Versal Portfolio Product Overview

Jason Vidmar

System Architect, MILCOM / SATCOM / Machine Learning

jasonv@xilinx.com

Mar 14, 2019

XILINX

Agenda

> Introducing Versal: The First ACAP

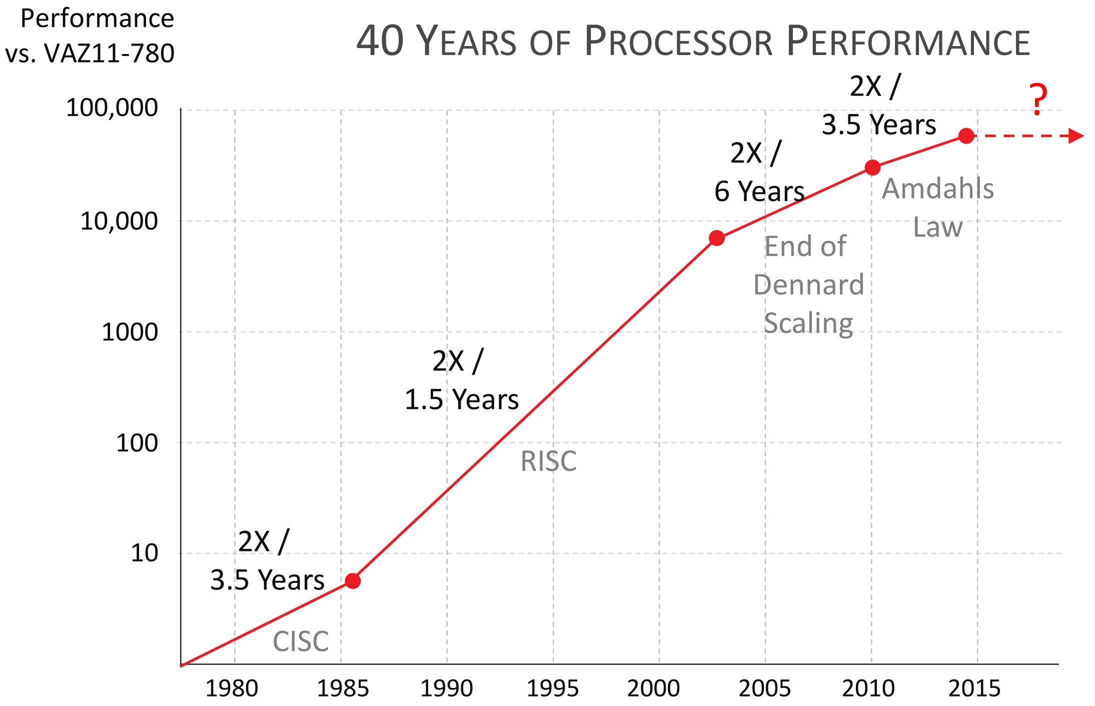
> Heterogeneous Acceleration Engines

> Key Architectural Blocks (Focus on AI Engines)

> Product Portfolio

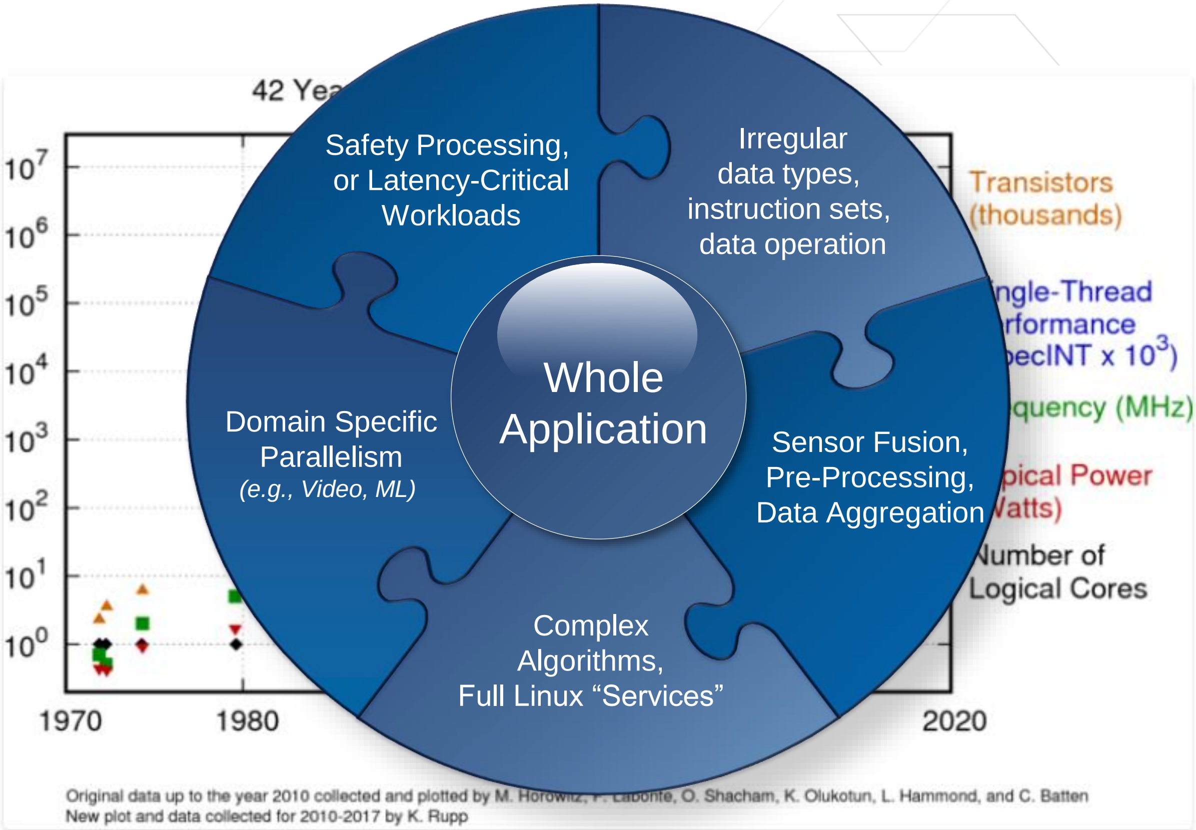
The Technology Conundrum .. And the Need for a New Compute Paradigm

Processing Architectures are Not Scaling



Source: John Hennessy and David Patterson, Computer Architecture: A Quantitative Approach, 6/e 20

A Single Architecture Can't Do It Alone



42 Yea
10^7
10^6
10^5
10^4
10^3
10^2
10^1
10^0
1970
1980
2020
Transistors (thousands)
ngle-Thread rformance ecINT x 10^3)
quency (MHz)
pical Power Vatts)
Number of Logical Cores
Safety Processing, or Latency-Critical Workloads
Irregular data types, instruction sets, data operation
Whole Application
Domain Specific Parallelism (e.g., Video, ML)
Sensor Fusion, Pre-Processing, Data Aggregation
Complex Algorithms, Full Linux "Services"

Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2017 by K. Rupp

WP505: “Versal: The First Adaptive Compute Acceleration Platform.”

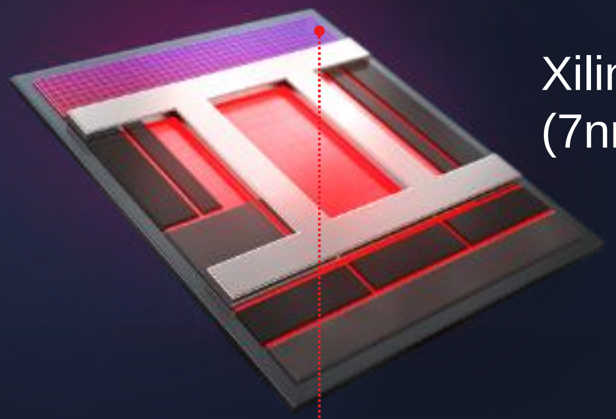
WP506: “Xilinx AI Engines and Their Applications.”

Disruptive Innovation Needed:

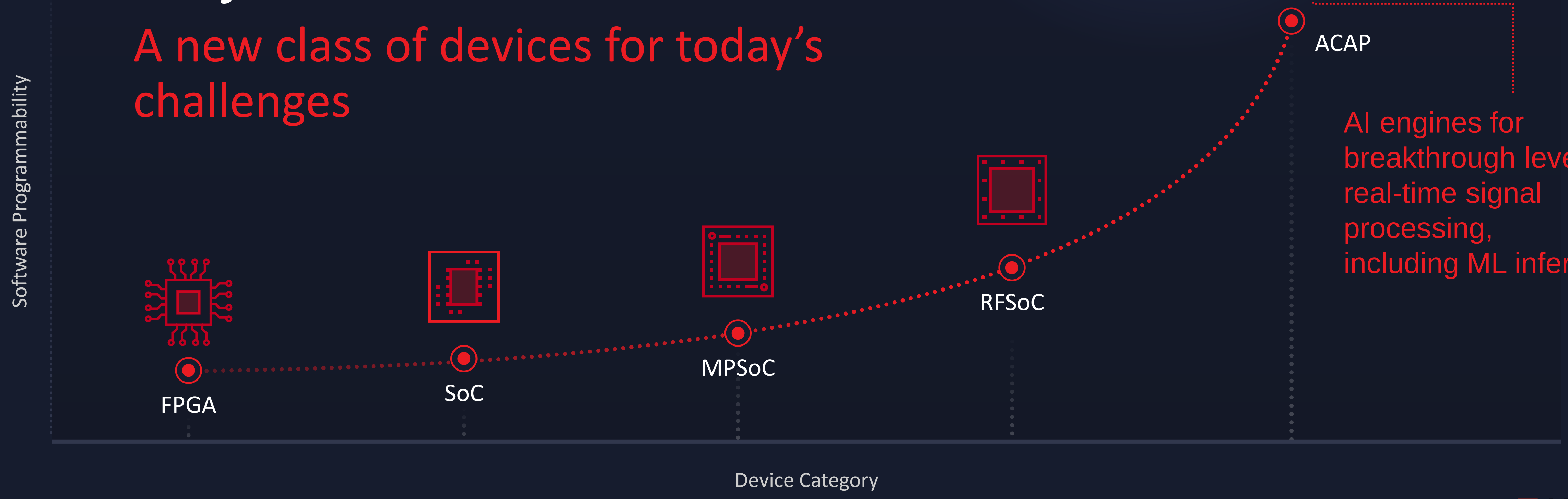
ACAP

Adaptive Compute Acceleration Platform

A new class of devices for today’s challenges



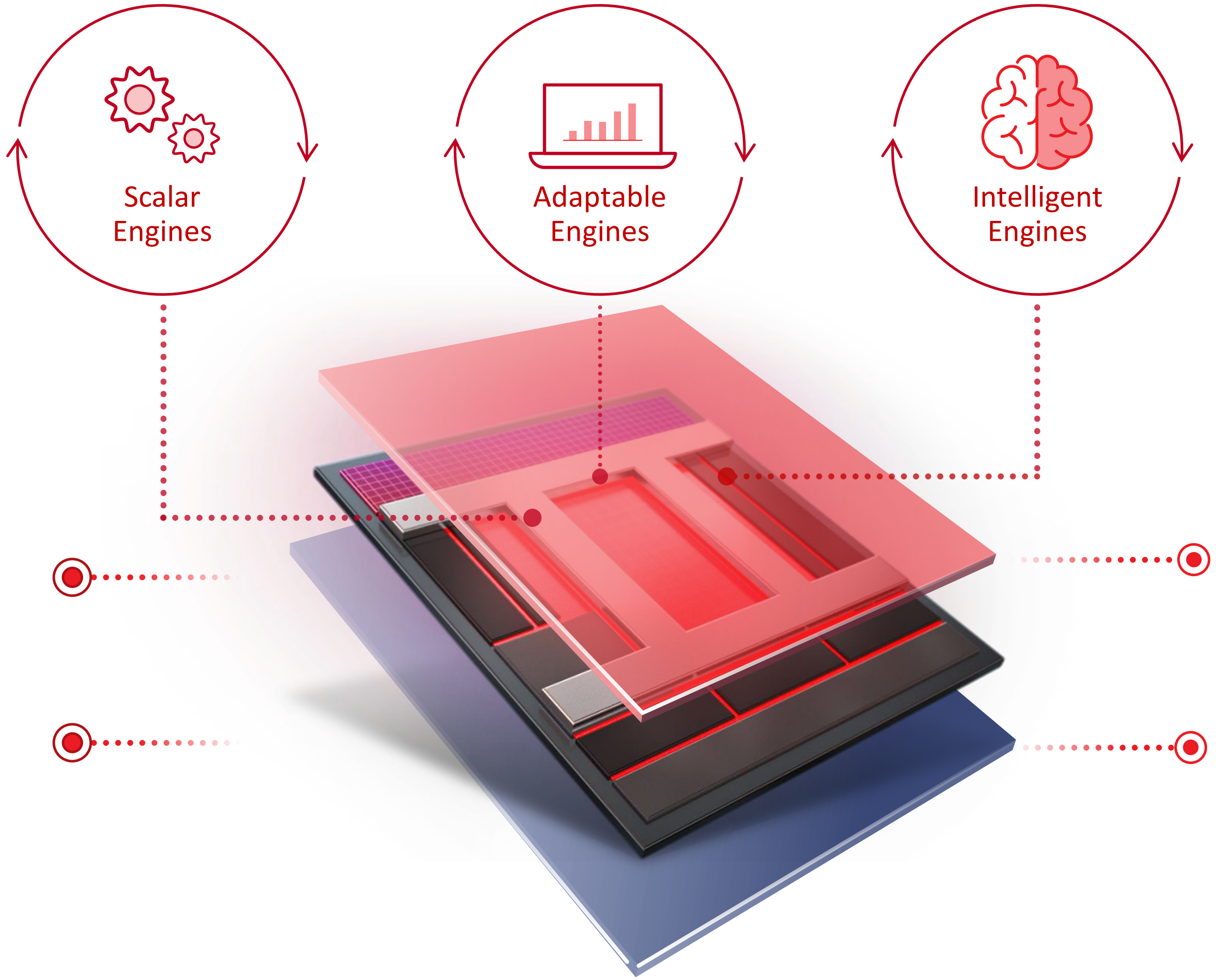
Xilinx Versal (7nm)



AI engines for breakthrough levels of real-time signal processing, including ML inference.

New Device Category: Adaptive Compute Acceleration Platform

COMPUTE ACCELERATION



Scalar Engines

Adaptable Engines

Intelligent Engines

ADAPTIVE

Diverse Workloads in Milliseconds

Future-Proof for New Algorithms

PLATFORM

Development Tools
HW/SW Libraries
Run-time Stack

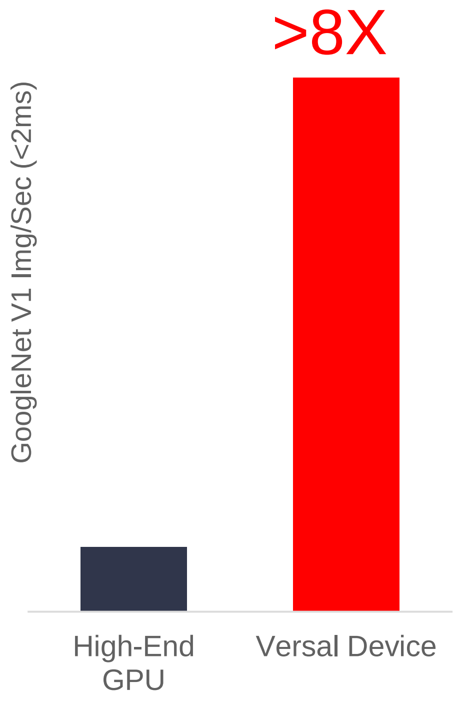
SW Programmable Silicon Infrastructure

Enabling Data Scientists, SW Developers, HW Developers

Breakthrough Performance for Cloud, Network, and Edge

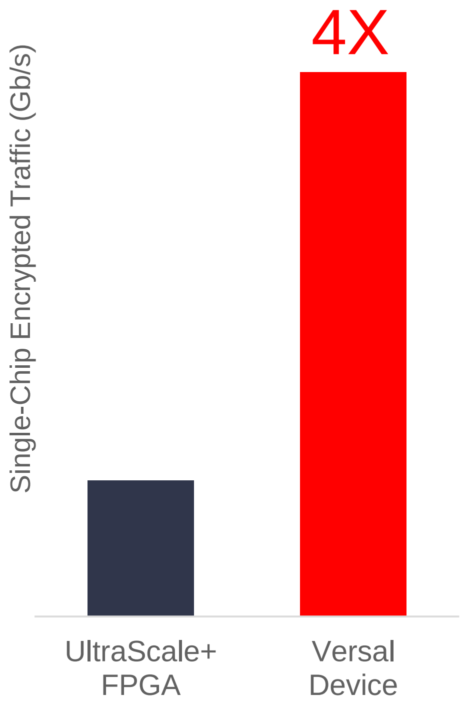
Cloud Compute

Breakthrough AI Inference



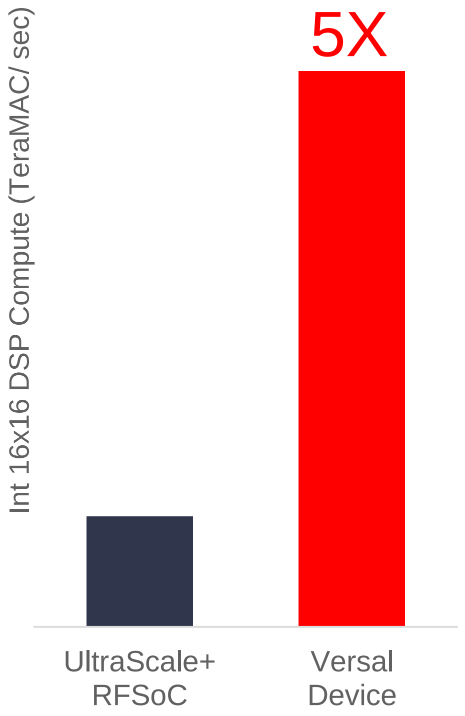
Networking

Multi-terabit Throughput



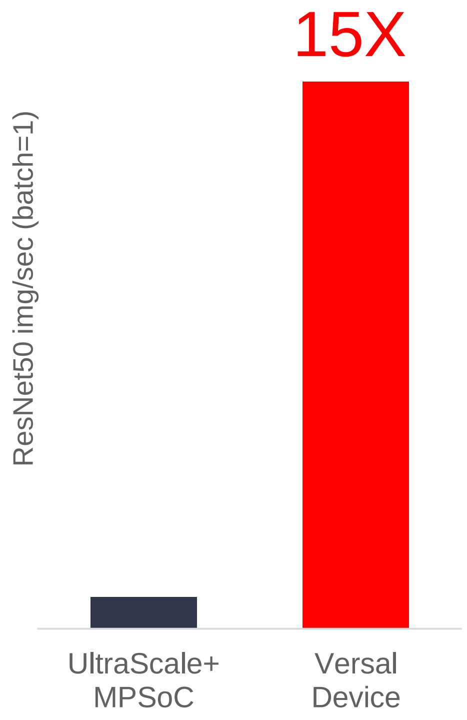
5G Wireless

Compute for Massive MIMO



Edge Compute

AI Inference at Low Power



Versal Architecture Overview

Adaptable Engines

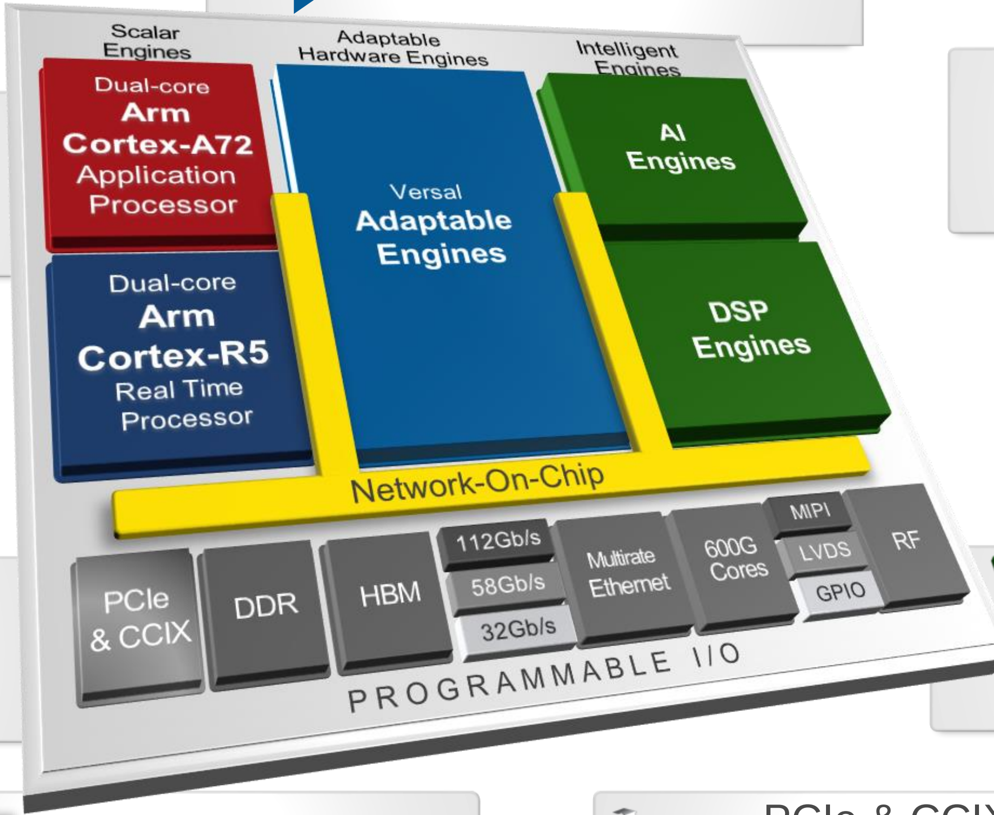
2X compute density

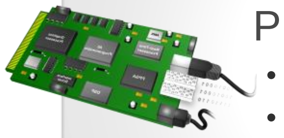
Scalar Engines

- Platform Control
- Edge Compute

Intelligent Engines

- AI Compute
- Diverse DSP workloads





Protocol Engines

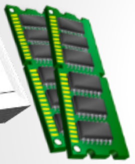
- Integrated 600G cores
- 4X encrypted bandwidth

Network-on-Chip

- Guaranteed Bandwidth
- Enables SW Programmability

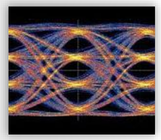
Programmable I/O

- Any interface or sensor
- Includes 3.2Gb/s MIPI



DDR Memory

- 3200-DDR4, 4266-LPDDR4
- 2X bandwidth/pin



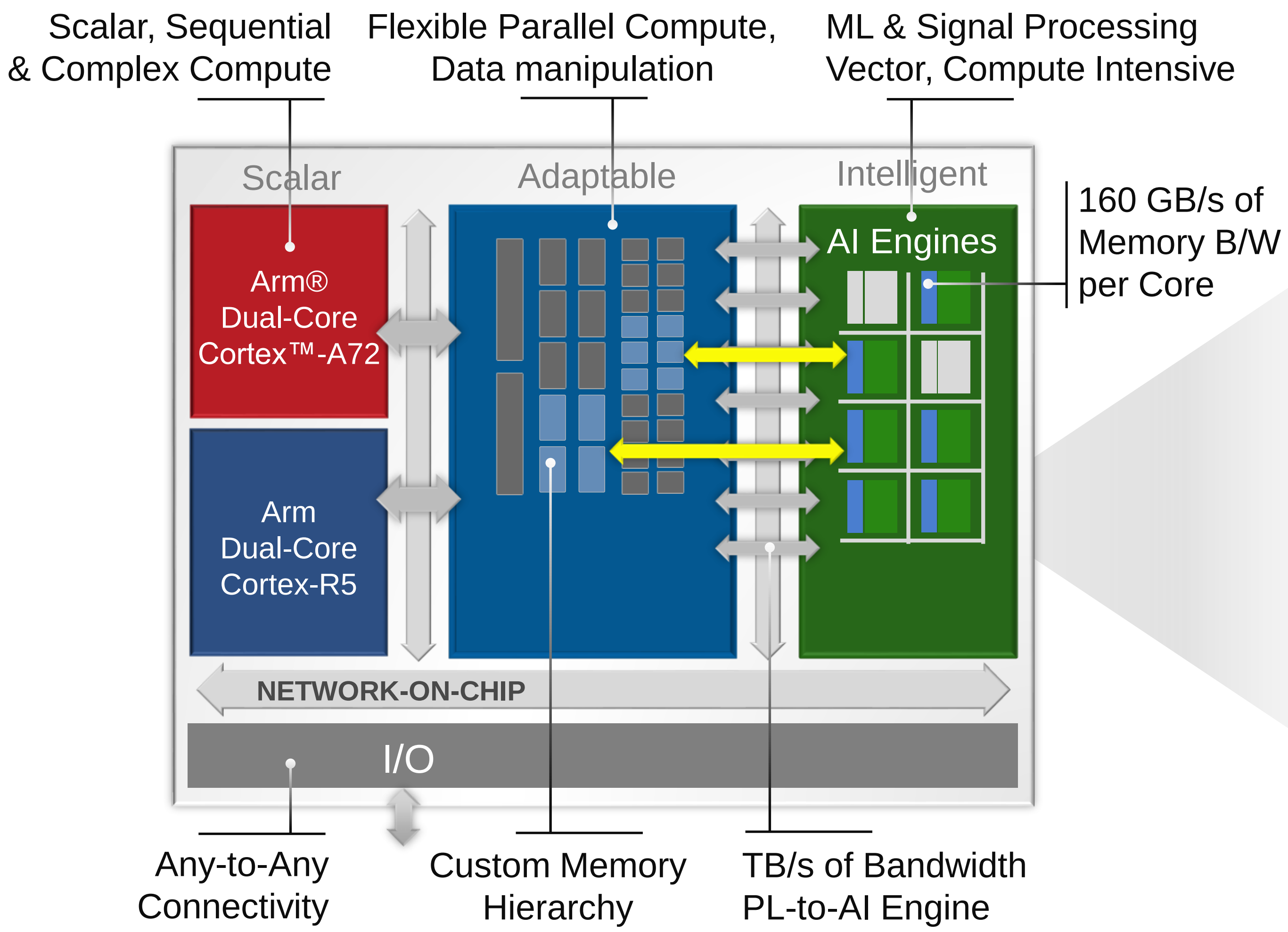
Transceivers

- Broad range, 25G →112G
- 58G in mainstream devices

PCIe & CCIX

- 2X PCIe & DMA bandwidth
- Cache-coherent interface to accelerators

Hardware Adaptable: Accelerating the Whole Application


Scalar, Sequential & Complex Compute
Flexible Parallel Compute, Data manipulation
ML & Signal Processing Vector, Compute Intensive
Scalar
Adaptable
Intelligent
Arm® Dual-Core Cortex™-A72
Arm Dual-Core Cortex-R5
AI Engines
160 GB/s of Memory B/W per Core
NETWORK-ON-CHIP
I/O
Any-to-Any Connectivity
Custom Memory Hierarchy
TB/s of Bandwidth PL-to-AI Engine

Heterogeneous Acceleration from Data Center to the Edge

- Video + AI
- Genomics + AI
- Risk Modeling + AI
- Database + AI
- Network IPS + AI
- Storage + AI

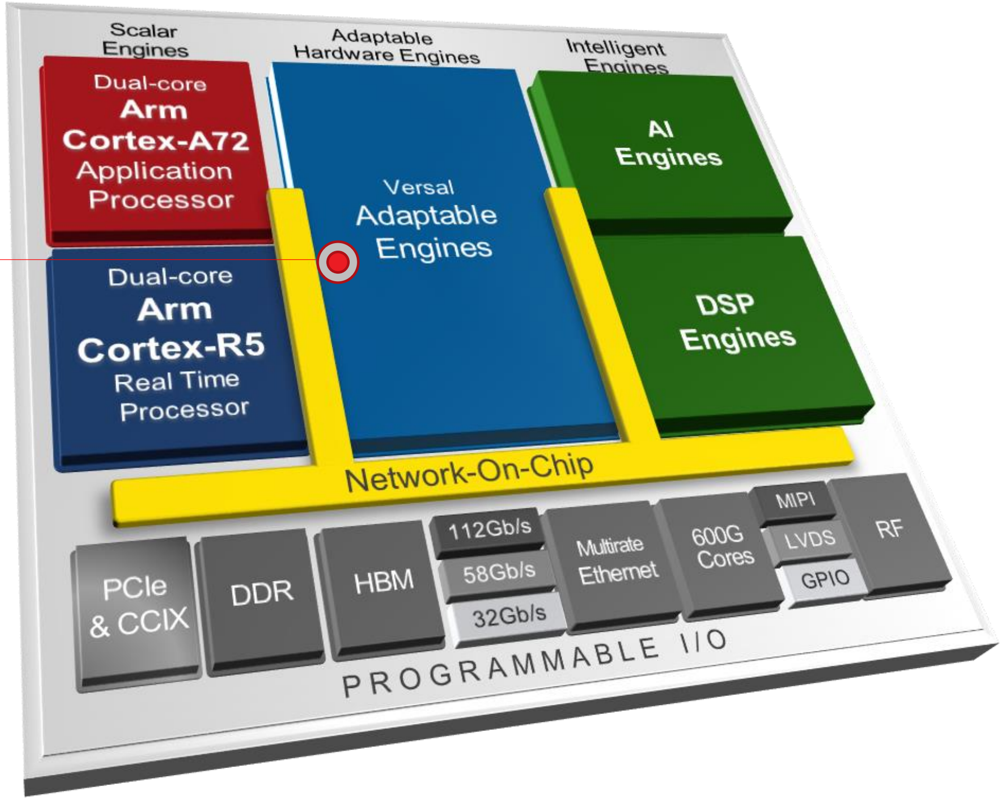
Adaptable Engines

Adaptable Hardware Engines

Programmable logic for fine-grained parallel processing, data aggregation, and sensor fusion

Programmable memory hierarchy to optimize compute efficiency

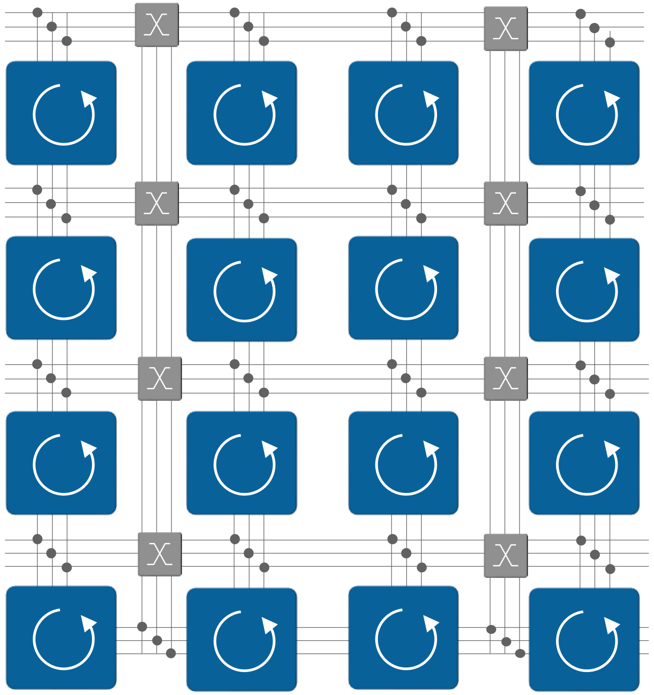
High bandwidth, low latency data movement between engines and I/O



Adaptable Engines:
Greater Compute Density for Any Workload

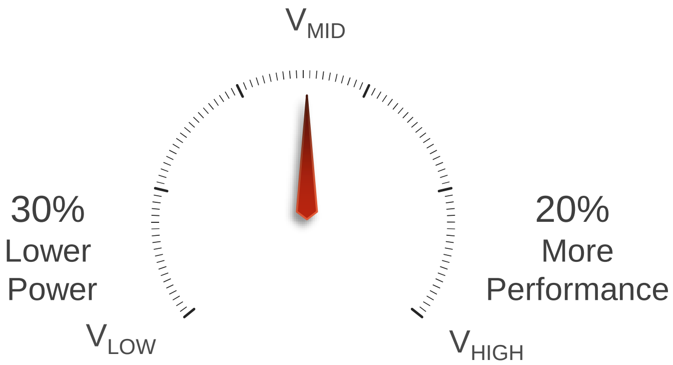
Re-Architected Hardware Fabric

- 4X density per logic block for more compute
- Less external routing→ greater performance
- Code and IP compatible with 16nm devices



Tune for Power & Performance

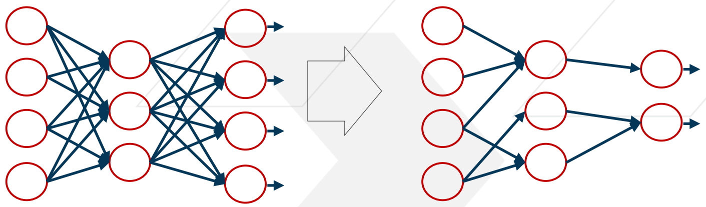
- Three operating voltages to choose from
- Balance power/performance for target app
- Equivalent to 3 speed grades in one device



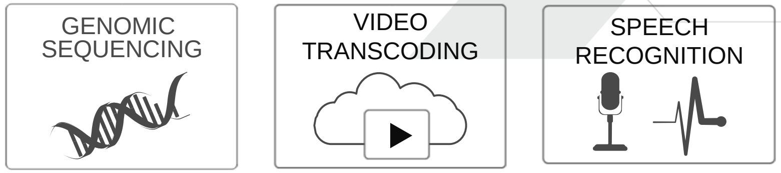
Adaptable to any Workload

- Bit-level precision (1 → 1,000) for any algorithm
- Improves ML efficiency (compression, pruning)
- Forward-compatible to lower precision neural networks, e.g., BNN

ML Inference and Optimizations
(e.g., pruning)

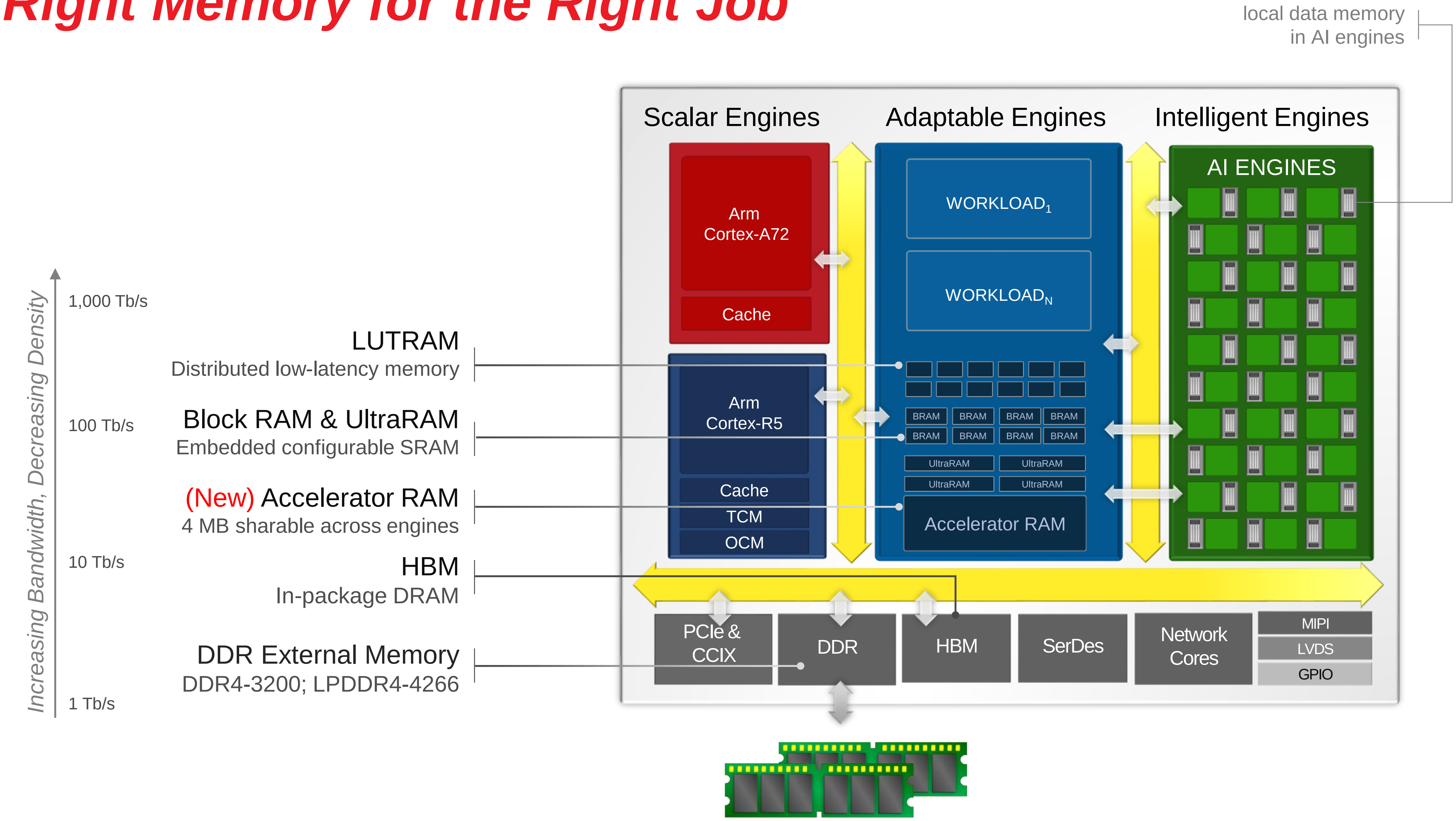


For Any Workload, e.g., …



Adaptable Memory Hierarchy

The Right Memory for the Right Job



Increasing Bandwidth, Decreasing Density

1,000 Tb/s

100 Tb/s

10 Tb/s

1 Tb/s

LUTRAM
Distributed low-latency memory

Block RAM & UltraRAM
Embedded configurable SRAM

(New) Accelerator RAM
4 MB sharable across engines

HBM
In-package DRAM

DDR External Memory
DDR4-3200; LPDDR4-4266

local data memory in AI engines

Intelligent Engines

Intelligent Engines for Diverse Compute

DSP Engines

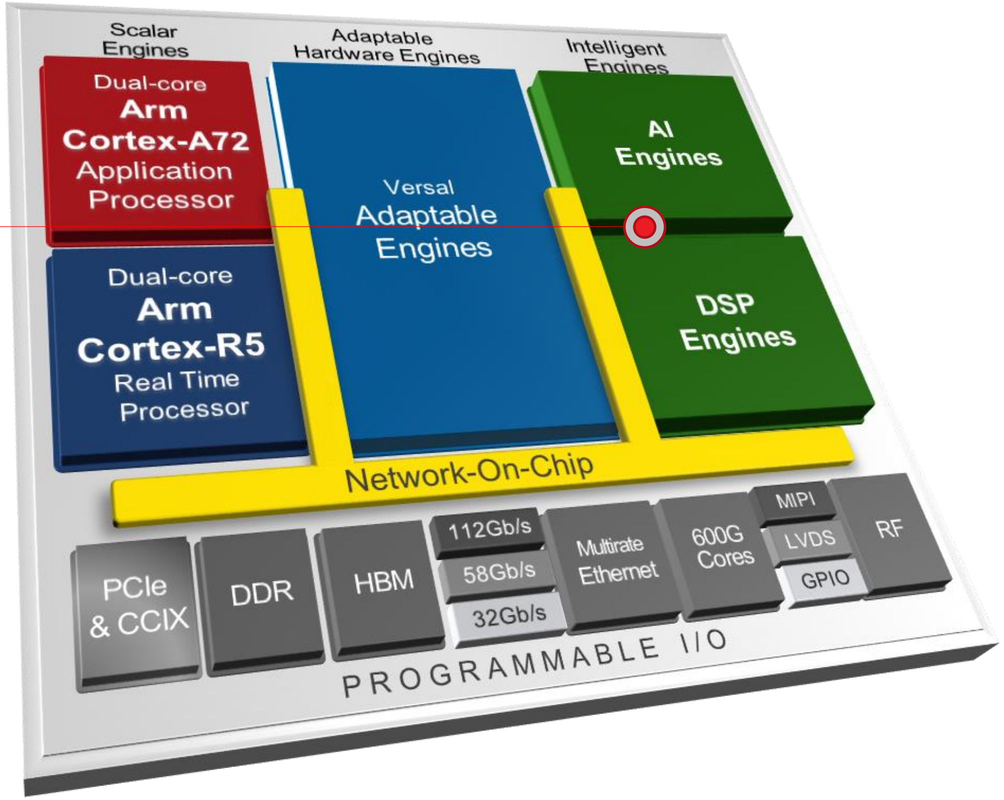
High-precision floating point & low latency

Granular control for customized data paths

AI Engines

High throughput, low latency, and power efficient

Ideal for AI inference and advanced signal processing



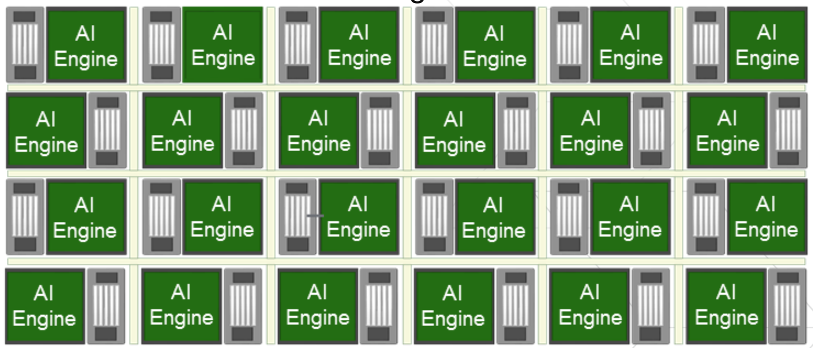
Intelligent Engines:
Digital Signal Processing Capability

Function	DSP48E2	DSP58
DSP Tile/Slice Type	**DSP48E2**	**DSP58**
Multiplier and MACC	**27x18**	**27x24**
32b/16b Single Precision Floating Point Multiply-Add	**Soft**	√
Complex 18b x Complex 18b	**N/A**	**2 x DSP58**
3 x Int8 Dot Product	**N/A**	√

AI Engine 2D Array

VLIW and SIMD Architecture

C/C++ Programmable

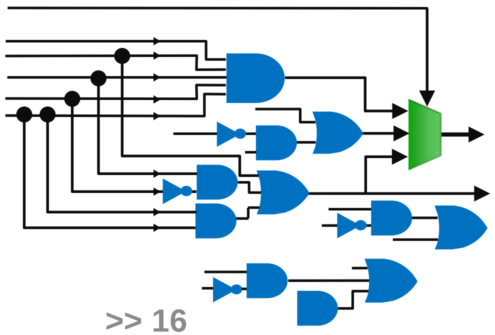


Capability

FPGA Fabric DSP

LUT and Memory

RTL Entry



>> 16

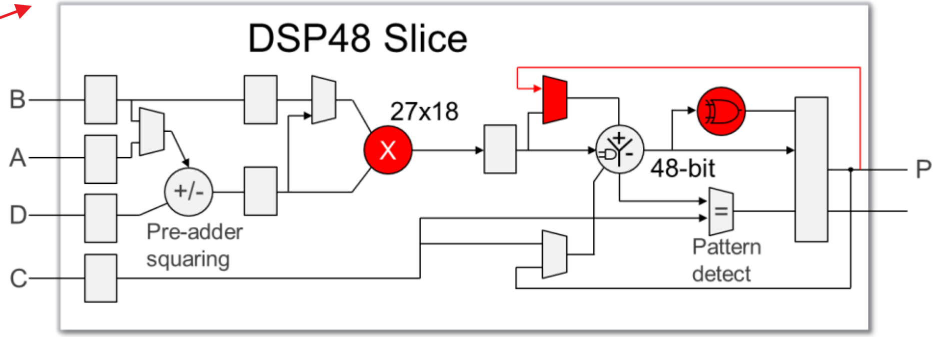
Capability

DSP48E2 Slice

Hardened MULT & ADDERS

ACC = ACC + (A × B)

RTL Entry

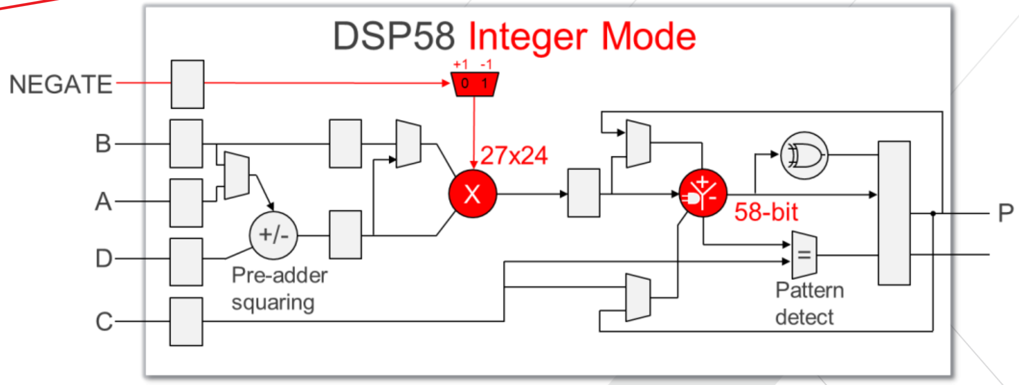


Capability

DSP58

Additional features

RTL Entry



XILINX

NoC for Ease of Use, Guaranteed Bandwidth, and Power Efficiency

High bandwidth terabit network-on-chip

> Memory mapped access to all resources

> Built-in arbitration between engines and memory

High Bandwidth, Low Latency, Low power

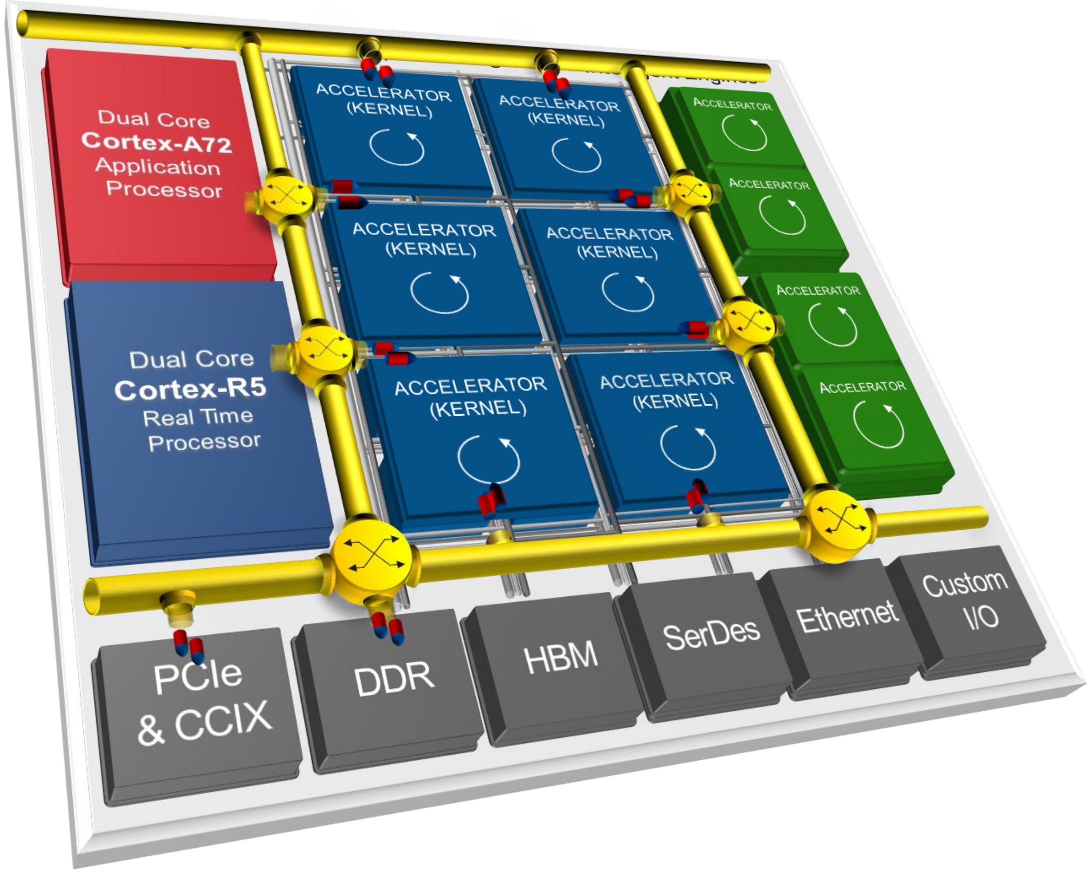
> Guaranteed QoS

> 8X power efficiency vs. FPGA implementations

Eases Kernel Placement

> Easily swap kernels at NoC port boundaries

> Simplifies connectivity between kernels



Introducing the “Integrated Shell”

‘Shell’: Pre-Built Core Infrastructure & System Connectivity

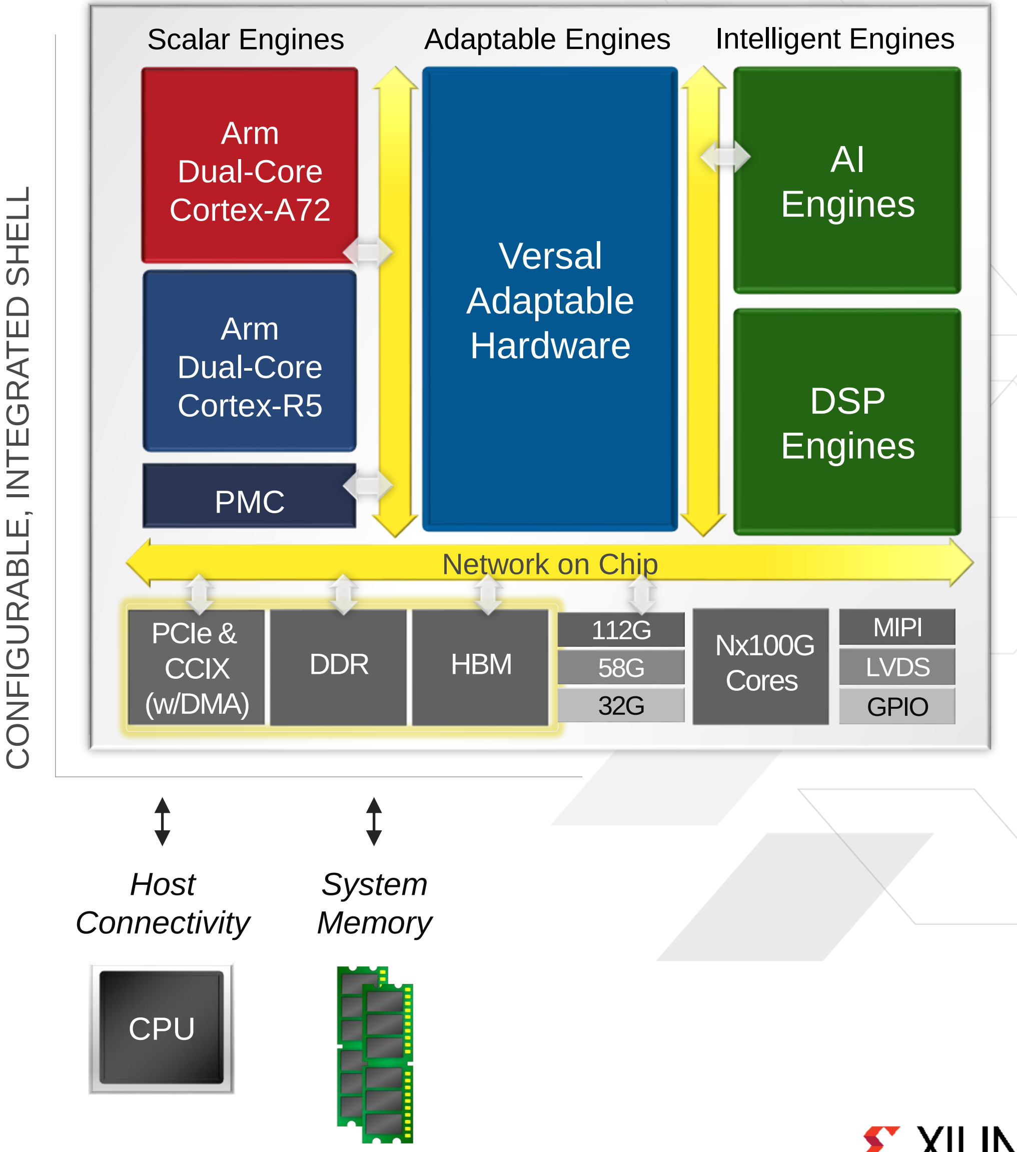
- External host interface
- Memory subsystem
- Basic interfaces (e.g., JTAG, USB, GbE)

Key Architectural Elements of the Shell

- Platform Management Controller (PMC)
- Integrated host interfaces: PCIe & CCIX, DMA
- Scalable Memory Subsystem: DDR4 & LPRDDR4
- Network-on-Chip for connectivity and arbitration

Greater Performance, Device Utilization, and Productivity

- More of the platform available for application’s workload(s)
- Target application runs faster with less device congestion
- Turn-key, pre-engineered timing closure – no debug



AI Engines

AI Engines

Massive AI Inference Throughput and Wireless Compute

1.3GHz VLIW / SIMD vector processors

- Versatile core for ML and other advanced DSP workloads

Massive array of interconnected cores

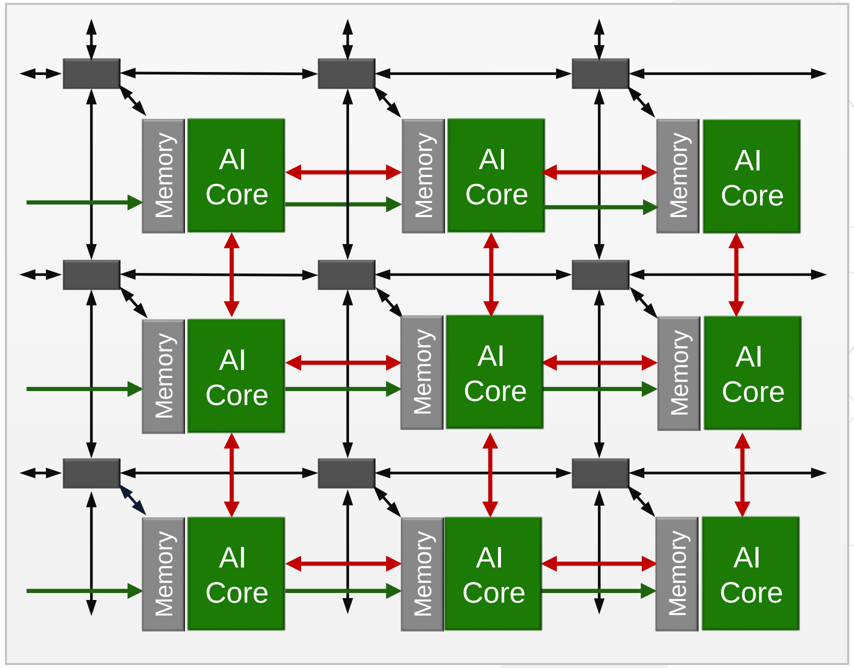
- Instantiate multiple tiles (10s to 100s) for scalable compute

Terabytes/sec of interface bandwidth to other engines

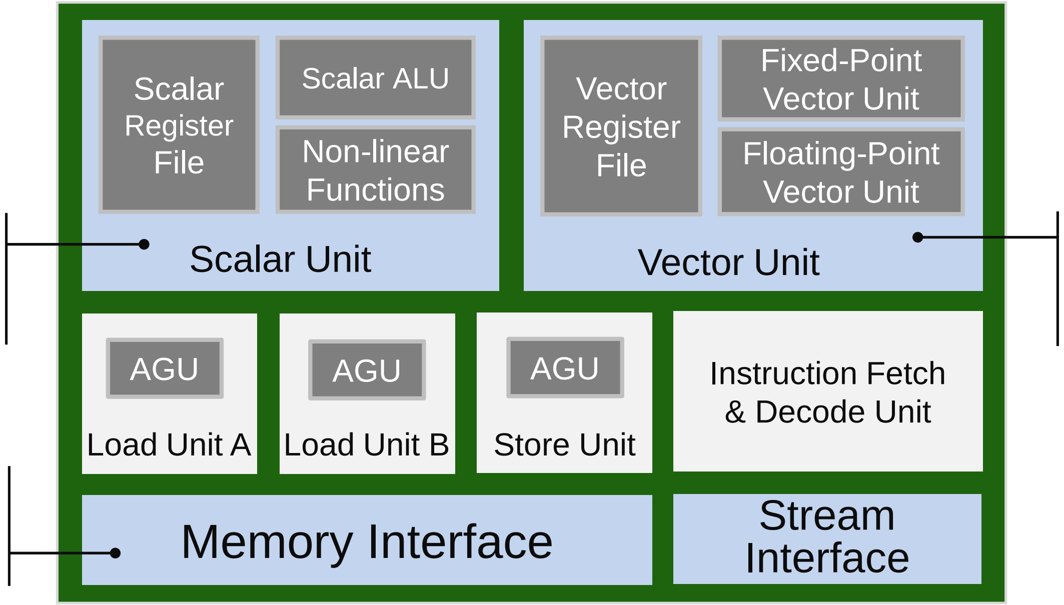
- Direct, massive throughput to adaptable HW engines
- Implement core application with AI for “Whole App Acceleration”

SW programmable for any developer

- C programmable, compile in minutes
- Library-based design for ML framework developers



AI Engine: Scalar Unit, Vector Unit, Load Units and Memory



32-bit Scalar RISC Processor

Vector Processor
512-bit SIMD Datapath

Local, Shareable Memory
- 32KB Local, 128KB Addressable

Instruction Parallelism: VLIW

7+ operations / clock cycle

- 2 Vector Loads / 1 Mult / 1 Store
- 2 Scalar Ops / Stream Access

Highly Parallel

Data Parallelism: SIMD

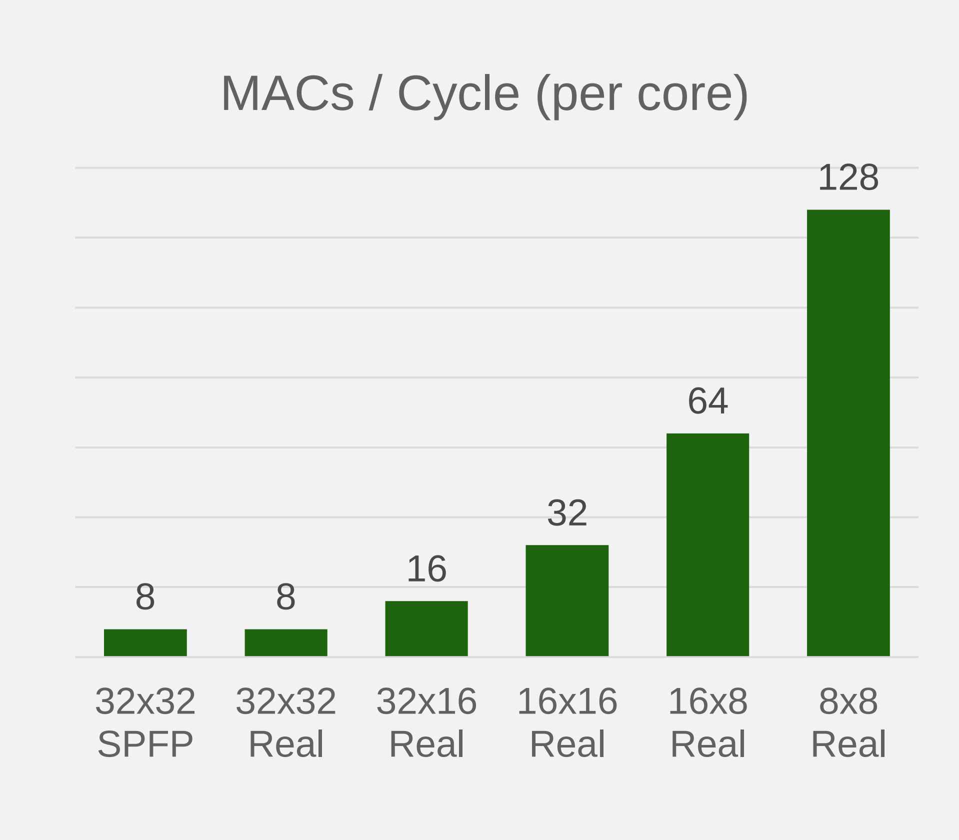
Multiple vector lanes

- Vector Datapath
- 8 / 16 / 32-bit & SPFP operands

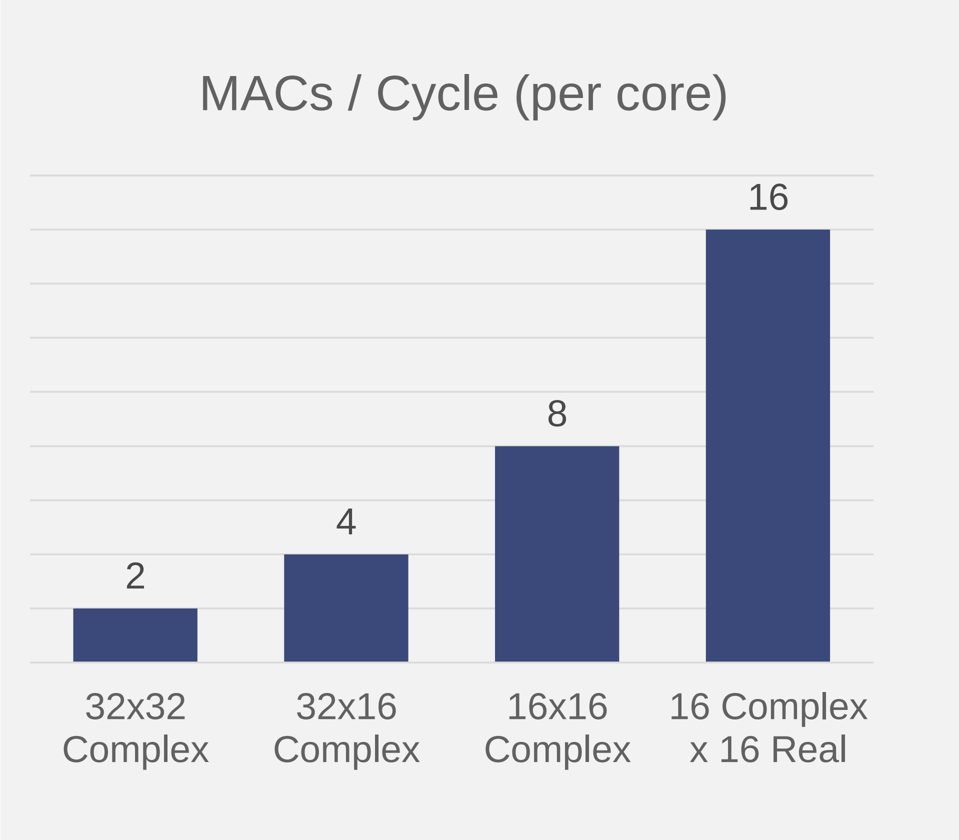
Up to 128 MACs / Clock Cycle per Core (INT 8)

Multi-Precision Support

AI Data Types



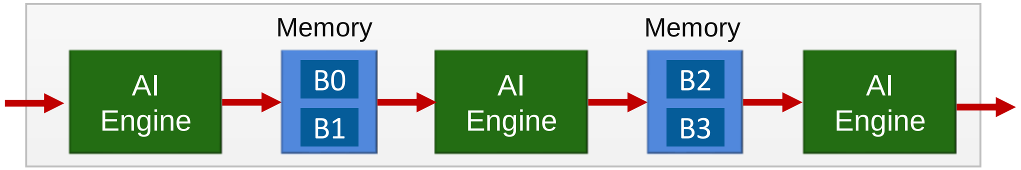
Signal Processing Data Types



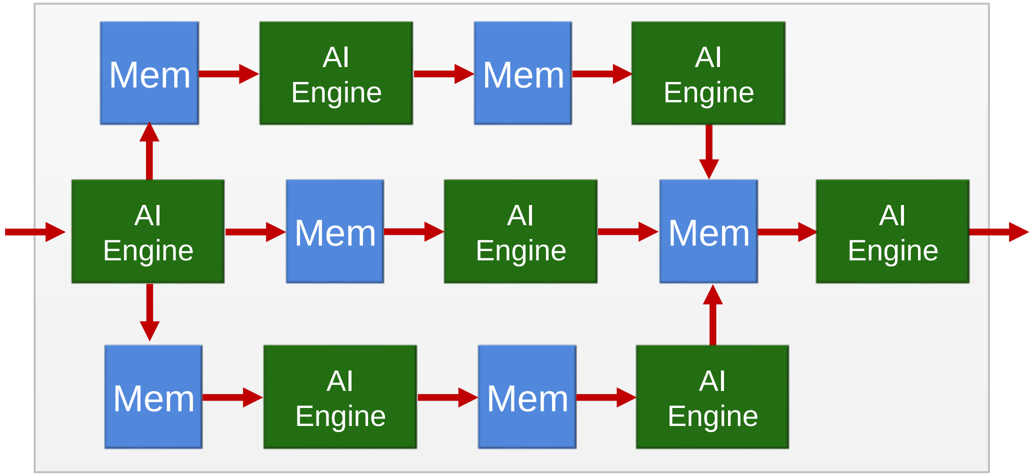
Data Movement Architecture

Memory Communication (neighbor)

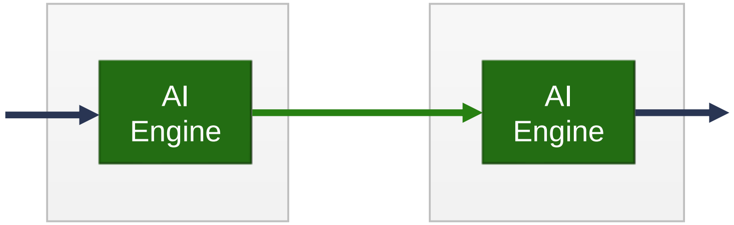
Dataflow Pipeline



Dataflow Graph

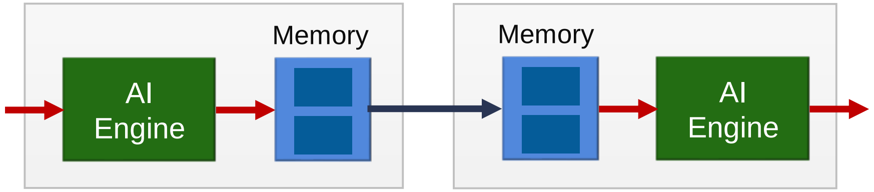


Cascade Streaming

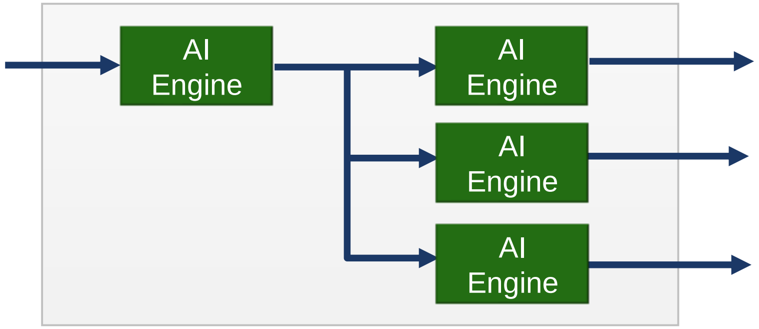


Streaming Communication (non-neighbor)

Non-Neighbor



Streaming Multicast

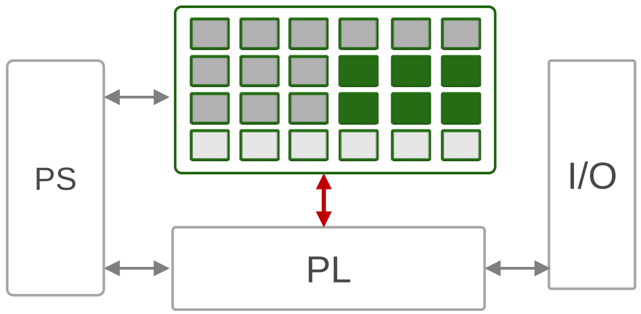


→ Memory Interface

→ Stream Interface

→ Cascade Interface

AI Engine Integration with Versal™ ACAP

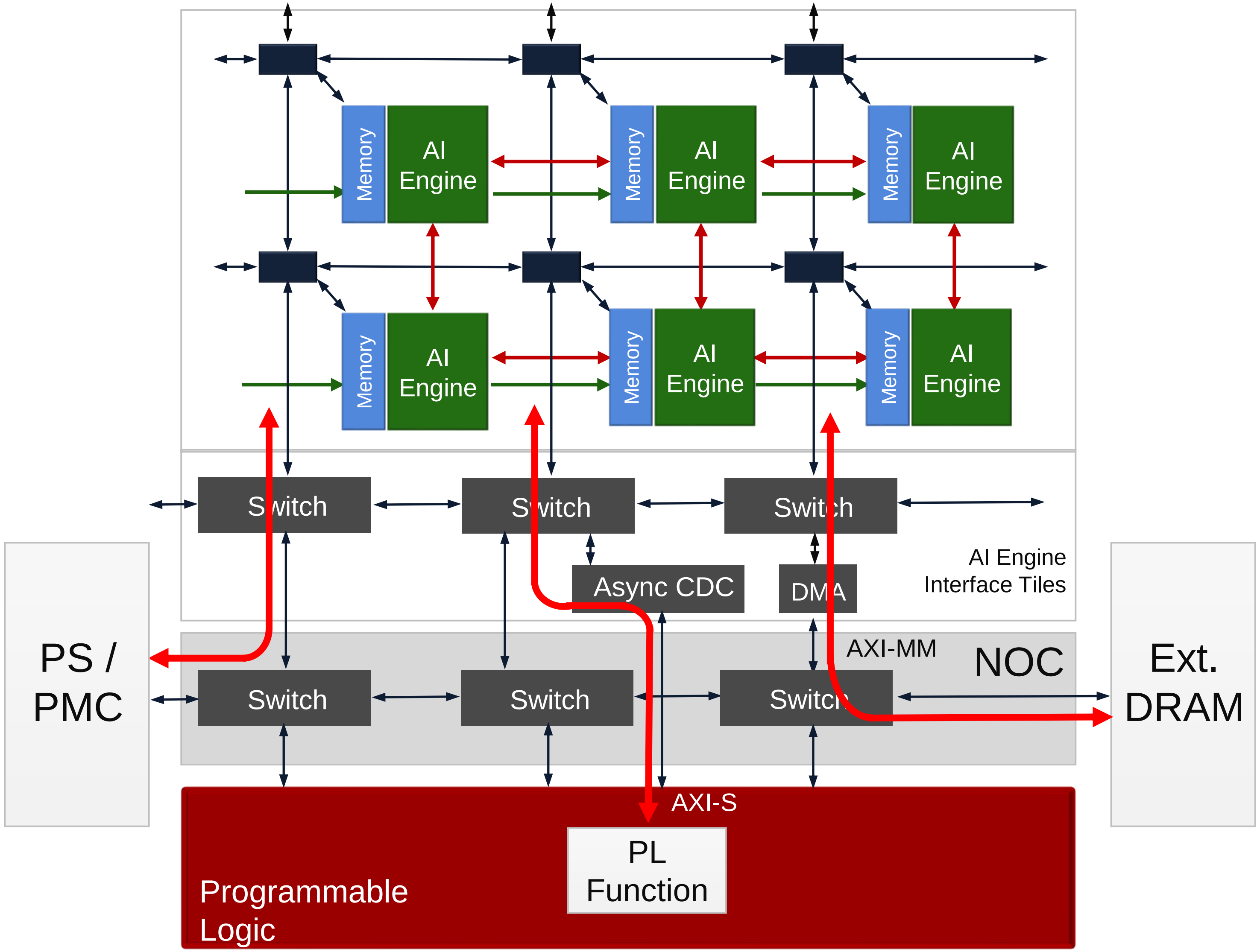


> **TB/s of Interface Bandwidth**

- AI Engine to Programmable Logic
- AI Engine to NOC

> **Leveraging NOC connectivity**

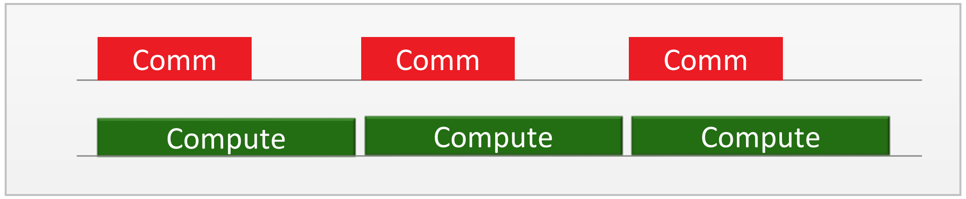
- PS manages Config / Debug / Trace
- AI Engine to DRAM (no PL req'd)



XILINX.

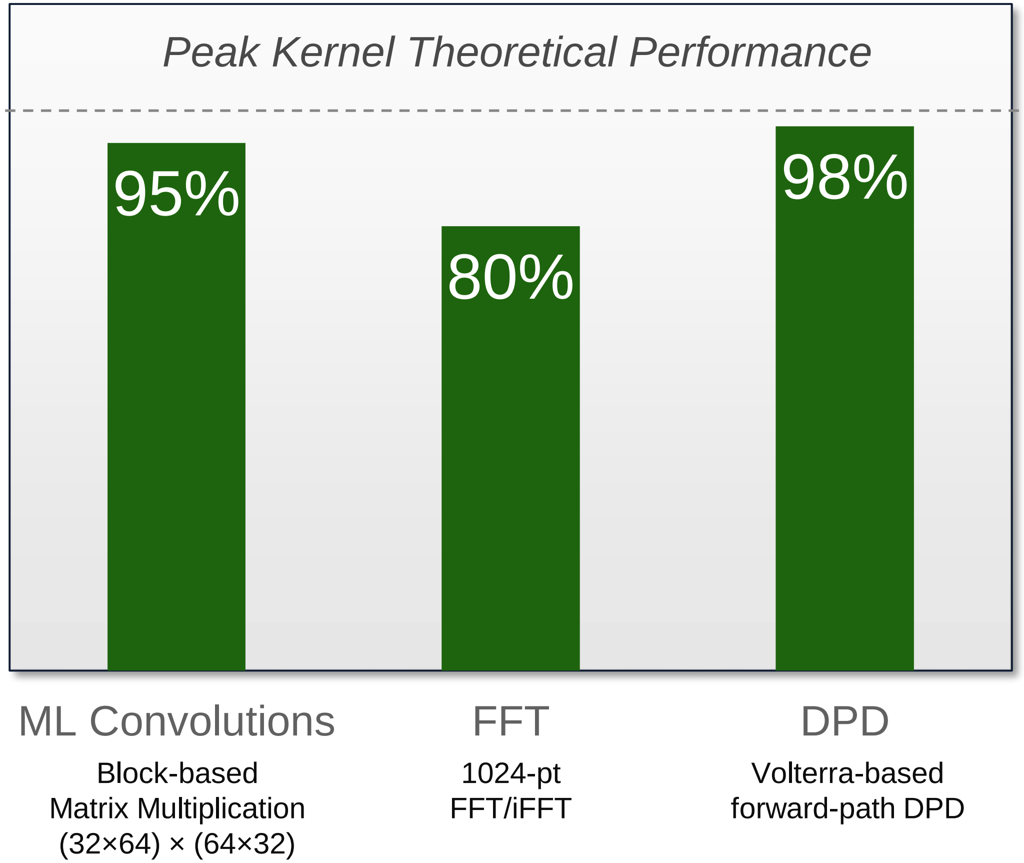
AI Engine Delivers High Compute Efficiency

- Adaptable, non-blocking interconnect
  - Flexible data movement architecture
  - Avoids interconnect “bottlenecks”
- Adaptable memory hierarchy
  - Local, distributed, shareable = extreme bandwidth
  - No cache misses or data replication
  - Extend to PL memory (BRAM, URAM)
- Transfer data while AI Engine Computes

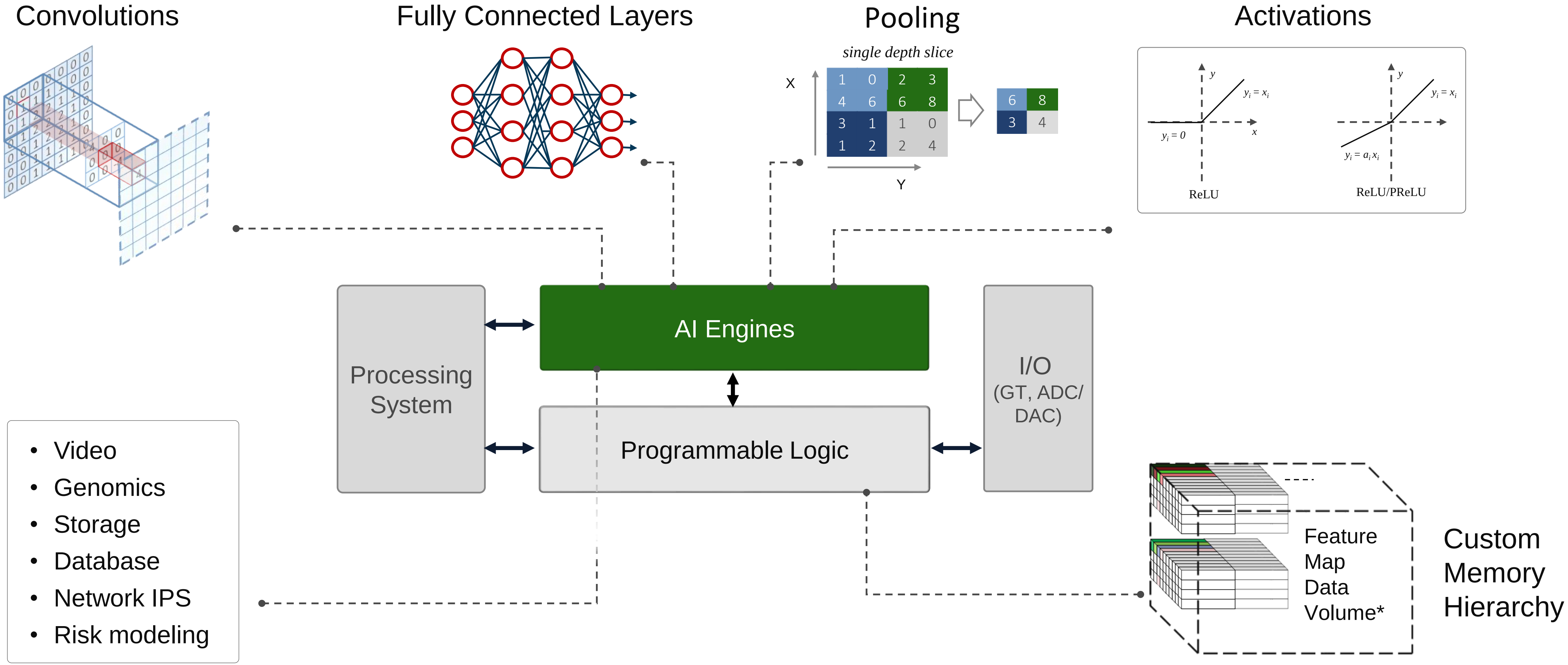


Overlap Compute and Communication

Vector Processor Efficiency



AI Inference on Versal™ ACAP



1	0	2	3
4	6	6	8
3	1	1	0
1	2	2	4

6	8
3	4

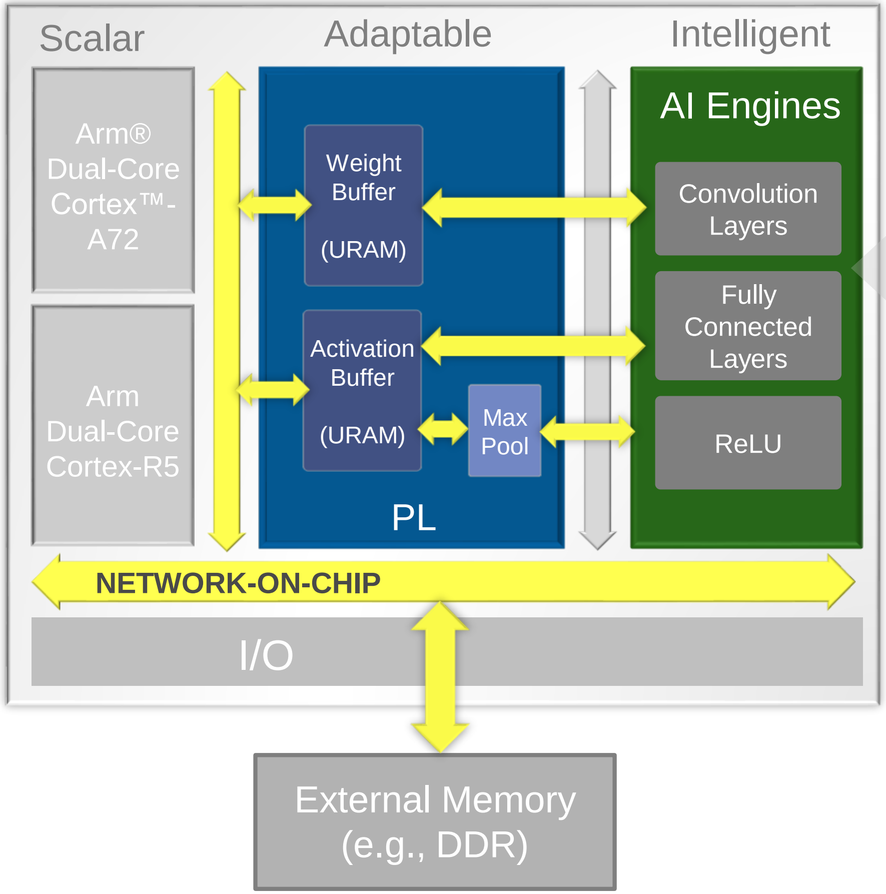
- Video
- Genomics
- Storage
- Database
- Network IPS
- Risk modeling

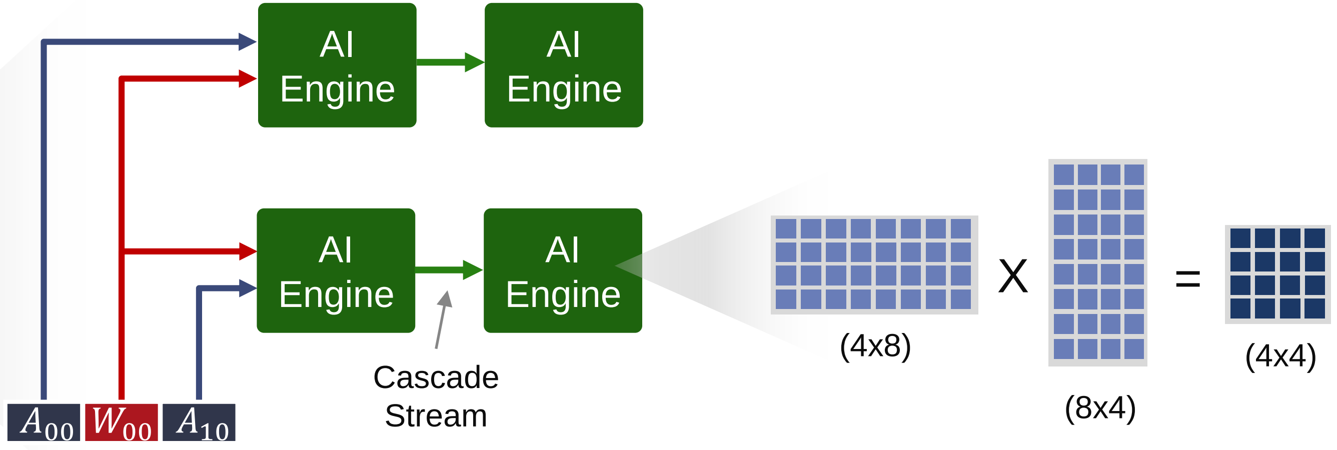
*Figure credit: https://en.wikipedia.org/wiki/Convolutional_neural_network

AI Inference Mapping on Versal™ ACAP

A = Activations
W = Weights

$$\begin{bmatrix} A_{00} & A_{01} \\ A_{10} & A_{11} \end{bmatrix} \times \begin{bmatrix} W_{00} & W_{01} \\ W_{10} & W_{11} \end{bmatrix} = \begin{bmatrix} A_{00} \times \boldsymbol{W_{00}} + A_{01} \times W_{10} & \dots \\ A_{10} \times \boldsymbol{W_{00}} + A_{11} \times W_{10} & \dots \end{bmatrix}$$



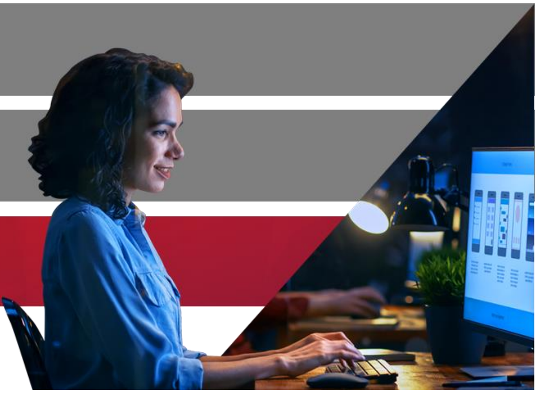


- Custom memory hierarchy
  - Buffer on-chip vs off-chip; Reduce latency and power
- Stream Multi-cast on AI interconnect
  - Weights and Activations
  - Read once: reduce memory bandwidth
- AI-optimized vector instructions (128 INT8 mults/cycle)

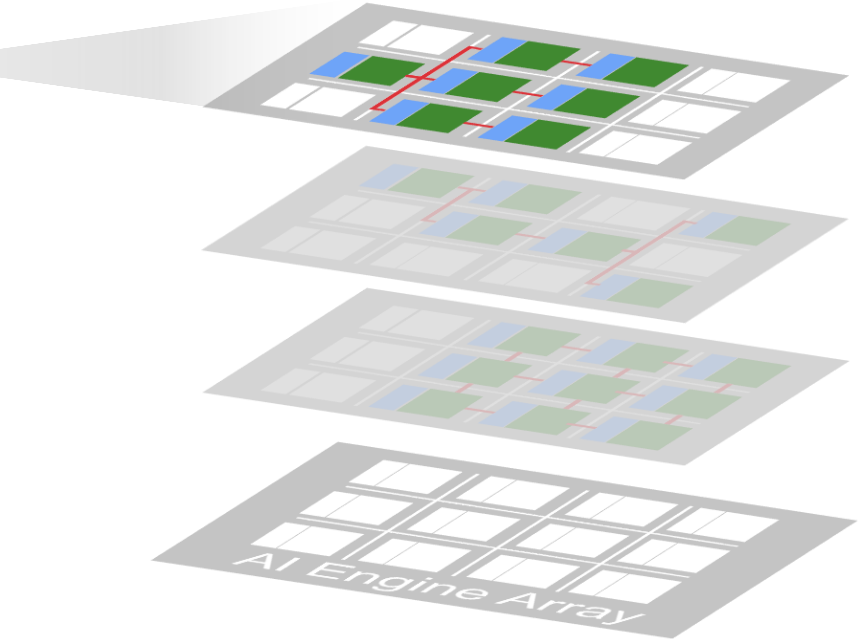
Frameworks for Any Developer

TOOLS	USER	SUPPORTED FRAMEWORKS
Frameworks	AI and Data Scientists	Caffe TensorFlow mxnet FFMPEG
New Unified Software Development Environment	Software Application Developers	
Vivado Design Suite	Hardware Developers	

XILINX VERSAL



Domain Specific Architecture (e.g. AI Inference)



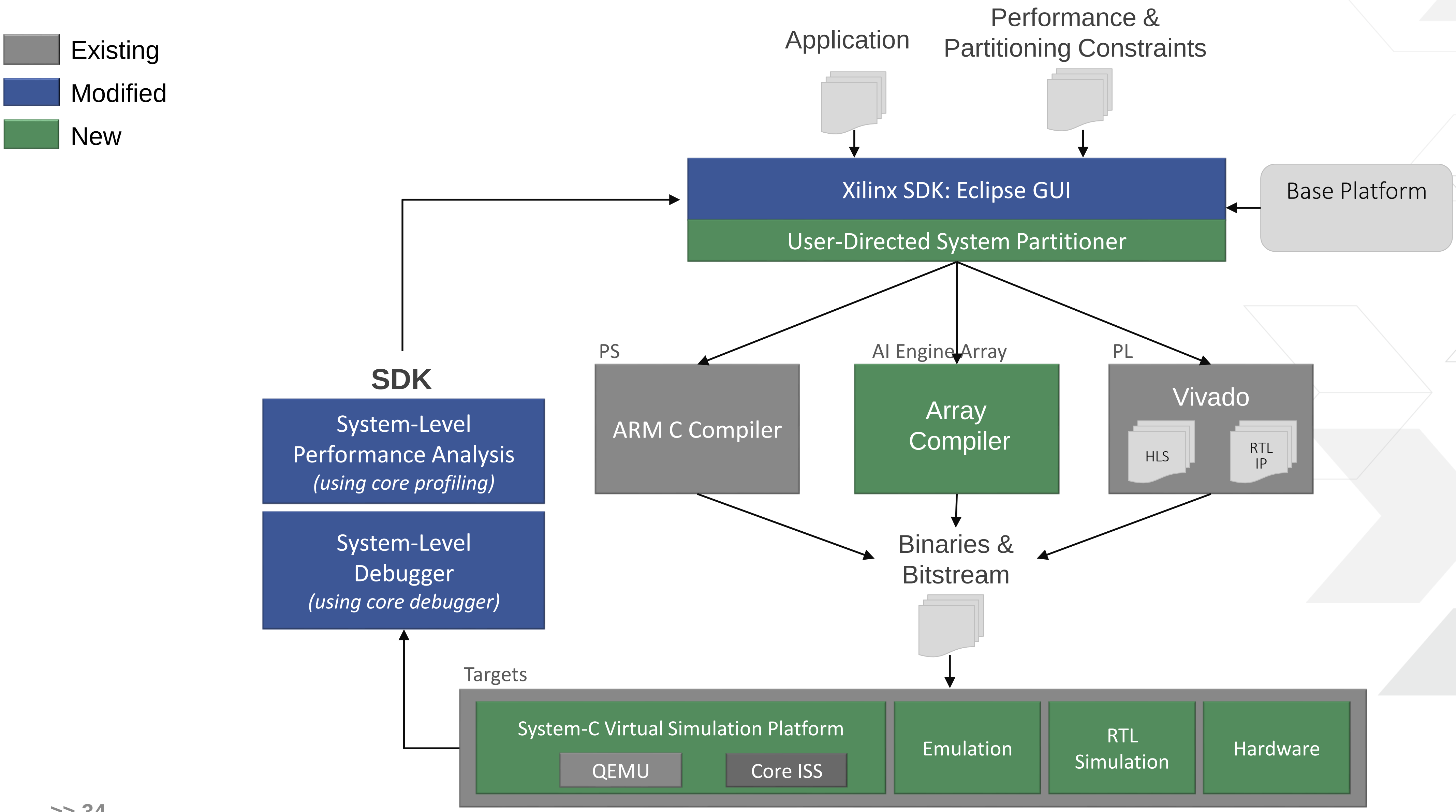
Architecture Overlay

Data Flow
w/ Xilinx libraries

Kernel Program
Data Flow w/ user defined libraries

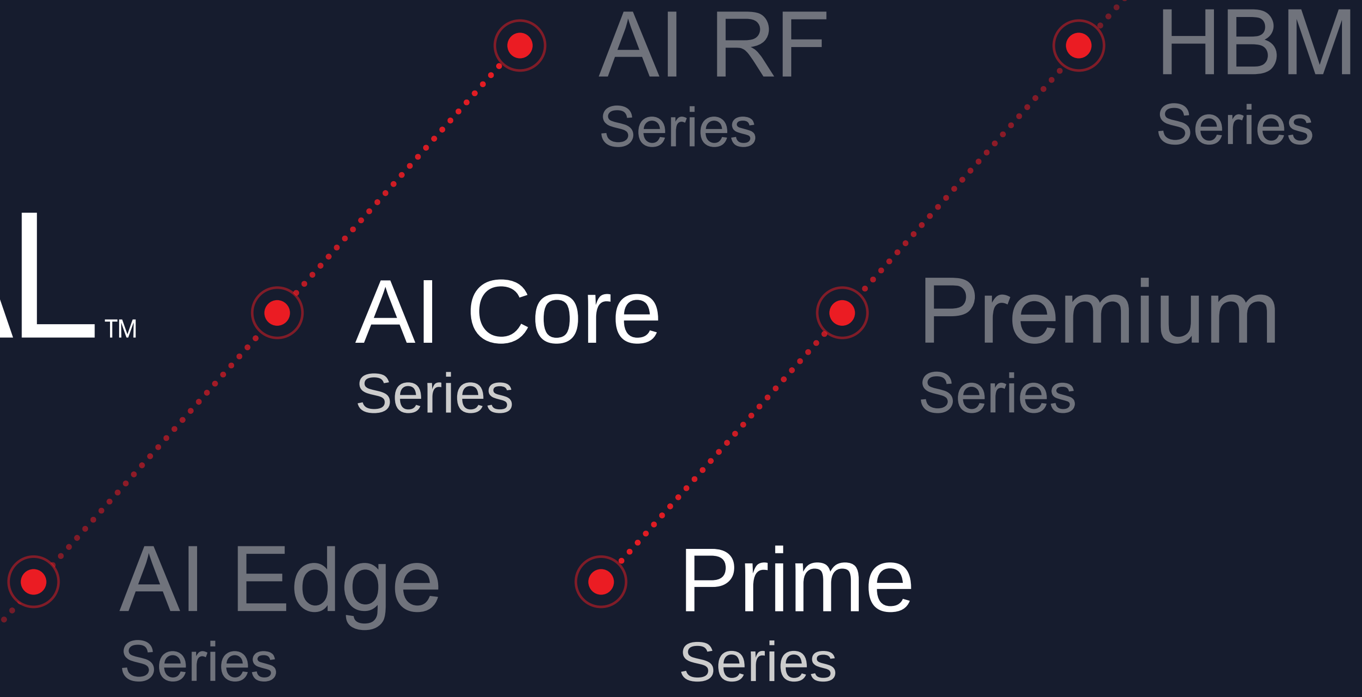
Target Domain Specific Architectures – No HW Design Experience Required

Unified Tool Chain for Device Programming



XILINX®

VERSAL™



AI RF Series

HBM Series

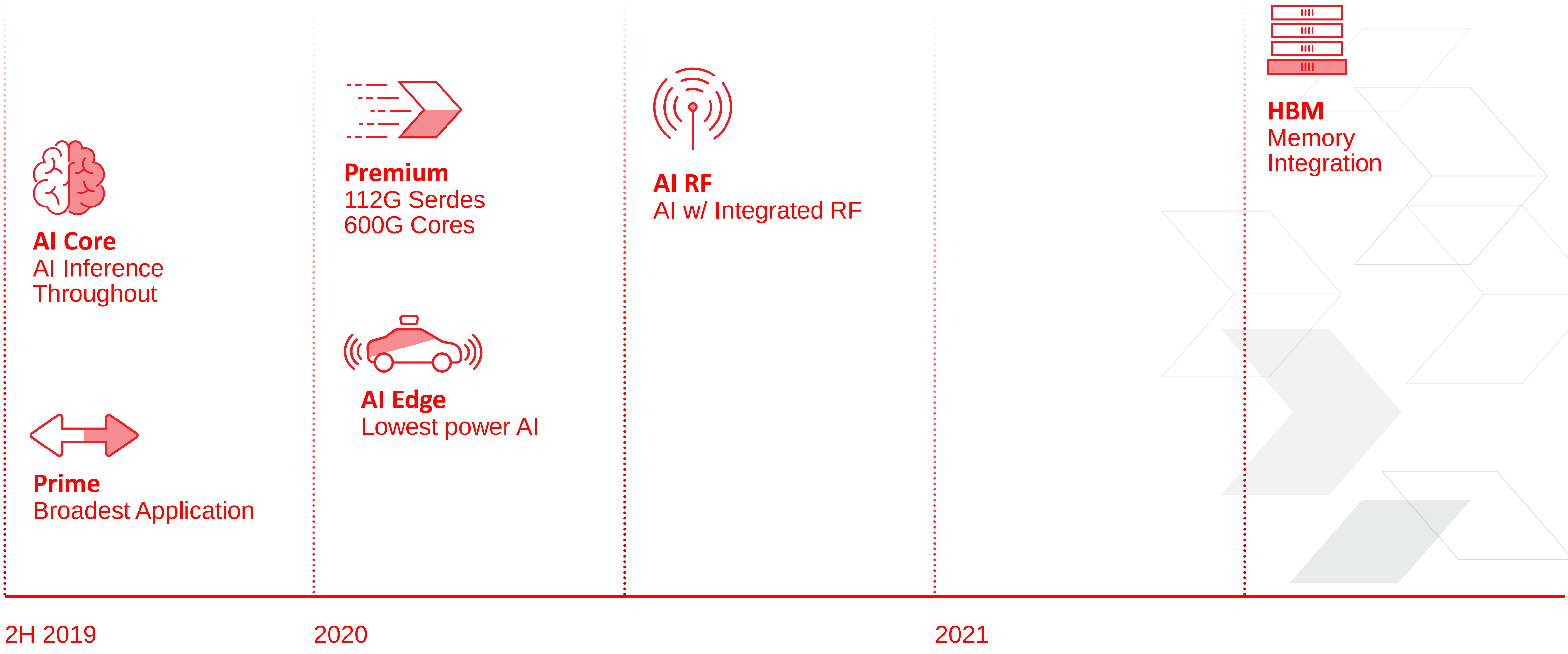
AI Core Series

Premium Series

AI Edge Series

Prime Series

Versal Roadmap



AI Core
AI Inference Throughout

Prime
Broadest Application

Premium
112G Serdes
600G Cores

AI Edge
Lowest power AI

AI RF
AI w/ Integrated RF

HBM
Memory Integration

2H 2019

2020

2021

Getting Started

Visit www.xilinx.com/versal

- > Watch ACAP Intro video
- > Subscribe to mailing list for the latest news

View documentation and resources

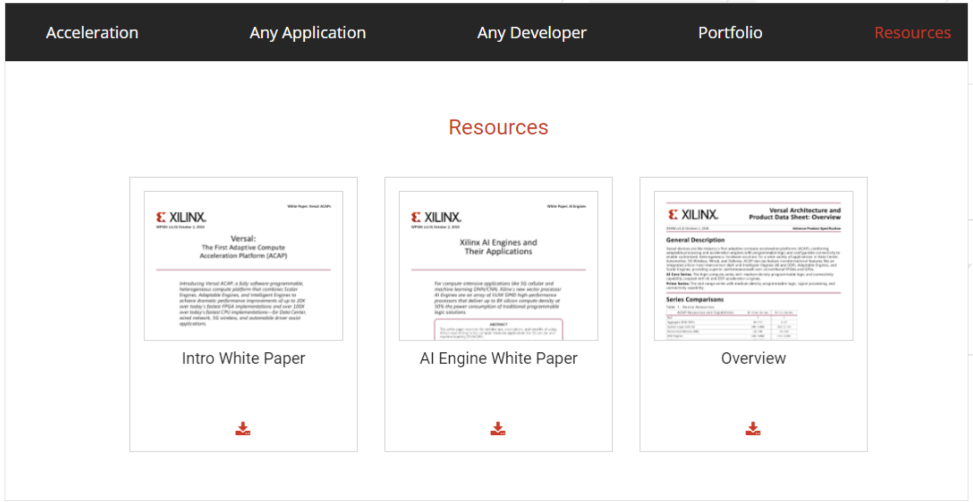
- > Data Sheet Overview
- > Product Tables
- > Versal Architecture and AI Engine White Papers

XILINX VERSAL™

Industry's First ACAP

Adapative Compute Acceleration Platform

Watch Video > Media Kit > Stay Informed >



Acceleration Any Application Any Developer Portfolio Resources

Resources

Intro White Paper

AI Engine White Paper

Overview

Key Take-Aways

Versal: The First ACAP

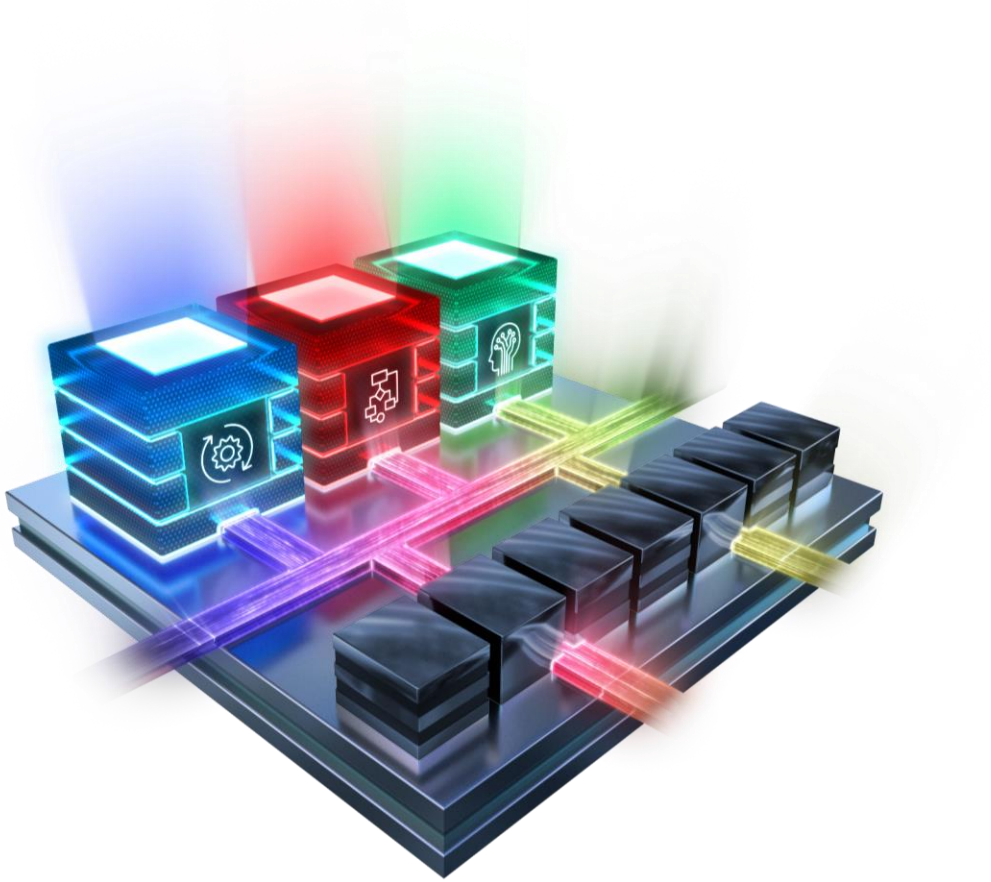
- Heterogeneous Acceleration
- For Any Application
- For Any Developer

Announcing Two Device Series

- Versal Prime Series for Broad Application
- Versal AI Core Series for Highest AI Throughput

Availability

- Early Access Program for SW and tools
- Devices Available 2H 2019



Building the Adaptable, Intelligent World