Designing an Efficient Edge AI System With Highly Integrated Processors

Manisha Agrawal
Product Marketing
Jacinto™ Processors

The expansion of automation is increasing from the factory floor to the front door.

At a glance

At a glance This white paper explains the requirements for building an efficient edge artificial intelligence (AI) system and how the TI TDA4 processor family can help optimize performance due to a heterogenous architecture.

1 **Defining AI at the edge**
Defining artificial intelligence at the edge. Many different kinds of systems can benefit from edge AI processing.

2 **What is an efficient edge AI system?**
What is a practical edge AI system? Consider which architecture and cores will best complete the tasks required of a system.

3 **Designing edge AI systems with the Jacinto 7 TDA4 SoC**
Designing edge AI systems with the Jacinto™ 7 TDA4 system on chip. The TDA4 processor family, is designed to achieve high-throughput, high-performance computing at low power and with lower system BOM costs.

Introduction

When consumers order a product online, automation increases efficiency throughout every step of the process, from creating raw materials, enhancing warehouse productivity and facilitating home delivery – sometimes only hours later. Continuing these remarkable advancements in automation will require better machine perception and intelligence with fewer mistakes, which can be achieved by bringing artificial intelligence (AI) to edge devices.

Creating faster, smarter and more accurate systems requires more data from more sensors, along with increasing amounts of processing power. However, more data and computing poses challenges to a system's performance, along with its power and cost requirements. System optimization and reduced development cycle times necessitate a practical approach to designing edge AI systems.

Defining AI at the edge

AI at the edge happens when AI algorithms are processed on local devices instead of in the cloud and is changing what is possible in industrial and automotive applications where deep neural networks (DNNs) are the main algorithm component. To operate efficiently in size-constrained, power and heat dissipation-constrained, and cost-constrained environments, edge AI applications require high-speed and low-power processing, along with advanced integrations unique to the application and its tasks. Figure 1 shows some of the applications where edge AI processing can be used to improve performance and efficiency. For example, edge AI systems that use vision input can implement a single camera for quality control on a production line, or multiple cameras to help support functional safety in a car or mobile robot.

Figure 1. *Intelligence at the edge exists in many different applications*

Edge AI systems can help improve efficiency in warehouses and factories; make cities, construction and agriculture safer and more efficient; and make homes and retail settings smart. Let's take a look at a few systems that require efficient edge AI processing:

1. Advanced driver assistance systems (ADAS). ADAS technology provides information insight into the environment around a vehicle to make driving more convenient, less stressful and safer. Most ADAS features are vision-based systems, taking high-resolution inputs from multiple camera sensors and using deep learning and computer vision algorithms to interpret those ADAS technology provides information insight into the environment around a vehicle to make driving more convenient, less stressful and safer. Most ADAS features are vision-based systems, taking high-resolution inputs from multiple camera sensors and using deep learning and computer vision algorithms to interpret those images.
2. Autonomous mobile robots and drones. For commercially viable robots, the system on chip (SoC) must process complex perception and navigation stacks at high speeds and low power, with optimized system costs. The SoC must also offload computationally intensive tasks such as image dewarping, stereo depth estimation, scaling, image pyramid generation and deep learning for maximum system efficiency.
3. Smart shopping carts. Smart shopping carts can calculate order totals when items are placed in the cart, recommend shopping list items and allow consumers to pay for groceries on the cart, enabling customers to have a more customized shopping experience and skip checkout lines. Most smart shopping carts have multiple vision sensors that automatically detect items with cameras and computer vision Smart shopping carts can calculate order totals when items are placed in the cart, recommend shopping list items and allow consumers to pay for groceries on the cart, enabling customers to have a more customized shopping experience and skip checkout lines. Most smart shopping carts have multiple vision sensors that automatically detect items with cameras and computer vision algorithms.
4. Edge AI boxes. Edge AI boxes are an intelligent extension of the camera systems used in retail stores, factories and buildings. High-throughput AI despite tight size constraints and power and heat dissipation challenges enable the box to perform intelligent processing on a greater number of cameras.
5. Machine vision cameras. Machine vision cameras for optical character recognition, object identification, defect detection and robotic arm guidance leverage embedded AI technologies to further simplify product development and improve system accuracy.

Table 1 lists system requirements from various applications.

	ADAS	**Robotics**	**Smart Retail**	**Machine Vision**	**Edge AI Box**
Deep learning accelerator	x	x	x	x	x
Multicamera image signal processing (ISP)	x	x	x	x	x
Vision accelerators	x	x	x	x	x
Depth and motion accelerators	x	x	x	x	x
Ethernet switch	x	x			x
Peripheral Component Interconnect Express (PCIe) switch	x	x			
Functional safety	x	x			

Table 1. *Key processing and components requirements of edge AI systems.*

What is an efficient edge AI system?

In an efficient edge AI system, DNNs cannot operate by themselves. An efficient AI system requires a complex vision pipeline, often including single or multicamera image processing, traditional computer vision and maybe even multiple DNNs. Some applications may also need video encoders and decoders. To process all of these inputs, a system needs high-performance computing. In addition, a system may require enhanced security and functional safety, increasing system complexity and cost.

An efficient edge AI system should be optimized for:

- Performance. The embedded processor must be able to deliver the speed, latency and accuracy that the system requires while also functioning reliably, even in harsh environments.
- Design constraints. The embedded processor must operate in designs with power and thermal constraints, including designs that are fanless, have passive cooling or need to operate for longer hours on battery power. The processor must also meet size and weight specifications to comply with physical constraints.
- Cost. Enabling processing that is high-performance and cost-effective will yield the lowest possible bill-of-materials (BOM) cost.

To build an efficient edge AI system, designers should consider which architecture and cores will best complete the tasks required of the system.

Selecting an SoC architecture

There are two embedded processor design options: a single-core or multi-core homogenous architecture or a multi-core heterogenous architecture, usually incorporating specialized processing capabilities to handle certain tasks. You should evaluate which architecture best meets the needs of your your edge AI system based on the required core types.

The goal of an edge AI system is to run AI, vision, video and other tasks on the best-suited core so that the resulting system is optimized for performance per watt and performance per teraoperations per second, as well as cost, size and weight. A heterogenous architecture that has the right cores for the right task is crucial for edge AI systems.

Not all processors with heterogenous architectures are designed equally. A silicon vendor has to select the right processing functions or processes and decide whether to accelerate those functions in hardware or make them configurable or programmable. They must also pay attention to the integration of cores into a system. The bus architecture and memory subsystem must enable efficient data movement between the cores.

Vision-based edge AI systems can be ineffective if the SoC has the incorrect core types for task acceleration, or too many cores not managed efficiently, or an inefficient bus infrastructure and memory subsystem.

Programmable core types and accelerators

Let's review the possible core types in edge AI systems:

CPUs

Central processing units (CPUs) are general-purpose processing units that can handle sequential workloads. They have great programming flexibility and benefit from a large existing code base. Generally, most edge AI systems have between two and eight CPU cores for managing platforms and feature-rich applications. CPUs are not a good fit for highly specialized tasks such as pixel-level imaging, computer vision and convolution neural network (CNN) processing, however. CPUs also have high power consumption but the lowest throughput of the different core types.

GPUs

Graphics processing units (GPUs) have hundreds to thousands of small cores that are a good fit for parallel processing tasks. Originally designed to implement a sequence of graphics operations, GPUs are common in deep learning applications and especially useful for training DNNs. One of the main drawbacks is that, because of the high number of cores, GPUs consume a lot of power and have higher on-chip memory requirements.

DSPs

Digital signal processors (DSPs) are power-efficient, specialized cores typically designed to solve multiple complex math problems. DSPs process real-time data at low power from real-world vision, audio, speech, radar and sonar sensors. DSPs help maximize processing per clock cycle. They are not as easy to program, however, requiring familiarity with the features of the DSP hardware, programming environment and optimization of DSP software to achieve the best performance.

ASICs

Application-specific integrated circuits (ASICs) and accelerators deliver maximum performance at the lowest power for system applications. They are popular choices when you know the core kernels for the function you want to accelerate. For example, core computation for CNNs always involves matrix multiplications. For traditional computer vision tasks, dedicated hardware accelerators can compute operations such as image scaling, lens distortion correction and noise filtering.

FPGAs

Field-programmable gate arrays (FPGAs) are a class of integrated circuits where it is possible to reprogram and target the hardware blocks for specific applications. They have lower power consumption than GPUs and CPUs but use more power than ASICs. The hardware is difficult to program, however, and requires expertise in hardware descriptor languages such as Verilog or Very High Speed IC Hardware Description Language.

Designing edge AI systems with the Jacinto 7 TDA4 SoC

The Jacinto 7 processor platform includes the TDA4 family, a scalable family of processors designed to achieve high-throughput, high-performance computing for edge AI systems at low power and with lower system BOM costs.

Based on a heterogenous architecture, the TDA4 SoC includes a multicore Arm® Cortex®-A72 microprocessor unit (MPU) and offloads computationally intense tasks such as deep learning inference, imaging, vision, video and graphics processing to specialized hardware accelerators and programmable cores as seen in Figure 2. An integrated processor with both a high-bandwidth interconnect architecture and a smart memory architecture enables high throughput. Integrating the advanced system components helps streamline the system BOM.

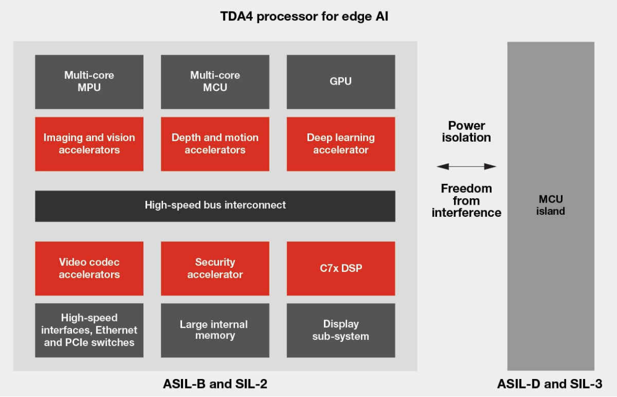
TDA4 processor for edge AI
Multi-core MPU
Multi-core MCU
GPU
Imaging and vision accelerators
Depth and motion accelerators
Deep learning accelerator
High-speed bus interconnect
Video codec accelerators
Security accelerator
C7x DSP
High-speed interfaces, Ethernet and PCIe switches
Large internal memory
Display sub-system
Power isolation
Freedom from interference
MCU island
ASIL-B and SIL-2
ASIL-D and SIL-3

Figure 2. *TDA4VM edge AI system partitioning.*

Deep learning accelerator

While suitable for other tasks, CPUs and GPUs are not the best cores to accelerate deep learning tasks. CPUs have throughput limitations and consume high power; GPUs consume the most power of all cores and have a large memory footprint.

The Jacinto TDA4 SoC integrates a deep learning accelerator comprising a matrix multiplication accelerator (MMA) in an ASIC, bolted to a programmable C71 DSP. The MMA enables high-performance (4K 8-bit fixed multiply accumulates per cycle) and low-power tensor acceleration, while the C71 DSP accelerates vector and scalar operations and manages the MMA.

The combination of the MMA and C71 DSP yields an accelerator that achieves among the highest performance (inference per second) and power (inference per watt) in the industry. The programming flexibility of the C71 core enables you to keep up with edge AI innovations. And when not used for deep learning, the core can process other computationally intensive tasks while consuming low power.

The smart memory architecture makes high utilization of the accelerator possible. The accelerator comes with its own memory subsystem; a dedicated 4D-programmable direct memory access (DMA) engine for data transfer; and specialized streaming hardware that can bring data directly from external memory to the functional units of the C71 core and MMA, bypassing the cache. Tiling and supertiling features minimize data transfer to and from external memory.

Table 2 shows an 8-bit fixed inference performance on the TDA4VM with an 8 teraoperations-per-second accelerator. Reported performance is with batch size 1, and a single 32-bit LPDD4.

Network	Image resolution	Frames Per Second (fps)
MobileNet_v1	224 × 224	741
Resnet-50 V1.5	224 × 224	162
SSD-MobileNets-V1	300 × 300	385

Table 2. *Inference benchmarks on MLPerf recommended models.*

Disclaimer: TI has used the MLPerf recommended models and guidelines for edge AI interference benchmarking. TI has not yet submitted the results to the MLcommons organization.

Imaging and computer vision hardware accelerators

Vision-based edge AI systems often include single or multicamera image processing and traditional computer vision tasks. In a CPU or GPU, these tasks consume lot of power and have throughput limitations.

The TDA4 SoC accelerates computationally intensive low-level brute-force pixel-processing vision tasks in hardware, such as ISP, lens distortion correction, multiscaling, and bilateral noise filtering in a vision processing accelerator core. A depth and motion perception accelerator core accelerates stereo depth estimation and dense optical flow to help enhance perception of the environment, as seen in Figure 3.

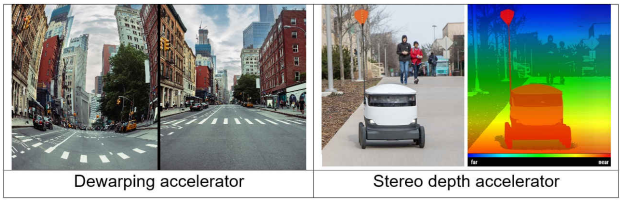

Figure 3. *Vision accelerator functions.*

Accelerating these tasks in hardware results in low power consumption and small size. Although these tasks are accelerated in hardware, their configurability offers flexibility by using the accelerator functions to best meet your system needs.

Such integration and acceleration removes the need for a custom ISP or FPGAs, and frees up CPU Mega Hertz for processing computationally intensive imaging and vision tasks in hardware. For example, a single vision processing accelerator core can process up to eight 2-megapixel or two 8-megapixel cameras at 30 fps. A depth and motion processing acceleration core can do stereo depth estimation at 80 megapixels per second and motion vectors at 150 megapixels per second.

Smart internal bus and memory architecture

Monitoring data movement and the memory architecture of a processor – in order to prevent various core blockages and delays when running multiple cores concurrently – can help maximize overall system performance.

The TDA4 SoC has a high-bandwidth bus interconnect with a nonblocking infrastructure and large internal memory. Multiple dedicated programmable DMA engines automate data movement at very high speeds.
This design enables high utilization of the hardware accelerators, with substantial double-data-rate (DDR) bandwidth savings. Reducing the number of DDR instances lowers the amount of power used by

DDR access, thus lowering the overall system power consumption.

Optimized system BOM

Let’s review the advanced integrated system components and features in the TDA4 SoC that can reduce system BOM cost savings for several types of edge AI applications:

- ISP. The integrated ISP core on the TDA4 eliminates the need for an external ISP or FPGA design. All single and multicamera AI applications such as machine vision, smart shopping carts, robotics and ADASs can benefit from this integration.
- Safety. The integrated Automotive Safety Integrity Level (ASIL) D and SIL 3-compliant safety microcontroller (MCU), with Cortex-R5 cores, helps achieve safety goals without an external safety MCU. With the rest of the processing also ASIL B/SIL 2-compliant, such an architecture enables ADAS, robotics, construction and agriculture electronic control unit applications.
- Ethernet and PCIe switches. Integrated Ethernet and PCIe switches eliminate the need for external switch components.
- Security. Integrated security accelerators offer state-of-the-art security support.
- DDR memory. Inline error-correcting code protection and fewer DDR memory instances (through smart memory) compared to typical memory architectures can result in cost savings.

Easy-to-use software development environment

A comprehensive software environment, shown in **Figure 4**, from TI enables you to employ a heterogenous architecture and access the full potential of silicon performance without having to learn TI hardware or proprietary software. Abstracting hardware accelerators through production-quality drivers, while also providing interfaces to a high-level operating system on the MPU for application development using industry-standard application programming interfaces (APIs), enables faster software development. Lower-level software from TI automatically accelerates imaging, vision, deep learning and multimedia tasks to the correct hardware accelerators, making high-performance application programming easy.

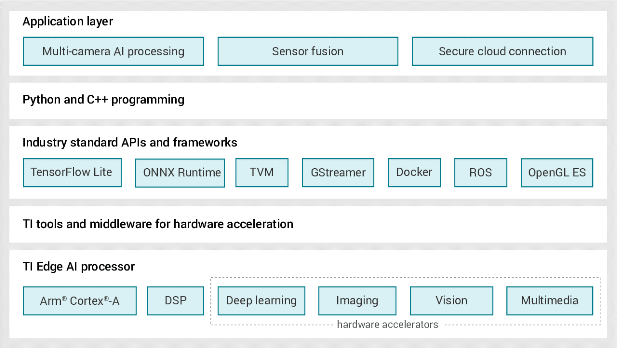

Figure 4. *Software development environment for edge AI applications.*

Conclusion

The adoption of a heterogeneous architecture in applications is growing. TI’s TDA4 processor family, with accelerated deep learning, vision and video processing, purpose-built system integration, and advanced component integration, enables commercially viable edge AI systems optimized for performance, power, size, weight and system costs. The TDA4 software development environment is built around open-source, industry-standard APIs, with automatic acceleration to hardware accelerators that enable faster edge AI application development.

AI is a rapidly evolving technology, fostering innovations in all dimensions of edge AI applications. It is pushing the boundaries of applications requiring higher computation needs. When enabled at lower power and lower system costs through the implementation of an embedded processor, edge AI can open up a whole new world of possibilities with embedded applications.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265