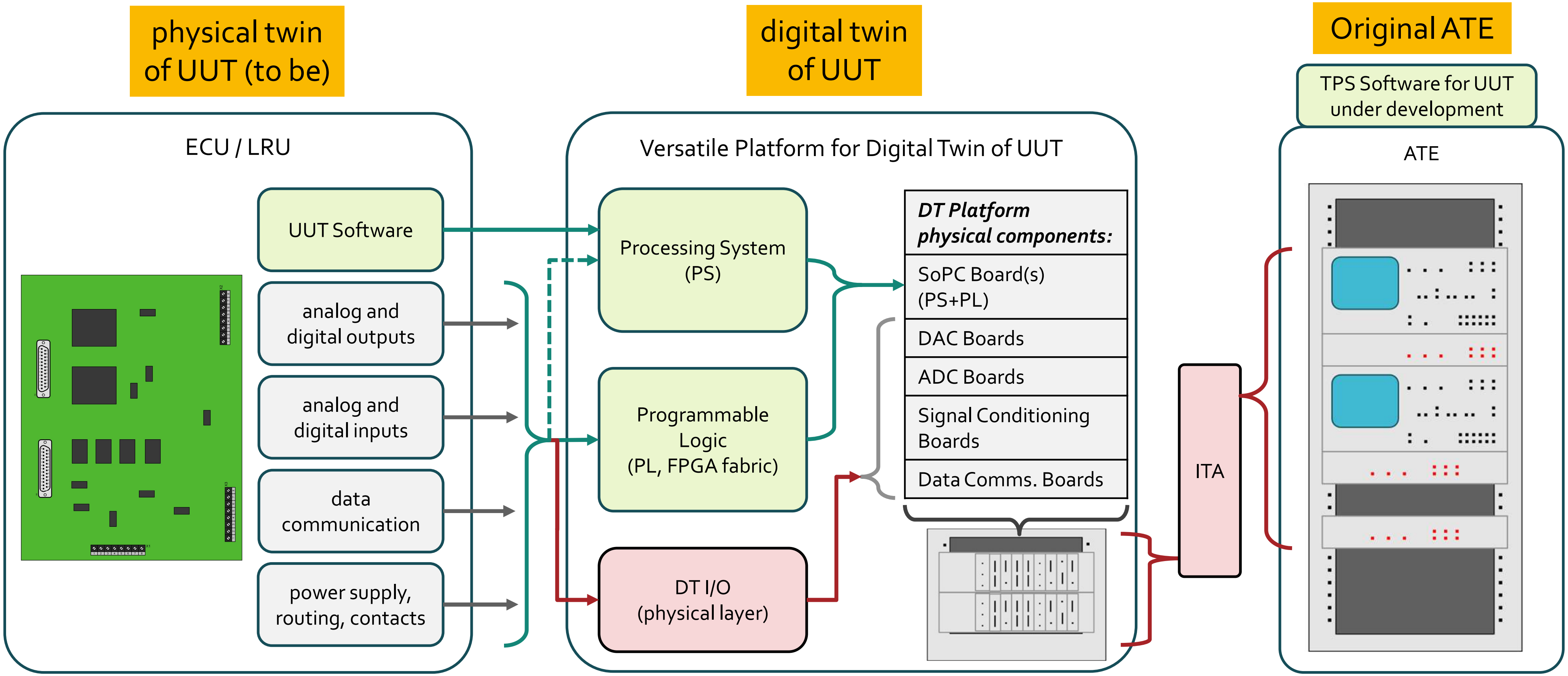


- UUTs are typically not purely digital assemblies. Therefore, the FPGA unit must be complemented with specific interface modules.
- When the ATE loads a new test suite, the DT may also need to be reconfigured via a Configuration & Management Interface (C&M).

Technical Approach

UUT mimicking

UUT functional categories	**How to mimic by FPGA-based DT?**
passive and analog electronic circuits	ADC and DAC interfaces as well as digitally emulated analog circuits using DSP principles (FPGA or software)
digital circuits	FPGA directly executes the same logical behavior
embedded software	a processor that executes a compatible software



physical twin of UUT (to be)

ECU / LRU

UUT Software

analog and digital outputs

analog and digital inputs

data communication

power supply, routing, contacts

digital twin of UUT

Versatile Platform for Digital Twin of UUT

Processing System (PS)

Programmable Logic (PL, FPGA fabric)

DT I/O (physical layer)

DT Platform physical components:

SoPC Board(s) (PS+PL)

DAC Boards

ADC Boards

Signal Conditioning Boards

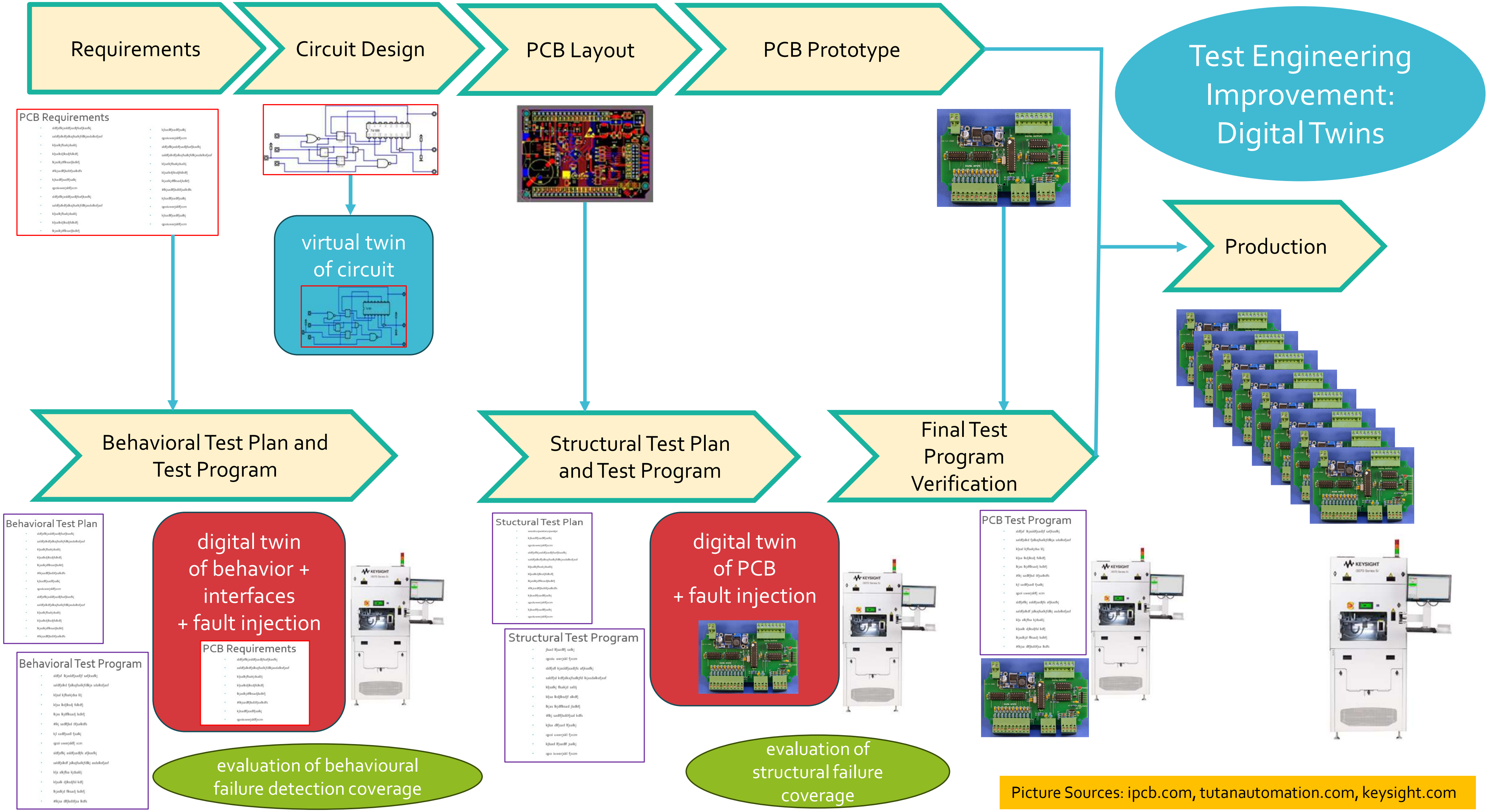
Data Comms. Boards

ITA

Original ATE

TPS Software for UUT under development

ATE



Requirements

Circuit Design

PCB Layout

PCB Prototype

Test Engineering Improvement: Digital Twins

PCB Requirements

virtual twin of circuit

Production

Behavioral Test Plan and Test Program

Structural Test Plan and Test Program

Final Test Program Verification

Behavioral Test Plan

Behavioral Test Program

digital twin of behavior + interfaces + fault injection

PCB Requirements

evaluation of behavioural failure detection coverage

Stuctural Test Plan

Structural Test Program

digital twin of PCB + fault injection

evaluation of structural failure coverage

PCB Test Program

Picture Sources: ipcb.com, tutanautomation.com, keysight.com

ATE-DT

A Digital Twin of an ATE

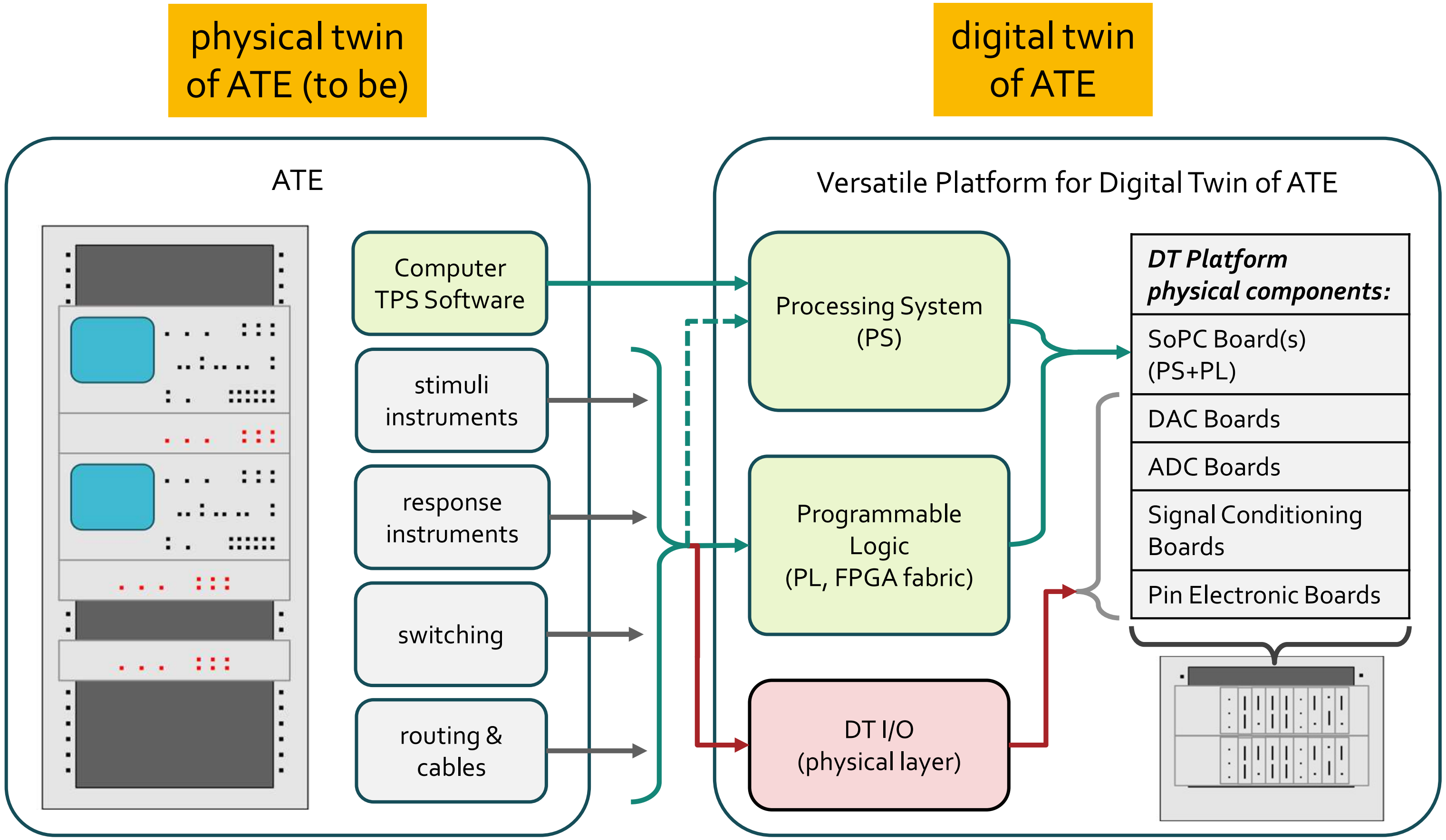
Motivation

- A starting point for evaluating the requirements for (a yet to be designed) ATE
  - Example: Replacing a legacy ATE with a new one that can run all TPSs already developed
- A tool for developing and debugging TPS
- ATE predictive maintenance
  - Alerts need for calibration
  - An indicator for remaining useful life of ATE
- A tool to find root causes in some instances of NFF

DT architecture

- Goal: Create a template which can be used for any ATE to be twinned.
  - A combination from SW and FPGA Hardware
  - customizable through different subsystems
- Subsystems DTs
  - Instruments DTs (including model for calibration need)
    - Measurement Instruments
    - Stimulus Instruments
    - Stimulus & Measurement Instruments
  - Switching DT (including model of aging and wear out)
  - Routing and cabling (including delays, bandwidth, attenuation)
- The same considerations as for the UUT-DT lead to the realization that FPGAs can form a suitable implementation platform.
  - Can we implement a complete ATE DT using a single FPGA?

Physical Twin and Digital Twin of an ATE transformation



physical twin of ATE (to be)

digital twin of ATE

ATE

Computer TPS Software

stimuli instruments

response instruments

switching

routing & cables

Versatile Platform for Digital Twin of ATE

Processing System (PS)

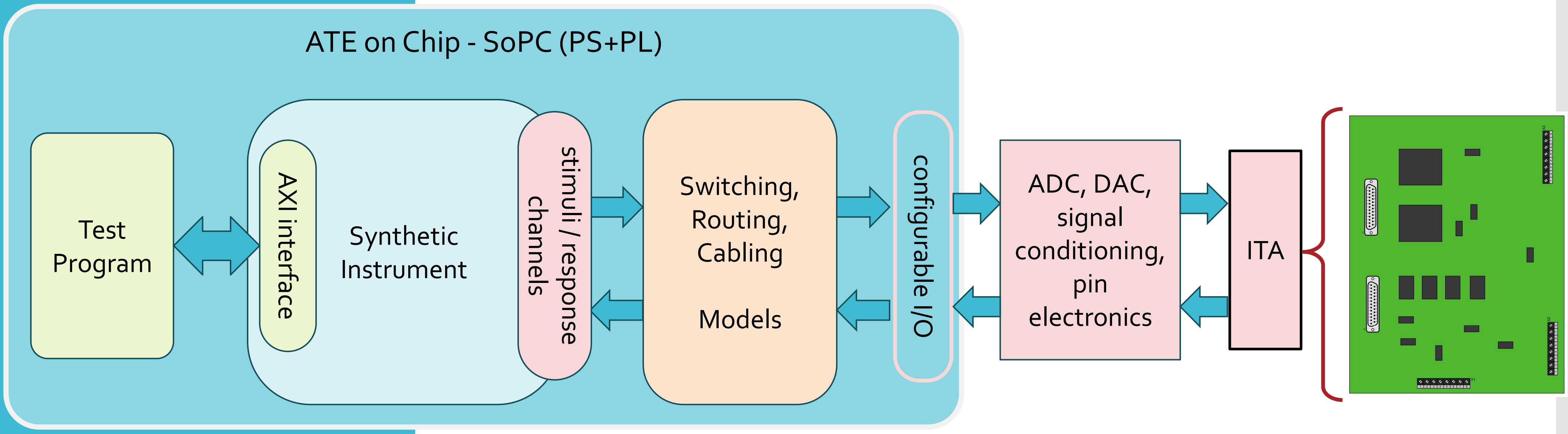
Programmable Logic (PL, FPGA fabric)

DT I/O (physical layer)

DT Platform physical components:

- SoPC Board(s) (PS+PL)
- DAC Boards
- ADC Boards
- Signal Conditioning Boards
- Pin Electronic Boards

ATE-on-Chip



ATE on Chip - SoPC (PS+PL)

Test Program

AXI interface

Synthetic Instrument

stimuli / response channels

Switching, Routing, Cabling

Models

configurable I/O

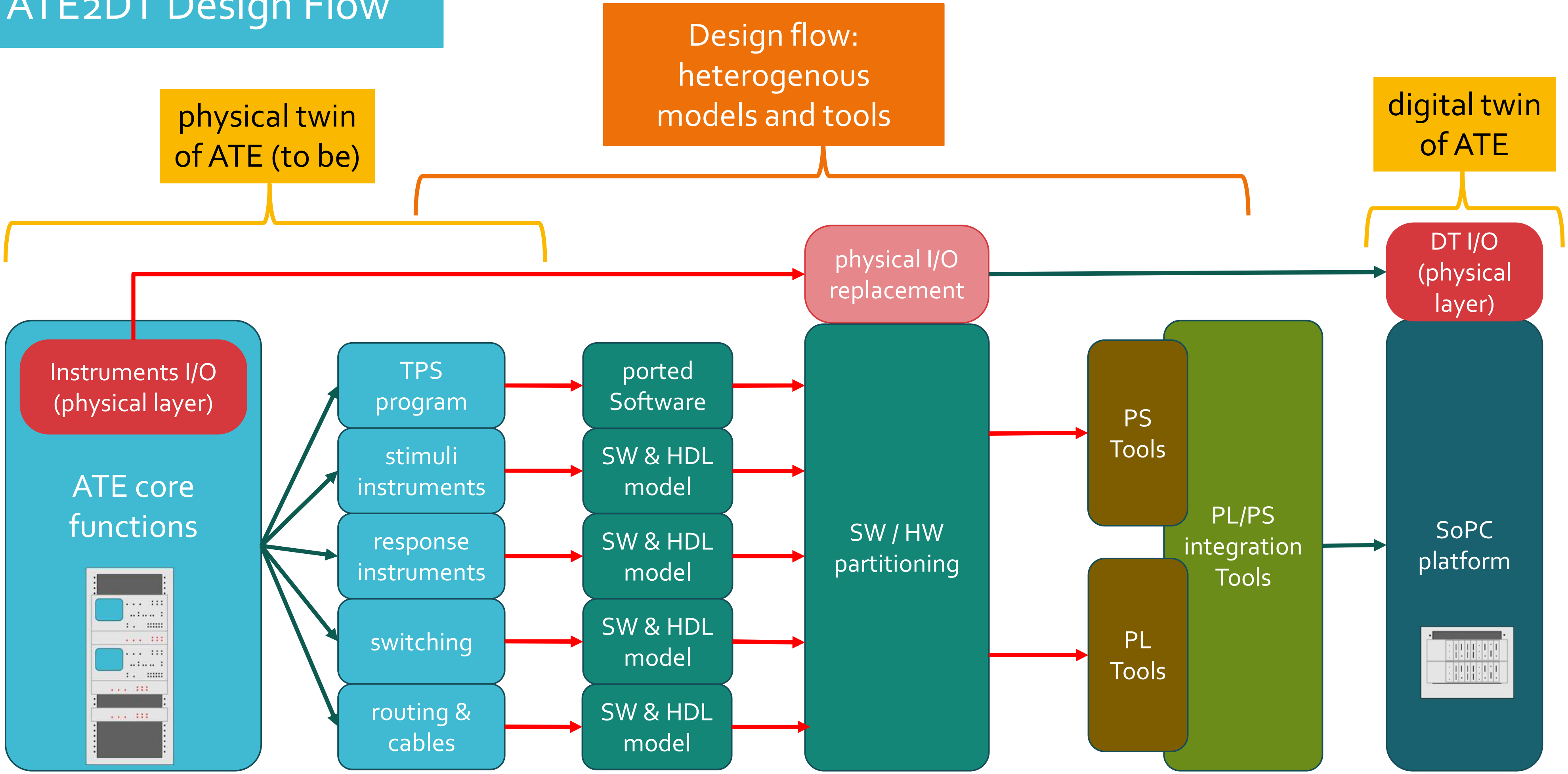
ADC, DAC, signal conditioning, pin electronics

ITA

Outlook

Digital Twin Design Flow is a Challenge!

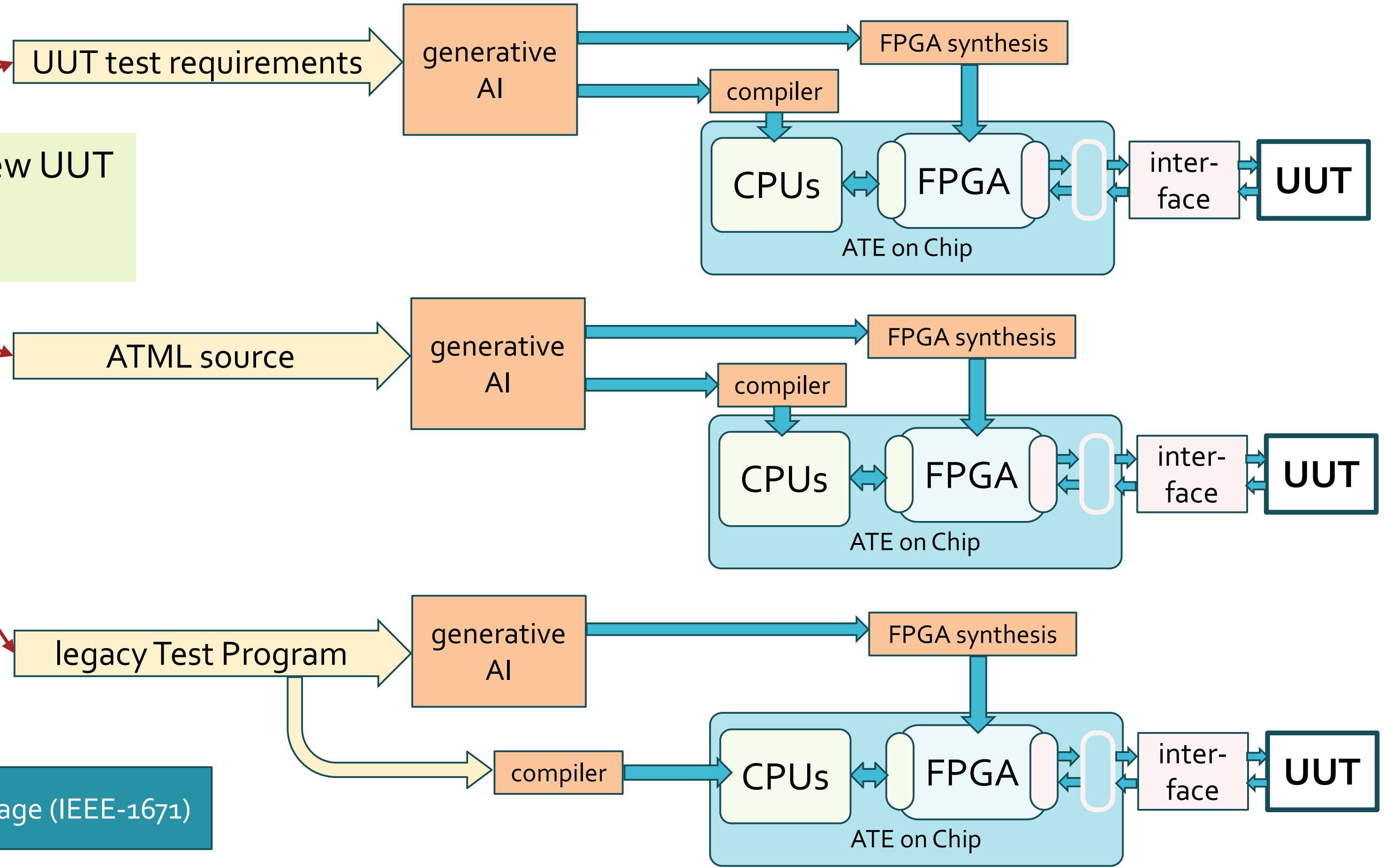
ATE2DT Design Flow



ATE:	Automatic Test Equipment	HDL:	Hardware Description Language	PL:	Programmable Logic (= FPGA fabric)
PS:	Processing System	SoPC:	System on Programmable Chip (= PS+PL)		

ITEA 2025 consortium of **ai4idea2metal**
- use case: “ATE on Chip”

use case scenarios



Scenario 1: Test design for new UUT

- 1.a: from requirements list
- 1.b: from ATML source

Scenario 2: legacy ATE replacement for existing UUTs

- keeps investment in Test Programs
- ATE on Chip replaces obsolete ATE

ATML: Automatic Test Markup Language (IEEE-1671)

portfolio of interface cards: Not further detailed here for better clarity.

Thank you very much for your attention!

We have some time for discussion.