Connectivity for
Automotive Light & Vision

COSEDA UGM 2024

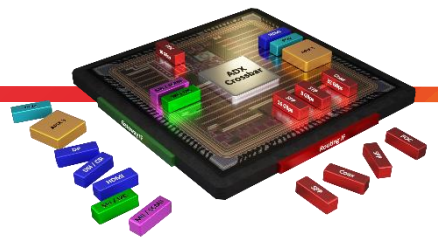
Paradigm Shift in Mixed Signal ASIC Design, Adopting SystemC and High-Level-Synthesis vs. Traditional RTL Design Flow

05.12.2024

inova Semiconductors

About us

- Fabless semiconductor manufacturer
- Founded in 1999
- Focus mainly on automotive
- Two main product lines
  - APIX -> APX
  - ISELED/ILAS



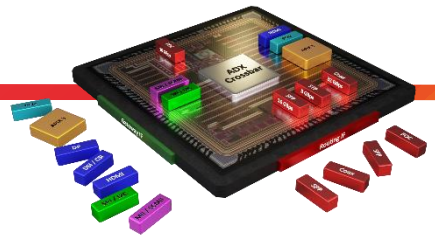
APIX Evolution

APIX®

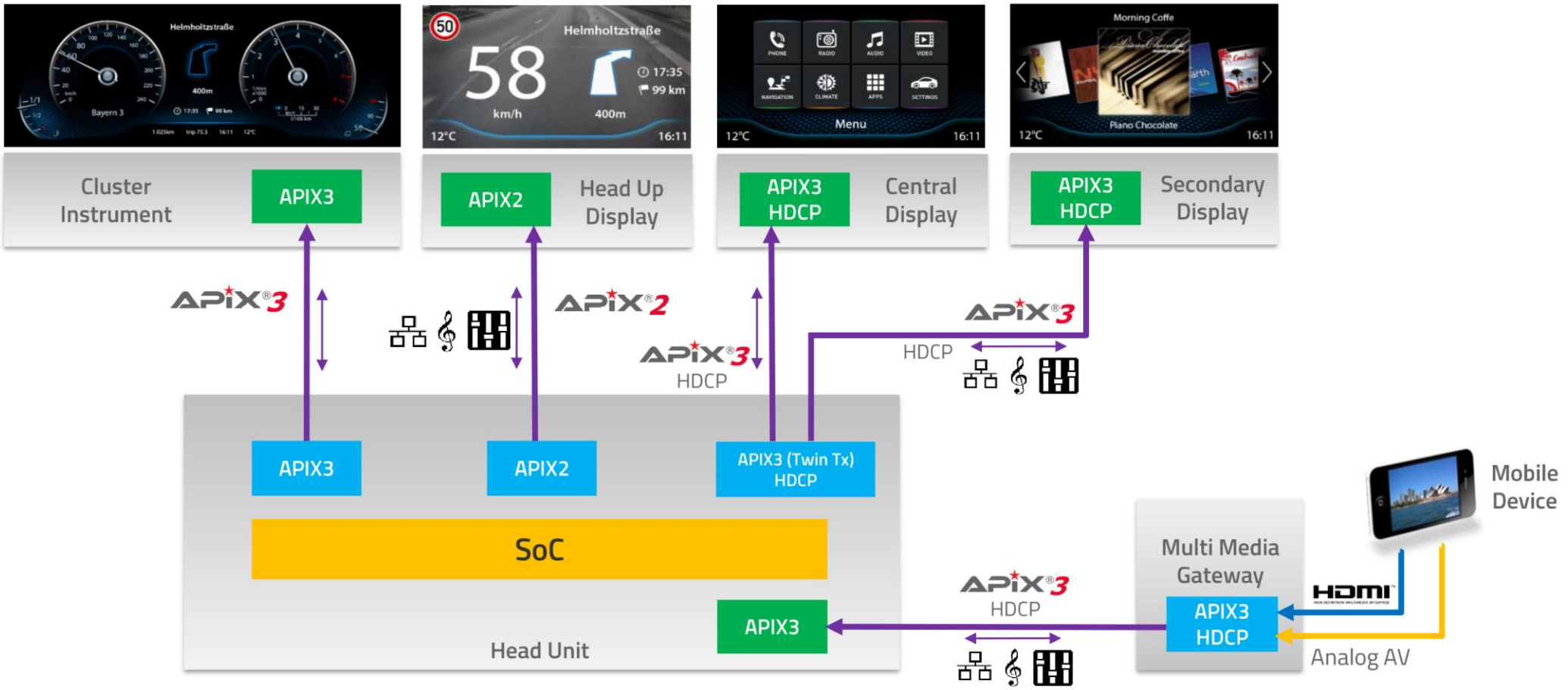
2008	2012	2018	2022
1 Gbps	**3 Gbps**	**12 Gbps**	**12 Gbps**
Video only	Multichannel HD Video Content Protection Ethernet Digital Audio	Multichannel UHD Video Content Protection Ethernet Digital Audio DSI & HDMI Digital Stream Compression Diagnostics Safety	Multichannel UHD Video Content Protection Ethernet Digital Audio DSI & HDMI Digital Stream Compression Diagnostics Safety Multistream Transport Display Port

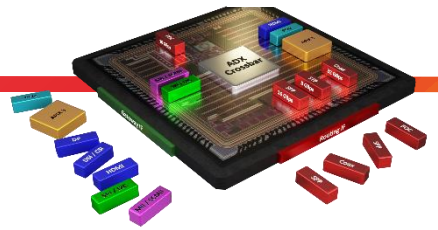
Coming soon

APXpress

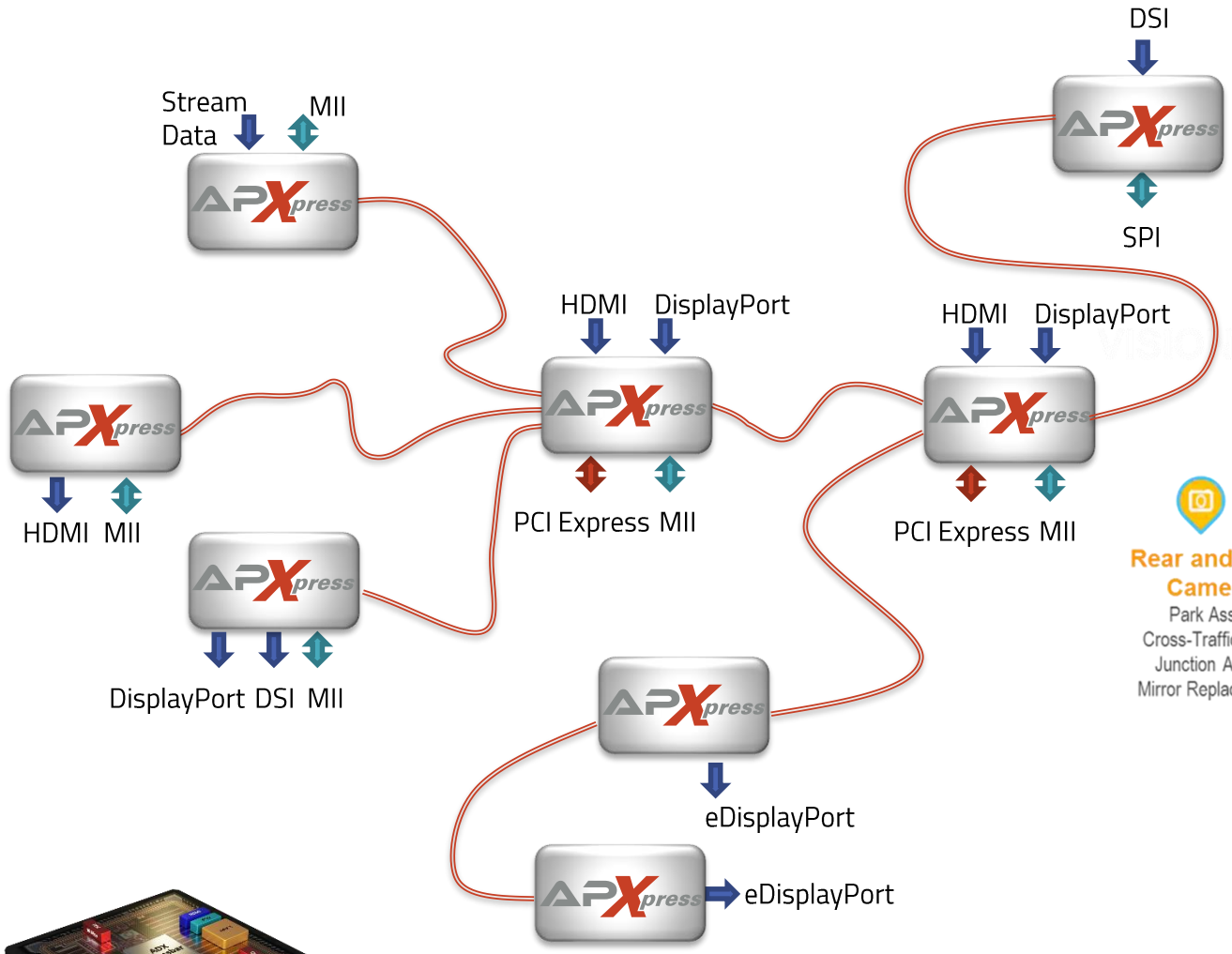


Current Customer Scenario

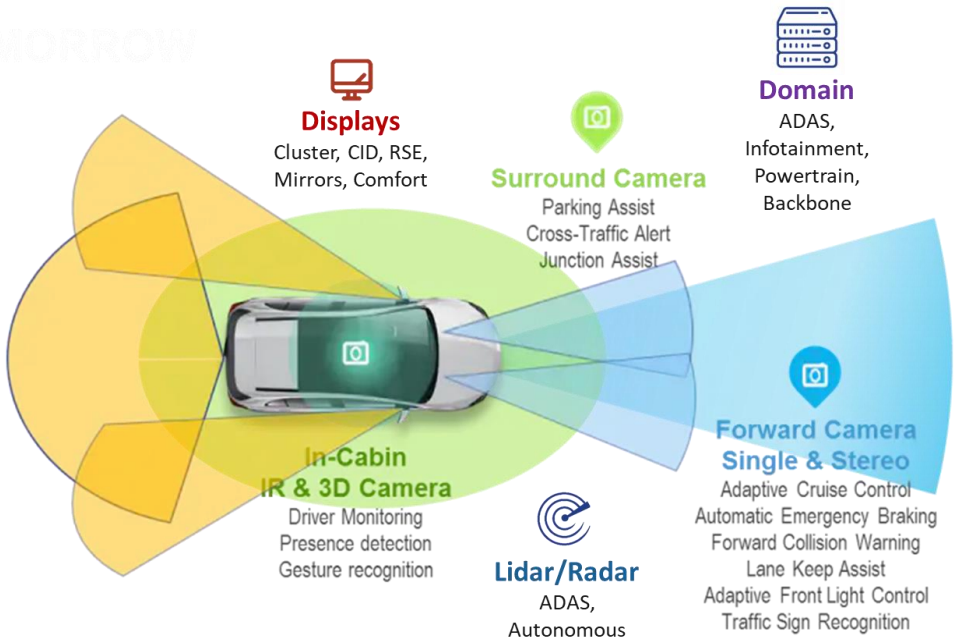


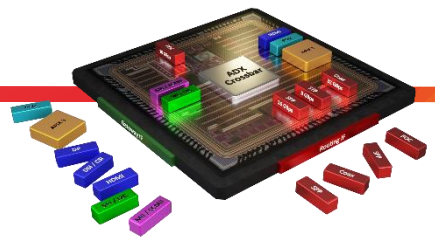


Shift to highly complex networks with massive bandwidth

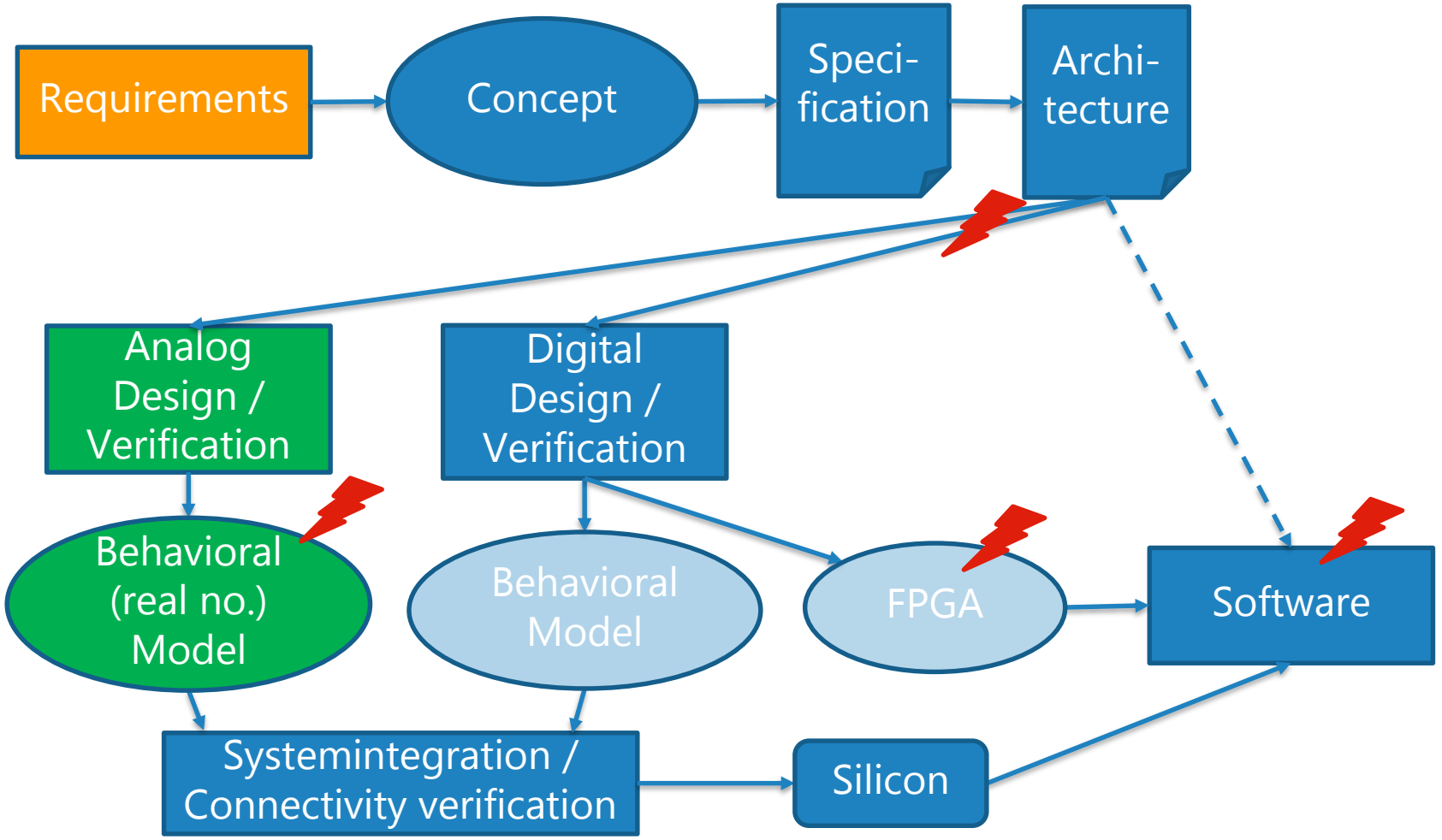


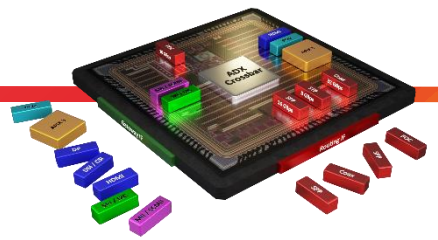
One technology bundles all data paths in the car using matching serial media.





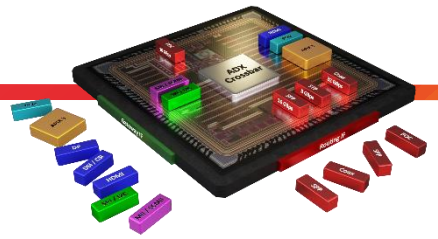
Issues with traditional Design Flow



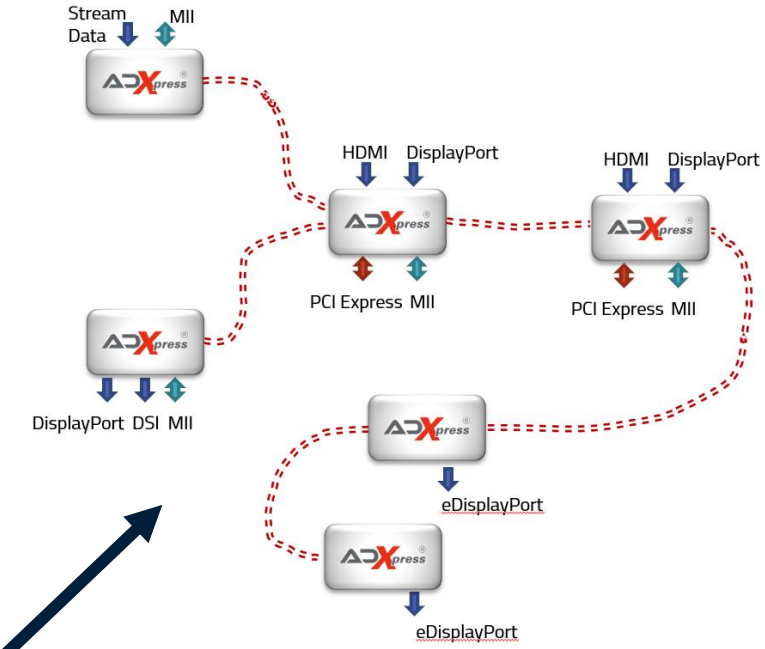


Need for change

- Tackle complexity by using SystemC as system-level language
- Executable specification
- Single language, single source approach from abstract algorithmic model to implementation
- HLS for automated path from functional SystemC to Verilog
- Re-use as much as possible
- Support for analog (SystemC AMS)
- IP protected model exchange with customers/suppliers possible
- HW/SW co-design – processor integration

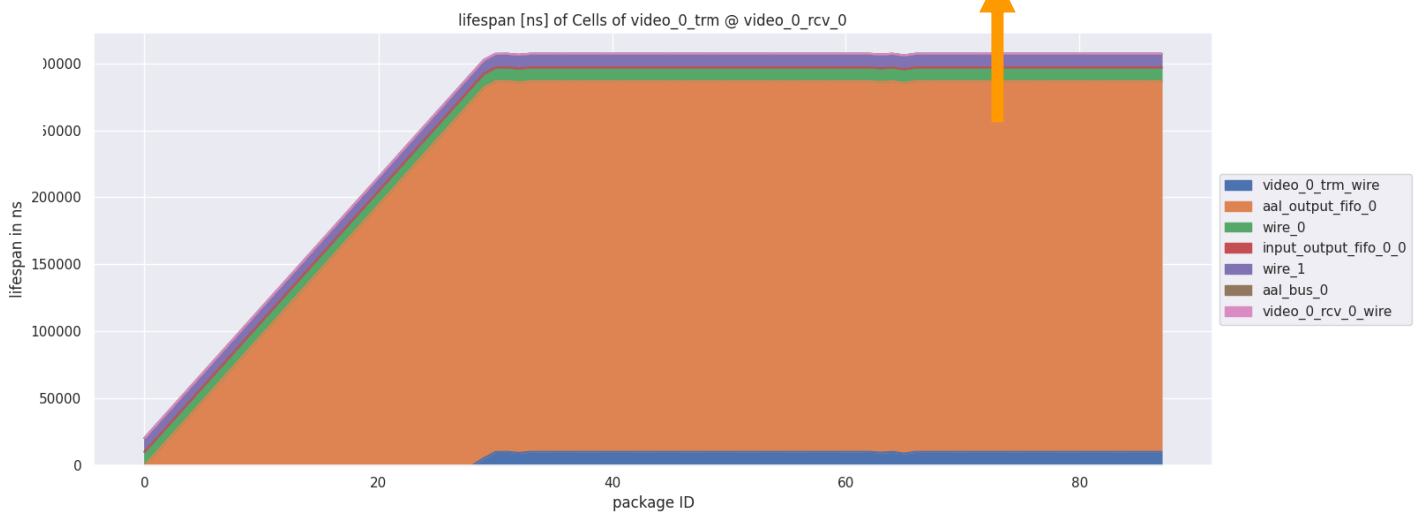


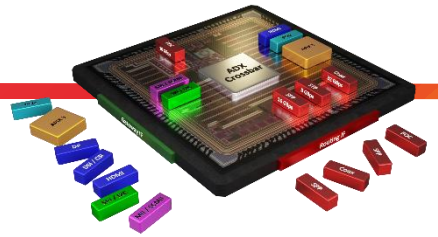
Abstract Modelling with SystemC



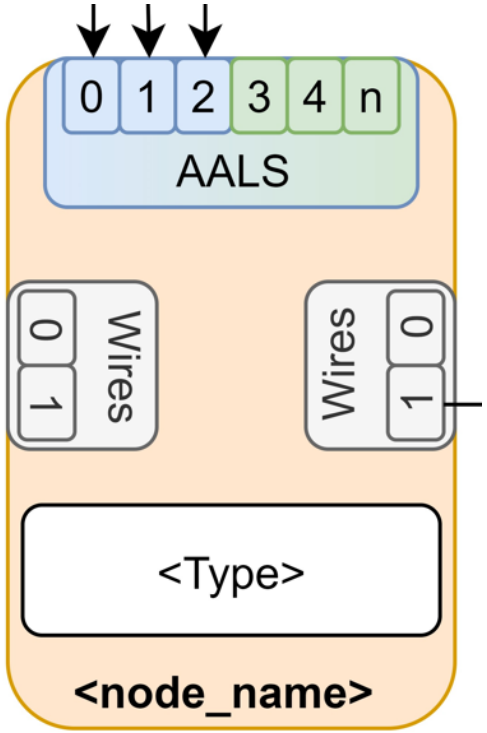
{JSON}

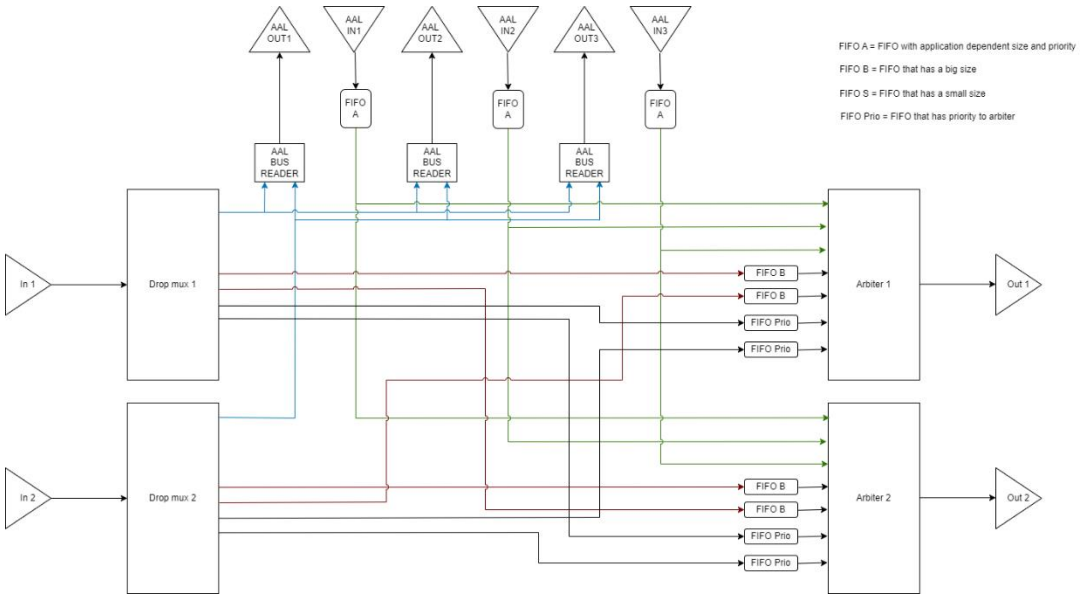
FIFO Overflow AAL

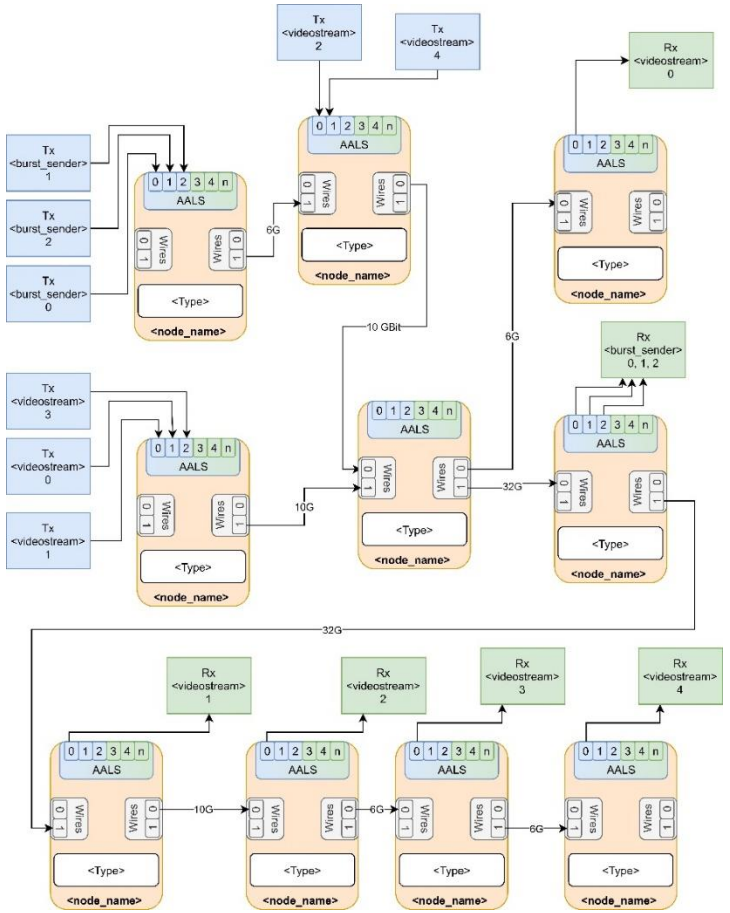


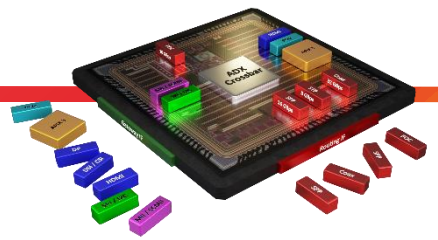


Abstract Modelling with SystemC: Find Bottlenecks





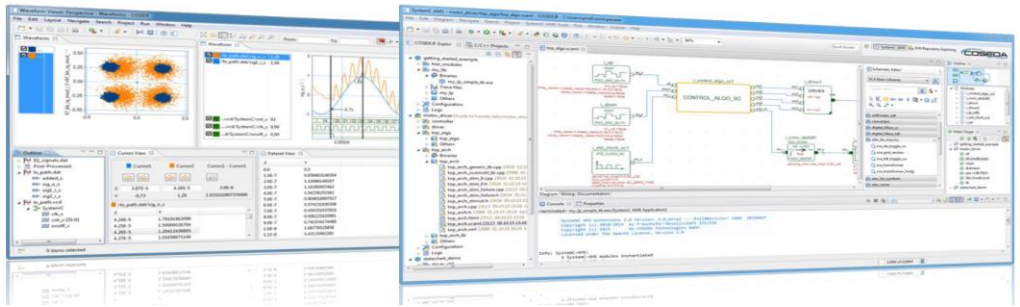




COSIDE & HLS Introduction

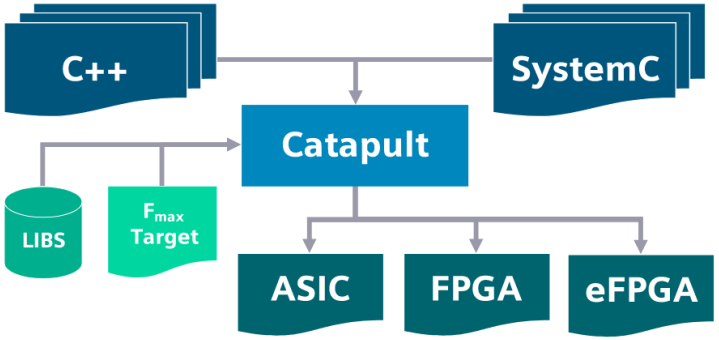
COSIDE: COSEDA Technologies

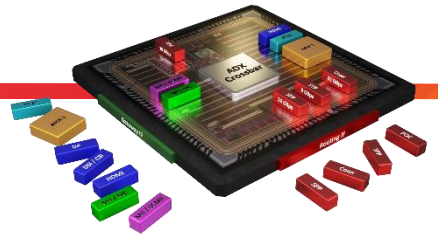
- System level design and verification tool
- SystemC and SystemC AMS as languages
- Architectural design exploration
- Hardware development in SystemC
- Hardware and software Co-Design
- Analog and mixed-signal HW & SW models
- Automated code generation



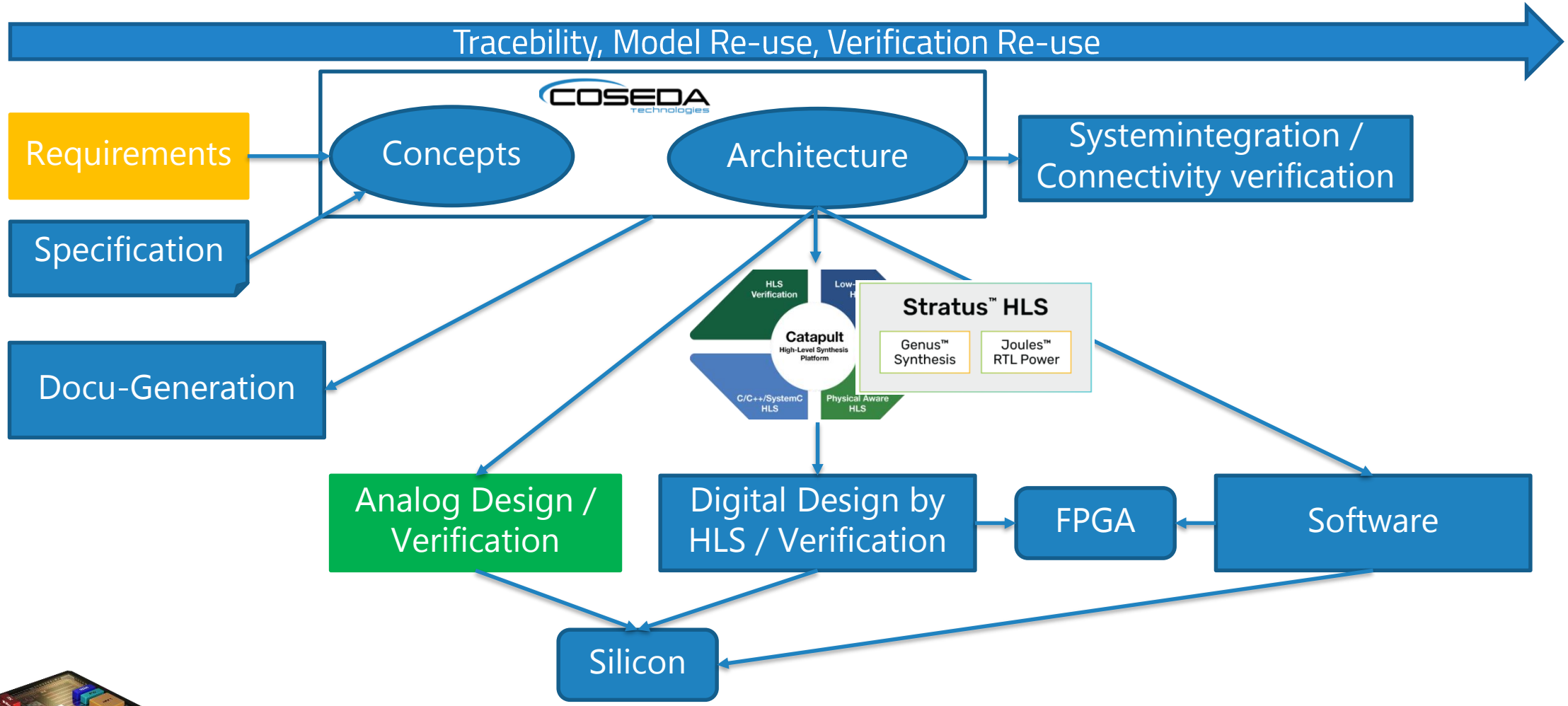
CATAPULT: Siemens EDA
Stratus: Cadence

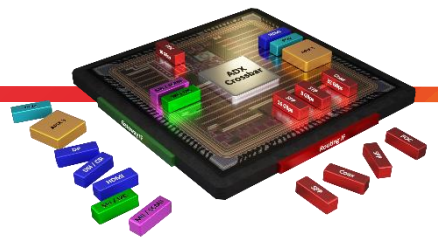
- High-Level-Synthesis & Verification tool
- Generate RTL based on SystemC (C++)
- RTL generation considers target node (pipelining, power etc.)
- Verification



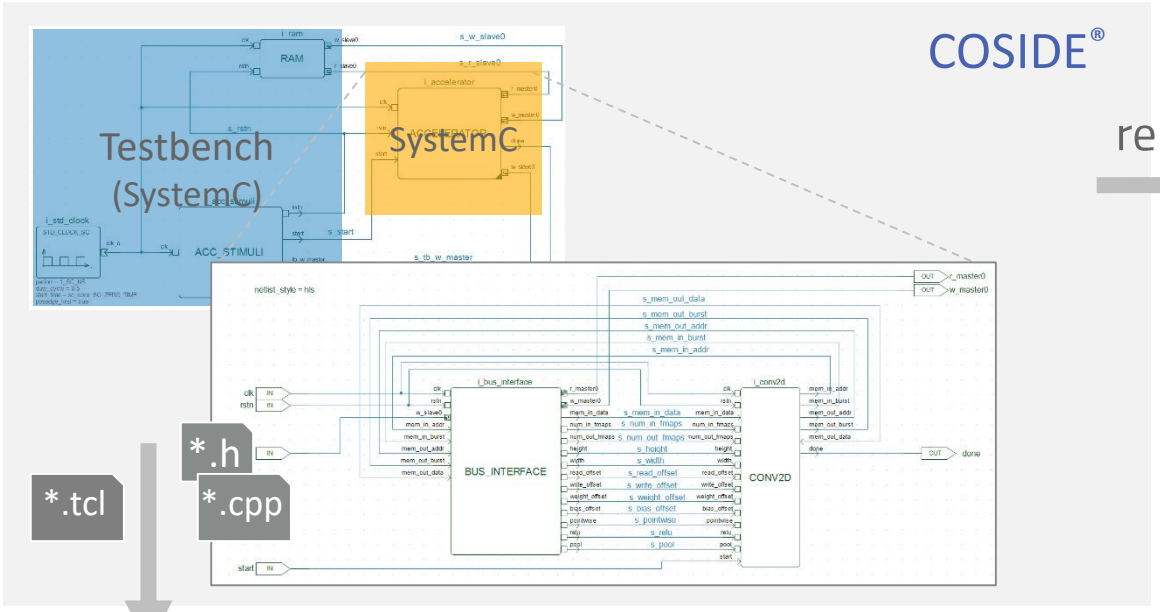


New Design Flow with SystemC: COSIDE® - Catapult® HLS Integration

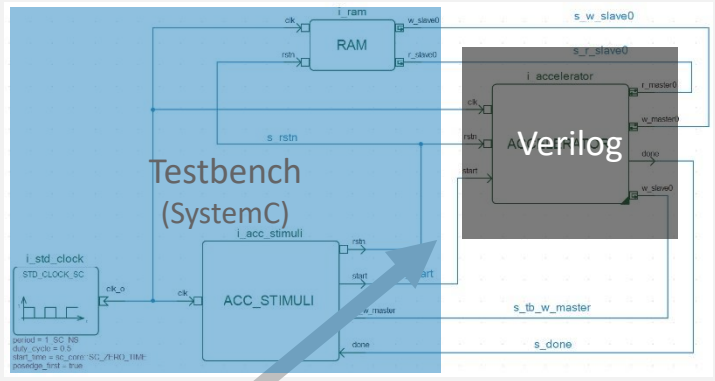




COSIDE® export to Catapult®



reuse

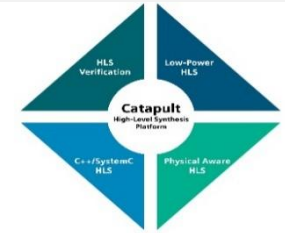


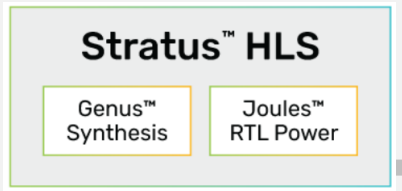
Verilator (import, built-in)

Questa/Xcelium (SystemC on top)

Multilanguage

verification





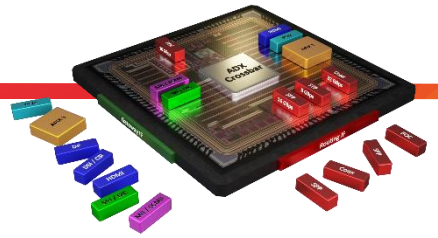
Catapult® / Stratus HLS

*.vhd

*.v

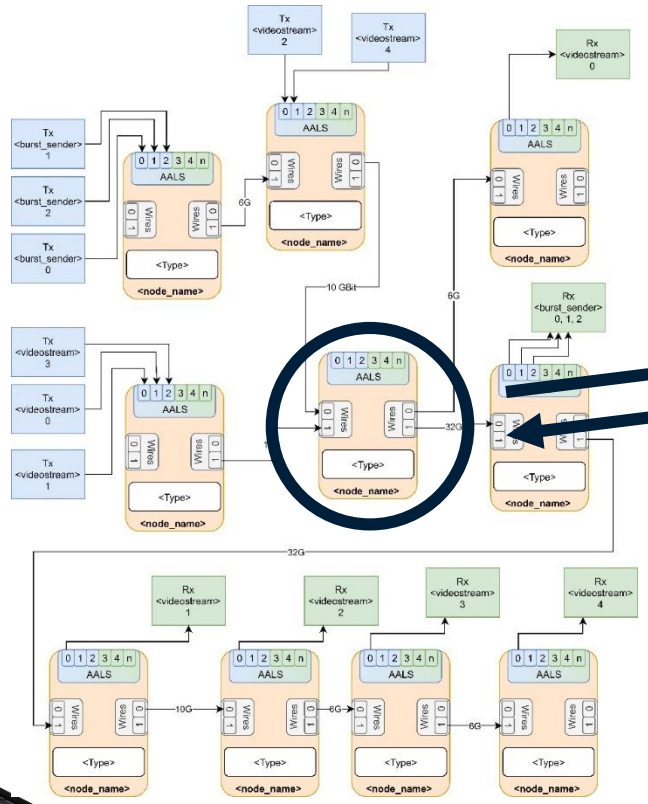
Logic synthesis tool

ASIC/FPGA

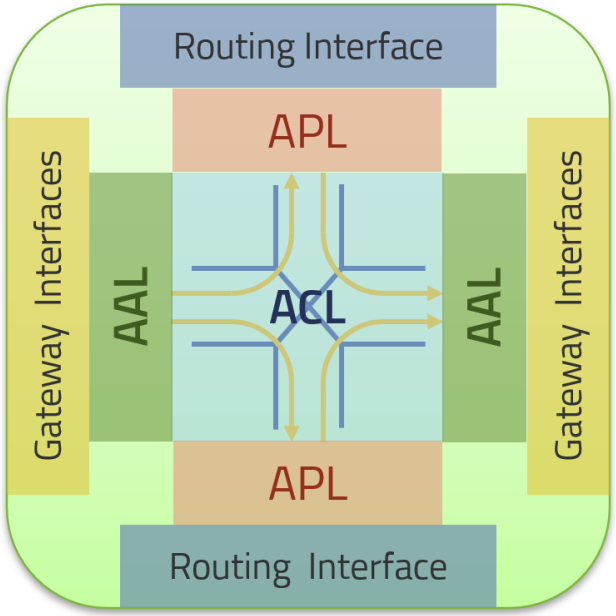


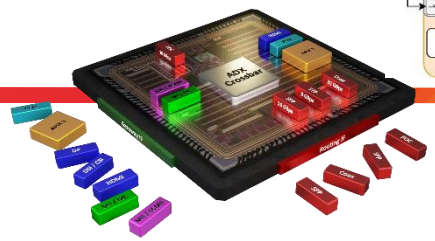
Extract Product from Abstract Model

Abstract Networksimulation

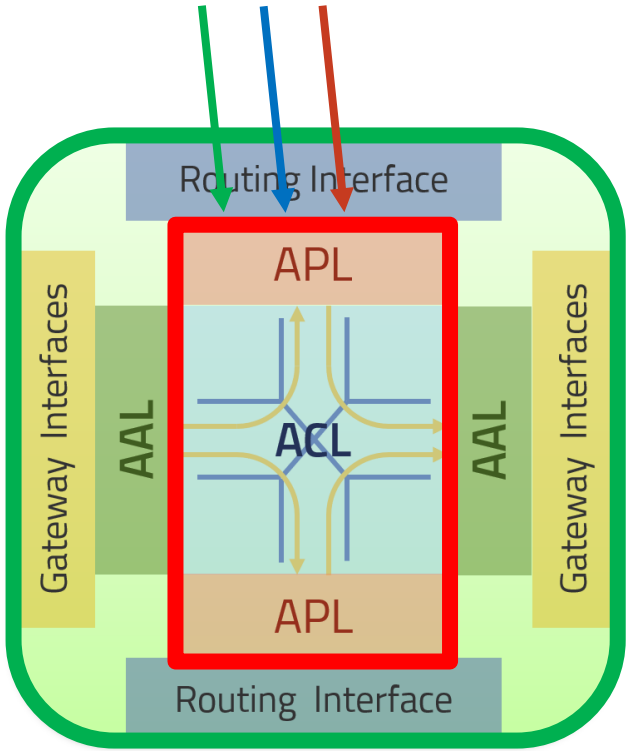


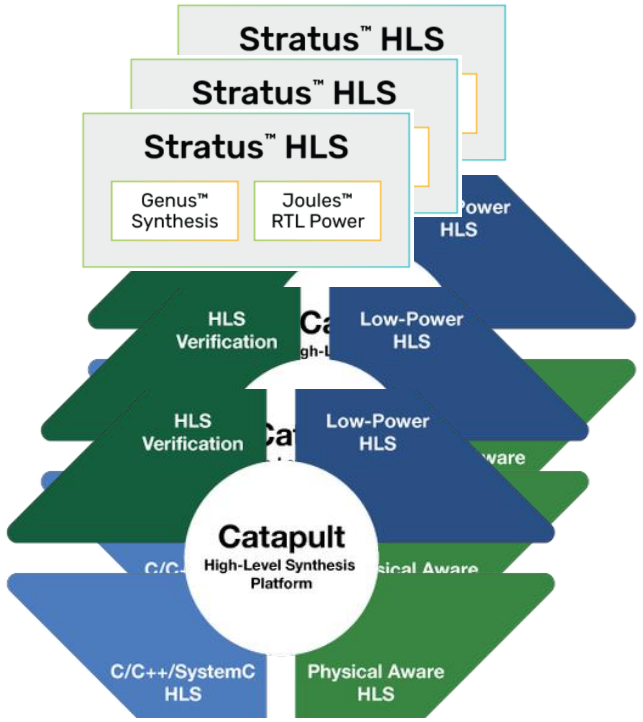
ADX Device

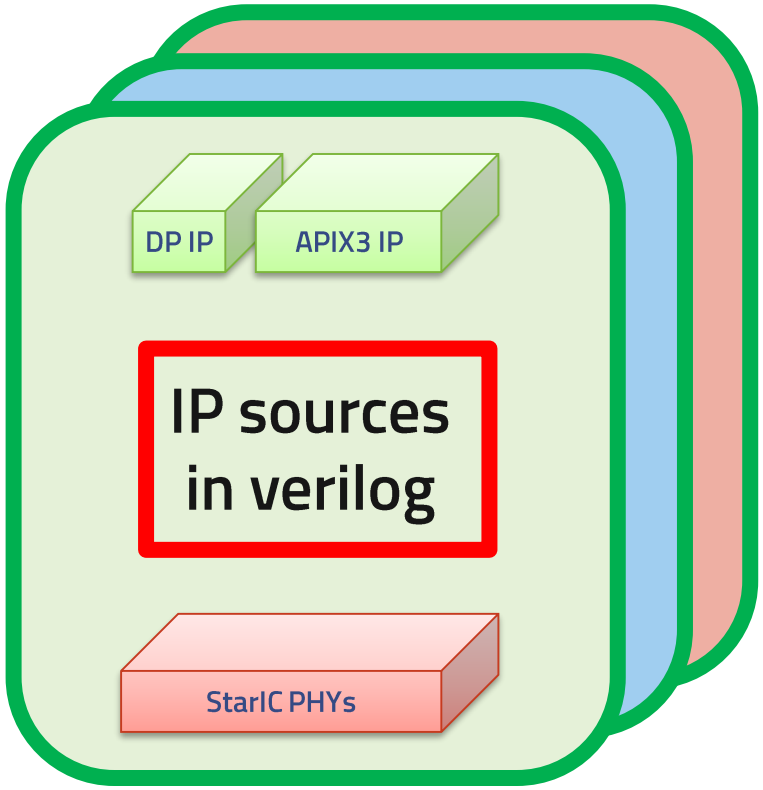




Our Goal: Easily generate new derivative Products



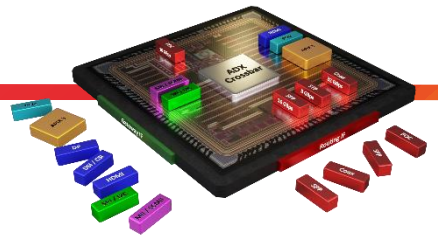




Generischer ADX-Core

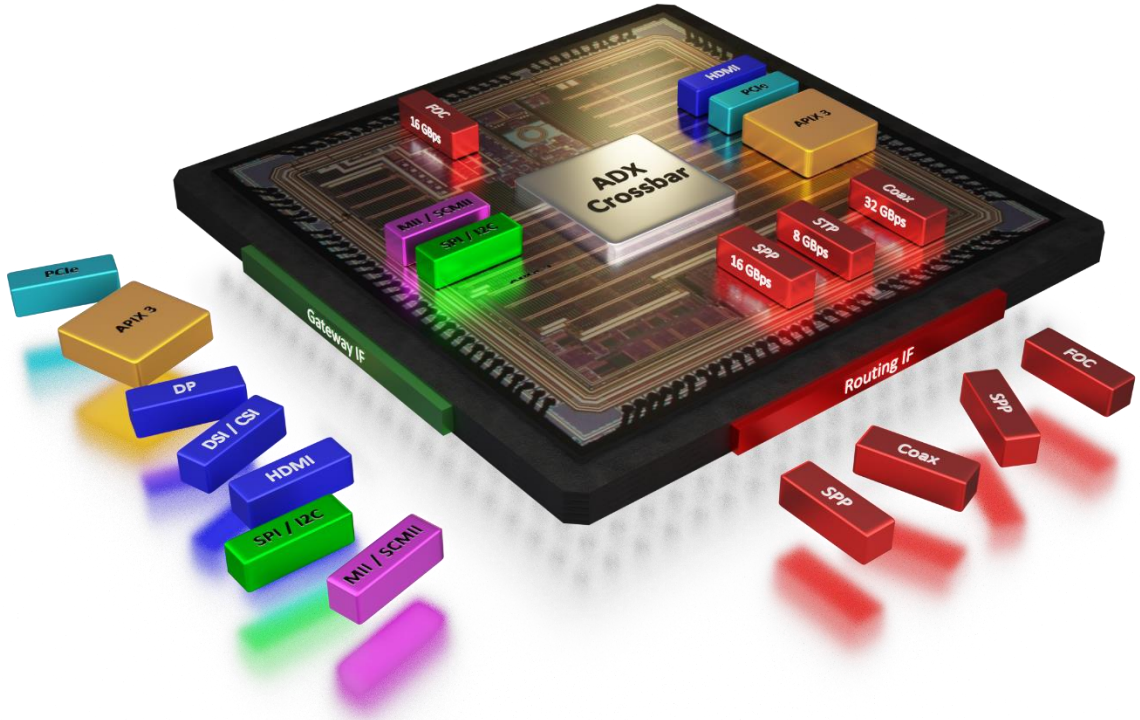
Multiple HLS runs

Derivate: 766TAQ 768TAQ 766RAQ



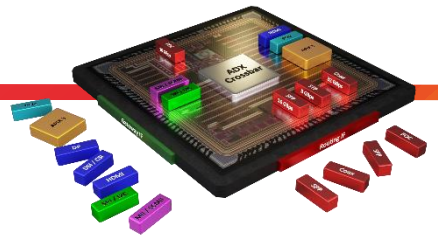
Derivative Products

- Easily generate new derivatives
- Development cycle shortens from several months to weeks
- Modular construction kit, driven by single source model in COSIDE

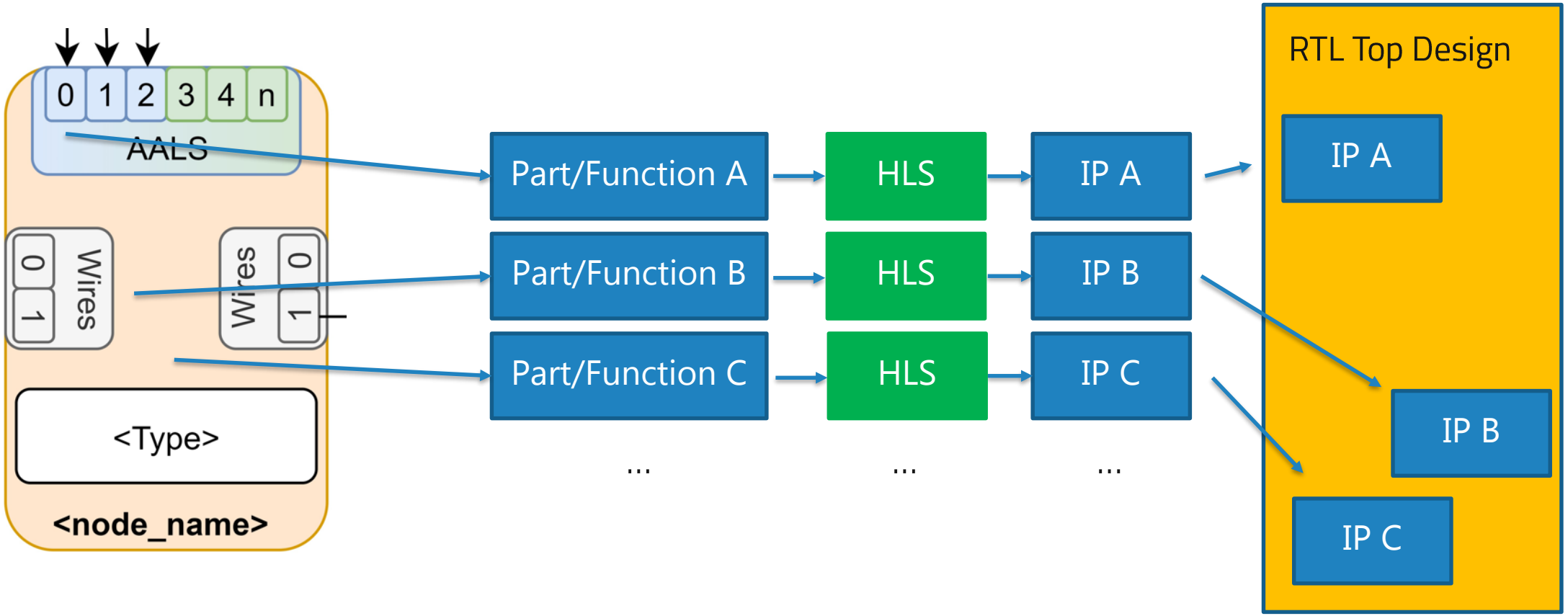


HLS: Node-Class

- One Node Superclass used in abstract simulation/modelling
- Abstract simulation is not HLS-able
- More refinement is possible via the use of „virtual“
- Once HLS-able check of RTL output with scverify (smoke test / multi language)
- Pipelining done by Catapult/Stratus, needs to be considered
- Divide and conquer

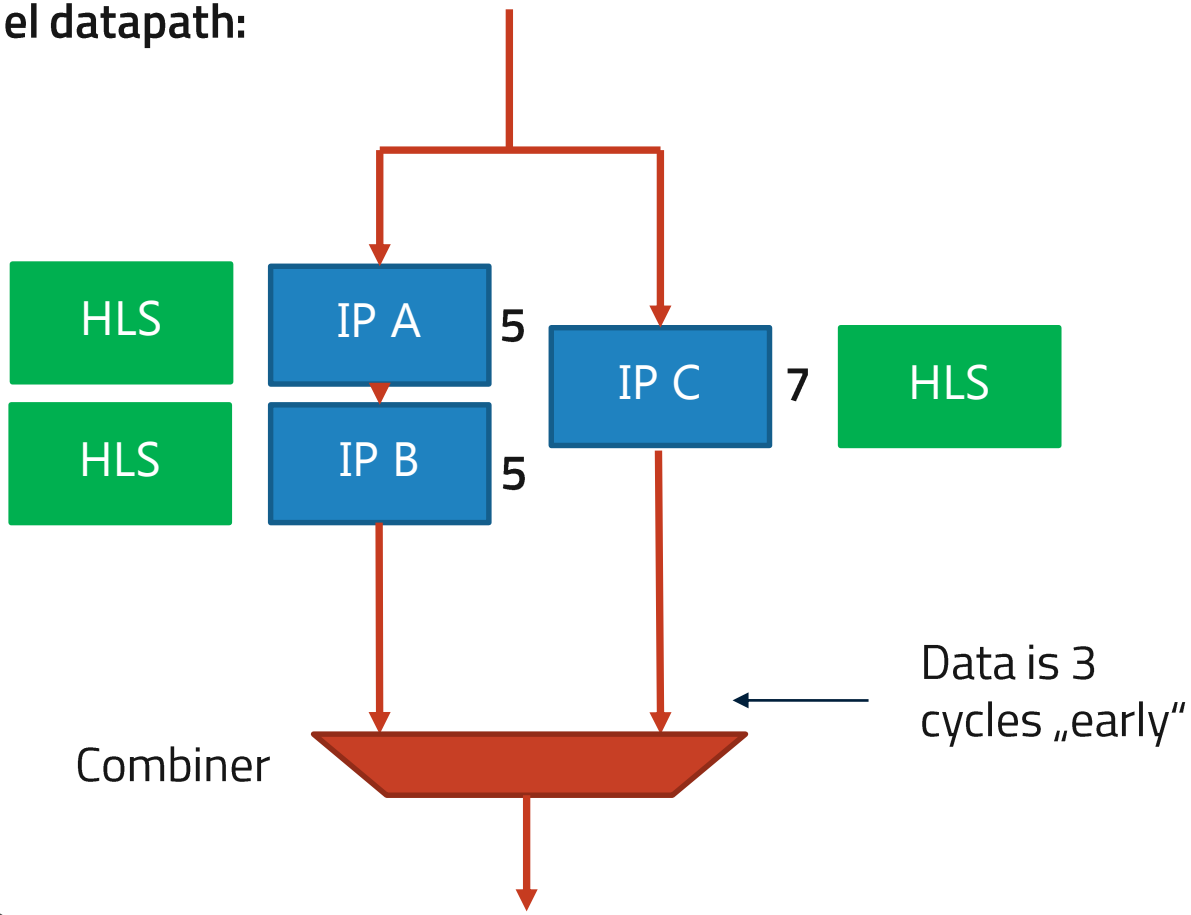


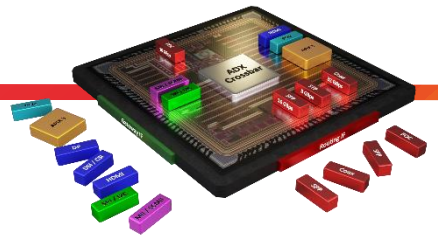
HLS: Node-Class



HLS: Pipelining and Channels

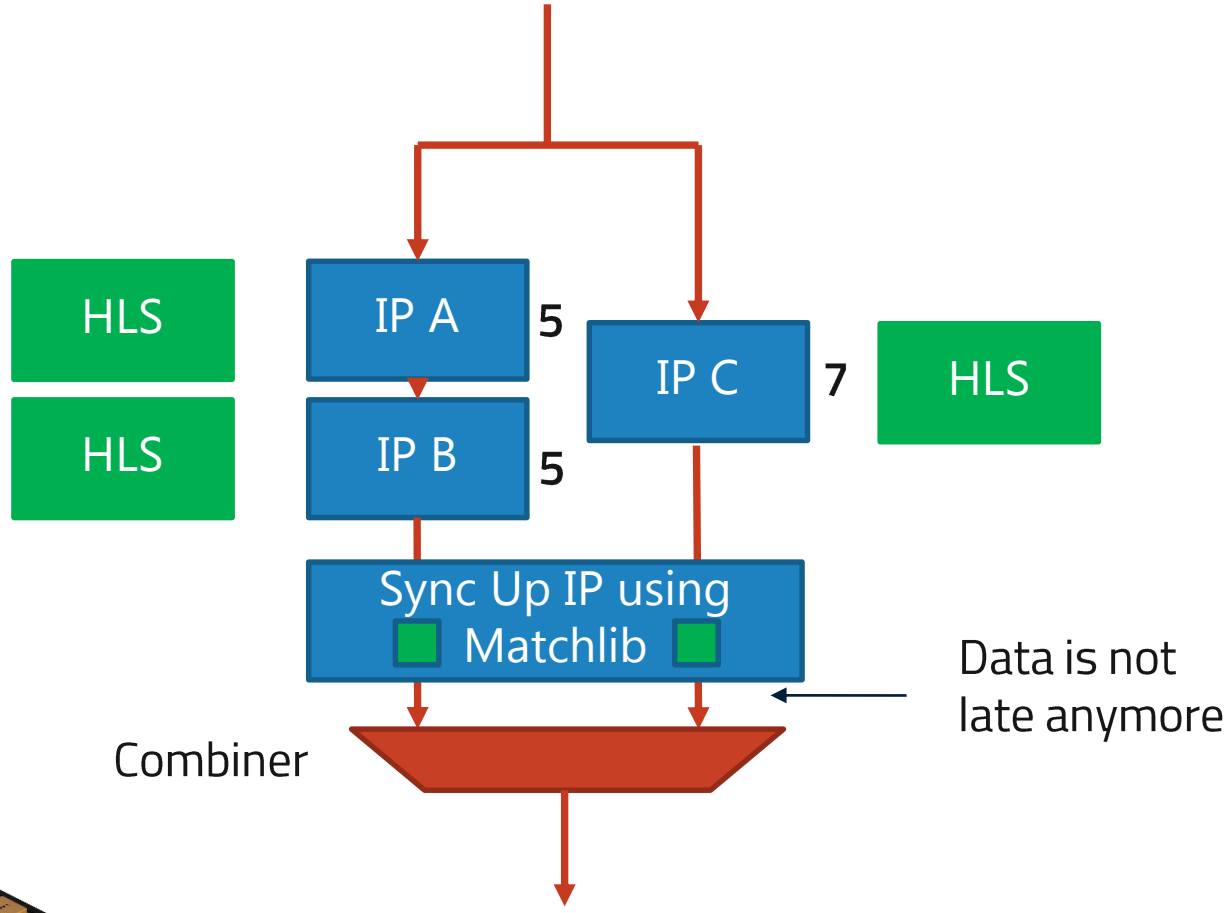
Issue: IPs with potentially changing latency used in parallel datapath:

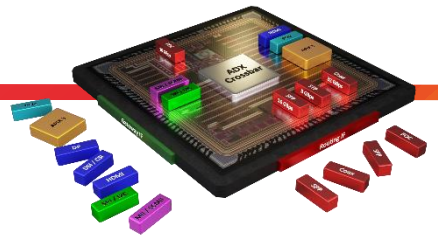




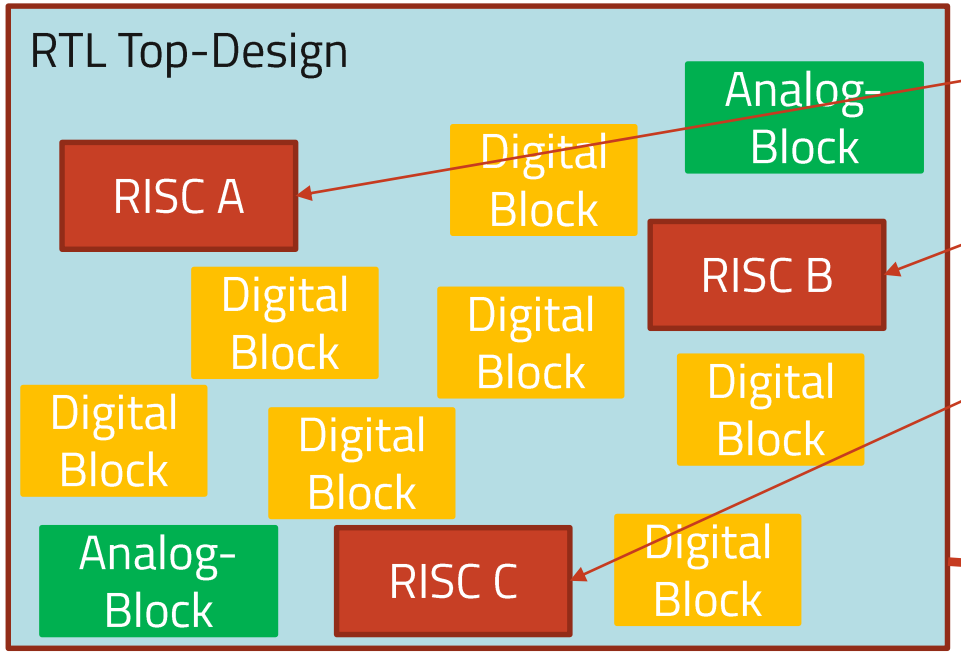
Catapult HLS: Pipelining and Channels

Solution: Use channels / Matchlib for sync-up





Firmware Development: The traditional way



.elf A

.elf B

.elf C

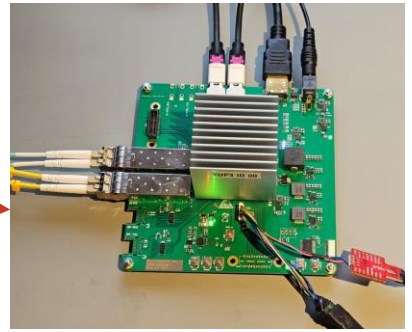
AMD Vivado

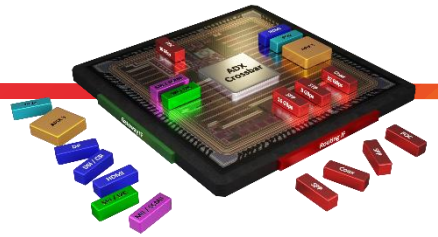
Functional description of RISC-Cores neccessary to run a .elf file directly

Very Accurate

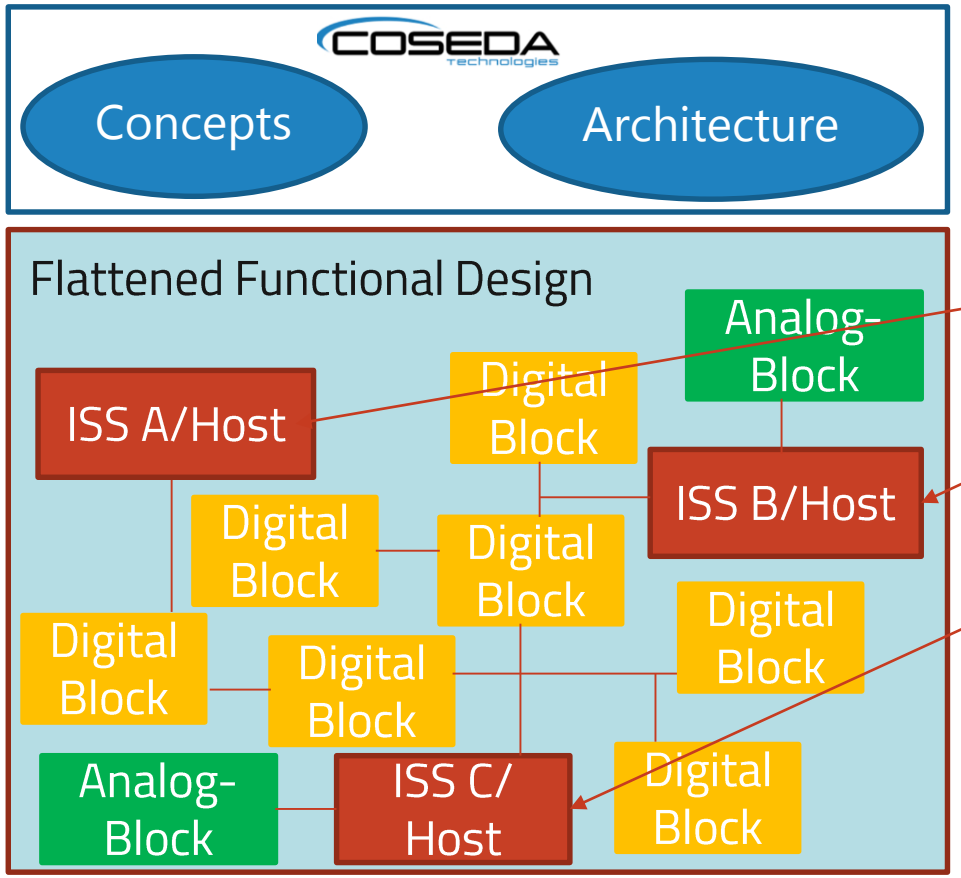
Can be taken directly into FPGA Prototype

Hard debugging





Firmware Development: The fast and early way using Instruction Set Simulator

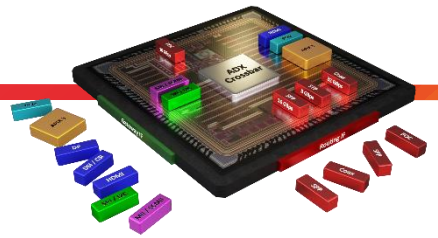


Simulation is much faster using ISS that simply do memory mapped transactions to the digital / analog blocks

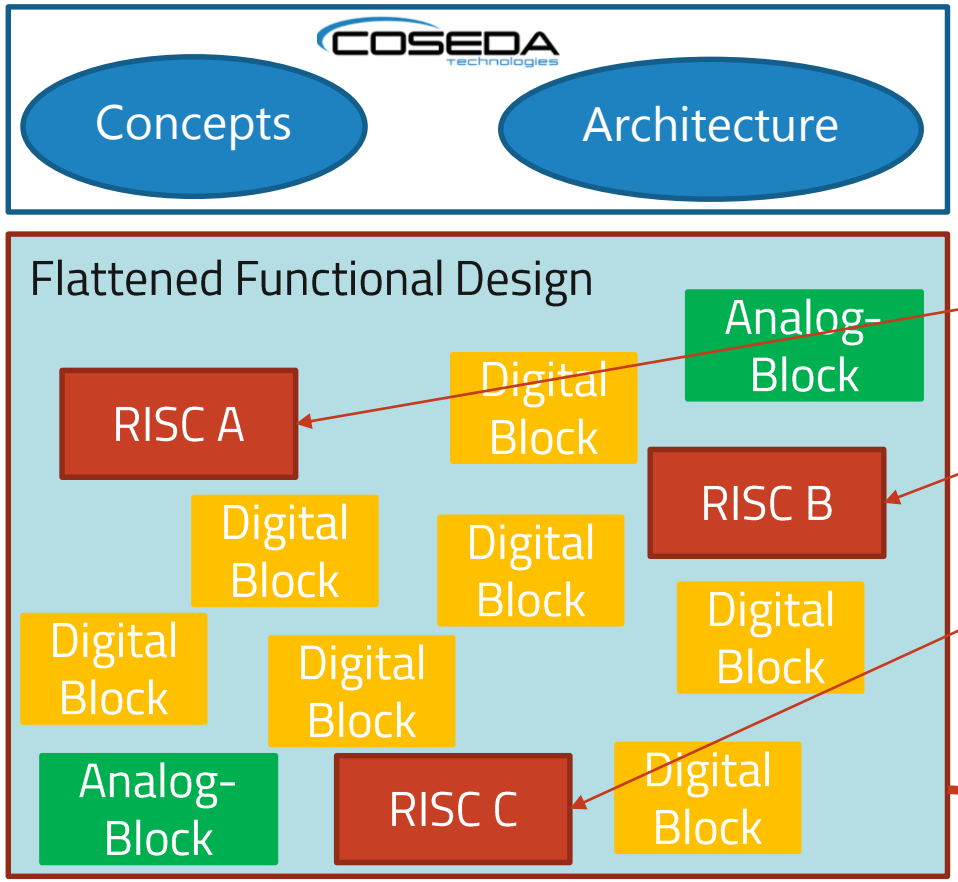
Start early with abstract design model

Host-Compile also possible

FW developers can basically work in parallel with digital designers



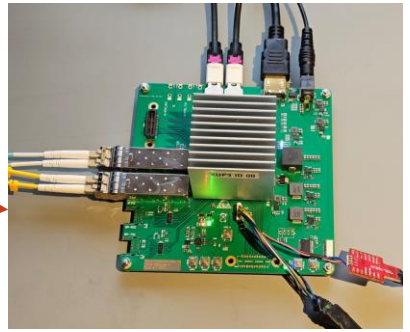
Firmware Development: The HLS way

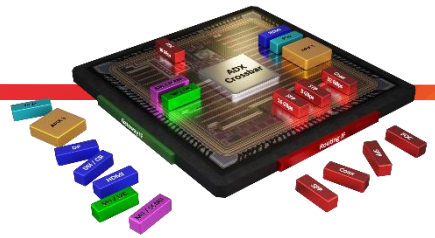


Enable traditional flow via HLS

Hardware-sign-off

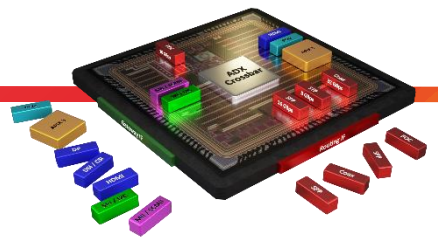
SystemC to FPGA/ASIC





Results / Impressions

- COSIDE and Catapult/Stratus work well together,
  => Excellent support from all three companies! Thank you!
- Mapping to FPGA successful
- Verilator very useful but:
  => Multi-Language-Simulation is necessary for ISO26262 flow with existing Verilog-IP
- We will get a SystemC-AMS model for our new transceiver (very useful!)
- It is key to understand what SystemC/C++ code constructs are possible with HLS and which are not
- Switching from RTL-Design to SystemC-Design can be very hard (carefully convince team)
  => Lower entry level to SystemC using a proper tool



Thank you