# Lecture 21
# EE 421 / CS 425
# Digital System Design

**Fall 2024**

Shahid Masud

# Topics

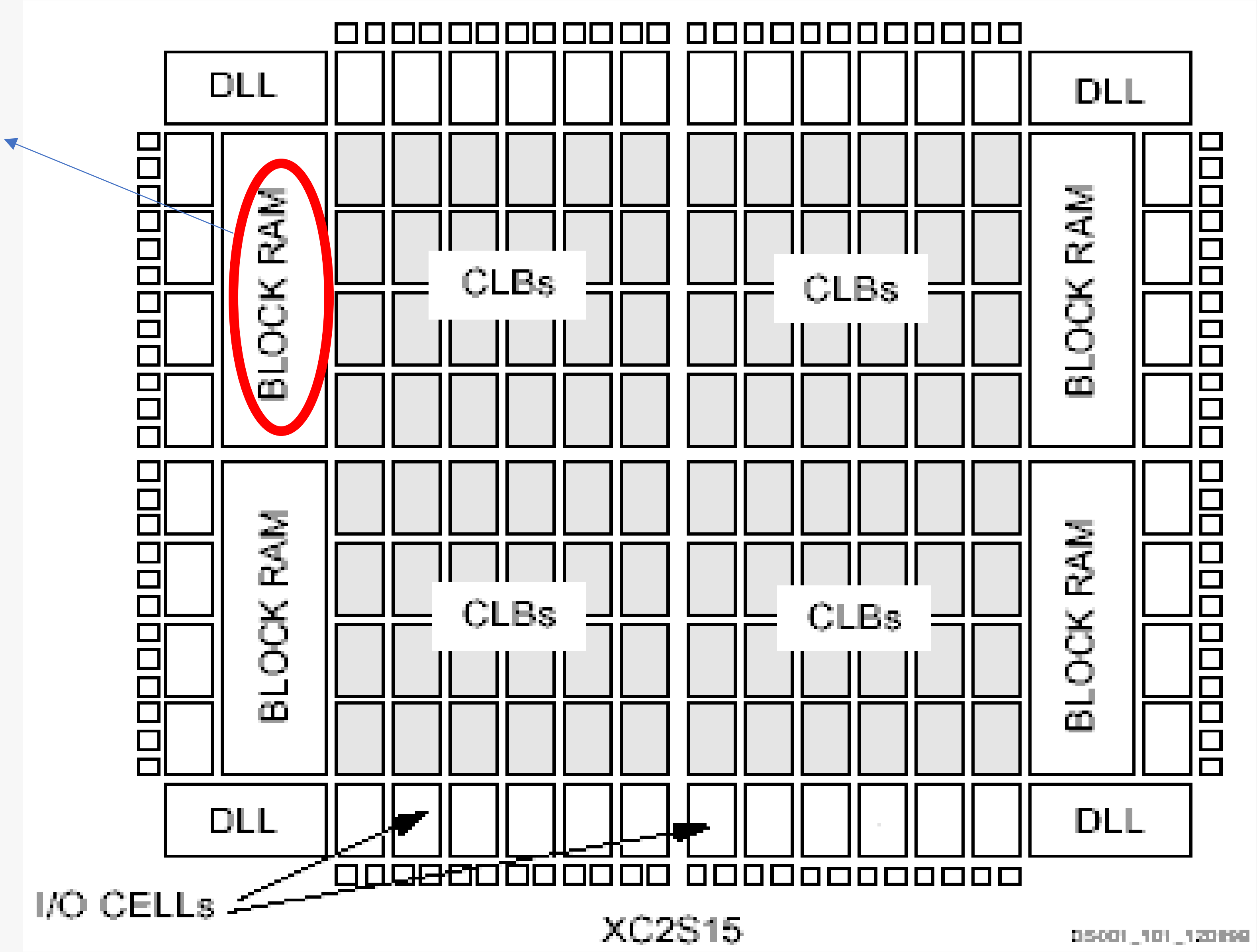
- Special Features in FPGA
- Sequential Implementation on CLB
- Memory
- Multipliers
- DSP Slices
- FIR and Symmetric Filters
- QUIZ 4 TODAY

# Specialized Modules in FPGAs

- Dedicated Memory
  - Single Port and Dual Port Embedded Memory Blocks – Block RAM
- Dedicated Arithmetic Units
  - Adders, Multipliers, Multipliers – Accumulators, Fast Carry Logic
- Digital Signal Processing Blocks – DSP Slice
  - FFT Butterfly Modules, FIR / IIR Filters,
  - IP Core Libraries for Encryption, Video Compression, Cloud Applications, etc.
- Embedded Processors
  - PowerPC, Microblaze, NIOS, ARM, MIPS, etc.
- Content Addressable Memory (CAM)
  - used in Branch Prediction, Caches inside CPU
- More and more features keep appearing in new FPGA devices
  - High Speed Interfaces, Security Features, RISC-V Support, etc.

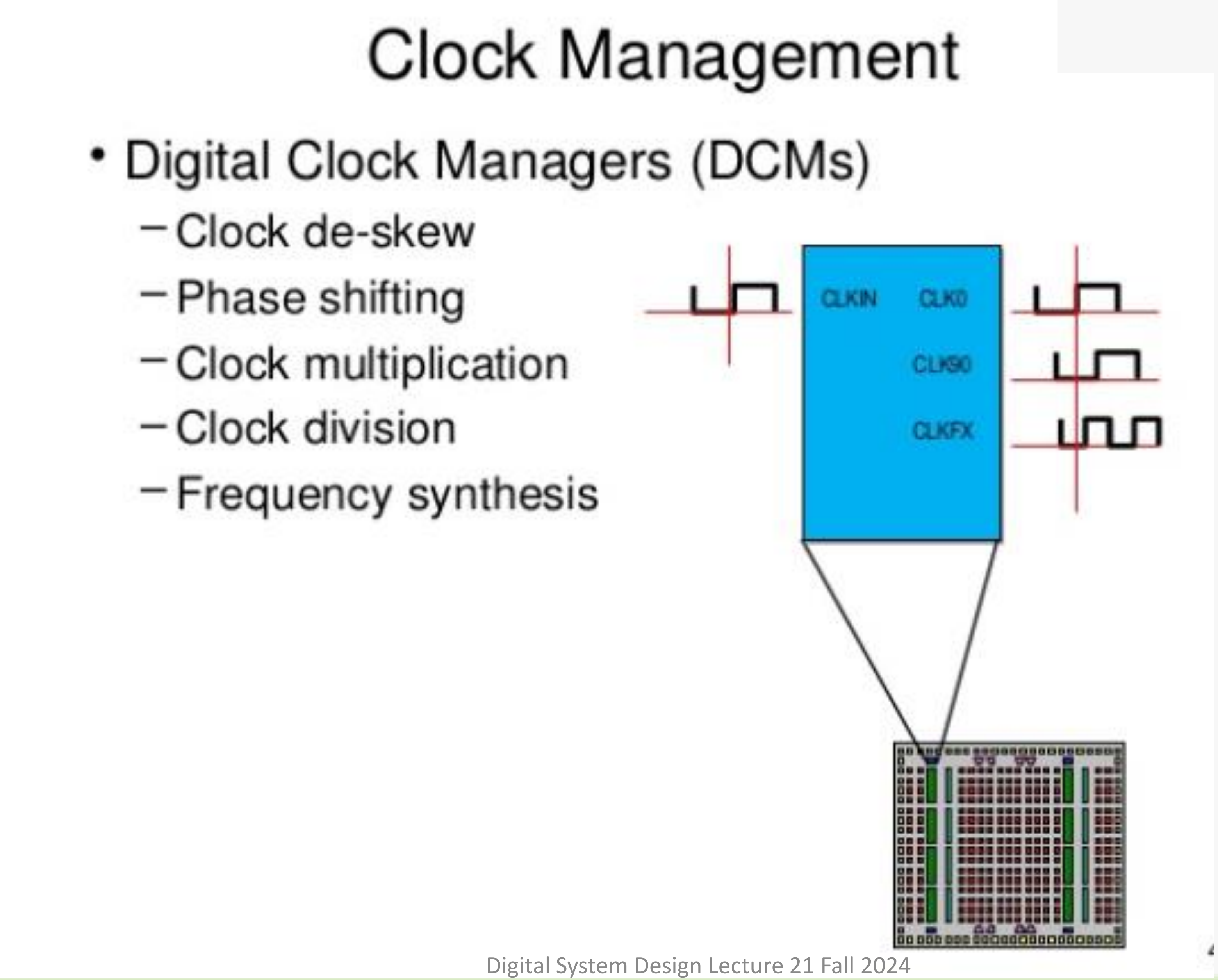
# Xilinx Spartan Architecture

Block RAM ?



DLL = Delay Locked Loop
For Clock Management

# Digital Clock Module in Xilinx FPGA



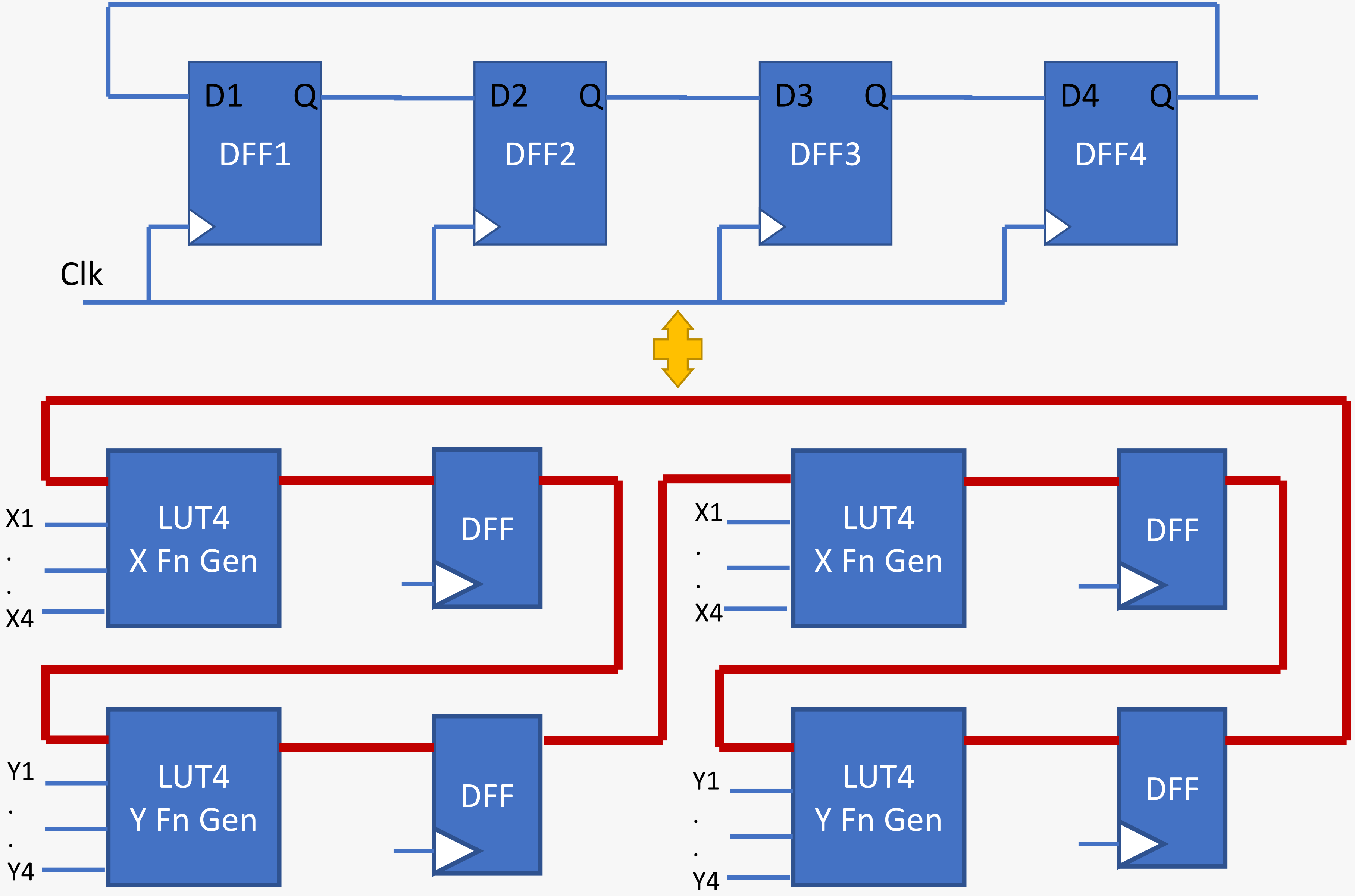
## Clock Management

- Digital Clock Managers (DCMs)
  - Clock de-skew
  - Phase shifting
  - Clock multiplication
  - Clock division
  - Frequency synthesis

# Implementation of memory in FPGA

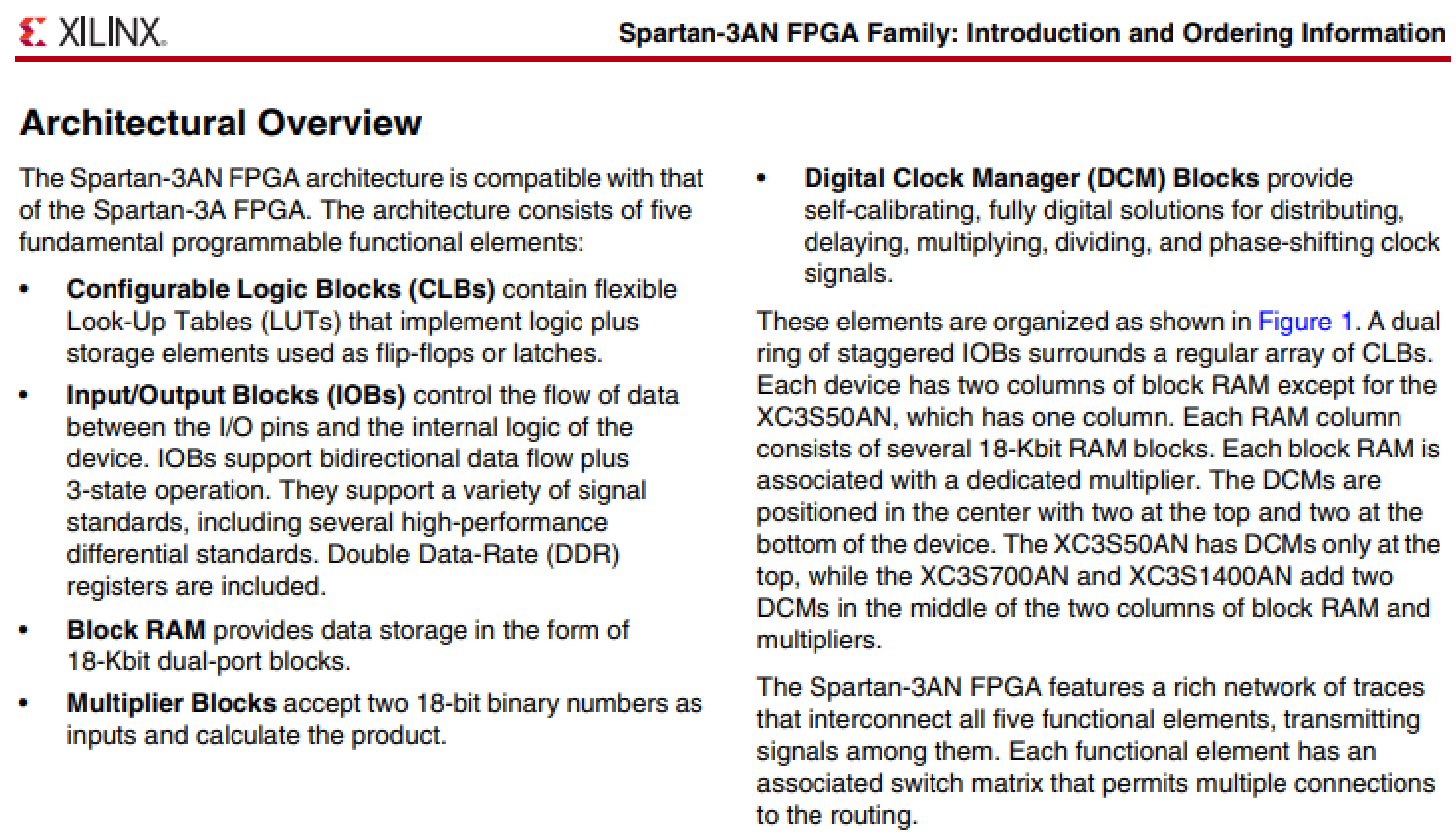
- Using LUT in CLBs – Distributed RAM
- Instantiating Block RAMs
- Provision of Dual port memory in modern FPGA

# Sequential Circuits in FPGA



D1=Q4
D2=Q1
D3=Q2
D4=Q3

# Multiplier Blocks – Xilinx Spartan-3AN



XILINX **Spartan-3AN FPGA Family: Introduction and Ordering Information**

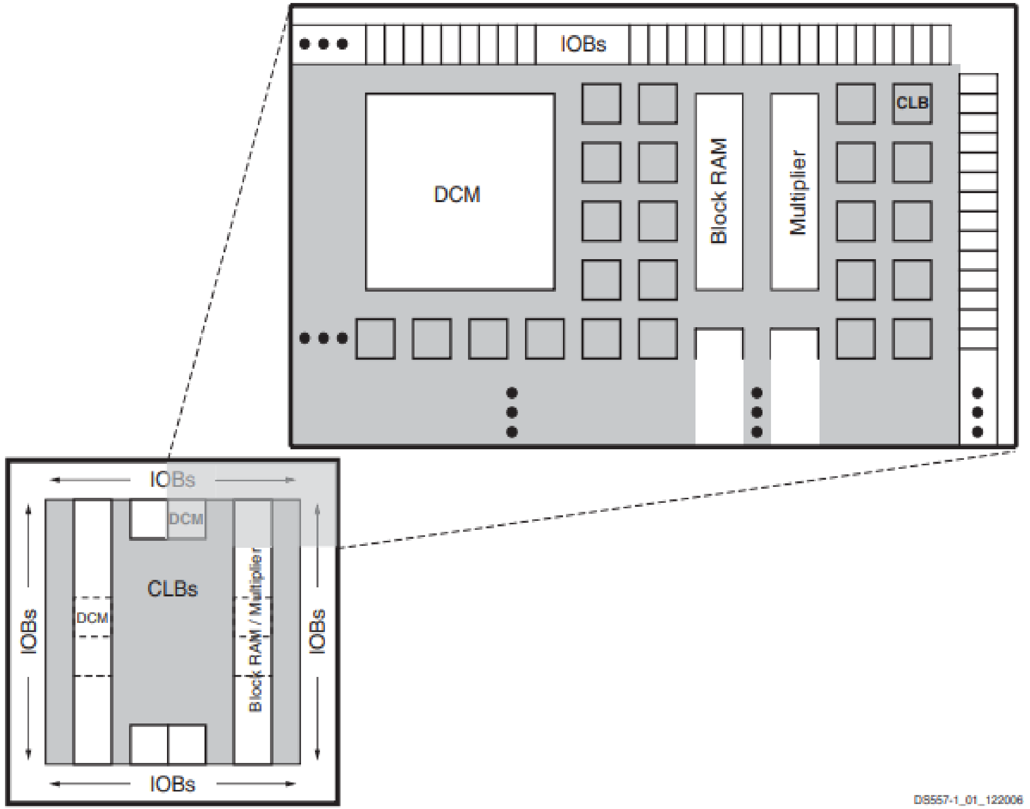
## Architectural Overview

The Spartan-3AN FPGA architecture is compatible with that of the Spartan-3A FPGA. The architecture consists of five fundamental programmable functional elements:

- **Configurable Logic Blocks (CLBs)** contain flexible Look-Up Tables (LUTs) that implement logic plus storage elements used as flip-flops or latches.
- **Input/Output Blocks (IOBs)** control the flow of data between the I/O pins and the internal logic of the device. IOBs support bidirectional data flow plus 3-state operation. They support a variety of signal standards, including several high-performance differential standards. Double Data-Rate (DDR) registers are included.
- **Block RAM** provides data storage in the form of 18-Kbit dual-port blocks.
- **Multiplier Blocks** accept two 18-bit binary numbers as inputs and calculate the product.
- **Digital Clock Manager (DCM) Blocks** provide self-calibrating, fully digital solutions for distributing, delaying, multiplying, dividing, and phase-shifting clock signals.

These elements are organized as shown in Figure 1. A dual ring of staggered IOBs surrounds a regular array of CLBs. Each device has two columns of block RAM except for the XC3S50AN, which has one column. Each RAM column consists of several 18-Kbit RAM blocks. Each block RAM is associated with a dedicated multiplier. The DCMs are positioned in the center with two at the top and two at the bottom of the device. The XC3S50AN has DCMs only at the top, while the XC3S700AN and XC3S1400AN add two DCMs in the middle of the two columns of block RAM and multipliers.

The Spartan-3AN FPGA features a rich network of traces that interconnect all five functional elements, transmitting signals among them. Each functional element has an associated switch matrix that permits multiple connections to the routing.



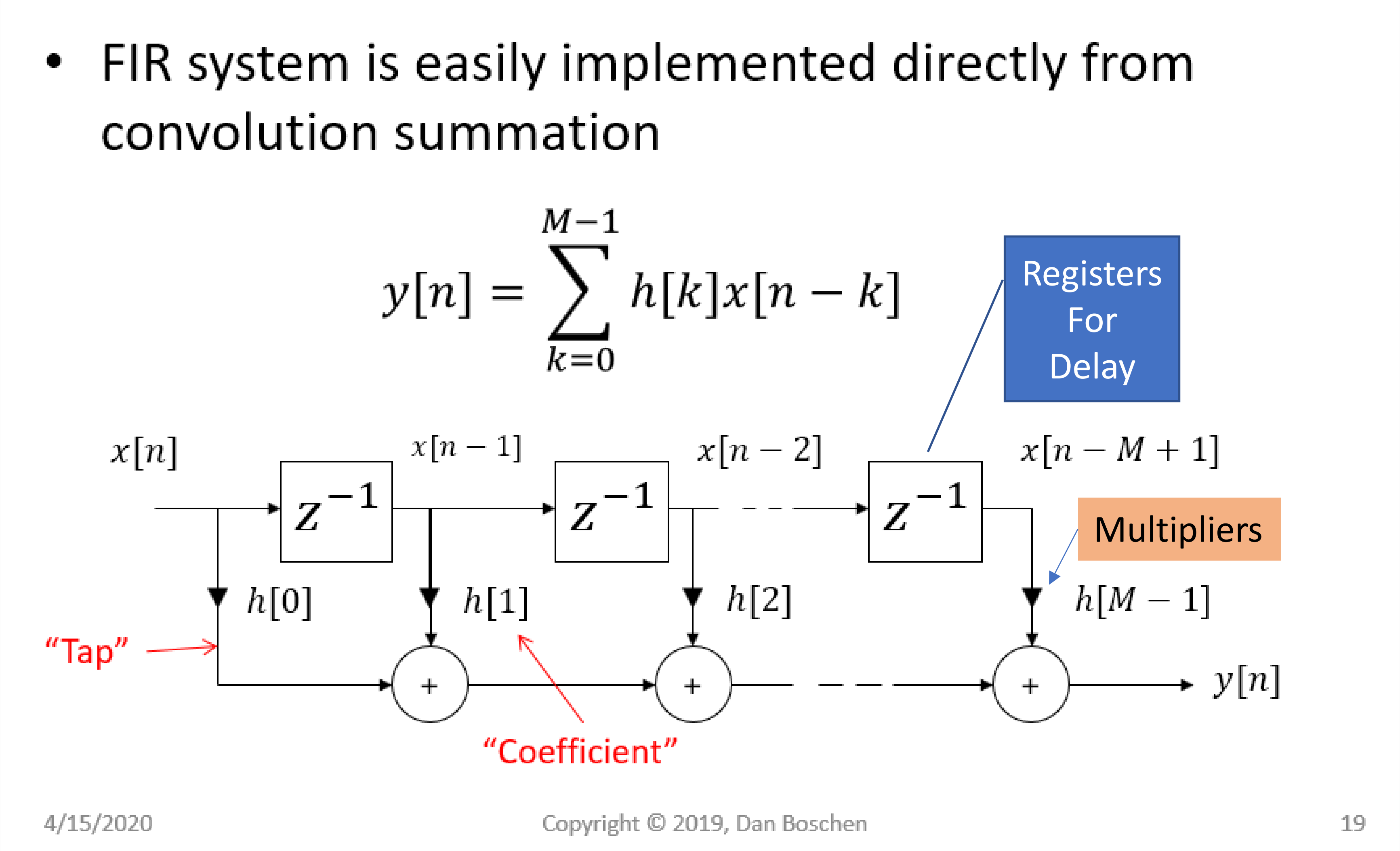
**Notes:**

1. The XC3S700AN and XC3S1400AN have two additional DCMs on both the left and right sides as indicated by the dashed lines. The XC3S50AN has only two DCMs at the top and only one Block RAM/Multiplier column.

*Figure 1:* **Spartan-3AN Family Architecture**

# DSP Features in modern FPGA
# Example FIR Filter Implementation

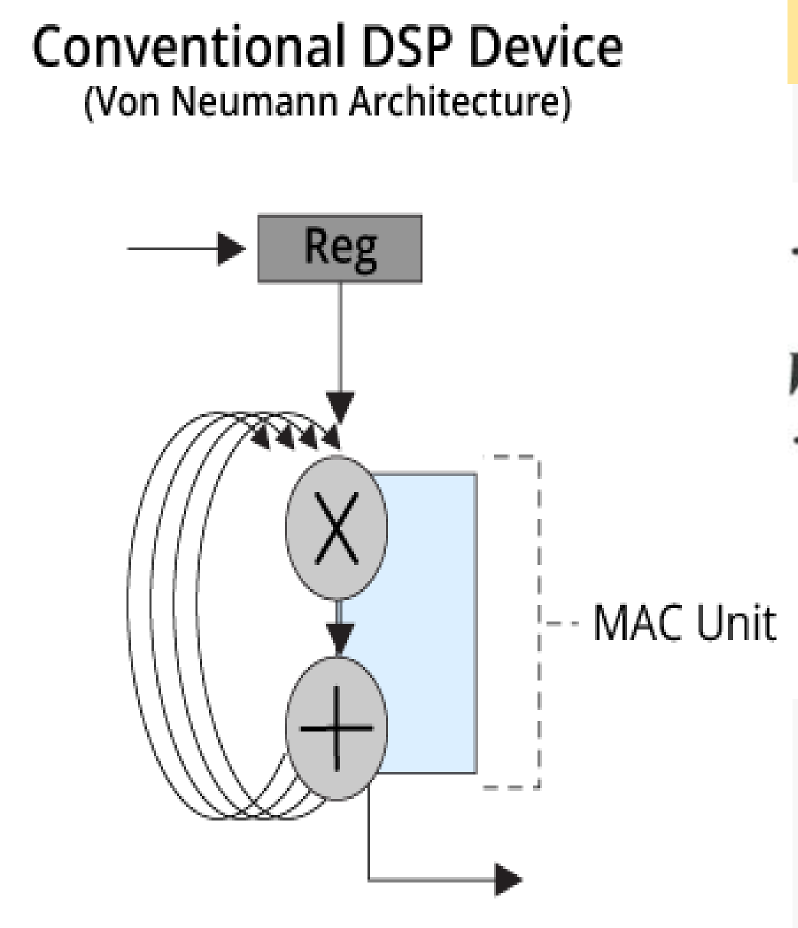
# FIR Filter Design



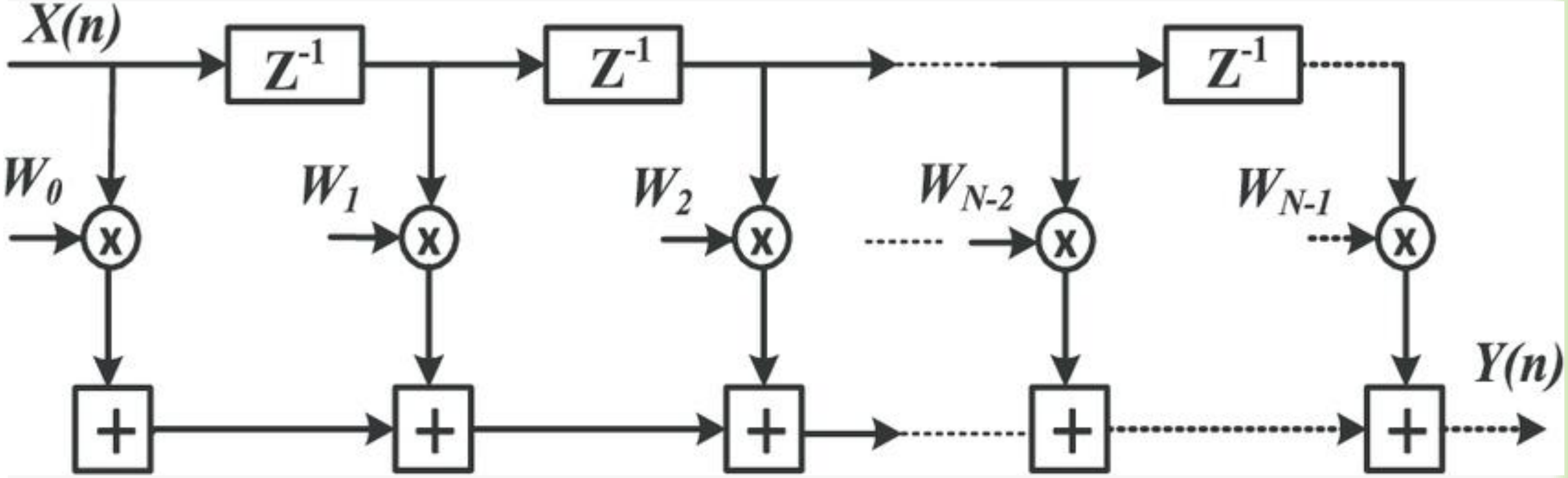
- FIR system is easily implemented directly from convolution summation

$$y[n] = \sum_{k=0}^{M-1} h[k]x[n-k]$$

# Implementation of DSP Filters



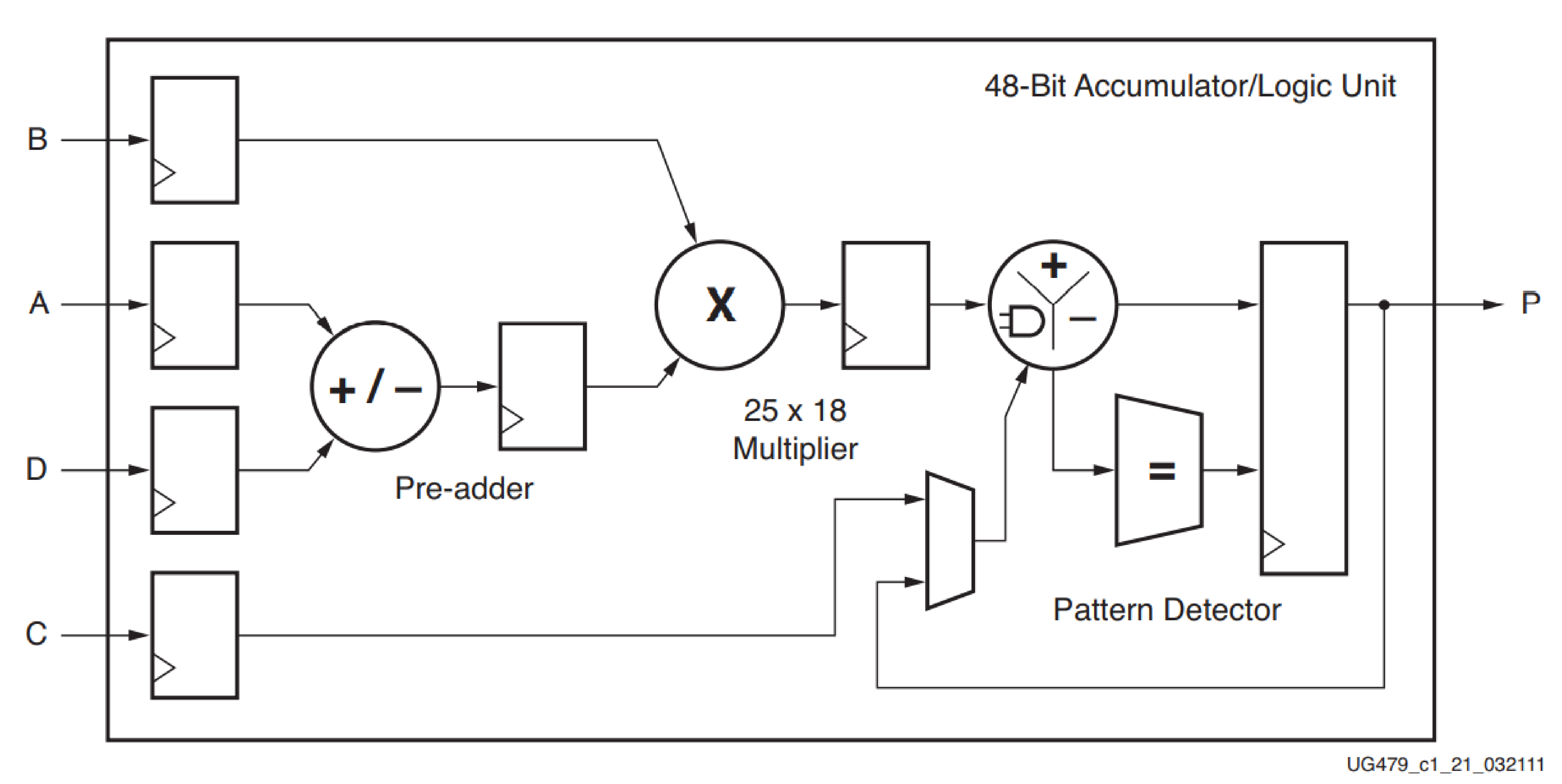
**Implementation of FIR filters in Digital Signal Processing**



**FIF filter mapping on**
**Software Programmable Device**

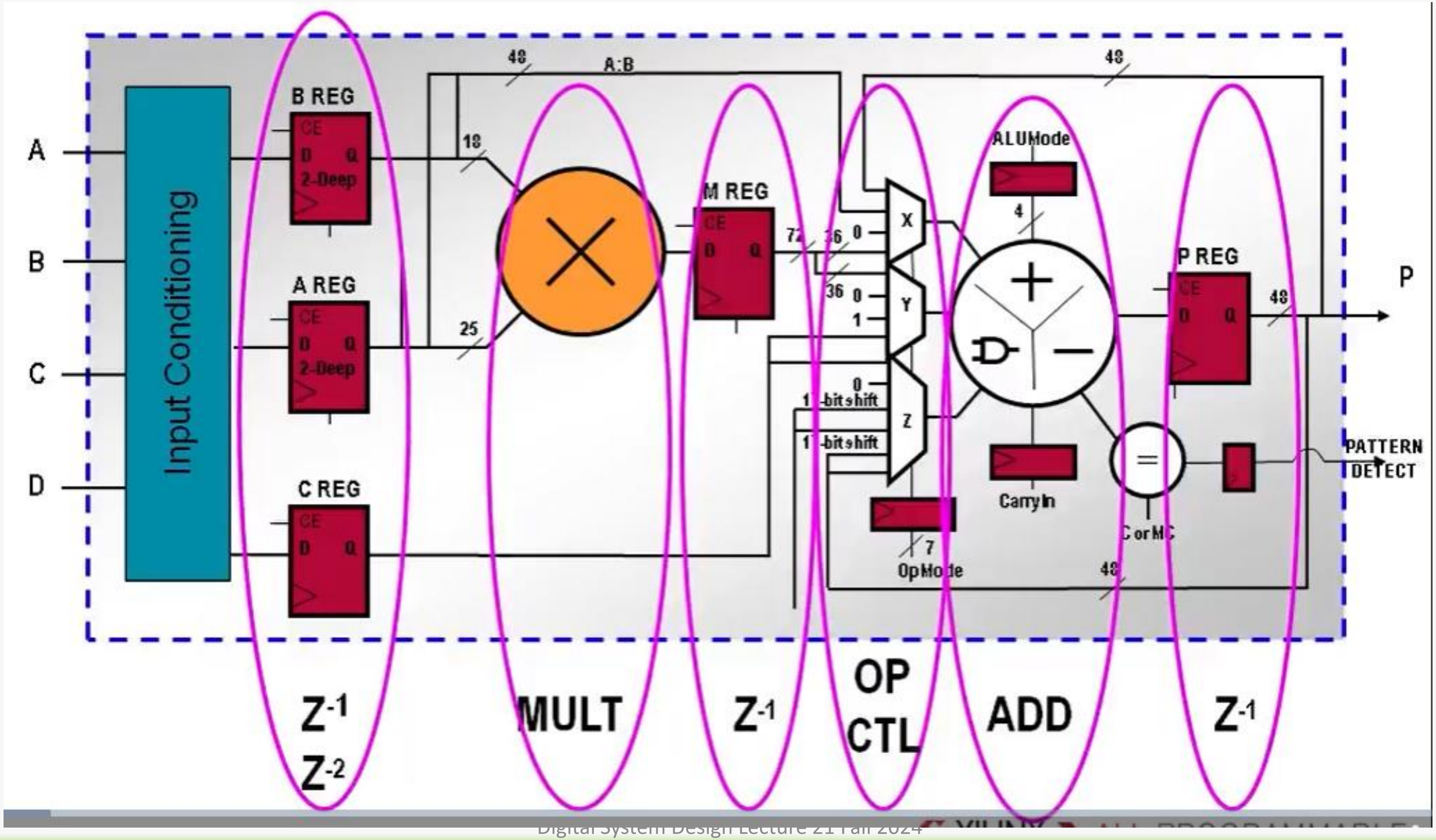
**FIF filter mapping on**
**a configurable Hardware Device**

# Basic Xilinx DSP48 Slice Architecture

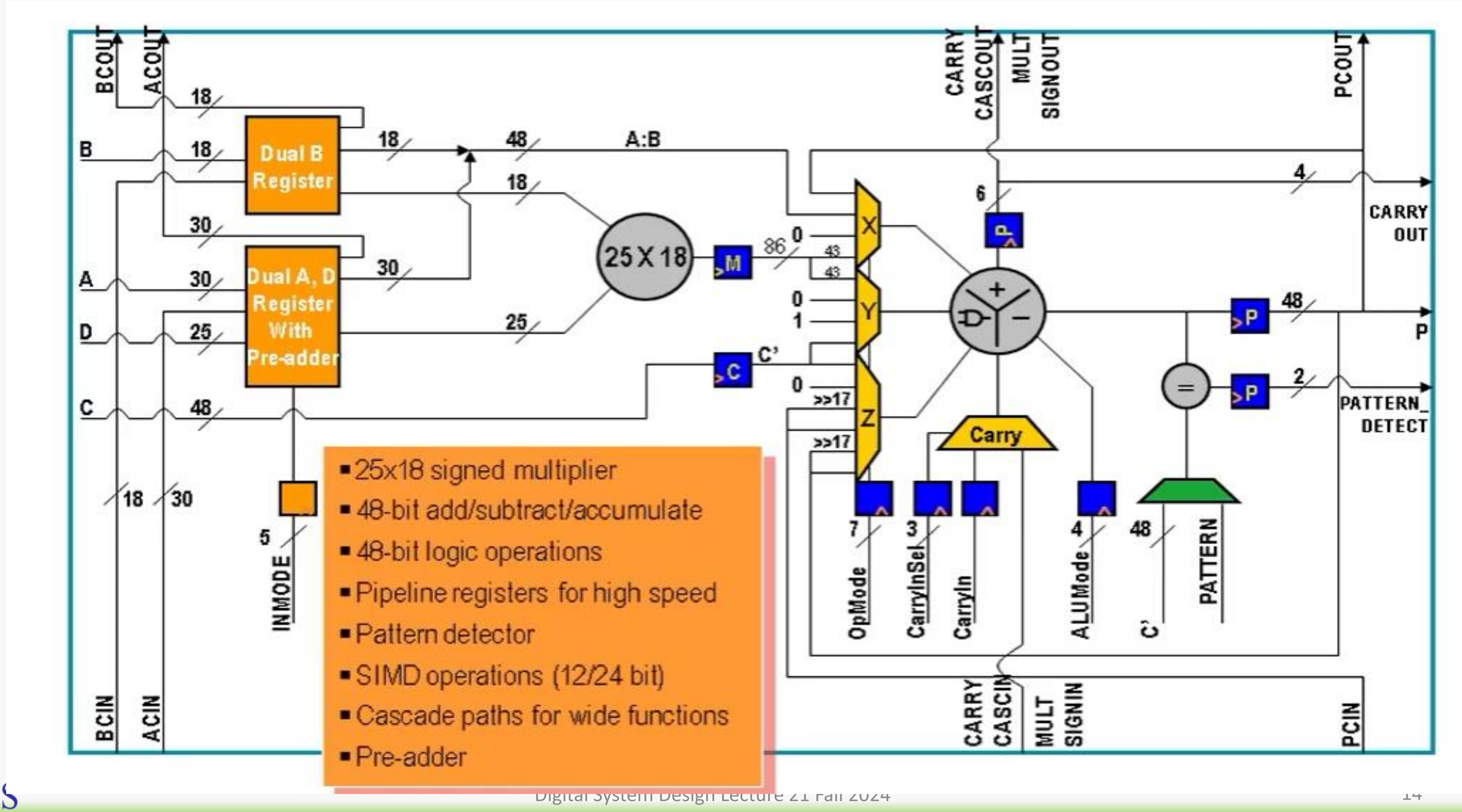


UG479_c1_21_032111

# DSP Slice Features



# Example of DSP Slice and Features



- 25x18 signed multiplier
- 48-bit add/subtract/accumulate
- 48-bit logic operations
- Pipeline registers for high speed
- Pattern detector
- SIMD operations (12/24 bit)
- Cascade paths for wide functions
- Pre-adder

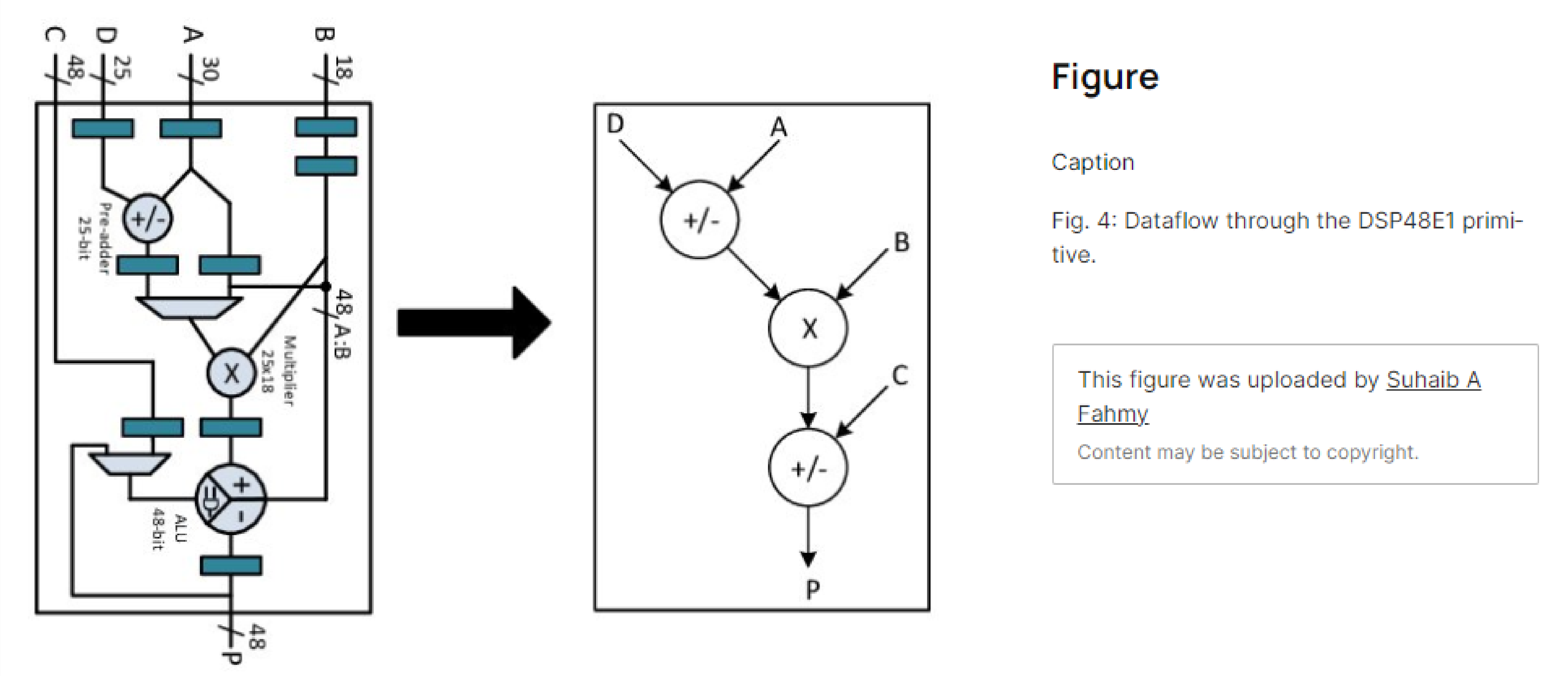
# Xilinx DSP48 Slice Functionality

- 25 x 18 two’s complement multiplier
- 48-bit Accumulator
- Power saving Pre-Adder for symmetric FIR filter implementation
- Single-Instruction-Multiple-Data (SIMD) arithmetic unit
- Dual 24-bit or Quad 12-bit Add/Sub/Acc
- Optional Logic Unit with 10 different operations on two operands
- Pattern Detector for convergent or symmetric rounding
- 96-bit wide Logic functions in conjunction with

Pattern Detector and Logic Unit

- Optional Pipelining and Dedicated Buses for Cascading

# Mapping Add and Mult on DSP Slice



## Figure

Caption

Fig. 4: Dataflow through the DSP48E1 primitive.

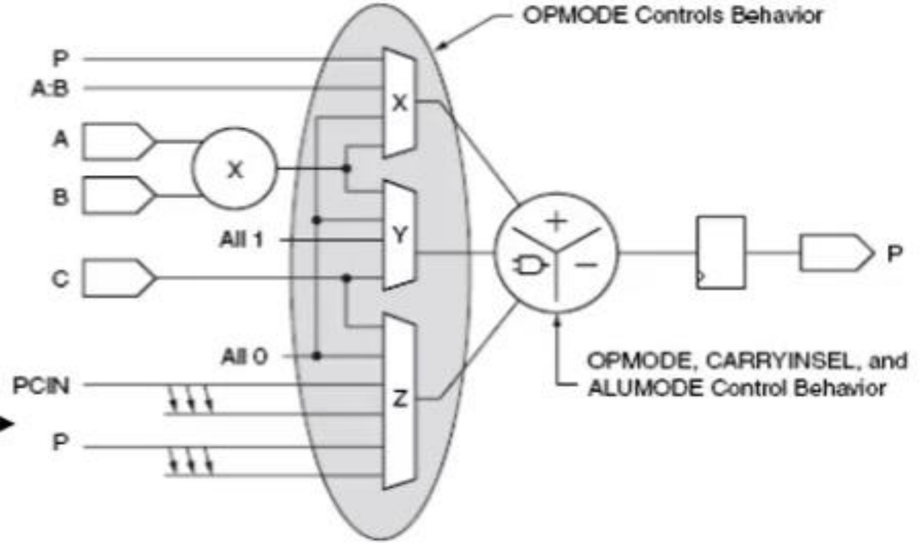
This figure was uploaded by Suhaib A Fahmy

Content may be subject to copyright.

# X, Y and Z Multiplexer

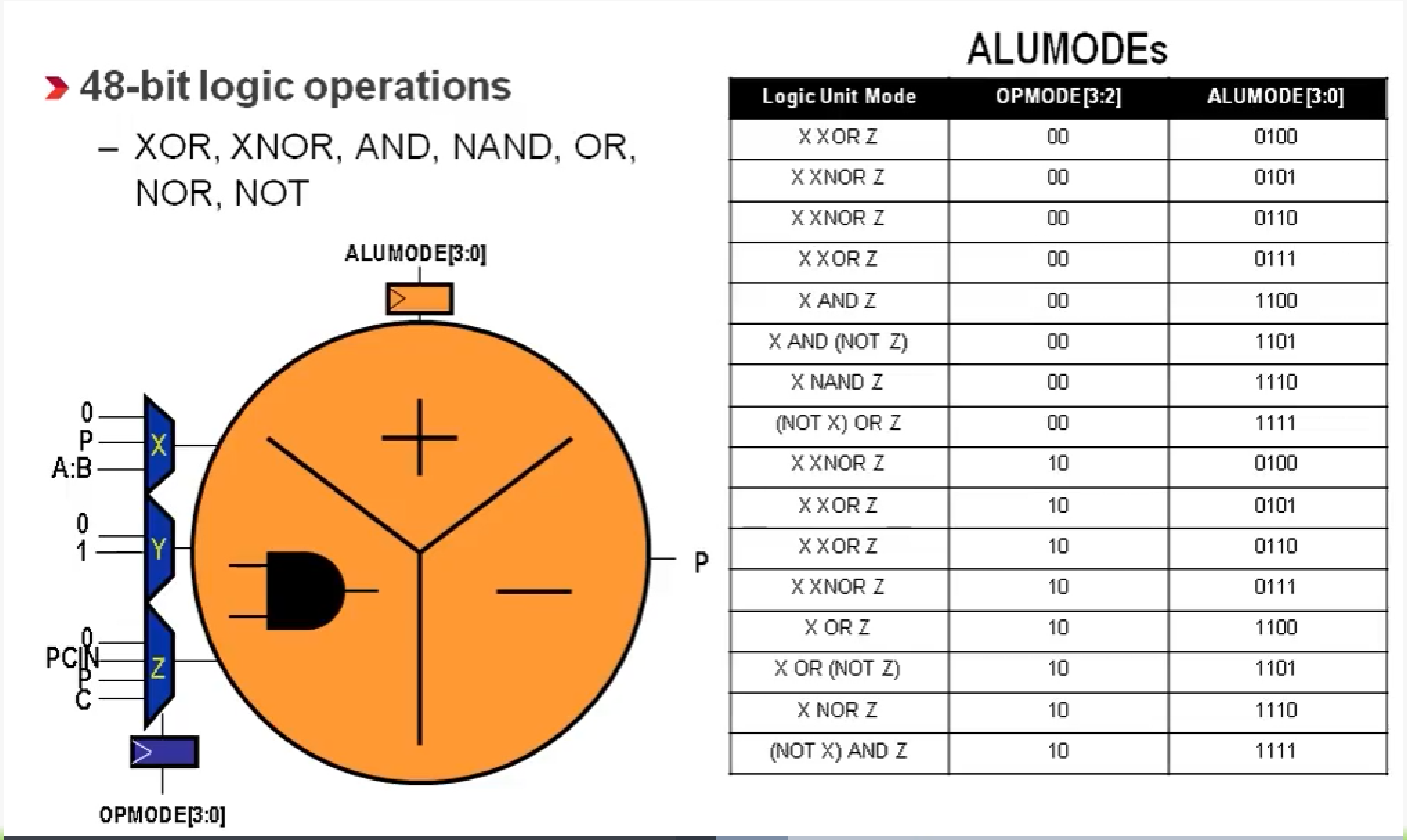
- Adder/subtractor operates on X, Y, Z and CIN operands
  - Table shows basic operations
- X, Y, and Z multiplexers allow for dynamic OPMODEs
- Multiplier output requires both X and Y multiplexers

| ALUMODE | Operation |
|---|---|
| 0000 | Z + X + Y + CIN |
| 0001 | -Z + (X + Y + CIN) – 1 |
| 0010 | -Z – X – Y – CIN – 1 |
| 0011 | Z – (X + Y + CIN) |
| Others | Logic Operations |



Normal or 17-bit right shifted with MSB fill for multi-precision arithmetic

# Two Input Logic Functions in DSP Slice



- 48-bit logic operations
  - XOR, XNOR, AND, NAND, OR, NOR, NOT

ALUMODEs

| Logic Unit Mode | OPMODE[3:2] | ALUMODE[3:0] |
|---|---|---|
| X XOR Z | 00 | 0100 |
| X XNOR Z | 00 | 0101 |
| X XNOR Z | 00 | 0110 |
| X XOR Z | 00 | 0111 |
| X AND Z | 00 | 1100 |
| X AND (NOT Z) | 00 | 1101 |
| X NAND Z | 00 | 1110 |
| (NOT X) OR Z | 00 | 1111 |
| X XNOR Z | 10 | 0100 |
| X XOR Z | 10 | 0101 |
| X XOR Z | 10 | 0110 |
| X XNOR Z | 10 | 0111 |
| X OR Z | 10 | 1100 |
| X OR (NOT Z) | 10 | 1101 |
| X NOR Z | 10 | 1110 |
| (NOT X) AND Z | 10 | 1111 |

# An Implementation of FIR using RAM

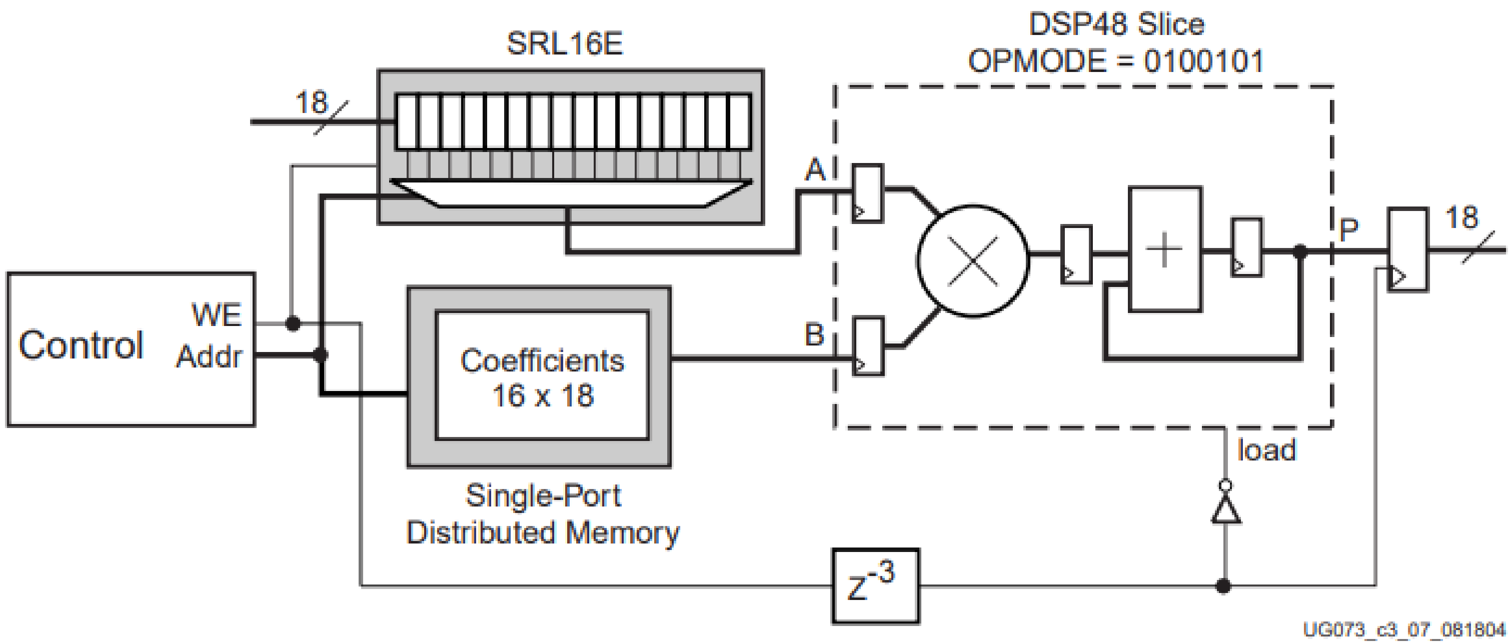


Figure 4-6: **Tap-Distributed RAM MAC FIR Filter**

# MAC Engine for FIR Filter in FPGA

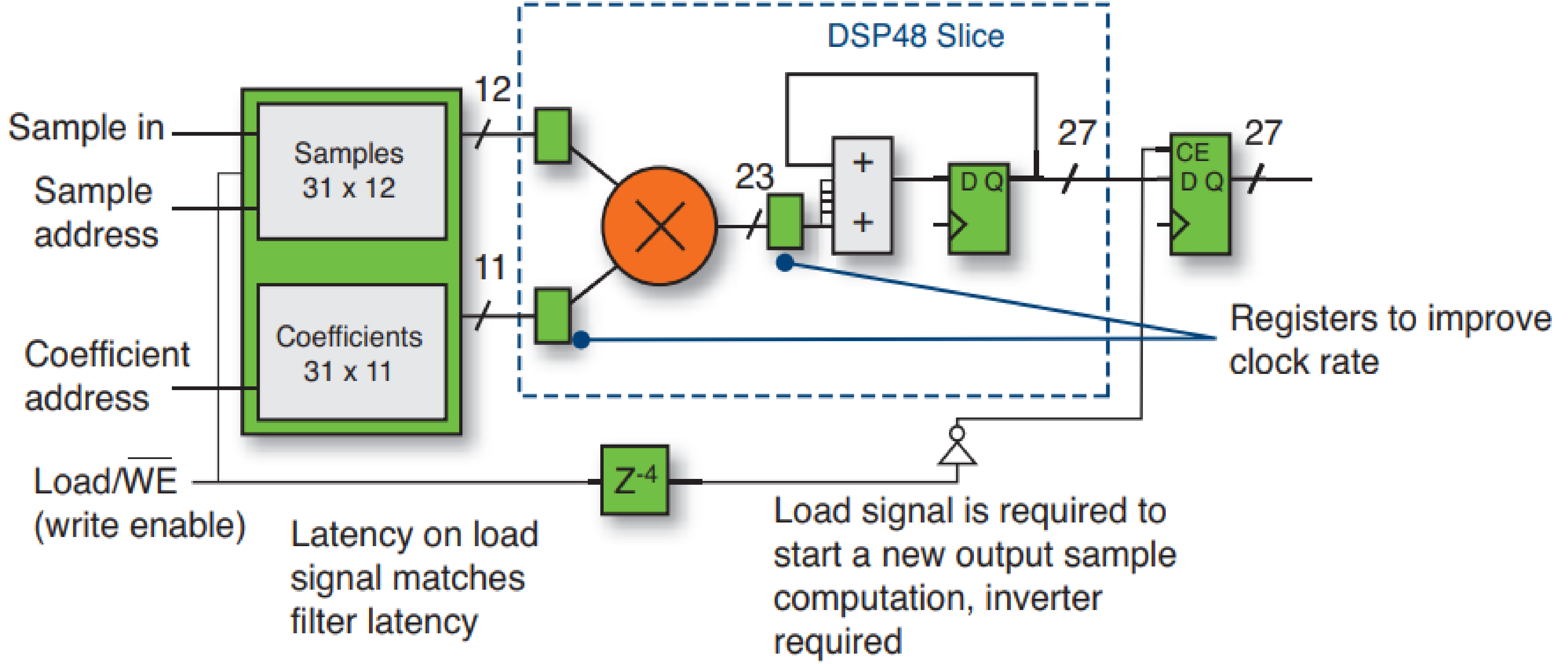
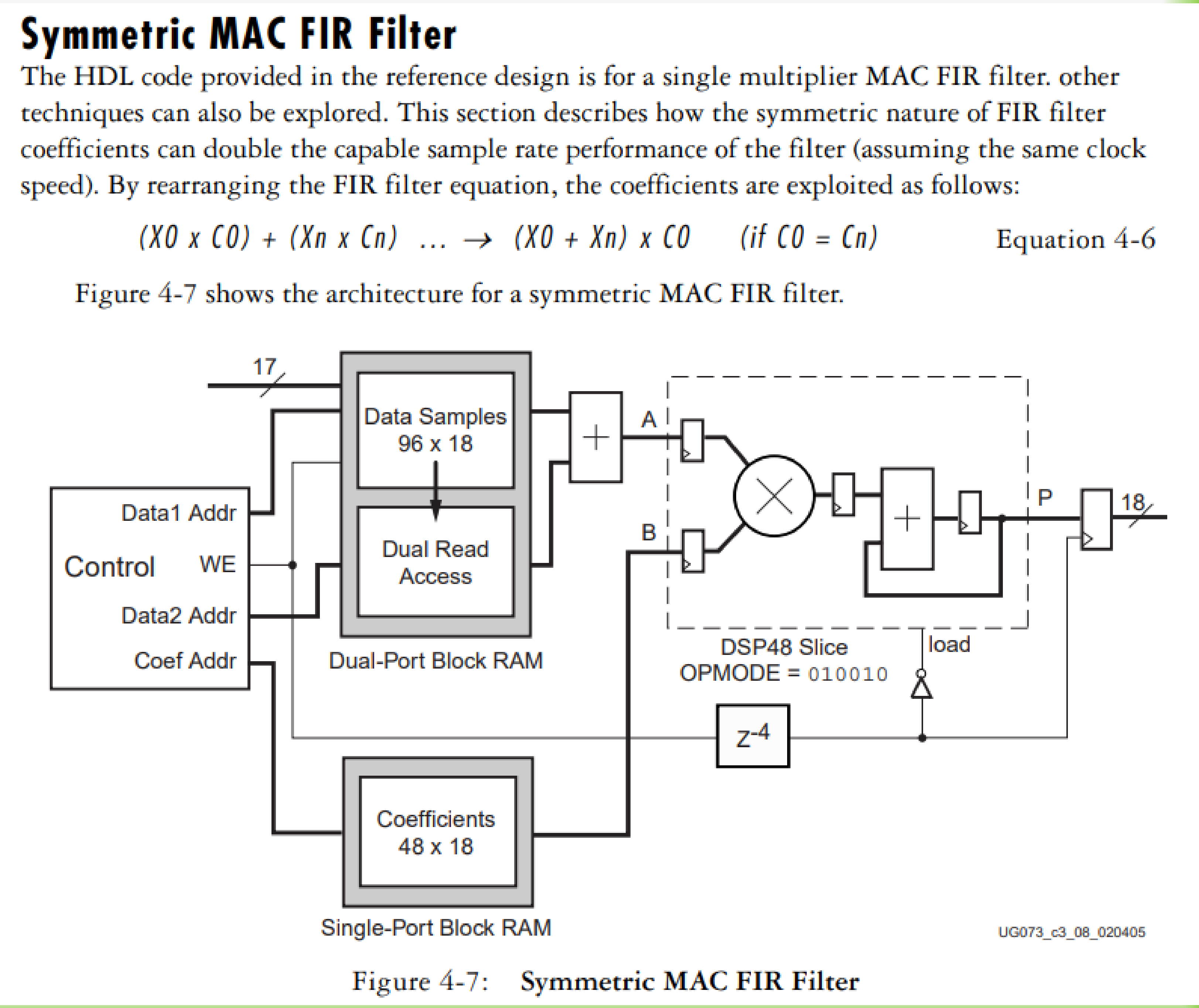

DSP48 Slice
Sample in
Sample address
Coefficient address
Load/$\overline{\text{WE}}$ (write enable)
Samples 31 x 12
Coefficients 31 x 11
12
11
23
27
27
D Q
CE D Q
+
+
$Z^{-4}$
Registers to improve clock rate
Latency on load signal matches filter latency
Load signal is required to start a new output sample computation, inverter required

Figure 4 – MAC engine FIR filter in an FPGA

# Symmetric FIR



## Symmetric MAC FIR Filter

The HDL code provided in the reference design is for a single multiplier MAC FIR filter. other techniques can also be explored. This section describes how the symmetric nature of FIR filter coefficients can double the capable sample rate performance of the filter (assuming the same clock speed). By rearranging the FIR filter equation, the coefficients are exploited as follows:

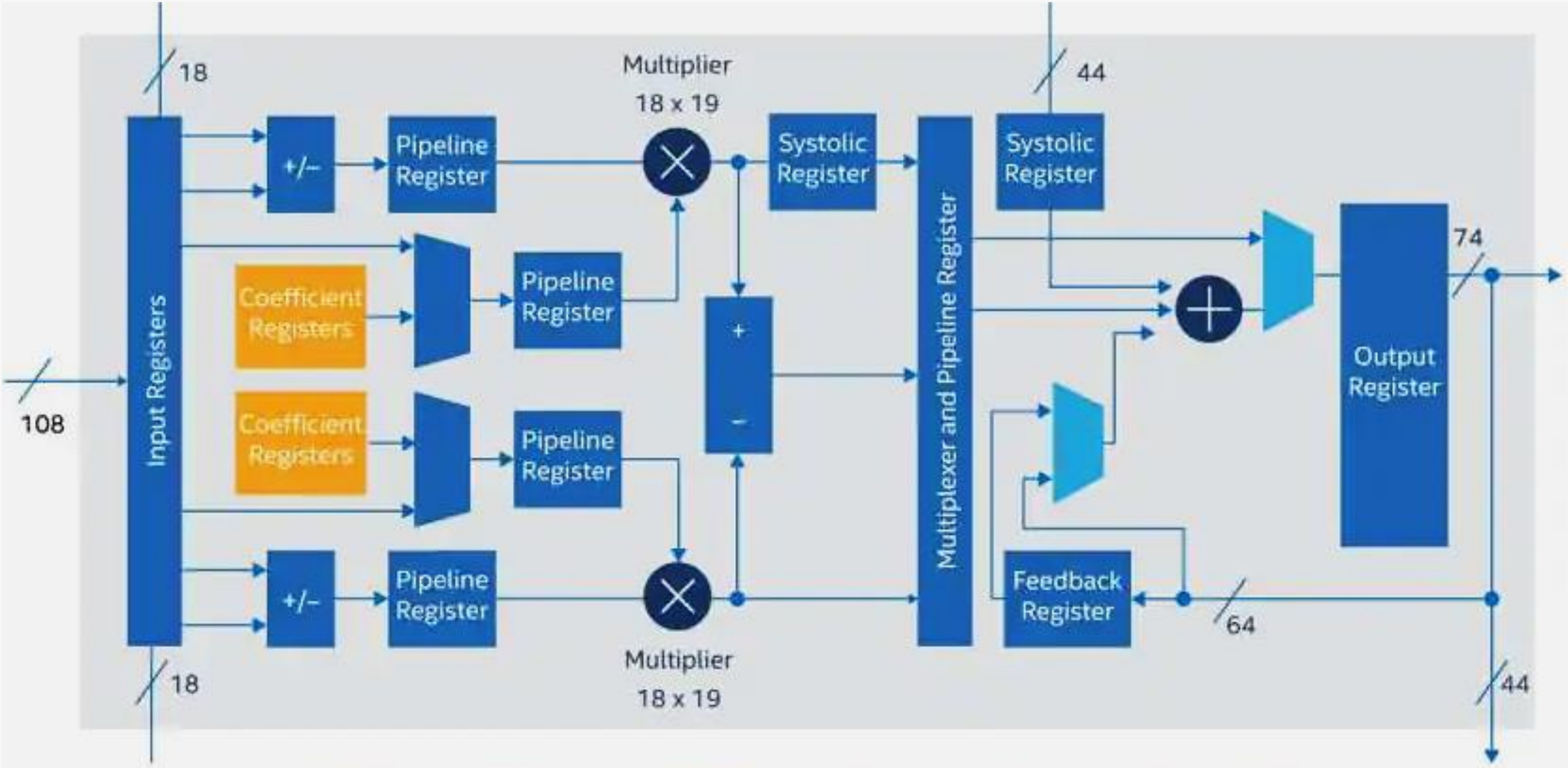
$$(X0 \times C0) + (Xn \times Cn) \ \ldots \rightarrow (X0 + Xn) \times C0 \quad (\text{if } C0 = Cn) \qquad \text{Equation 4-6}$$

Figure 4-7 shows the architecture for a symmetric MAC FIR filter.

Figure 4-7: **Symmetric MAC FIR Filter**

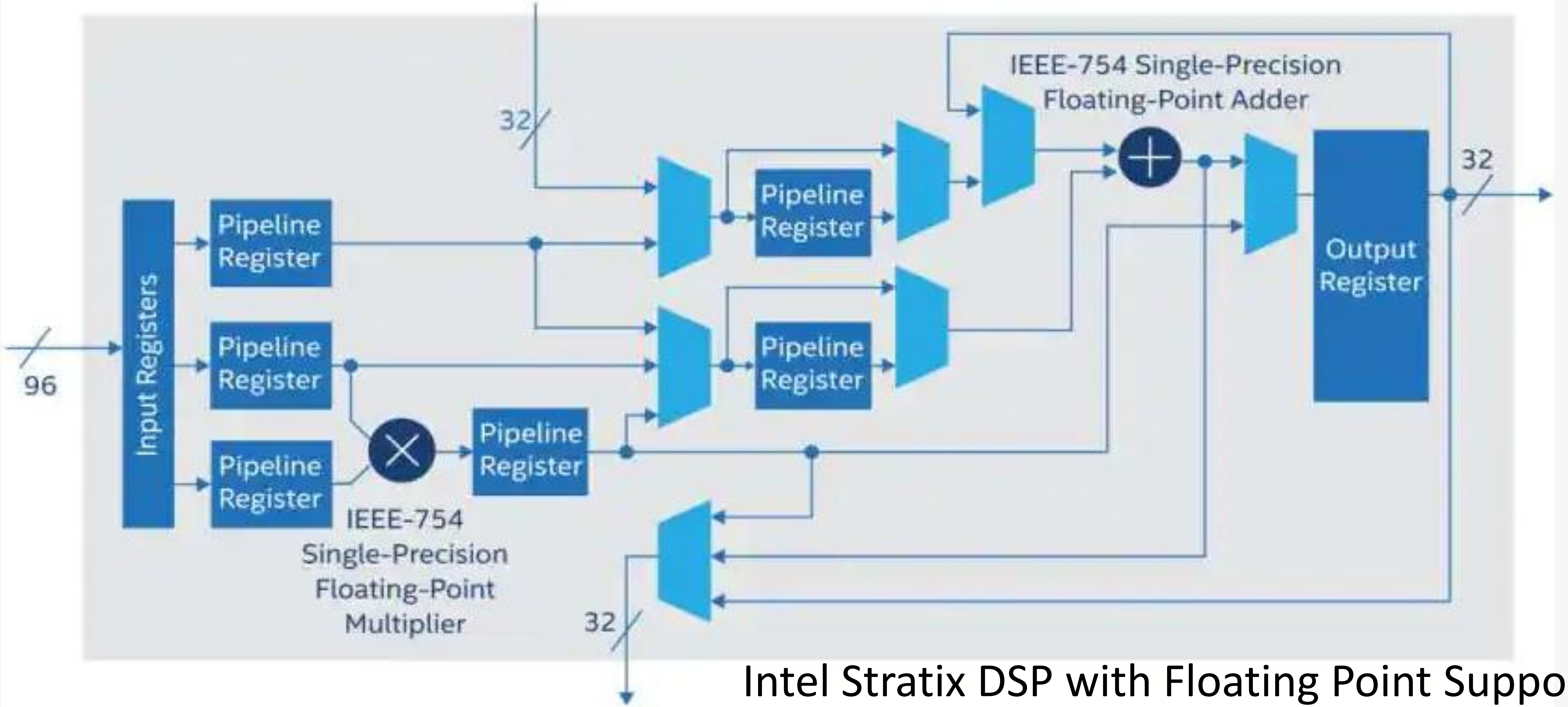
# Intel Stratix Slice

## Intel Stratix DSP Block – Fixed Point



Intel® Stratix® 10 Device DSP Block: Standard-Precision Fixed Point

# Intel Stratix DSP Slice with Floating Point



Intel Stratix DSP with Floating Point Support

Intel® Stratix® 10 Device DSP Block: Single-Precision Floating Point

# Further Reading

- https://www.xilinx.com/video/fpga/7-series-dsp-resources.html