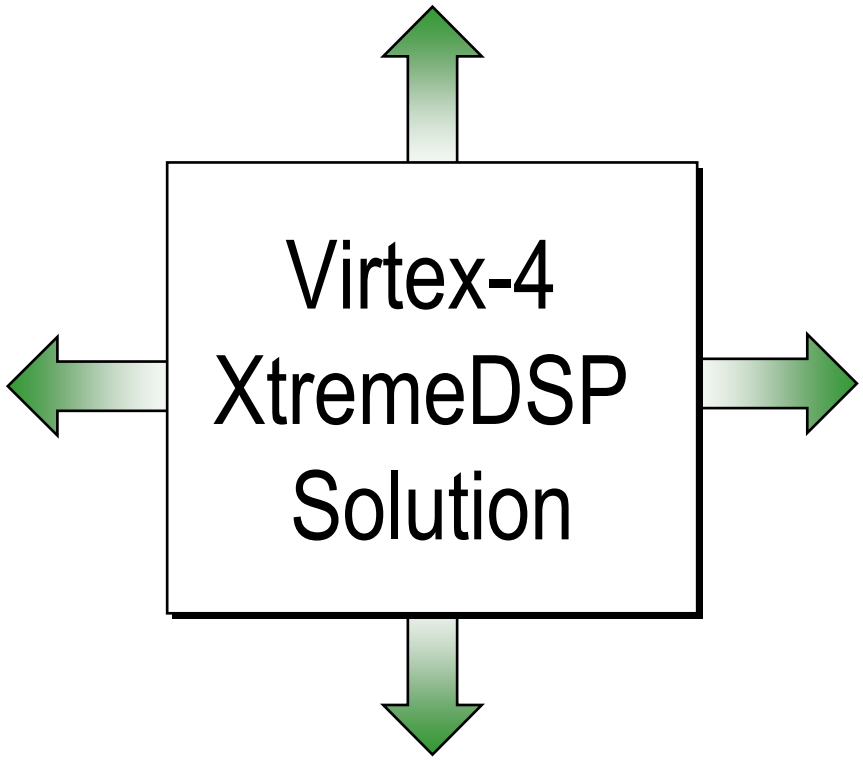
DSP Design Techniques for Best Performance, Power and Cost

Niall Battson
DSP Divisional Marketing

Virtex™-4 DSP Advantages

LOWER POWER
(1/12 the power of V-II Pro)



HIGHER PERFORMANCE
(2x the performance of V-II Pro)

SLICE REDUCTION
(All designs in this presentation use at least 50% less slices than V-II Pro)

MAXIMUM FLEXIBILTY
(Opmode of the XtremeDSP Slice provides over)

Agenda

- Virtex-4 Family
- The Xtreme DSP Slice
- Filter Techniques
- Case Studies
  - Digital Up Converter
  - 2-D

The Virtex-4 Family

Virtex-4 Family

Virtex-4 is the first Xilinx family to introduce three separate platforms optimized for different application domains. This fundamental shift provided the greatest silicon efficiency and optimal cost.

Virtex-4 LX Platform
Optimized for high-performance Logic

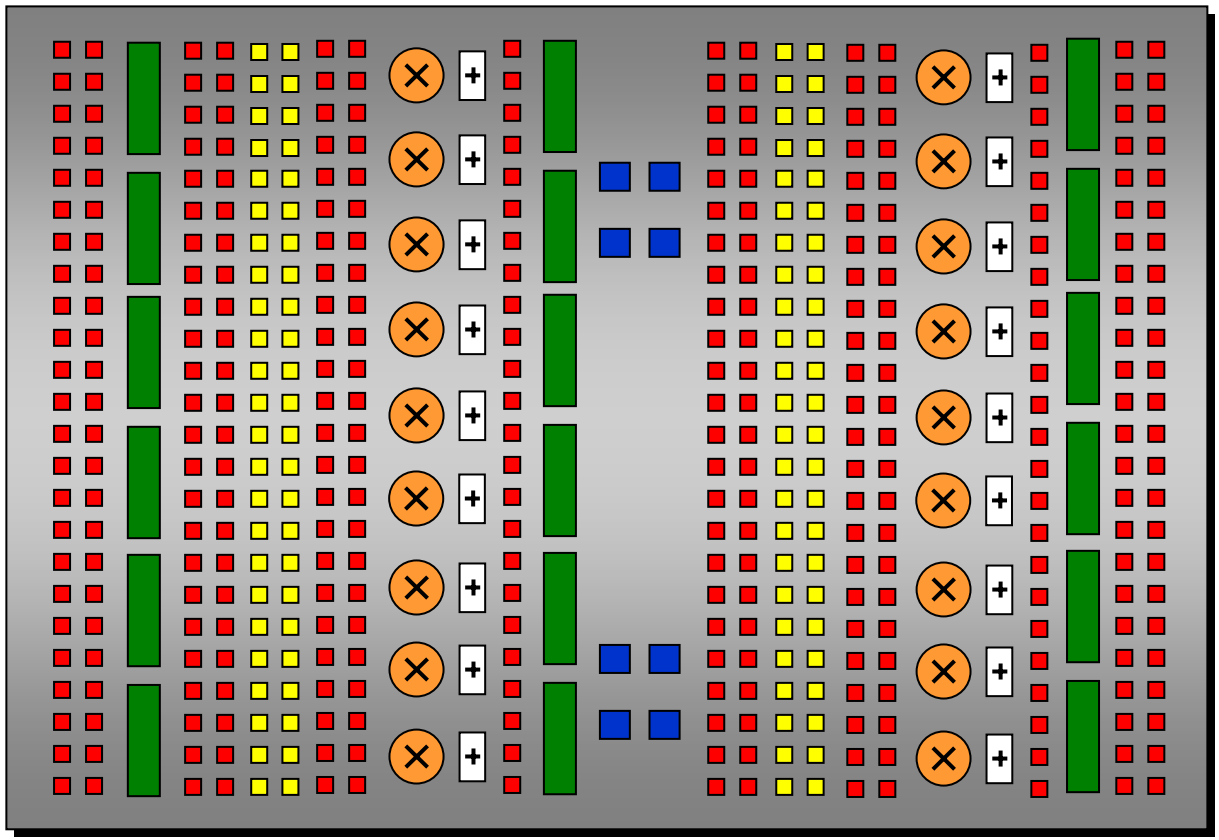
Virtex-4 FX Platform
Optimized for Embedded Processing and high-speed Serial Connectivity

Virtex-4 SX Platform
Optimized for high-performance Signal Processing

Virtex-4 SX

Note:

The SX family emphasizes Xilinx commitment to **DSP** applications by providing a strong skew toward to dedicated arithmetic units versus logic.

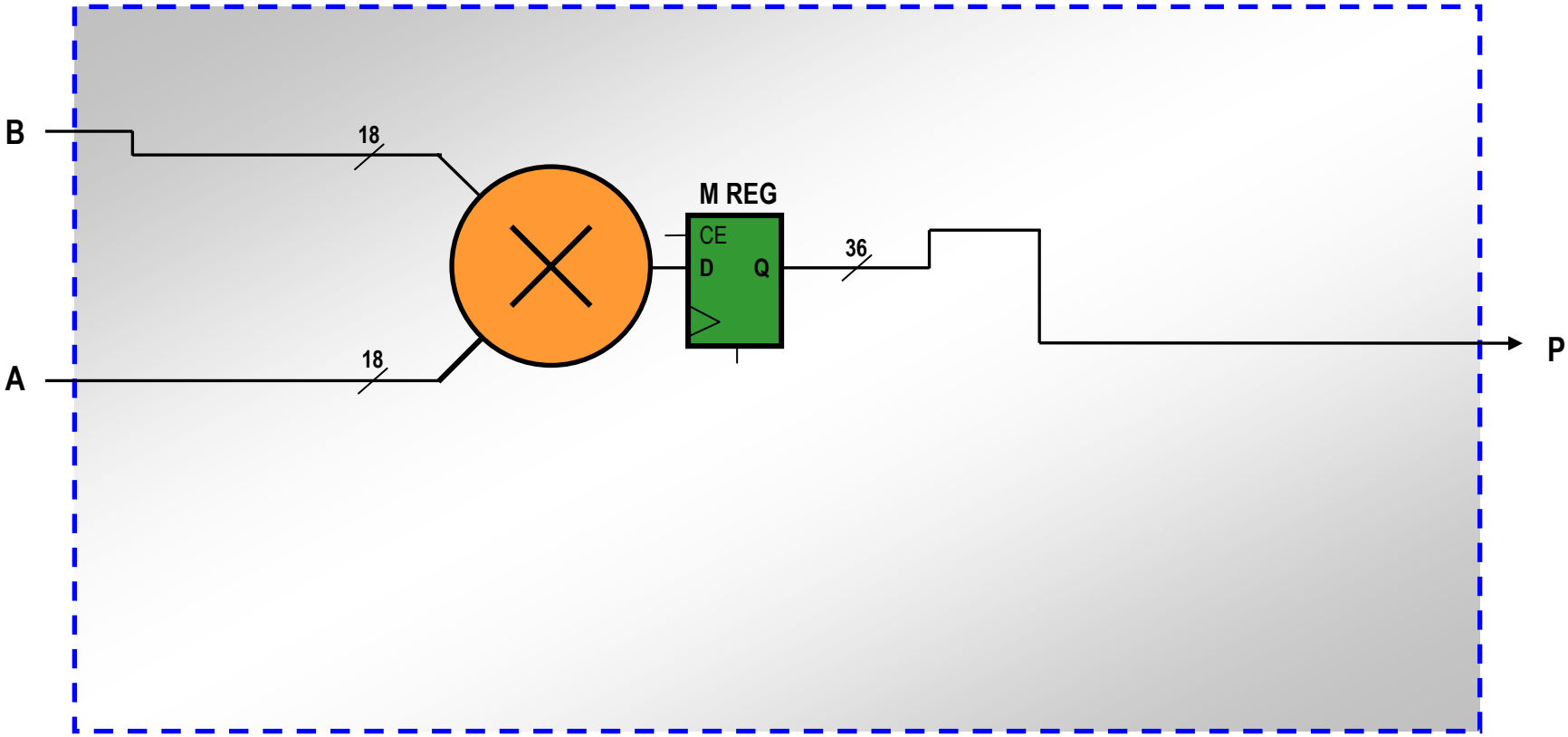


4VSX25	2,650 CLB	128 BRAM			128 XtremeDSP Slices
Largest device - 4VSX55	6,144 CLB	320 BRAM			512 XtremeDSP Slices

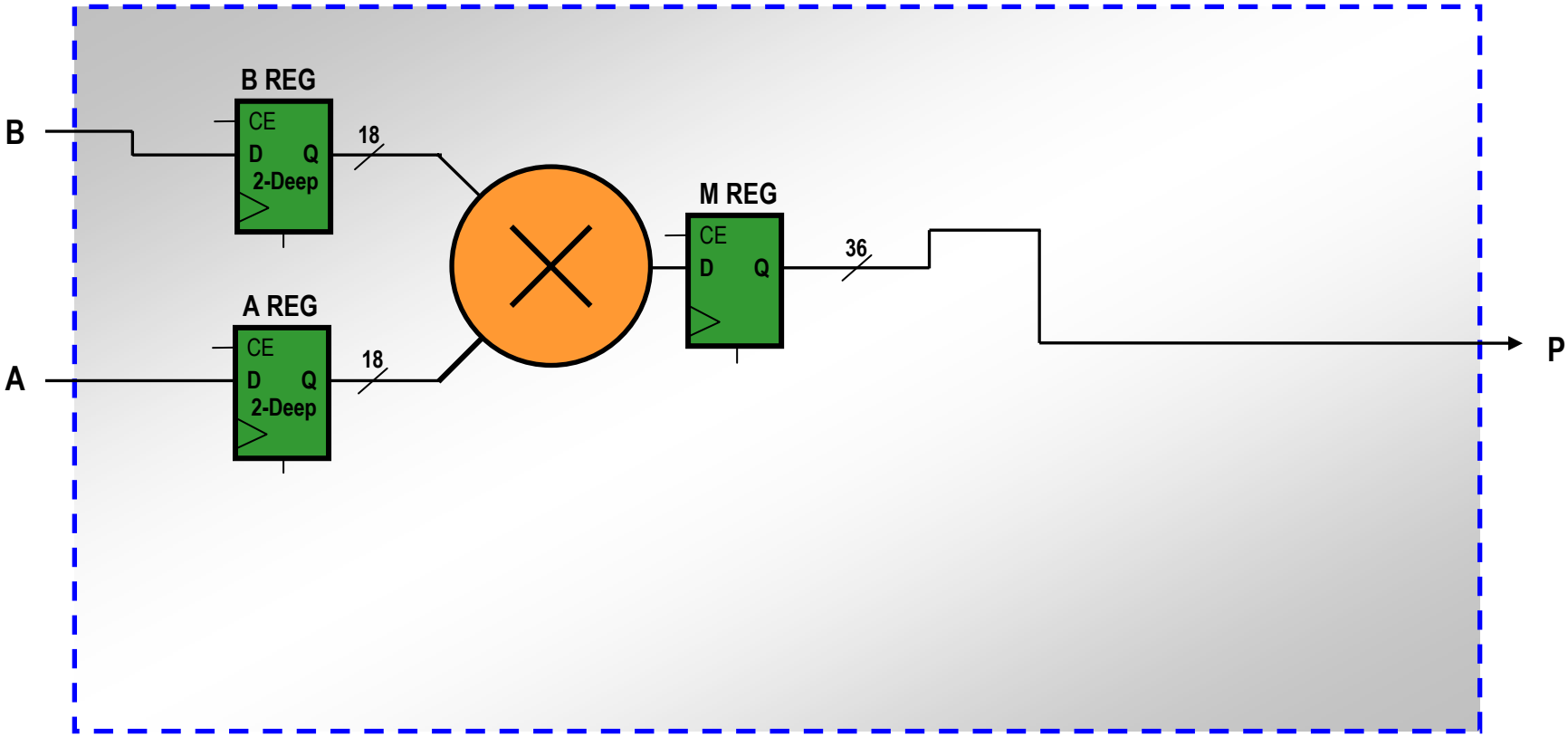
The XtremeDSP™ Slice

(also known as “DSP48”)

The XtremeDSP Slice

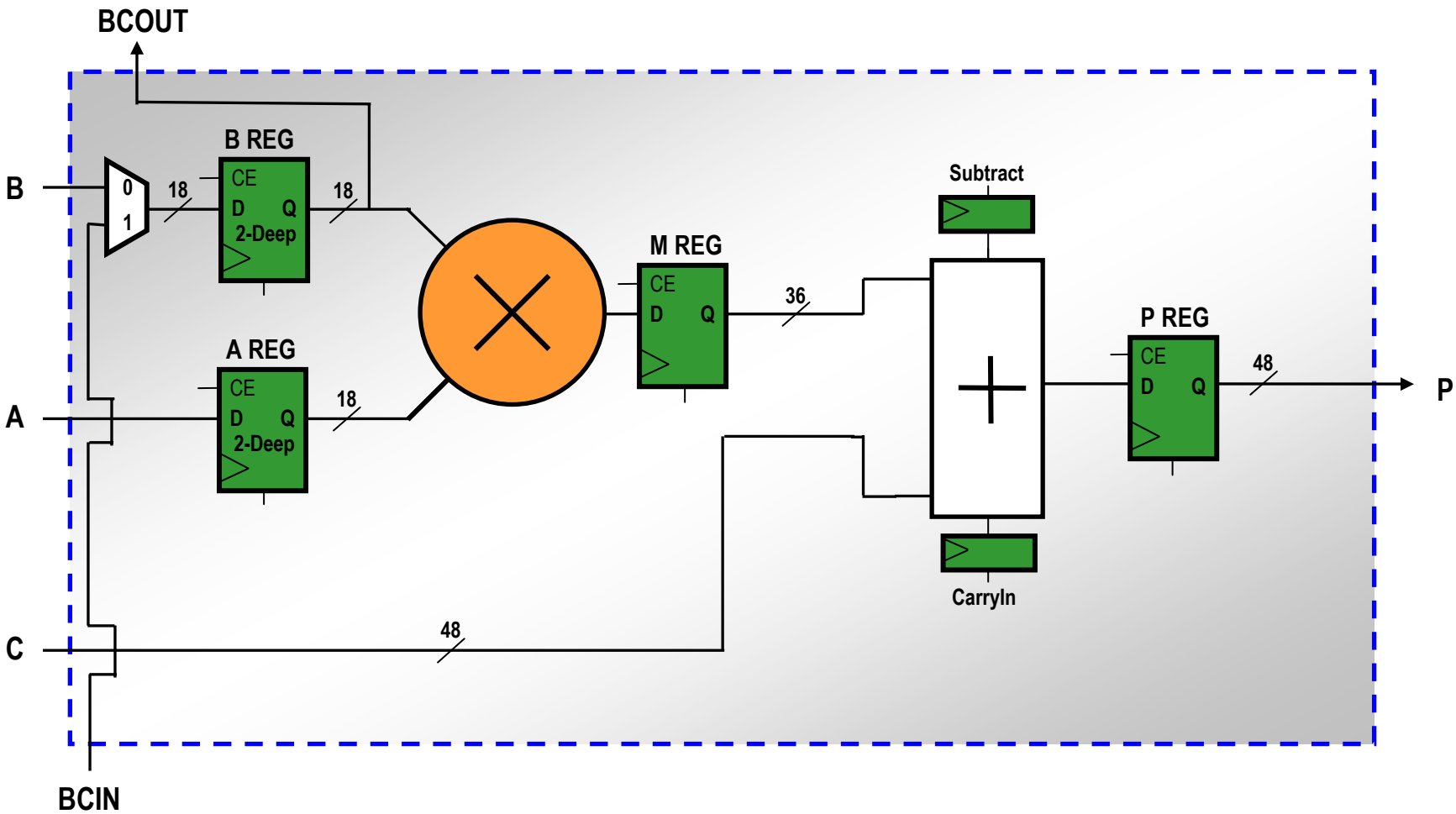


The XtremeDSP Slice



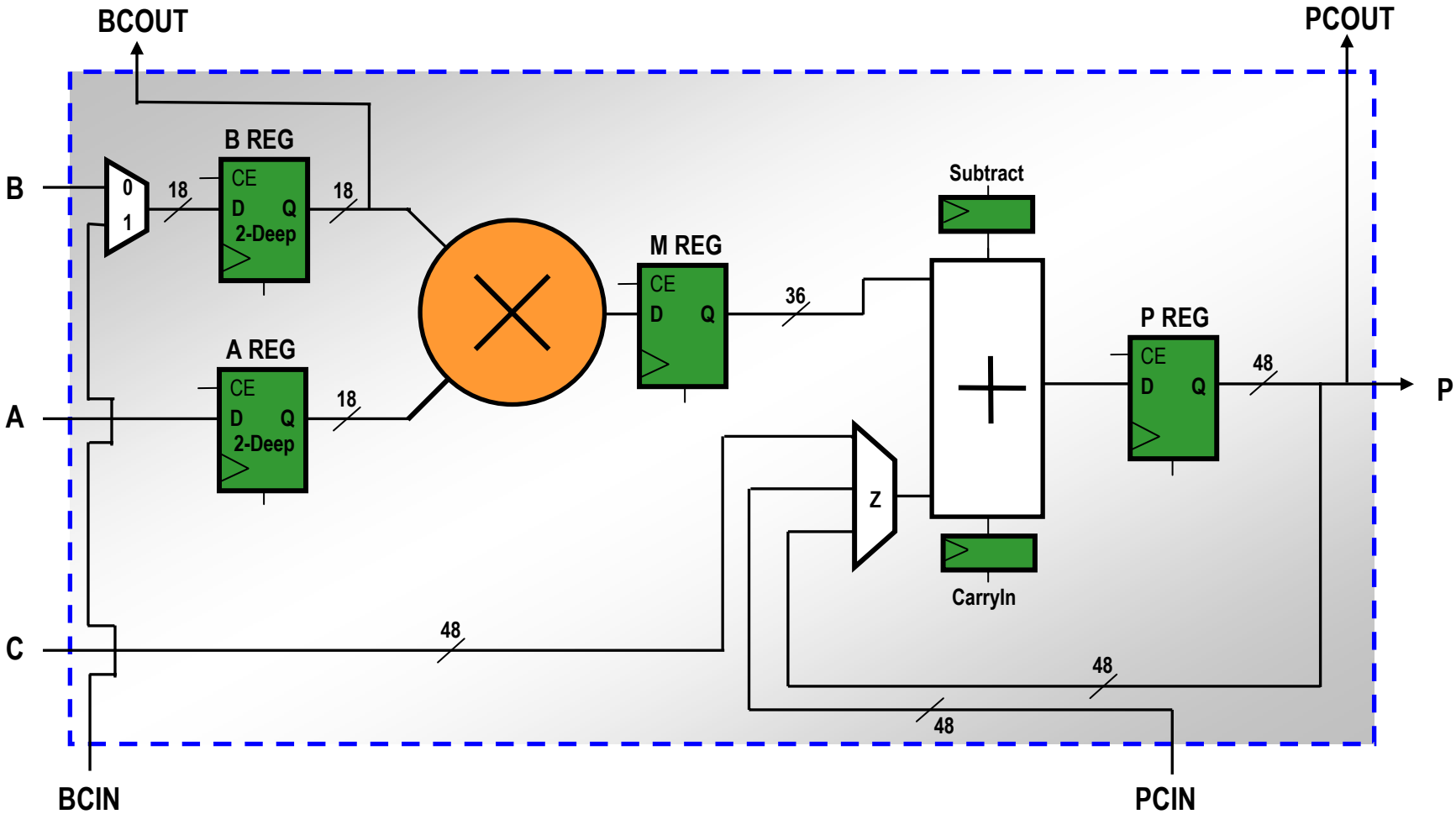
500 MHz maximum frequency in the fastest speed grade

The XtremeDSP Slice



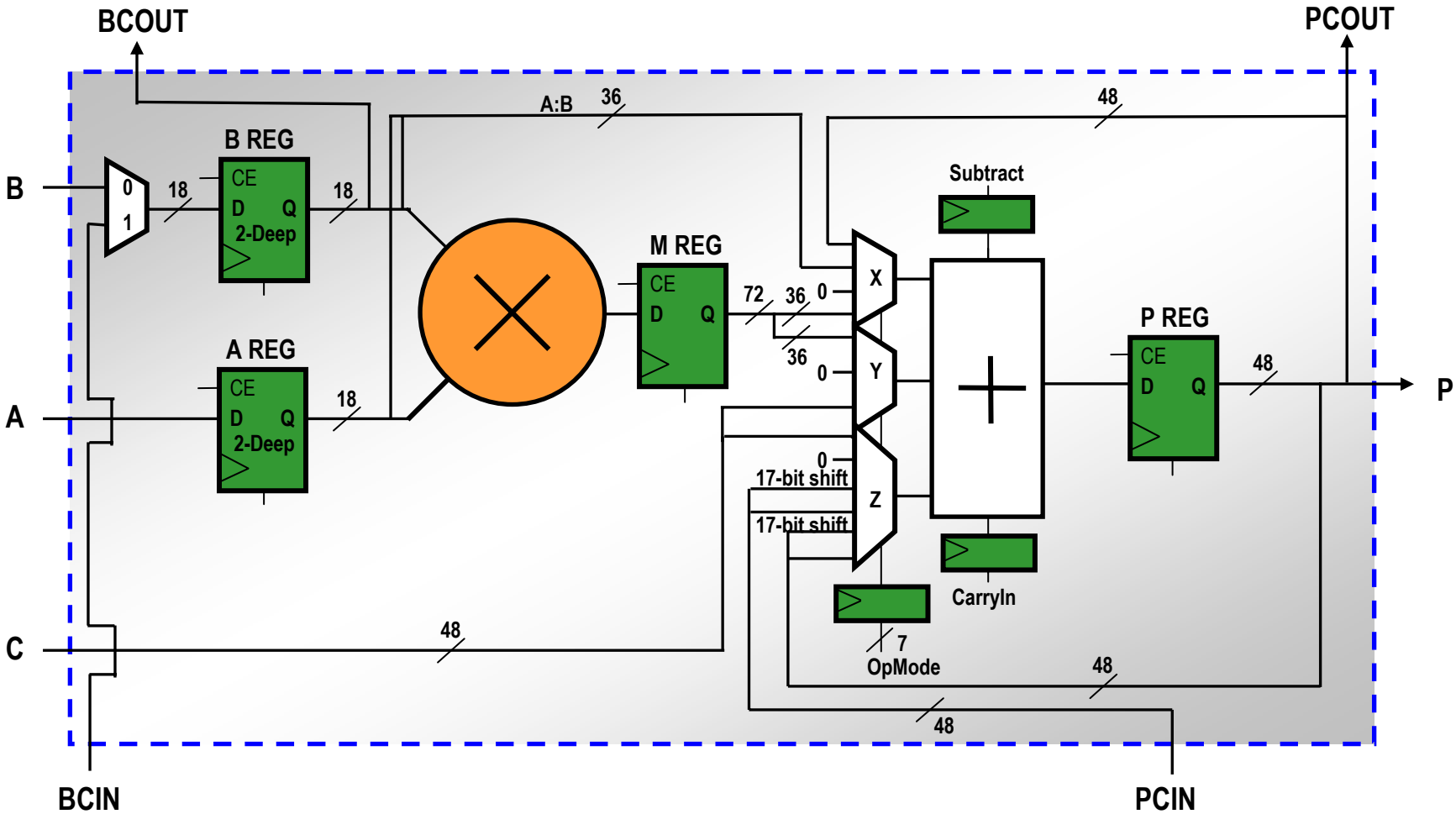
500 MHz maximum frequency in the fastest speed grade

The XtremeDSP Slice



500 MHz maximum frequency in the fastest speed grade

The XtremeDSP Slice



500 MHz maximum frequency in the fastest speed grade

Dynamically Reconfigurable DSP OPMODEs

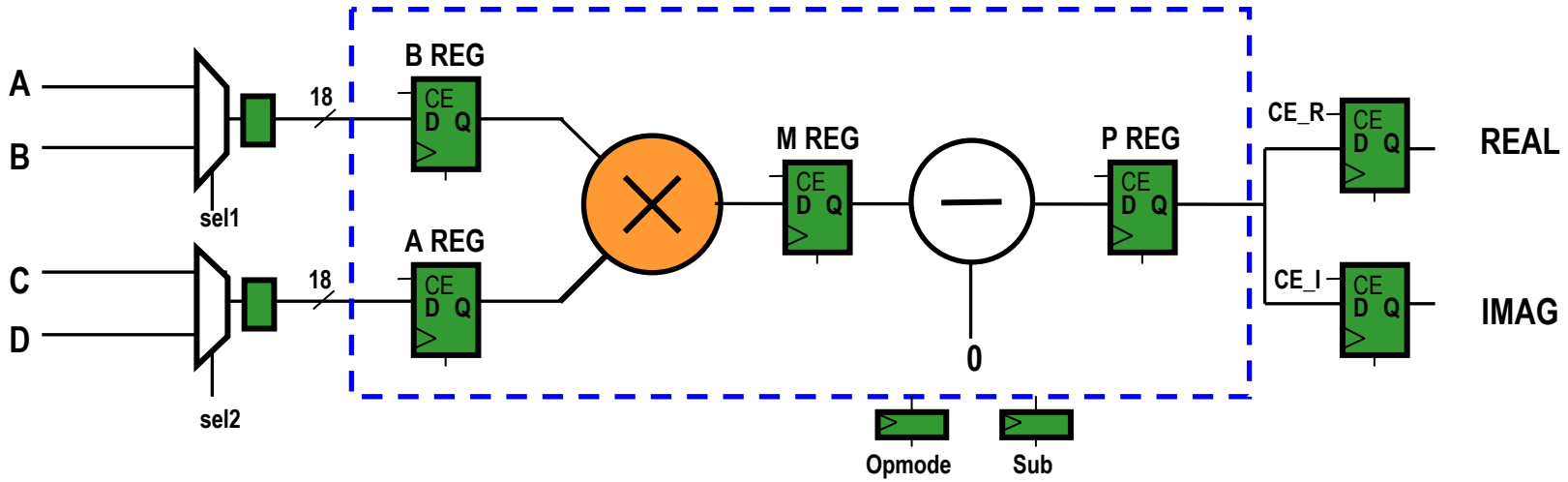
OpMode	Z			Y		X		Output
	6	5	4	3	2	1	0	
Zero	0	0	0	0	0	0	0	+/- Cin
Hold P	0	0	0	0	0	1	0	+/- (P + Cin)
A:B Select	0	0	0	0	0	1	1	+/- (A:B + Cin)
Multiply	0	0	0	0	1	0	1	+/- (A * B + Cin)
C Select	0	0	0	1	1	0	0	+/- (C + Cin)
Feedback Add	0	0	0	1	1	1	0	+/- (C + P + Cin)
36-Bit Adder	0	0	0	1	1	1	1	+/- (A:B + C + Cin)
P Cascade Select	0	0	1	0	0	0	0	PCIN +/- Cin
P Cascade Feedback Add	0	0	1	0	0	1	0	PCIN +/- (P + Cin)
P Cascade Add	0	0	1	0	0	1	1	PCIN +/- (A:B + Cin)
P Cascade Multiply Add	0	0	1	0	1	0	1	PCIN +/- (A * B + Cin)
P Cascade Add	0	0	1	1	1	0	0	PCIN +/- (C + Cin)
P Cascade Feedback Add Add	0	0	1	1	1	1	0	PCIN +/- (C + P + Cin)
P Cascade Add Add	0	0	1	1	1	1	1	PCIN +/- (A:B + C + Cin)
Hold P	0	1	0	0	0	0	0	P +/- Cin
Double Feedback Add	0	1	0	0	0	1	0	P +/- (P + Cin)
Feedback Add	0	1	0	0	0	1	1	P +/- (A:B + Cin)
Multiply-Accumulate	0	1	0	0	1	0	1	P +/- (A * B + Cin)
Feedback Add	0	1	0	1	1	0	0	P +/- (C + Cin)
Double Feedback Add	0	1	0	1	1	1	0	P +/- (C + P + Cin)
Feedback Add Add	0	1	0	1	1	1	1	P +/- (A:B + C + Cin)
C Select	0	1	1	0	0	0	0	C +/- Cin
Feedback Add	0	1	1	0	0	1	0	C +/- (P + Cin)
36-Bit Adder	0	1	1	0	0	1	1	C +/- (A:B + Cin)
Multiply-Add	0	1	1	0	1	0	1	C +/- (A * B + Cin)
17-Bit Shift P Cascade Select	1	0	1	0	0	0	0	Shift(PCIN) +/- Cin
17-Bit Shift P Cascade Feedback Add	1	0	1	0	0	1	0	Shift(PCIN) +/- (P + Cin)
17-Bit Shift P Cascade Add	1	0	1	0	0	1	1	Shift(PCIN) +/- (A:B + Cin)
17-Bit Shift P Cascade Multiply Add	1	0	1	0	1	0	1	Shift(PCIN) +/- (A * B + Cin)
17-Bit Shift P Cascade Add	1	0	1	1	1	0	0	Shift(PCIN) +/- (C + Cin)
17-Bit Shift P Cascade Add Add	1	0	1	1	1	1	1	Shift(PCIN) +/- (A:B + C + Cin)
17-Bit Shift Feedback	1	1	0	0	0	0	0	Shift(P) +/- Cin
17-Bit Shift Feedback Feedback Add	1	1	0	0	0	1	0	Shift(P) +/- (P + Cin)
17-Bit Shift Feedback Add	1	1	0	0	0	1	1	Shift(P) +/- (A:B + Cin)
17-Bit Shift Feedback Multiply Add	1	1	0	0	1	0	1	Shift(P) +/- (A * B + Cin)
17-Bit Shift Feedback Add	1	1	0	1	1	0	0	Shift(P) +/- (C + Cin)

- Over 40 Different Modes
- Each XtremeDSP Slice individually controllable
- Change operation in a single clock cycle
- Enables resource sharing for maximum utilization

Dynamic Opmode

Complex Multiplier

$$(a+jb).(c+jd) = [a.c - b.d] + j[b.c + a.d]$$



CLK Cycle	Function	OPMODE	Sub	Sel1	Sel2	CE_R	CE_I

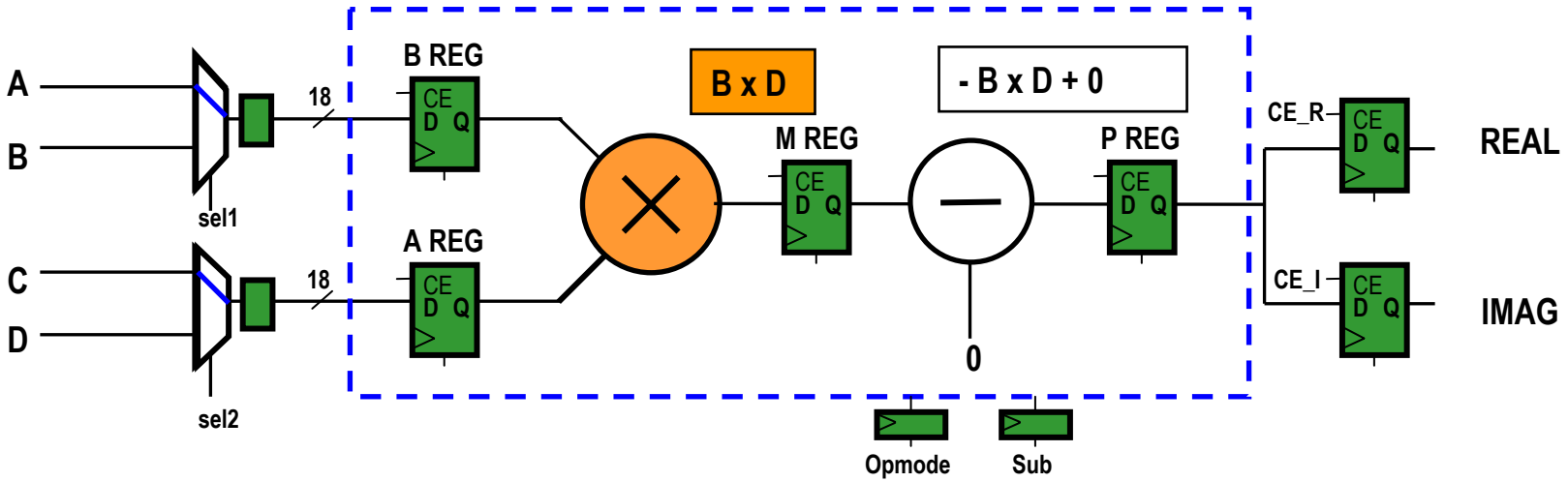
NOTE: Control signals can be stored in a Disributed Memory

Performance
400 Mhz
Size:
1 XDSP Slice
59 Slices
(5 for control)

Dynamic Opmode

Complex Multiplier

$$(a+jb).(c+jd) = [a.c - b.d] + j[b.c + a.d]$$



CLK Cycle	Function	OPMODE	Sub	Sel1	Sel2	CE_R	CE_I
1	Multiply Subtract	0001010	1	0	0	0	1

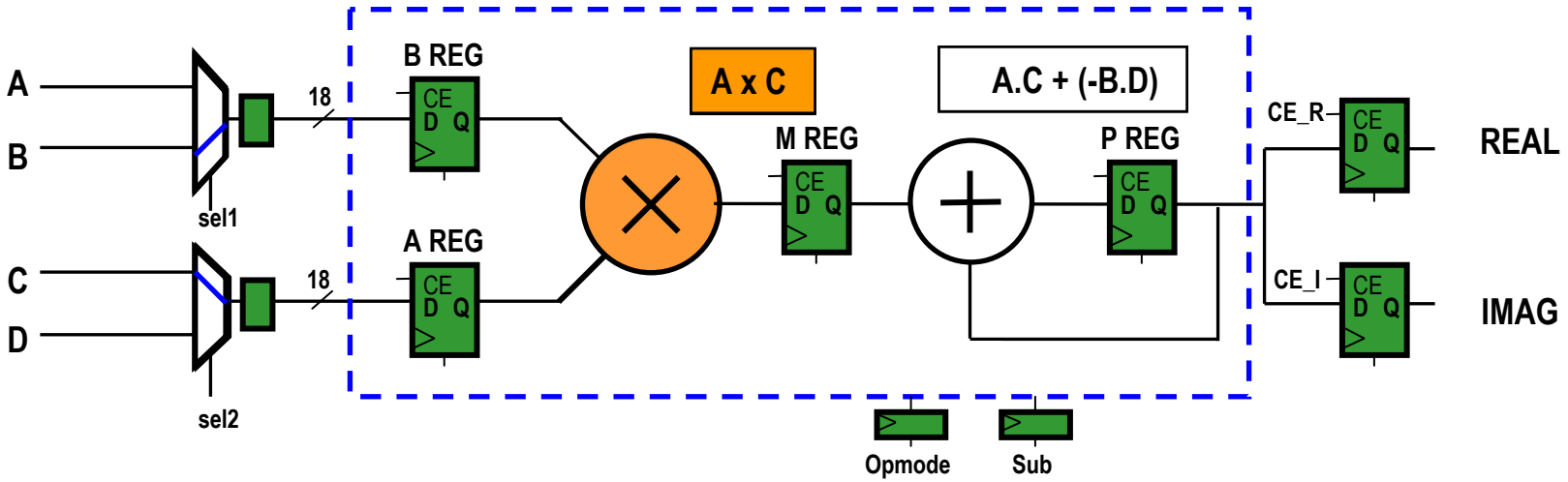
NOTE: Control signals can be stored in a Disributed Memory

Performance
400 Mhz
Size:
1 XDSP Slice
59 Slices
(5 for control)

Dynamic Opmode

Complex Multiplier

$$(a+jb).(c+jd) = [a.c - b.d] + j[b.c + a.d]$$



CLK Cycle	Function	OPMODE	Sub	Sel1	Sel2	CE_R	CE_I
1	Multiply Subtract	0001010	1	0	0	0	1
2	Multiply Accumulate	0101010	0	1	0	0	0

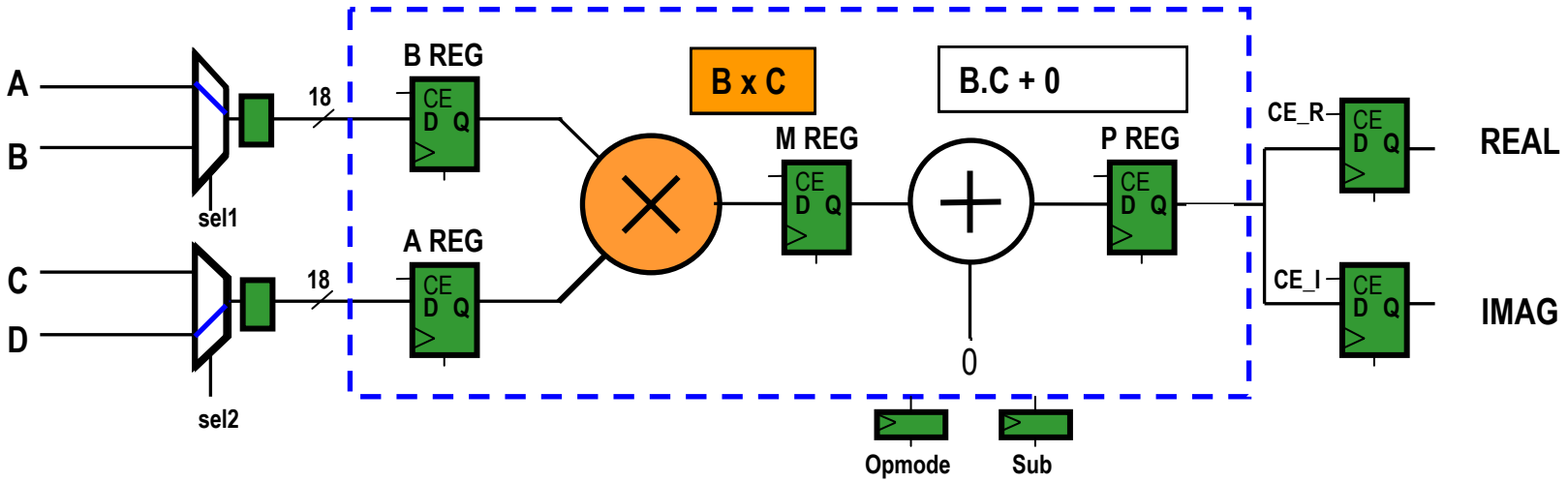
NOTE: Control signals can be stored in a Disributed Memory

Performance
400 Mhz
Size:
1 XDSP Slice
59 Slices
(5 for control)

Dynamic Opmode

Complex Multiplier

$$(a+jb).(c+jd) = [a.c - b.d] + j[b.c + a.d]$$



CLK Cycle	Function	OPMODE	Sub	Sel1	Sel2	CE_R	CE_I
1	Multiply Subtract	0001010	1	0	0	0	1
2	Multiply Accumulate	0101010	0	1	0	0	0
3	Multiply	0001010	0	0	1	1	0

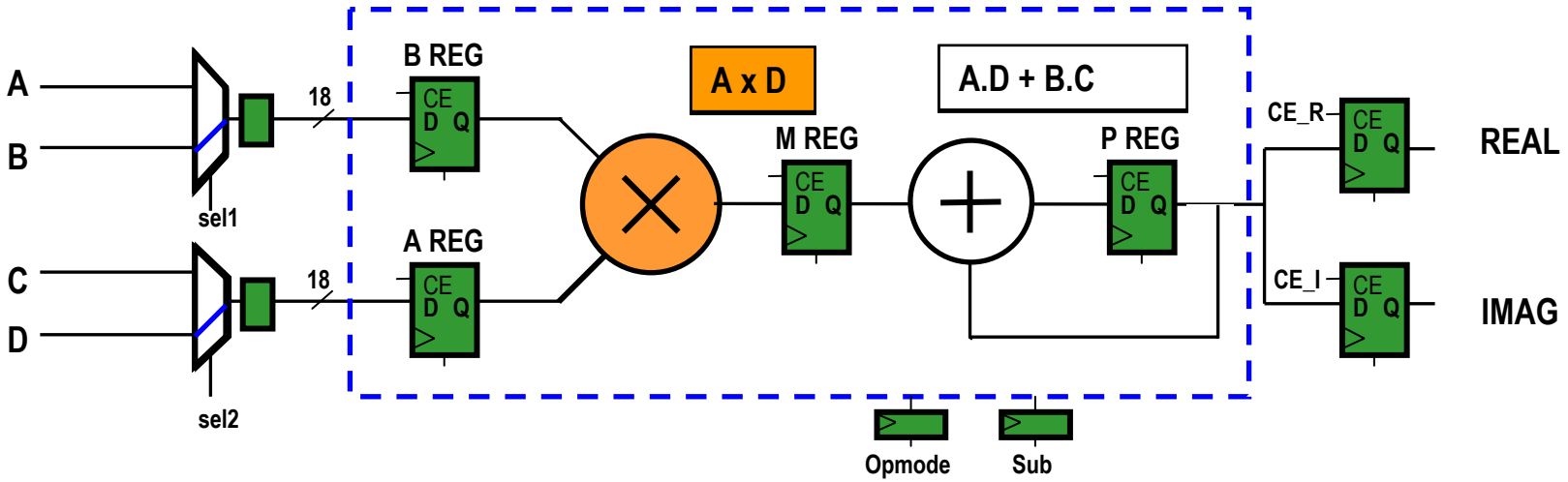
NOTE: Control signals can be stored in a Disributed Memory

Performance
400 Mhz
Size:
1 XDSP Slice
59 Slices
(5 for control)

Dynamic Opmode

Complex Multiplier

$$(a+jb).(c+jd) = [a.c - b.d] + j[b.c + a.d]$$



CLK Cycle	Function	OPMODE	Sub	Sel1	Sel2	CE_R	CE_I
1	Multiply Subtract	0001010	1	0	0	0	1
2	Multiply Accumulate	0101010	0	1	0	0	0
3	Multiply	0001010	0	0	1	1	0
4	Multiply Accumulate	0101010	0	1	1	0	0

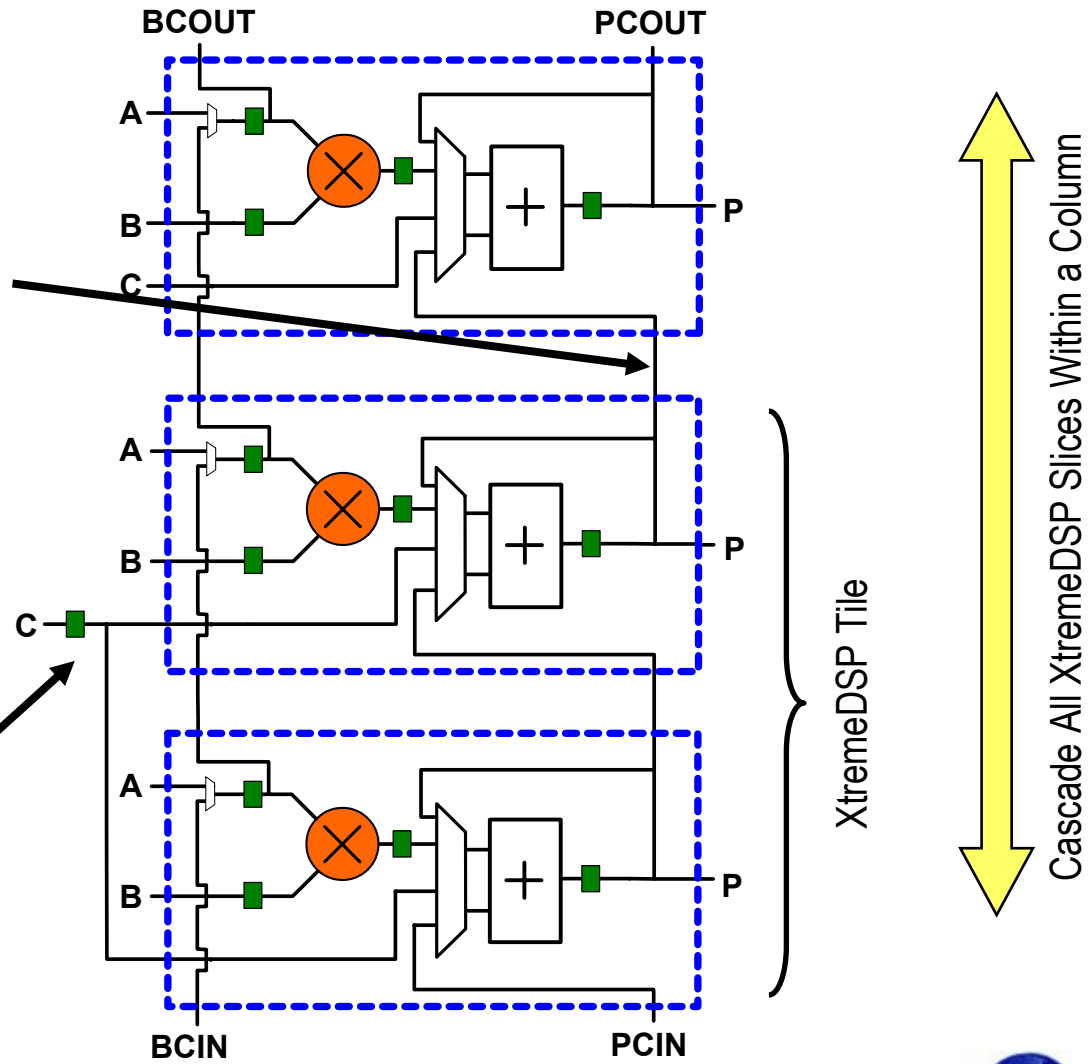
NOTE: Control signals can be stored in a Disributed Memory

Performance
400 Mhz
Size:
1 XDSP Slice
59 Slices
(5 for control)

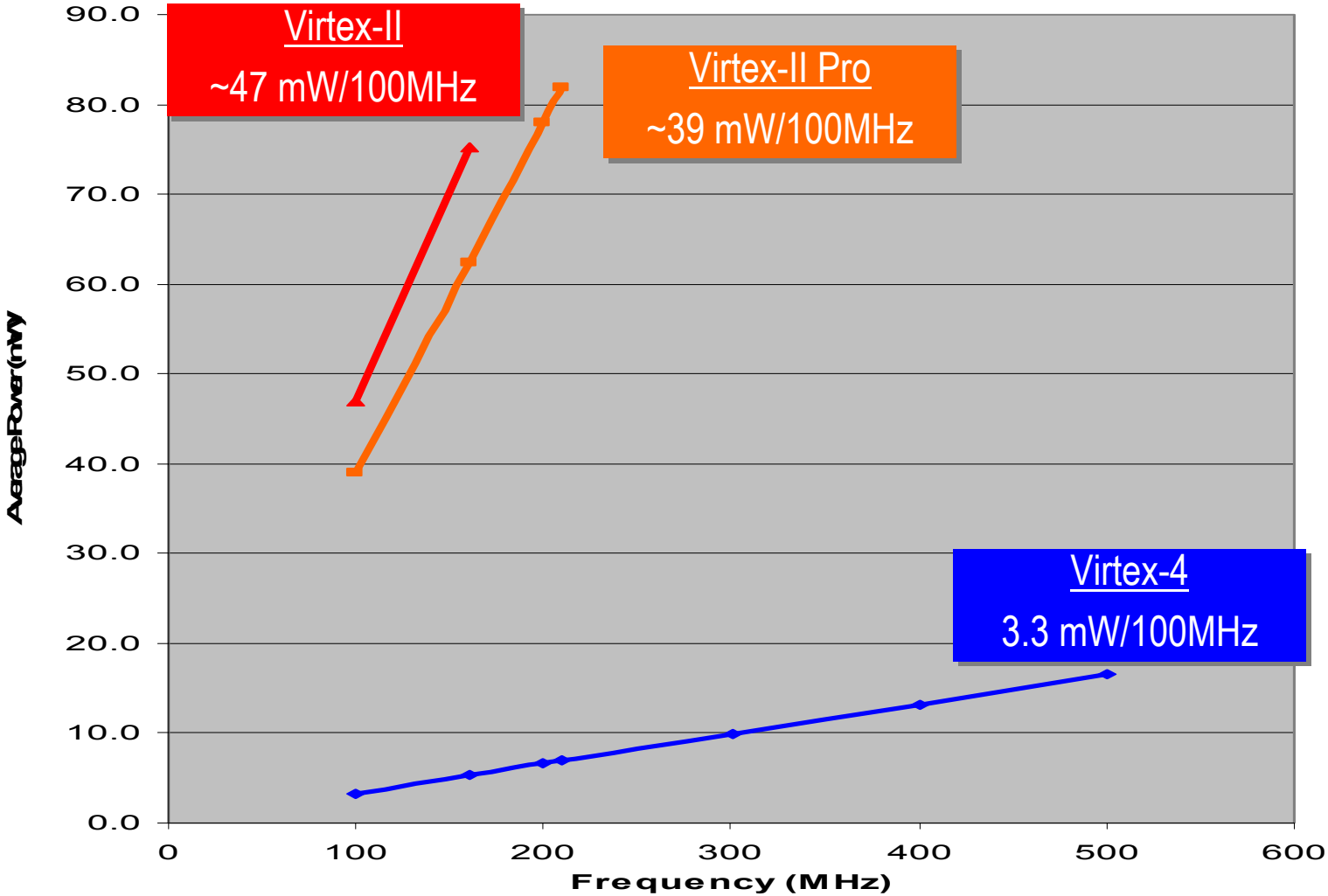
XtremeDSP Slice Cascade

XtremeDSP Slices can be cascaded together with direct dedicated interconnect from slice to slice. A Columns length determined by the number of XtremeDSP Slices in the column

C Input is shared between two XtremeDSP Slices or a single tile. Be careful to select algorithms that do not depend on a C input for each XtremeDSP Slice



DSP48 Slice Power Consumption

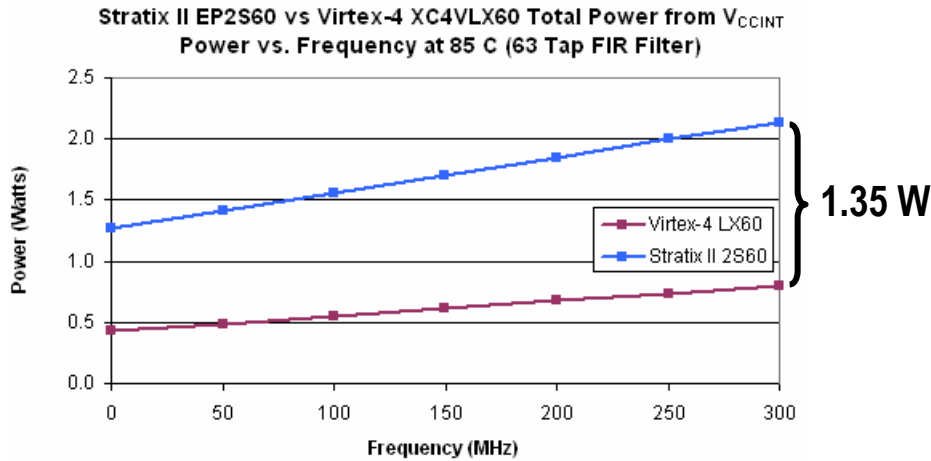


Conditions: TT, 25C, nominal voltage, Fully pipelined multiply-add mode, random vectors

DSP48 Power Test for 63 Tap FIR Filter

(Stratix II EP2S60 and Xilinx Virtex-4 XC4VLX60)

Description	**Test using 63 section asymmetrical taps with 18 bit data stream and fixed 18 bit coefficients. Virtex-4 uses 63 DSP48 blocks all in a single column. Stratix II uses 4 tap sections in a DSP block. Reconciling summation of 4 tap chunks is handle by Stratix II 3 input adders in layers of 6, 2, and 1. Same Stimulus VHDL code.**
Virtex-4 Logic Functions	**64 DSP48 and 0 Slices (1 DSP Block used as stimulus of the filter)**
Stratix II Logic Functions	**128 9 Bit DSP Elements and 187 ALMs (1/4 of 1 DSP Block is used as a stimulus for the filter)**

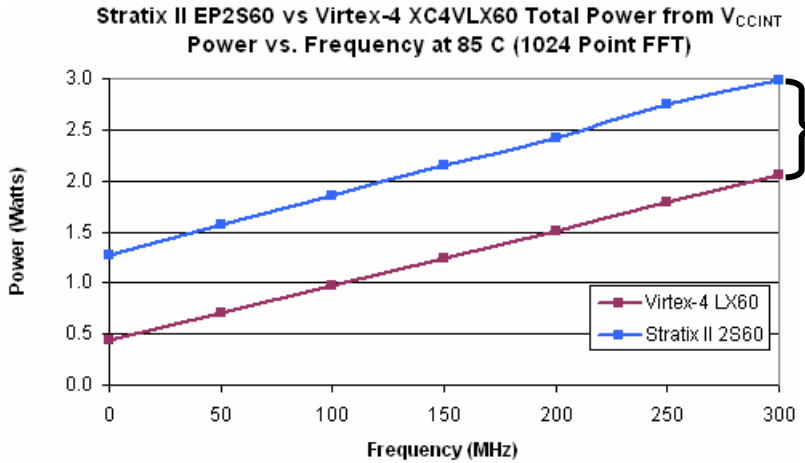


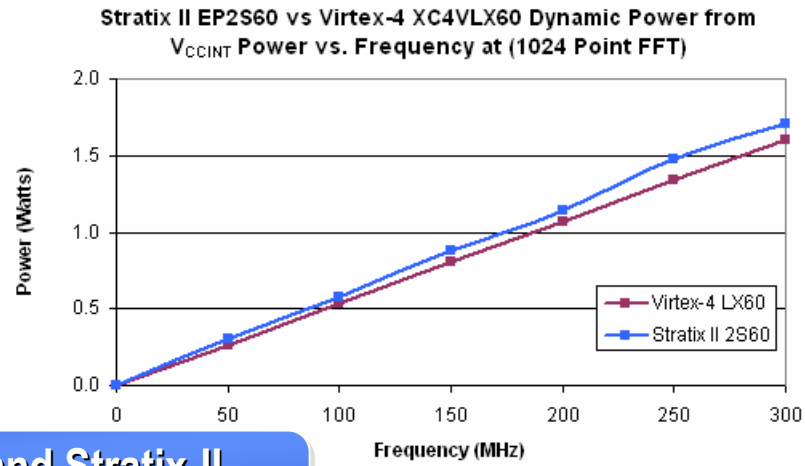
> 1 Watt of power difference between Virtex-4 and Stratix-II in DSP Applications

Measured 1024 Point FFT Power

(Stratix II EP2S60 and Xilinx Virtex-4 XC4VLX60)

Description	Test is on a streaming 1024 Point FFT. Test uses small amount of DSP48 block in Virtex-4 and equivalent processing in Stratix II. Design has several thousand LUTs, registers and block RAM.
Virtex-4 Logic Functions	2548 Slices, 17 DSP48s, and 9 BlockRAM
Stratix II Logic Functions	2917 ALMS, 38 M4K Blocks, 2 M512 Blocks, and 24 - 9 Bit DSP Elements





> 1 Watt of power difference between Virtex-4 and Stratix-II in DSP Applications

Filter Techniques

For this analysis the software used was:

ISE 7.1.1i
Quartus 4.1 sp1
FIR Compiler 3.2.1

The FIR Filter

The most common DSP function implemented in a Xilinx devices is the Finite Impulse Response filter:

Viewed as an Equation

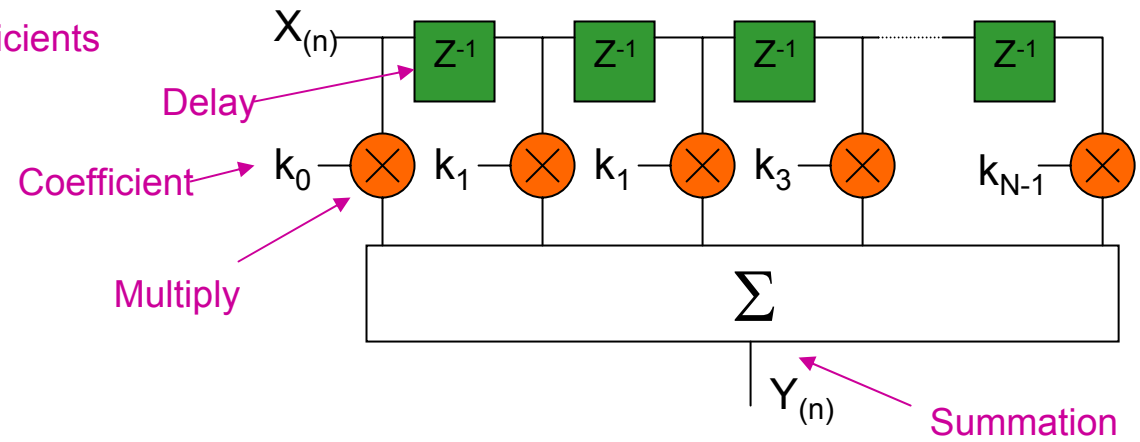
$$Y_{(n)} = \sum_{i=0}^{i=N-1} k_i . X_{(n-i)}$$

Coefficients

Accumulate N times

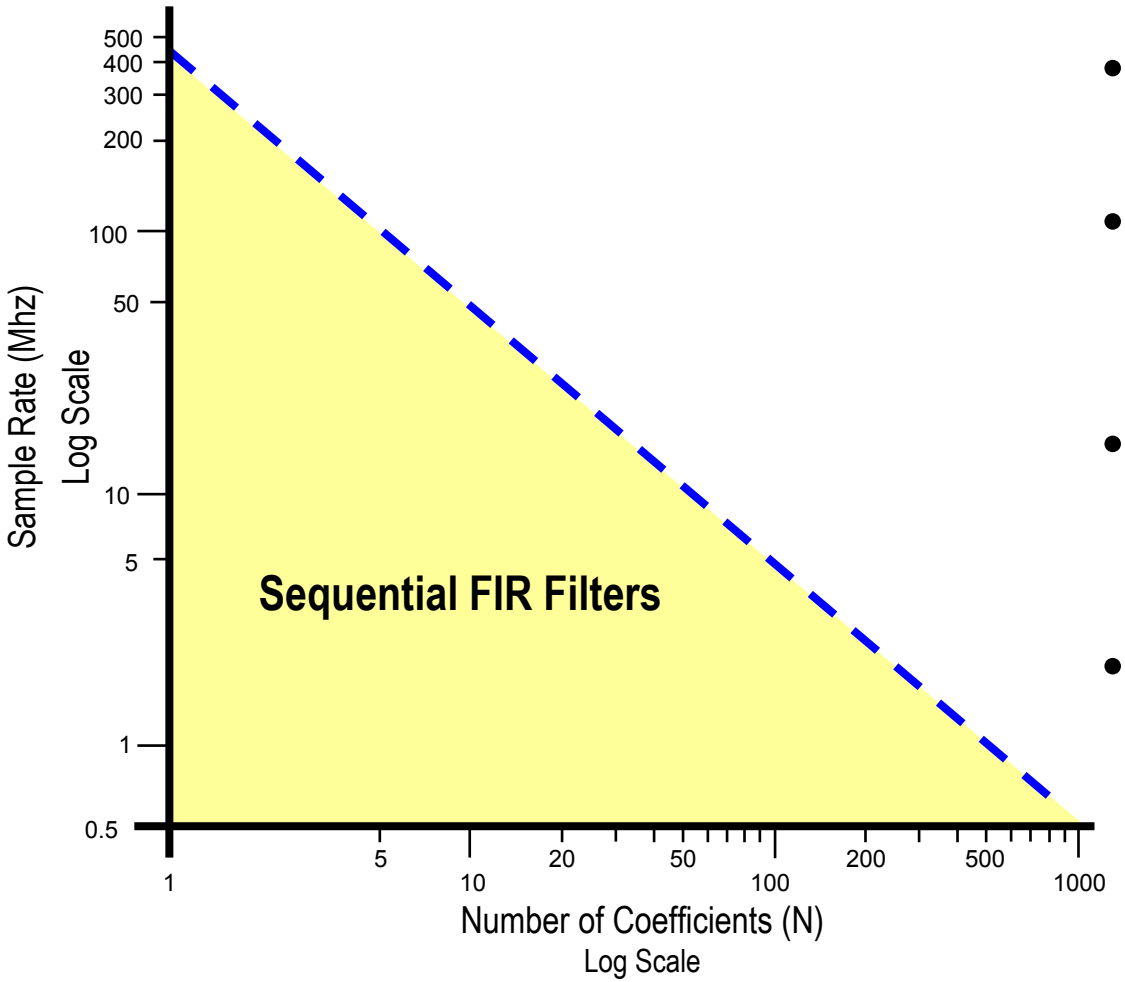
Multiply

Viewed as a Diagram



How do we implement these filter in Virtex-4?

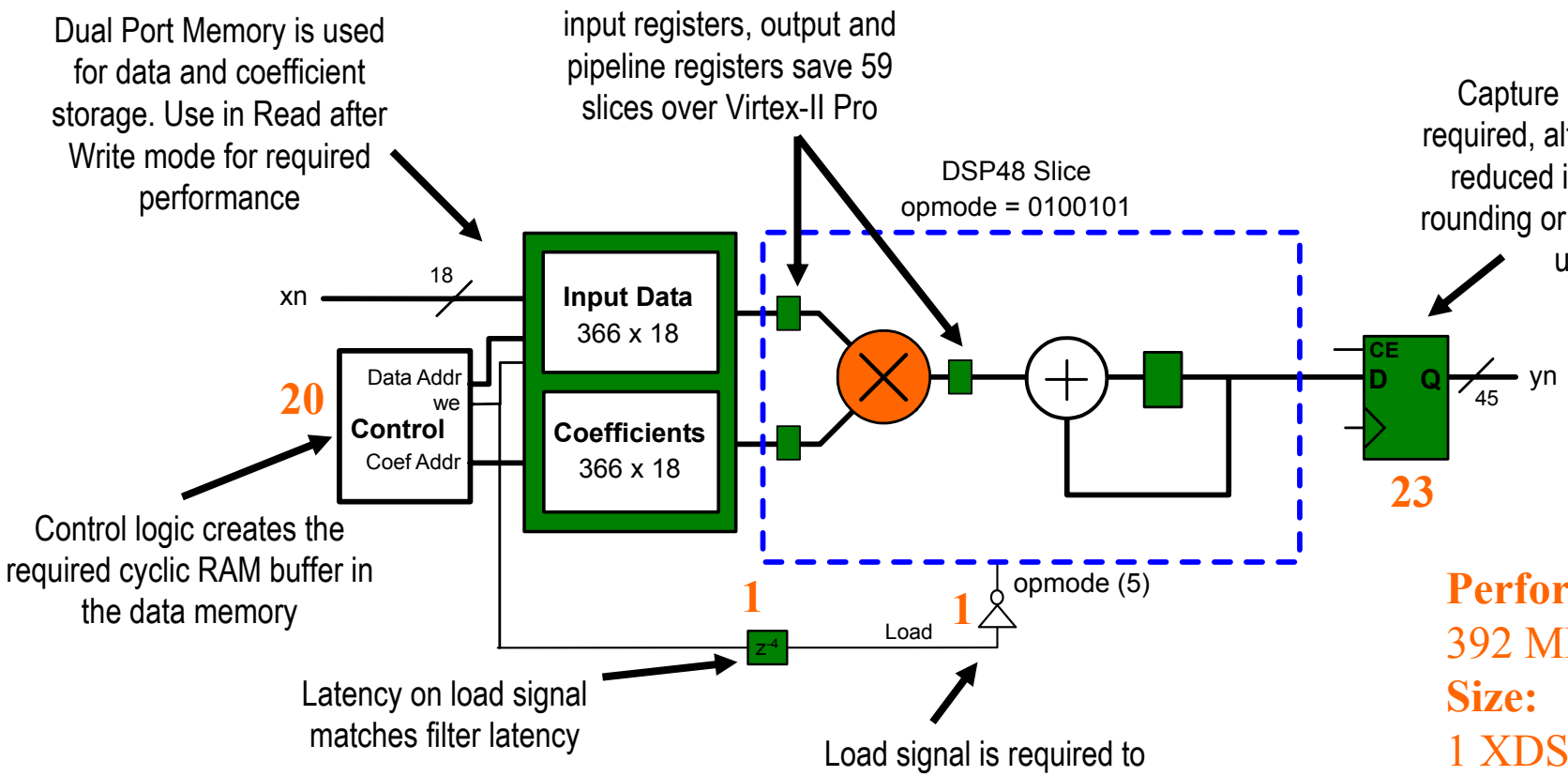
Sequential FIR Filters



- Firstly consider one multiplier based FIR Filters.
- Processing of the filter coefficients is done in a sequential fashion
- Line is where this architecture can no longer meet performance requirements
- Line has been raised in Virtex-4 due to higher clock performance of Xtreme DSP Slice

Virtex-4 MAC FIR Filter

Filter Specification: Sampling Frequency = 1.2288 Mhz, Coefficients = 366

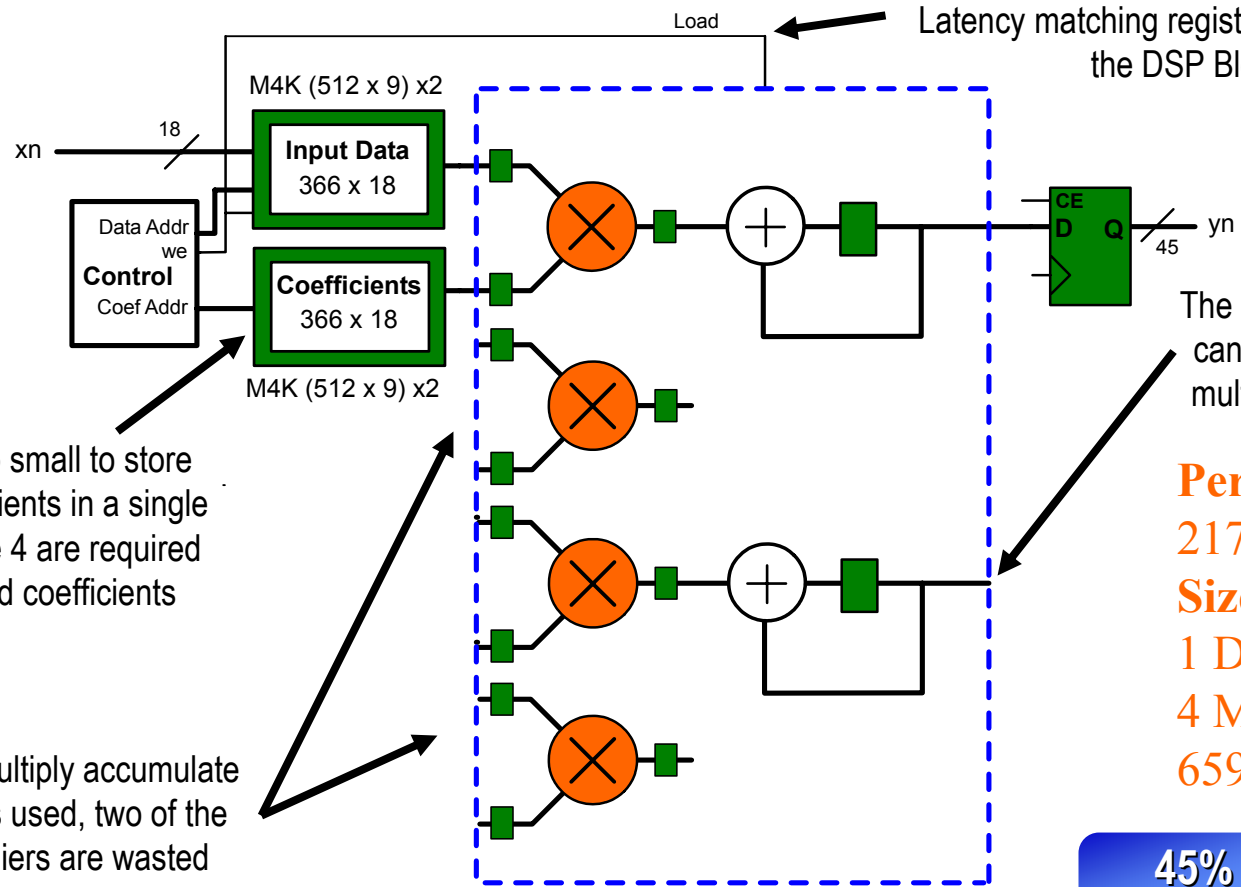


Max Sample Rate = $\frac{\text{Clock Rate}}{\text{Number of Taps}}$

Performance:
392 MHz (-10)
Size:
1 XDSP Slice
1 Block RAM
45 Slices

Stratix-II MAC FIR Filter

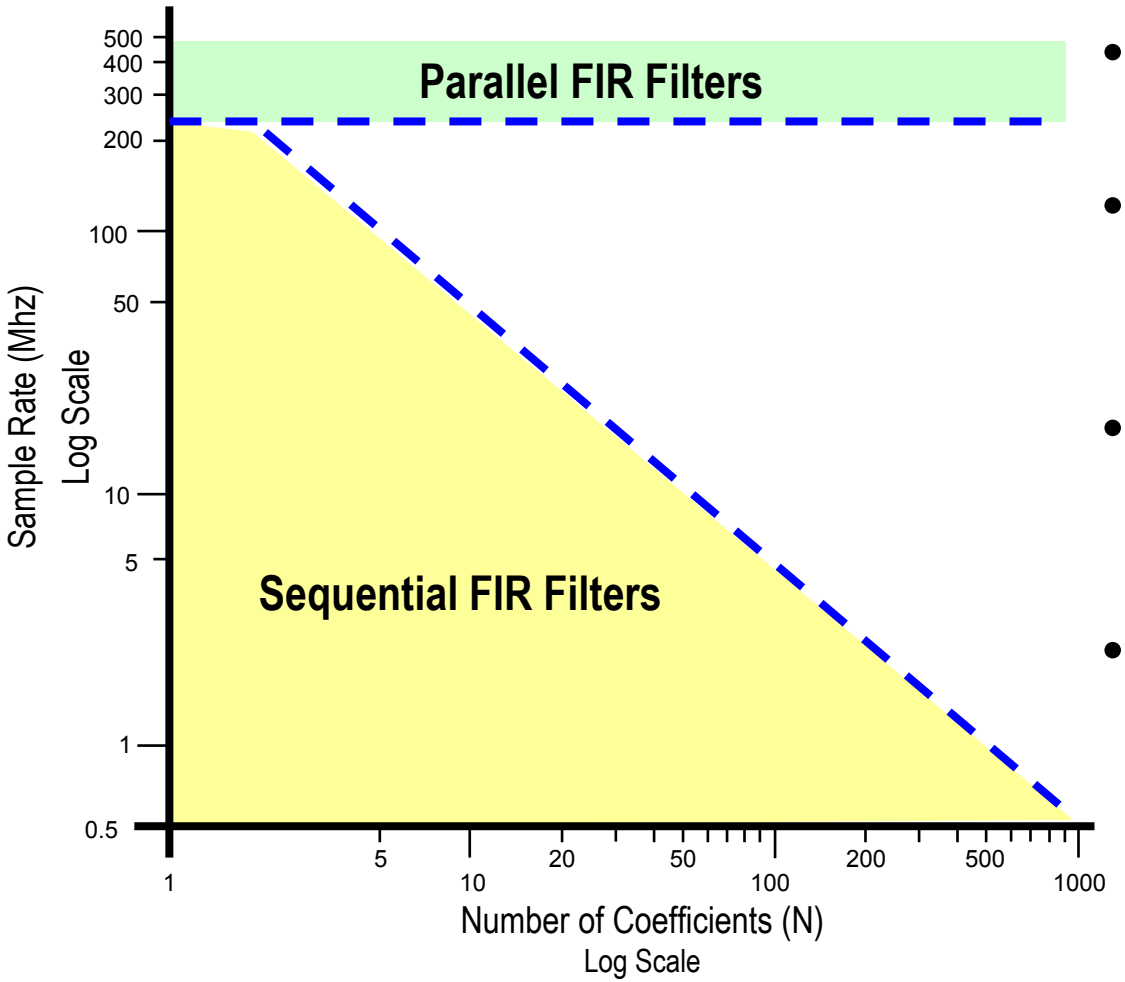
Filter Specification: Sampling Frequency = 1.2288 Mhz, Coefficients = 366



Performance:
217 MHz (-5)
Size:
1 DSP Block (1 Mult)
4 M4K
659 ALMs

45% Slower Performance

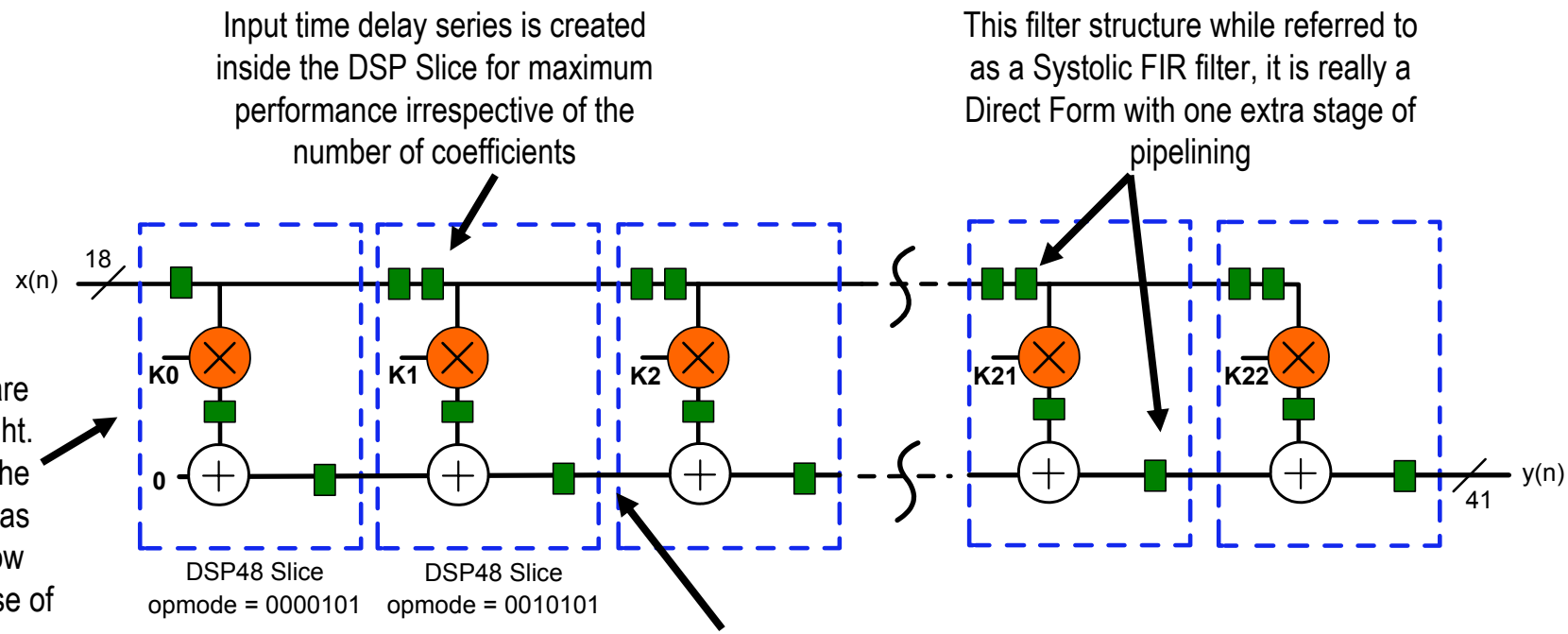
Parallel FIR Filters



- Now consider one multiplier per coefficient
- Processing of the filter coefficients is done in a parallel fashion
- Line is where this architecture is required as less than 2 clock cycles are available
- Line has been raised in Virtex-4 due to higher clock performance of Xtreme DSP Slice

Virtex-4 Systolic FIR Filter

Filter Specification: Sampling Frequency = 400 Mhz, Coefficients = 23

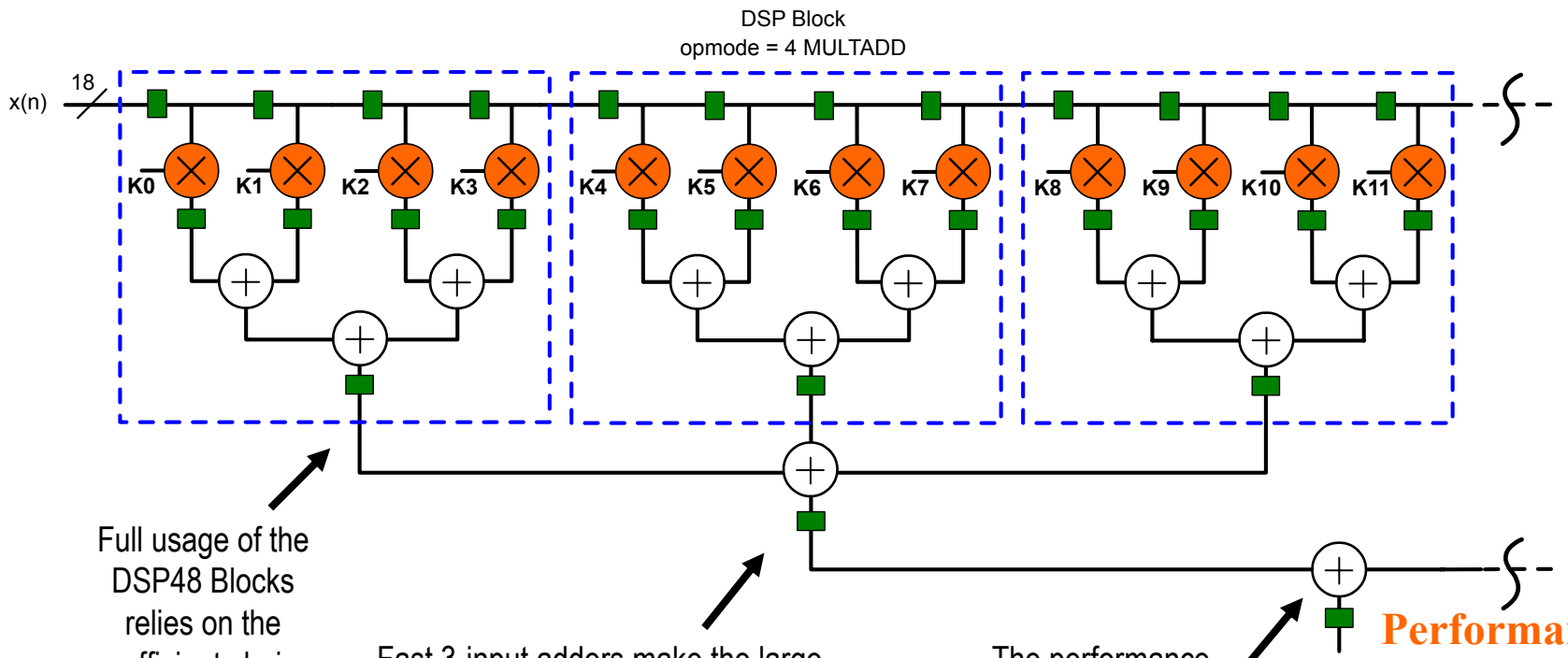


Max Sample Rate = Clock Rate

Performance:
400 MHz (-10)
Size:
23 XDSP Slice

Stratix-II Parallel FIR Filter

Filter Specification: Sampling Frequency = 400 Mhz, Coefficients = 23



Full usage of the DSP48 Blocks relies on the coefficients being divisible by 4

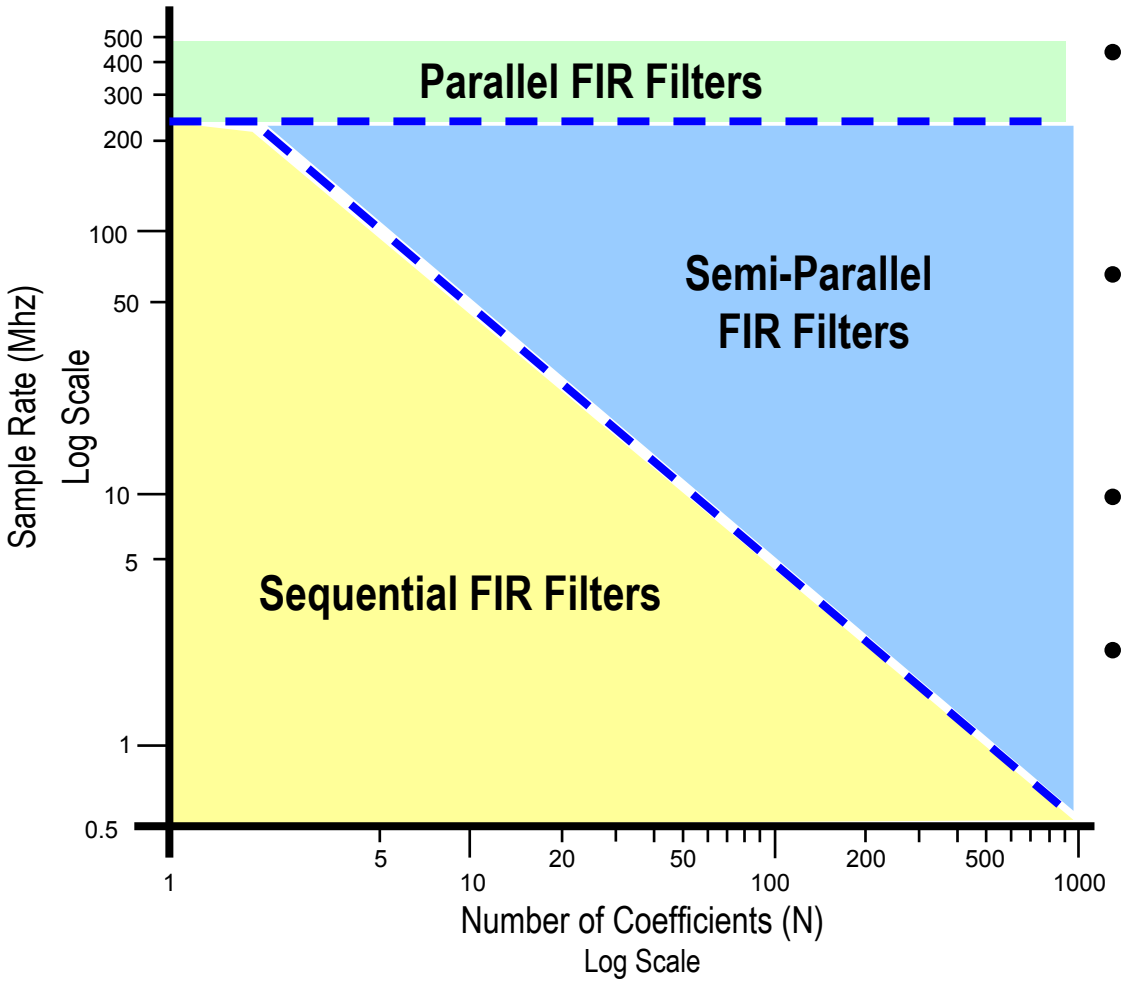
Fast 3-input adders make the large adder trees much more resource efficient

The performance bottleneck will still always be the fabric adder, which will greatly limit performance

48% Slower Performance

Performance:
207 MHz (-5)
Size:
6 DSP Blocks
(23 Mults)
744 ALMs

Semi-Parallel FIR Filters



- Now consider in between scenario. Multiple coefficients per multiplier (M).
- Processing of the filter coefficients is done in a semi-parallel fashion
- Boundary lines determined by the other techniques
- Line has been raised in Virtex-4 due to higher clock performance of Xtreme DSP Slice

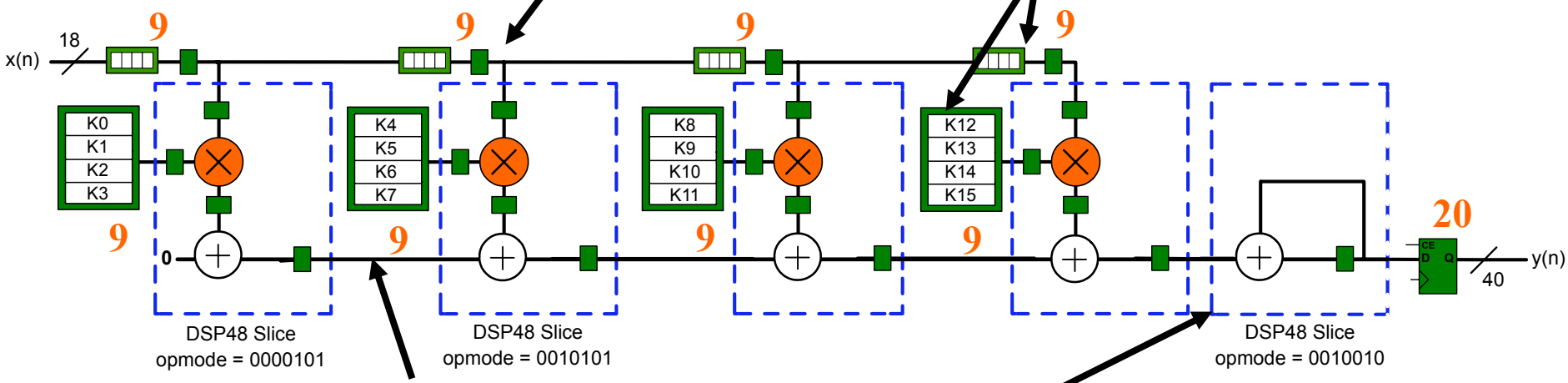
Virtex-4

4 Multiplier Systolic SP FIR

Filter Specification: Sampling Frequency = 74.176 MHz, Coefficients = 16

Input time delay series is created outside the XtremeDSP Slice as SRL16E are required to store the set of inputs to drive each engine

The important thing to note about the addressing is that each SRL16E and coefficient memory buffer have identical addressing



The adder chain pipeline register is compensated for in the addressing of the memories. Hence each buffers address is delayed by one clock cycle

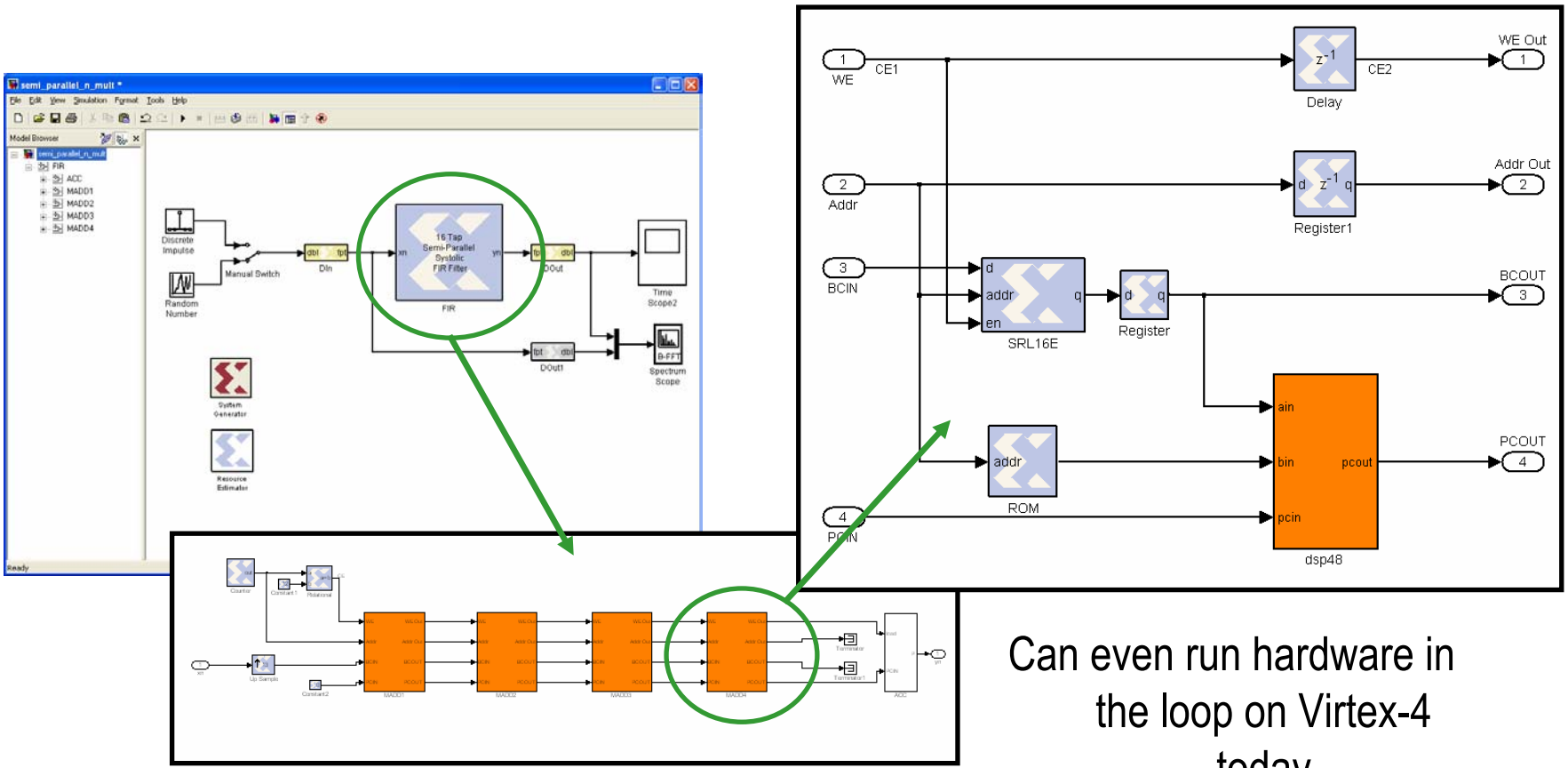
An extra Xtreme DSP Slice is require to accumulate the results over the 4 clock cycles required before the slower capture register grabs the final output

Performance:
400 MHz (-10)
Size:
5 XDSP Slice
104 Slices

$$\textbf{Max Sample Rate} = \frac{\text{Clock Rate x Number of Multipliers}}{\text{Number of Taps}}$$

4 Multiplier Systolic SP FIR

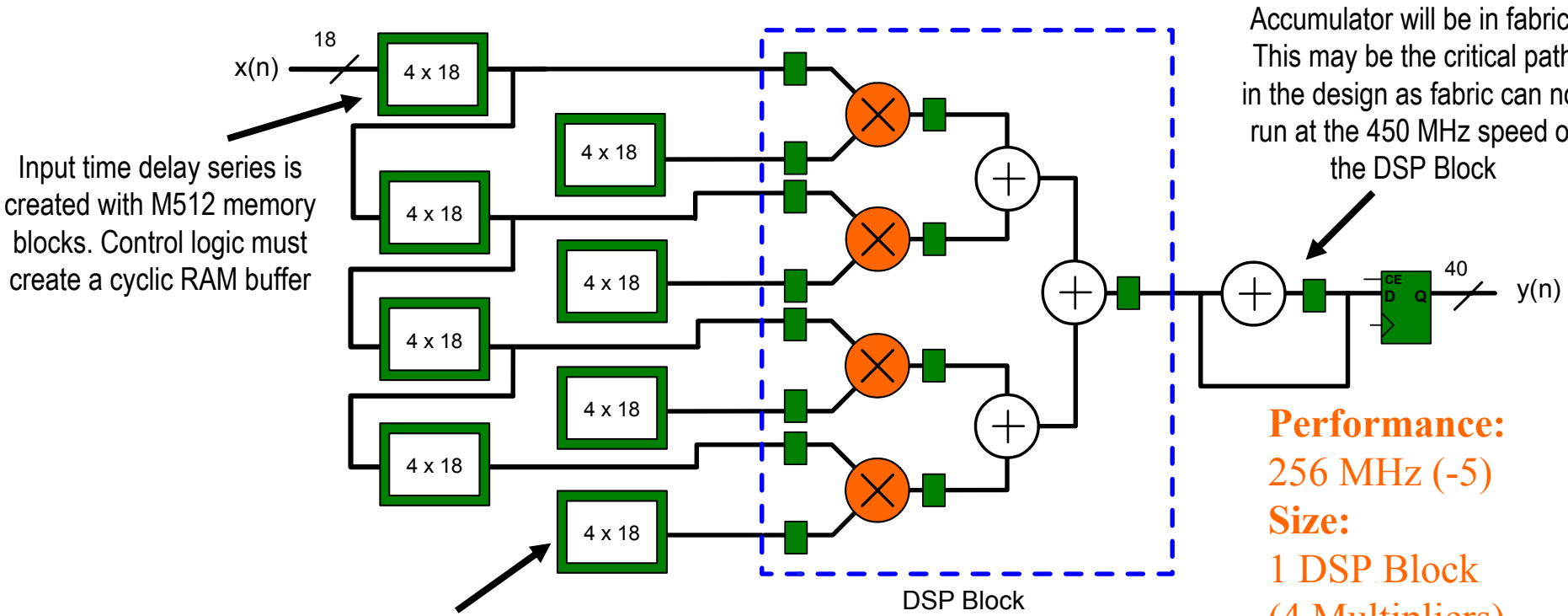
System Generator Implementation



Can even run hardware in the loop on Virtex-4 today

Stratix-II
4 Multiplier Semi-Parallel FIR

Filter Specification: Sampling Frequency = 74.176 MHz, Coefficients = 16



Performance:
256 MHz (-5)
Size:
1 DSP Block
(4 Multipliers)
188 ALMS

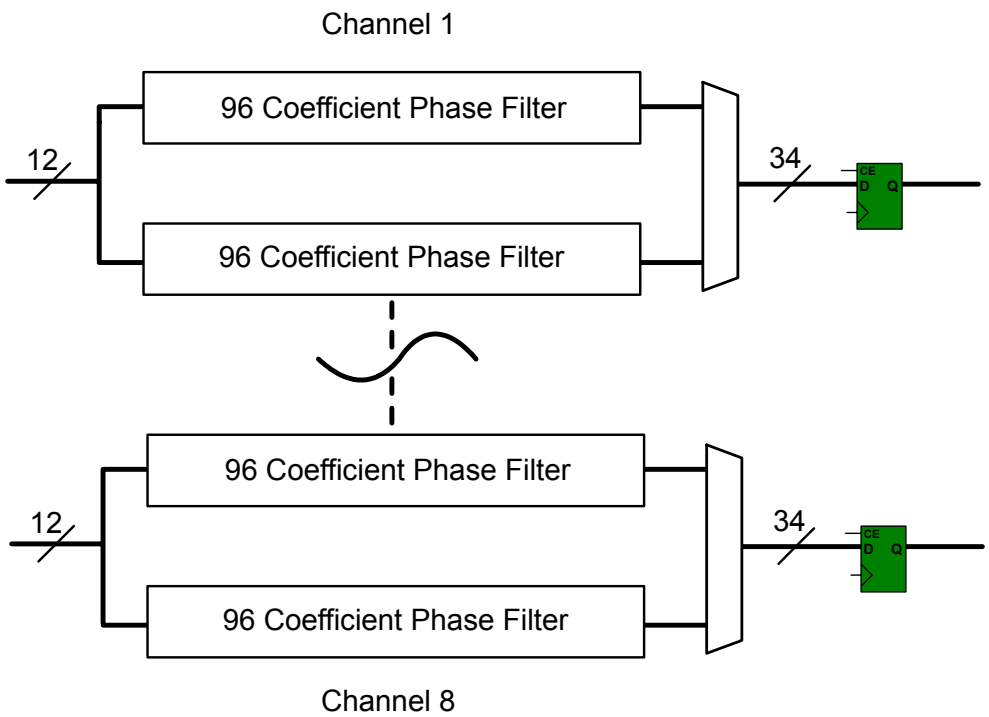
36% Slower Performance

Multi-Channel Multi-Rate FIR Filters

Virtex-4
Multi-Channel Multi-Rate FIR

Filter Specification:

Input Frequency = 3.84 Mhz, Coefficients = 192, Interpolation Rate Change = 2,
Channels = 8, Data Width = 12-bit, Coefficient Width = 15-bits



Slicing up the Pie:

Total number of coefficients = 1536

96 x 16 is the coefficient Matrix:

Option 1:

16 Sequential MACC Engines

96 clk cycles, Clock Speed: 368 Mhz

Option 2:

1 Semi-parallel 12 Multiplier FIR

8 cycle per phase, 16 phases = 128 clk cycles

Clock Speed: 491.52 Mhz

Option 3: Increase coefficients to 196

1 Semi-parallel 14 Multiplier FIR

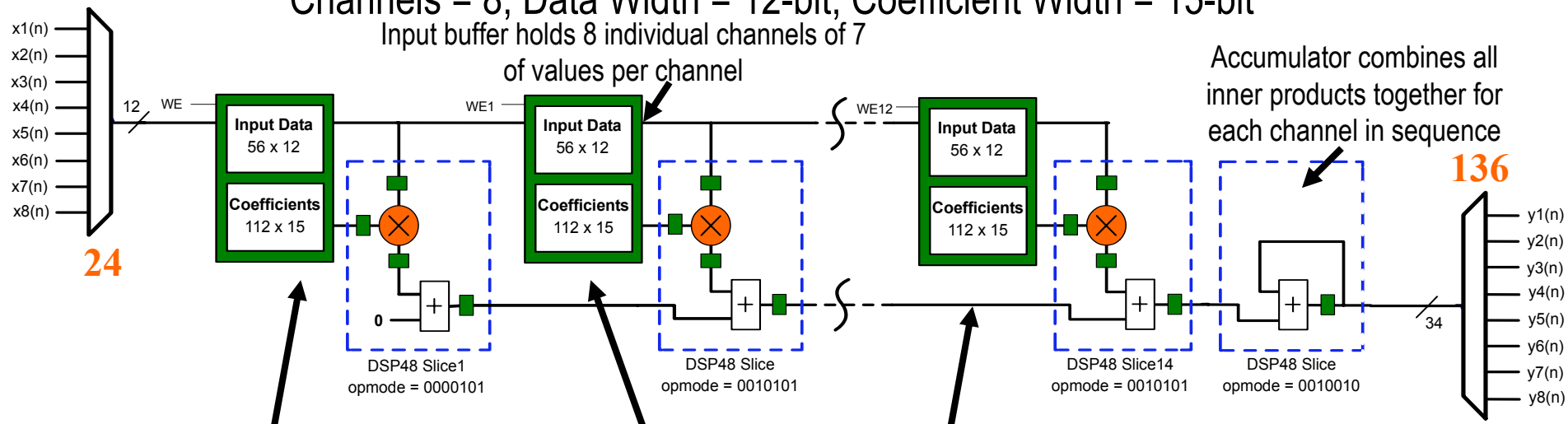
7 cycle per phase, 16 phases = 112 clk cycles

Clock Speed: 430.08 Mhz

Virtex-4
Multi-Channel Multi-Rate FIR

Filter Specification:

Input Frequency = 3.84 Mhz, Coefficients = 192+4, Interpolation Rate Change = 2,
Channels = 8, Data Width = 12-bit, Coefficient Width = 15-bit



Input buffer holds 8 individual channels of 7 of values per channel

Accumulator combines all inner products together for each channel in sequence

Embedded Control technique is used to reduce slice count

Addressing scheme is the same as the semi-parallel techniques we have learnt about

Coefficient buffer holds 7 coefficients of each 16 individual phase filter

Performance

447 MHz (-11)

Size:

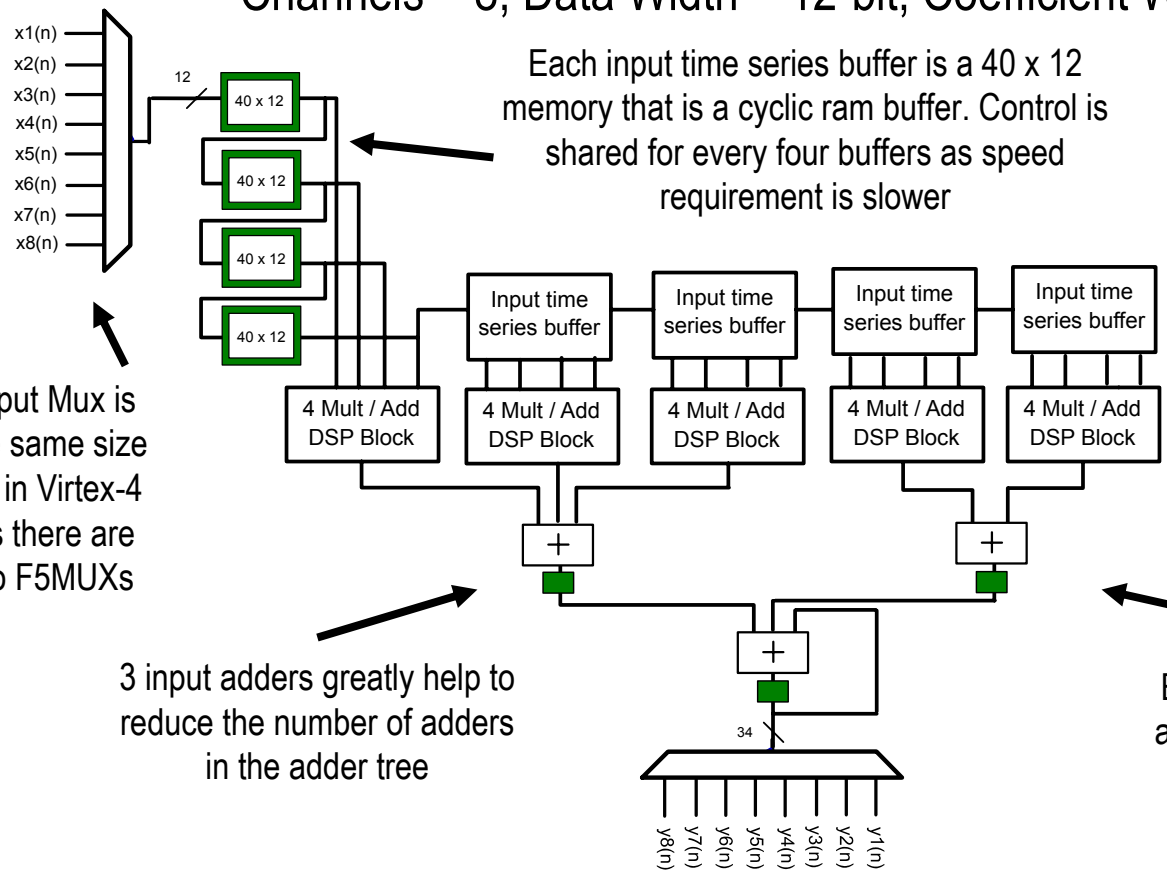
15 XDSP Slice
14 Block RAM
233 Slices
(73 for control)

$$\textbf{Max Input Sample Rate} = \frac{\text{Clock Rate x Number of Multipliers}}{\text{Number of Taps x Number of Channels}}$$

Stratix-II
Multi-Channel Multi-Rate FIR

Filter Specification:

Input Frequency = 3.84 Mhz, Coefficients = 192, Interpolation Rate Change = 2, Channels = 8, Data Width = 12-bit, Coefficient Width = 15-bit



Each input time series buffer is a 40 x 12 memory that is a cyclic ram buffer. Control is shared for every four buffers as speed requirement is slower

Required Clock speed is 295 MHz.

Input Mux is the same size as in Virtex-4 as there are no F5MUXs

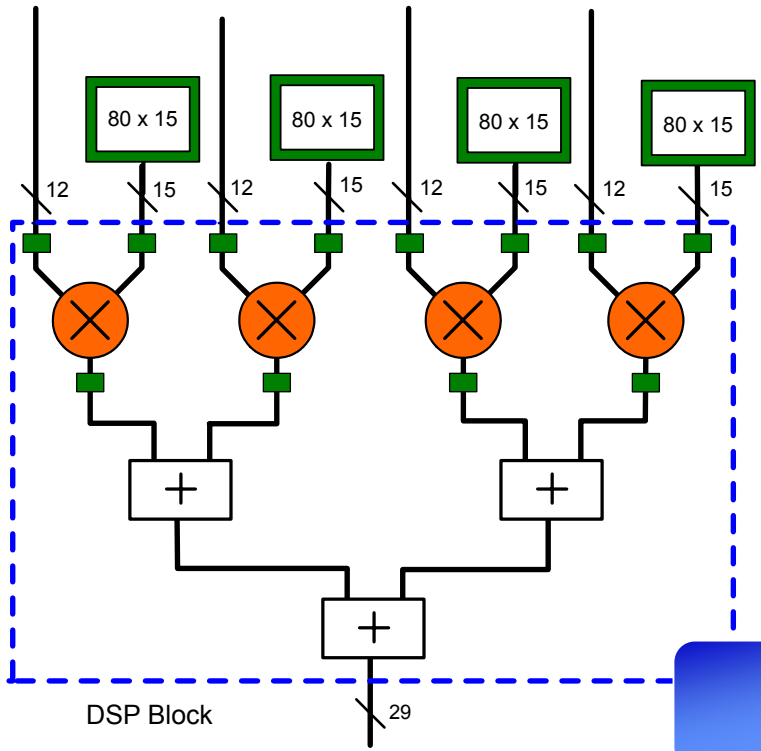
3 input adders greatly help to reduce the number of adders in the adder tree

External Fabric adders are always required to combine DSP Block results

Stratix-II
Multi-Channel Multi-Rate FIR

Filter Specification:

Input Frequency = 3.84 Mhz, Coefficients = 192, Interpolation Rate Change = 2, Channels = 8, Data Width = 12-bit, Coefficient Width = 15-bit



80 x 15 memories are required for the coefficients. Each bank of four memories can be addressed by the same counter as this is a Direct Form Type I structure and speed can still be met.

40% Slower Performance

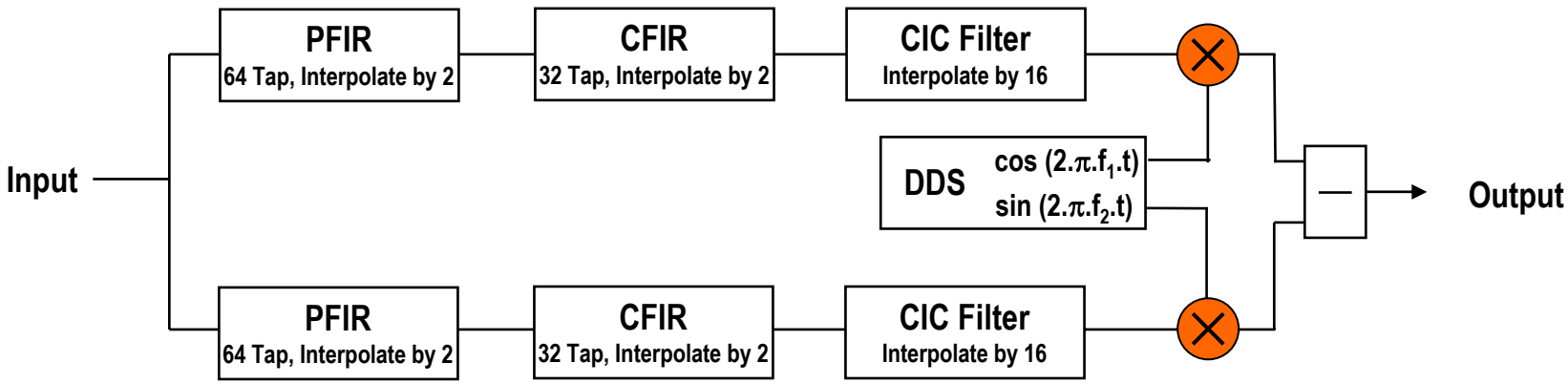
Performance
270 Mhz **(-4)**
Size:
5 DSP Blocks
(20 multipliers)
30 M4K Blocks
1271 ALMs

Case Study 1: DUC

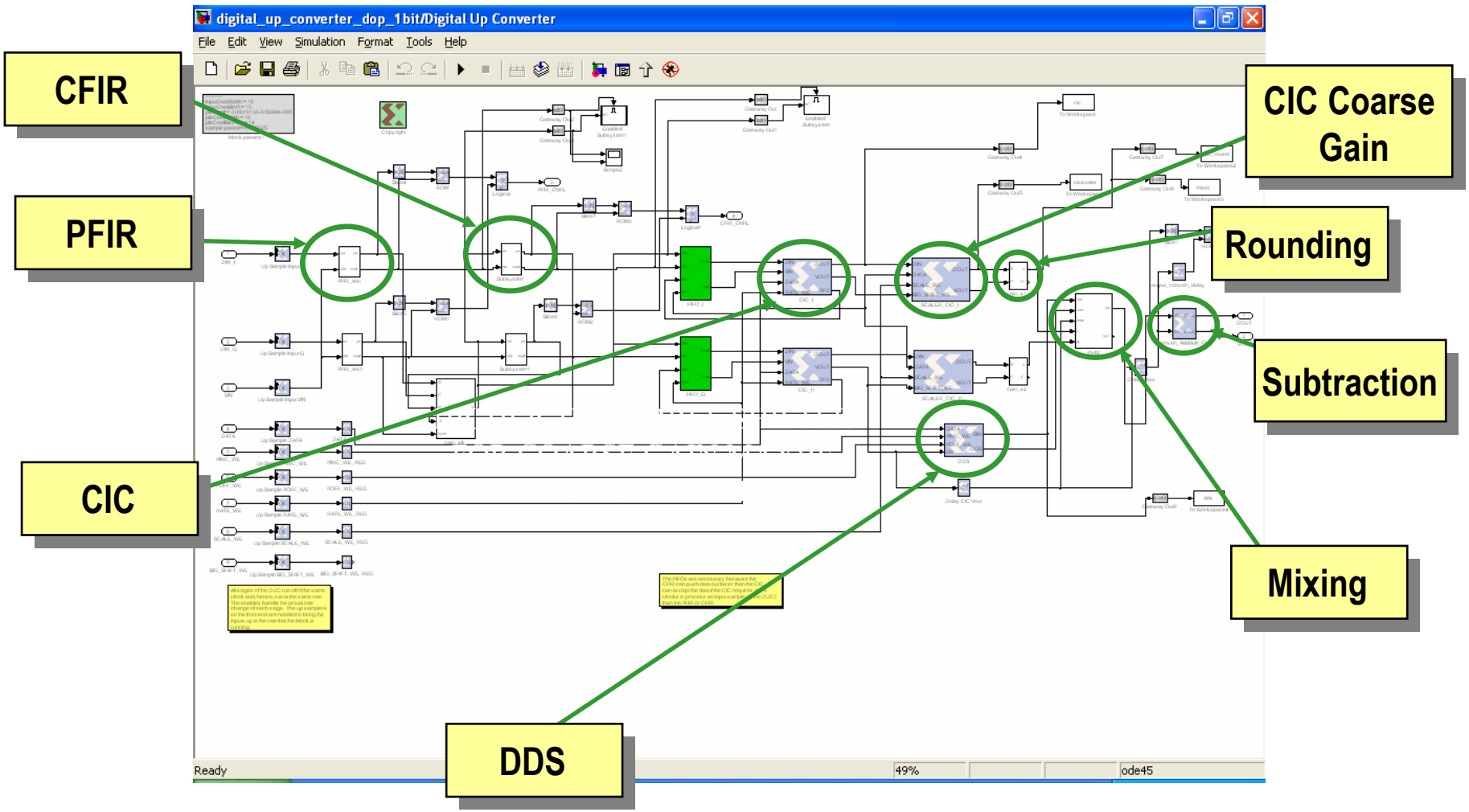
DUC Specification:

Output Frequency = 450 MSPS

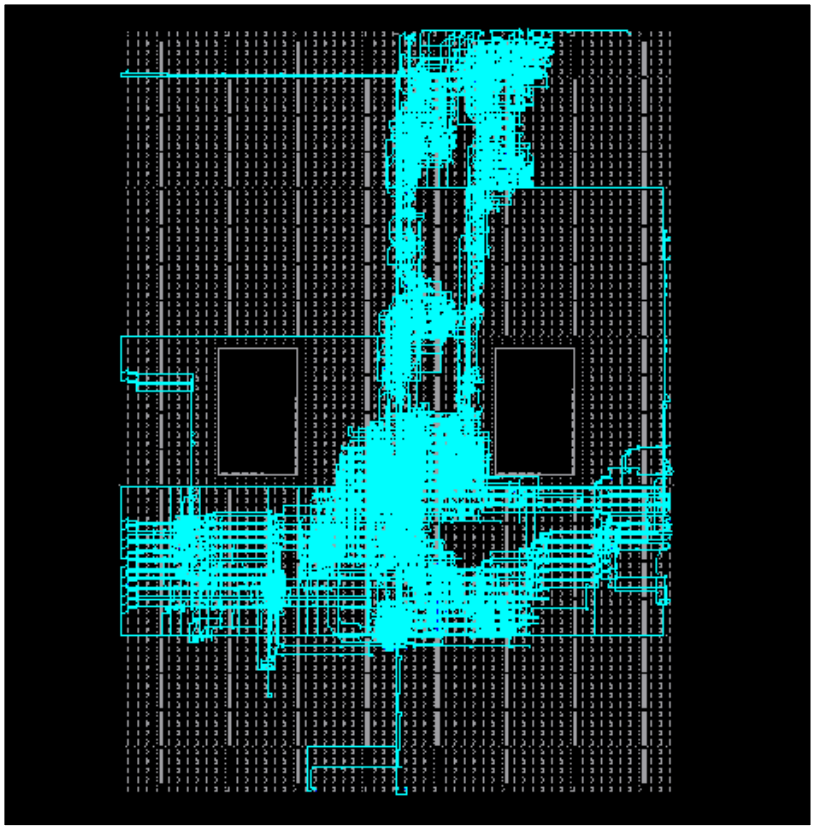
DDS: SFDR = 84dB, **CIC**: 5 Stage, Interpolation Rate = 1:16, **CFIR**: 32 Coefficients, Interpolation Rate = 1:2, **PFIR:** 64 Coefficients, Interpolation Rate 1:2



DUC: Sysgen Model



Case Study: DUC



DUC SIZE: (V-II Pro)
6 Embedded Mults
2,328 Flip-Flops
2,076 LUTs
10 Block RAM
Performance: 202 MHz

DUC SIZE: (V-4)
27 XDSP Slice
692 Flip-Flops
977 LUTs
10 Block RAM
Performance: >400 MHz

Case Study 2: 2-D FIR

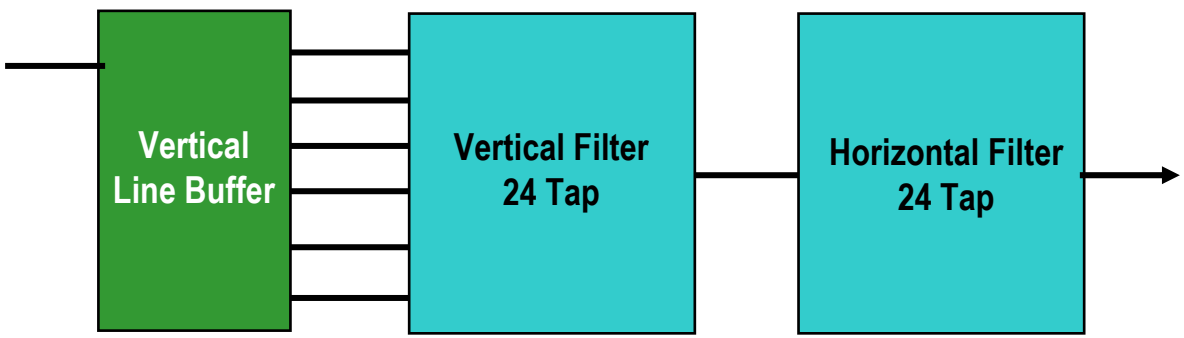
2-D FIR Specification:

Frame Rate = 60 Hz, Active Frame Size = 1440 x 1080, Single Channel

Separable FIRs : Sample Rate = 111.38 MSPS, 24 Tap Re-loadable, 10-bit Data, Folding Factor = 4

$$y(n,m) = \sum_{k=-N}^{k=+N} h_0(k) \left\{ \sum_{l=-N}^{l=+N} h_1(l).x(n-k, n-l) \right\}$$

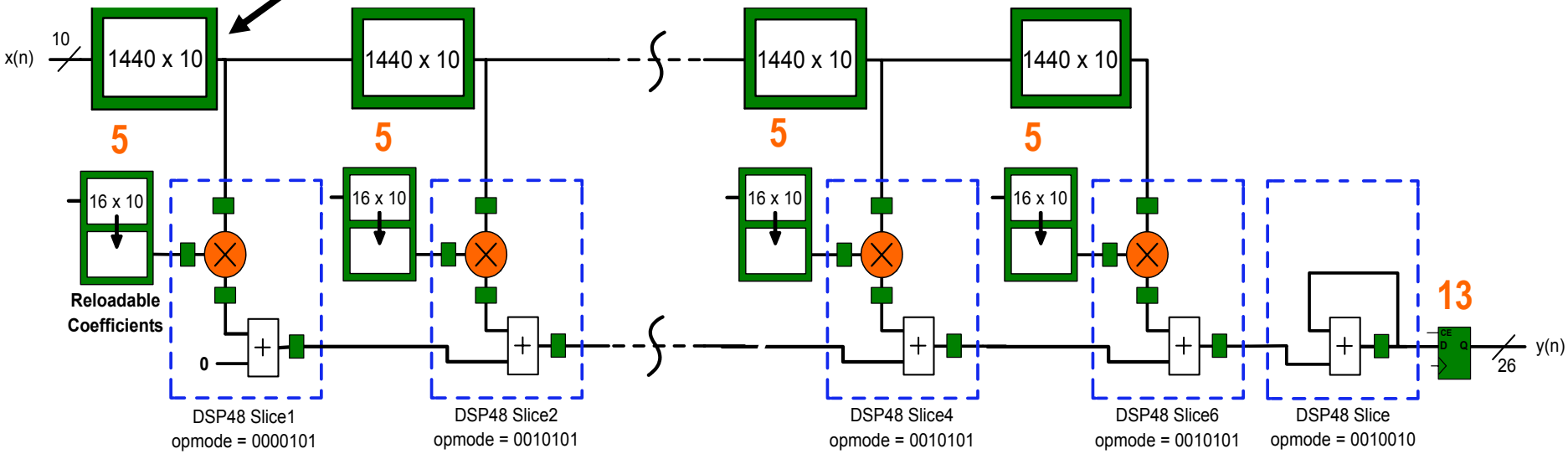
$$= \sum_{l=-N}^{l=+N} h_1(k) \left\{ \sum_{k=-N}^{k=+N} h_0(l).x(n-k, n-l) \right\}$$



Case Study 2: Vertical FIR & Line Buffer

Vertical Line Buffers are formed out of Single Port Block RAMs using a cyclic RAM buffer

Note: The Semi-Parallel Systolic Filter forms the foundation for both of the filters.



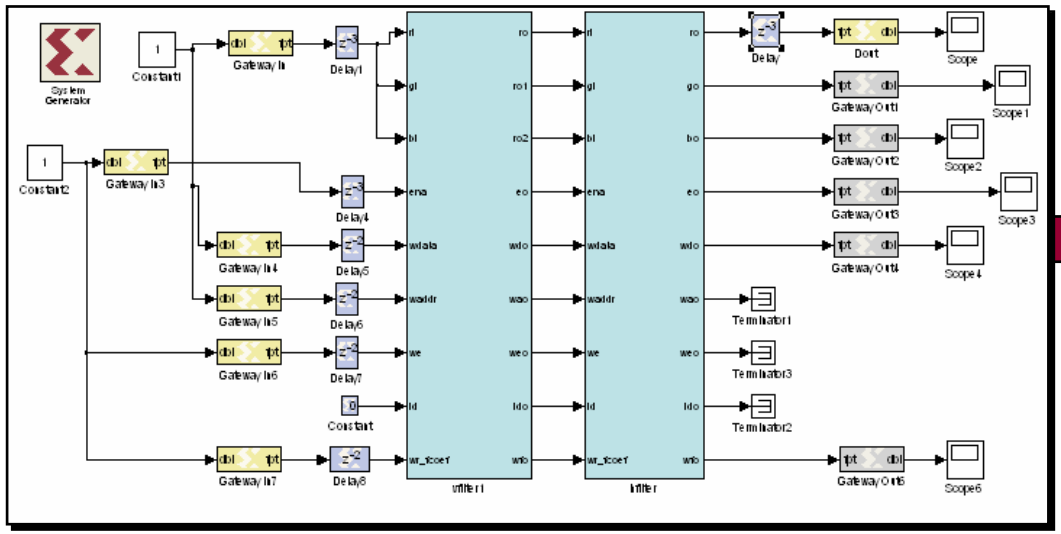
Vertical FIR Size:
7 XDSP Slice
43 Slices + Control
30 Block RAM

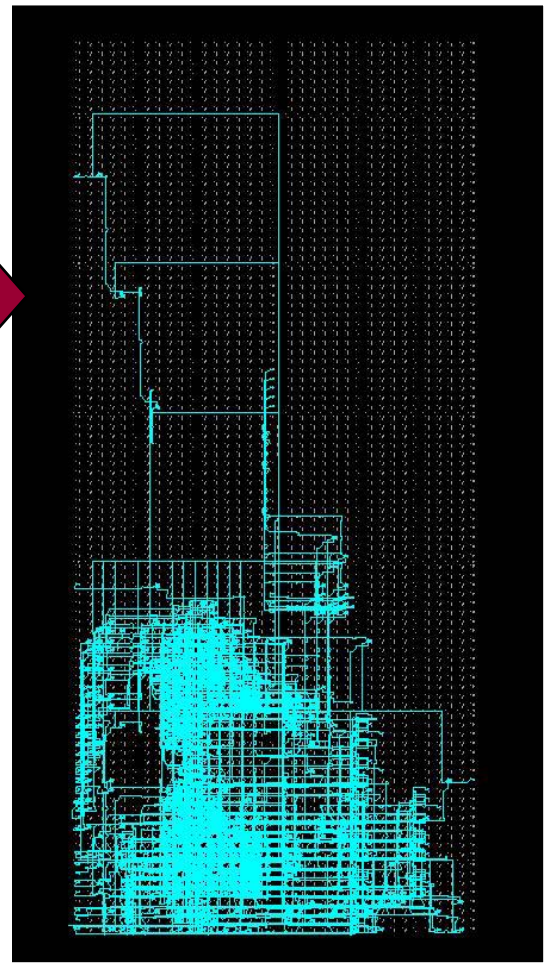
Re-loadable Coefficient memories created out of small Dual Port Distributed Memories

6 multipliers are required to process the data stream

$$\text{Number of Multipliers} = \frac{\text{Input Sample rate x Number of Taps}}{\text{Clock speed}} = \frac{111.38 \times 24}{450} = 5.94$$

Case Study: 2-D FIR





2-D FIR SIZE: (V-II Pro)
12 Embedded Mults
1,325 Flip-Flops
890 LUTs
30 Block RAM
Performance: 229.8 MHz

2-D FIR SIZE: (V-4)
15 XDSP Slice
560 Flip-Flops
414 LUTs
30 Block RAM
Performance: 446 MHz

Conclusion - The Check List

Is the design running as fast as possible?
(500 Mhz for fastest speed grade. 50% faster that Stratix-II. Resources can be saved by making sure the design runs at full speed.)

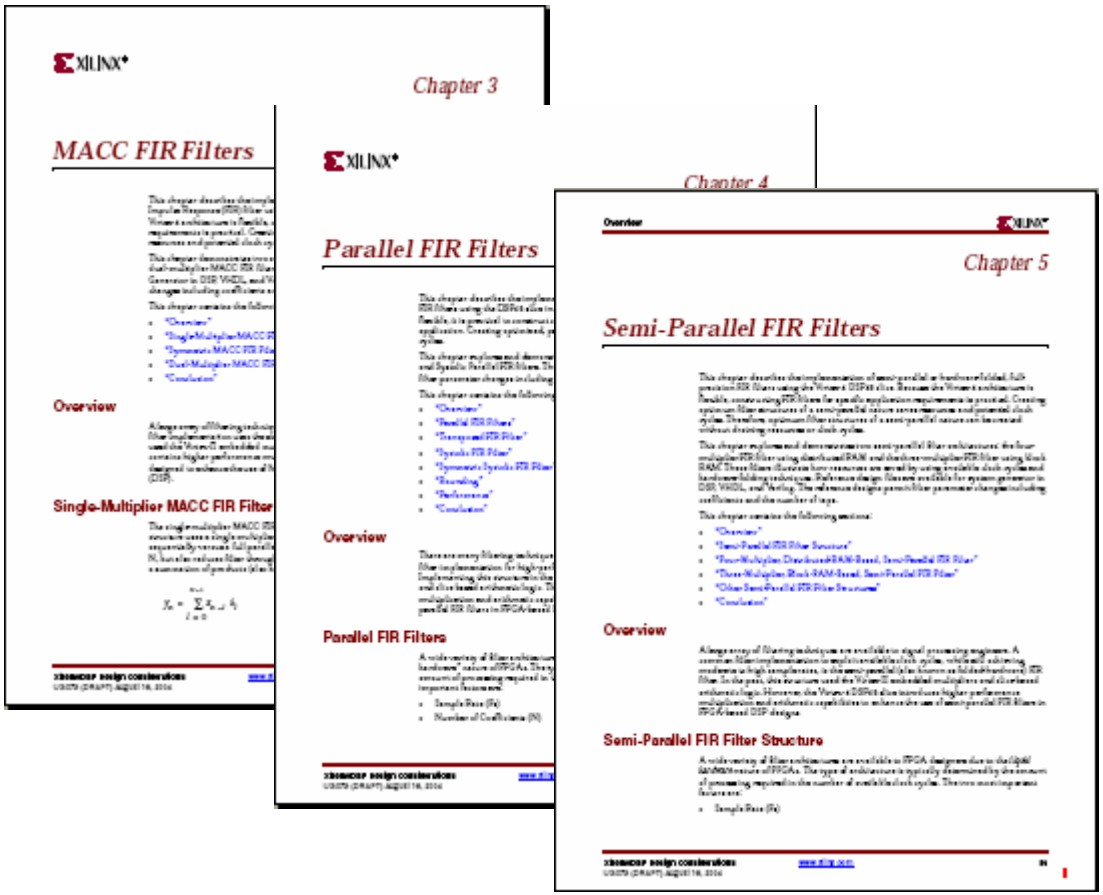
Is the XtremeDSP Slice being utilized fully?
(Fabric slices can be saved by better exploiting the Xtreme DSP Slice which leads to less power. Greater than 1 Watt less power that Stratix-II)

Are Adder Chains being used instead of trees?
(The XtremeDSP Slice is designed to support adder cascades)

Knowledge is Power

"The next best thing to knowing something is knowing where to find it"

- Samuel Johnson



5 Application Notes available in the **Virtex-4 User Guide** in regard to implementation specifics

Many Reference Designs in:

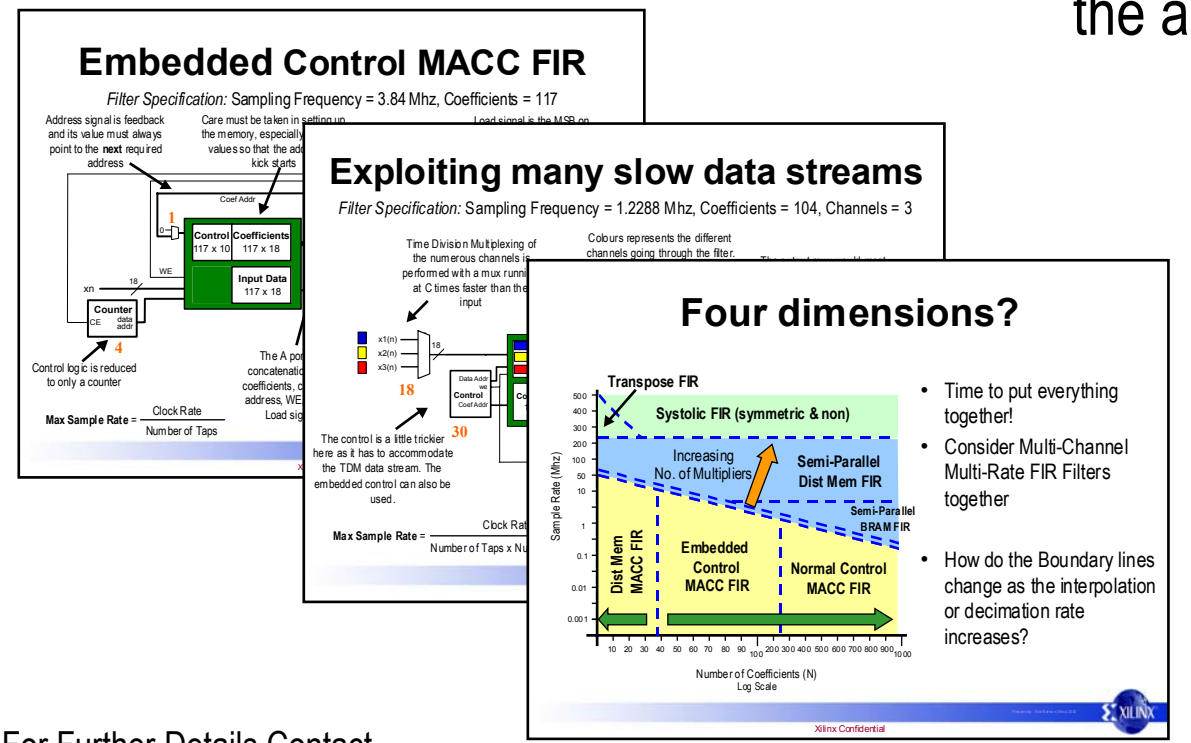
- **VHDL**
- **Verilog**
- **System Generator for DSP**

For Further Details visit…..

www.xilinx.com/dsp

Knowledge is Power

"The desire of knowledge, like the thirst of riches, increases ever with the acquisition of it." - *Sterne*



For Further Details Contact…..

MySupport.xilinx.com

Over 500 Pages

2 DSP Courses:

- DSP Implementation Techniques (3 day)

- DSP Design Flow (3 day)

To educate students on efficient DSP design in Xilinx FPGAs using the latest system level design tools