# ARC-V RMX-100 Series Processors

## Highlights

- RISC-V ISA 32-bit processors for ultra-low power embedded applications
- Base RV32E / RV32I ISA + optional extensions
- Single and double precision floating point
- High degree of configurability
- Support for custom instructions
- Support for 4KB to 64KB L1 instruction cache
- Support for up to 2 MB of closely coupled memories and direct mapping of peripherals
- Native Arm AMBA® AHB5™ and AXI5 interfaces
- Optional single and multicycle multiplier and HW divide module
- ECC support
- RISC-V AIA compliant interrupt handling
- N-Trace real-time trace debugging
- Easy programming support with MIPS Metaware C/C++ Compiler
- Broad third-party and open-source software development tools support
- Full compatibility with existing RISC-V code base

## Target Applications

- Industrial: Motor control, smart metering
- Automotive: Sensors, keyless entry, body electronics, safety management
- Consumer: AIoT, hearables
- Storage: Consumer SSDs, eMMC, UFS, SD cards
- Networking: LPWAN, M2M, BLE control, WAP

## Overview

The MIPS ARC-V™ RMX-100 series processors are optimized for use in embedded applications where power and area are the utmost concern.

The ARC-V RMX-100 processors are based on the RISC-V instruction set architecture (ISA) and support both RV32E and RV32I ISAs with a high degree of configurability. The processors feature a balanced 3-stage Harvard architecture pipeline that provides efficient throughput.

The ARC-V RMX-100 features up to 64KB of level 1 (L1) instruction cache and up to 2MB each of closely coupled instruction and data memories (CCM).

To enable easy software development, the ARC MetaWare Development Toolkit features a rich software library. The ARC-V RMX-100 processors maintain the high code density and offer excellent performance within a very small footprint.

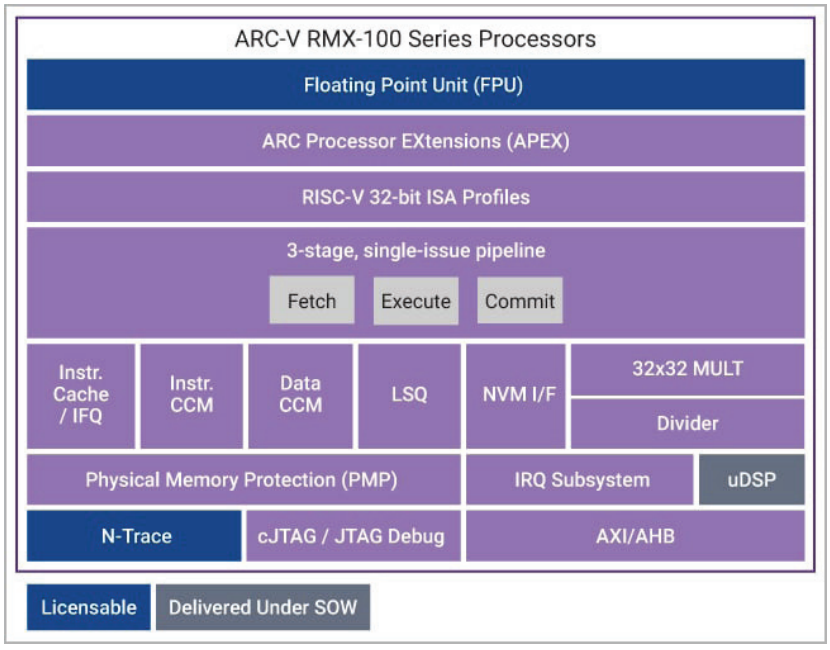


Figure 1: ARC-V RMX-100 Series Embedded Processor Block Diagram

## Features

- 32-bit RISC-V embedded CPU with balanced 3-stage pipeline
- Up to 2MB instruction and data closely coupled memories (CCM)
- Optional 4 KB to 64 KB L1 instruction cache
- Architectural clock gating and enhanced sleep instructions
- Supervisor mode for enhanced security
- RISC-V Memory Protection Unit to control access rights to the memory
- Support for World ID's target-side access control / firewalling
- ECC support
- Integrated watchdog timer
- 32x32 multiplier / configurable hardware divider
- RISC-V Advanced Interrupt Architecture (AIA) core-level up to 512 interrupts
- Optional APLIC with vector nesting and multi-core support
- Optional 2x16-bit DSP instruction extensions "micro-DSP" (uDSP)
- Native Arm AMBA® AXI5™, AHB5 interfaces, JTAG and Compact
- JTAG (cJTAG) debug interface

## Pipeline

The ARC-V RMX-100 cores have a low-latency 3-stage pipeline that is optimally balanced to achieve very low power consumption with excellent embedded performance. The pipeline is designed to give longer access to memory, allowing maximum clock speeds at all process nodes. The pipeline is based on the Harvard architecture with separate instruction and data memory storage that can be simultaneously accessed. The pipeline supports precise exceptions with a commit point after the second stage.

## RISC-V ISA: Flexible Profile Options

The ARC-V RMX-100 processors leverage the versatile RISC-V instruction set architecture (ISA), offering robust support for both the RV32E and RV32I 32-bit ISAs. These processors are designed with a high degree of configurability, enabling flexible ISA extension options to efficiently tailor to specific application requirements.

| Base ISA / Extension | Description | Default Profile | Mini Profile |
|---|---|---|---|
| ISA | RISC-V Base ISA – 32-bit | RV32I | RV32E |
| AIA | Advanced Interrupt Architecture | ✓ | ✓ |
| Zifencei | Instruction fence | ✓ | ✓ |
| Zihintpause | Pause hint | ✓ | ✓ |
| Zicbom, Zicbop | Cache block management | ✓ | ✓ |
| Zicsr | CSR instructions | ✓ | Opt |
| Zce | Compressed instructions | ✓ | Opt |
| Zba, Zbb, Zbs | Bit manipulation instructions | ✓ | Opt |
| Zicond | Integer Conditioning Operations | ✓ | Opt |
| M | Integer multiply and divide | Opt | Opt |
| A | Atomic instructions | Opt | Opt |
| Zfinx, Zdinx | Floating Point | Opt | Opt |
| N-Trace | RISC-V Trace Specification for Nexus IEEE 5001 | Opt | Opt |
| World ID | RISC-V World-ID extension | Opt | Opt |
| Supervisor Mode | RISC-V supervisor mode | Opt | n/a |

Figure 2: ARC-V RMX-100 RISC-V ISA Configurability

## Configurable Options

The ARC-V RMX-100 processor cores support a broad range of configurable options, enabling optimization for a specific application's performance, power and size requirements. The included ARChitect configuration tool features a graphical interface and produces verified RTL and synthesis scripts that are compatible with industry-standard design flows. With ARChitect, designers can add or remove features that improve the efficiency of the core for their application, including options such as custom instructions, multipliers, hardware divide, memory configuration, timers, interrupts, and much more.

## Memory Architecture

### L1 Cache Memory (instruction only)

The ARC-V RMX-100 processors feature support for instruction L1 cache that can be configured for 4 K, 8 K, 16 K, 32 K or 64 K size. The instruction cache is build-time configurable to support 1-way or 2-way associativity, and a user-selectable line size of 8, 16, 32, or 64 bytes. The caches can be individually configured to support line locking and invalidate, and to offer debug visibility.

### Closely Coupled Memories

The ARC-V RMX-100 processor supports up to 2 MB of closely coupled memory (CCM) for both instruction and data. The CCM is implemented as separate memory spaces for the Instruction Closely Coupled Memory (ICCM) and Data Closely Coupled Memory (DCCM) and can be used with cache memory to facilitate maximum system performance and flexibility. Both memory spaces can be accessed every clock cycle and both ICCM and DCCM can be read and written to from outside the core through AHB or AXI target interfaces.

## System Architecture and Interfaces

### Register File and Program Counter

The RMX-100 processor supports 31 base integer registers with an additional 32 floating registers as a configuration option.

### Bus Interfaces

The RMX-100 cores have native support for the Arm AMBA® AXI5, AHB5™ bus protocols. This is a build-time option with the AXI interface as the default selection. These enable the solutions to be easily connected to the SoC infrastructure in most chips without incurring any delay or complication in the bus interface.

### Multipliers

Optional single cycle or multi-cycle 32x32 multipliers are available to designers, with the ability to implement one or both at build time. The required number of clocks in the 32x32 multiplier can be configured up to 10 cycles to complete the multiply, with the default configuration being without a multiplier.

### DSP Extensions (Optional)

The RMX-100 can be enhanced with "micro-DSP" (uDSP) instructions to accelerate various DSP algorithms, such as FFT and FIR, which are beneficial for specific applications like ADC processing. The DSP-enhanced implementation features dual 16-bit MAC DSPs, enabling efficient control and DSP processing on a single core.

### Error Protection

RMX-100 cores provide support for error protection and caches where present. The different protection schemes may be combined to achieve several levels of protection against malicious or misbehaving code in critical applications. ARC-V RMX-100 supports two error detection techniques: single-bit error correction, double-bit error detection (SECDED) and parity which detects single-bit errors. Data-only protection or data and address protection are both supported for ECC.

### Interrupt Architecture

The RMX-100 series features the RISC-V Advanced Interrupt Architecture (AIA) with local and system level options. It includes an integrated core-level interrupt controller supporting up to 512 interrupts and optional system-level Advanced Platform Level Interrupt Controller (APLIC).

### ARC Processor EXtension (APEX) Interface (RISC-V-defined custom extensions)

The processors are designed to be extendable with the addition of custom instructions. These instruction extensions may include more processor and auxiliary registers, new instructions, and additional condition code tests. Custom instructions enable designers to efficiently add their proprietary hardware to the processor to further increase application performance.

## Optional Features (Separately Licensed)

- FPU Floating Point Unit offers single and double precision math instructions
- ARC Trace I/F provides N-Trace and real-time trace debugging features

## Complete Suite of Development Tools

To facilitate rapid development, the processors are supported by a complete suite of development tools via Visual Studio Code IDE Plugin. This includes the MetaWare Development Toolkit that generates performance optimized code ideal for deeply embedded applications.

MIPS offers a suite of GNU tools (ARC GNU) for developers targeting real-time operating systems (RTOS) as well as bare metal systems. The ARC GNU Toolchain includes the GCC compiler and GDB debugger as well as a number of utilities and libraries that make up a complete software toolchain.

Additional development tools are available as part of the RISC-V ecosystem.

| | | |
|---|---|---|
| **Compile** | MetaWare Compiler | • Optimize your code for size and performance<br>• Leverage core-specific features to reduce cost and increase performance<br>• Utilize your user-defined instructions to achieve design goals |
| | GNU GCC Compiler | • Freely access an open source solution with the GCC compiler |
| **Debug** | MetaWare Debugger | • Easily debug multiple targets with the same user interface<br>• Quickly profile hotspots in your code<br>• Use scripting to increase productivity |
| | JTAG Debuggers | • Efficiently bring-up hardware with tools from ARC 3rd Party Tools and Software partners |
| | GNU GDB Debugger | • Use the open source GDB debugger to debug real and simulated targets |
| **Deploy** | Functional and Cycle-Approximate Simulators | • Develop and debug software before hardware is available<br>• Simulate large programs with very fast ARC-V models<br>• Quickly optimize your software with near cycle-accurate simulation |
| | Zephyr Real Time OS | • Open source RTOS optimized by MIPS for ARC processors |

## Documentation

The following documentation is available for the MIPS ARC-V RMX-100 processors:

- ARC-V RMX-100 Series Technical Reference Manual
- ARC-V RMX-100 Series Integration Guide

## Testing, Compliance and Quality

Verification of the MIPS ARC-V RMX-100 series processors follows a bottom-up verification methodology from block level through system level. Each functional block within the product follows a functional, coverage-driven test plan.

The plan includes testing for RISC-V ISA compliance as well as state- and control- specific coverage points that have been exercised using constrained pseudo-random environments and a random instruction sequence generator.

## Deliverables

The MIPS ARC-V RMX-100 processors are delivered in Verilog HDL in the ARChitect IP Library. The HDL is configurable by the user and output from the ARChitect IP Configurator tool. To test that the product performs as expected, a basic testbench of Customer Confidence Tests (CCT) is included.

.

## About MIPS:

MIPS by GlobalFoundries delivers software to silicon with RISC-V for building physical AI platforms. MIPS delivers software-hardware co-design, optimized AI, and custom ASSP design and manufacturing. Together with ARC, MIPS delivers the open, standards-based processor IP portfolio for embedded applications. Physical AI is built on MIPS.

For more information, visit **www.mips.com/arc.**