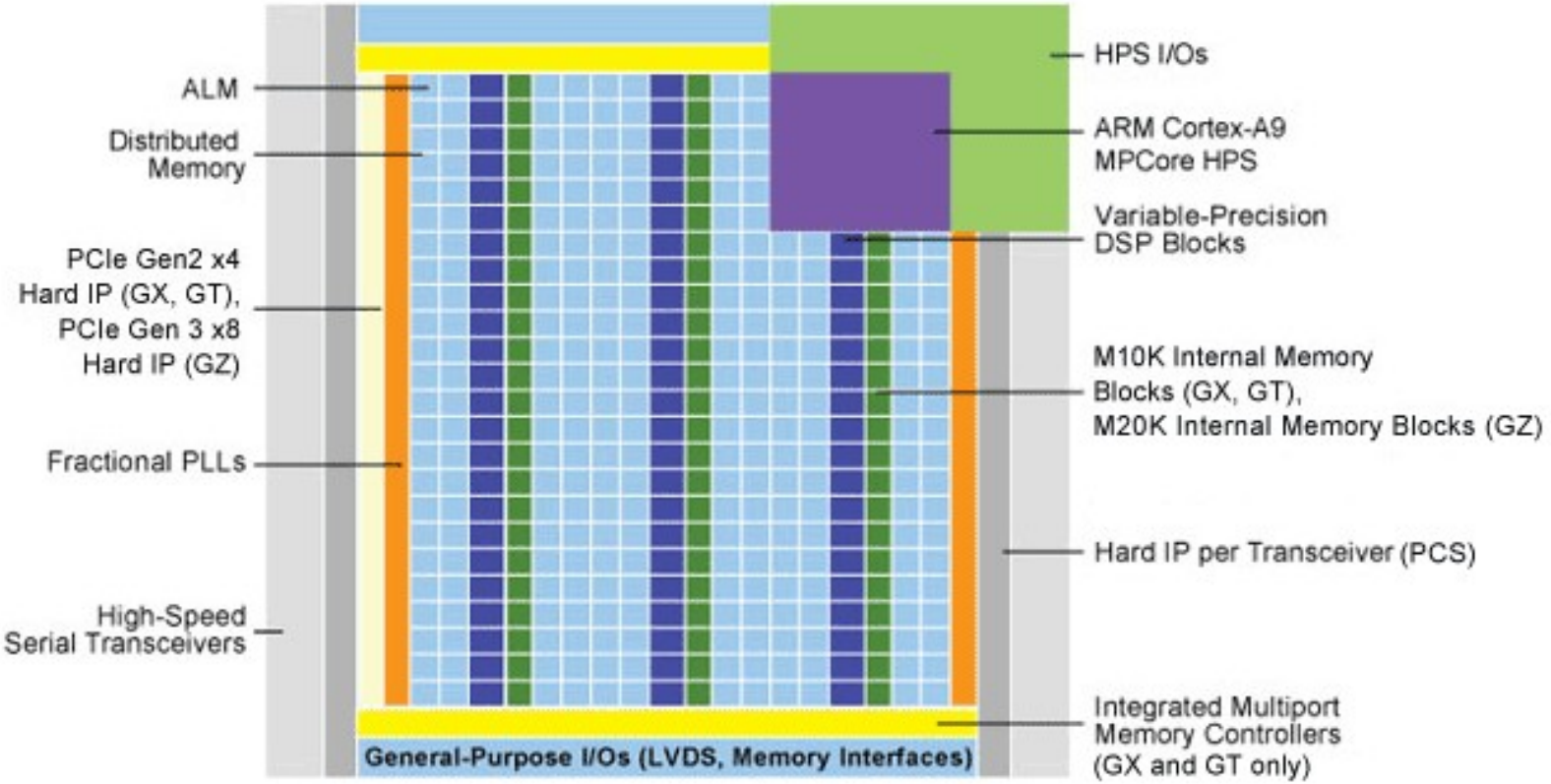
Hardware and Software for AI
FPGA

Tushar B. Kute,
http://tusharkute.com

FPGA

- The acronym FPGA stands for Field Programmable Gate Array. It is an integrated circuit that can be programmed by a user for a specific use after it has been manufactured.
- Contemporary FPGAs contain adaptive logic modules (ALMs) and logic elements (LEs) connected via programmable interconnects.
- These blocks create a physical array of logic gates that can be customized to perform specific computing tasks. This makes them very different from other types of microcontrollers or Central Processing Units (CPUs), whose configuration is set and sealed by a manufacturer and cannot be modified.

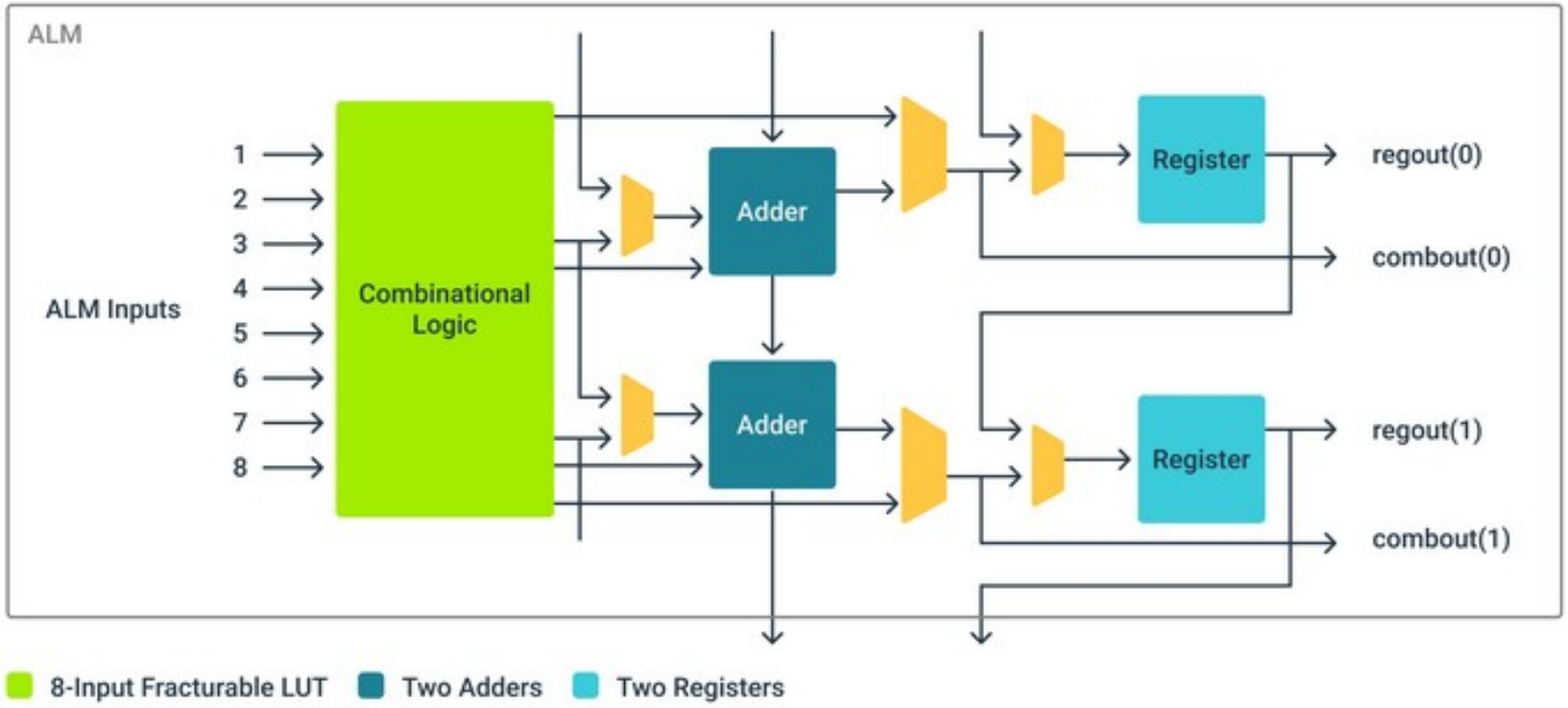
FPGA


ALM
Distributed Memory
PCIe Gen2 x4 Hard IP (GX, GT), PCIe Gen 3 x8 Hard IP (GZ)
Fractional PLLs
High-Speed Serial Transceivers
HPS I/Os
ARM Cortex-A9 MPCore HPS
Variable-Precision DSP Blocks
M10K Internal Memory Blocks (GX, GT), M20K Internal Memory Blocks (GZ)
Hard IP per Transceiver (PCS)
Integrated Multiport Memory Controllers (GX and GT only)
General-Purpose I/Os (LVDS, Memory Interfaces)

FPGA

- The first programmable circuits were very simple and contained only logic gates. This was enough to perform many logical functions where zeros and ones were the inputs and outputs.
- With time, programmable circuits were becoming more and more powerful. In programmable circuits, you program logic cells that can work as registers, adders, multiplexers or lookup tables. How the cells work and its structure can both be changed while the circuit is working.
- A circuit can be reprogrammed to perform different functions, for example, that of a processor in the ARM architecture, a network interface card or a video encoder, to name three.

FPGA


ALM
ALM Inputs
1
2
3
4
5
6
7
8
Combinational Logic
Adder
Adder
Register
Register
regout(0)
combout(0)
regout(1)
combout(1)
8-Input Fracturable LUT
Two Adders
Two Registers

How FPGA works?

- FPGAs consist of logical modules connected by routing channels. Each module is made up of a programmable lookup table that is used to control the elements that each cell consists of and to perform logical functions of the elements that make up the cell.
- In addition to the lookup table, each cell contains cascaded adders enabling addition to be done. Subtraction can also be done by changing the logical states of the input.
- Beyond these, there are also registers (logical elements used to perform the simplest memory functions) and multiplexers (switching elements).

How FPGA works?

- FPGAs can also include static and dynamic on-chip memories, depending on the specific manufacturer model.
- In addition, in FPGAs you can find ready components, such as CPU cores, memory controllers, USB controllers or network cards.
- These components are so popular that there is no need to implement them in the FPGA structure. Instead, you can use an already manufactured component.

Programming FPGA

- FPGAs are mainly used to design application-specific integrated circuits (ASICs). First, you design the architecture of such a circuit.
- Then, you use an FPGA to build and check its prototype. Errors can be corrected. Once the prototype works as expected, an ASIC project is created and manufactured based on the FPGA design.
- This allows you to save time, as manufacturing an integrated circuit can be a very complex and time-consuming process.
- It also saves money, as one FPGA can be used to prepare many iterations of the same project.

Programming FPGA

- FPGAs are also used in real-time systems where response time plays a crucial role.
- In standard CPUs, response time is not set and you do not know precisely when you will receive a response after the initial signal appears.
- To minimize or keep it within a given range, real-time operating systems are used. Still, in the scenarios where a fast response time (under milliseconds) is necessary, this falls short.
- To solve this problem, the requested algorithm needs to be implemented in FPGA using combinational or sequential logic to ensure a response time that is always the same and under milliseconds.

Programming FPGA

- The notion of “FPGA programming” may be a little misleading. After all, there is no real program to run sequentially, such as CPUs or GPUs both have.
- FPGA programming consists in creating hardware architecture that will execute a requested algorithm and describe it in a hardware description language (HDL).
- Consequently, the building blocks of this algorithm will not be a memory register and a set of operations to be performed, as with standard programs executed by CPUs or GPUs. An “FPGA program” will consist of low-level elements including logic gates, adders, registers and multiplexers.

Hardware acceleration

- Hardware acceleration is a main FPGA use case. In a nutshell, repetitive and compute-intensive tasks are offloaded from a computer or a server to dedicated hardware such as FPGAs. Tasks that usually fall to CPUs are offloaded to hardware.
- Enabling display graphics, Graphic Processing Units (GPUs) are the most popular and widely used hardware for this type of operation.
- Of course, they can also be used to perform computations, but only of a specific type.

Hardware acceleration

- Acceleration with FPGAs works in the same way as hardware acceleration. The only difference lies in the implementation.
- From the server's point of view, both types of acceleration are the same. The main advantage of FPGA technology is its flexibility. It is easy to change the hardware acceleration even for hardware currently in use.
- You can also release updates or have many implementations that work on the same board.

Future of FPGA

- Going forward, the FPGA market is set to expand. Major manufacturers of standard CPUs are expanding their product portfolio by acquiring companies specializing in FPGAs.
- In 2015, Intel bought Altera, a US-based manufacturer of programmable logic devices (PLDs), while last year AMD acquired Xilinx, the company that invented FPGA architecture.

Future of FPGA

- FPGAs will also be more widely used in networking. Apart from programmable logic cells, they will contain highly specialized silicon elements, i.e. network interface controllers. You can also expect network-specific circuits to be developed.
- From the developer's point of view, FPGA circuits will contain more logic gates, allowing us to implement more complex functionalities.
- You will be able to put more network functionalities in a single circuit and piece of hardware equipment. Of course, this will make the entire implementation more complex too.

FGPA and AI

- Today, FPGAs are gaining prominence in another field: deep neural networks (DNNs) that are used for artificial intelligence (AI).
- Running DNN inference models takes significant processing power.
- Graphics processing units (GPUs) are often used to accelerate inference processing, but in some cases, high-performance FPGAs might actually outperform GPUs in analyzing large amounts of data for machine learning.

FGPA and AI

- Microsoft is already putting Intel FPGA versatility to use for accelerating AI. Microsoft's Project Brainwave provides customers with access to Intel Stratix FPGAs through Microsoft Azure cloud services.
- The cloud servers outfitted with these FPGAs have been configured specifically for running deep learning models.
- The Microsoft service lets developers harness the power of FPGA chips without purchasing and configuring specialized hardware and software.
- Instead, developers can work with common open-source tools, such as the Microsoft Cognitive Toolkit or TensorFlow AI development framework.

Thank you

This presentation is created using LibreOffice Impress 5.1.6.2, can be used freely as per GNU General Public License

@mitu_skillologies

/mITuSkillologies

@mitu_group

/company/mitu-skillologies

MITUSkillologies

Web Resources
http://mitu.co.in
http://tusharkute.com

contact@mitu.co.in

tushar@tusharkute.com