ANDES TECHNOLOGY

RISC-V® CON

Enabling RISC-V Total Computing Solutions into Future Applications

Wen Wang

Director of Custom Computing Business Unit

Andes Technology

Agenda

- Andes Overview
- NX27V Vector Processor
- 45-Series Superscalar Processors
- ACE Streaming Port
- Summary

AndesCore™ V5 Processor Lineup

	25-Series: Fast & Comp	27-Series: MemBoost	45-Series: Superscalar
Application Processing with Multicore & SMP Linux	AX25MP A25MP		AX45MP A45MP
Application Processing with Single-core & Linux	AX25 A25	AX27/AX27L2 A27/A27L2	AX45 A45
Data Processing with DSP or Vector	D25F	NX27V	D45
Embedded Control with Integrated FPU	NX25F N25F		NX45 N45

First RISC-V Vector Core

First RISC-V DSP-Capable Cores

N22 2-stage (700 MHz)

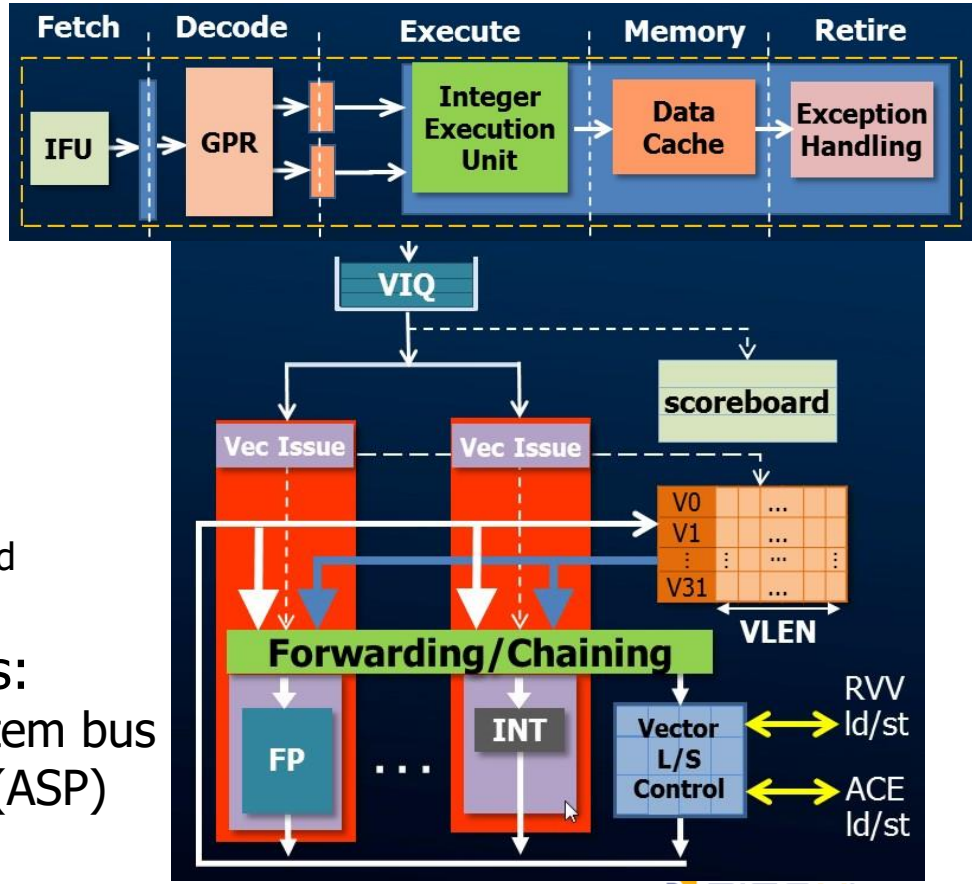
5-stage: 1.1 GHz Coremark/MHz: 3.53 → **+70%** **8-stage:** 1.2 GHz Coremark/MHz: 5.50

Notes: 1. Core naming: with **"X" is 64-bit** (e.g. NX25F) and no "X" is 32-bit (e.g. N25F)
2. V5's common features include RV-IMACN, Caches, LM, ECC/parity, Branch Prediction, CoDense™, PowerBrake, StackSafe™, **ACE (Andes Custom Extension™)**; Frequencies is the worst case at **28nm**.

AndesCore™ NX27V Vector Processor

AndesCore™ NX27V: Overview

- 1st RISC-V commercial vector core
- Support latest spec v0.10 draft
- RVV data formats:
  - Standard: int8~int64, fp16~fp64
  - Andes-extended: bfloat16 and int4
- An efficient scalar unit
- A powerful Vector Unit (VPU):
  - RVV starts execution after retired
  - Multiple functional units
    - parallel, chainable and most fully pipelined
  - VLEN & SIMD width: 128, 256, 512
- Independent memory access paths:
  - RVV load/store thru dcache and system bus
  - ACE load/store thru Streaming Port (ASP)

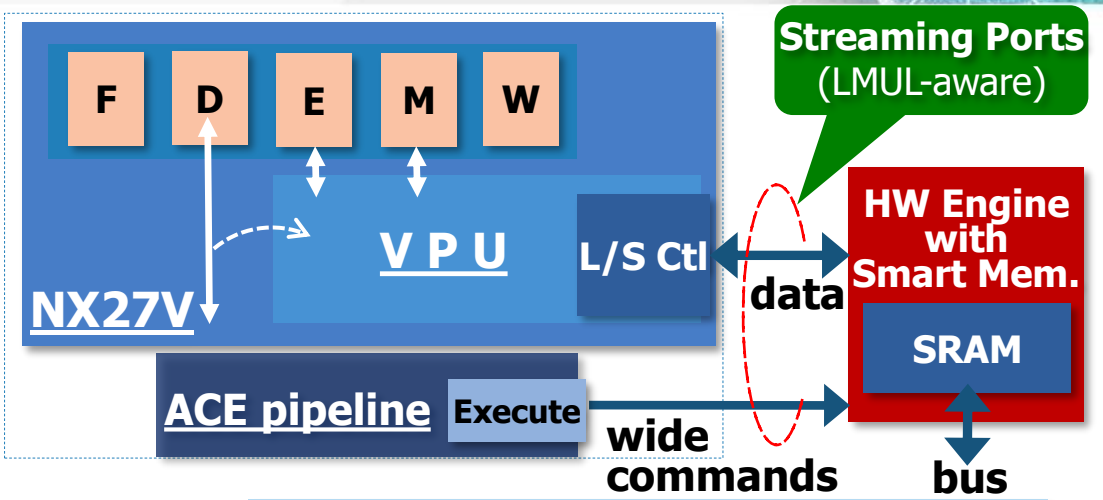


NX27V: Vector Processor Unit

- Powerful vector engine for vector arithmetic instructions
  - overlapped with subsequent independent instructions, and
  - chained to dependent vector instructions.
- Optimization tips:
  - Move data to close to the processor (e.g. thru **DMA**) and **re-arrange the data layout** on the fly for the efficient accesses
  - **Fuse loops** to reduce load/store instructions when possible
    - E.g. loop(1..n) C= A+B;
loop(1..n) E= C*D;
} loop (1..n) E= (A+B)*D;
- First RISC-V commercial vector processor: 5 customer projects now

NX27V: ACE Streaming Port (ASP)

- Common SoC scenarios:
  - DMA-equipped HW engine accelerates structured computations (e.g. CNN)
  - **Efficient & programmable** pre/post-processing is needed → algorithm differentiation

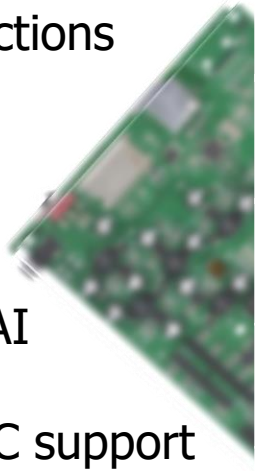


- Sol: Connect HW engine to NX27V ASP
  - ACE load/store instructions thru ASP to/from V/F/X registers
  - Custom-defined addressing modes
    - e.g. address auto-increment/wrap-around
- Benefits: efficient and programmable

```
insn svload {
  operand= {out vr data, io addrCtl addr,
                imm2 mode, . . . };
  csr_op= {vl};
  streaming_port= load;
  csim= . . .
  . . .
};
```

RVV Tools and Performance

- Standard tools in AndeSight™
  - AndeSim™: Cycle simulator
  - Compiler: intrinsic functions
  - Assembler
  - Debugger and ICE
  - Computation libraries
- Advanced tools:
  - **LLVM** IR support for AI compilers
  - **AndeSysC**™: SystemC support for AndeSim™
  - **AndesClarity**™: GUI-based pipeline visualizer and analyzer



RVV Functions	NX27V Speedup[1]
F32 basic function	**23x**
F32 32x32 GEMM	**43x**
Q7 CNN HWC[2] (33,33,51)	**35x**
Q7 Relu CNN	**81x**

Note 1: Compared to pure C scalar code compiled with high optimization; both vector and scalar code ran on the NX27V FPGA with 512/256-bit VLEN/SIMD, 256-bit bus.
Note 2: HWC(Height, Width, Channel)

NX27V Performance Data

Features	NX27V (RVV 0.10)		
VLEN/SIMD	256/256	512/256	512/512
ELEN (bit)	32 (int+fp)		
Max Frequency (worst case)[1]	All frequencies >=1.2G		
Gate Count (gates)[1]	From 1.6M to 2.6M		
Dynamic Power (uW/MHz)[1]	< 17		
F32 GEMM 32X32 (cycle)	Around 6,200	Around 4,800	Around 3,400

Note 1: TSMC 7nm **mixed VT** 240H library, **V-extension, 128-entry BTB, 16-entry PMP/PMA and 32KB I/D$, AXI BUS** with I/O constraint, power are core only. Frequency condition: 0.675v/-40°c, SS; Dynamic power condition: 0.75v/25°c, TT, Dhrystone program, logic synthesis

AndesCore™ 45-Series Superscalar Processor

AndesCore™ 45-Series Lineup

NX45
64bit Baseline
RV64 GCN

N45
32bit Baseline
RV32 GCN

High Performance Embedded System

D45
Baseline + P-ext* (DSP/SIMD)

High Performance with DSP/SIMD

AX45
RV64GCNP + MMU

AX45MP
L2 Cache Coherence

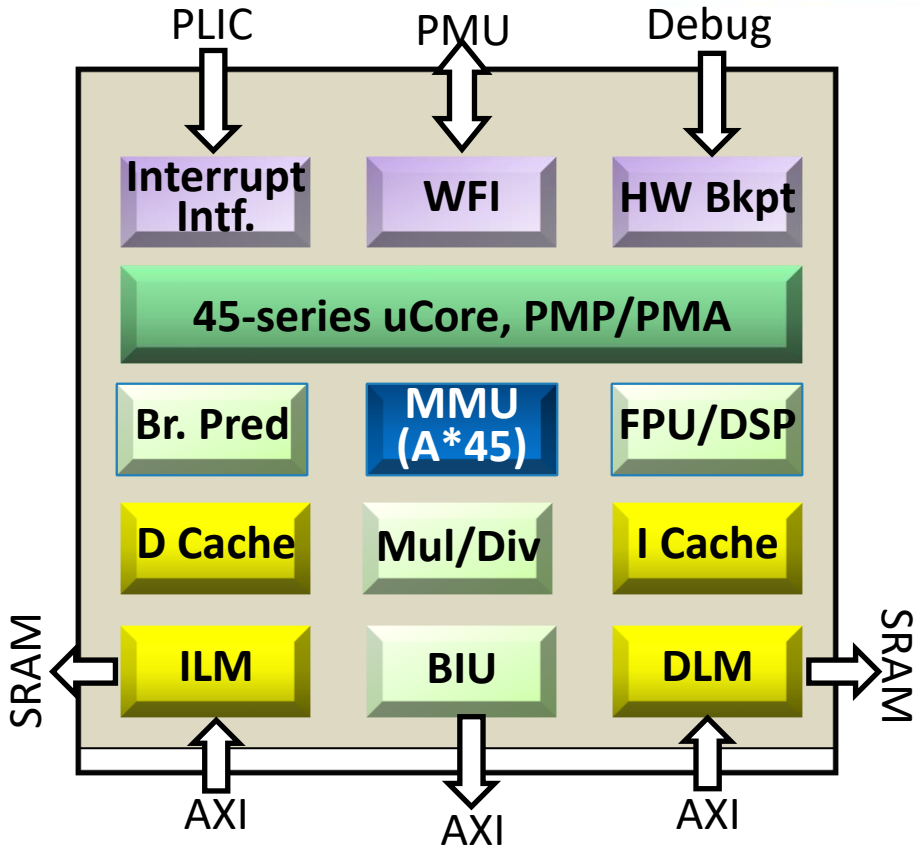
A45
RV32GCNP +MMU

A45MP
L2 Cache Coherence

Application Processor, single or multicore Linux capable

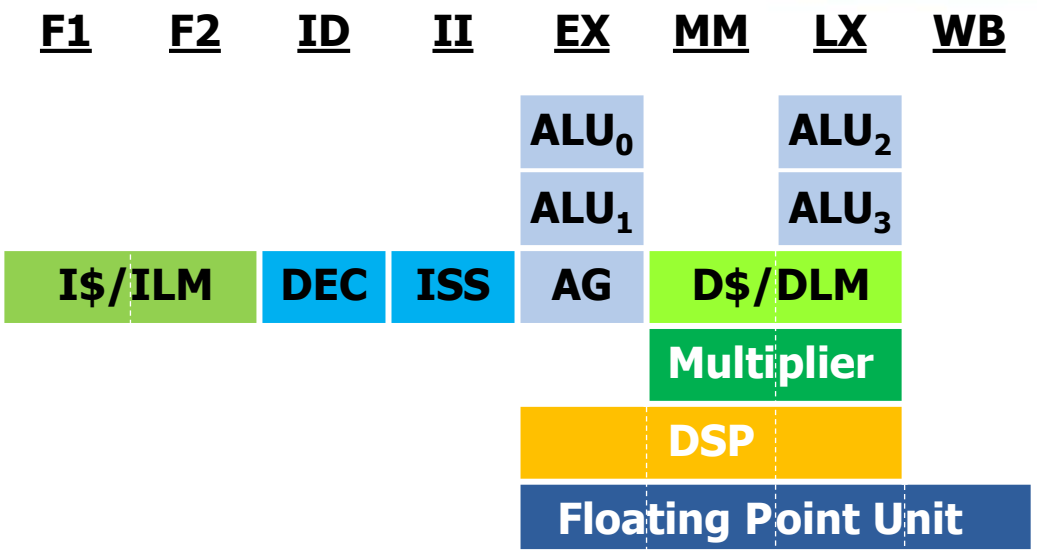
AndesCore™ 45-Series Overview

- **8-stage In-Order Dual-Issue**
- **AndeStar™ V5 ISA:**
  - RV*GCN (S/D FPU): All Series
  - RV*P-ext (DSP/SIMD): D45/A(X)45
  - MMU for Linux Applications: A(X)45
  - Andes extensions: All Series
- **L1 I/D Caches:**
  - 8KB up to 64KB w/ 64B cache line size
  - Direct, 2-way or 4-way set associative
  - Instruction and data cache lock
  - Optional Parity or ECC error protection
- **I/D Local Memory**
  - 4KB up to 16MB
  - SRAM interface support
  - Slave port accesses from bus masters
  - Optional ECC error protection

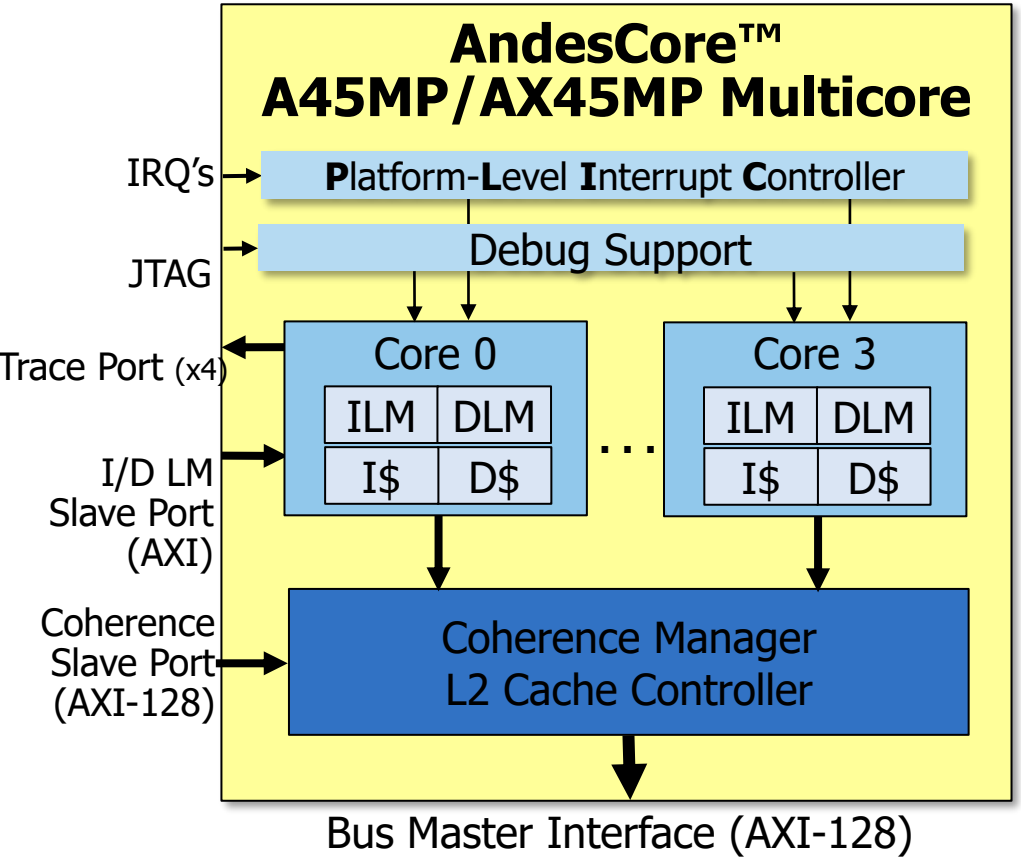


AndesCore™ 45-Series Overview

- **Dual-issue most instruction pairs**
  - Except for 2 MUL/FPU/DSP/LD_ST and some special dependent ALU instruction pairs
  - Late ALUs enable 0-cycle load-use penalty
- **MemBoost support**
- **Low Power Dynamic Branch Prediction**
- **Unaligned data accesses**
- **Fast or Small Multiplier**
- **Physical memory support**
  - Up to 16 region for **PMP** and **PMA**
- **Virtual Memory Support**
- **Privilege modes**
  - Machine/User mode
  - Optional Supervisor mode

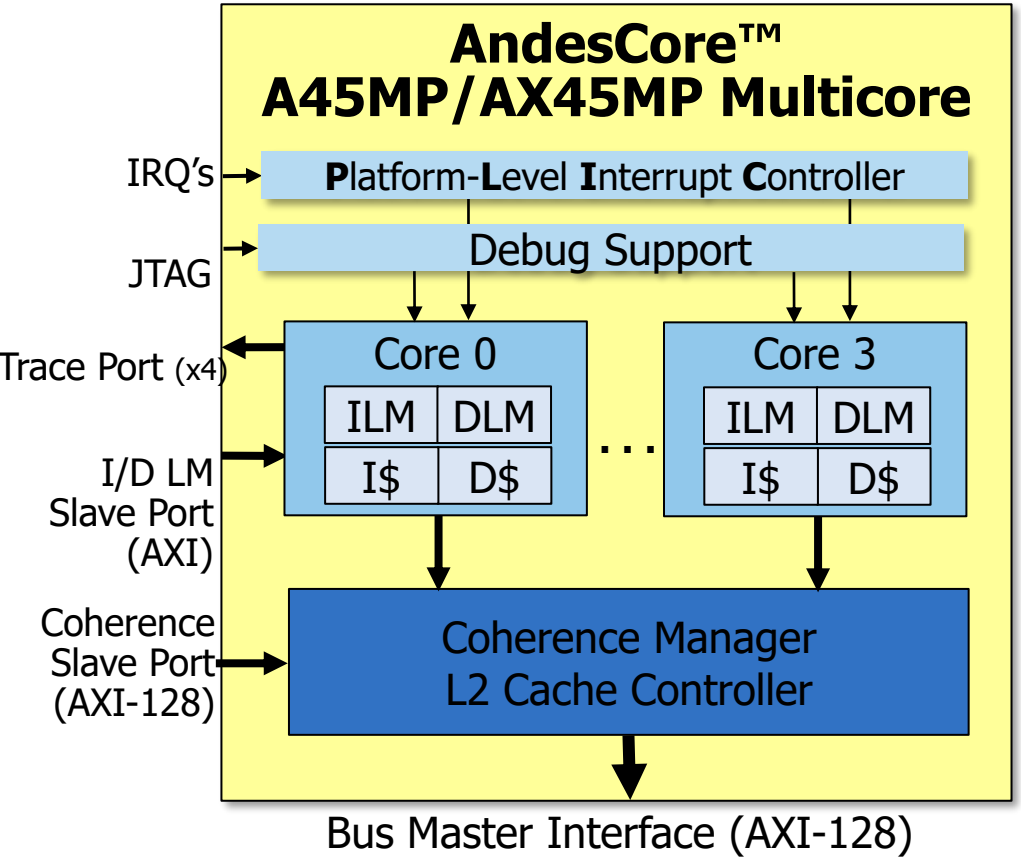


A(X)45MP: Cache-Coherent Multicore



- **1~4 A45/AX45 CPUs:**
  - RV-GCNP ISA + V5 extensions
  - Supporting SMP Linux
- **Bus Interfaces**
  - LM slave port
  - Coherence slave port
  - AXI bus master interface
    - N:1 synchronous clock ratio
- **PLIC for interrupt handling**
- **Debug/trace support**

A(X)45MP: Cache-Coherent Multicore



- **Coherence Manager**
  - MESI cache coherence protocol
  - IO coherence for cacheless masters
- **L2$ Controller (optional)**
  - Size: 128KB up to 2MB
  - Line size: 64B
  - 16-way with pseudo random replacement and writeback
  - Prefetching support
    - Instruction: 1/2/3 lines after a miss
    - Data: 2/4/8 lines after consecutive linear misses (tracking 8 address sequences)
  - Linux-capable configuration
    - Sv32 for 32-bit and Sv39/Sv48 for 64-bit
    - Shared TLB: 32~512 entries
    - ITLB/DTLB: 4 or 8 entries

45-Series Performance Data

Features	N45	D45	A45	NX45	AX45
AndeStar™ ISA version	RV32GCN	RV32GCNP	RV32GCNP	RV64GCN	RV64GCNP
Configurations	256-entry BTB, PMP & PMA 16-entry, MemBoost, 32KB I/D$				
MMU/PA bits	N/A		Sv32:34	N/A	SV39:38; Sv48:47
CoreMark/MHz [1]	>5.6			>5.5	
DMIPS/MHz (no-inline)[1]	>2.96			>3.1	
Max. Frequency (Hz), worst[2]	> 1.8G				
Core Area with scan (mm^2)[2]	From 0.024 to 0.05				
Dynamic Power (uW/MHz)[2]	<10				

Note 1: BSP v5.1.0 mculib-v5 compiler, following Dhrystone's no-inline ground rules
Note 2: TSMC 7nm FIN FET ULVT/LVT/SVT, High Speed L1 Cache Memory Compiler. Frequency condition: worst: SSGNP/0.675V/-40°c, typical: TT/0.75v/+85°c. Power and area: typical corner, with I/O constraint, die area and power are core only, 75% utilization

A(X)45MP Performance Data

Features	A45MP	AX45MP
Virtual Memory	Sv32	Sv39
Configurations	IO Coherence Port, Fast multiplier, 256-entry BTB, PMP and PMA	
	32KB I/D$, MemBoost, D$ outstanding: 8; L2$: 1MB	
	DSP, DP FPU, MMU w/128 TLB-entry, M+U+S modes	
SPECint2K	0.31	0.32
SPECint2K6		3.15
Max. Freq (Hz), worst[1]	>1.58G	
Core gate Count (K gates)	480	643
L2 Gate Count (K gates)	Around 600	
Cluster Floorplan Area (mm^2)[2]	From 0.089 to 0.206	From 0.102 to 0.258

Note1: Library: tcbn07_bwph240l8p57pd_base_multivtssgnp_0p675v_m40c_cworst_CCworst_T;SRAM Library: 7nm Fin FET High Speed L1 Cache Memory Compiler, 75% utilization
Note 2: Area from 1-core to 4-cores

45-Sereis Performance Enhancement

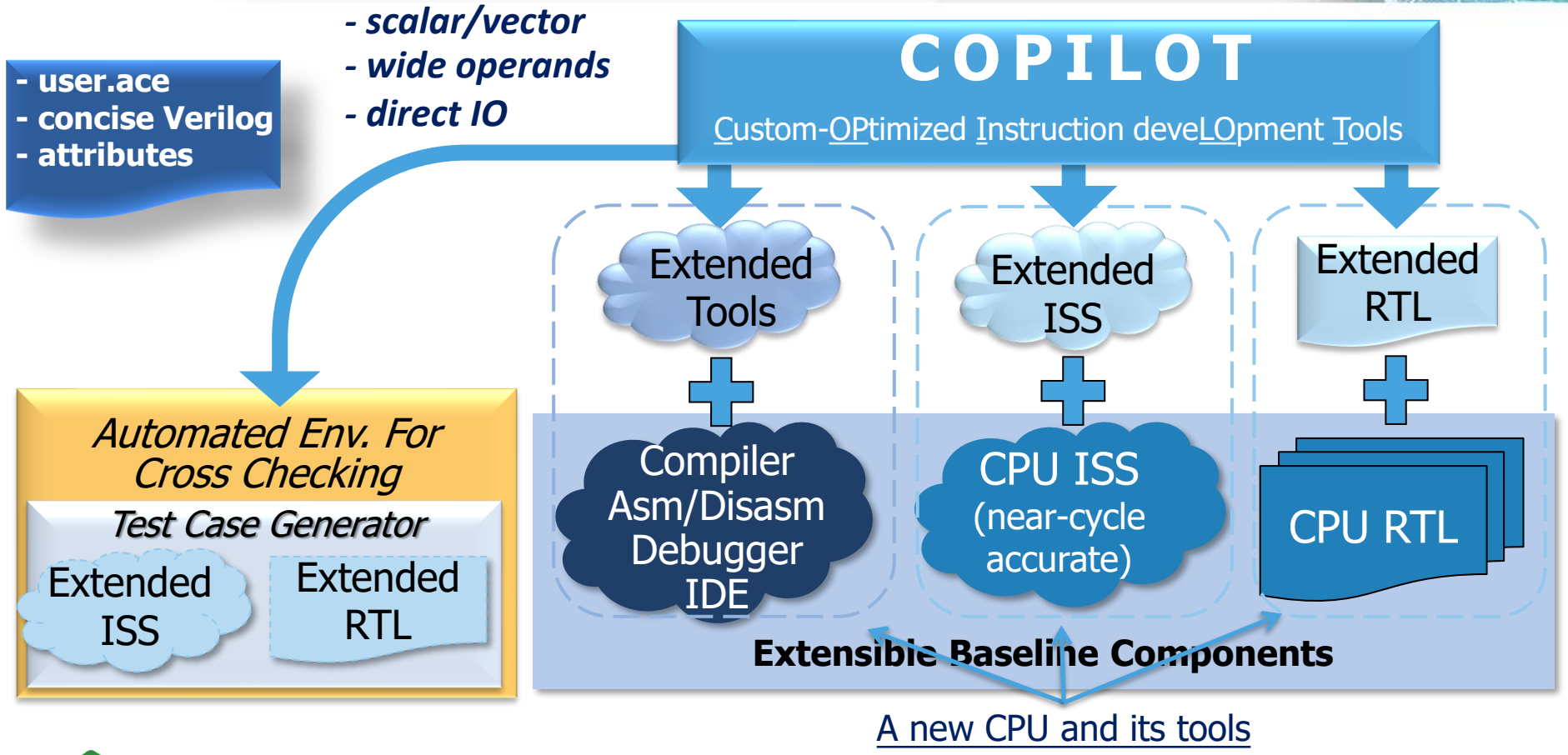
- Total compute performance (at 28nm):

Coremark®	45-series (1.2 GHz)	27-series (1.1 GHz)	Speedup (Per-MHz)	Speedup (Total Perf.)
RV32	**5.66**	3.58	1.58	**1.72**
RV64	**5.50**	3.53	1.56	**1.70**

  - 70% higher than the 27-series
  - With less than 50% increase in logic area and power
- Memory bandwidth (C copy): 45-series is 35% higher than 27-series
- Running up to 2.4 GHz at 12nm

ACE Streaming Port

Andes Custom Extension™ Framework



ACE Features

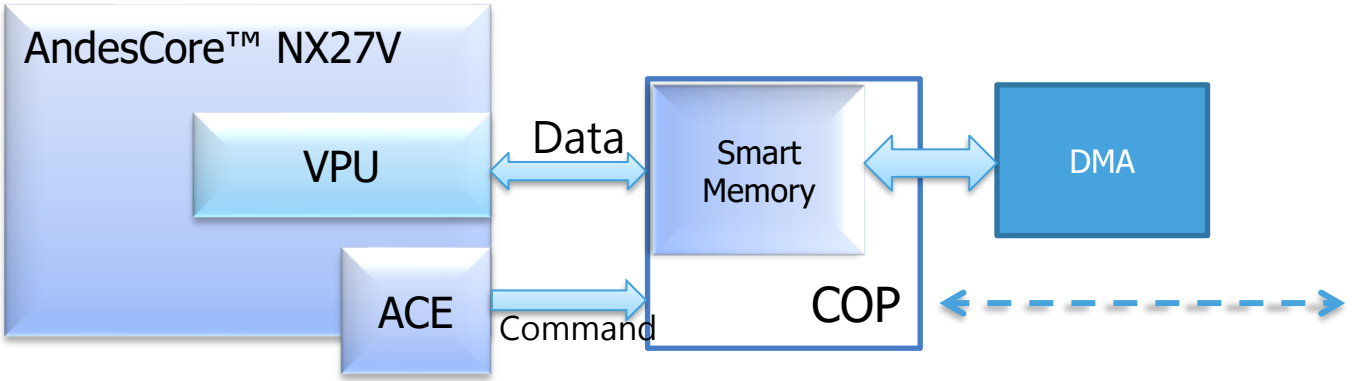
Items	Description	
Instruction	scalar	single-cycle, or multi-cycle
	vector	for loop, or do-while loop
	background option	retire immediately, and continue execution in the background. Ap plicable to scalar and vector.
Operand	standard	immediate, GPR, baseline memory (thru CPU)
	custom	- ACR (ACE Register), ACM (ACE Memory), ACP (ACE Port) - With arbitrary width and number
	implied option	Implied operands don't appear in mnemonic
Auto Generation	- Opcode assignment: automatic by default - All required tools, and simulator (C or SystemC) - RTL code for instruction decoding, operand mapping, dependence checking, input accesses, output updates - Logic sharing - Waveform control file	

Andes Custom Port

- COPILOT can generate customized ports on CPU
  - Can access external data directly
  - Can control external devices
  - Can be accessed by external devices directly

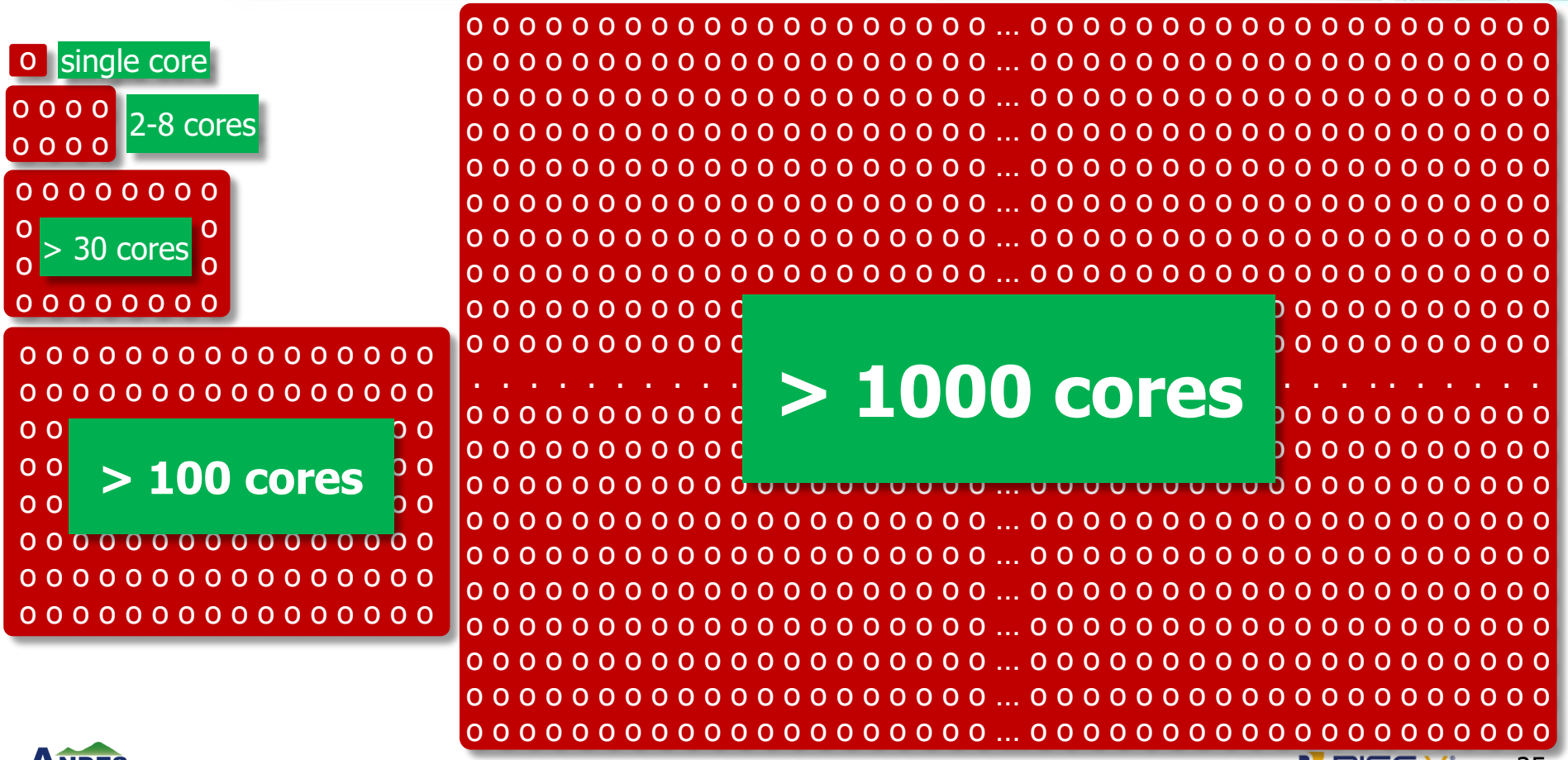
ACE Streaming Ports

- **ACE load/store instructions utilizing streaming ports**
  - Non-blocking: A "command" to custom coprocessor (COP) and its data movement are decoupled
- **Data channel:**
  - Exchange data between Smart Memory and VPU
  - Control by the commands send out from ACE
- **Command channel:**
  - Internal info: used by CPU only
    - CPU register(s): XRF, FRF, and LMUL-aware VRF
  - Fixed info: used by both CPU and COP
    - data size
    - memory address: from auto-incremented address pointers
      - Optional wrap-around for ring buffer semantics
      - If address isn't aligned, COP should align the data
  - Custom info: for COP only, not used by CPU/ACE



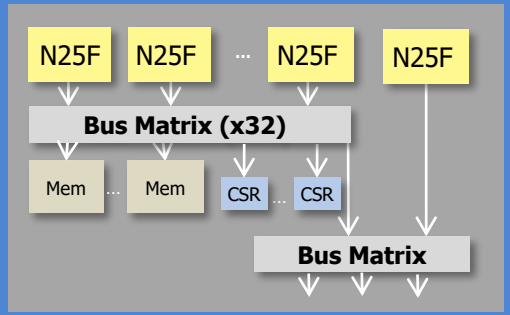
Summary

Andes Core for Now and Future

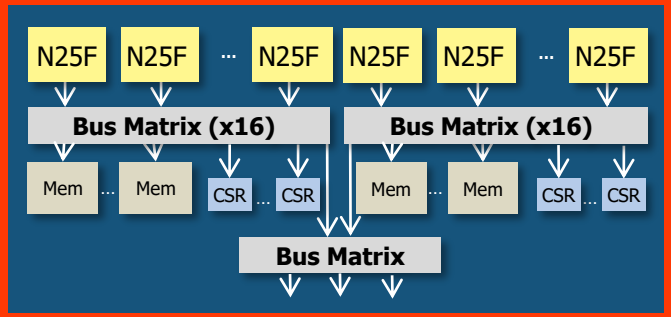


Multiple Cores Application

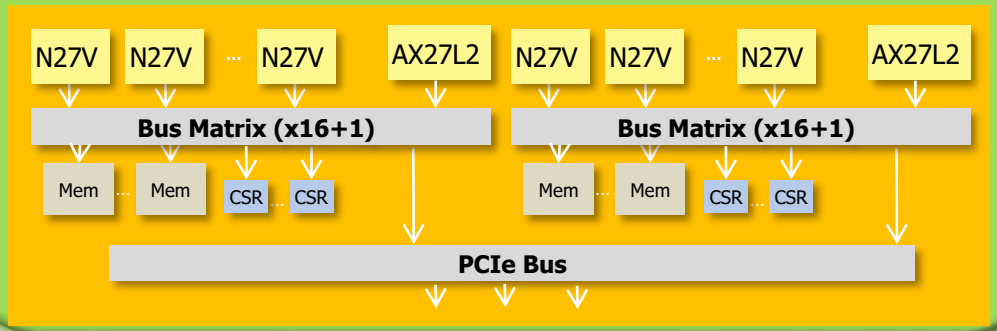
Channel Controller for SSD



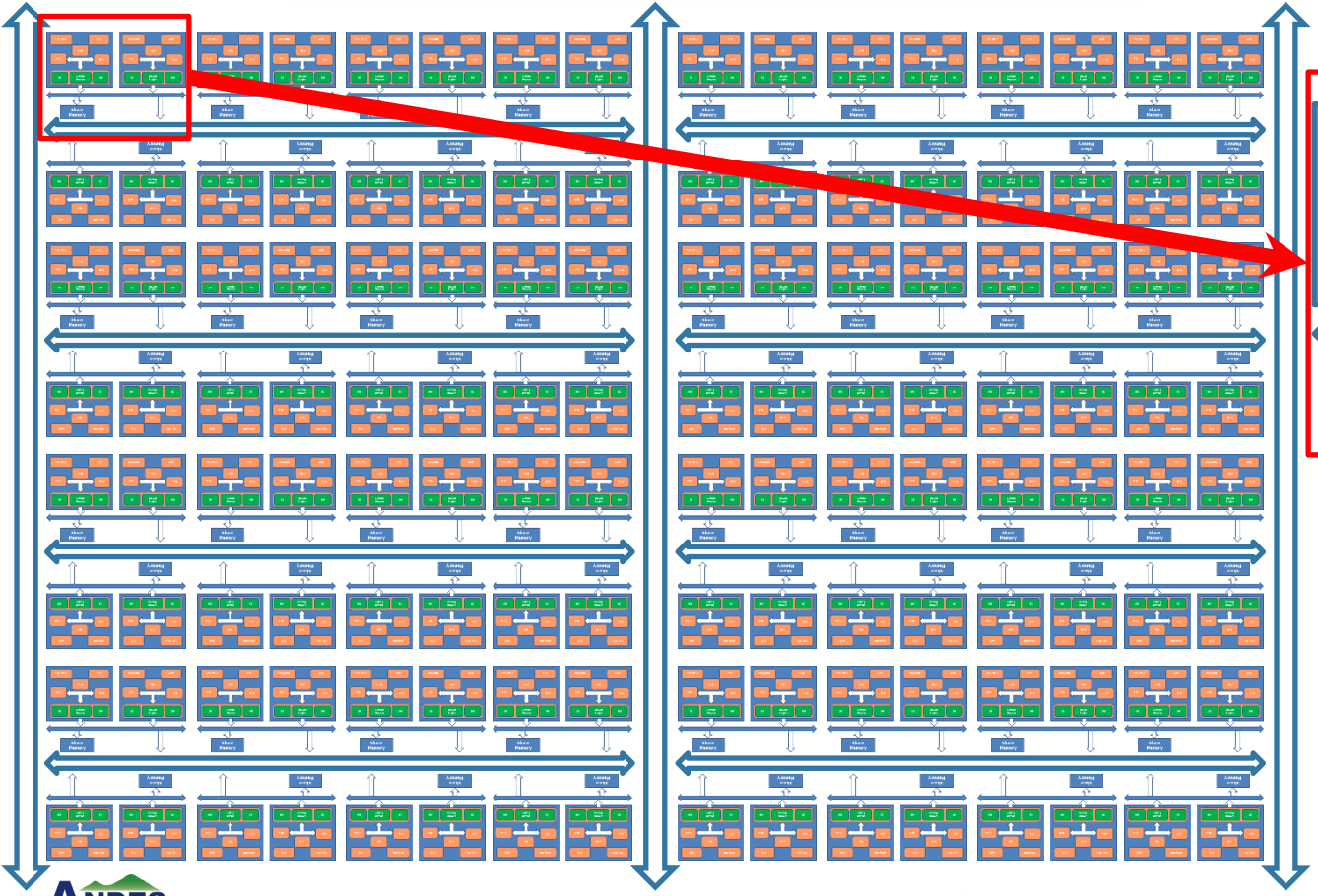
5G OpenRAN small cells

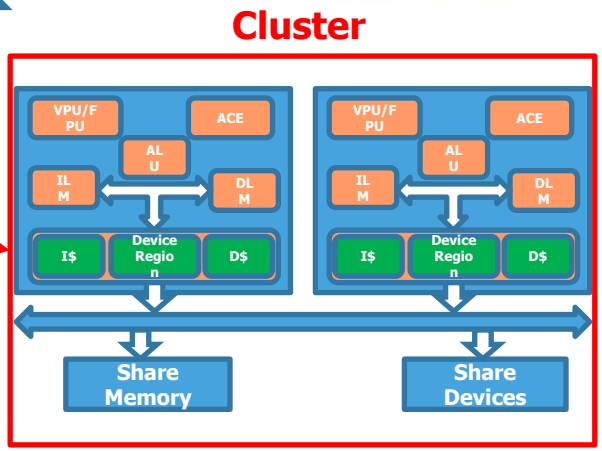


Vector Processor for AI Server



Multiple Clusters Application



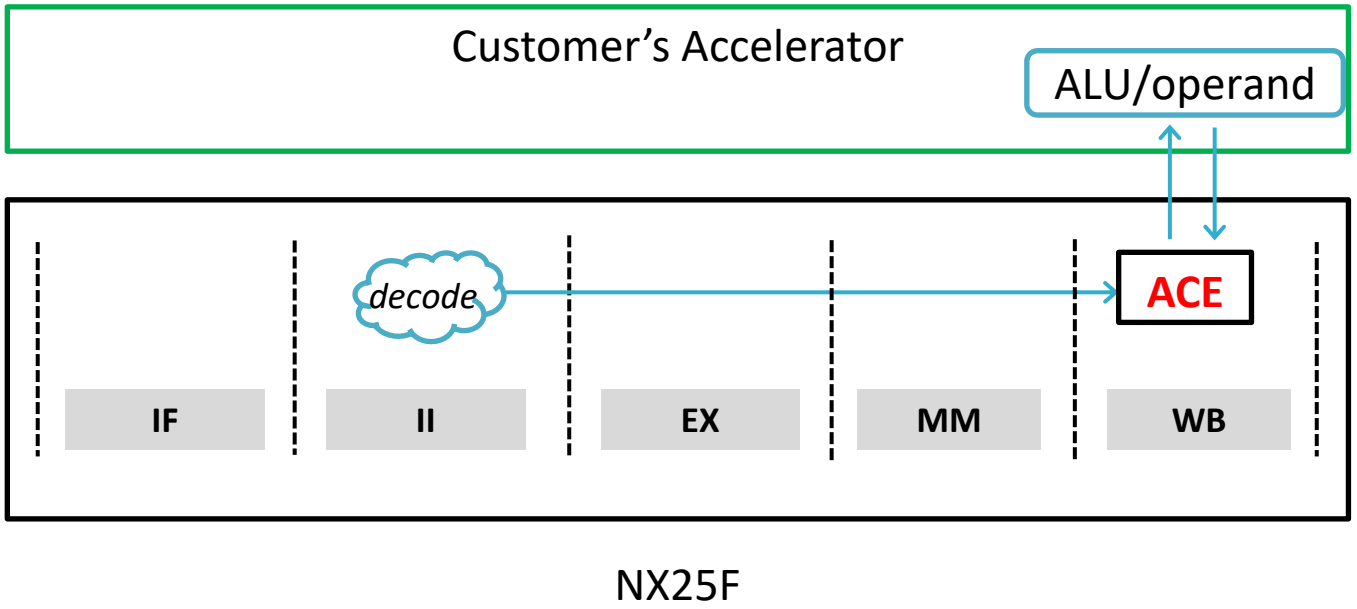


Customized by Andes

- Reusing clusters minimize Risk
- Easier to Debug
- Easier to Program
- Easier Migration of Tasks

More Application in RISC-V

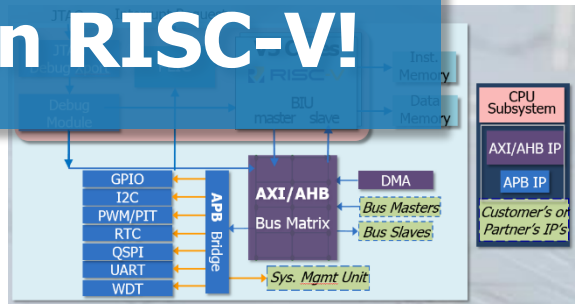
- **Sharing ALU and operand in customer's accelerator with ACE**



Summary

- Rich MCU support:
  - Smaller code size (CoDense™), code memory support, low power
  - Comprehensive AndeSight™ IDE for development, debugging and analysis
  - Strong compiler support
- Comprehensive HW/SW solutions for data-intensive computations:
  - DSP extension: sensors, audio/voice, small image and slow video
  - Vector extension: high-data-rate computations such as AI, video, 5G and HPC
  - Custom Extension: programmable acceleration with extreme efficiency
- Battle-tested platforms for flexible SoC construction
- Rich SW stack
  - From [illegible] Open RTOS
  - To kernel with debug, analysis and power mgmt feature
- Flexible customization **(CCBU)**
  - Dedicated service team
  - Time to market

Andes is Your Best Choice on RISC-V!



Thank You!

ANDES
TECHNOLOGY