What are FPGAs?

Mahmut Taylan Kandemir

CSE 531

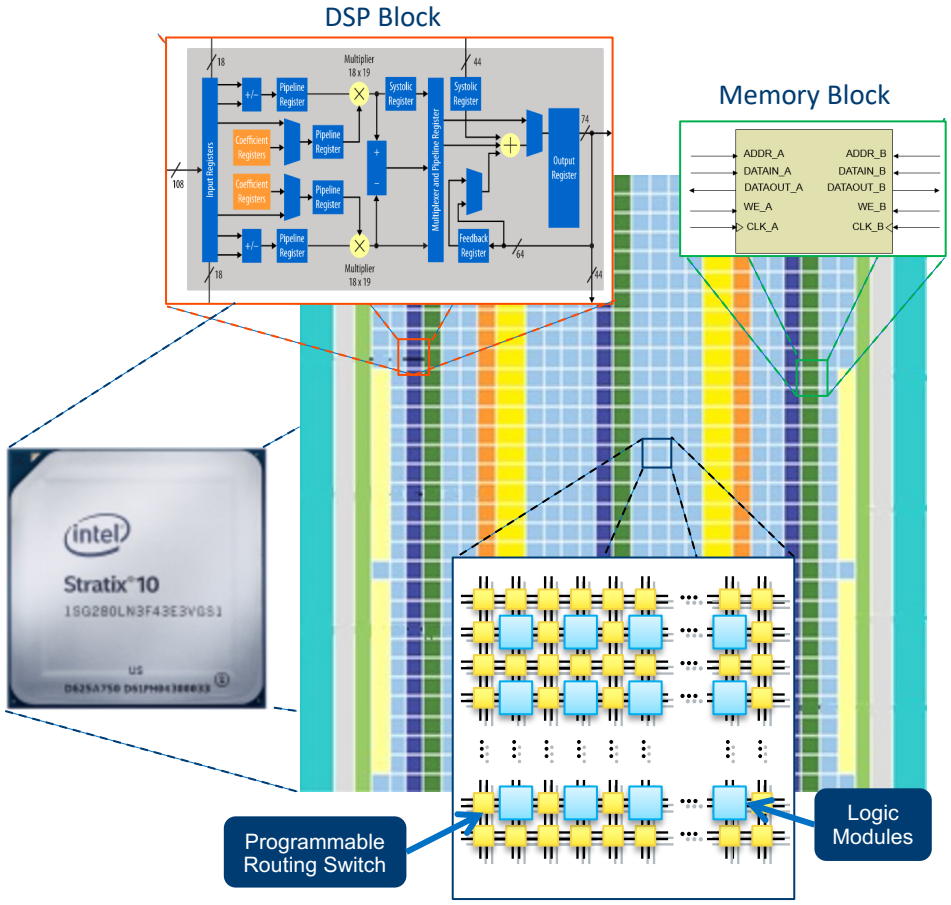
ACK: INTEL

FPGA Architecture

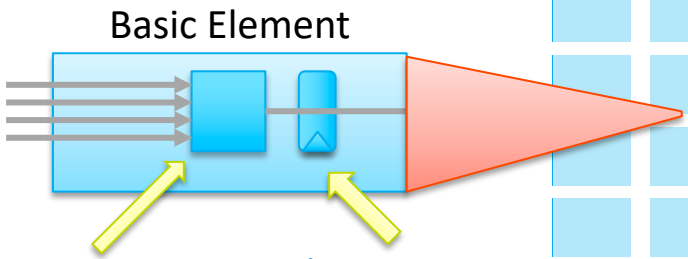
Field Programmable Gate Array (FPGA)

- Millions of logic elements
- Thousands of embedded memory blocks
- Thousands of DSP blocks
- Programmable interconnect
- High speed transceivers
- Various built-in hardened IP
- Soft cores

Used to create **Custom Hardware!**



FPGA Architecture: Basic Elements

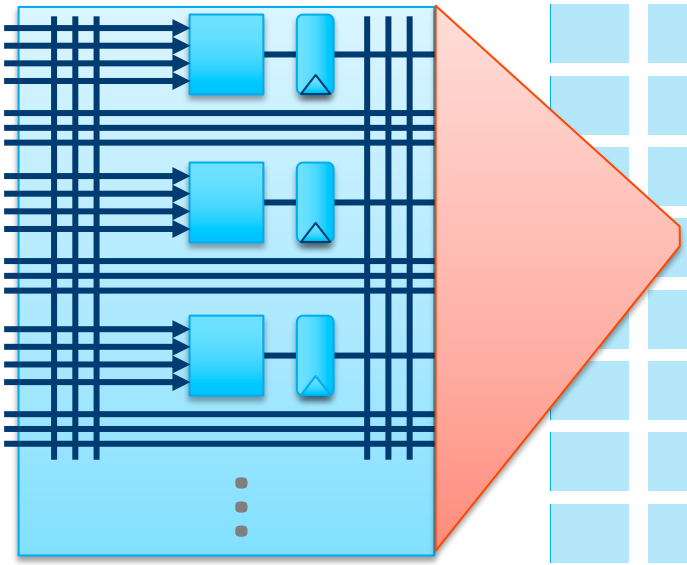


1-bit configurable operation

1-bit register (store result)

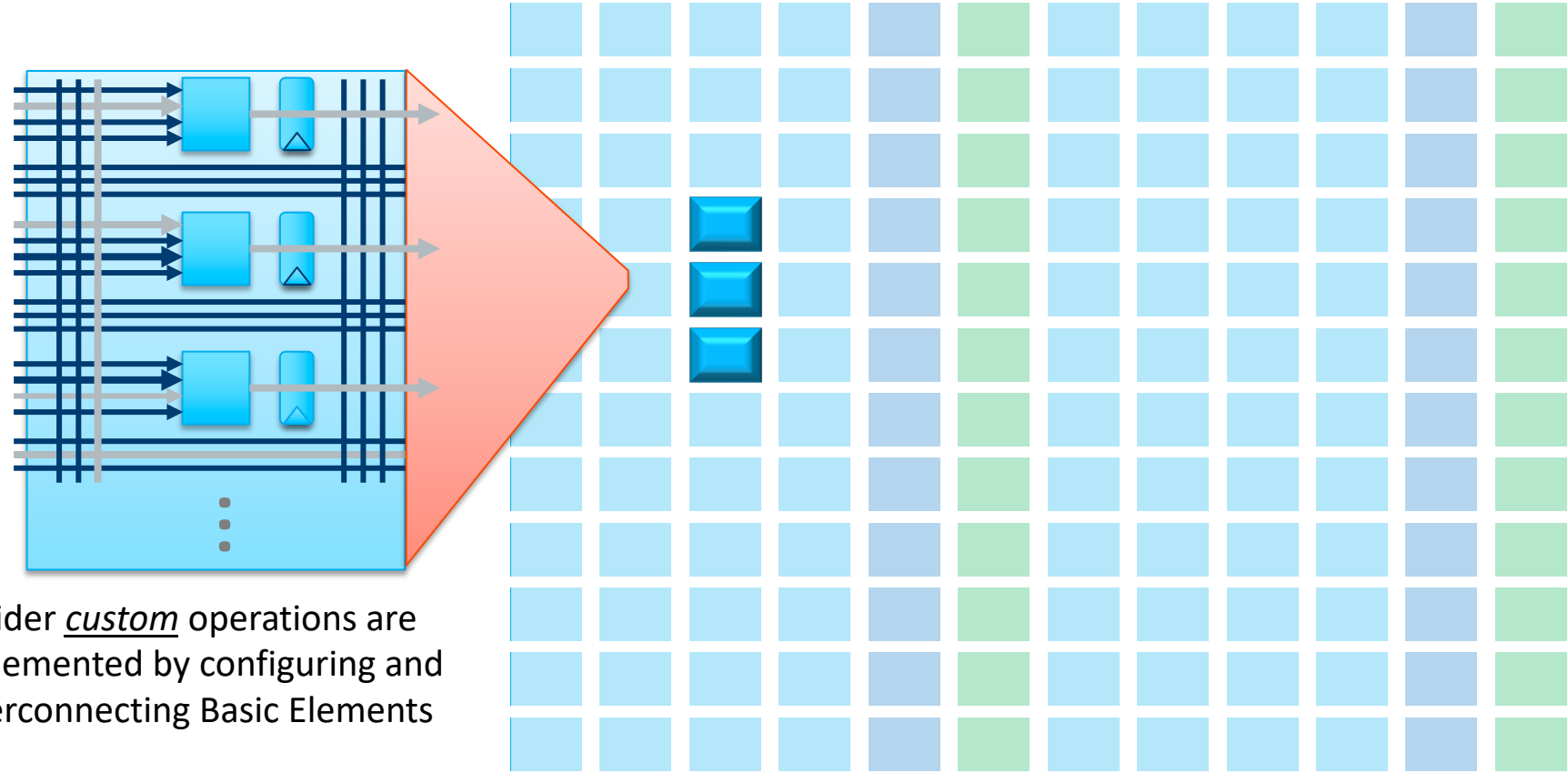
Configured *to perform any 1-bit operation:*
AND, OR, NOT, ADD, SUB

FPGA Architecture: Flexible Interconnect



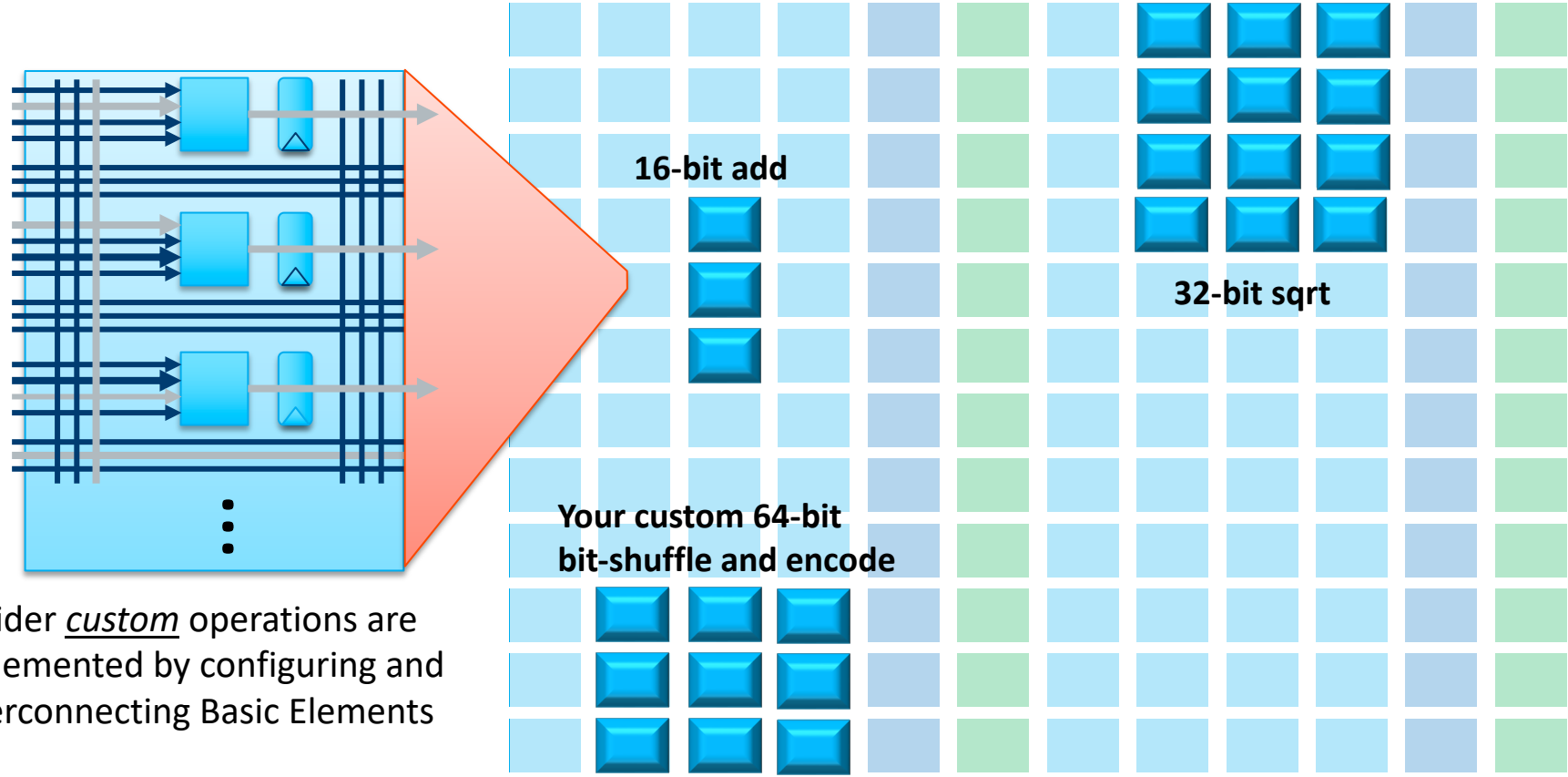
Basic Elements are surrounded
with a
flexible interconnect

FPGA Architecture: Flexible Interconnect



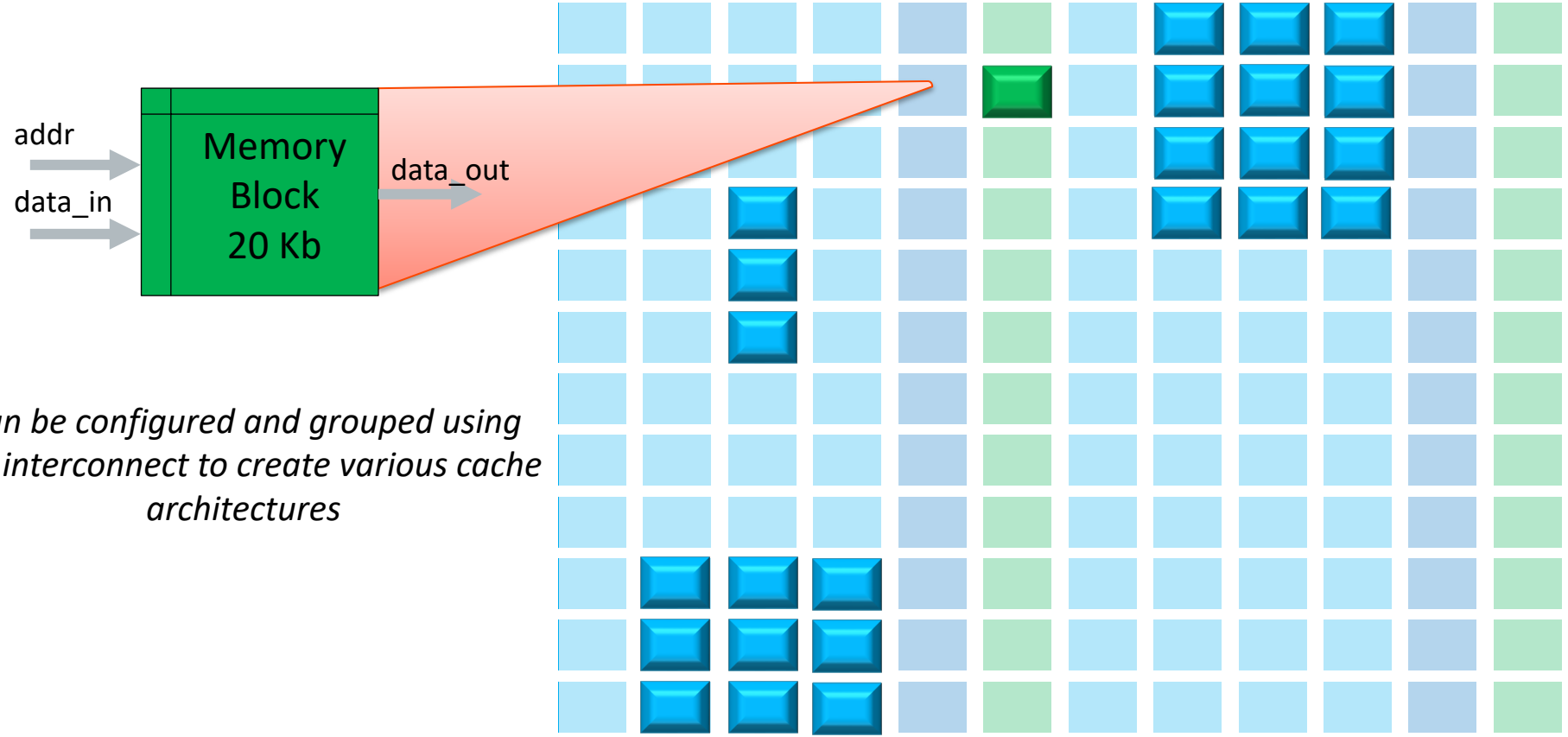
Wider *custom* operations are implemented by configuring and interconnecting Basic Elements

FPGA Architecture: Custom Operations Using Basic Elements



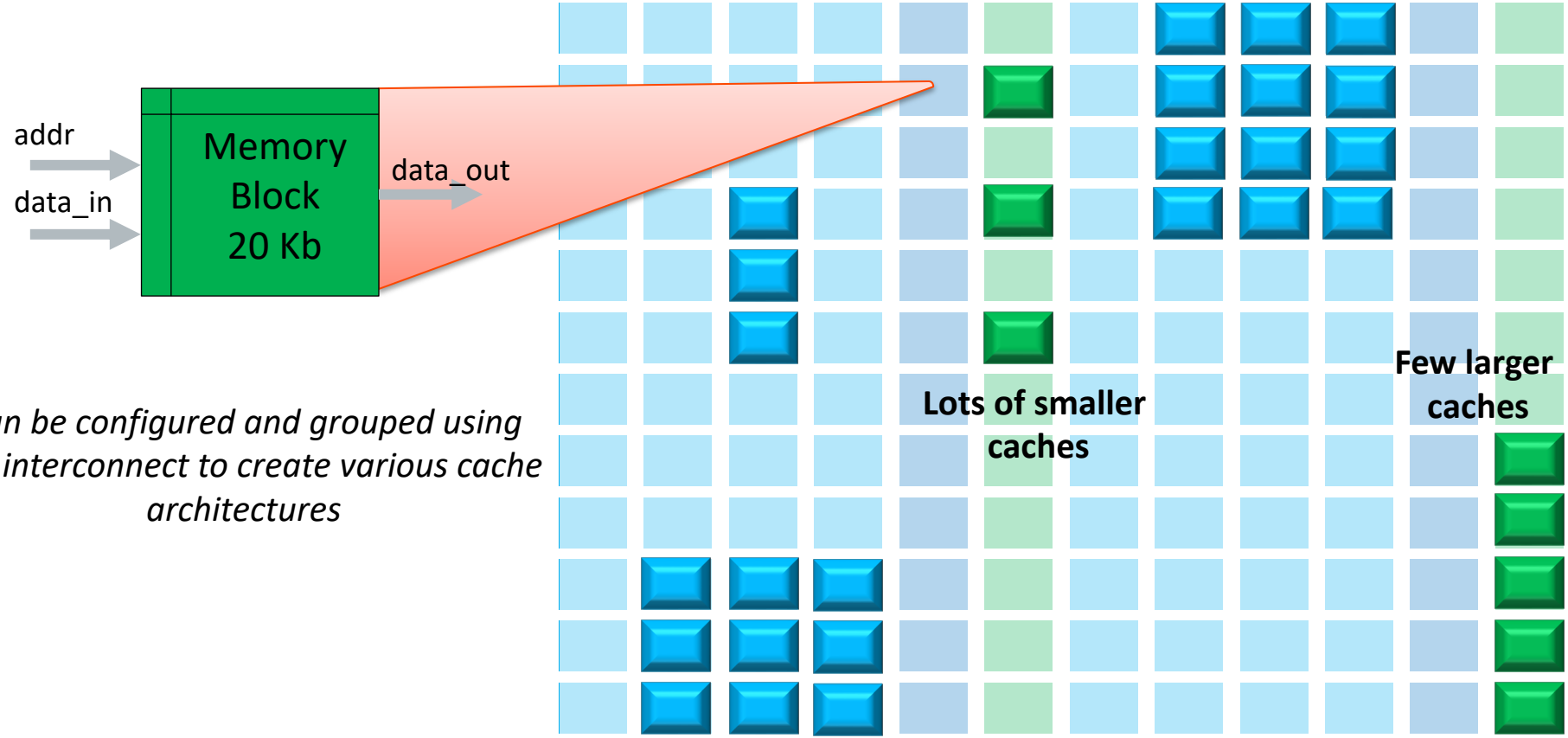
Wider *custom* operations are implemented by configuring and interconnecting Basic Elements

FPGA Architecture: Memory Blocks



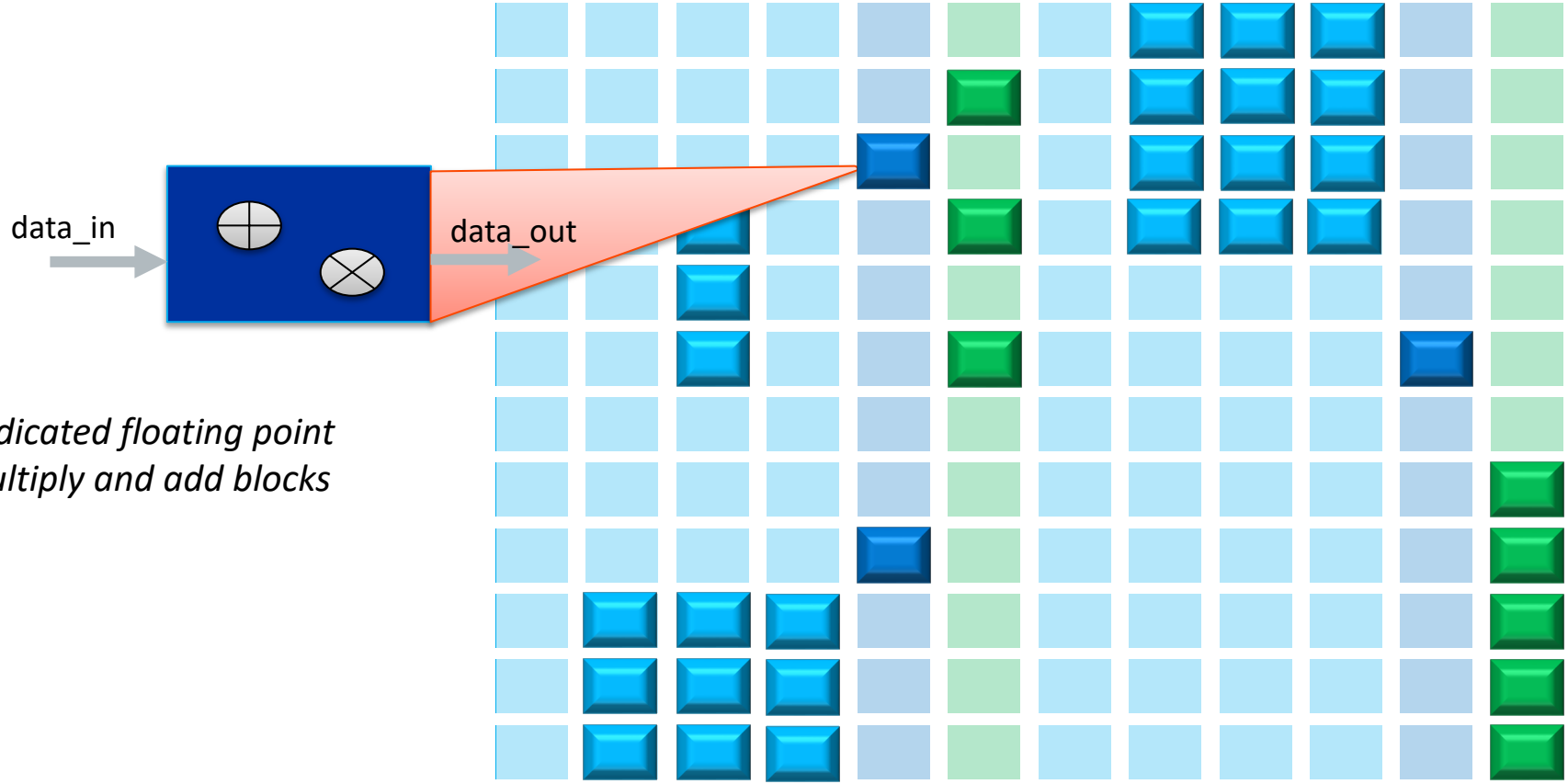
Can be configured and grouped using the interconnect to create various cache architectures

FPGA Architecture: Memory Blocks



Can be configured and grouped using the interconnect to create various cache architectures

FPGA Architecture: Floating Point Multiplier/Adder Blocks

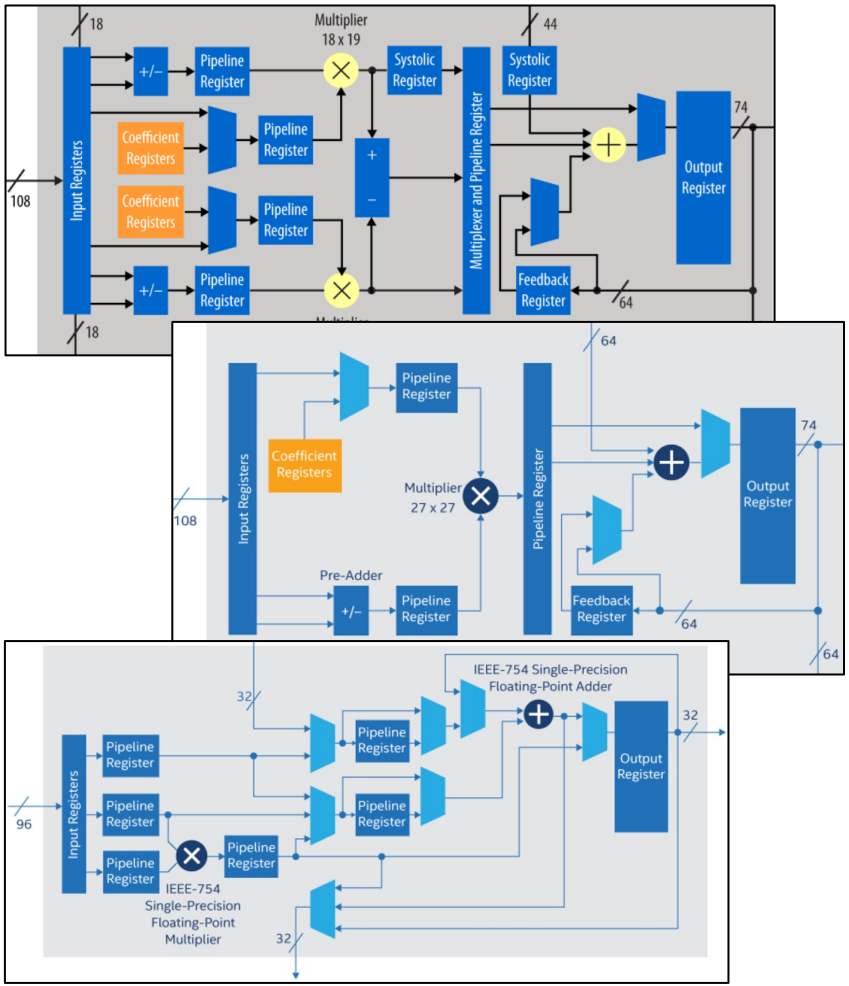


Dedicated floating point multiply and add blocks

DSP Blocks

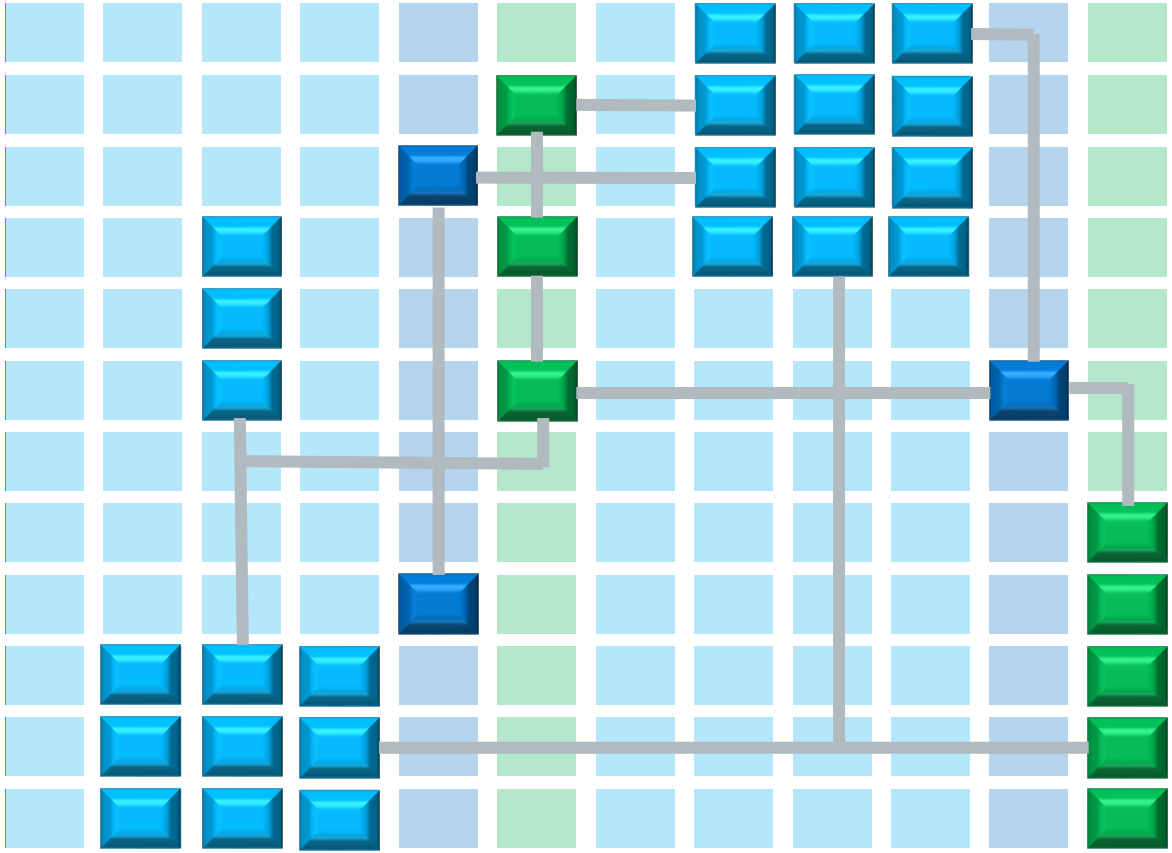
Thousands Digital Signal Processing (DSP) Blocks in Modern FPGAs

- Configurable to support multiple features
  - Variable precision fixed-point multipliers
  - Adders with accumulation register
  - Internal coefficient register bank
  - Rounding
  - Pre-adder to form tap-delay line for filters
  - Single precision floating point multiplication, addition, accumulation



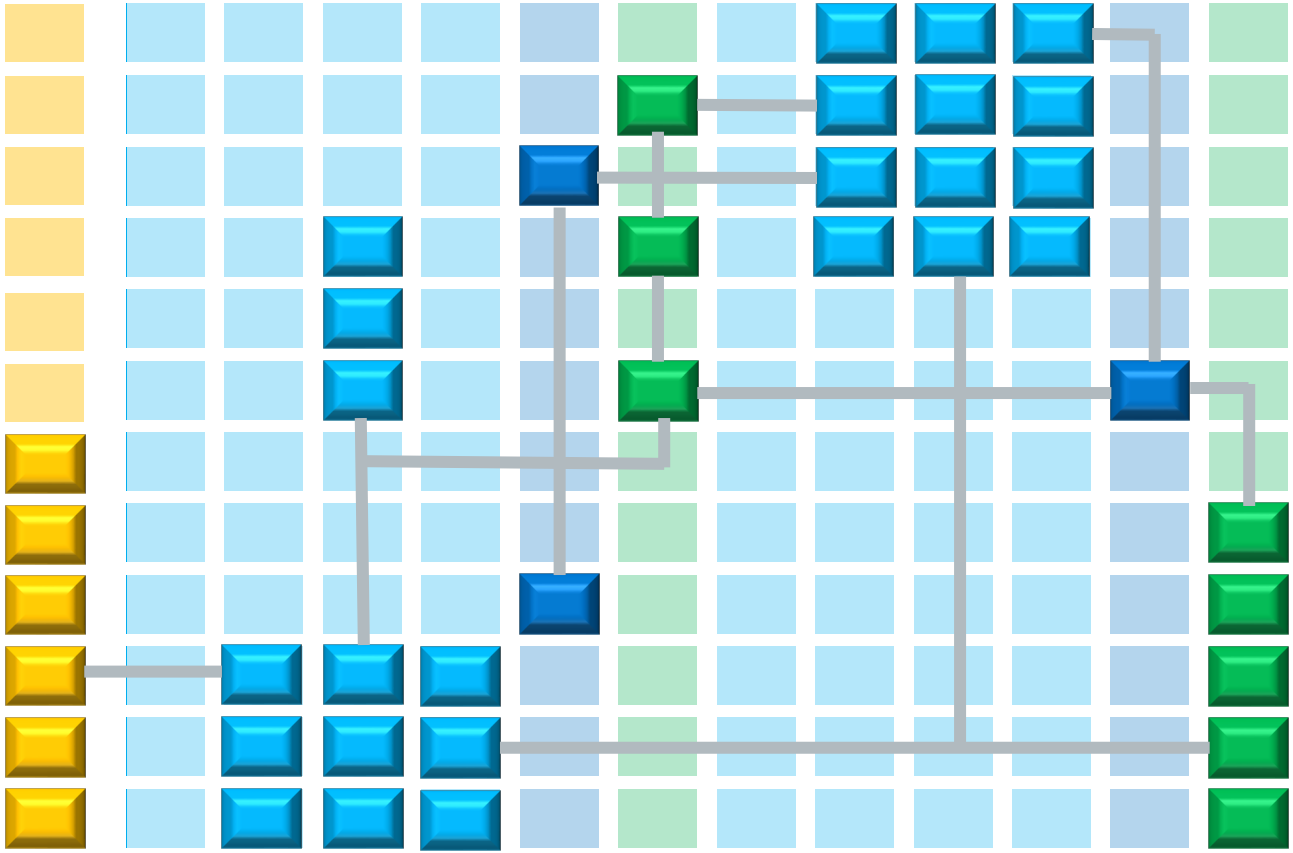
FPGA Architecture: Configurable Routing

*Blocks are connected into a **custom data-path** that matches your application.*



FPGA Architecture: Configurable IO

*The **Custom data-path** can be connected directly to **custom or standard IO interfaces** for inline data processing*



FPGA I/Os and Interfaces

Hardened Memory Controllers

- Available interfaces to off-chip memory such as HBM, HMC, DDR SDRAM, QDR SRAM, etc.

High-Speed Transceivers

- Provide any variety of protocols for moving data in and out of the FPGA

Hard IP for PCI Express standard

Phase Lock Loops (PLLs)

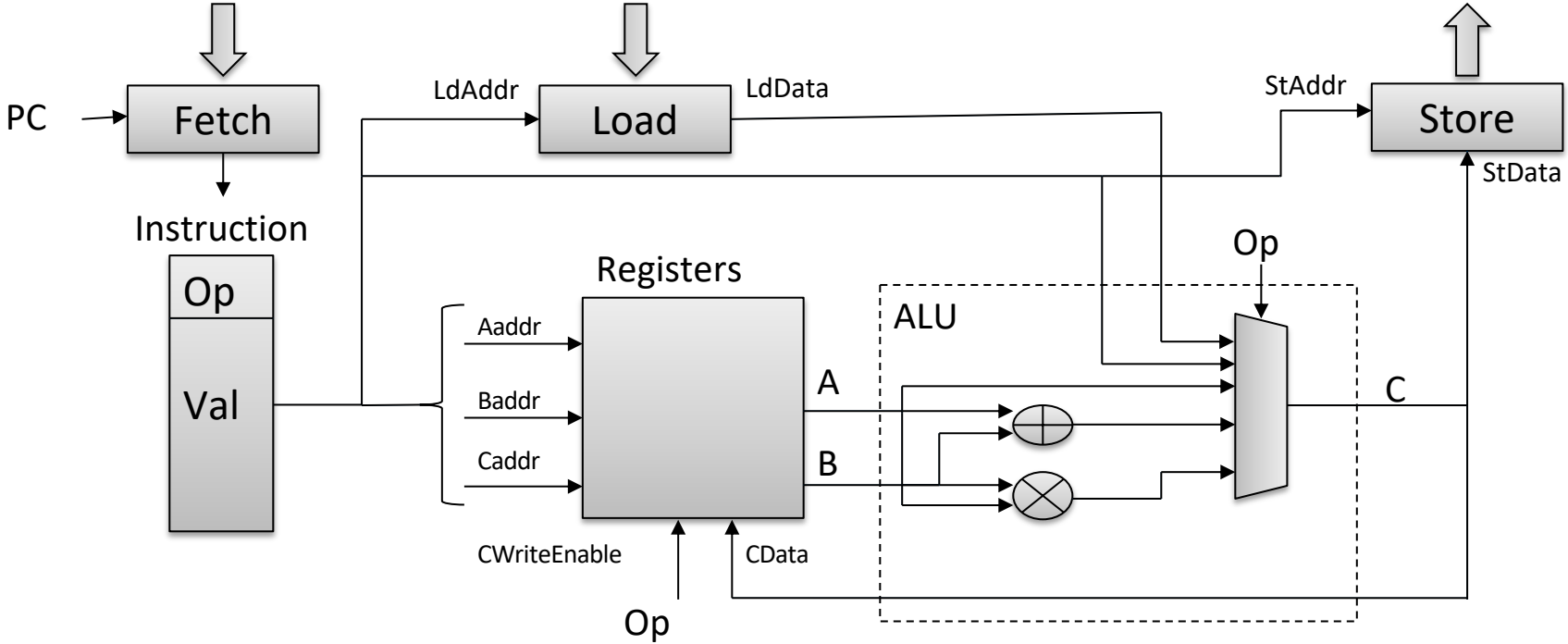
Mapping a Simple Program to an FPGA

Mem[100] += 42 * Mem[101]

CPU instructions

R0 ← **Load** Mem[100]
R1 ← **Load** Mem[101]
R2 ← **Load** #42
R2 ← **Mul** R1, R2
R0 ← **Add** R2, R0
Store R0 → Mem[100]

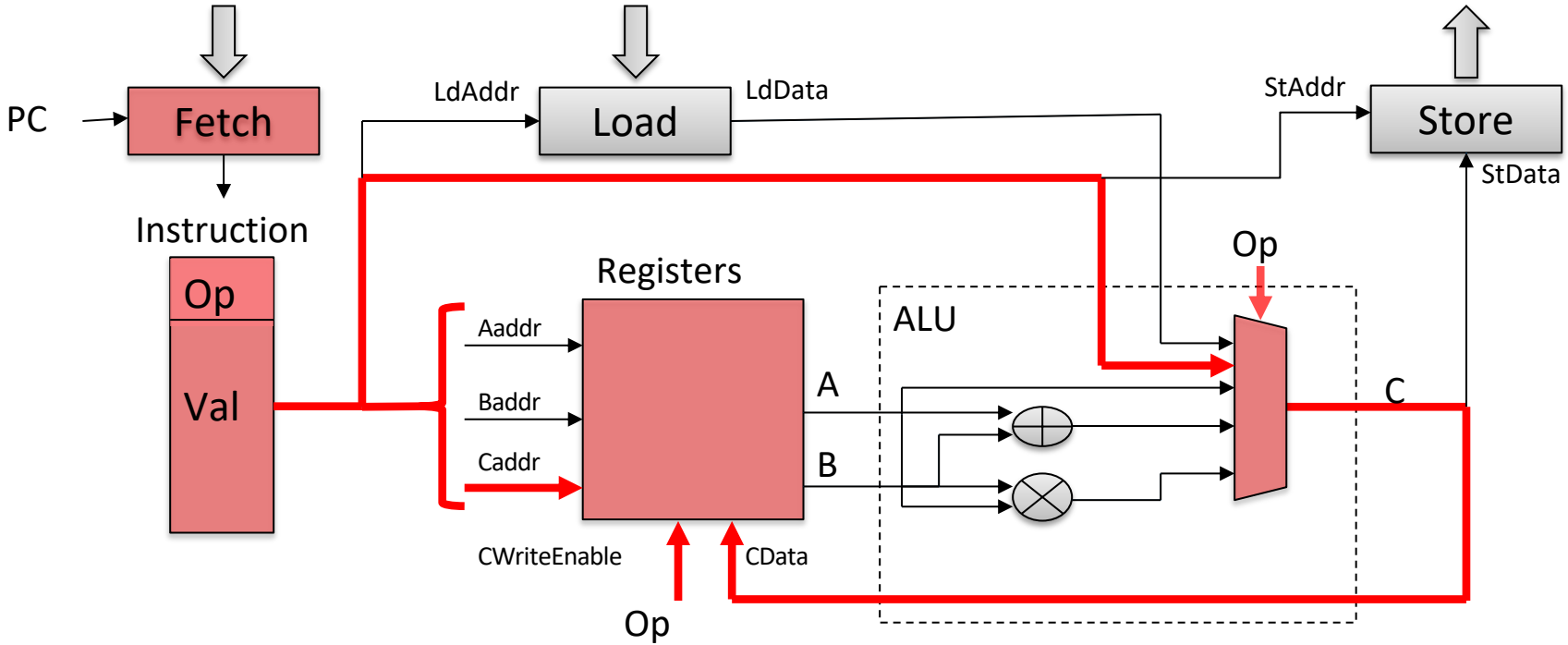
First let's take a look at execution on a *simple* CPU



Fixed and general architecture:

- **General "cover-all-cases" data-paths**
- **Fixed data-widths**
- **Fixed operations**

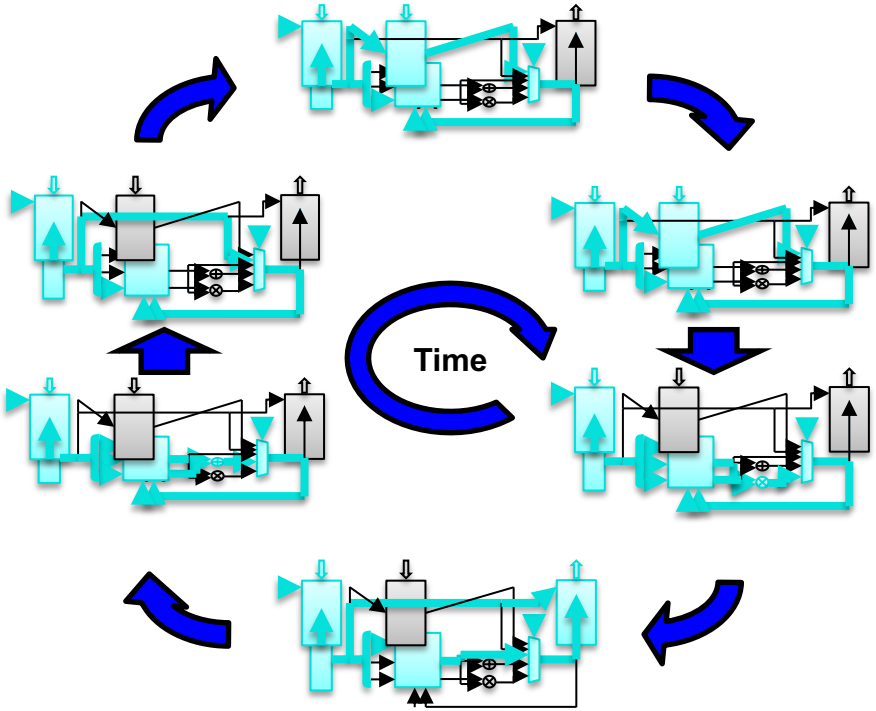
Looking at a Single Instruction



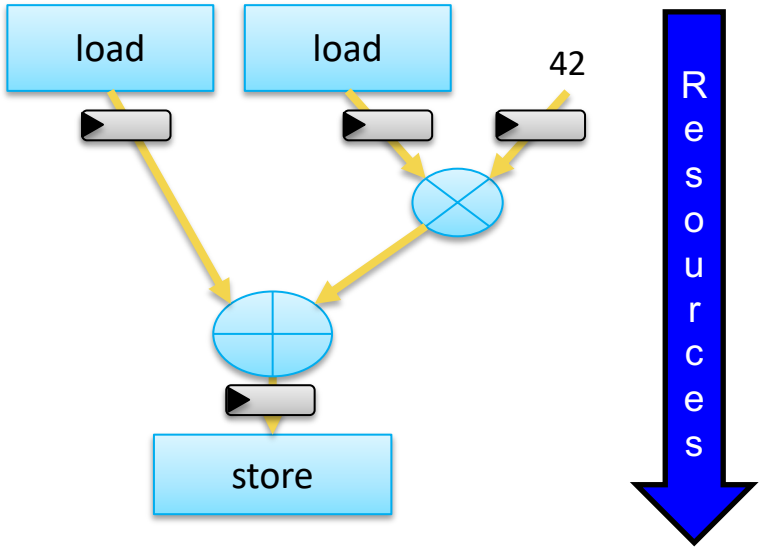
Very inefficient use of hardware!

Sequential Architecture vs. Dataflow Architecture

Sequential CPU Architecture



FPGA Dataflow Architecture

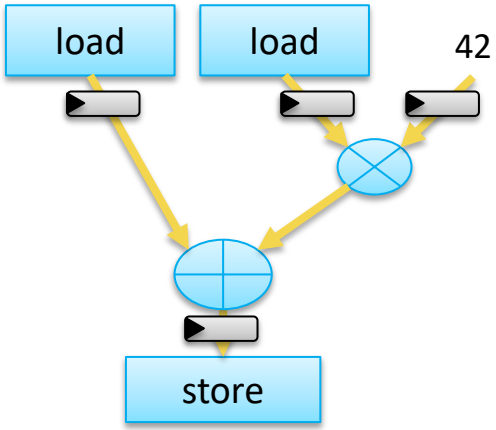


Custom Data-Path on the FPGA Matches Your Algorithm!

High-level code

```
Mem[100] += 42 * Mem[101]
```

Custom data-path



Build exactly what you need:

Operations

Data widths

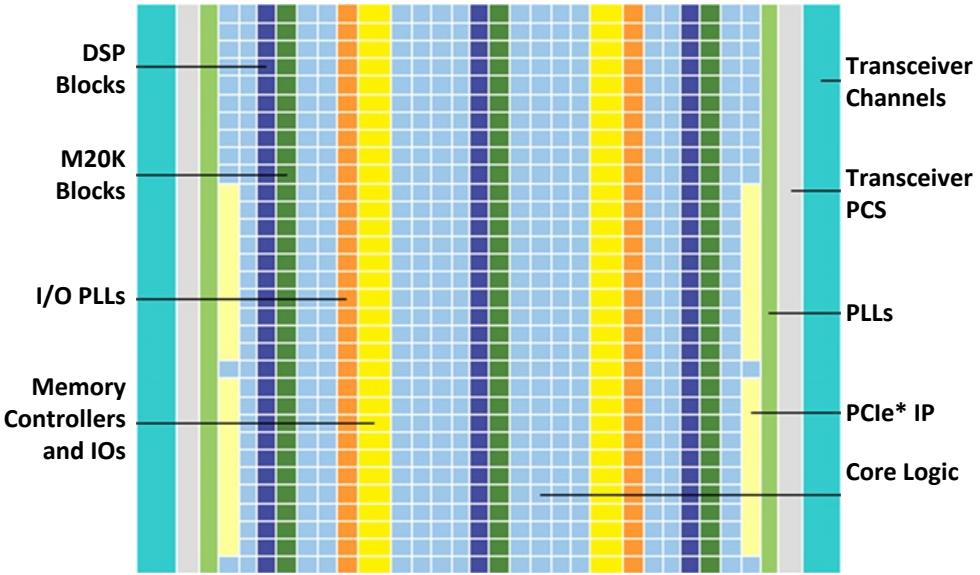
Memory size & configuration

Efficiency:

Throughput / Latency / Power

Advantages of Custom Hardware with FPGAs

- **Custom hardware!**
- Efficient processing
- Fine-grained parallelism
- Low power
- Flexible silicon
- Ability to reconfigure
- Fast time-to-market
- Many available I/O standards

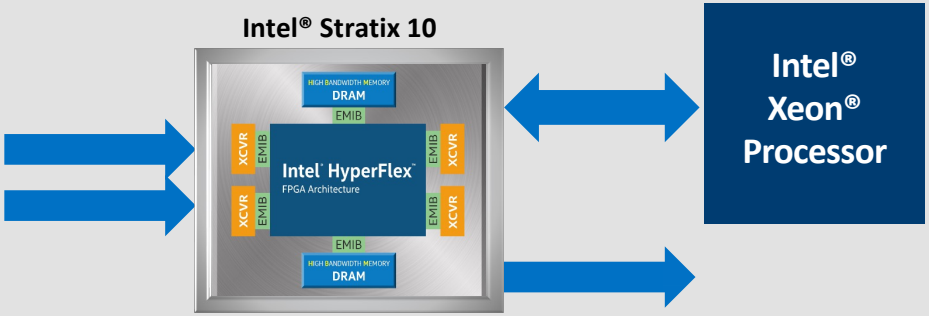


Why FPGAs for DL Inference

FPGAs Provide Flexibility to Control the Datapath

Compute Acceleration/Offload

- Workload agnostic compute
- FPGAaaS
- Virtualization



Inline Data Flow Processing

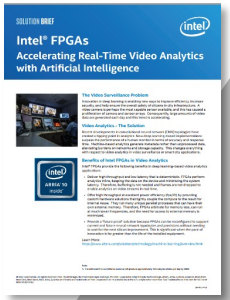
- Machine learning
- Object detection and recognition
- Advanced driver assistance system (ADAS)
- Gesture recognition
- Face detection

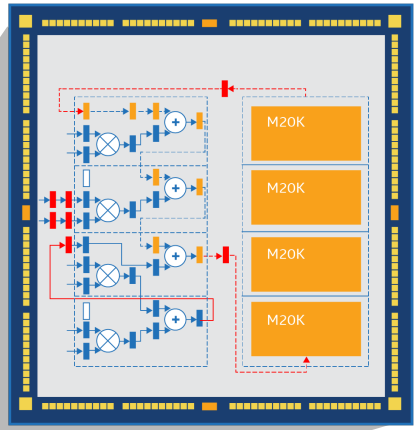
Storage Acceleration

- Machine learning
- Cryptography
- Compression
- Indexing

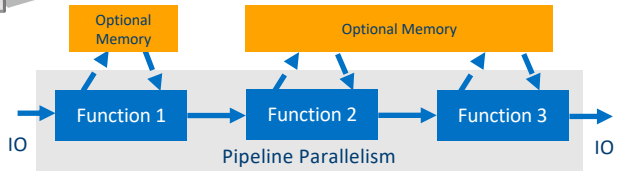
Why FPGAs for Machine Learning?

Convolutional Neural Networks are Compute Intensive





Fine-grained & low latency between compute and memory

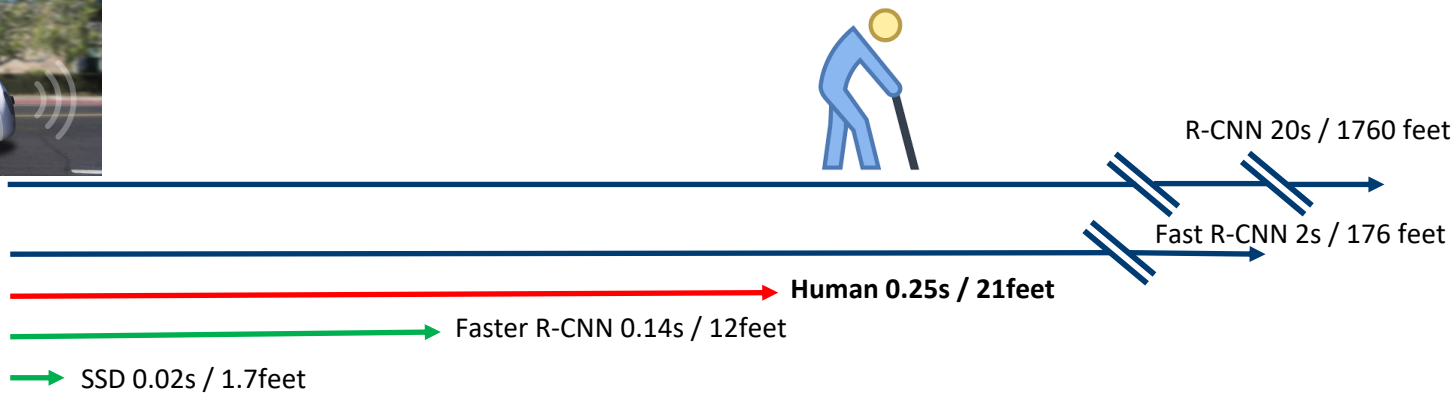


Feature	Benefit
Highly parallel architecture	Facilitates efficient low-batch video stream processing and reduces latency
Configurable Distributed Floating Point DSP Blocks	FP32 9Tflops, FP16, FP11 Accelerates computation by tuning compute performance
Tightly coupled high-bandwidth memory	>50TB/s on chip SRAM bandwidth, random access, reduces latency, minimizes external memory access
Programmable Data Path	Reduces unnecessary data movement, improving latency and efficiency
Configurability	Support for variable precision (trade-off throughput and accuracy). Future proof designs, and system connectivity

Deterministic Latency Matters for Inference

Automotive example:

- Latency impacts response time and distance
- Factors that impact latency – batch size / IO latency
- Need to perform better than human

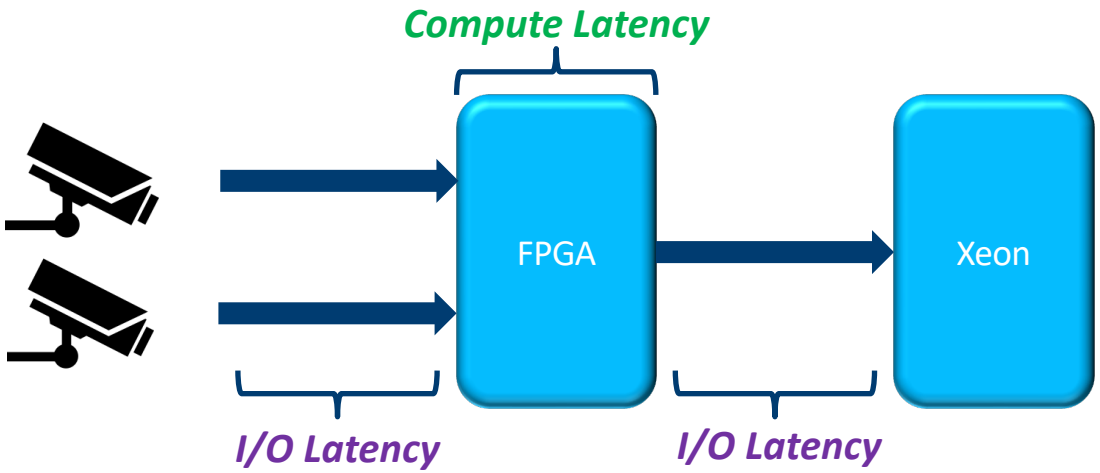


FPGAs Provide Deterministic System Latency

FPGAs leverages parallelism across the entire chip to reduce compute latency

FPGAs has flexible and customizable IOs with low & deterministic I/O latency

System Latency** = **I/O Latency** + **Compute Latency



FPGA Flexibility Supports Arbitrary Architectures

Many efforts to improve efficiency in network development around limitations of GPU

- Batching
- Reduce bit width
- Sparse weights
- Sparse activations
- Weight sharing
- Compact network

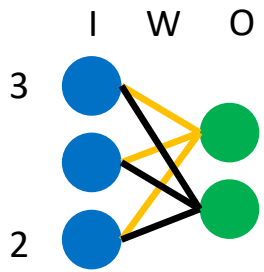
LeNet [IEEE] AlexNet [ILSVRC'12] VGG [ILSVRC'14] GoogleNet [ILSVRC'14] ResNet [ILSVRC'15] XNORNet

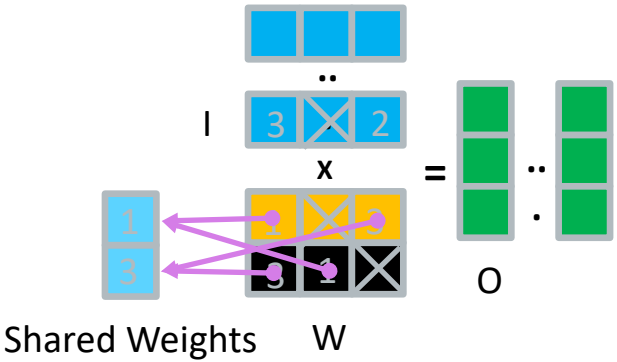
BinaryConnect [NIPS'15] TernaryConnect [ICLR'16]

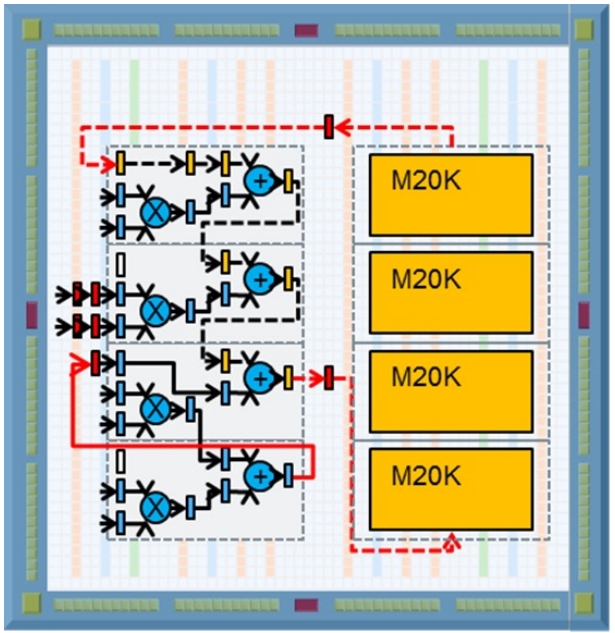
Spatially SparseCNN [CIFAR-10 winner '14] Pruning [NIPS'15] SparseCNN [CVPR'15]

DeepComp [ICLR'16] HashedNets [ICML'15]

SqueezeNet







CNN Inference Implementation Requirements

High throughput, feed forward data flow

Many floating point multiplies and accumulate operations

- e.g. 8 TFLOP performance in Stratix 10

High bandwidth local storage for filter data and partial sums

- e.g. 58 TB/s internal memory bandwidth in Stratix 10

Flexibility for different topologies and different problems

Summary

Deep Learning (DL) is a type of machine learning for extracting patterns from data using neural networks

DL neural networks are built and trained using frameworks and combining various layers

FPGAs are made up of a variety of building blocks

Through FPGA development tools, one can translate **code** into **custom hardware**

FPGAs provide a flexible, deterministic low-latency, high-throughput, and energy-efficient solution for accelerating the constantly changing networks and precisions for DL inference