SOLUTION BRIEF

Artificial Intelligence
Smart Classroom

Artificial Intelligence and Machine Learning

Corerain
鲲 云 科 技

Executive Summary

FPGAs have long been recognized as excellent implementation platforms for artificial intelligence and machine learning (AI/ML) inference. With hundreds or thousands of available multiplier/accumulators (MACs), FPGAs provide ample resources for performing the computations required by convolutional neural networks (CNNs), recurrent neural networks (RNNs), and other deep neural networks (DNNs). However, fully utilizing the capabilities of those resources has required expertise in hardware-description language (HDL) programming in Verilog or VHDL. Corerain has developed a high-performance, low-power AI acceleration engine called the Custom AI Streaming Accelerator (CAISA), which can tap as much as 90% (on Intel® Arria® 10 GX 1150) of an FPGA's full performance potential without the need for HDL programming. Using Corerain's CAISA engine and the associated RainBuilder end-to-end tool chain, AI/ML application developers can now take advantage of FPGA-level application performance while using familiar deep-learning (DL) frameworks such as TensorFlow, Caffe and ONNX.

The Business Challenge

FPGAs can be harnessed as chips, which are incorporated into systems at the board level, or as programmable accelerator cards (PACs), which are plugged into existing system-expansion slots. In either case, AI/ML application developers require tools and techniques that provide a short time to market. Software developers are likely to take the card-based approach and require easy-to-use development tools that can quickly convert AI/ML applications developed using industry-standard frameworks into fast implementations running on the selected FPGA-based PAC. Hardware developers need ready-to-use AI/ML intellectual property (IP) they can drop into their FPGA-based system designs and easy-to-use tools that can be supplied to the team's application developers. Both types of development teams will also require appropriate C/C++ application programming interfaces (APIs) to control the accelerated CNNs and DNNs from software running on a host processor.

IP developers have taken many approaches to develop engines that efficiently implement CNNs, RNNs, and DNNs. Custom hardware architectures tuned to a specific neural network deliver performance but at the expense of flexibility. These custom architectures are permanently wedded to the network they are designed to implement and cannot be adapted to other networks. Soft co-processors based on SIMD architectures are programmable but they do not exploit an FPGA's full performance potential and limits the amount of acceleration that the FPGA can provide.

Corerain's CAISA engine is based on a streaming architecture that can extract as much as 90% (on Intel Arria 10 GX 1150) of the theoretical peak performance of an FPGA†. CAISA is scalable, so it can be sized to fit in a variety of FPGAs. This flexibility allows application designers to scale a design for performance or cost, depending on the application requirements.

Rather than being limited to just one or a few neural networks, Corerain's CAISA architecture supports nearly all of the CNN networks in use today. Corerain's RainBuilder development tools can automatically convert models developed with popular AI/ML frameworks including TensorFlow and Caffe into applications that run directly and efficiently on the FPGA-based CAISA engine.

Authors

Hao Jiang
Product Director
Corerain Technologies Co., Ltd

Shaojun Wang
Partner and COO
Corerain Technologies Co., Ltd

Xinyu Niu
Founder and CEO
Corerain Technologies Co., Ltd

Lora Luan
Marketing Director
Corerain Technologies Co., Ltd

Steve Leibson
Senior Marketing Engineering Manager
Intel® Sales and Marketing Group

CAISA: The Solution to High-Performance and Easy-to-Use AI Inference

Corerain provides the CAISA engine as IP and incorporated into FPGA-based PACs. The RainBuilder tool chain—which includes a compiler, runtime generator, and drivers —works with both versions of the CAISA engine. The available PACs include:

- The Rainman Acceleration Card based on an Intel Arria 10 SX 160 FPGA for front-end applications
- The Nebula Acceleration Card based on an Intel Arria 10 GX 1150 FPGA for edge and data center applications

CAISA-compatible PACs include boards based on the Intel Arria 10 and Intel Stratix® 10 FPGA families.

Hardware IP—The CAISA Streaming Graph Architecture

As the name indicates, the CAISA engine is a streaming-based architecture that is quite different from a traditional, instruction-based processor architecture. The latest hardware architecture from Corerain, CAISA 2.0, is a highly optimized streaming graph architecture for high-performance DL acceleration.

$$\text{MAC Efficiency} = \frac{\text{Measured Performance}}{\text{Peak Performance}}$$

We define architecture efficiency as the ratio between measured performance and the theoretical peak performance. The measured performance is the actual performance when running CNN inference, and the peak performance is calculated by the multiplication of operating frequency and the number of deployed arithmetic operators. In short, the architecture efficiency indicates on average how many percentages of deployed arithmetic operators are actively calculating at each clock cycle. While the architecture efficiency of instruction-based architectures is often between 10%-20%, the streaming architecture eliminates instruction-control hardware so that the CAISA engine is able to ensure as high as 90% (on Intel Arria 10 GX 1150) efficiency†. This high utilization guarantees that the engine can always deliver high performance with excellent energy efficiency. The streaming graph architecture employs highly parallel execution streams with no control-flow overhead, resulting in extremely low latency. Figure 1 is a block diagram of the CAISA engine.

The CAISA engine includes a full set of basic DL operators covering almost all commonly used DL models. The CAISA engine is a fully self-contained, fully functional AI accelerator that provides flexibility and scalability in a variety of AI/ML applications depending on the application's data density and the available FPGA hardware resources.

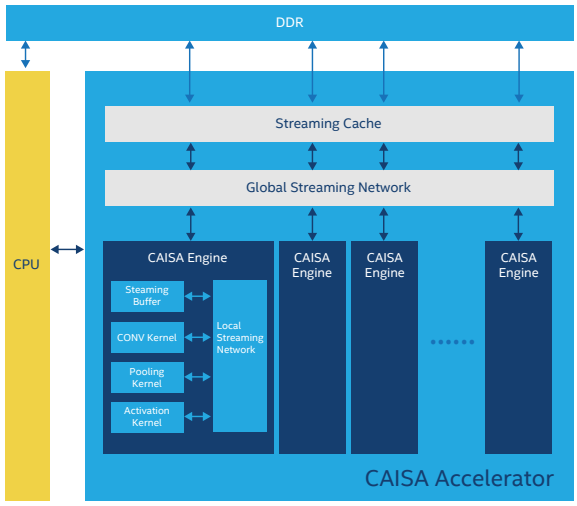

Figure 1. Block Diagram of CAISA Engine

RainBuilder – Bridging between Algorithm and Hardware

Without dedicated software tools, working with FPGA acceleration is very likely to cause headaches for algorithm and application (software) engineers. To bridge the gap, Corerain has developed the RainBuilder tool chain for its CAISA architecture. The RainBuilder tool removes the need to tinker directly with the logic resources on the FPGA. As shown in Figure 2, The RainBuilder tool consists of three major modules— a compiler, runtime, and driver.

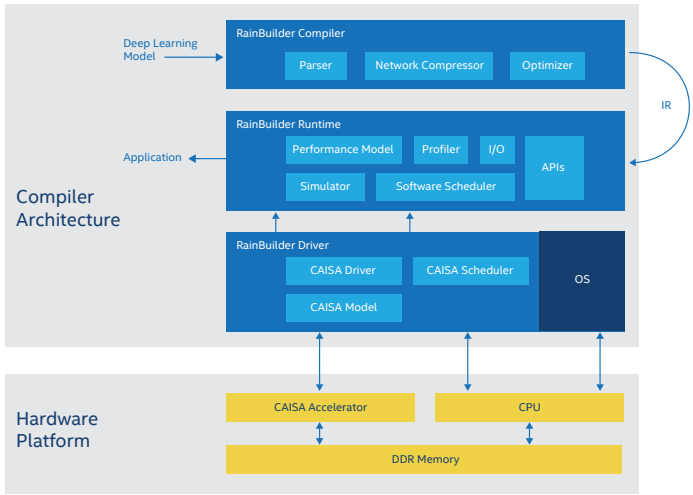

Figure 2. Block Diagram of the RainBuilder Tool

The RainBuilder Compiler

The RainBuilder compiler is an algorithm compilation module that automatically extracts structure and coefficients from a DL model. The compiler then converts the DL model to a streaming-graph intermediate representation (SG IR). The RainBuilder compiler module supports most popular DL frameworks including TensorFlow and Caffe. During conversion, the compiler optimizes the model for the CAISA architecture to guarantee efficient runtime acceleration.

The RainBuilder Runtime

The RainBuilder runtime module provides C/C++ APIs to control the resulting CAISA-based model. You can easily call these APIs to test and run an SG IR model on the CAISA engines. In addition, the runtime module can be extended with simulation kits, quantization modules, etc. The runtime module also supports customized functions, making it a convenient software kit for engineers to develop and deploy AI-based solutions.

The RainBuilder Driver

The RainBuilder driver is a transparent firmware layer that automatically handles all the operations and I/Os drivers for the CAISA engines. This layer between the hardware and the software eliminates the need for you to know about the hardware details of the CAISA architecture or FPGA design, which results in a CPU/GPU like software-development experience.

Using the RainBuilder tool, a developer can quickly transfer a DL algorithm to a CAISA-compatible SG IR set, which runs directly on the CAISA accelerator without additional hardware development. This capability dramatically improves the efficiency of high-performance, AI-application deployment on FPGA-based hardware accelerators.

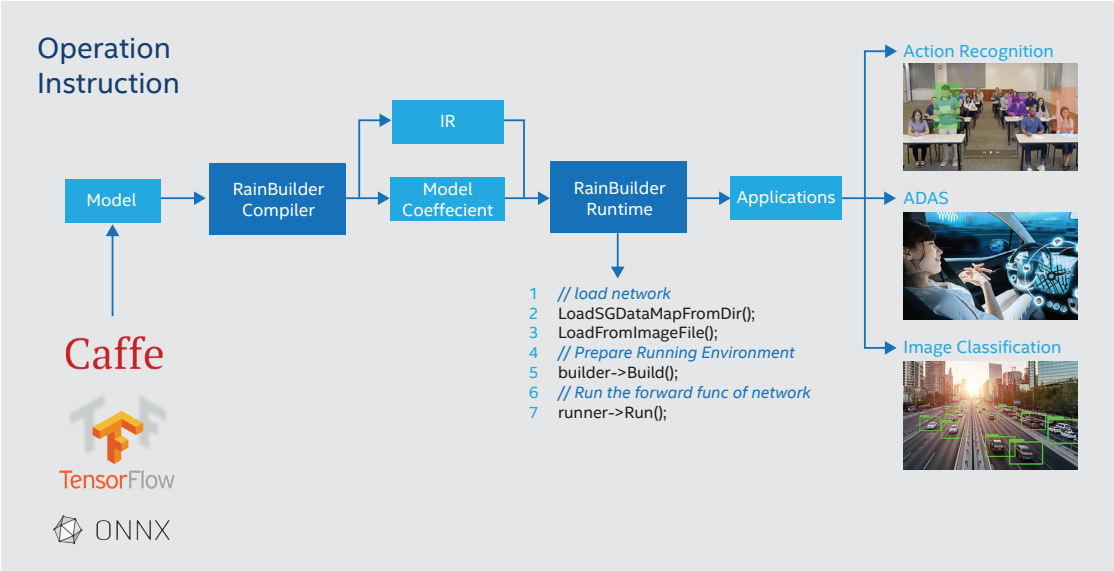

Figure 3. Developing Procedure Using the RainBuilder Tool

Developing and Deploying DL Models with CAISA

Development and deployment of DL models using CAISA is easy, with the well-defined interfaces between each system-wide layer.

As mentioned above, the RainBuilder runtime provides C/C++ APIs for users to develop and test their DL models. The runtime API is an open-source API set for developers, especially algorithm developers. This API makes it easy for a developer to test a single DL model at the algorithm level, which permits developers to estimate the performance of and to adjust their model designs quickly.

The RainBuilder API is also an efficient software interface for third-party modules. Corerain provides an I/O sub-module, called I/O Service, that takes advantage of this feature. The I/O Service module wraps the runtime as a DL acceleration service, once the algorithm has been implemented. This facility helps application engineers to quickly build system-level applications that implement both the DL models and the complicated logic and controls on top of the program.

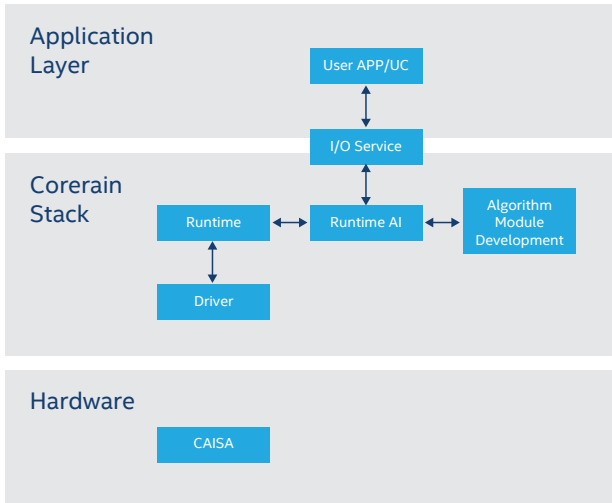

Figure 4. Block Diagram of a CAISA-Based Application Deployment

Solution Value

DL acceleration is a dynamic application area in both research and in industry. Intel FPGAs provide a competitive platform for providing programmable capability down to RTL level to meet the fast-paced development requirements for AI R&D projects. Corerain's CAISA engine and its RainBuilder tool chain make the processing power of Intel FPGAs even more accessible by offering not only high-performance, but also energy-efficient DL acceleration with a user-friendly development interface.

The CAISA engine is a highly efficient DL accelerator design that can use as much as 90% (on Intel Arria 10 GX 1150) of an FPGA's theoretical peak performance at runtime while delivering 4.5x more performance than CPUs or GPUs†. Moreover, the combination of Intel FPGAs and the CAISA engine offers significant power-consumption improvement (approximately 75% less power usage) when compared to a GPU with the same order of performance.

By using the CAISA architecture in conjunction with Intel FPGAs, you can deploy a sophisticated DL solution in just a few steps, and can easily achieve a high performance with low latency and power consumption in a compact, Intel-based system.

Solution Benefits

- Corerain's CAISA architecture is highly extensible and customizable with external FPGA modules
- Corerain's CAISA architecture maximizes the use of FPGA hardware resources: 90% (on Intel Arria 10 GX 1150) utilization†
- Corerain's RainBuilder tool set provides a CPU/GPU-like development environment
- Corerain's CAISA architecture and RainBuilder tools support most AI algorithms
- Corerain's CAISA architecture delivers high performance with low latency and low power consumption

Use Cases

Smart classroom

Thanks to the flexibility of Corerain's CAISA architecture, it is possible to squeeze this powerful DL engine to a lightweight Intel® Arria® 10 SX 160 FPGA module, which can then be integrated to a compact system based on an Intel Architecture (IA) processor.

Figure 5 shows an application for smart classroom. The model running on the CAISA engine is a multi-gesture classifier that can recognize people standing up, raising their hand, resting their head on a table, or just looking around. With this AI-enabled system installed in front of a classroom, students' in-lesson behavior statistics can be used to provide feedback for teaching quality. For example, a limited amount of hand raising and standing up might indicate a lack of interaction between teachers and students. As a result, students might gradually lose their interest in a course.

Figure 5. Smart Class OPS System

BRIEF SUMMARY

- End-to-end gesture detection based on CAISA
- Compact and easy to integrate
- 4-gesture analysis:
  - Standing up
  - Raising hand
  - Resting head on table
  - Looking around
- Object detection, tracking, feature extraction, etc.

PERFORMANCE

- HD video gesture detection
- Model input size: 256x256 pixels
- Minimum object detected: 60x60 pixels @ 1080P
- Accuracy up to 86%[1]

City Surveillance – Multi-Channel, Real-Time Object Detection

By applying the CAISA architecture to a mid-range Intel Arria 10 SX 660 FPGA and adding it to a Network Video Recorder (NVR) system based on an Intel Core® i3 processor transforms the system into an AI-enabled, smart-edge computing module. This system can perform for DL-based image processing in real time on as many as 16 video channels streamed from cameras, including object detection, feature extraction, and so on. As shown in Figure 6, sixteen IP cameras are set up as the video source of the NVR, and a single Intel Arria 10 FPGA card is equipped to detect people captured on each of the cameras with real-time display feedback of all 16 video streams.

Figure 6. 16-Channel Real-Time Object Detection

BRIEF SUMMARY

- AI edge server solution for smart city, smart campus, etc.
- Compatible with traditional non-AI surveillance system
- Low-power and lightweight DL accelerator
- Object detection, tracking, feature extraction

PERFORMANCE

- Object detection: 16-channel HD video at 12.5 FPS/channel
- Minimum object detected: 60x60 pixels at 1080P
- High accuracy up to 99.3%†
- End-to-end tool chain available for FPGA deployment (TensorFlow and Caffe supported)

Conclusion

Corerain has developed a high-performance, low-power AI acceleration engine called CAISA, which can tap as much as 90% (on Intel Arria 10 GX 1150) of an FPGA's full performance potential without the need for HDL programming. Using Corerain's CAISA engine and the associated RainBuilder end-to-end tool chain, AI/ML application developers can now take advantage of FPGA-level application performance while using familiar deep-learning frameworks such as TensorFlow and Caffe. Corerain's solution allows developers to quickly deploy high-performance, low-latency, and power-efficient AI inference accelerators at low cost. These AI accelerators are well-suited to both edge and data center applications, including surveillance, smart city, smart education, smart industry, big-data analytics, high-volume target detection, and more.

Where to Get More Information

- Discover Intel FPGAs at www.intel.com/fpga. Find out more about Intel innovation for AI at www.intel.com/fpga-ai
- For more information about Corerain's AI acceleration solutions, visit https://community-en.corerain.com or contact Corerain directly at sales@corerain.com.

Intel does not control or audit third-party data.You should review this content, consult other sources, and confirm whether referenced data are accurate.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors.

Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

Performance results are based on testing as of March 2019 and may not reflect all publicly available security updates. See configuration disclosure for details. No product or component can be absolutely secure.

§ CONFIGURATIONS

SPEC.	OPS	NVR
CPU	Intel® Core™ i3-8100	Intel Core i7-7700
FPGA	Intel Arria® 10 SX 160	Intel Arria 10 SX 660
Memory	DDR4 8 GB	DDR4 16 GB
Storage	M.2 128 GB	mSATA 32 GB

The OPS was tested with 1 USB webcam input (Logitech V-U0028 Webcam) and the NVR was tested with 16 IP camera inputs (Hikvision DS-2CD1221(D)-I3).

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. Check with your system manufacturer or retailer or learn more at [intel.com].

† Tests measure performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. Consult other sources of information to evaluate performance as you consider your purchase. For more complete information about performance and benchmark results, visit www.intel.com/benchmarks.