ARC EV7x Embedded Vision Processors with DNN Accelerator

Highlights

- Optimized for high frame-rate and video resolution embedded vision and imaging applications
- Integrates 32-bit scalar core, 512-bit vector DSP, and optional DNN accelerator
- DNN accelerator delivers up to 35 TOPS performance
- ASIL B or D Ready EV7xFS products integrate safety-critical hardware features
- IEEE 754-compliant vector floating point unit option offers single or half precision operations and advanced math functions
- AES encryption option protects valuable data such as training and sensor data
- High productivity MetaWare EV Development Toolkit supports OpenCV, OpenVX and OpenCL C, Tensorflow, Caffe, and ONNX standards

Target Applications

- Automotive driver assistance systems (ADAS) and autonomous driving
- Smart home / IoT
- Augmented/mixed reality
- Robotics
- Surveillance
- Drones
- Facial recognition payment
- Multi-function printers
- Digital still cameras

Overview

The DesignWare® ARC® EV71, EV72, and EV74 Embedded Vision Processor IP provides high performance, low power, area efficient solutions for a stand-alone computer vision and/or AI algorithms engine or as an accelerator for vision-enabled SoCs. The EV7x family is designed for power sensitive vision applications ranging from low power facial recognition, to augmented reality, to high-performance automotive autonomous vehicle vision.

The EV7x family integrates a high-performance 32-bit scalar core, a 512-bit vector DSP, an optional vector floating point unit and an optional deep neural network (DNN) engine for complete computer vision and deep learning algorithm (CNN/RNN) coverage to perform fast and accurate object detection, classification, and scene segmentation. An optional AES-XTS encryption engine protects sensitive coefficient and graph topology data as well as user's biometric data.

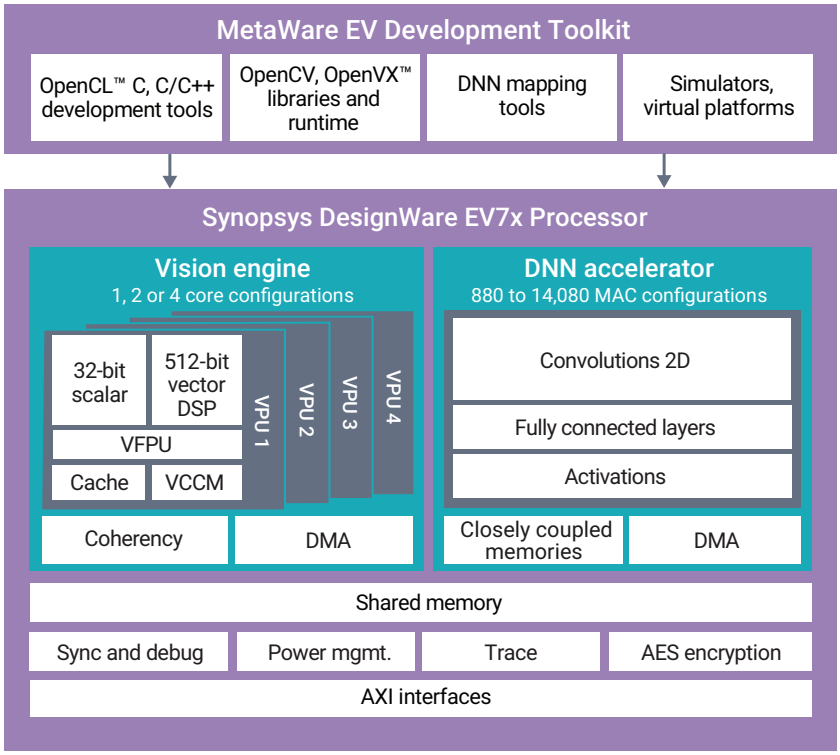

Figure 1: DesignWare EV7x Processor architecture

The EV71's vision engine has one vector processing unit (an integration of a 32-bit scalar CPU with a 3-way 512-bit wide SIMD vision digital signal processor, optional vector floating point unit, and configuration cache and closely coupled memories). The EV72 vision engine has twice the performance with two vector processing units. The EV74 features four vision processing units for the highest level of performance. All three vision processors can be configured with the optional DNN accelerator, which can be scaled from 880 MACs to 1,760, 3,520, 7,040, or 14,080 MACs.

The optional DNN accelerator is a programmable deep neural network engine for fast and accurate detection of a wide range of objects such as faces, pedestrians, and hand gestures. In addition to supporting convolutional neural networks (CNNs), the DNN supports batched LSTMs (long short-term memories) for applications that require time-based results, such as predicting the location of a pedestrian based on their observed path and speed.

The processors are highly scalable and configurable, enabling users to tailor them to their specific application requirements and create a range of products that can be programmed with a single toolchain. The combination of heterogeneous processing units provides the EV7x family with the flexibility of software solutions with the high performance and low power consumption of dedicated hardware.

To speed application software development, the EV7x processor family is supported by MIPS' ARC MetaWare EV Development Toolkit, a comprehensive software programming environment based on embedded vision standards including OpenCV, OpenVX™, and the OpenCL™ C language. The toolkit consists of a C/C++ compiler, an optimized C compiler, debugger, instruction set simulators, OpenVX runtime, and a DNN mapping tool. The DNN mapping tool automatically converts neural networks trained using popular frameworks, like Tensorflow, into optimized executable code for the programmable DNN accelerator.

EV Vision Engine

The EV7x vision engine features a quad issue super vector architecture with a 32-bit, high-performance scalar pipeline, and a 512-bit wide SIMD Vector DSP (VDSP) that are optimally balanced to achieve excellent performance with low power consumption. The parallel processing capabilities are ideal for high-performance OpenCV or OpenVX computer version kernels, DSP algorithms, or non-batched RNN and LSTM neural networks. The vector processing unit executes one scalar and three vector instructions (128-bit instruction bundle) per cycle. The vision engine is supported by configurable instruction and data cache for scalar operations and vector closely-coupled memory (VCCM) – a single cycle access RAM – for vector processing.

The vision engines' register file has 32 32-bit registers. The register file can be constructed from fast, single cycle access memory or flip-flops, and supports one or two write ports (one is the default) and (two) read ports. The vision engine features 64-bit load double and store double instructions that can be optionally included in the processor at build time. These are single instructions that load or store 64-bits of data to and from register pairs. There is no additional cycle penalty due to the wider and banked DCCM that support non-aligned loads and stores. The separate instruction and data L1 cache that can be independently configured for 4K, 8K, 16K, 32K or 64KB size. The I-cache supports 2- and 4-way associativity, and a 32-, 64-, and 128-byte line size. The caches can be individually configured to support line locking and invalidate, and to offer debug visibility. The data cache implements the MOESI protocol and supports cache-to-cache transfers. The EV71, EV72, and EV74 Processors support 8KB to 128KB of data closely coupled memory (DCCM), and 32KB to 256KB of vector memory (VCCM) for the VDSP. The CCM is implemented as a separate memory space and can be accessed every clock cycle.

DMA for the VPU(s) is provided by a Streaming Transfer Unit (STU) which is tightly coupled with the VCCM. The STU brings in external data for vector DSP processing, supports 1D and 2D (image) transfers and IO Coherency, and has 4x 128b connections to the AXI bus. The coherency unit, the optional clustered shared memory and hardware support for synchronization (semaphores, interrupt dispatches, small messages in a mailbox) allows the multiple VPUs of the EV72 and the EV74 to be integrated seamlessly.

The VDSP is configurable up to 512-bits wide with vector support for dual 8x8, 16x16, dual 16x16 or 32x32 MAC processing. The DSP SIMD capability can deliver 128 dual 8-bit, 64 dual 16-bit, 32 16-bit, or 16 32-bit MACs per cycle. The VDSP unit includes a vector register file with up to 32 512-bit wide registers. The VDSP supports full predication and has scatter/gather instructions and hardware to maximize performance.

The EV7x vision engines are based on the ARCv2 RISC ISA, which supports advanced processor capabilities. The architecture and pipeline are designed to meet the needs of next-generation system-on-chip (SoC) applications

and enable the deployment of a full range of 32-bit processors. The ARCv2 ISA enables the implementation of complex, heterogeneous SoCs with processors that are precisely targeted to meet the specific performance and power requirements for each instance on the SoC.

The EV7x includes an optional IEEE-754 compliant Vector Floating Unit that supports both full (32-bit) and half (16-bit) floating point operations. This high performance VFPU implementation uses pipelined, high-performance floating point components to achieve up to 512 GFLOPS (4 VPUs, 1 GHz, half-precision). The VFPU also supports an extensive set of math functions including: div, √x, 1/√x, sin(x), cos(x), log_2 (x), 2^x “, “e^x, atan2(x).

DNN Accelerator

The optional embedded DNN accelerator adds scalable deep learning and AI capabilities to the EV7x family. The DNN accelerator is optimized for CNNs and batched or convolutional Recurrent Neural Networks (RNNs) or LSTMs and supports advanced software features to support the latest pruning, compression, and layer merging techniques to increase performance and minimize bandwidth. The DNN can be configured from 880 multiply-accumulators (MACs) up to 14,080 MAC versions. Most of the MACs are used for 2D convolutions while a portion is dedicated to 1D convolutions needed for fully connected layers. The DNN datapath supports 8- and 12-bit data precision. The DNN accelerator supports flexible activation functions, including ReLU, PReLU, ReLU6, tanh and sigmoid. The EV7x supports all CNNs including popular networks such as MobileNet, GoogLeNet, ResNet, Yolo, Faster R-DNN, and ICNet. Designers can run CNN graphs originally trained for 32-bit floating point hardware on the EV7x’s DNN accelerator using 8- or 12-bit resolution significantly reducing the power and area of their designs while maintaining high levels of detection accuracy.

In addition to supporting CNNs, the DNN supports batched LSTMs (long short-term memories) for applications that require time-based results, such as predicting the location of a pedestrian based on their observed path and speed.

The DNN accelerator is supported by a high-performance DMA for transferring image data from external memory into the internal closely coupled memories.

Cluster Shared Memory

A low-latency shared data memory is included in the processor to support information passing and coordination between the multiple CPUs and the DNN processing element cores. This memory is used as a software-managed scratch pad and is configurable from 0 to 8MB. To allow for larger sizes, the memory is internally multi-banked, but this is invisible to the software. It includes arbitration to support concurrent access from the CPU cores and/or the DNN processing elements. The shared subsystem data memory is optional.

Embedded Encryption Engine

The DesignWare ARC EV7x Vision Processors offer optional AES-XTS encryption engines to protect data passing from on-chip memory to the vision processor and DNN accelerator. The AES-XTS engine prevents high-value data such as training datasets and personal biometric data, including facial recognition and retina scans, from being exploited.

Real-Time Trace

The DesignWare ARC Real-Time Trace (RTT) unit is a hardware helps trace executed instructions or program flow and data. ARC RTT generates Nexus 5001 class 3-compliant trace messages. The RTT system can be set up in many different configurations which need to be specified as build-time configurations by including the trace generator in the core and the RTT module at build time. ARC RTT can support on- and off-chip memory setups to suit your application tracing needs. Bus Interface.

The EV7x processor has native support for the Arm® AMBA® AXI™ bus protocol. The AXI bus is 64- or 128-bits wide to improve system throughput.

Interrupts and Exceptions

The EV7x processor supports up to four output interrupt pins, and up to three input interrupt lines. These can be used, for example, to synchronize with an external host. The host can also raise an interrupt by writing in a memory-mapped register or by driving an interrupt input pin on the EV7x processor.

EV7xFS for Functional Safety

For automotive and other applications requiring additional reliability and functional safety, the ARC EV7xFS processors provide ASIL B or D Ready support to accelerate ISO 26262 certification of automotive SoCs. The functional safety-enhanced processors offer hardware safety features, safety monitors, and lockstep capabilities that enable designers to achieve stringent levels of functional safety and fault coverage without significant impact on power or performance. In addition, the EV7xFS offers a "hybrid" option that enables users to select required safety levels up to ASIL D in software and post-silicon.

SoC Integration

The EV7x processors are designed to integrate seamlessly into a SoC. They can be used with any host processor and operate in parallel with the host. The EV7x family includes support for synchronization with the host through message passing and interrupts. In addition, part of the EV7x processor memory map can be made visible to the host. These features enable the host to maintain control while allowing all vision processing to be offloaded to the EV7x processor, reducing power, and accelerating vision computation. The EV7x processors can access image data stored in a memory mapped area of the SoC or from off-chip sources independently from the host through the Arm AMBA AXI standard system interface if required.

Comprehensive Software Environment

The ARC MetaWare EV Development Toolkit is a complete set of tools that provides everything needed to program the EV7x processors. The MetaWare EV Toolkit includes the MetaWare C/C++ Compiler, MetaWare Debugger and Instruction Set Simulator and adds an OpenCL C compiler for writing vision kernels for the vector DSP. It also includes an OpenVX Kernel Library and Runtime software, as well as an OpenCV Library. A mapping tool to map DNN graphs to the DNN accelerator is also provided. For system simulation, the EV Virtualizer Development Kit (EV VDK) is a virtual prototype of an EV system that allows for early software development. All this functionality is contained in one comprehensive suite of tools to provide a high productivity development environment for the creation of embedded vision applications with the EV7x processors.

The DNN mapping tool is provides a complete CNN software environment to support Caffe and Tensorflow neural network frameworks with additional frameworks supported via the industry-backed ONNX interchange format. DNN graph training is done off-line, typically on a server farm, and the resulting graph is programmed into the object detection engine by the user with the DNN graph mapping tool.

Components		Description
Compilers and Debuggers	MetaWare C/C++ Compiler, OpenCL C Compiler, MetaWare Debugger	• Develop highly optimized code with an efficient C/C++ compiler for scalar processors • Use OpenCL C to develop efficient vision kernels • Debug code with a comprehensive source-level debugger and profiler
Simulators	Includes fast nSIM Instruction Set Simulator and EV VDK	• Use a fast simulator to develop and debug vision software before hardware is available • Use a virtual prototype of an EV Processor system to start early software development
Libraries	OpenCV Library, OpenVX Kernel Library	• Use a standard library of open source functions for common vision applications with the OpenCV library • Use OpenCL C language to develop kernels for use in the OpenVX environment
Graph Mapping	DNN Mapping Tool	• Map neural network graphs to the DNN accelerator
Runtime	OpenVX Runtime	• Develop high-performance vision applications with the OpenVX runtime framework

Table 1: DesignWare MetaWare EV Toolkit components

The primary programming framework is the OpenVX runtime software that manages execution on the EV7x processors. It enables the development of high-performance vision applications. The OpenVX framework includes the 59 standard OpenVX kernels and an additional 70 extension kernels. OpenVX is a Khronos standard for acceleration of embedded vision algorithms. Client-defined OpenVX functions are supported, with the kernels described in standard C/C++ or in OpenCL. OpenVX graphs are automatically mapped, tiled (where possible), and executed on the EV7x processor and DNN accelerator. OpenCV and OpenVX are complementary and used together in vision applications.

OpenCL C is an open standard programming language (developed by the Khronos Group) that supports vectorization and is used to ease the programming of the 512-bit wide VDSP. OpenCL C is a C-like language and is used with the EV7x processors to develop kernels that are executed in the OpenVX graph.

The MetaWare EV Development Toolkit includes a DNN mapping tool that analyzes neural networks trained using popular frameworks and automatically generates the executable for the DNN accelerator. For maximum flexibility and future-proofing, the tool can also distribute computations between the vision CPU and DNN resources to support new and emerging neural network algorithms as well as customer-specific DNN layers.

Documentation

The following documentation is available for the DesignWare EV7x Processors:

- EV7x Processor Release Notes
- EV7x Processor Getting Started guide
- ARCv2 Programmers Reference
- EV7x Processor Databook
- DNN Accelerator Databook

Testing, compliance and quality verification of the EV7x 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 ARCv2 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 DesignWare EV71, EV72, and EV74 Processors and optional DNN accelerator are delivered as Verilog HDL in the ARChitect IP Library. The HDL is configured 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.**