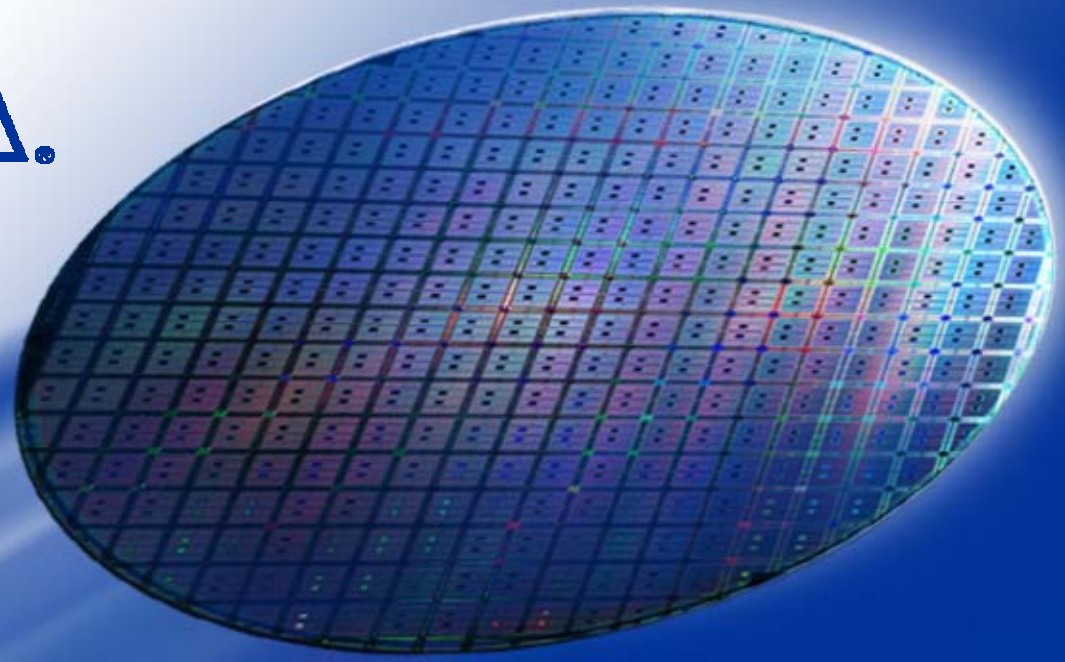


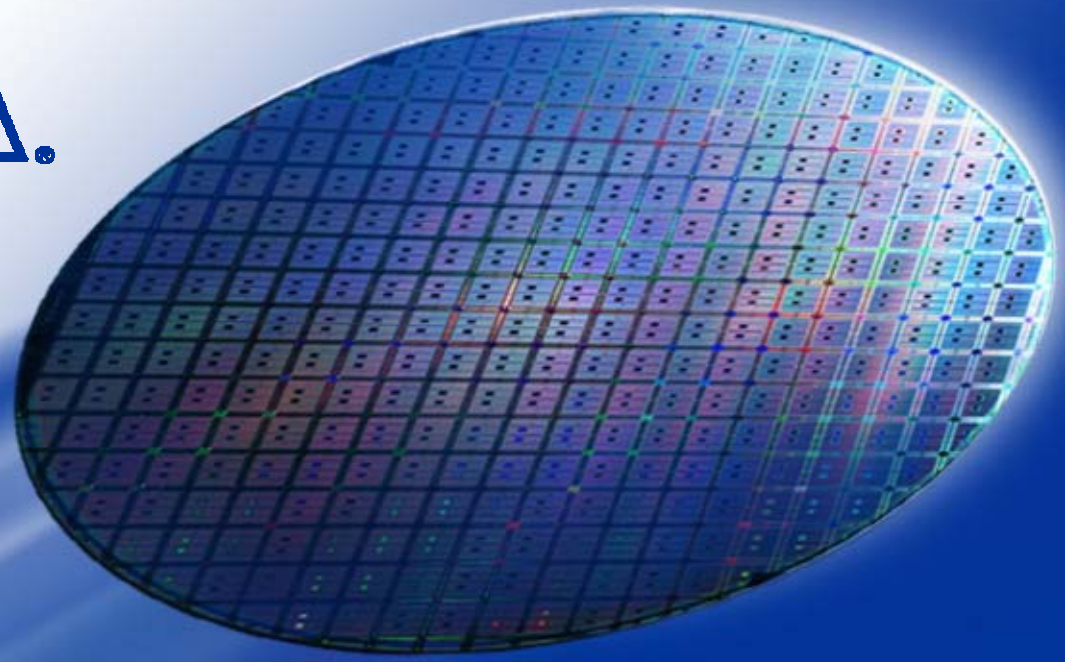
Optimizing a High Performance 32-bit Processor for Programmable Logic

Paul Metzgen

16th November 2004

Agenda

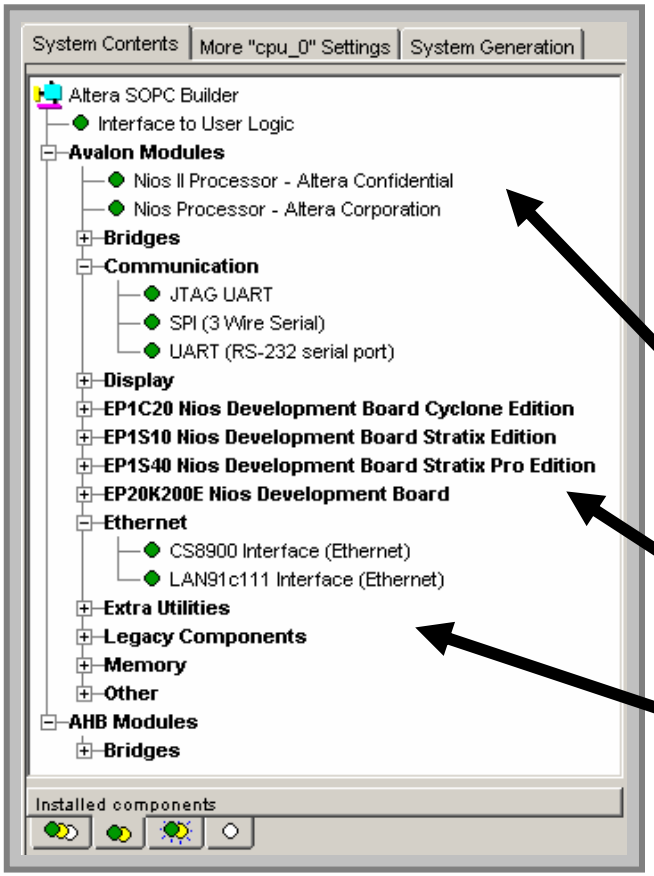
- System Design on FPGAs
  - Brief Overview of Altera's SOPC Tools
- Architecting Designs for FPGAs
  - Different Design Trade-offs
- Case Study: The Design of Nios II
  - Implementing Multiplexers in FPGAs
  - Optimizing Multiplexers in Nios II



System Design on FPGAs

Overview of Altera's SOPC Toolflow

Altera's SOPC Builder



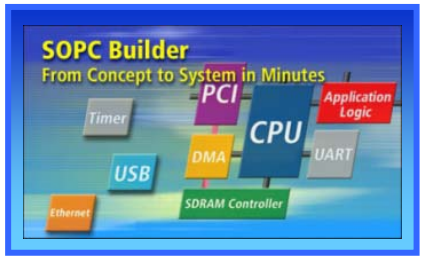
Peripheral Set

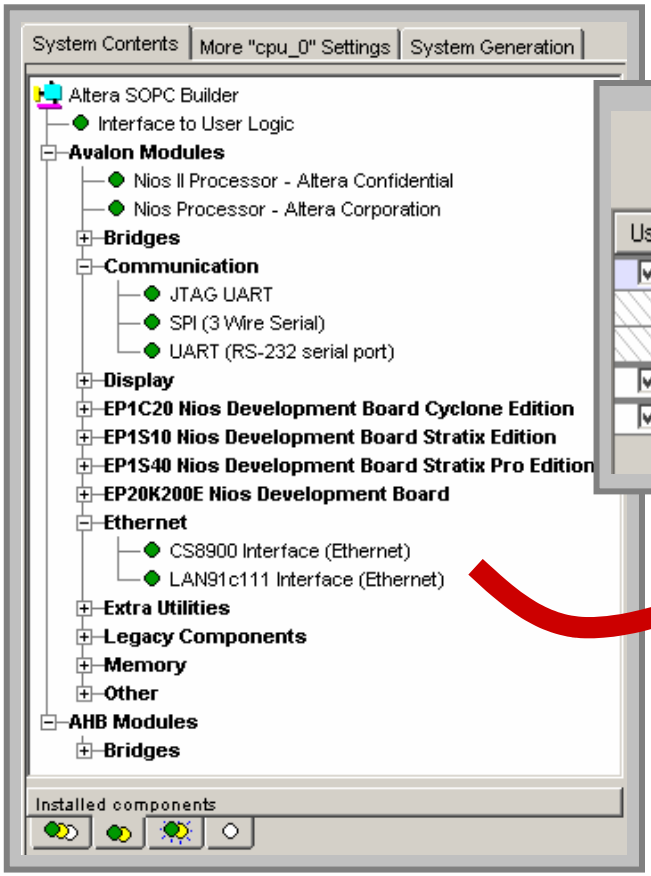
Can also add your own (eg:

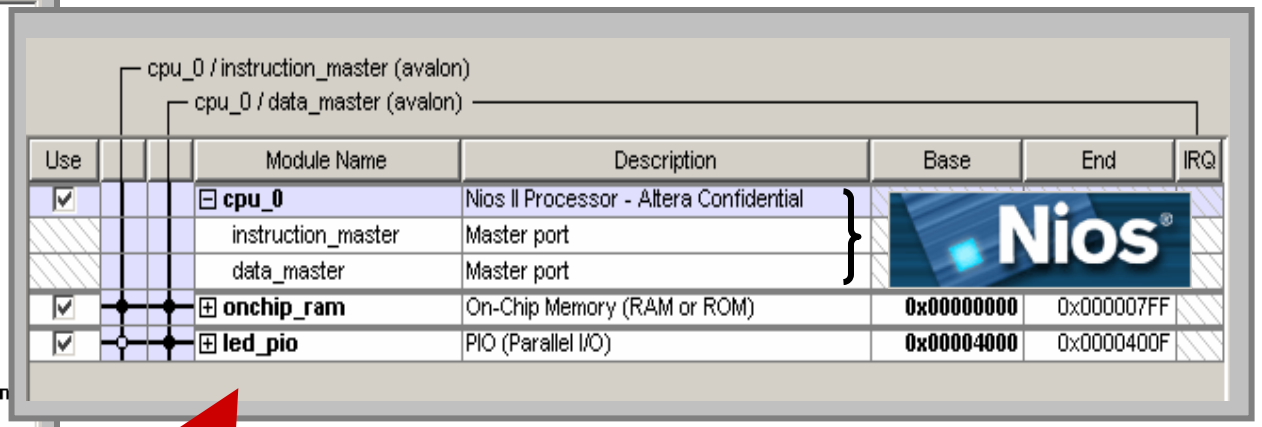
– custom peripherals,

– accelerators)

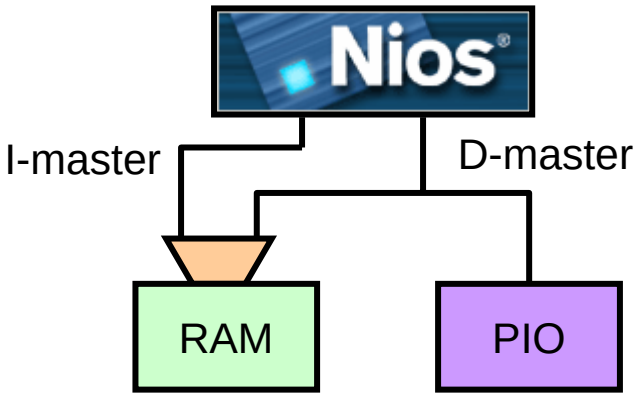
Altera's SOPC Builder



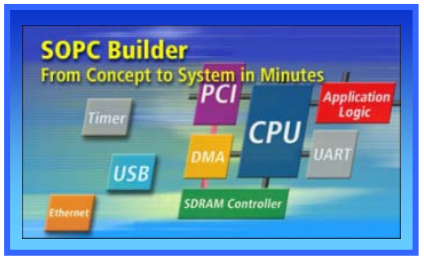




Can specify system connectivity



Altera's SOPC Builder



System Contents | More "cpu_0" Settings | System Generation

- Altera SOPC Builder
  - Interface to User Logic
- **Avalon Modules**
  - Nios II Processor - Altera Confidential
  - Nios Processor - Altera Corporation
  - **Bridges**
  - **Communication**
    - JTAG UART
    - SPI (3 Wire Serial)
    - UART (RS-232 serial port)
  - **Display**
  - **EP1C20 Nios Development Board Cyclone Edition**
  - **EP1S10 Nios Development Board Stratix Edition**
  - **EP1S40 Nios Development Board Stratix Pro Edition**
  - **EP20K200E Nios Development Board**
  - **Ethernet**
    - CS8900 Interface (Ethernet)
    - LAN91c111 Interface (Ethernet)
  - **Extra Utilities**
  - **Legacy Components**
  - **Memory**
  - **Other**
- **AHB Modules**
  - **Bridges**

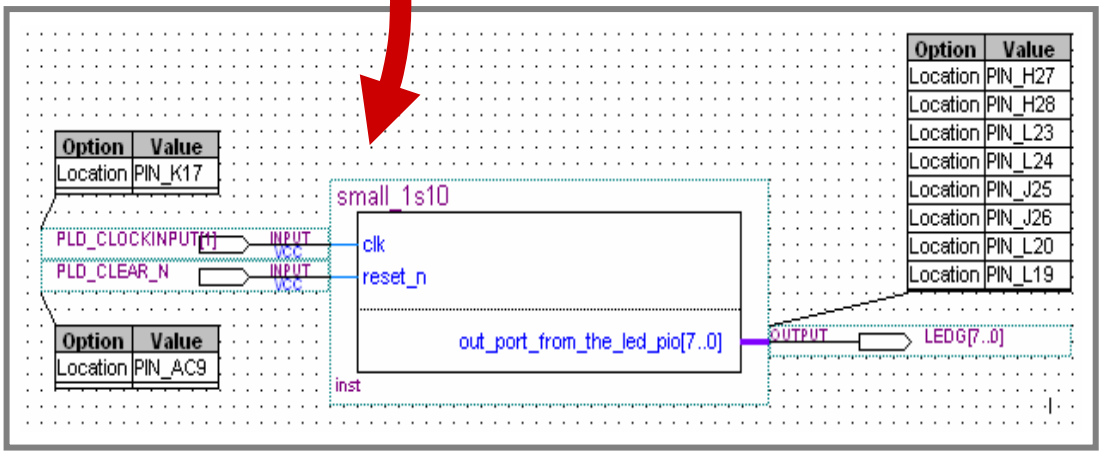
Installed components

cpu_0 / instruction_master (avalon)
cpu_0 / data_master (avalon)

Use	Module Name	Description	Base	End	IRQ
☑	**cpu_0**	Nios II Processor - Altera Confidential			
	instruction_master	Master port			
	data_master	Master port			
☑	**onchip_ram**	On-Chip Memory (RAM or ROM)	**0x00000000**	0x000007FF	
☑	**led_pio**	PIO (Parallel I/O)	**0x00004000**	0x0000400F	

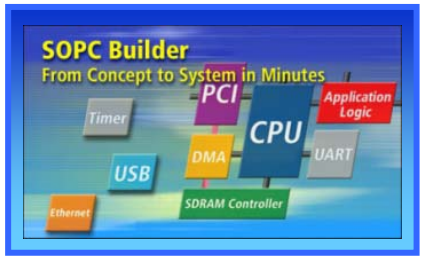
Nios®

Automatic Logic & Bus Generation



ALTERA®

Altera's SOPC Builder



System Contents | More "cpu_0" Settings | System Generation

- Altera SOPC Builder
  - Interface to User Logic
  - **Avalon Modules**
    - Nios II Processor - Altera Confidential
    - Nios Processor - Altera Corporation
    - **Bridges**
    - **Communication**
      - JTAG UART
      - SPI (3 Wire Serial)
      - UART (RS-232 serial port)
    - **Display**
    - **EP1C20 Nios Development Board Cyclone Edition**
    - **EP1S10 Nios Development Board Stratix Edition**
    - **EP1S40 Nios Development Board Stratix Pro Edition**
    - **EP20K200E Nios Development Board**
    - **Ethernet**
      - CS8900 Interface (Ethernet)
      - LAN91c111 Interface (Ethernet)
    - **Extra Utilities**
    - **Legacy Components**
    - **Memory**
    - **Other**
  - **AHB Modules**
    - **Bridges**

Installed components

cpu_0 / instruction_master (avalon)
cpu_0 / data_master (avalon)

Use	Module Name	Description	Base	End	IRQ
☑	**cpu_0**	Nios II Processor - Altera Confidential			
	instruction_master	Master port			
	data_master	Master port			
☑	**onchip_ram**	On-Chip Memory (RAM or ROM)	**0x00000000**	0x000007FF	
☑	**led_pio**	PIO (Parallel I/O)	**0x00004000**	0x0000400F	

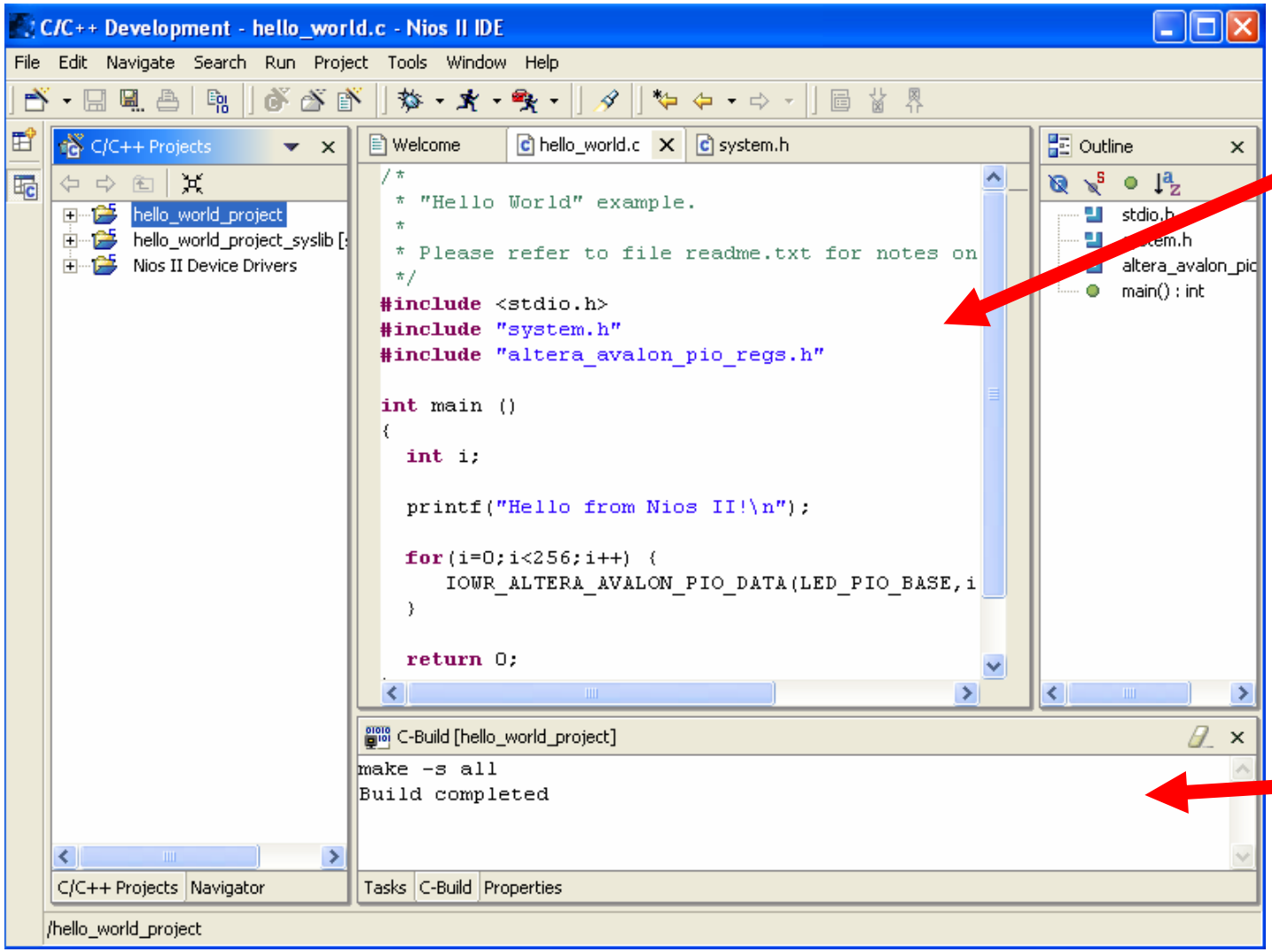
Automatic Device Driver Generation

system.h

```
#define ONCHIP_MEMOR_RAM_NAME "/dev/onchip_memory_ram"
#define ONCHIP_MEMORY_RAM_TYPE "altera_avalon_onchip_memory2"
#define ONCHIP_MEMORY_RAM_BASE 0x00010000
#define ONCHIP_MEMORY_RAM_ALLOW_MRAM_SIM_CONTENTS_ONLY_FILE 0
#define ONCHIP_MEMORY_RAM_RAM_BLOCK_TYPE "M4K"
#define ONCHIP_MEMORY_RAM_GUI_RAM_BLOCK_TYPE "Automatic"
#define ONCHIP_MEMORY_RAM_WRITEABLE 1
#define ONCHIP_MEMORY_RAM_DUAL_PORT 0
#define ONCHIP_MEMORY_RAM_SIZE_VALUE 512
#define ONCHIP_MEMORY_RAM_SIZE_MULTIPLE 1
#define ONCHIP_MEMORY_RAM_CONTENTS_INFO ""
```

Nios® II IDE INTEGRATED DEVELOPMENT ENVIRONMENT

Nios II IDE

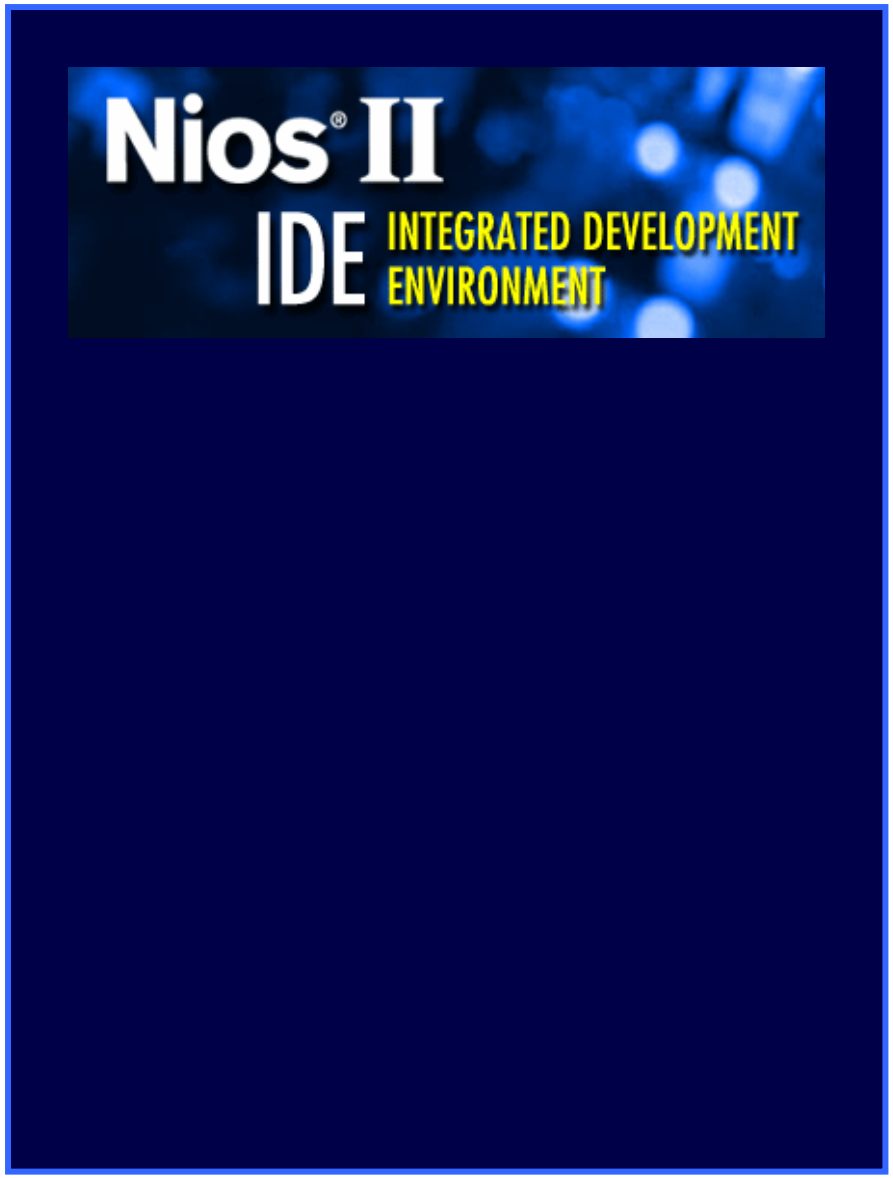


File Viewer Window

Terminal window

ALTERA®

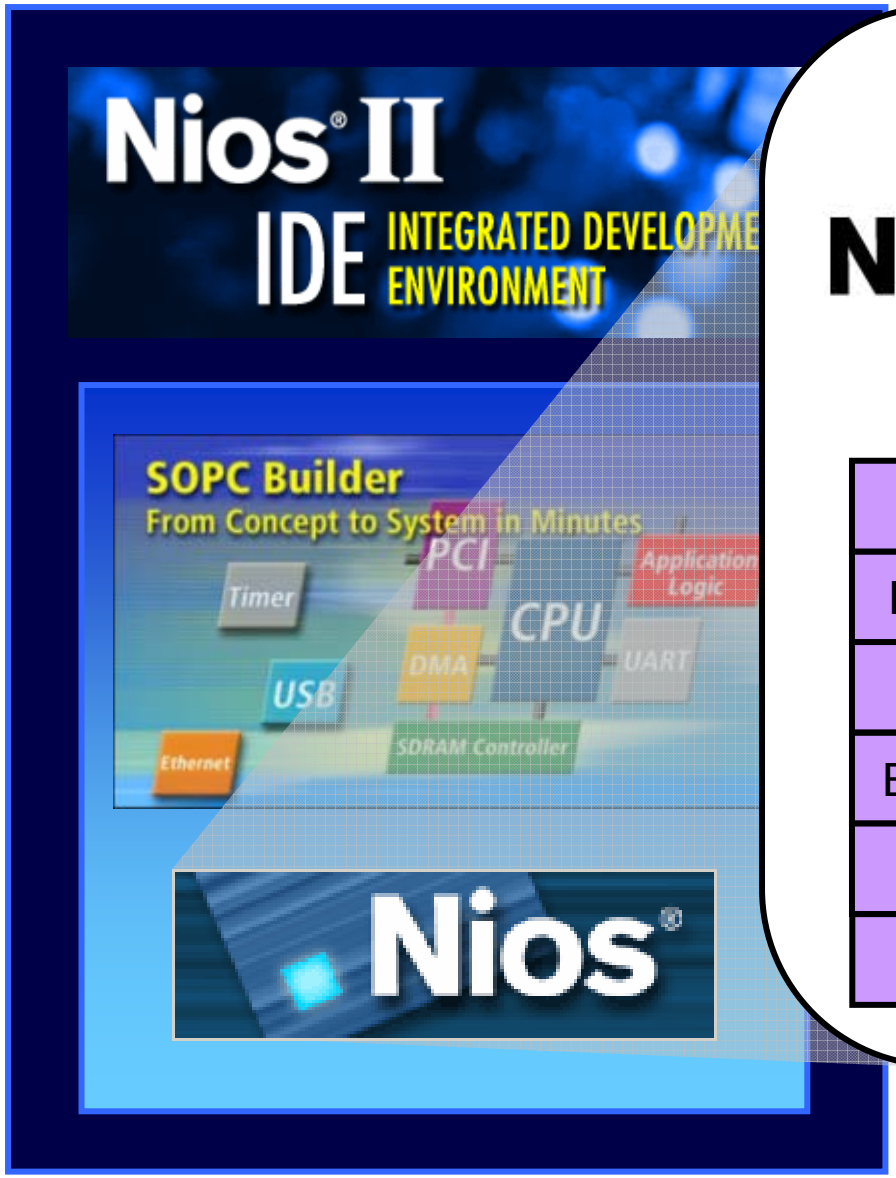
SOPC Toolflow: Summary



SOPC Toolflow: Summary

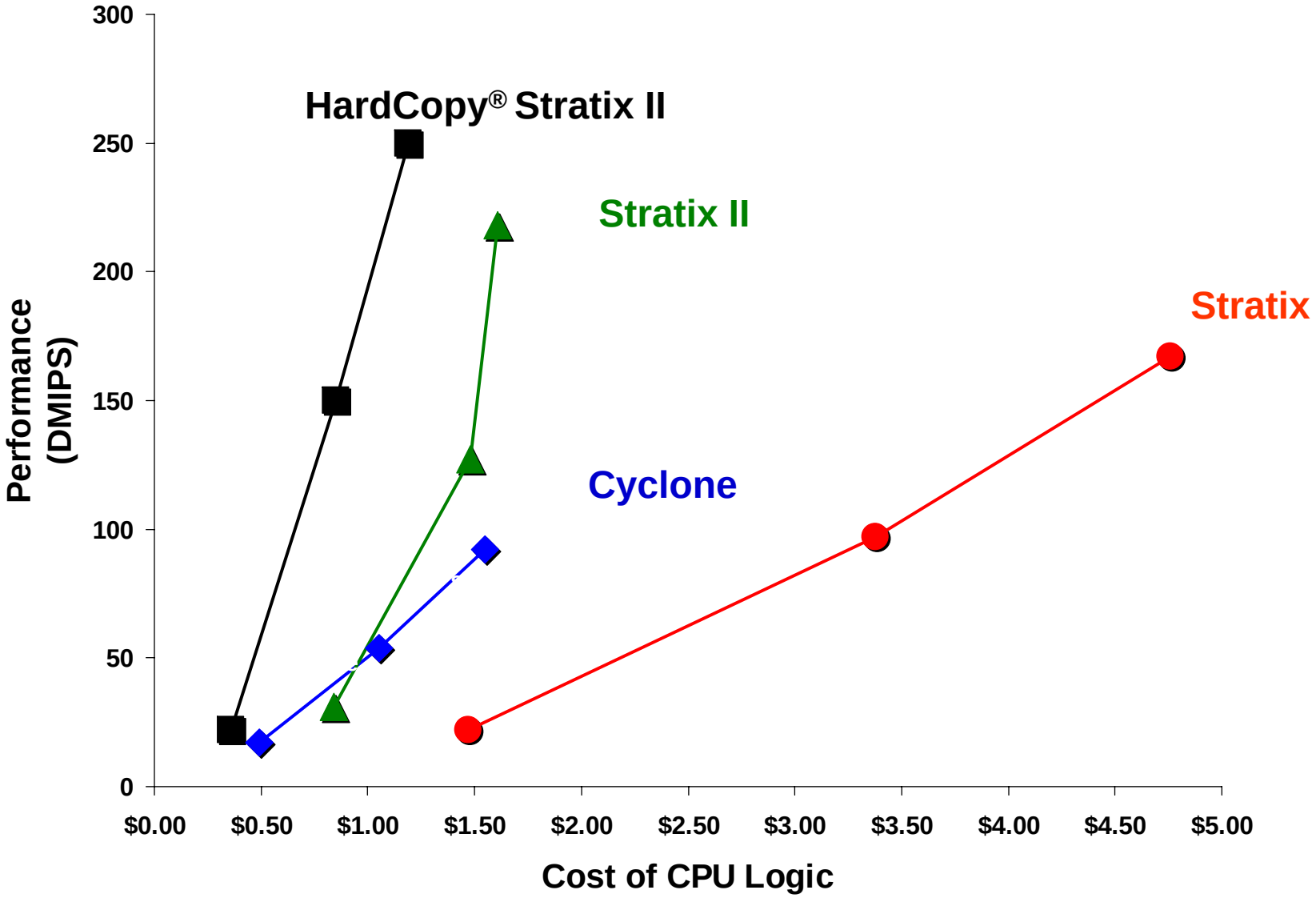
SOPC Toolflow: Summary

Nios II Family of Processors:

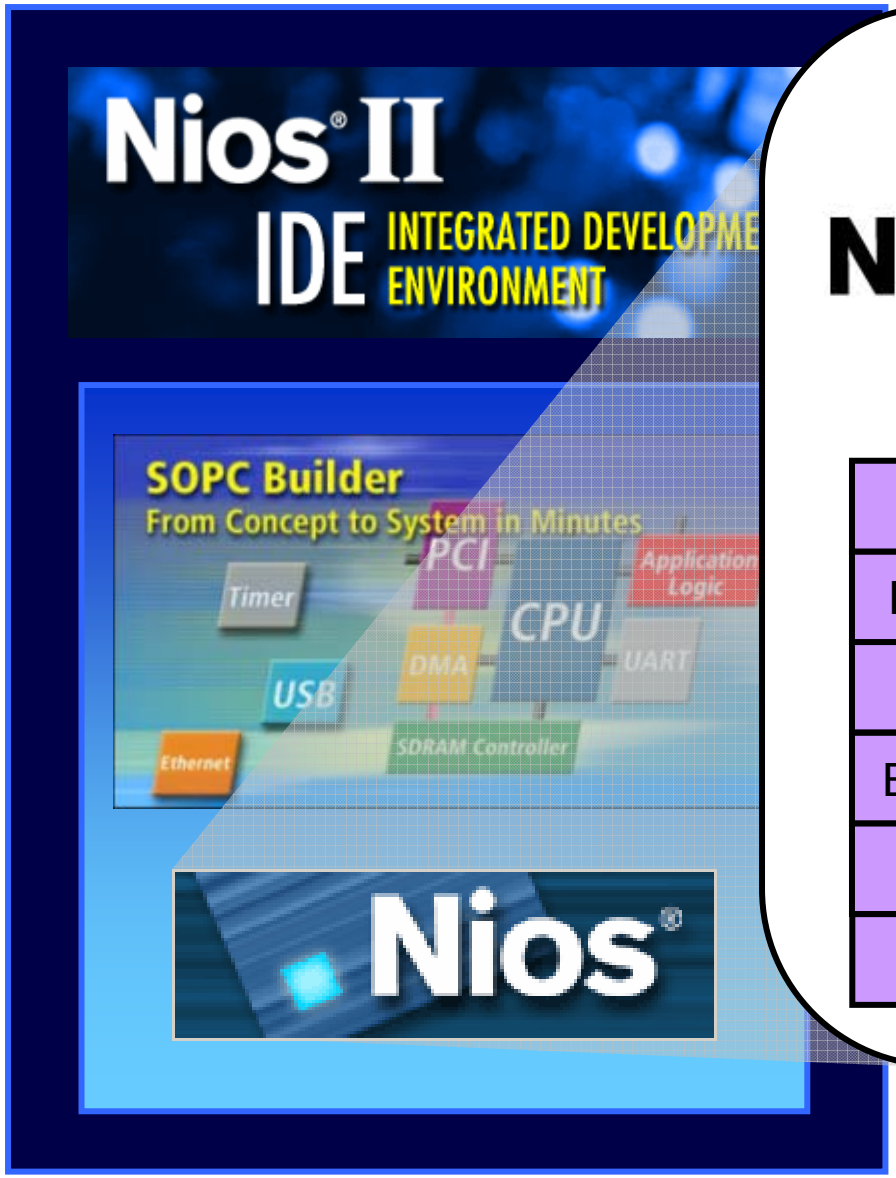


Nios® II	*Fast*	Standard	Economy
Size (LEs)	**1800**	**1400**	**700**
Performance	**7.5x**	**4.7x**	**1.0x**
Pipeline	6-stage	5-stage	5-cycle
Br. Prediction	Dynamic	Static	no
I$ - Cache	**yes**	**yes**	no
D$ - Cache	**yes**	no	no

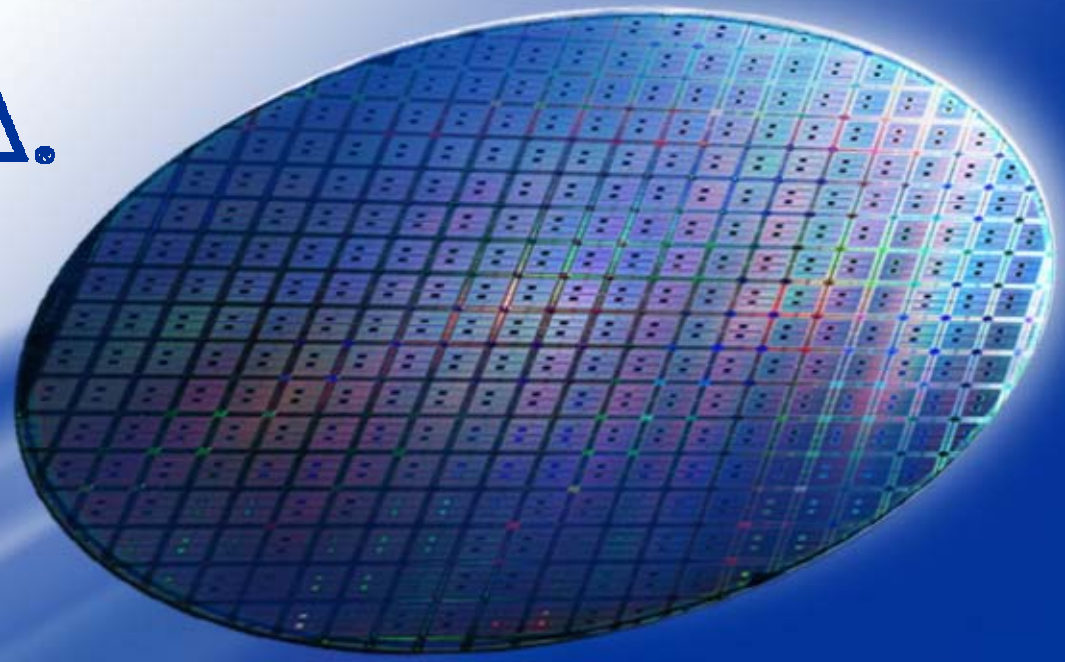
Processor Cost vs. Performance



Nios II Family of Processors:



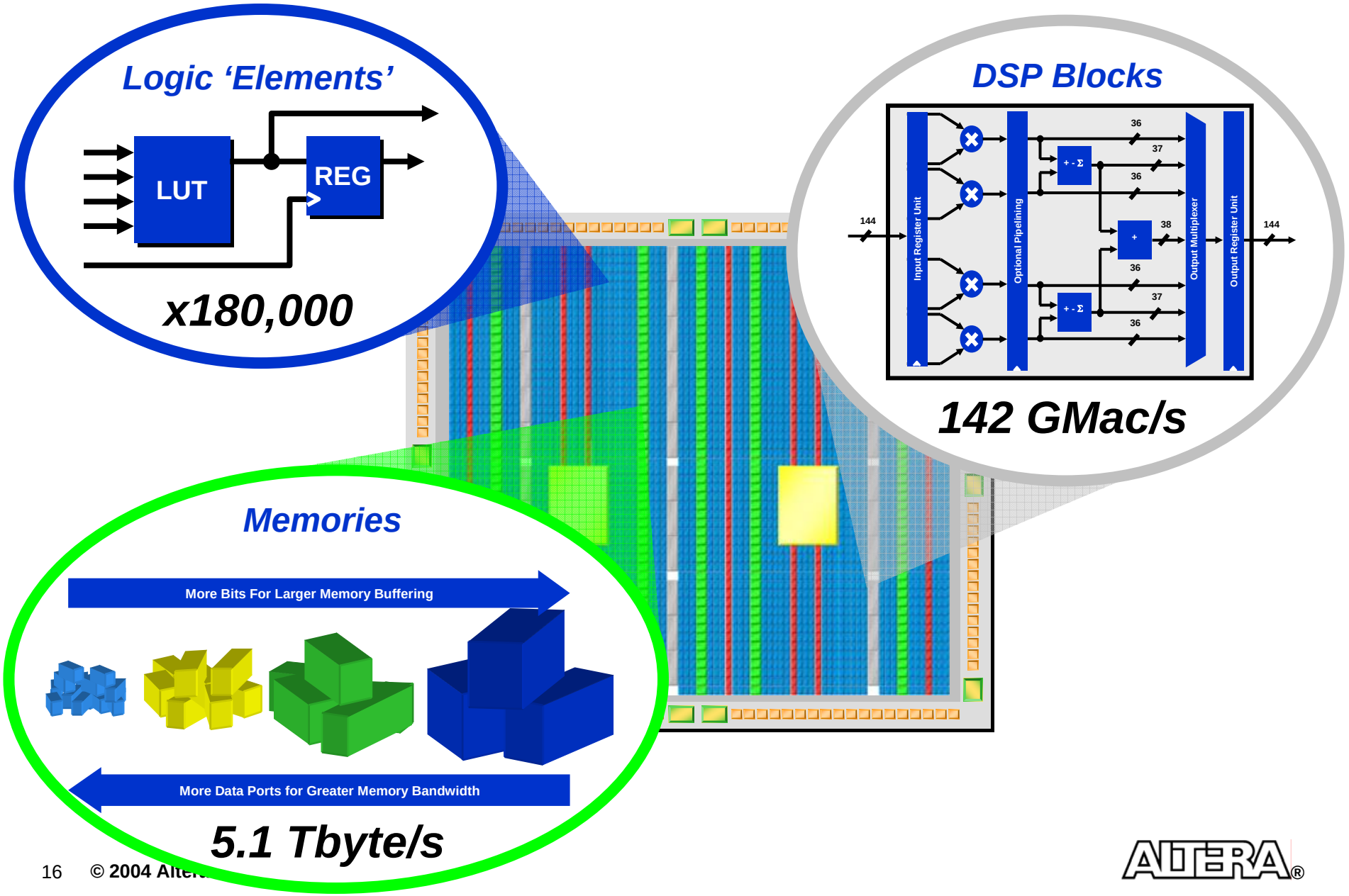
Nios® II	*Fast*	Standard	Economy
Size (LEs)	**1800**	**1400**	**700**
Performance	**7.5x**	**4.7x**	**1.0x**
Pipeline	6-stage	5-stage	5-cycle
Br. Prediction	Dynamic	Static	no
I$ - Cache	**yes**	**yes**	no
D$ - Cache	**yes**	no	no



Architecting Designs for FPGAs

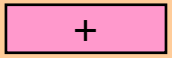
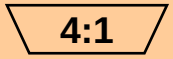
Different Design Trade-offs

Making the most of the Available Resources



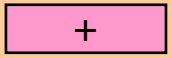
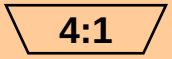
Relative Area Costs

Area Cost

	ASIC	FPGA
Memory 	High	
Multipliers 	High	
Registers 	Medium	
Adders 	Medium	
Multiplexers 	Low	

Relative Area Costs

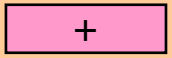
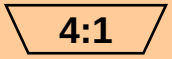
Area Cost

	ASIC	FPGA
Memory 	High	
Multipliers 	High	
Registers 	Medium	Low
Adders 	Medium	Low
Multiplexers 	Low	

} **Free Register with every Lookup Table (independently accessible)**

Relative Area Costs

Area Cost

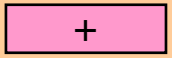
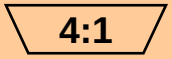
	ASIC	FPGA
Memory 	High	Medium
Multipliers 	High	Medium
Registers 	Medium	Low
Adders 	Medium	Low
Multiplexers 	Low	

‘Hard’ Optimized ASIC Blocks

Free Register with every Lookup Table (independently accessible)

Relative Area Costs

Area Cost

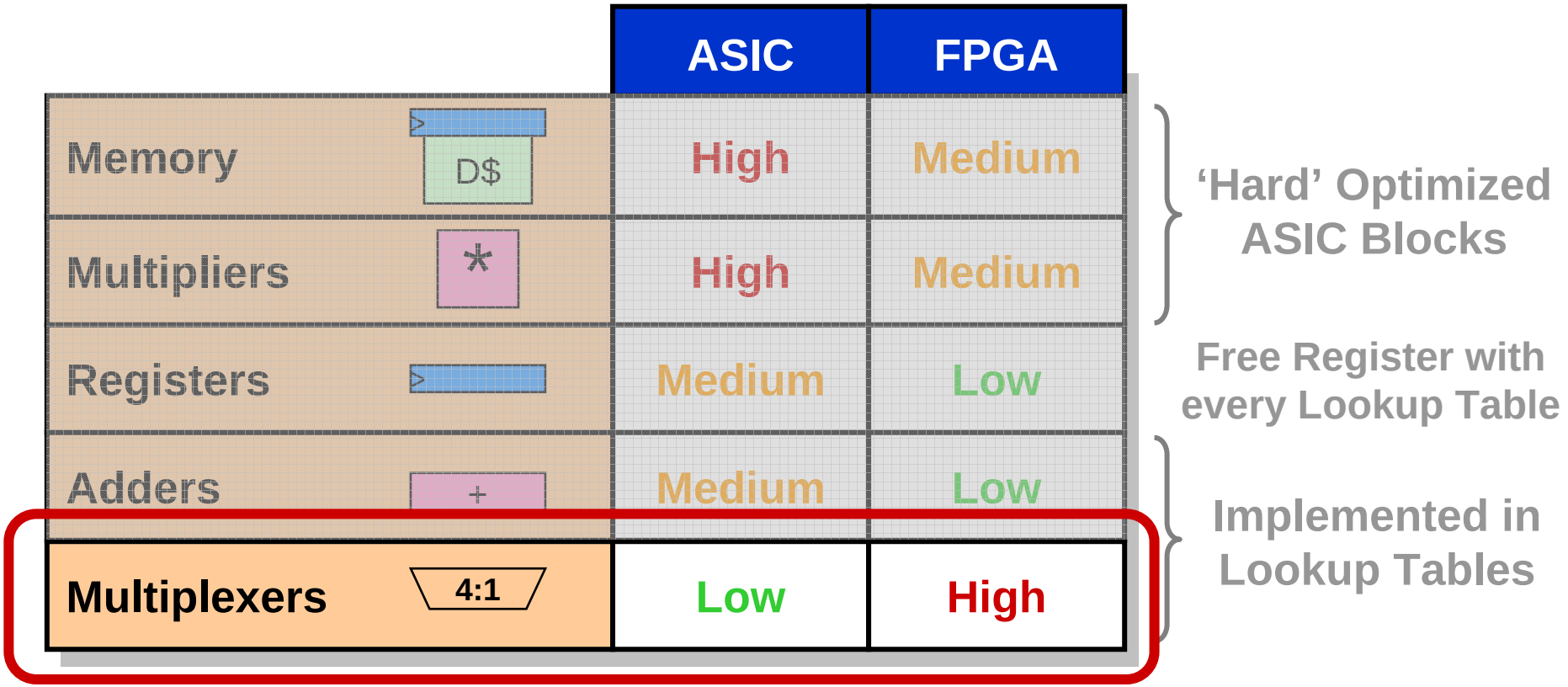
	ASIC	FPGA
Memory 	High	Medium
Multipliers 	High	Medium
Registers 	Medium	Low
Adders 	Medium	Low
Multiplexers 	Low	High

'Hard' Optimized ASIC Blocks

Free Register with every Lookup Table (independently accessible)

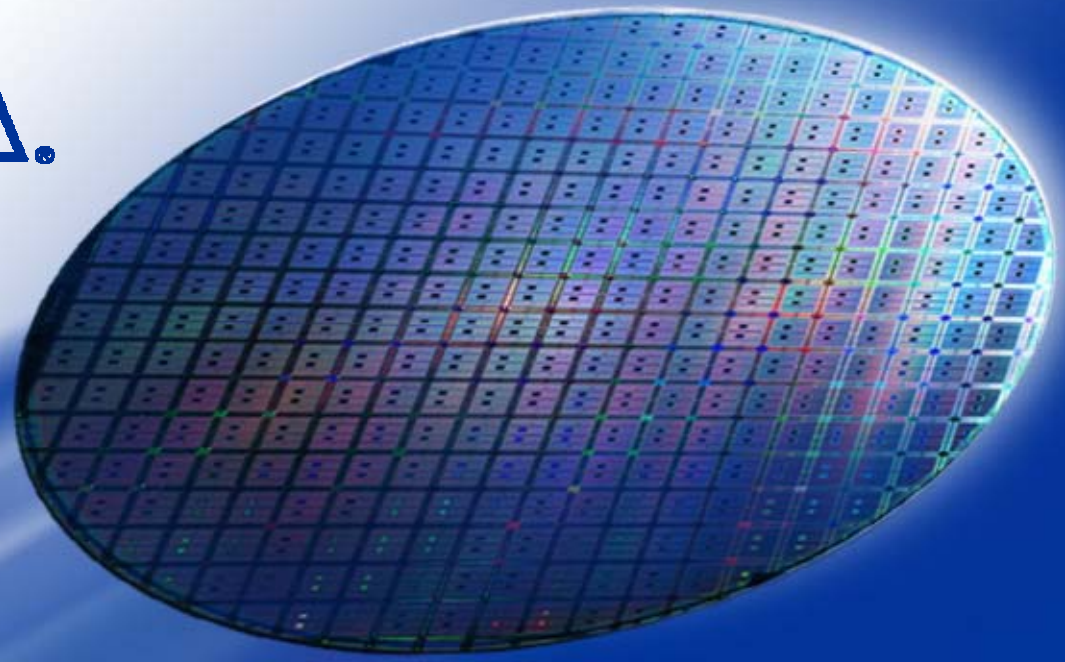
Implemented in Lookup Tables

Relative Area Costs



"The Key to Optimizing Designs for an FPGA

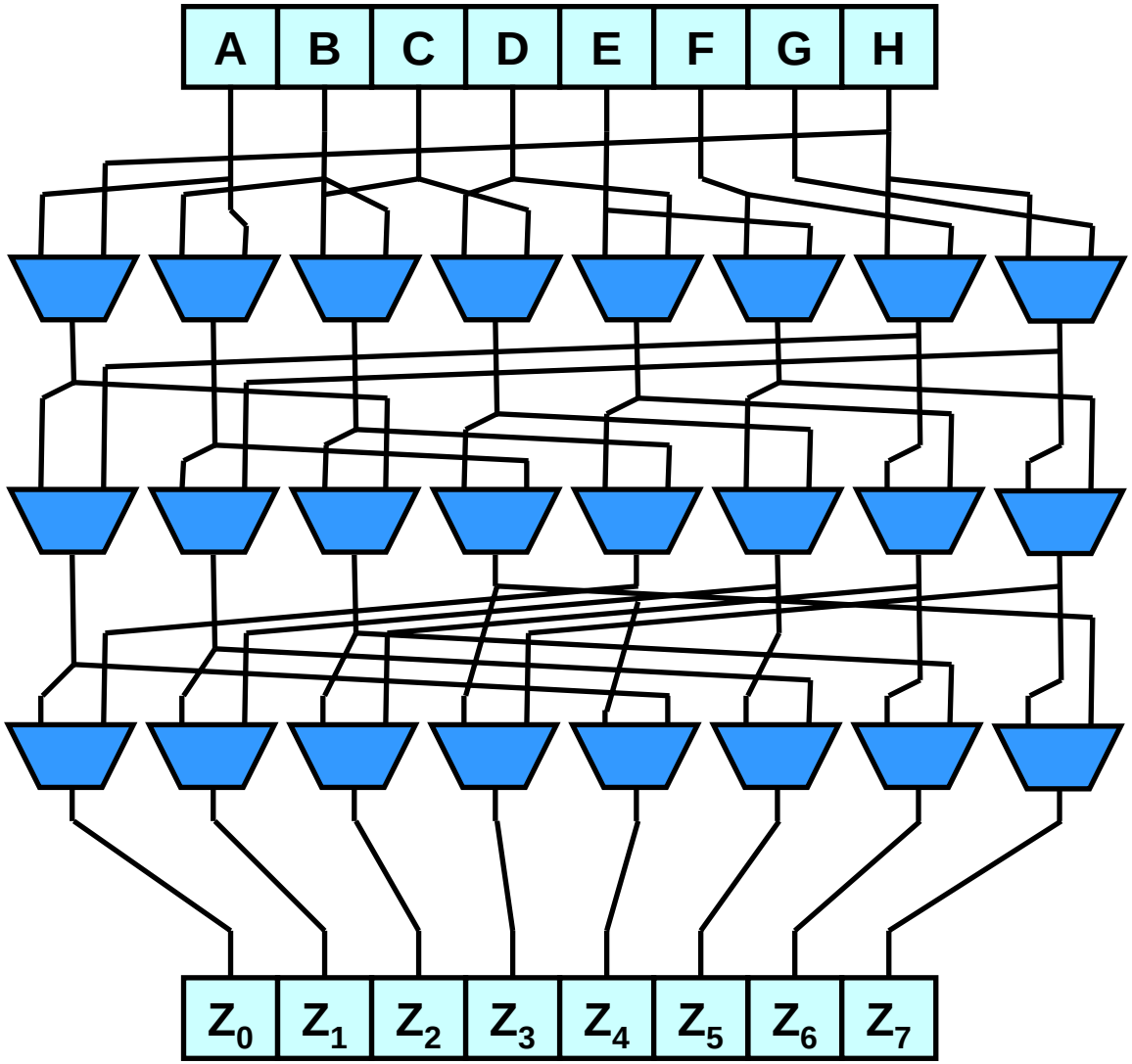
…is to Optimize the Multiplexers"



Architecting Designs for FPGAs

Barrel-Shifts using Multipliers

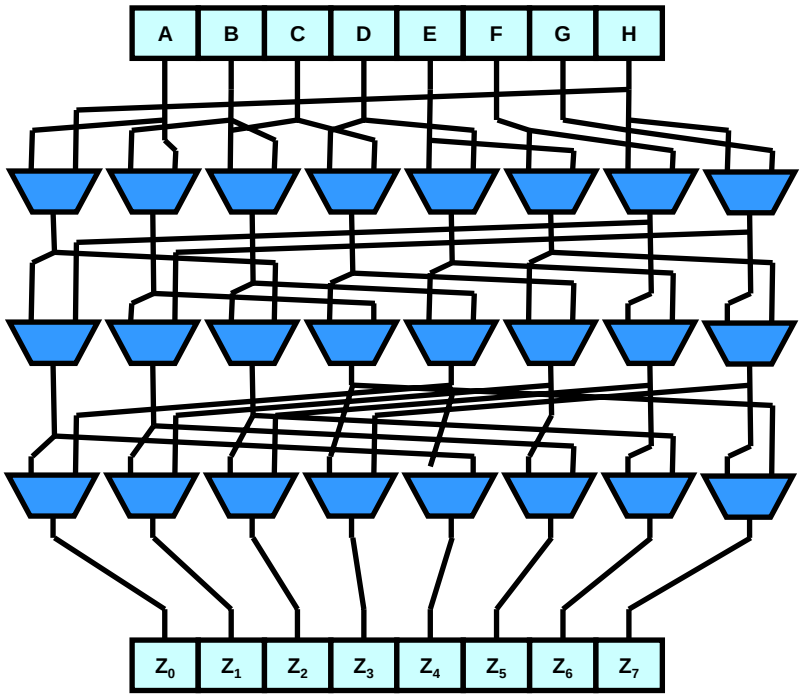
A Barrel-Shifter Using Multiplexers

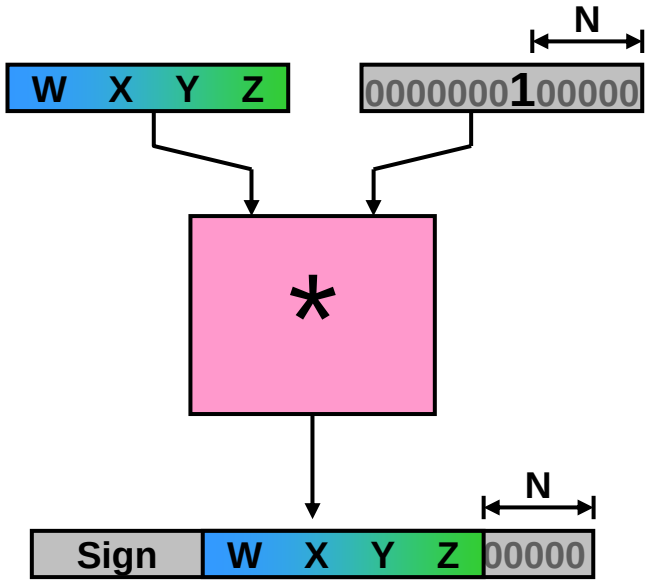


$N \log_2 N$ LEs

160 LEs
for a 32-bit Barrel Shifter

Barrel Shifter Using Multipliers

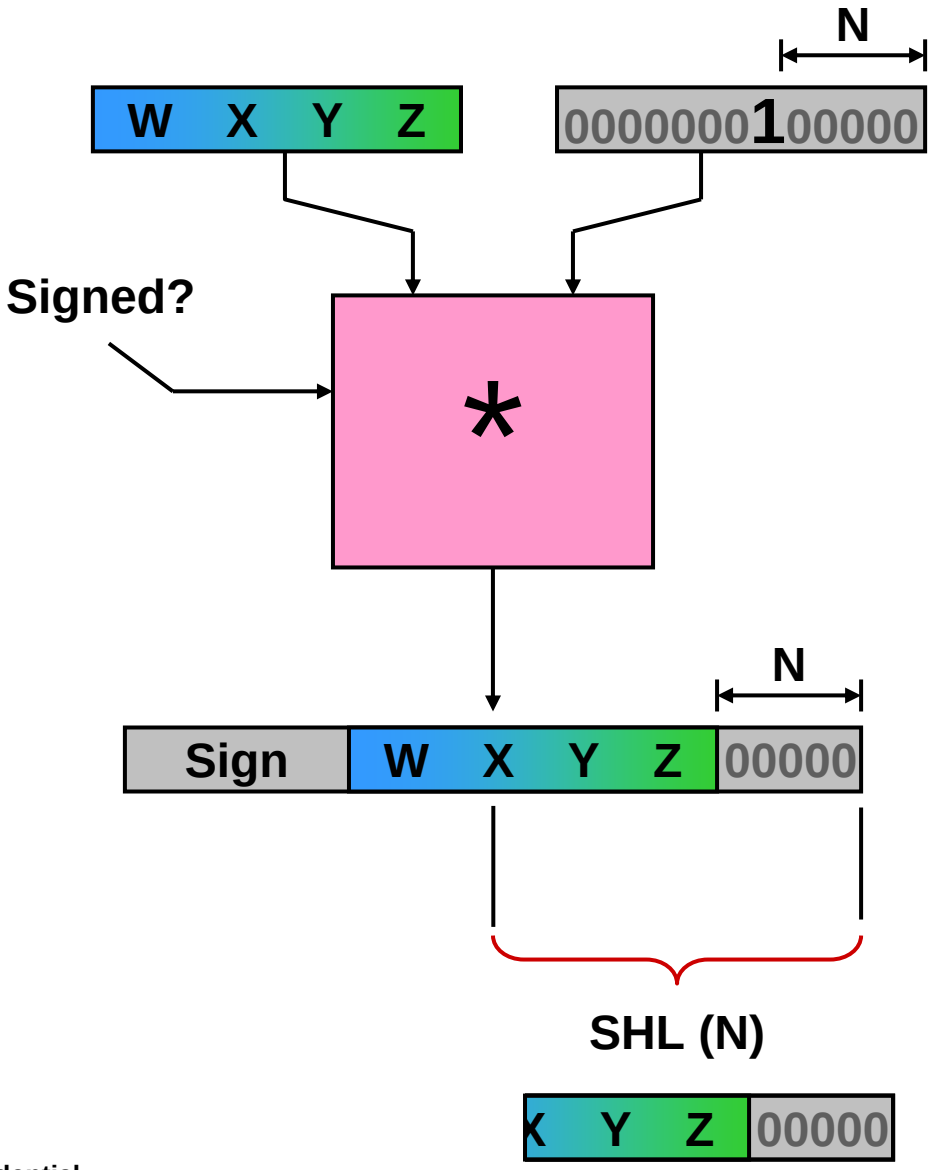




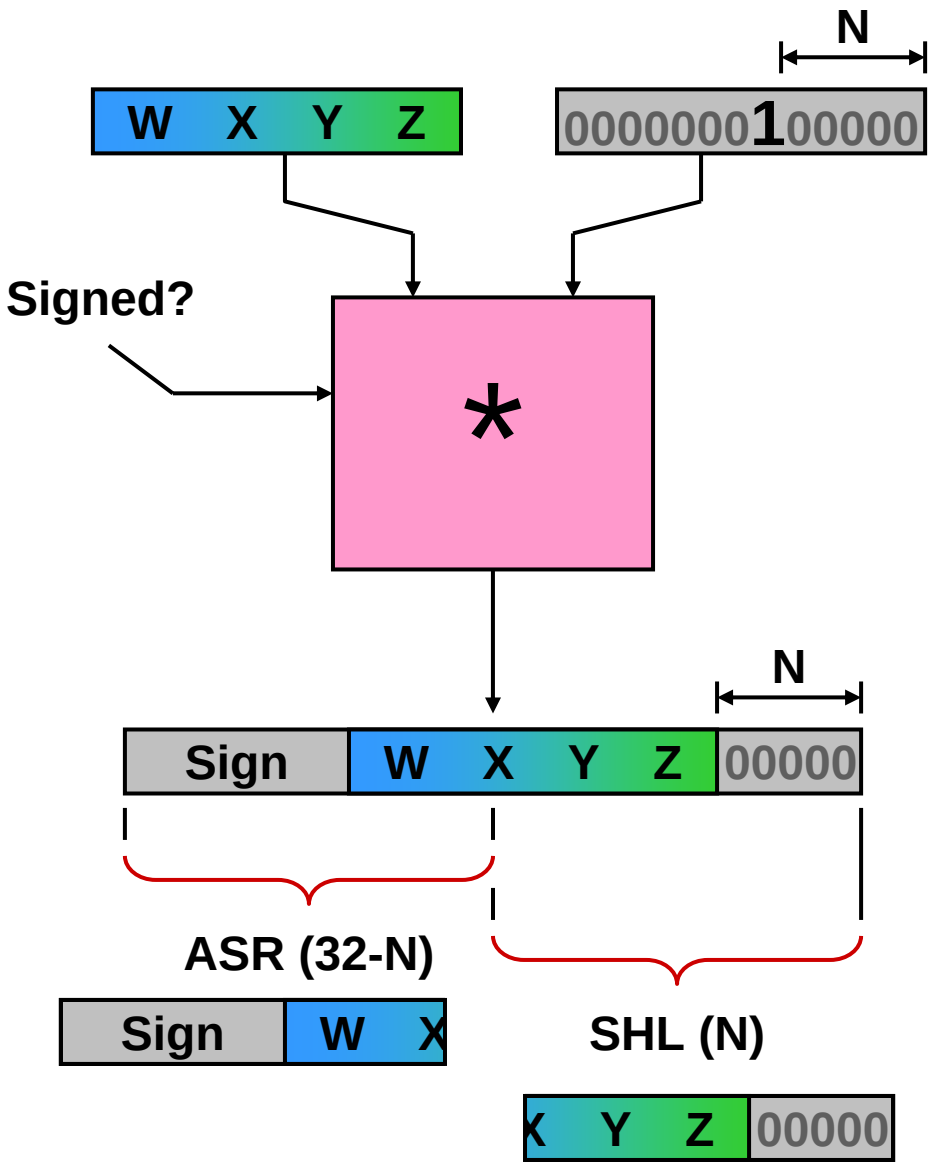
Area Cost

	ASIC	FPGA
Multiplexers 4:1	Low	High
Multipliers *	High	Medium

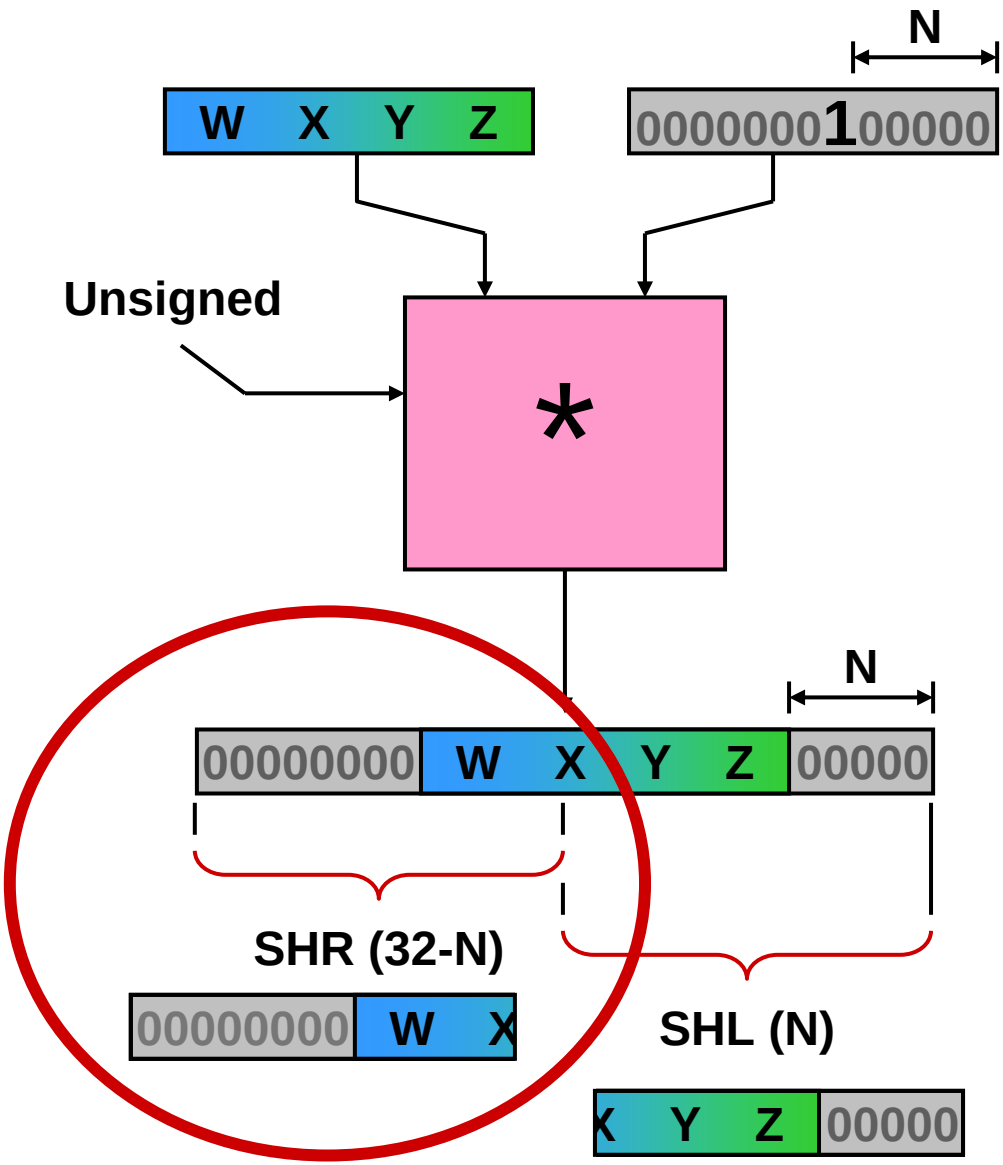
Shifters using Multipliers



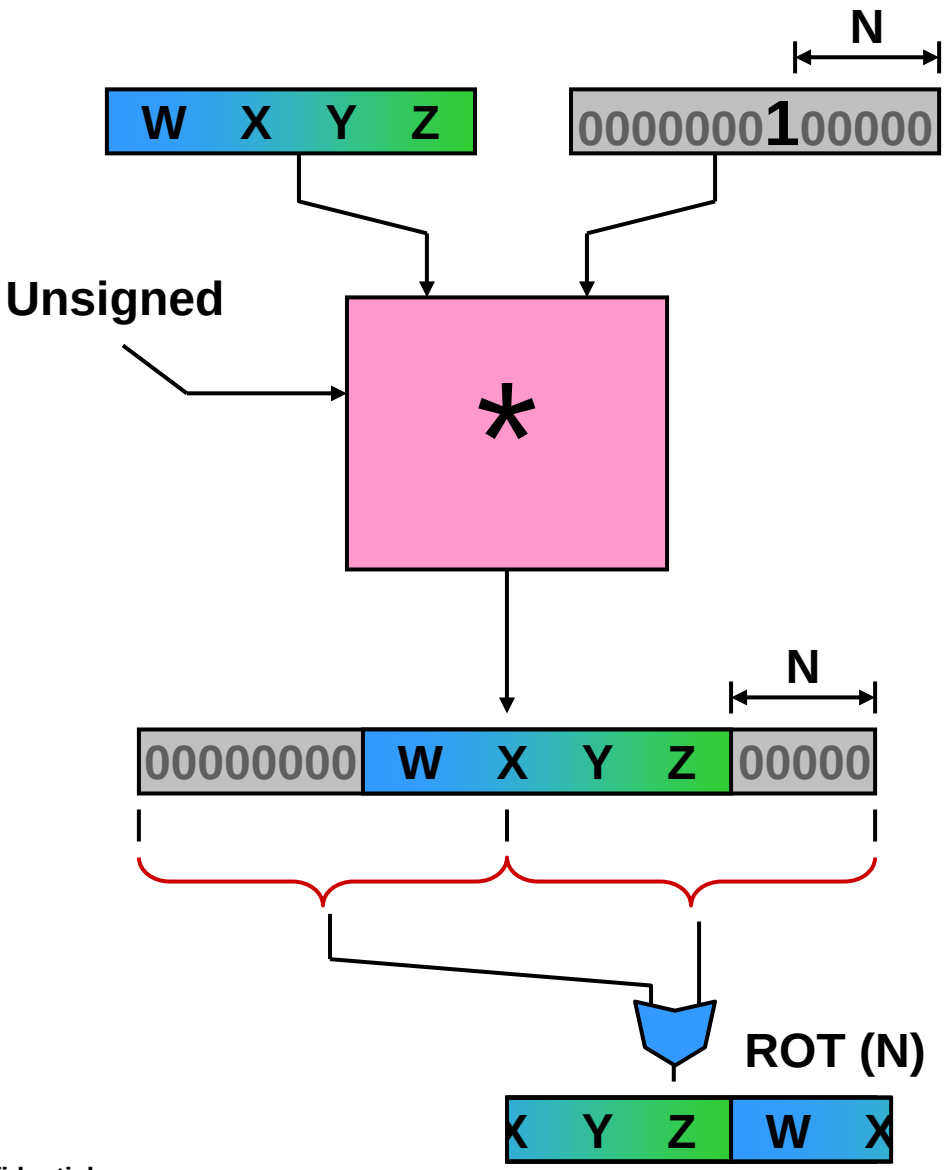
Shifters using Multipliers



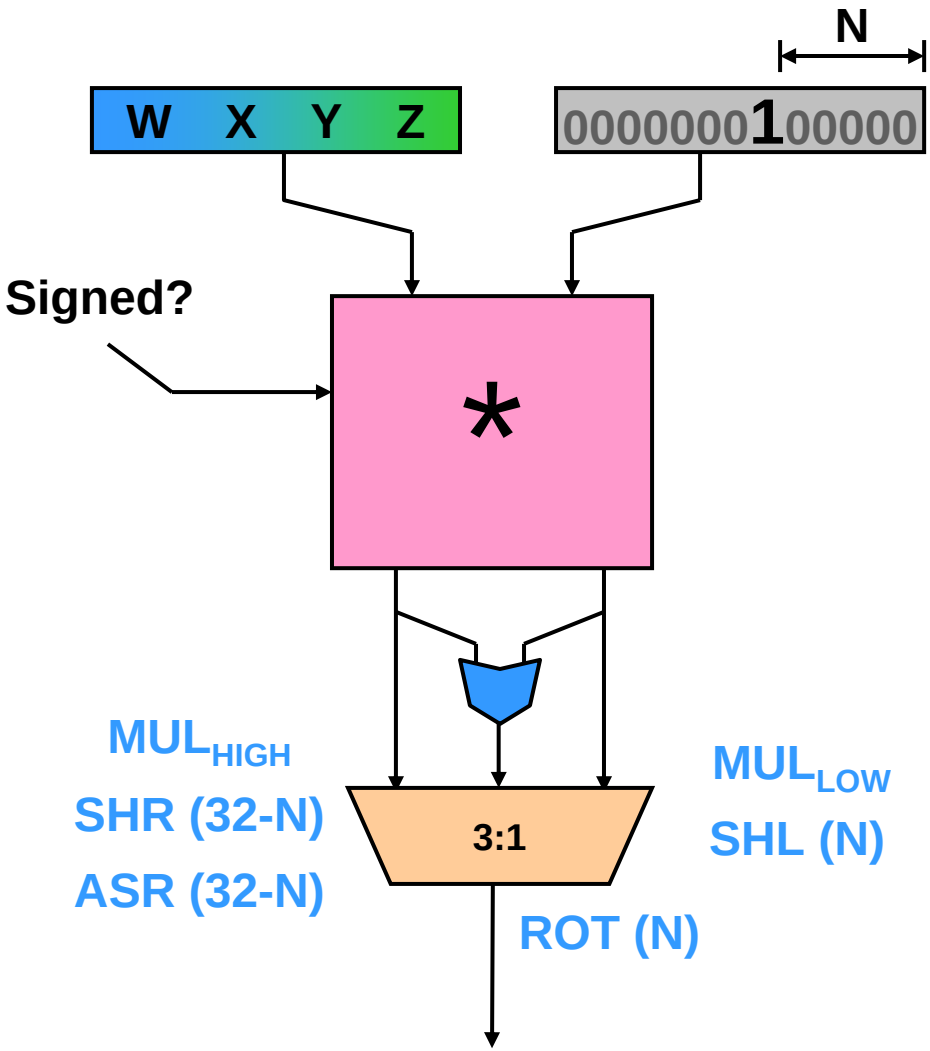
Shifters using Multipliers

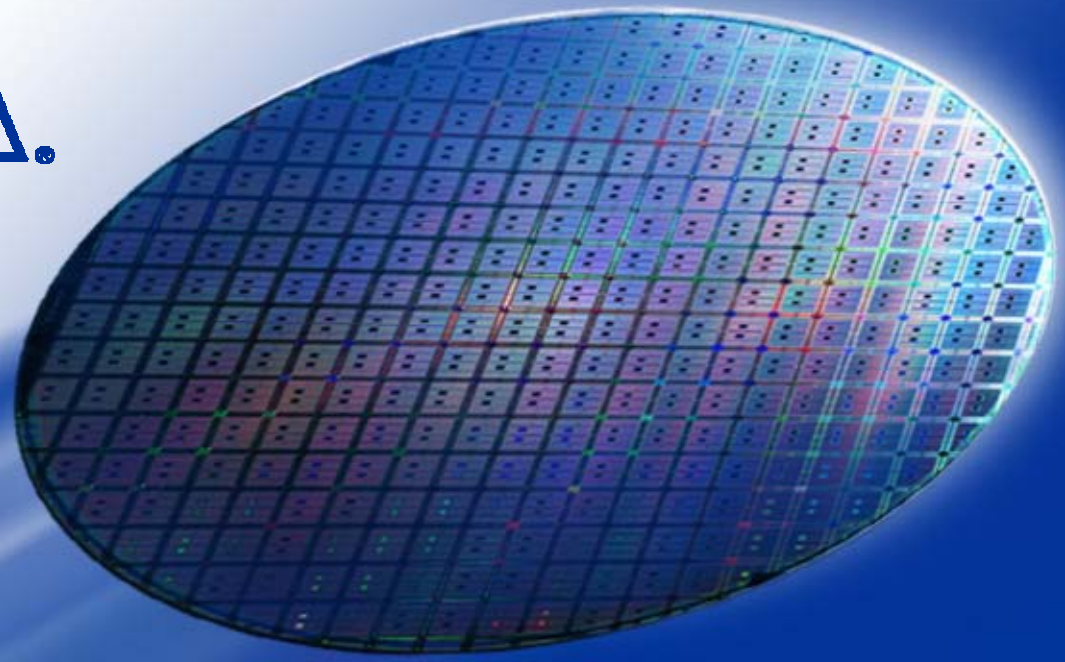


Shifters using Multipliers



Shifters using Multipliers

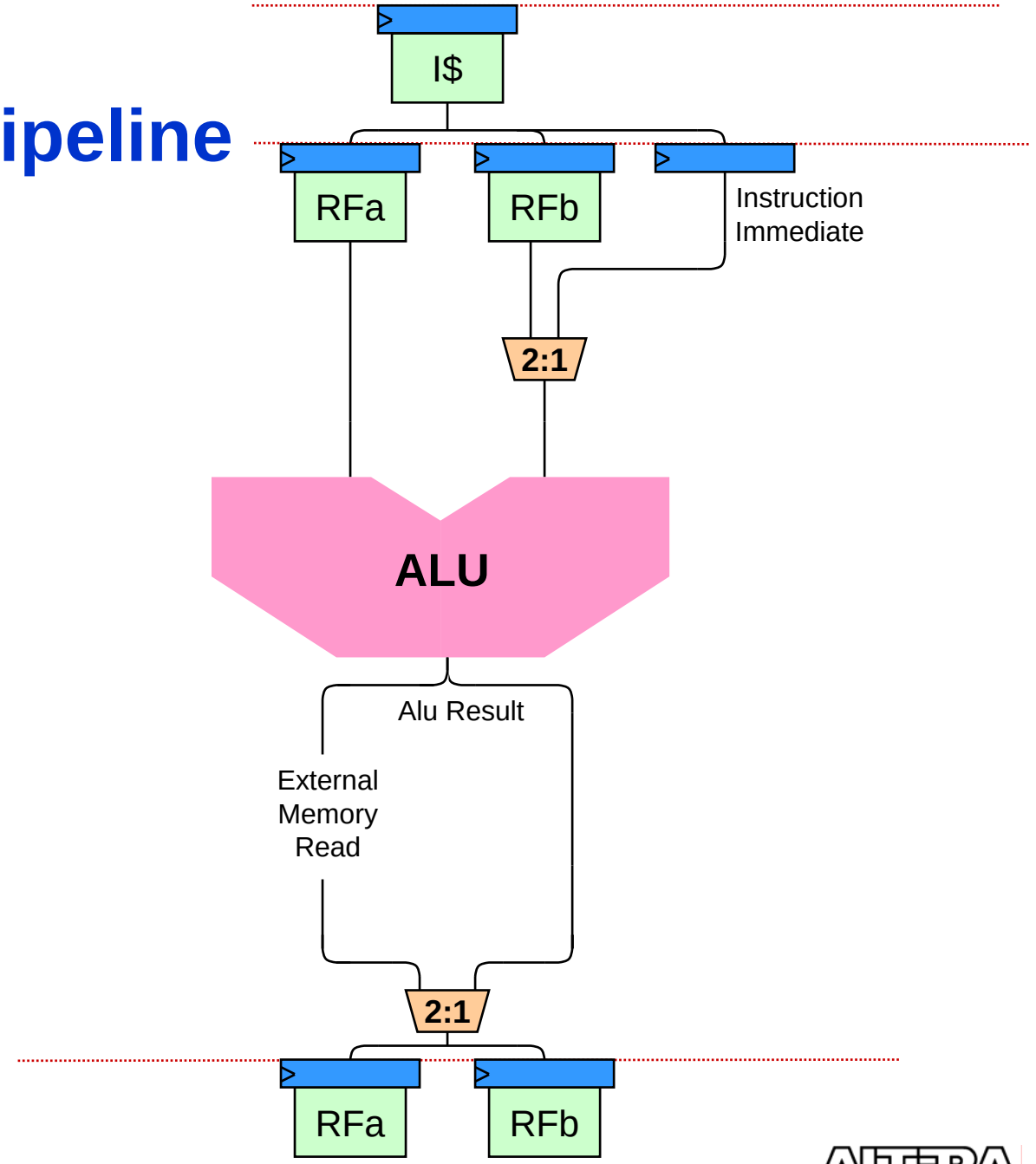




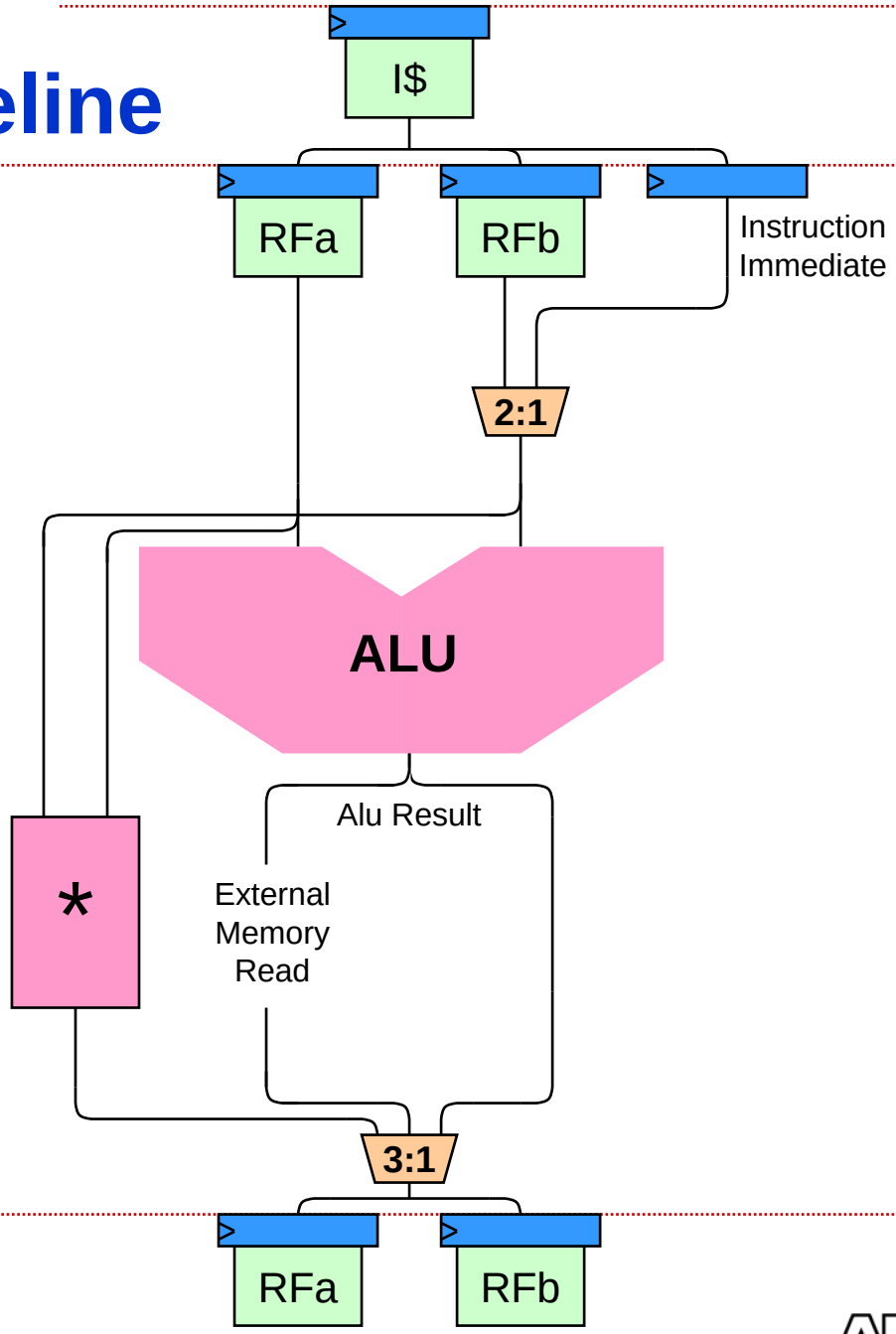
Case Study: The Design of Nios II

The ALU

Case Study:
The NIOS II Pipeline

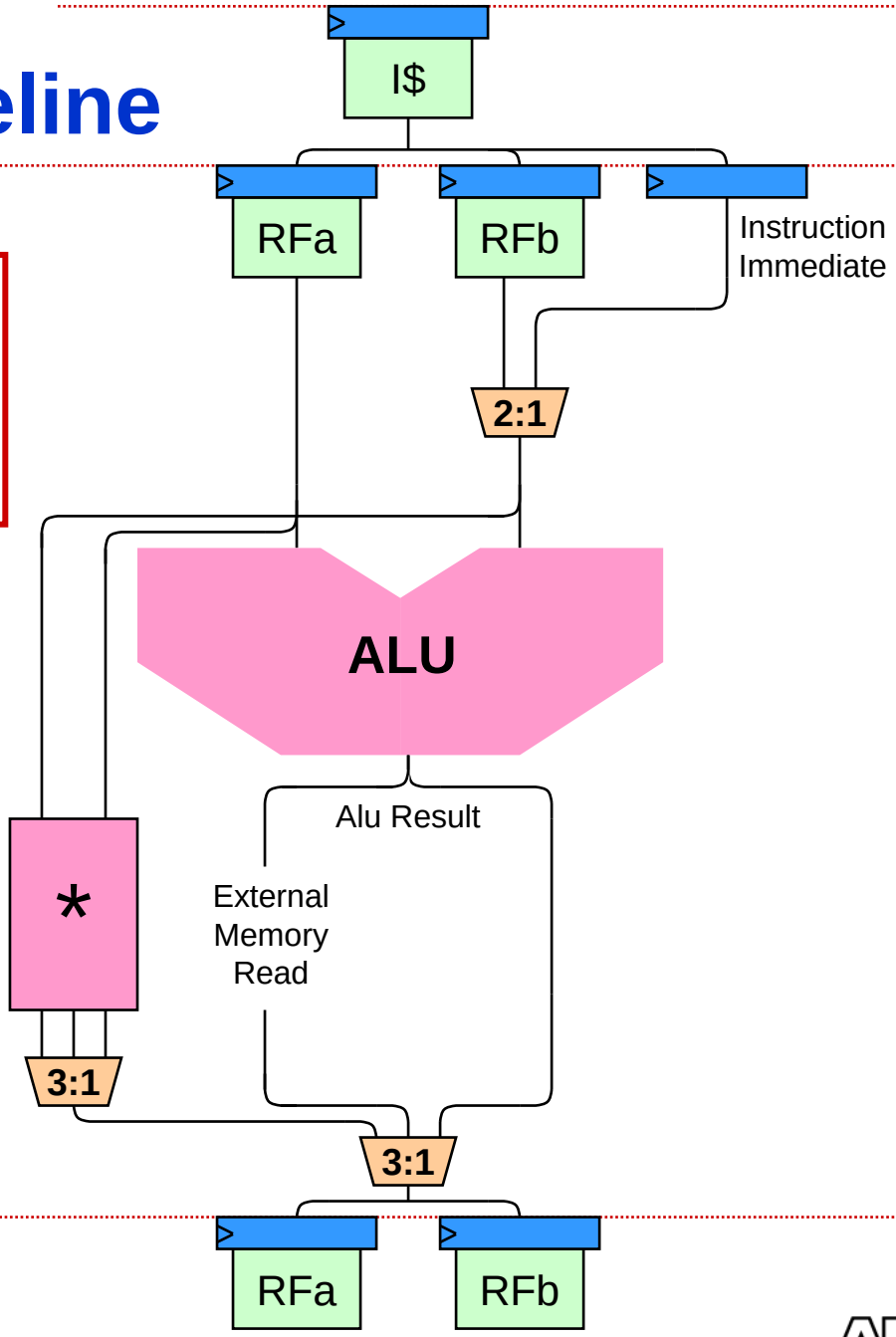


The NIOS II Pipeline

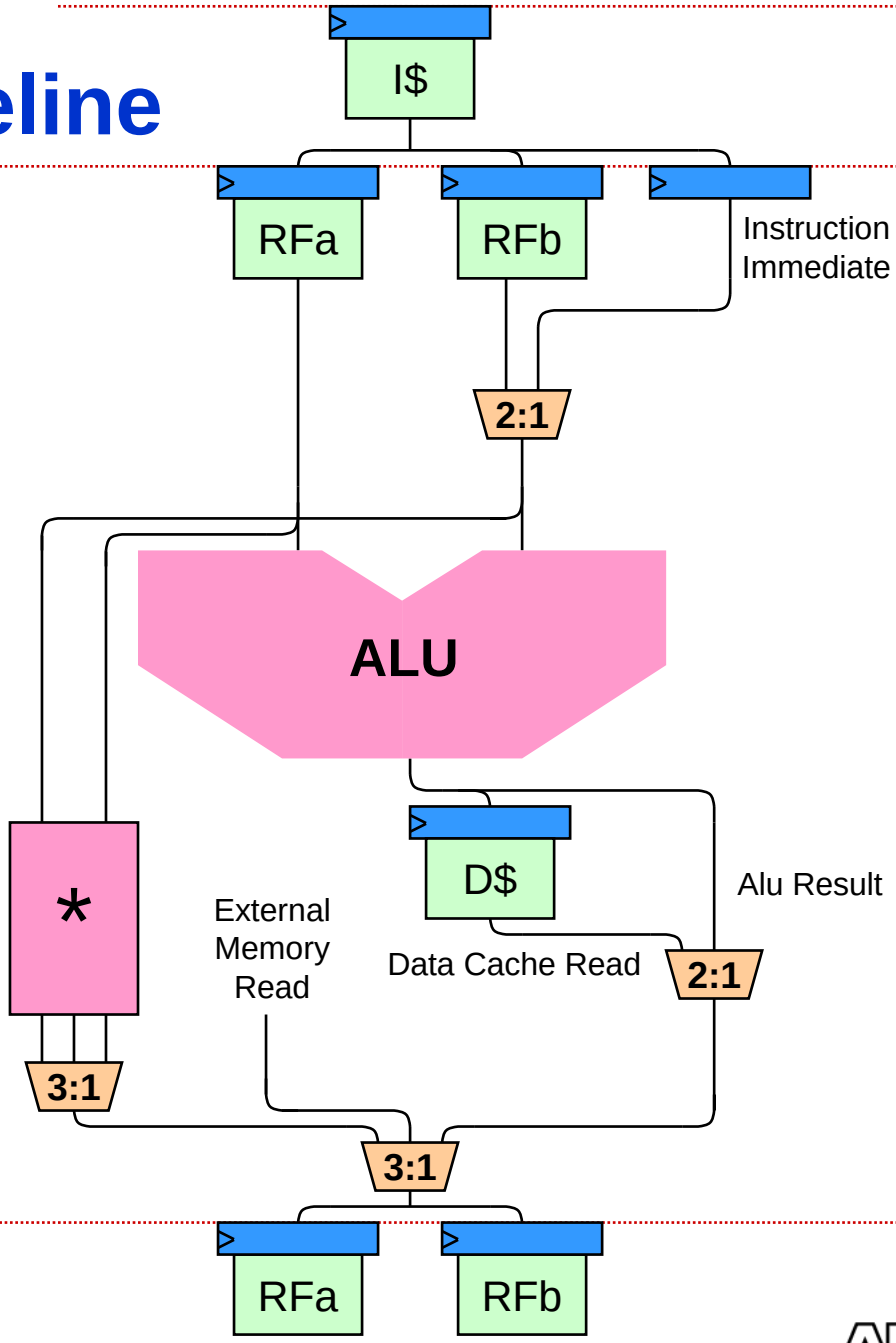


The NIOS II Pipeline

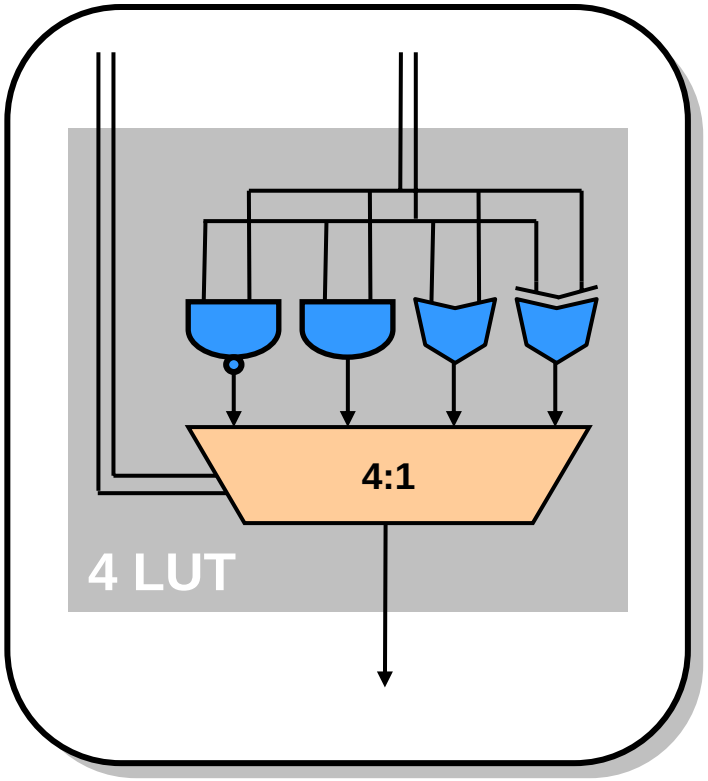
Multiplier is used for **Barrel-Shifts** as well as **Multiplication**

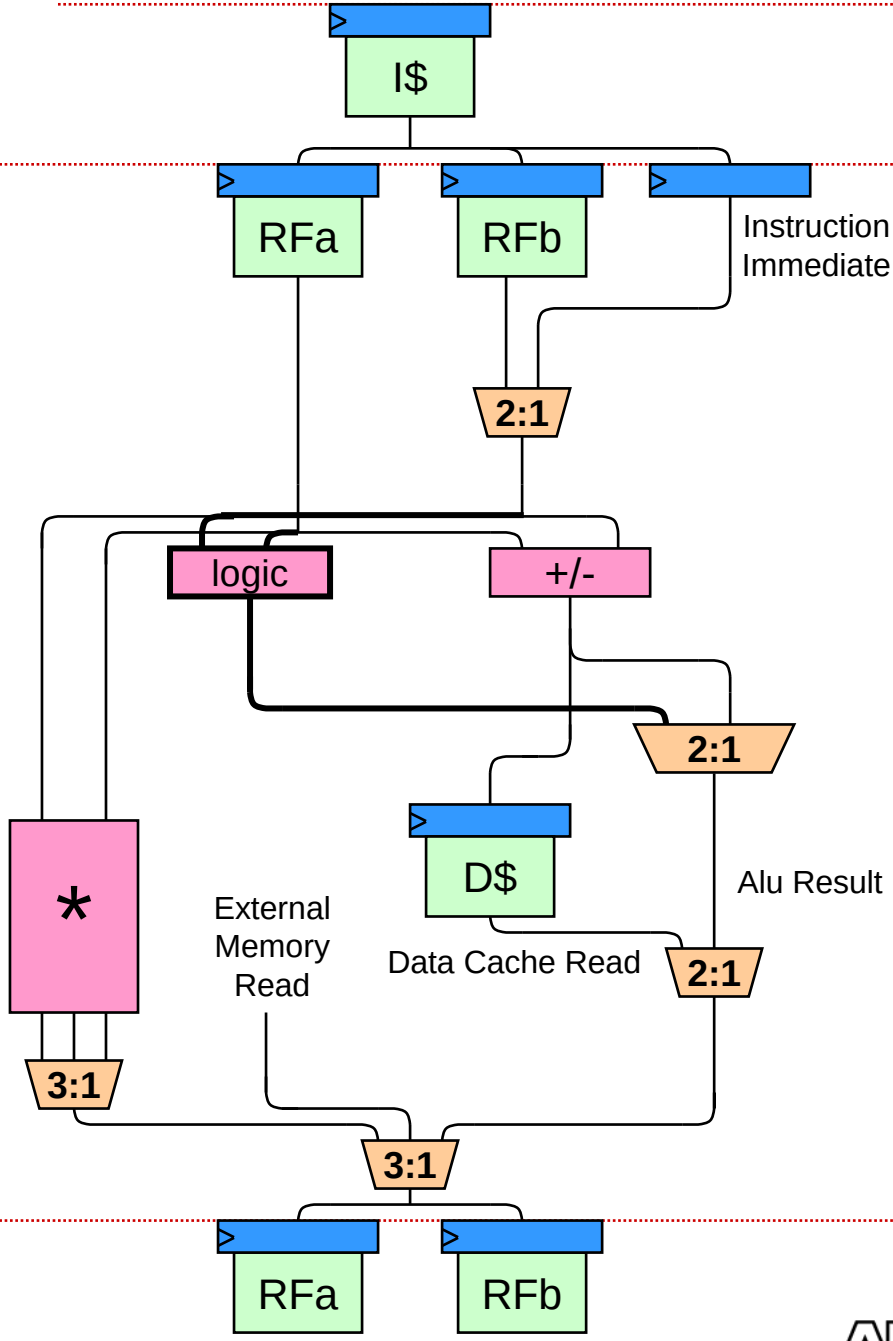


The NIOS II Pipeline

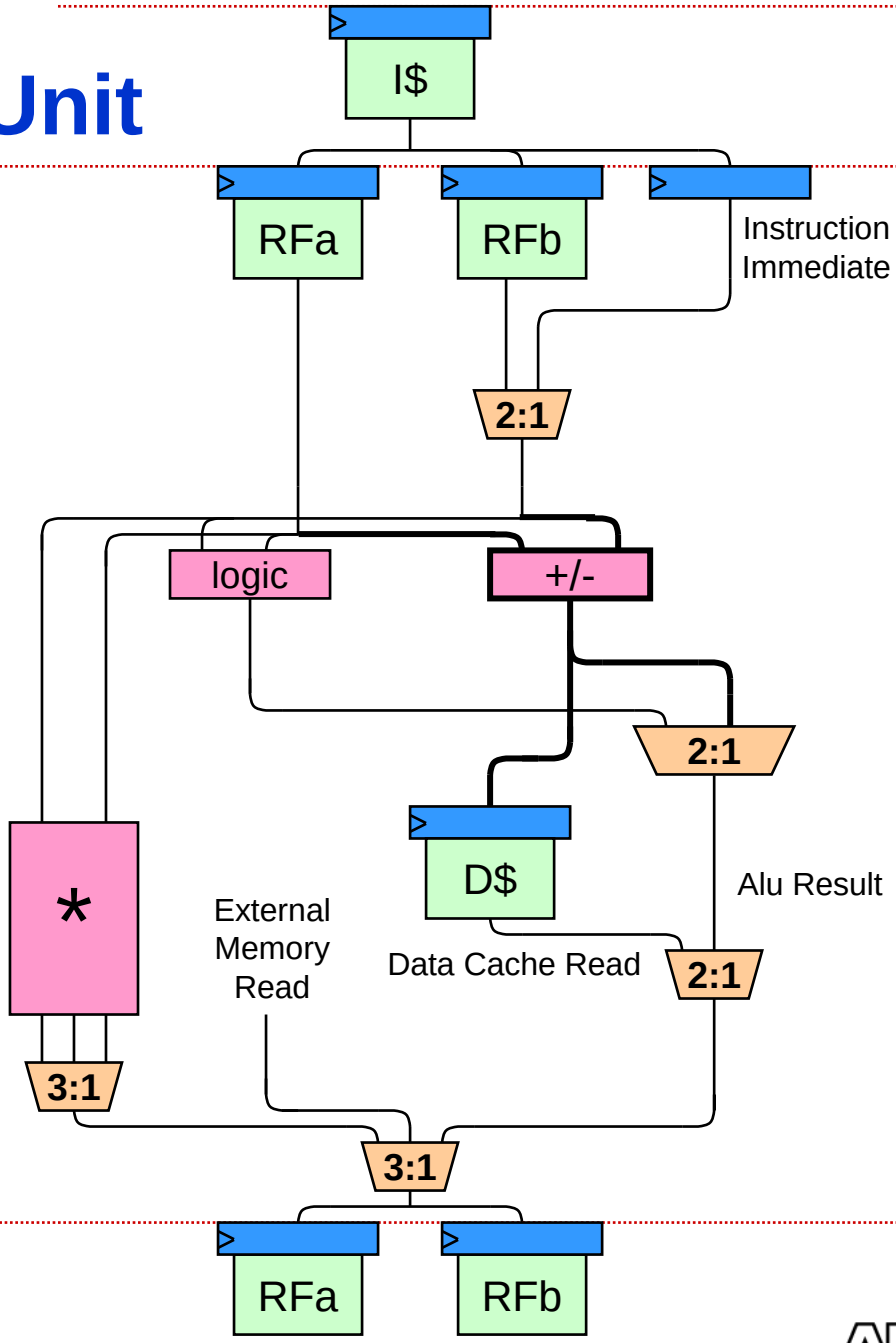


The Logic Unit





The Arithmetic Unit



The Comparator Unit

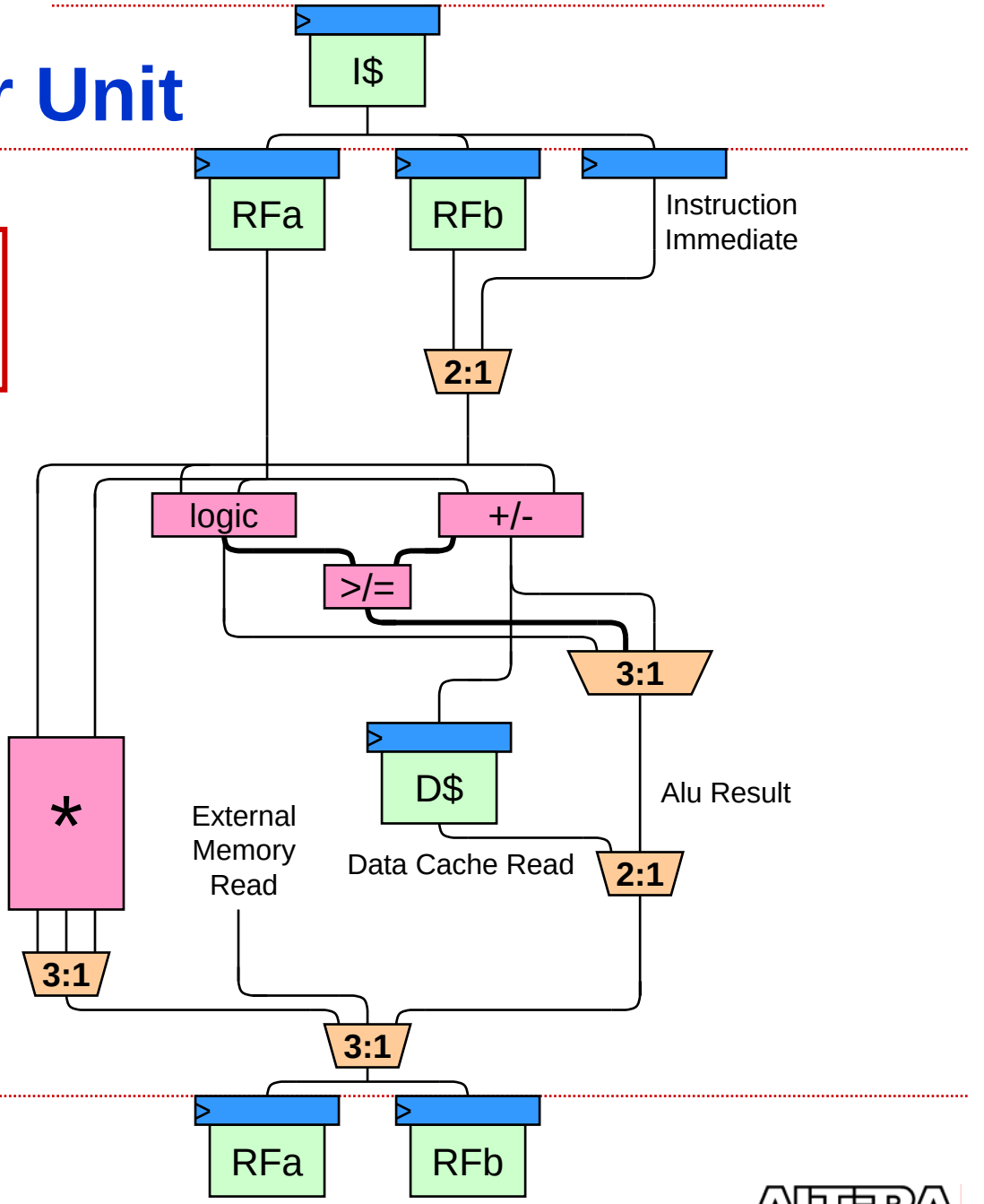
Nios II has no explicit Flags

CMP.op r3, r2, r1

IF (**r2 op r1**)

THEN **R3** = **0x00000001**

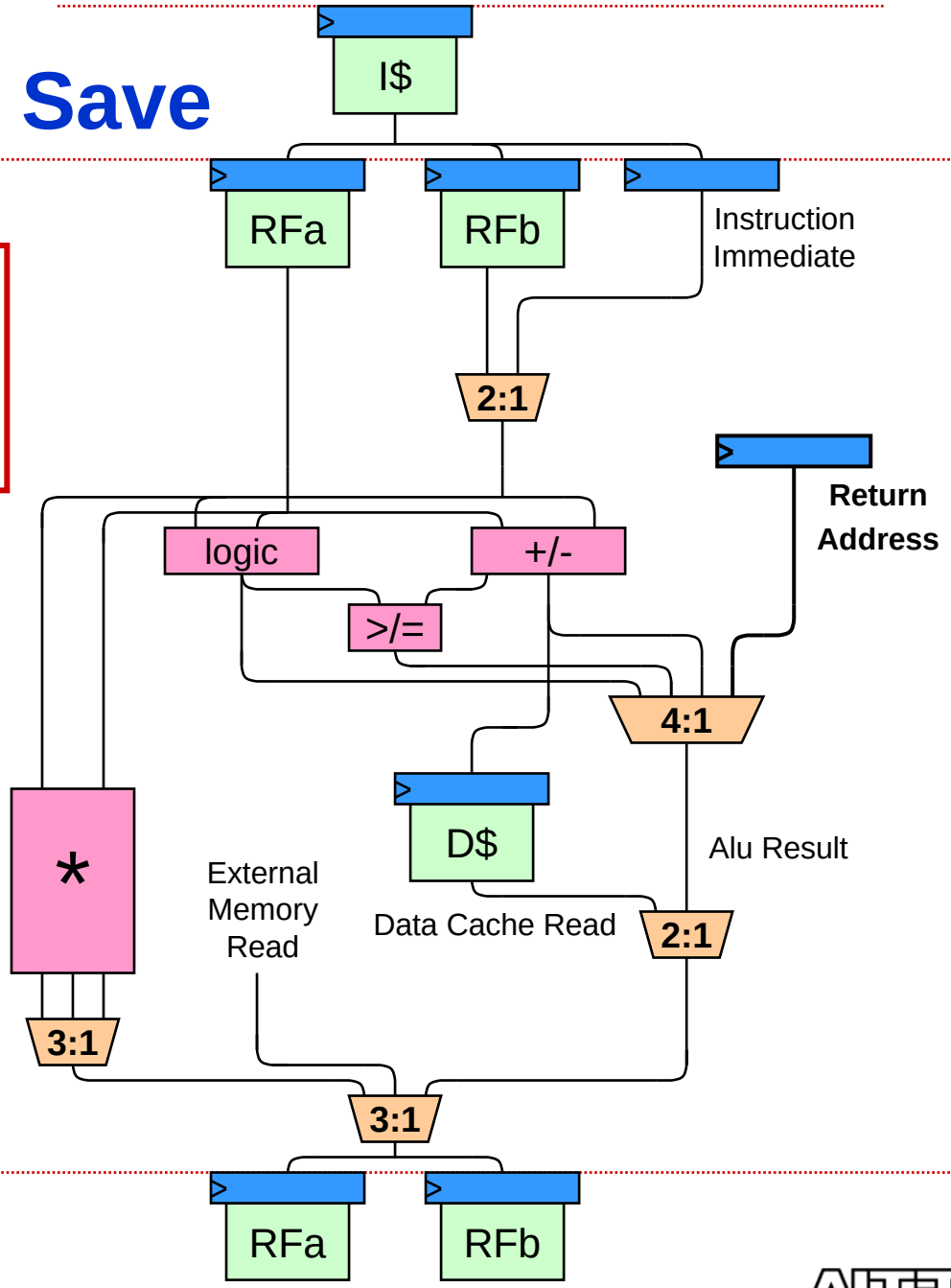
ELSE **R3** = **0x00000000**

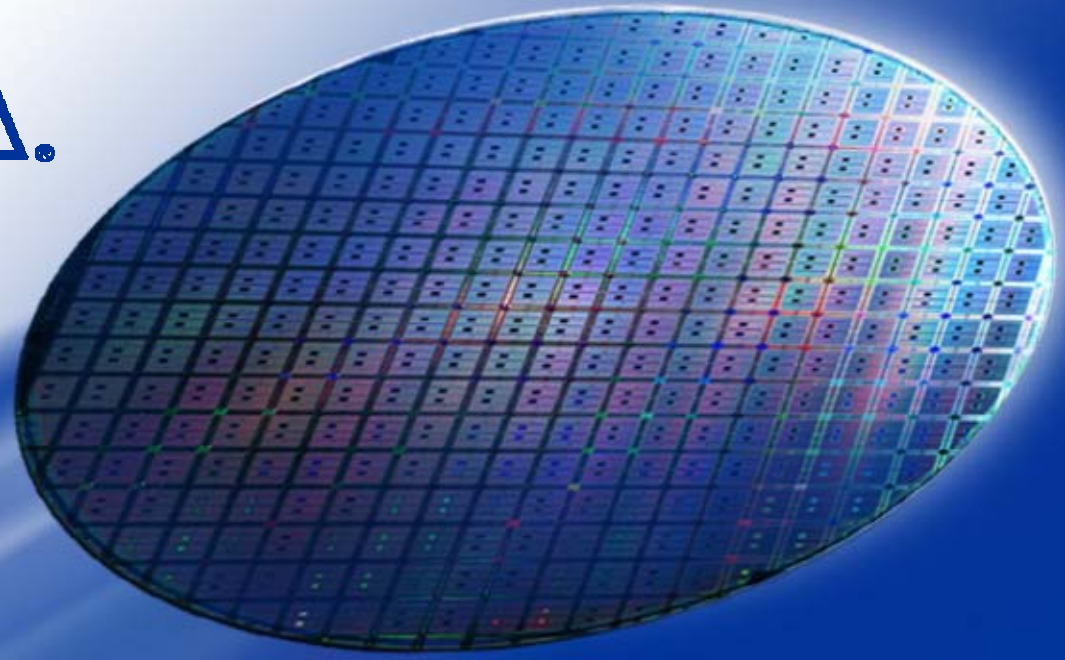


Return Address Save

Return Address is saved in a Link Register

Instructions that save Return Address
CALL
TRAP
INTERUPT
BREAK



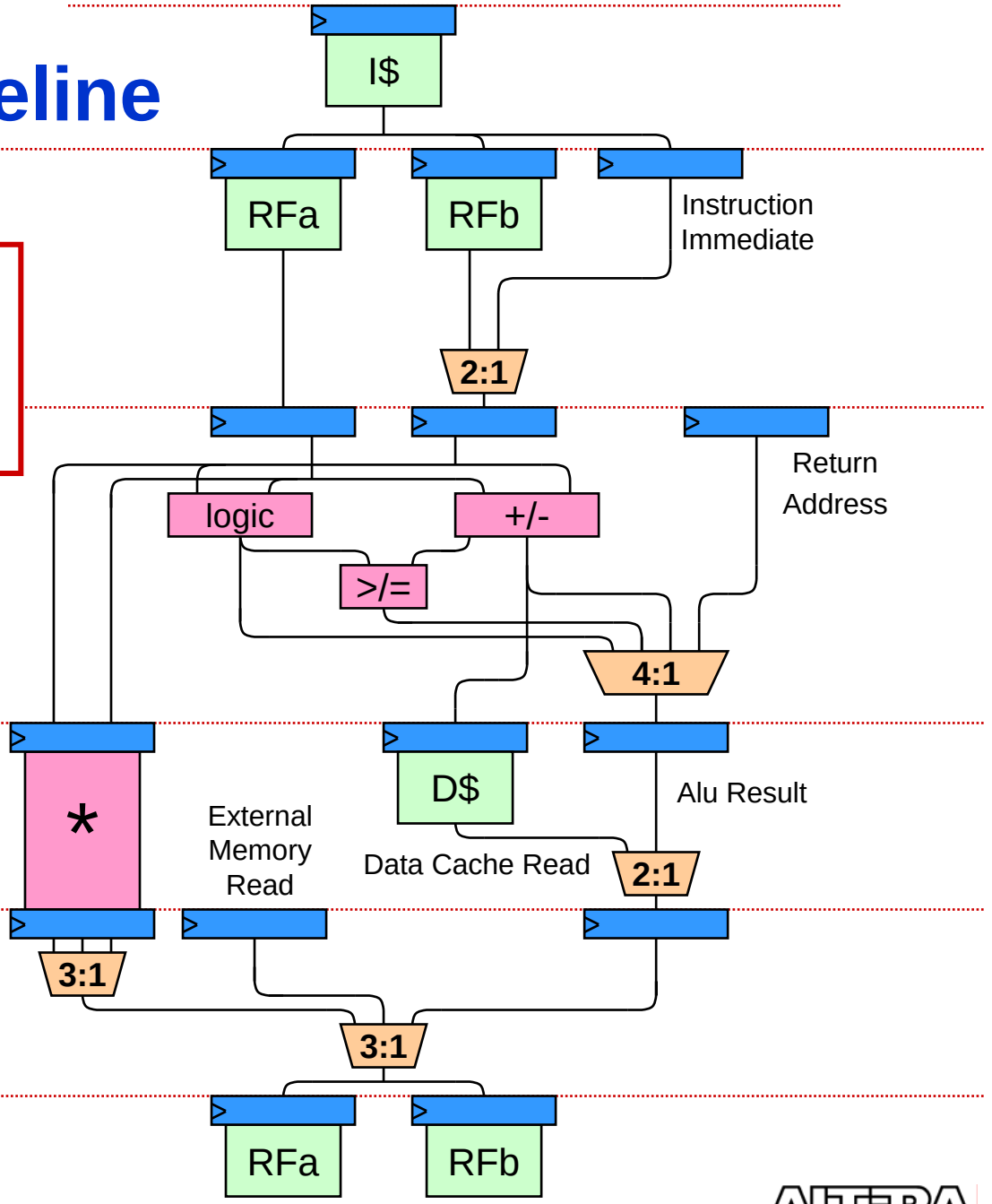


Case Study: The Design of Nios II

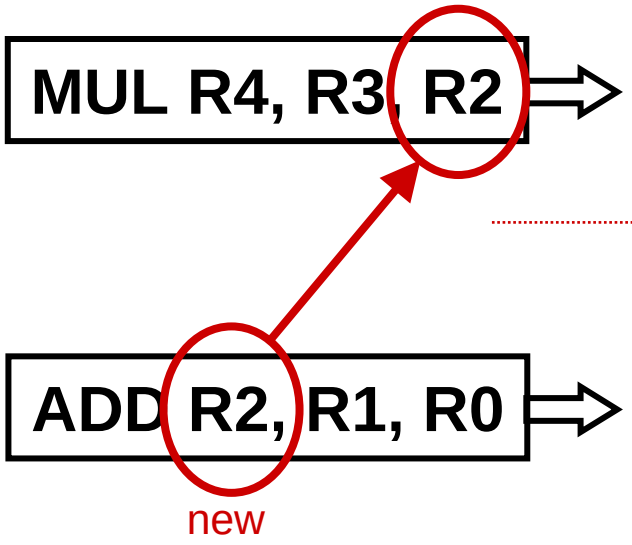
Increasing the Clock Rate

The NIOS II Pipeline

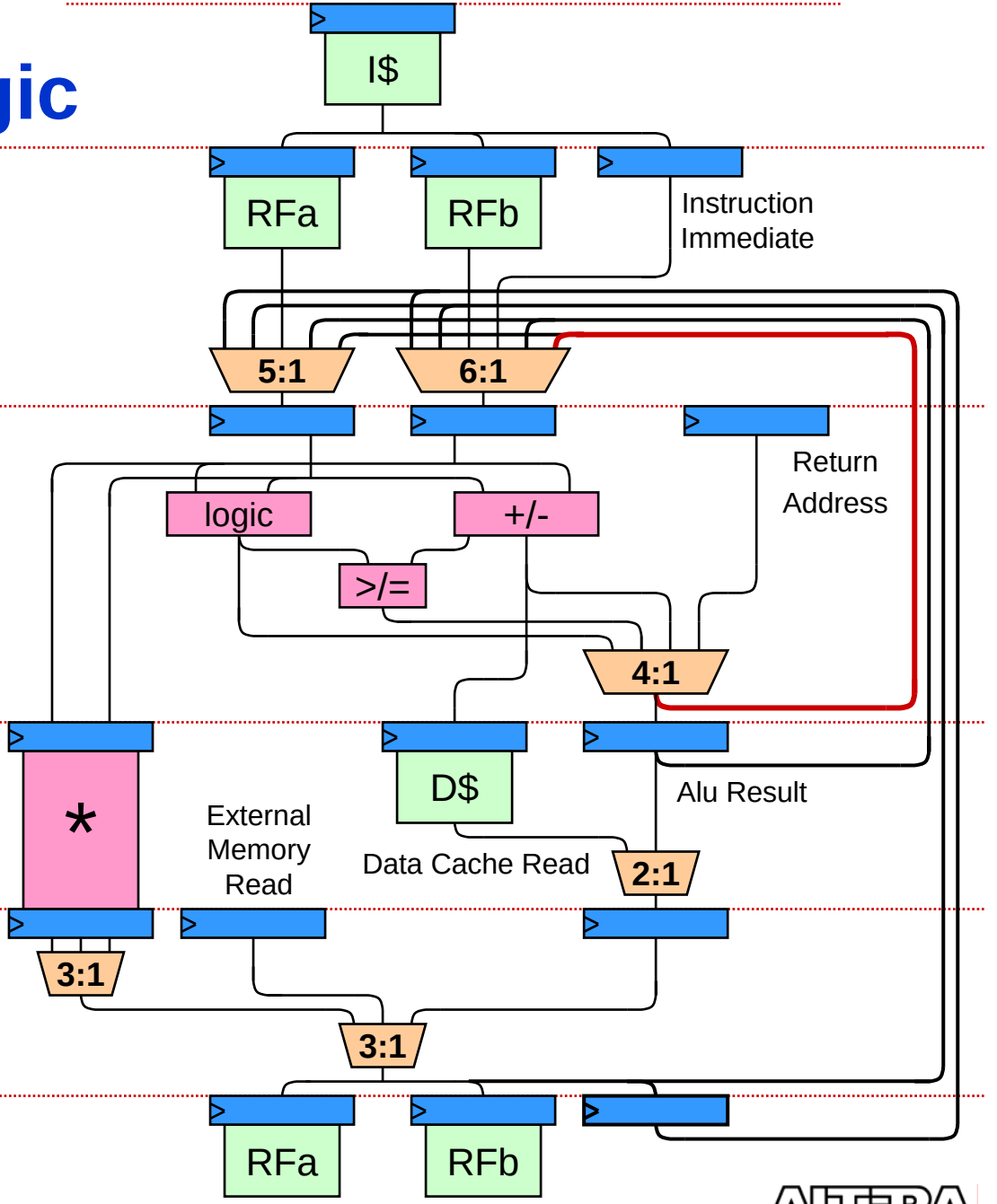
Pipeline to achieve a high Clock Rate (fmax)

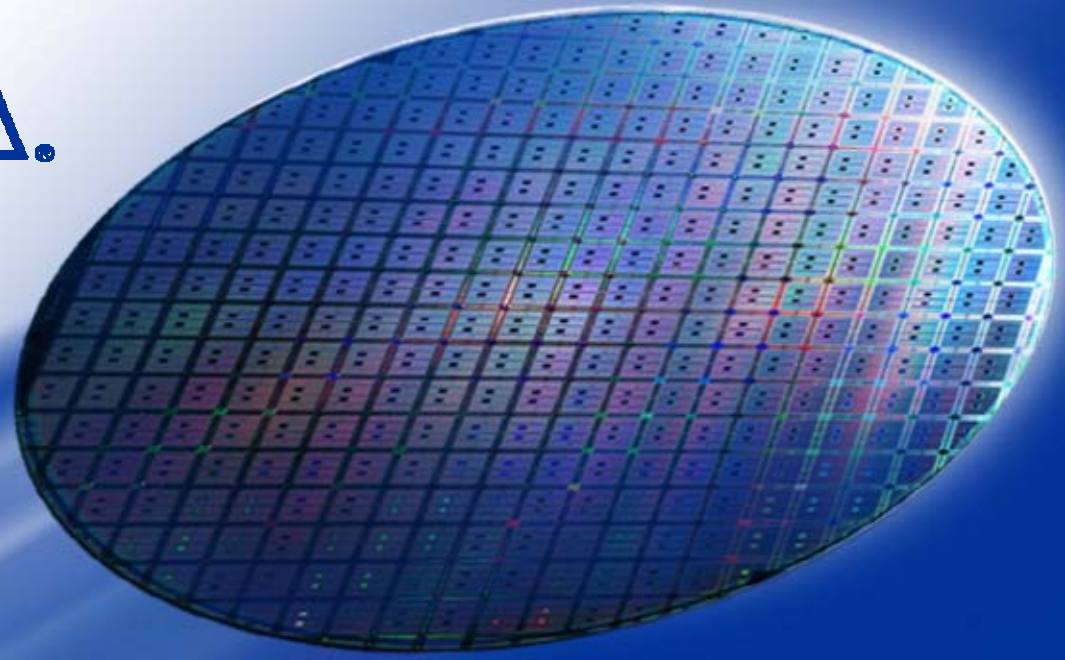


Forwarding Logic



Fowarding needed to update out-of-date values in the pipeline

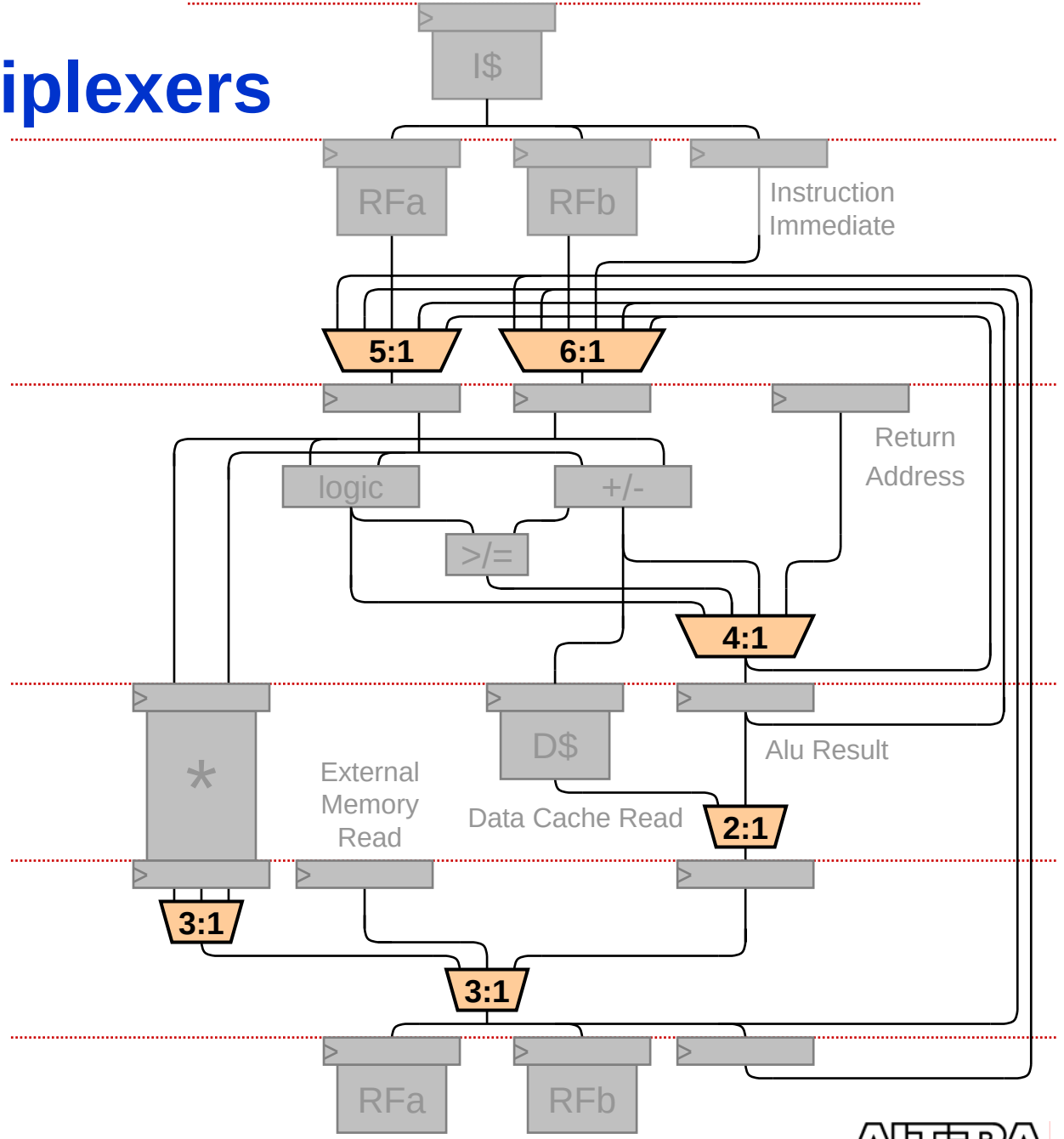




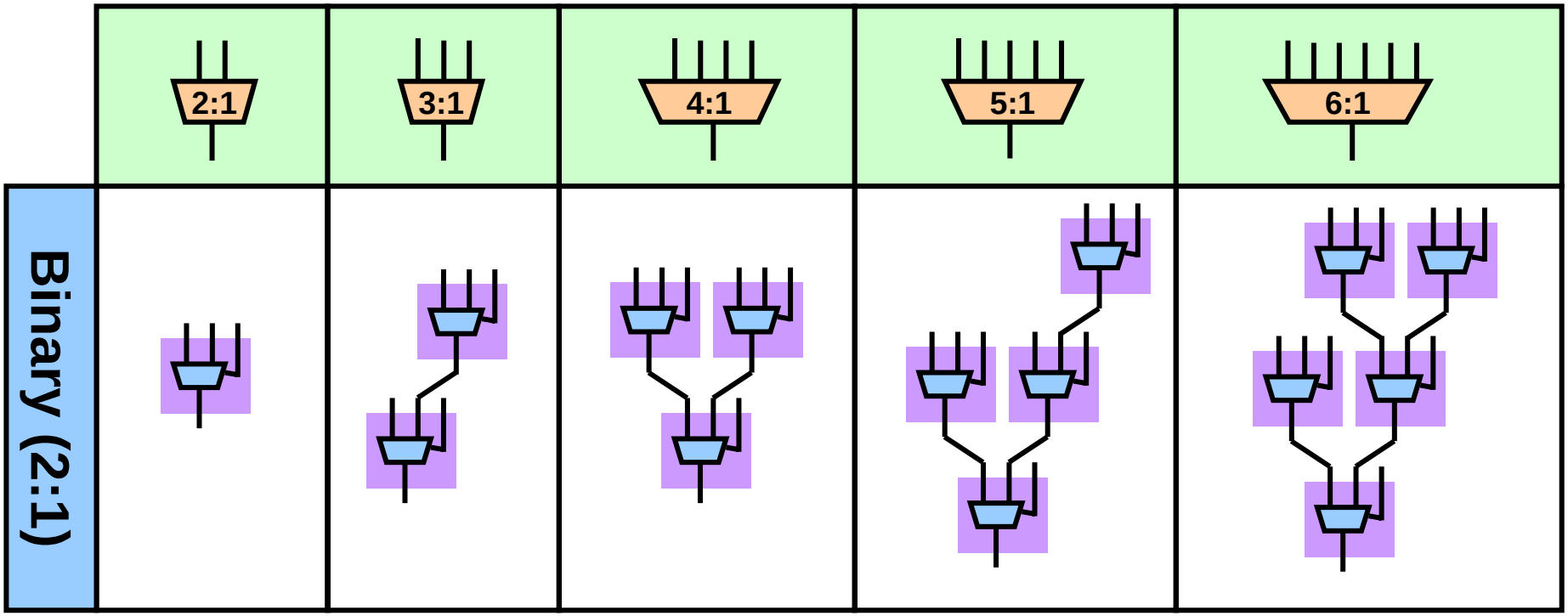
Case Study: The Design of Nios II

The Cost of Multiplexers

NIOS II Multiplexers

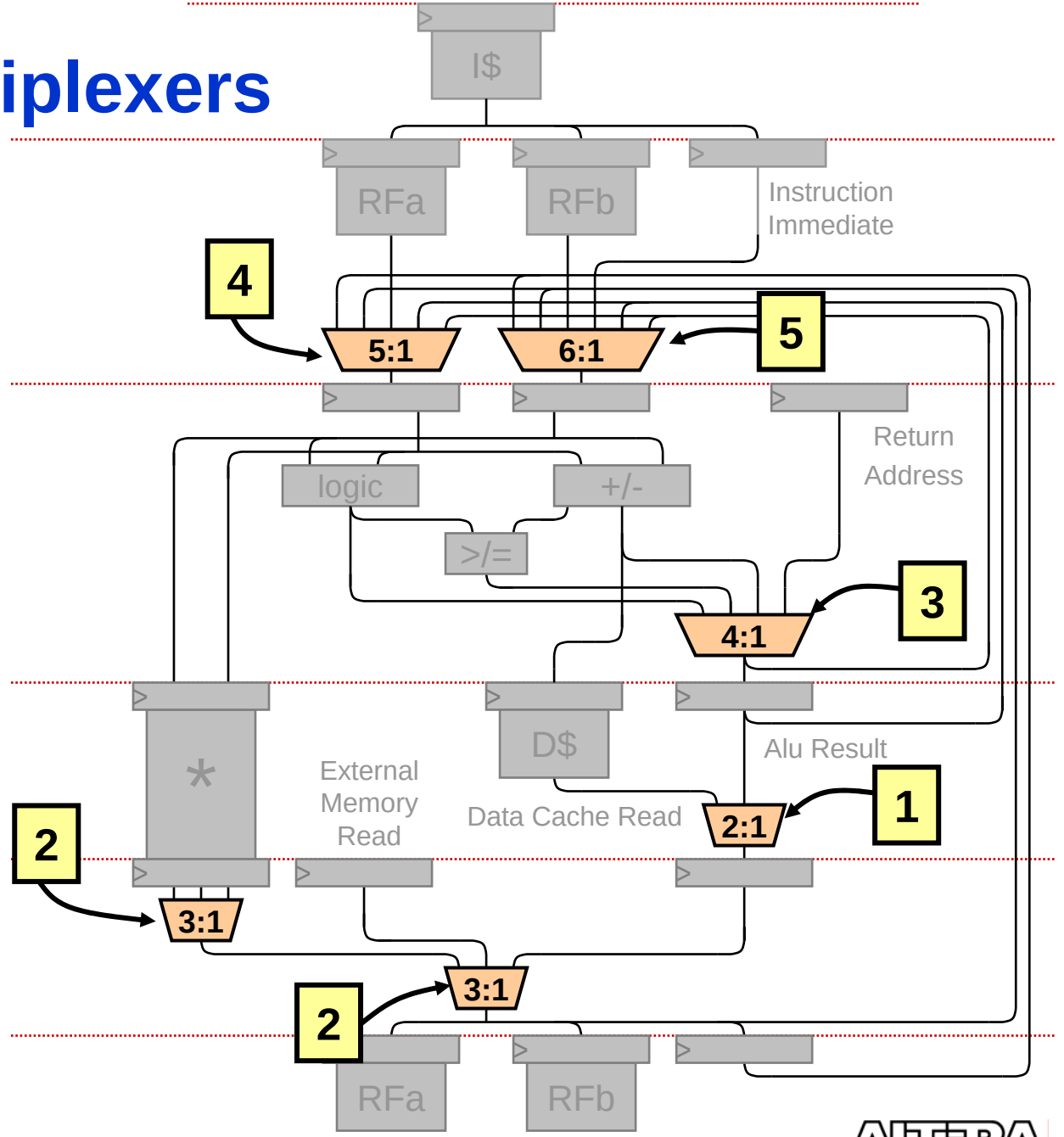


What is the Cost of a Multiplexer…?



Natural Implementation Choice for an ASIC

NIOS II Multiplexers



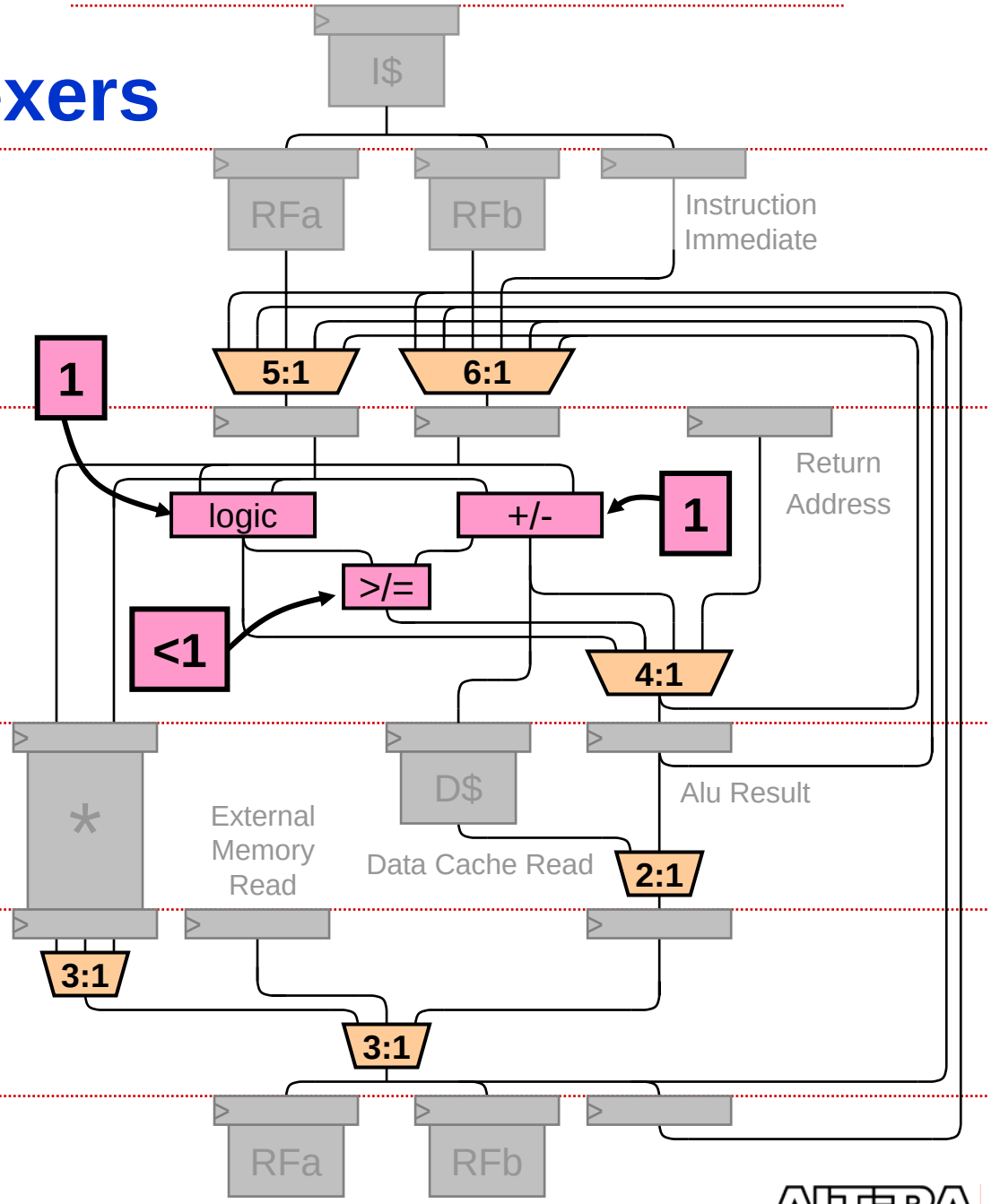
544 LEs
(17 x 32bits)

NIOS II Multiplexers

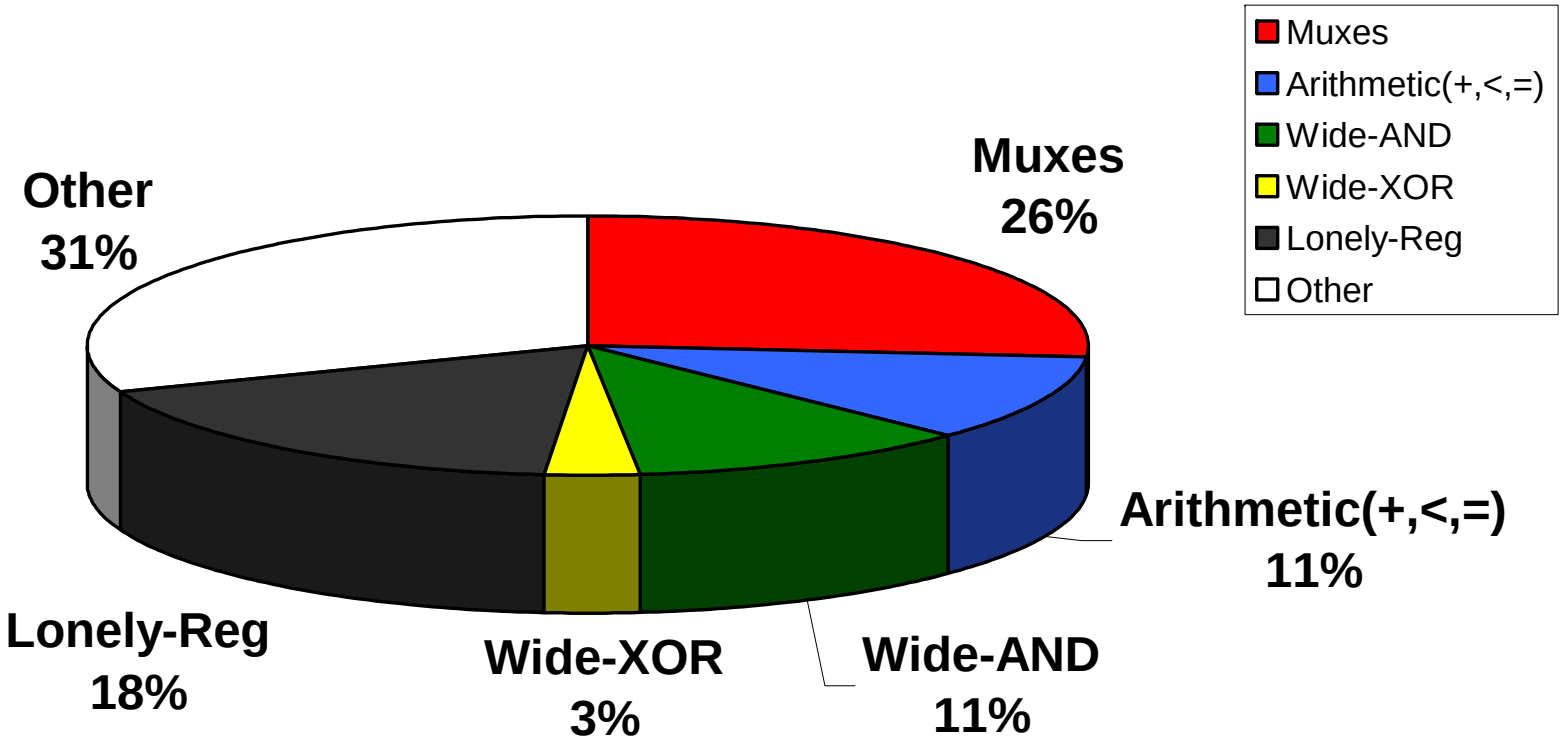
78 LEs

Multiplexer Cost is Dominant

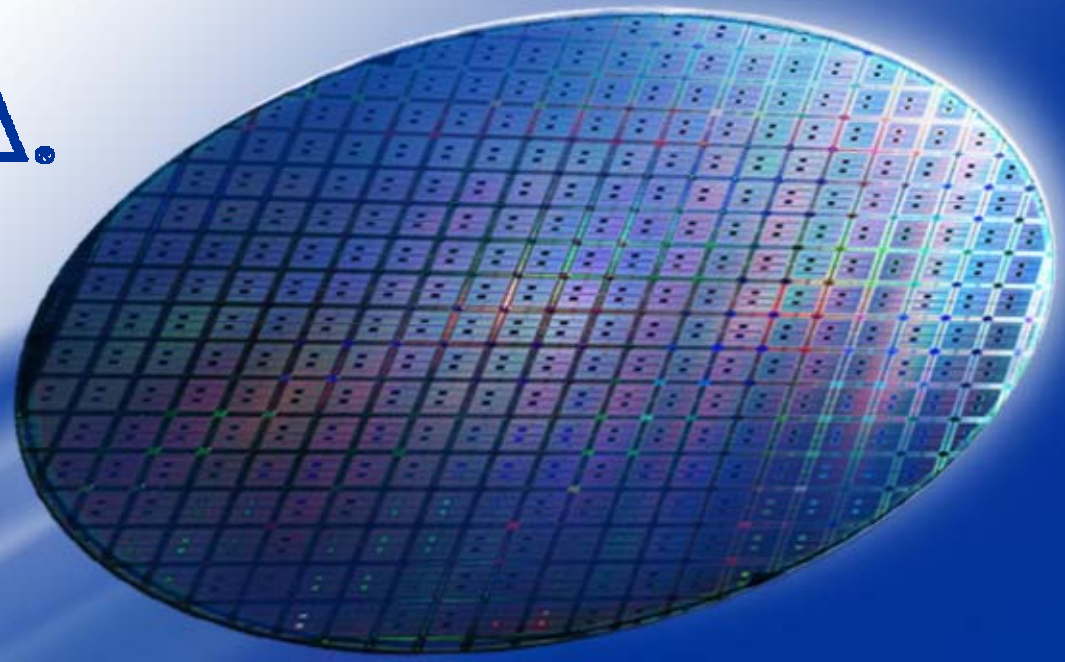
544 LEs
(17 x 32bits)



Area Usage in 100 Customer Designs



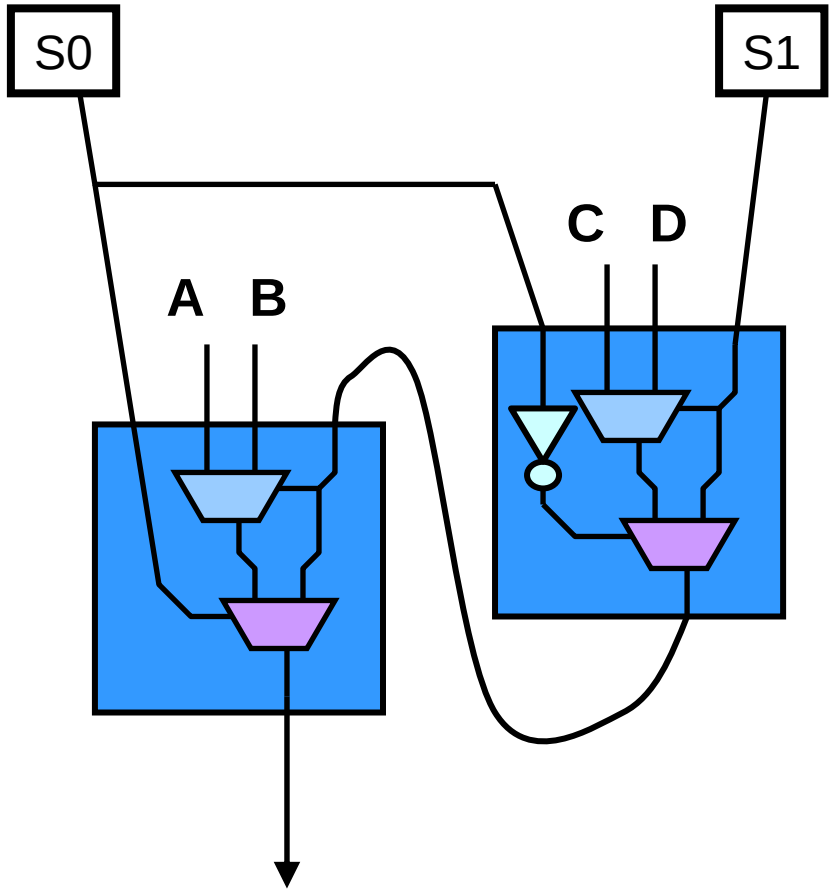
Many Designs contain lots of Multiplexers !



Multiplexers in FPGA

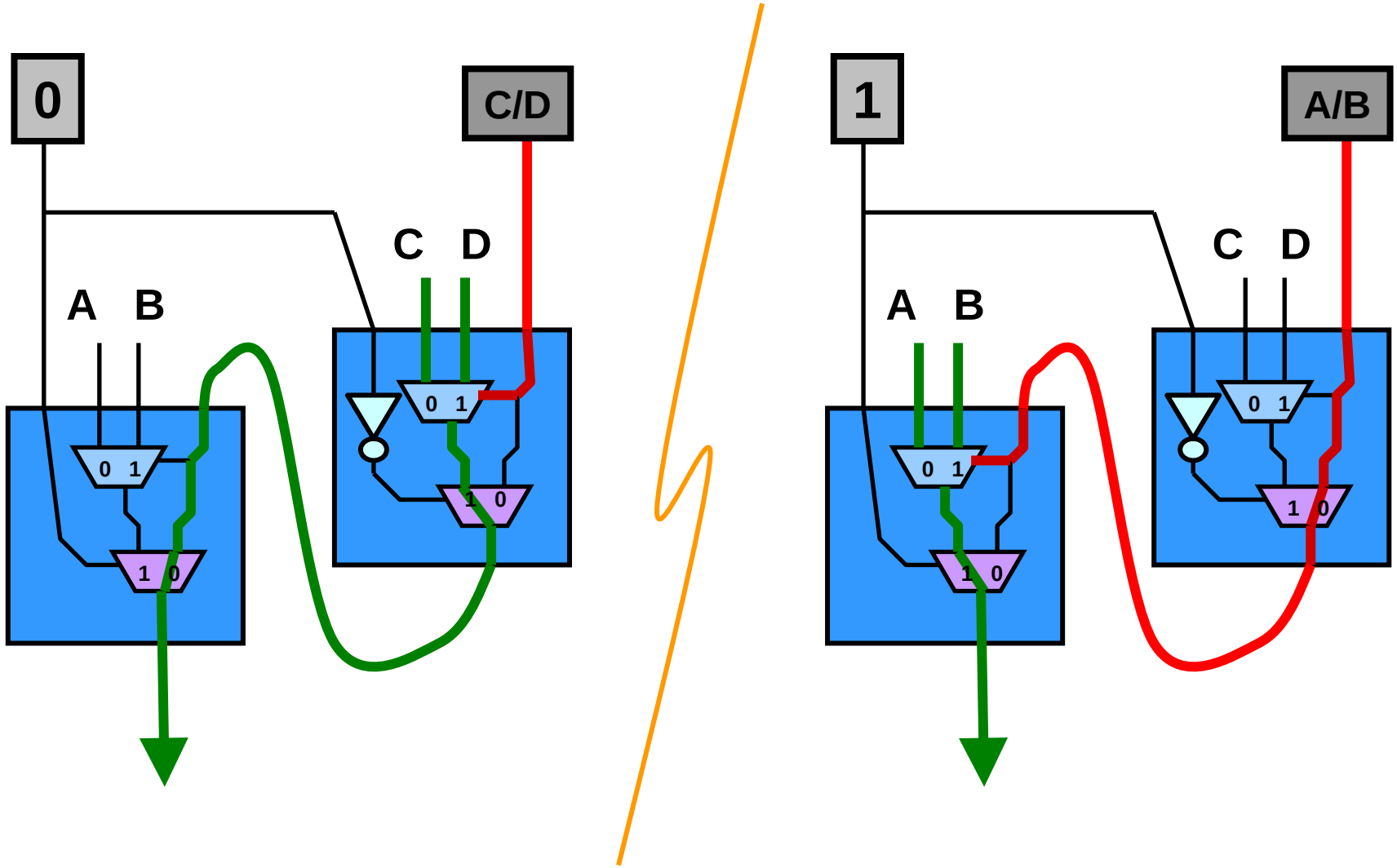
Low-Cost Multiplexers

Efficient 4:1 Mux on Stratix

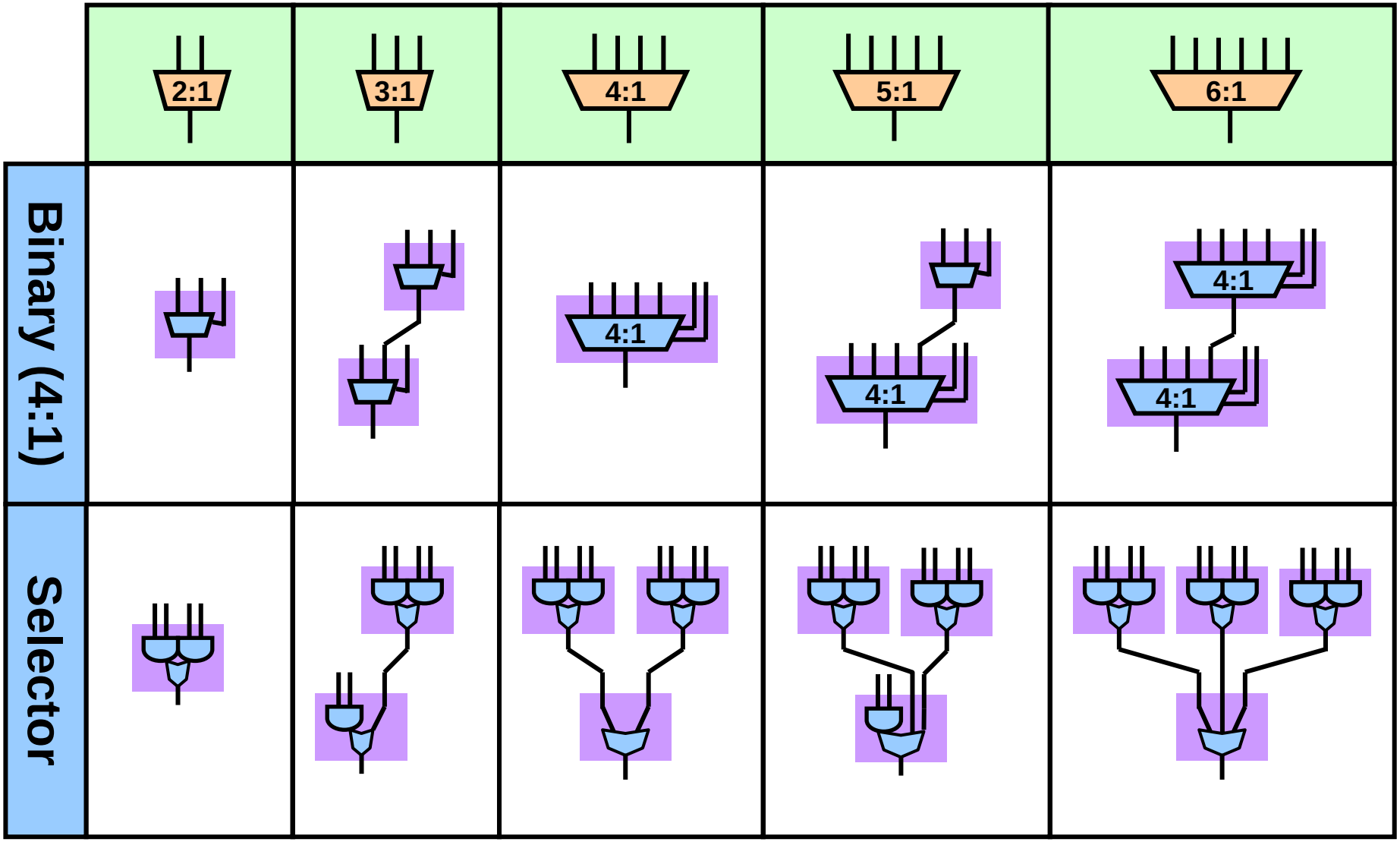


Uses just 2 LEs.

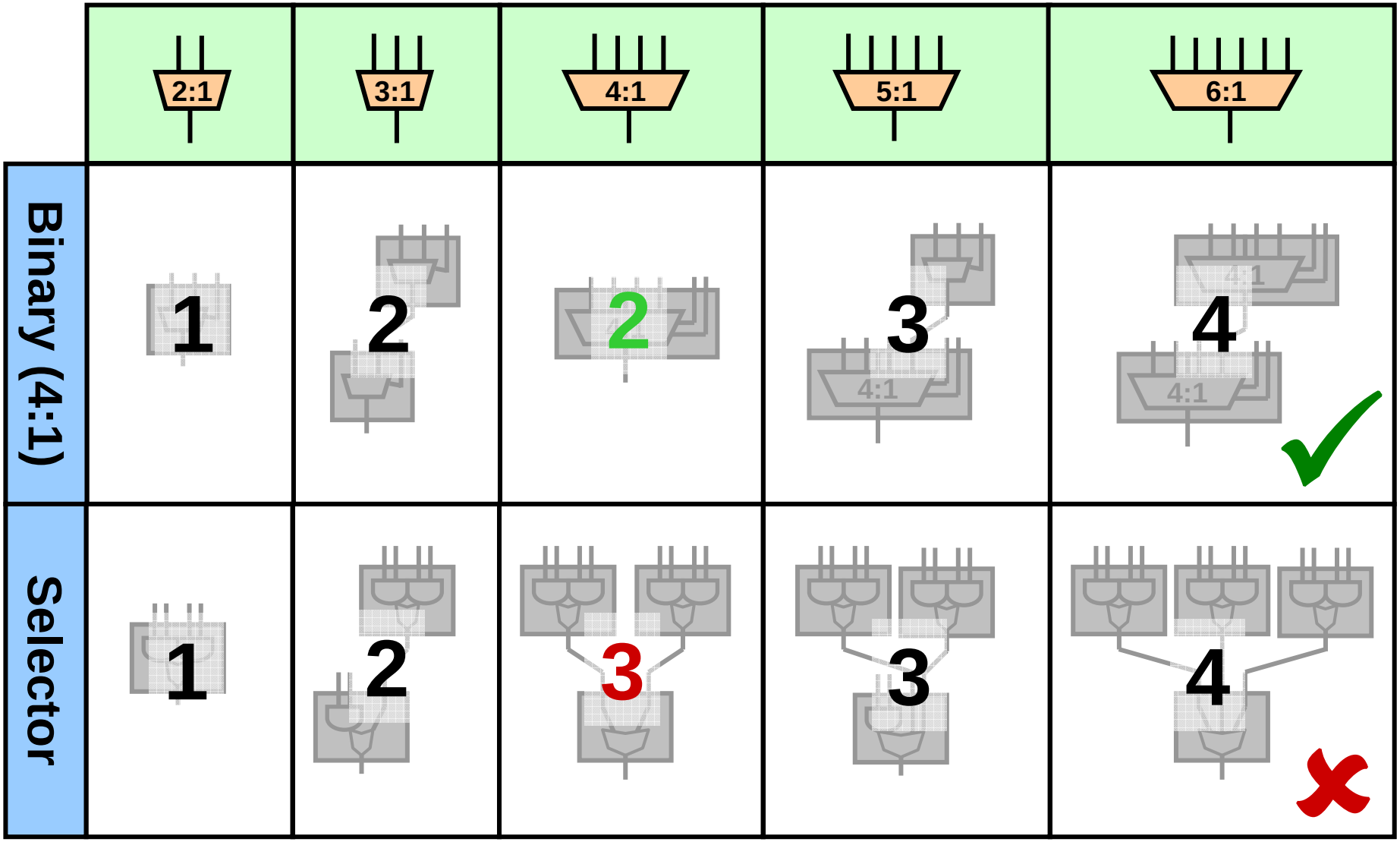
Efficient 4:1 Mux on Stratix: How it works



The Improved Cost of Binary Multiplexers



The Improved Cost of Binary Multiplexers

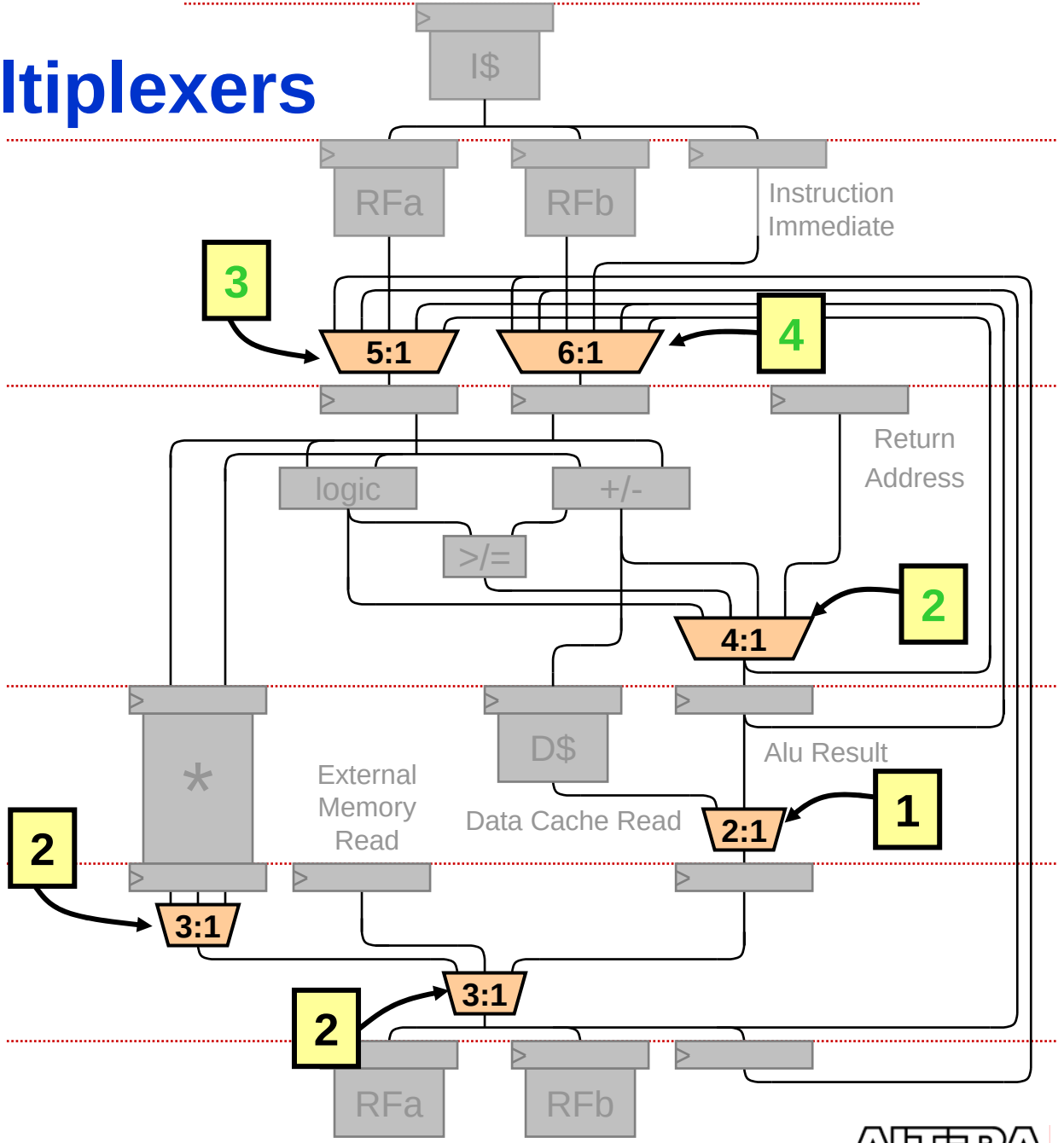


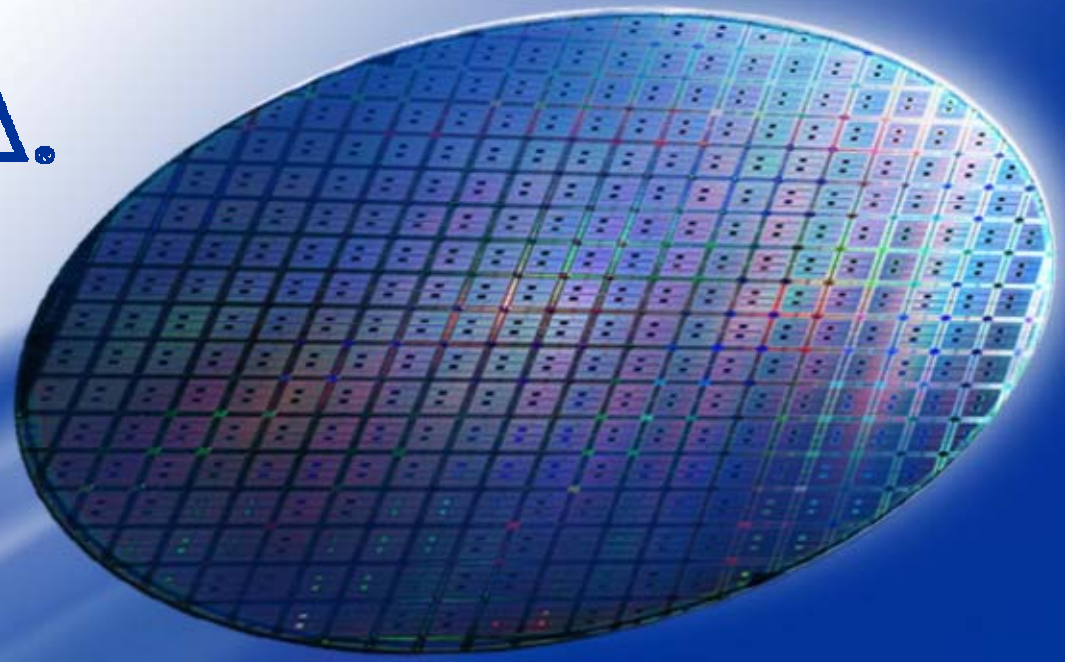
Efficient Multiplexers

544 LEs
(17 x 32bits)

-18%

448 LEs
(14 x 32bits)





Multiplexers in FPGA

Registered Multiplexers

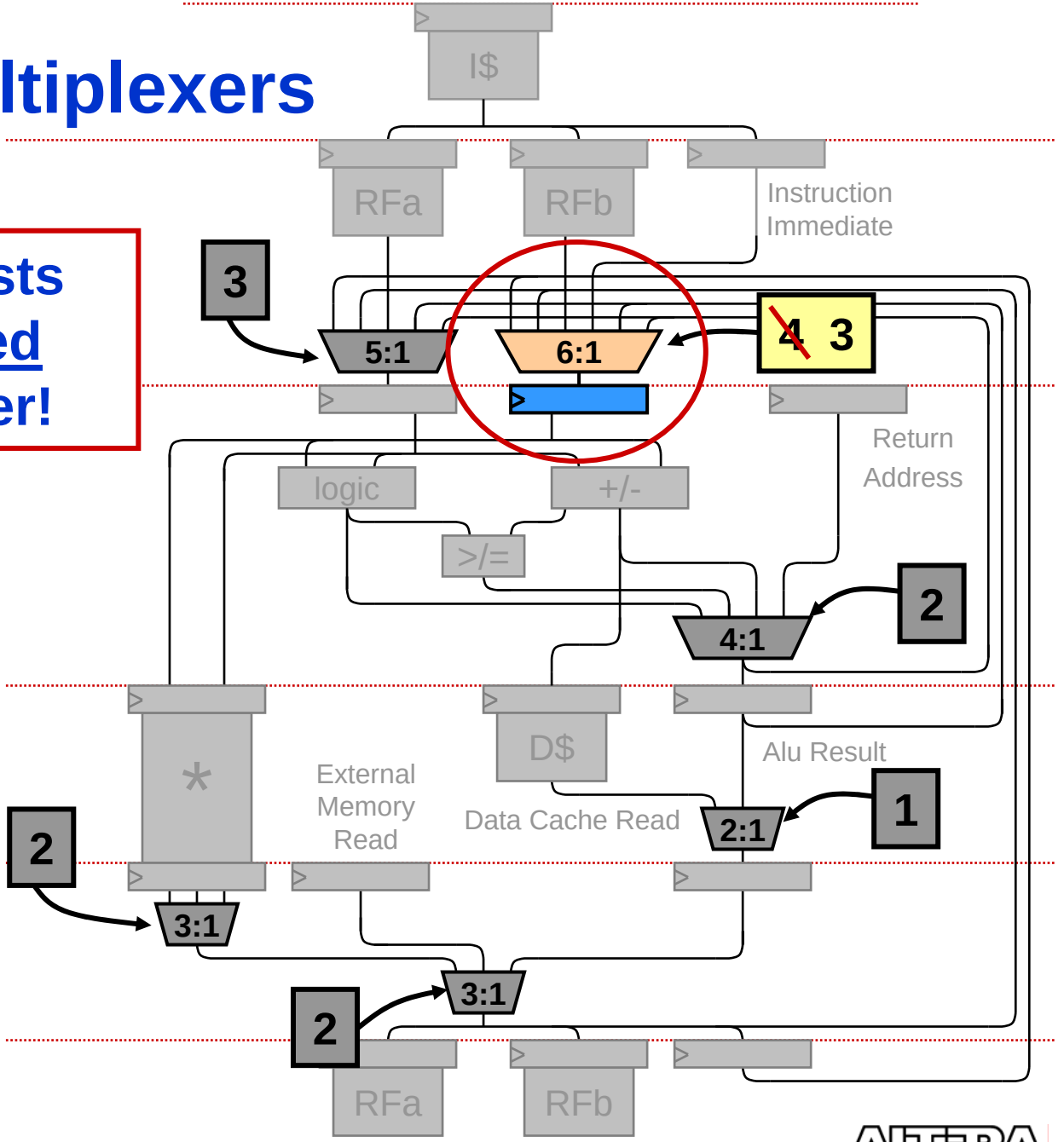
Efficient Multiplexers

Multiplexer costs can be reduced using a register!

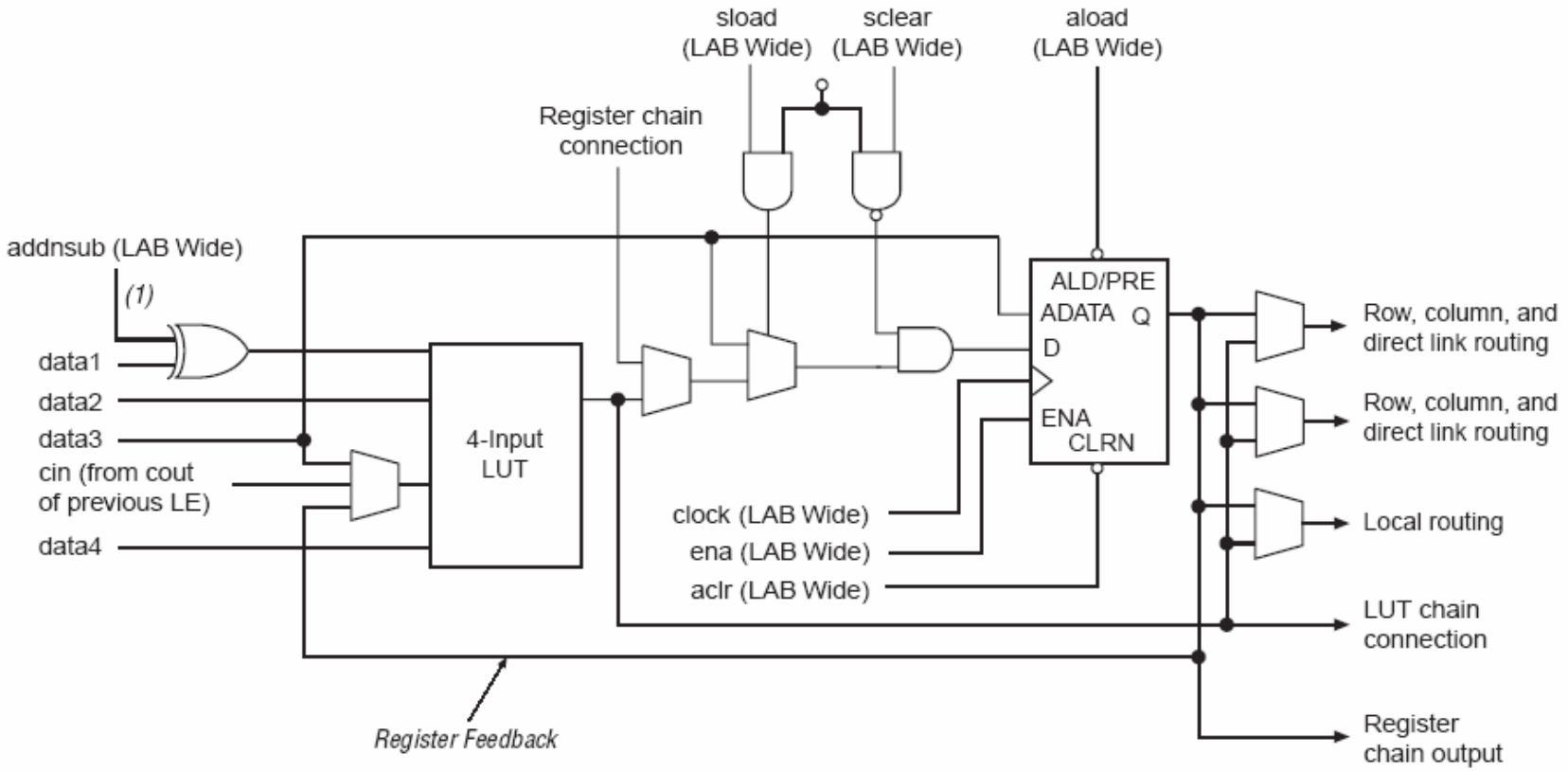
544 LEs
(17 x 32bits)

-24%

416 LEs
(13 x 32bits)



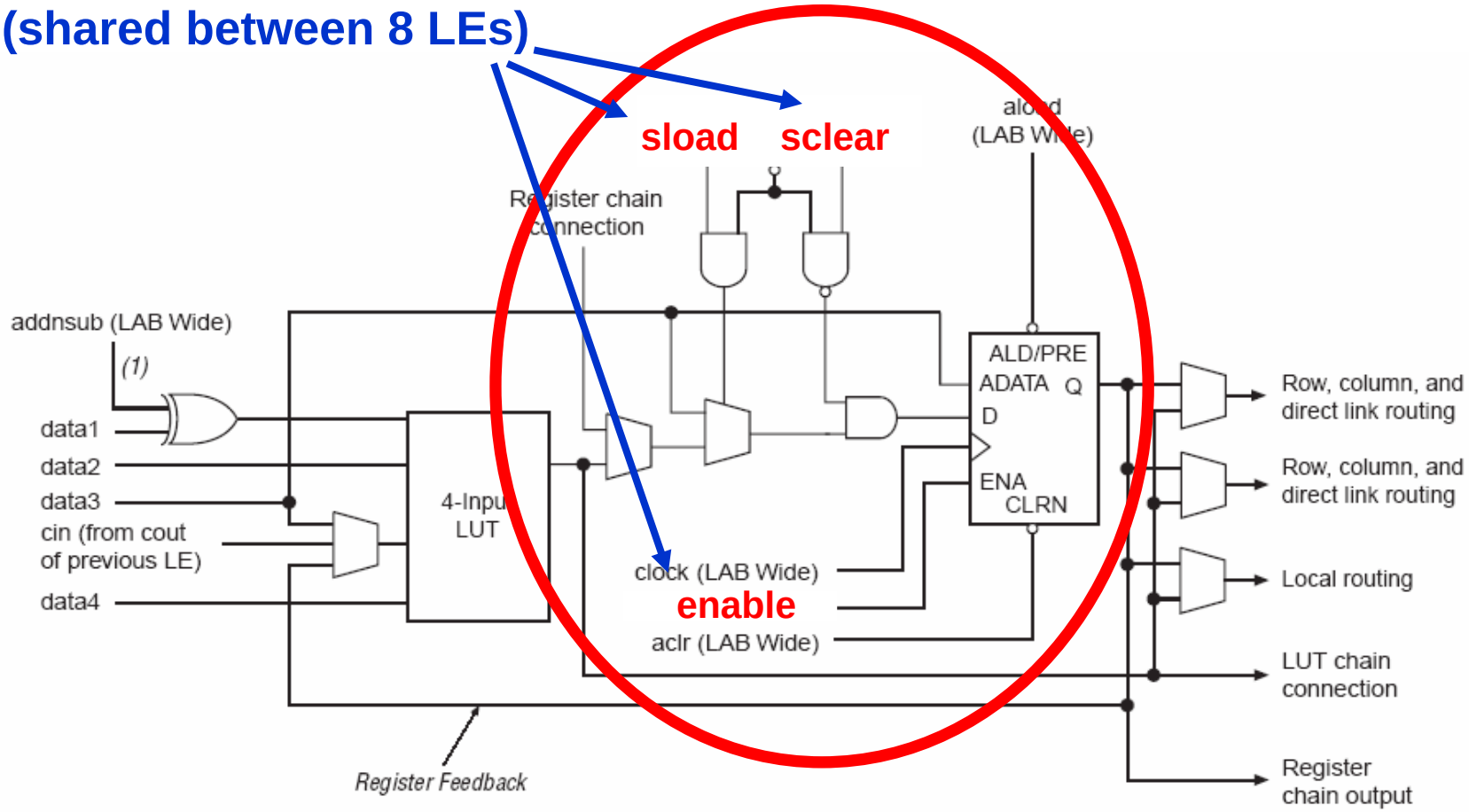
The Stratix LE



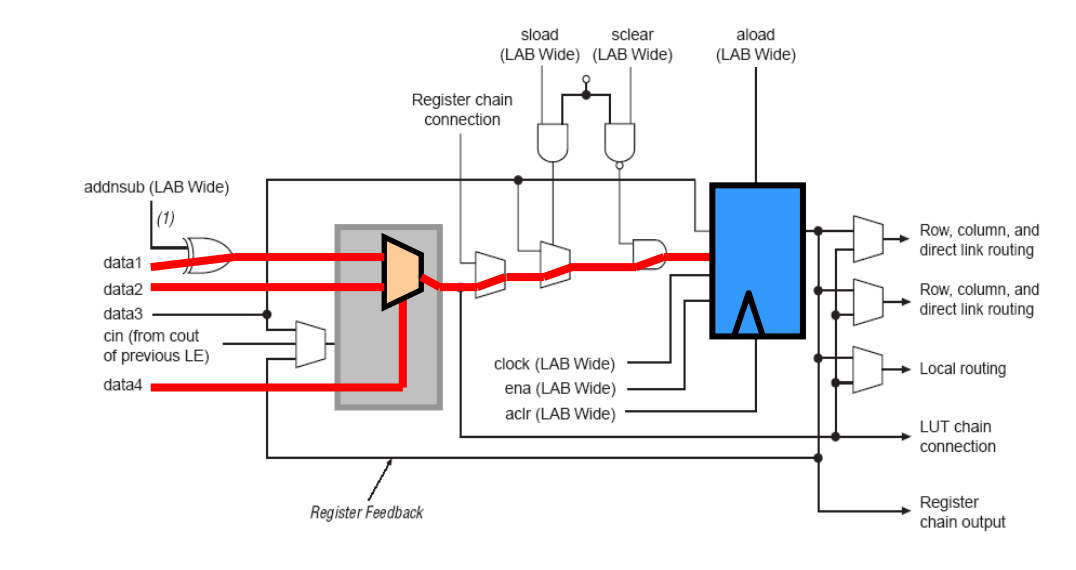
The Stratix LE

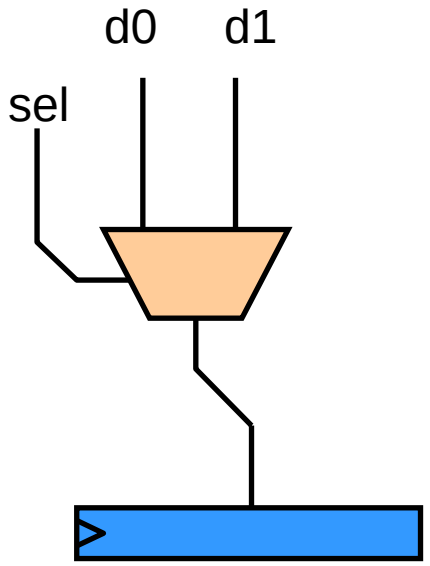
Additional Lab-wide signals

(shared between 8 LEs)

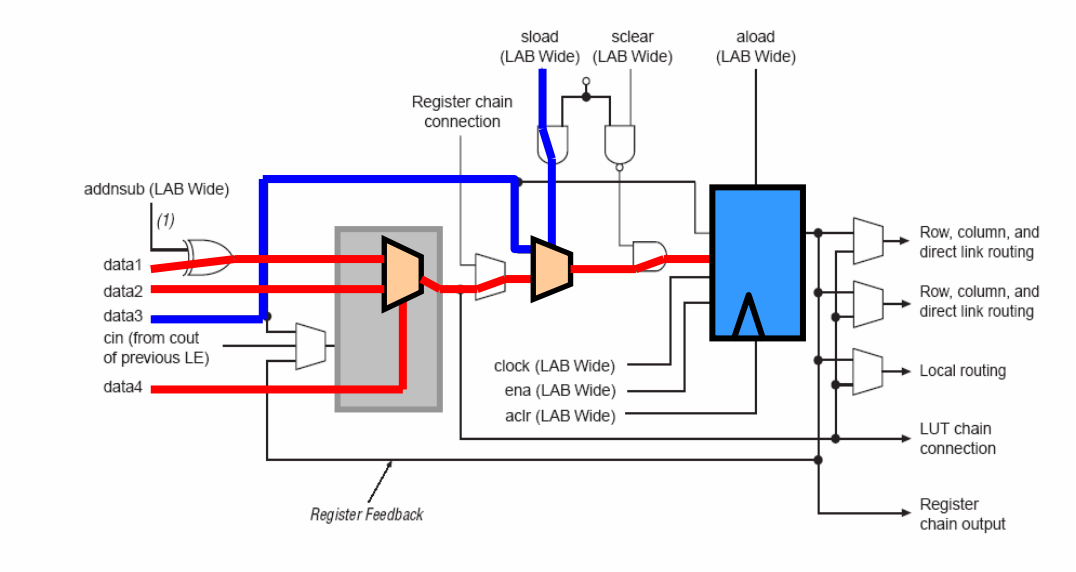


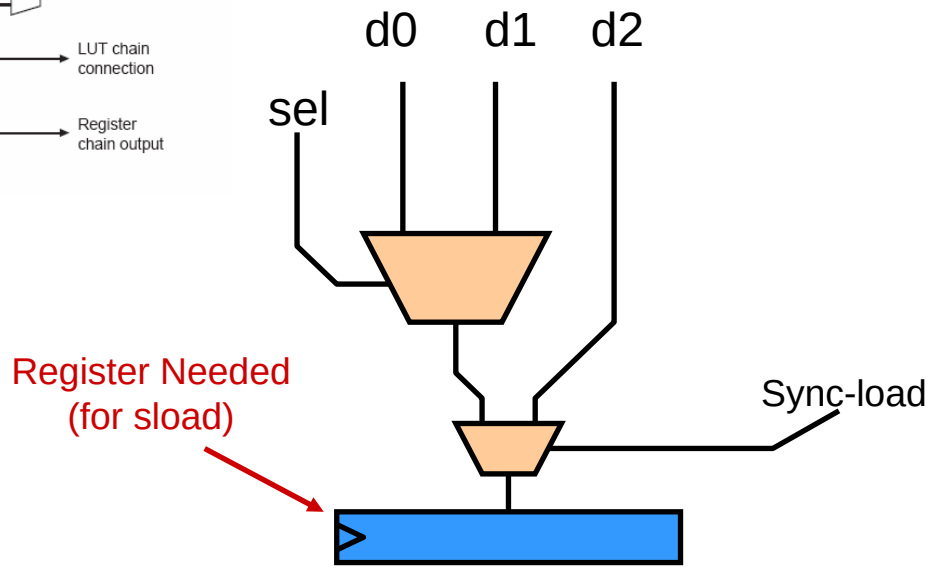
2:1 Mux in 1 LE



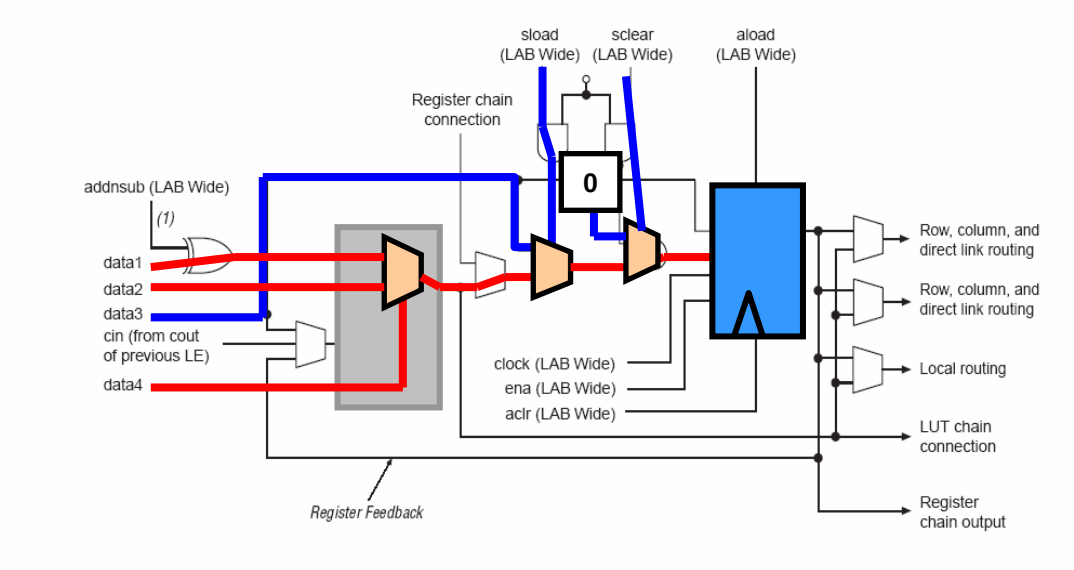


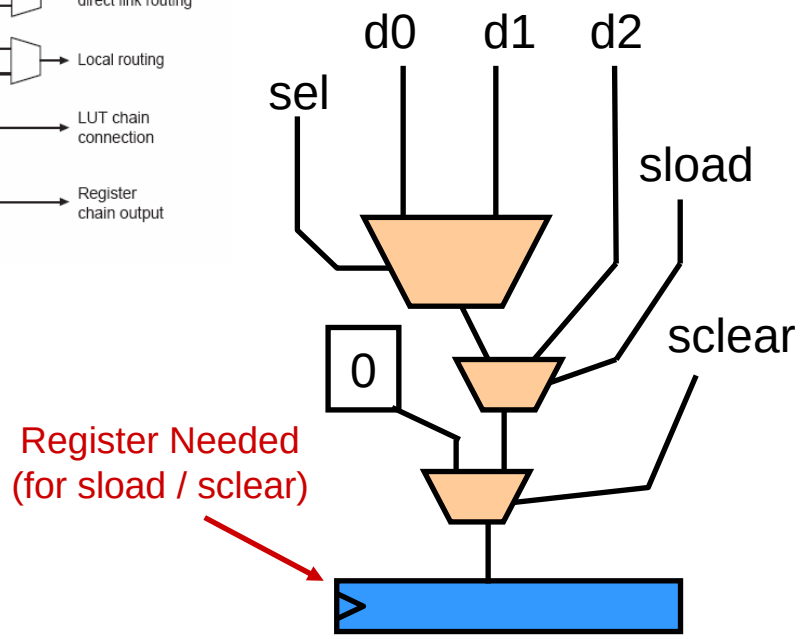
3:1 Mux in 1 LE





4:1 Mux in 1 LE

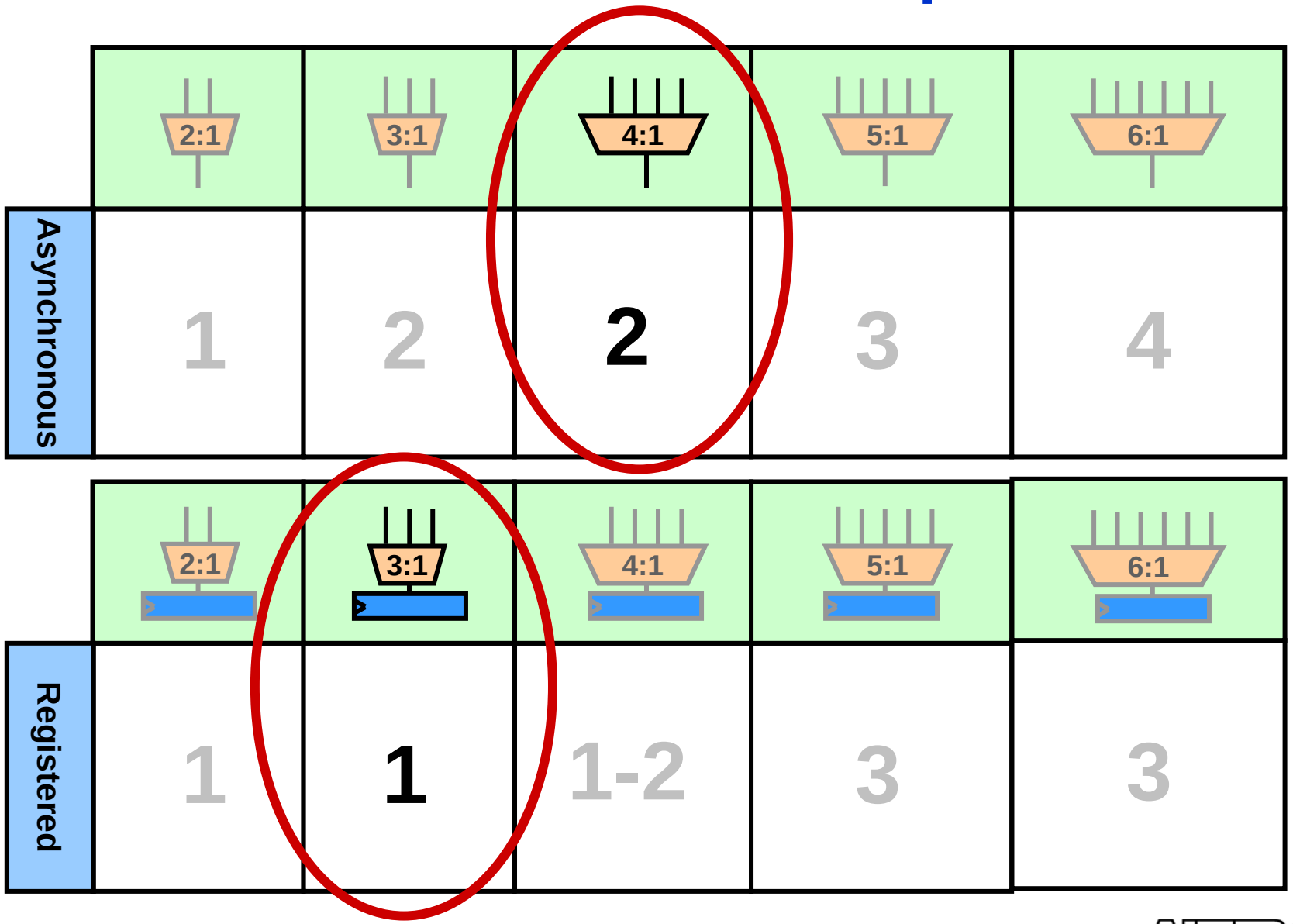




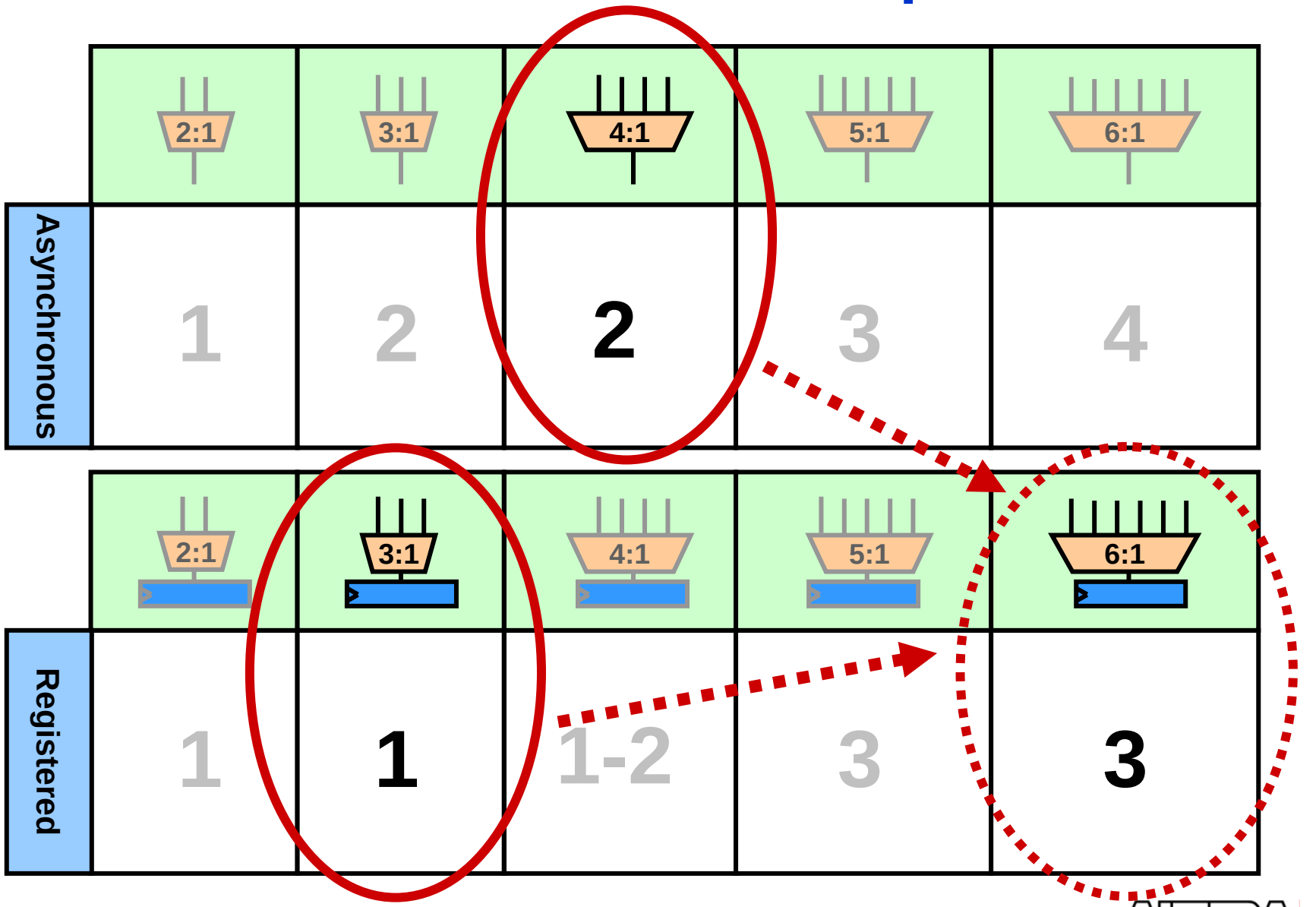
The Cost of Multiplexers

	2:1	3:1	4:1	5:1	6:1
Asynchronous	1	2	2	3	4
Registered	1	1	1-2	3	3

The Most Cost Effective Multiplexers



The Most Cost Effective Multiplexers



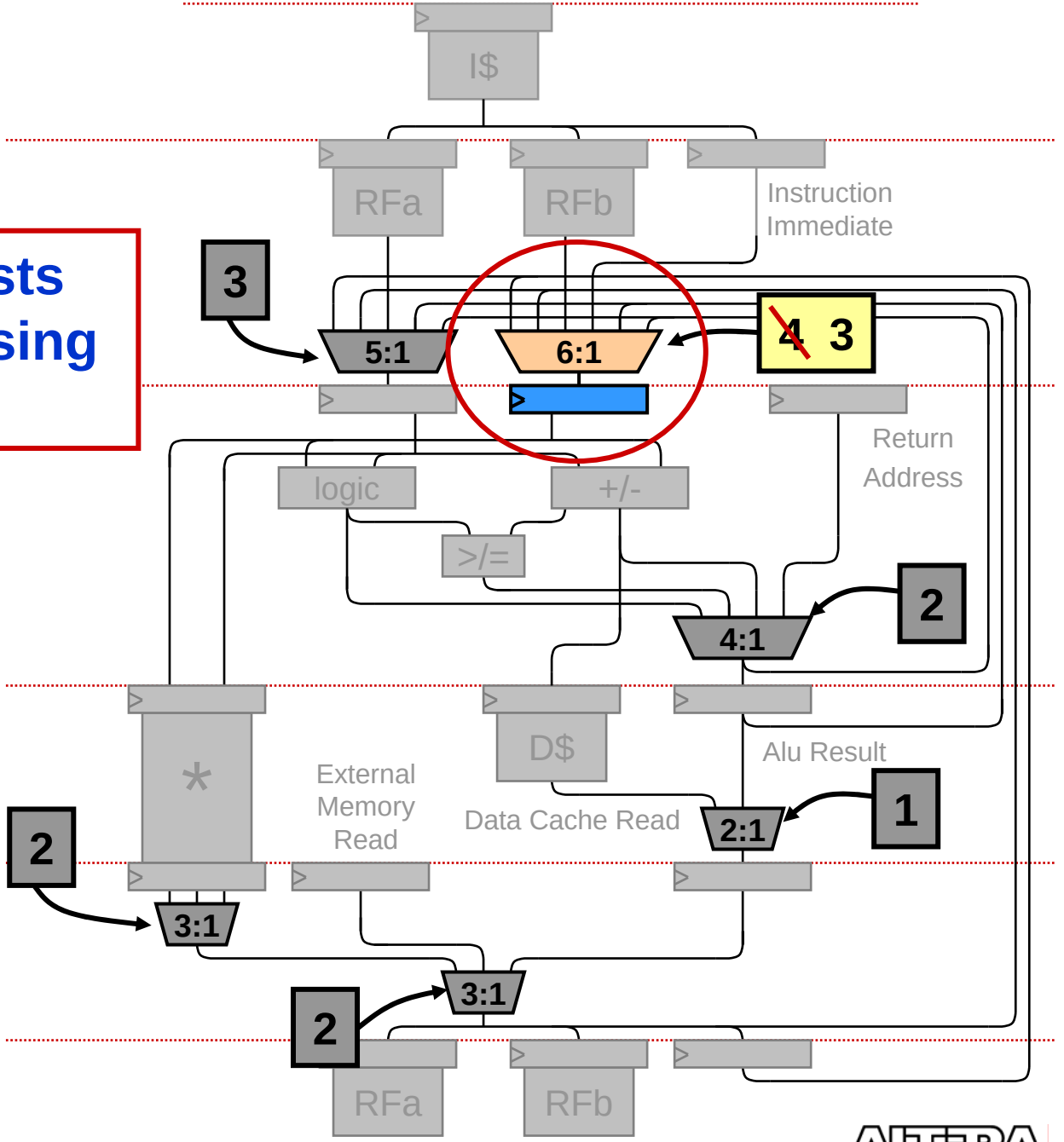
Recap:

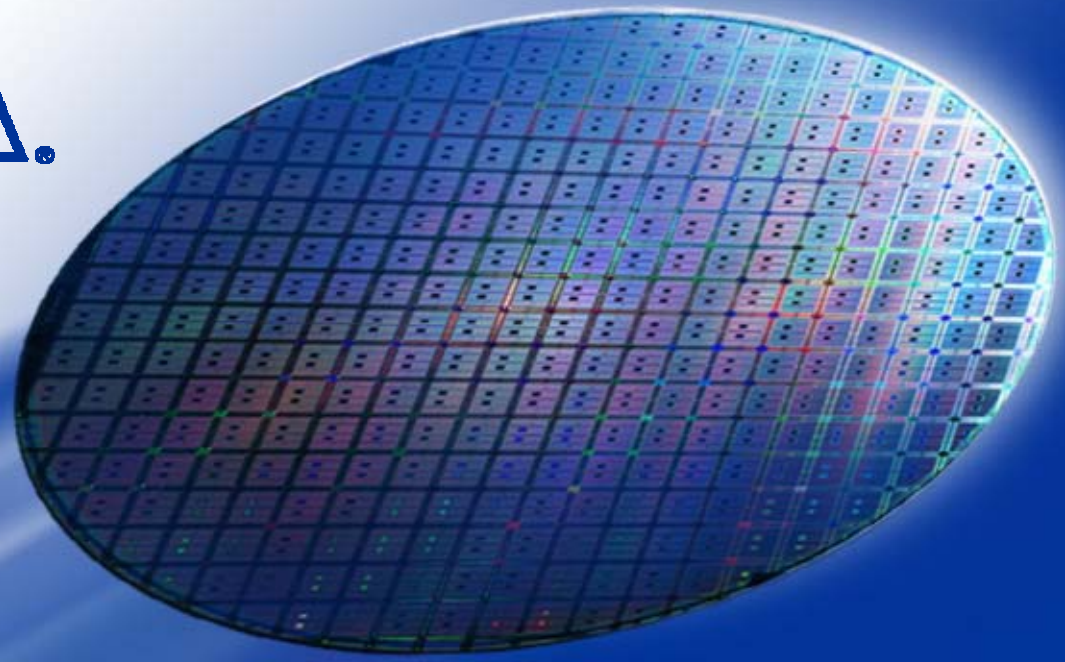
Multiplexer costs were reduced using a register!

544 LEs
(17 x 32bits)

-24%

416 LEs
(13 x 32bits)

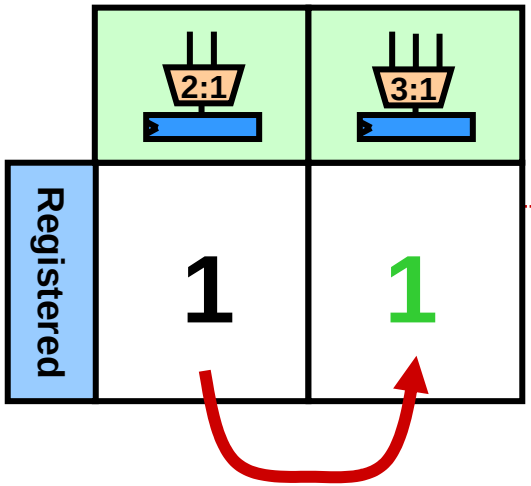




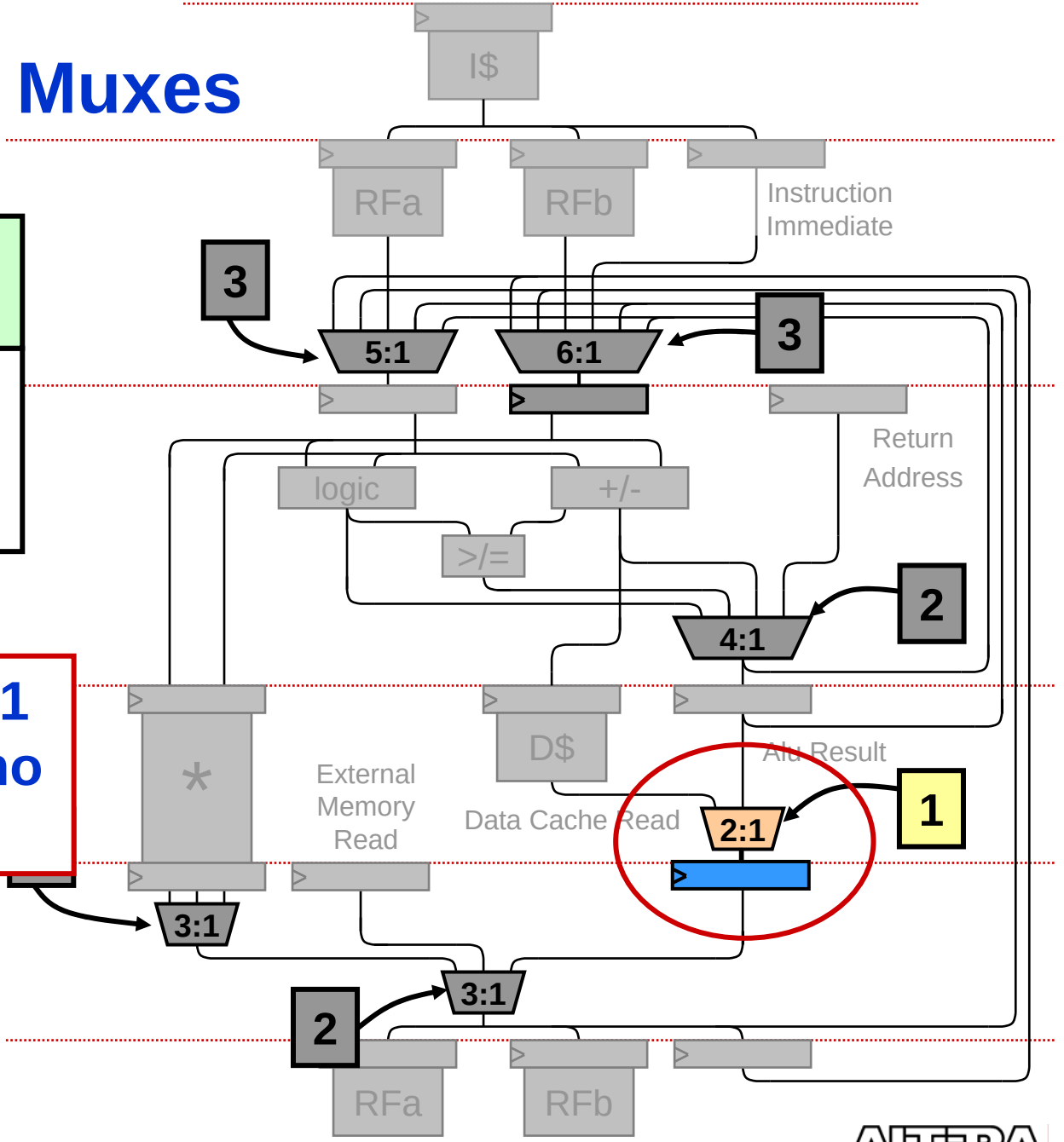
Optimizing Multiplexers in Nios II

Restructuring Techniques

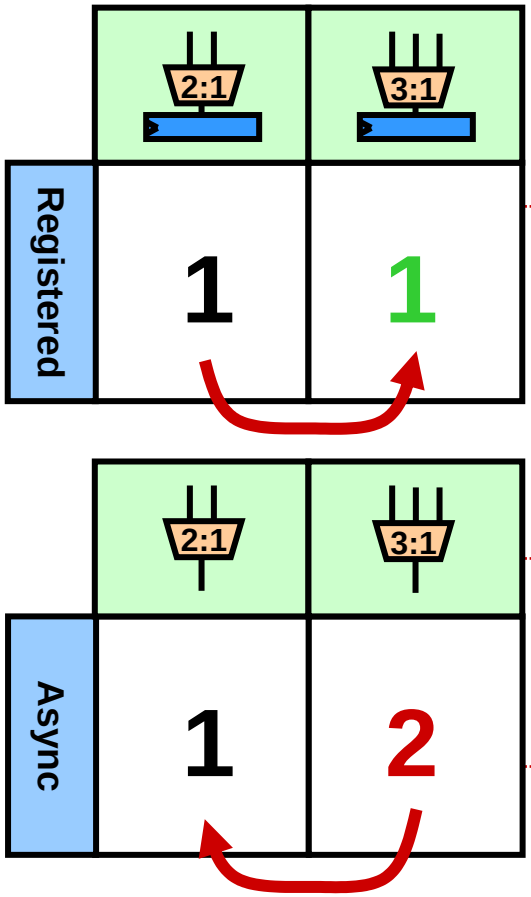
Underutilized Muxes

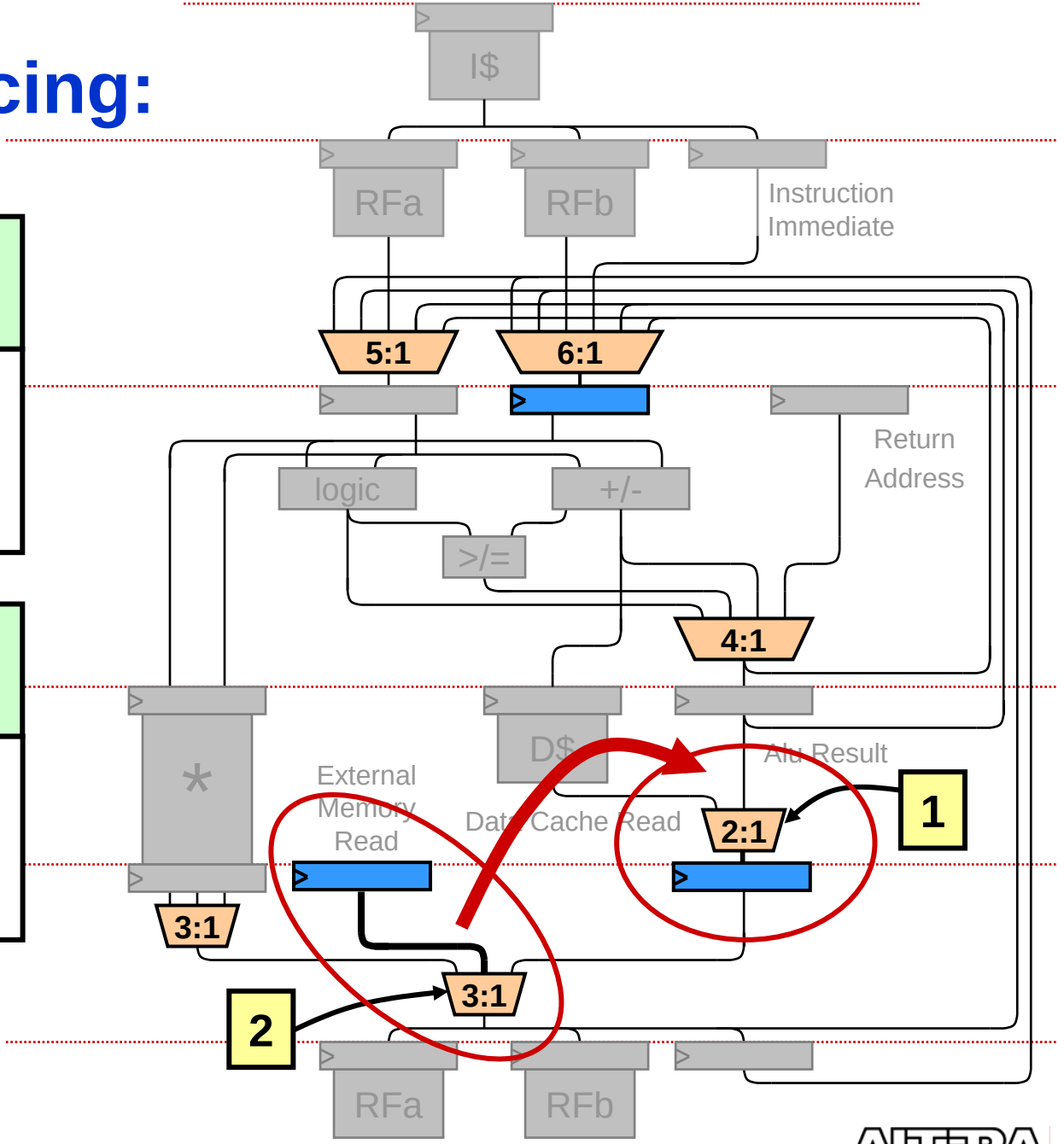


Can extend 2:1 to be a 3:1 at no extra cost!



Input Balancing:



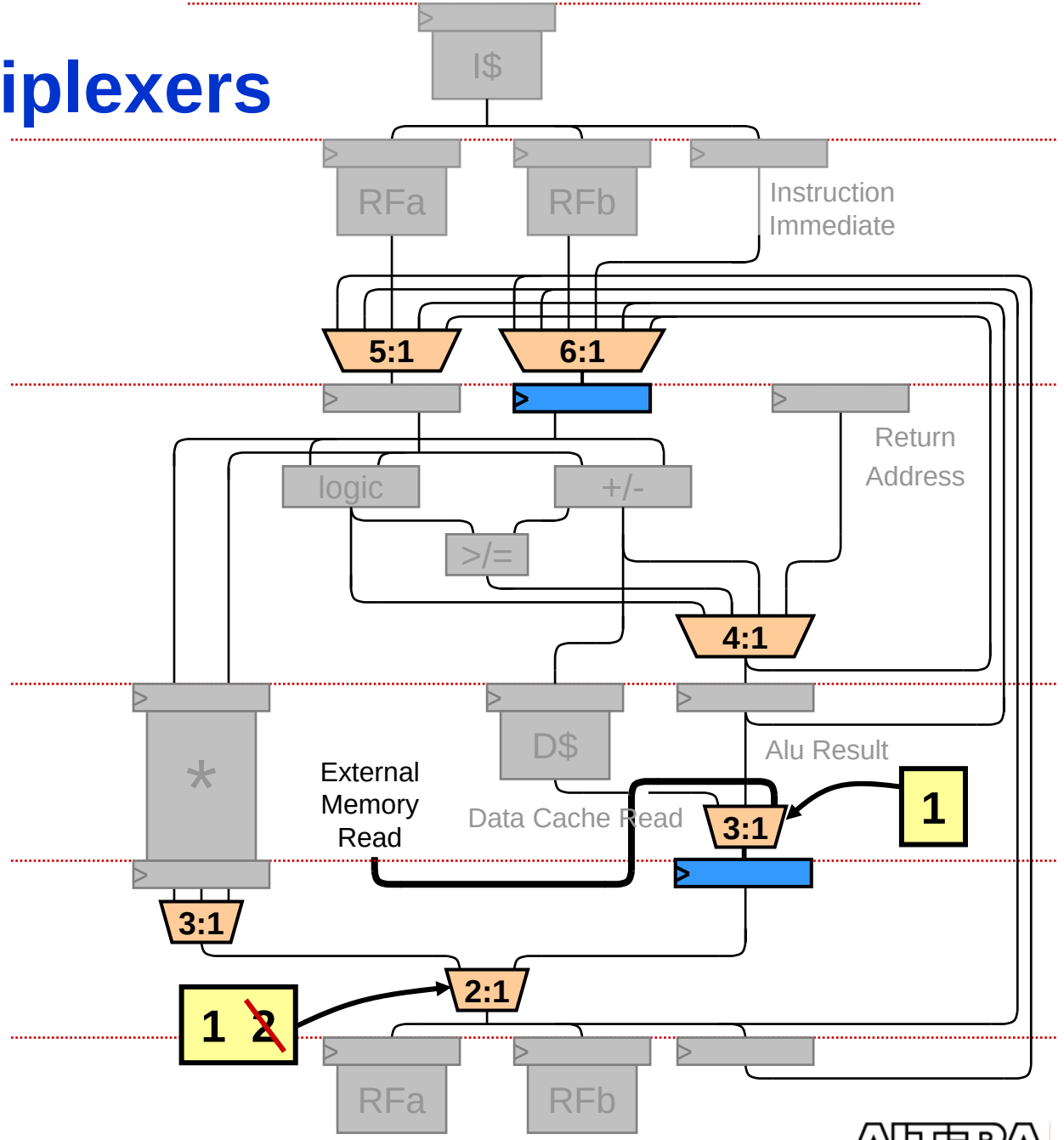


NIOS II Multiplexers

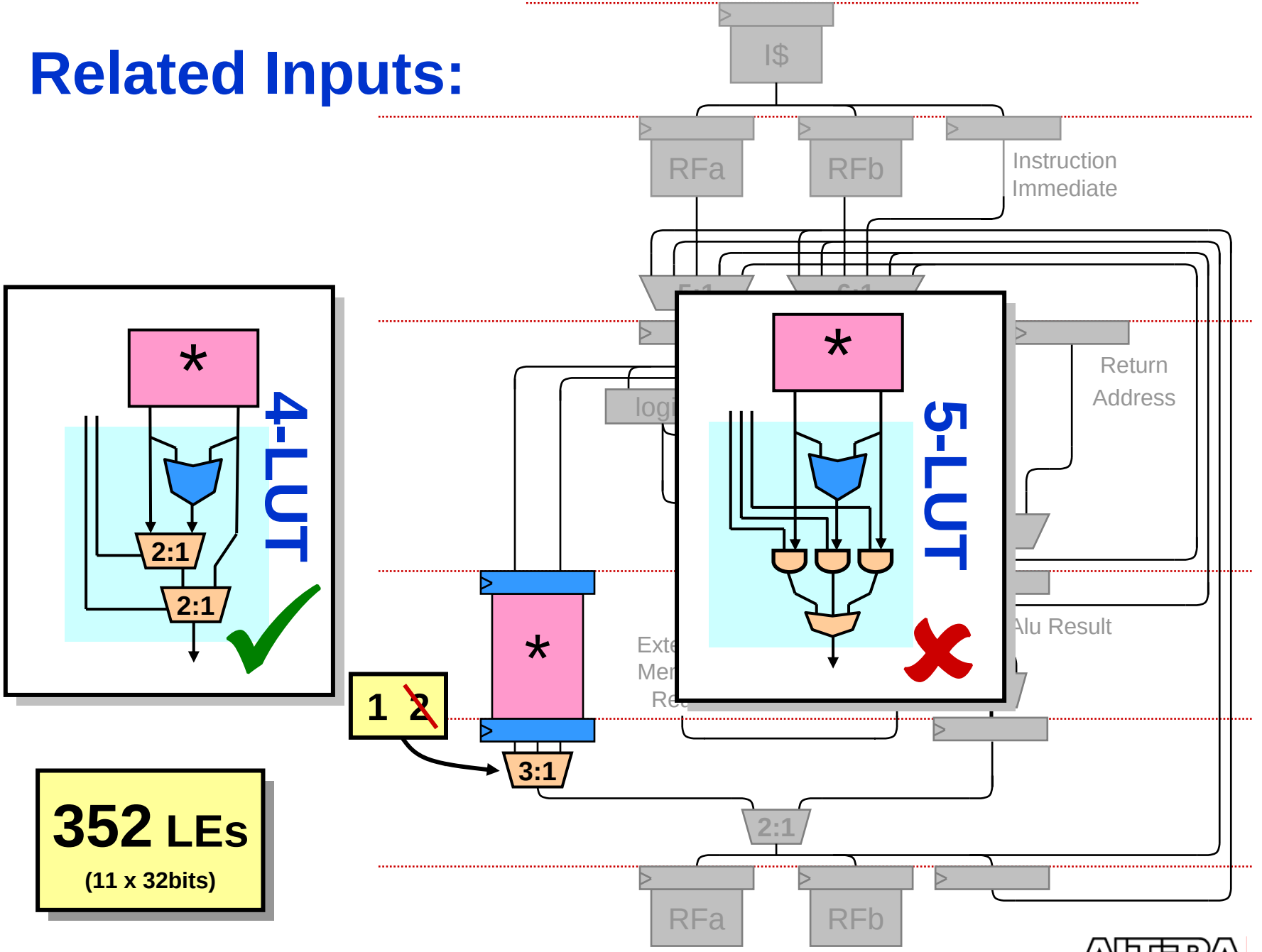
416 LEs
(13 x 32bits)

-8%

384 LEs
(12 x 32bits)



Related Inputs:

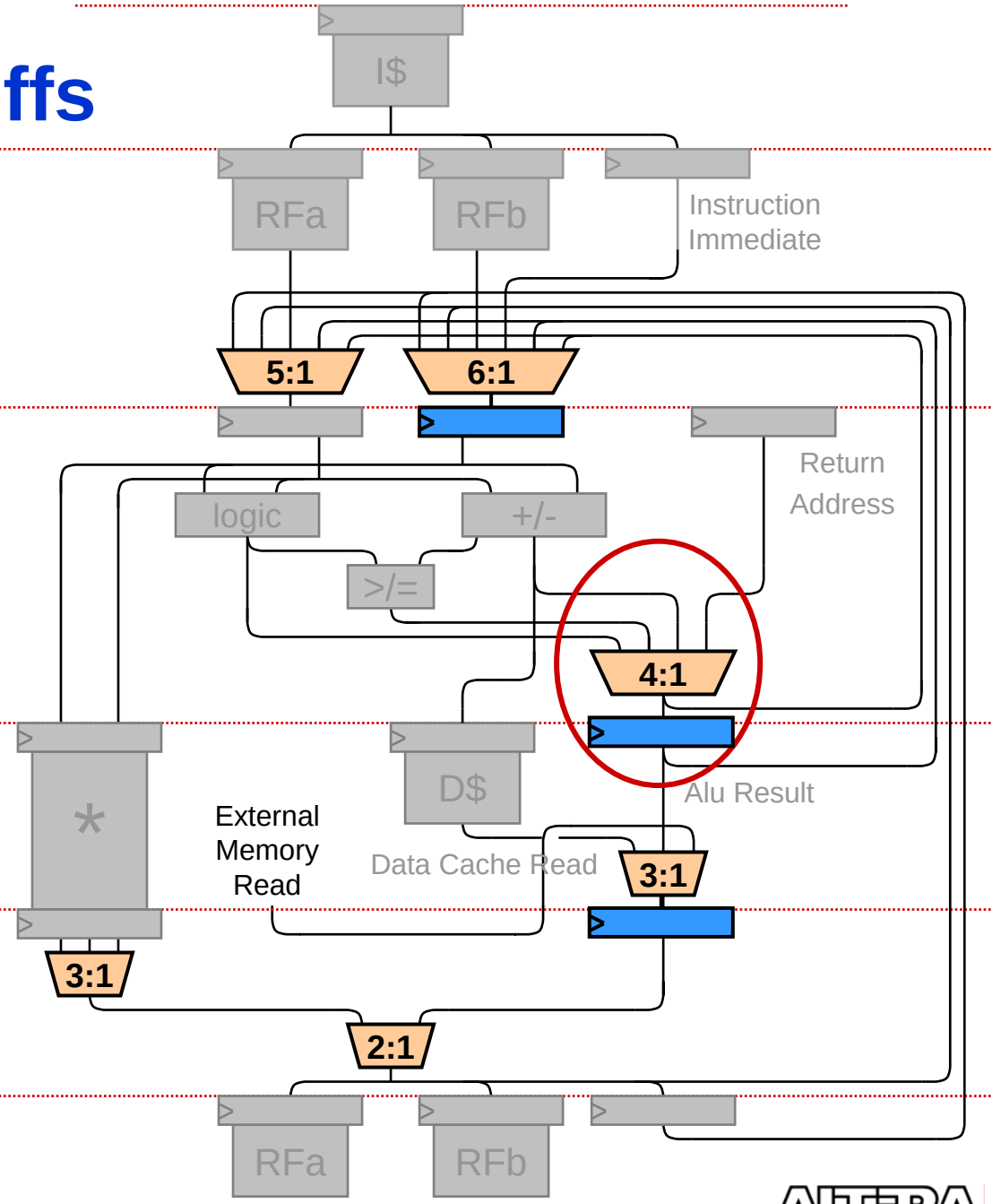


352 LEs
(11 x 32bits)

Design Trade-offs

	cycles
CALL	3
TRAP	3
INTR	3
BREAK	3

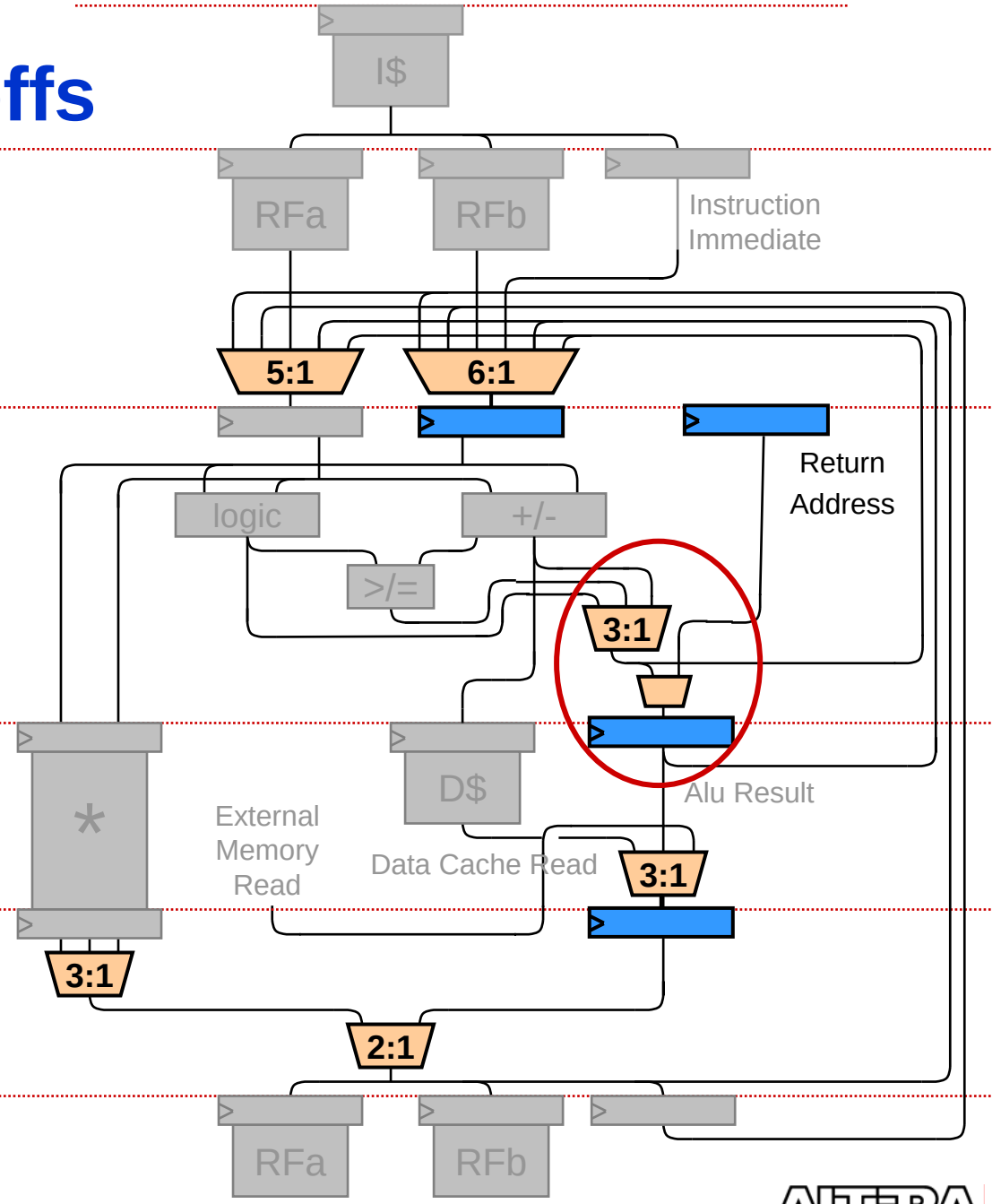
No need to Forward Return Address Early



Design Trade-offs

	cycles
CALL	3
TRAP	3
INTR	3
BREAK	3

No need to Forward Return Address Early



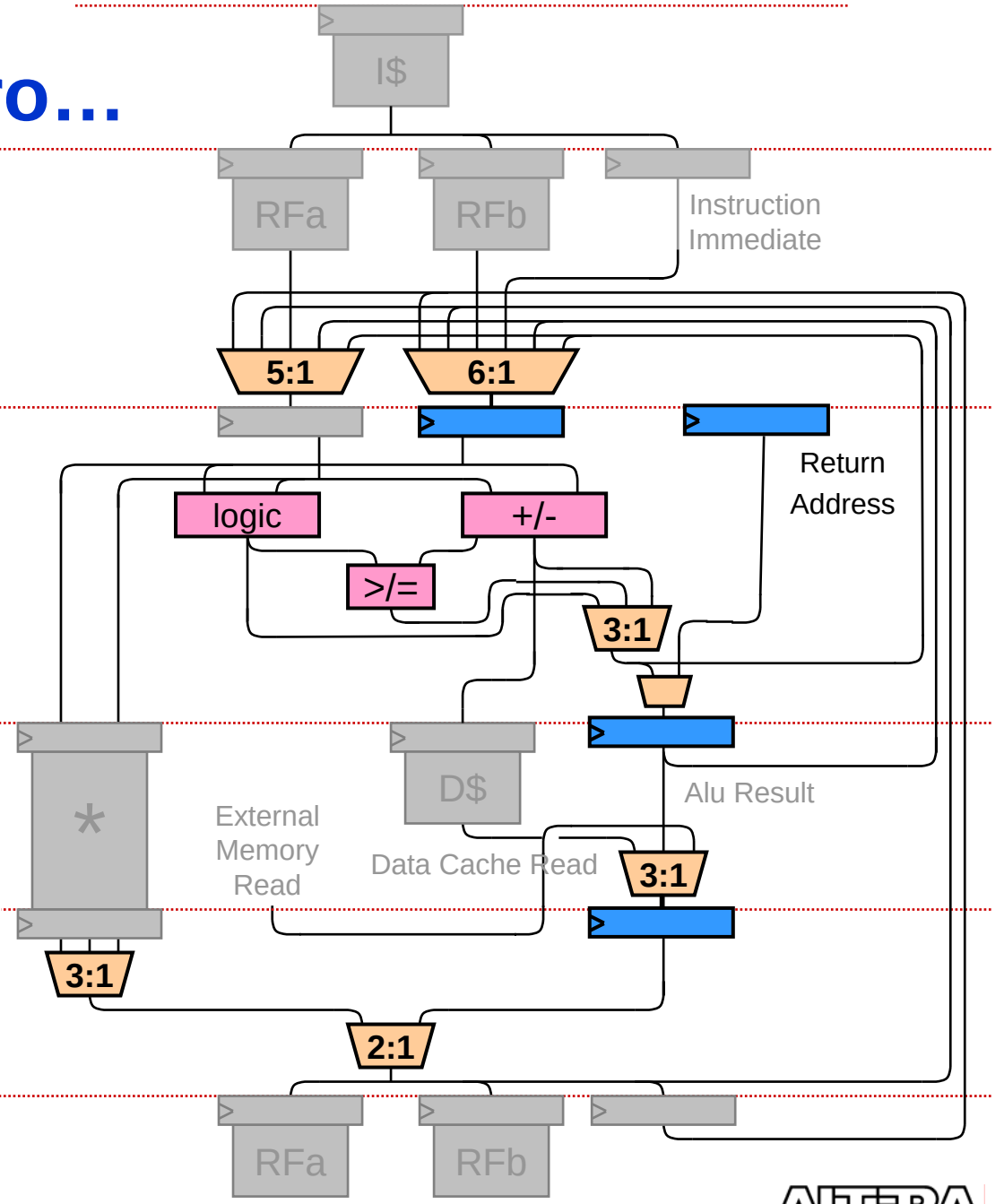
Forwarding Zero…

CMP.op r3, r2, r1

IF (**r2 op r1**)
THEN **R3** = **0x00000001**
ELSE **R3** = **0x00000000**

Mostly 0's

Can use Synchronous Reset instead of multiplexer input.



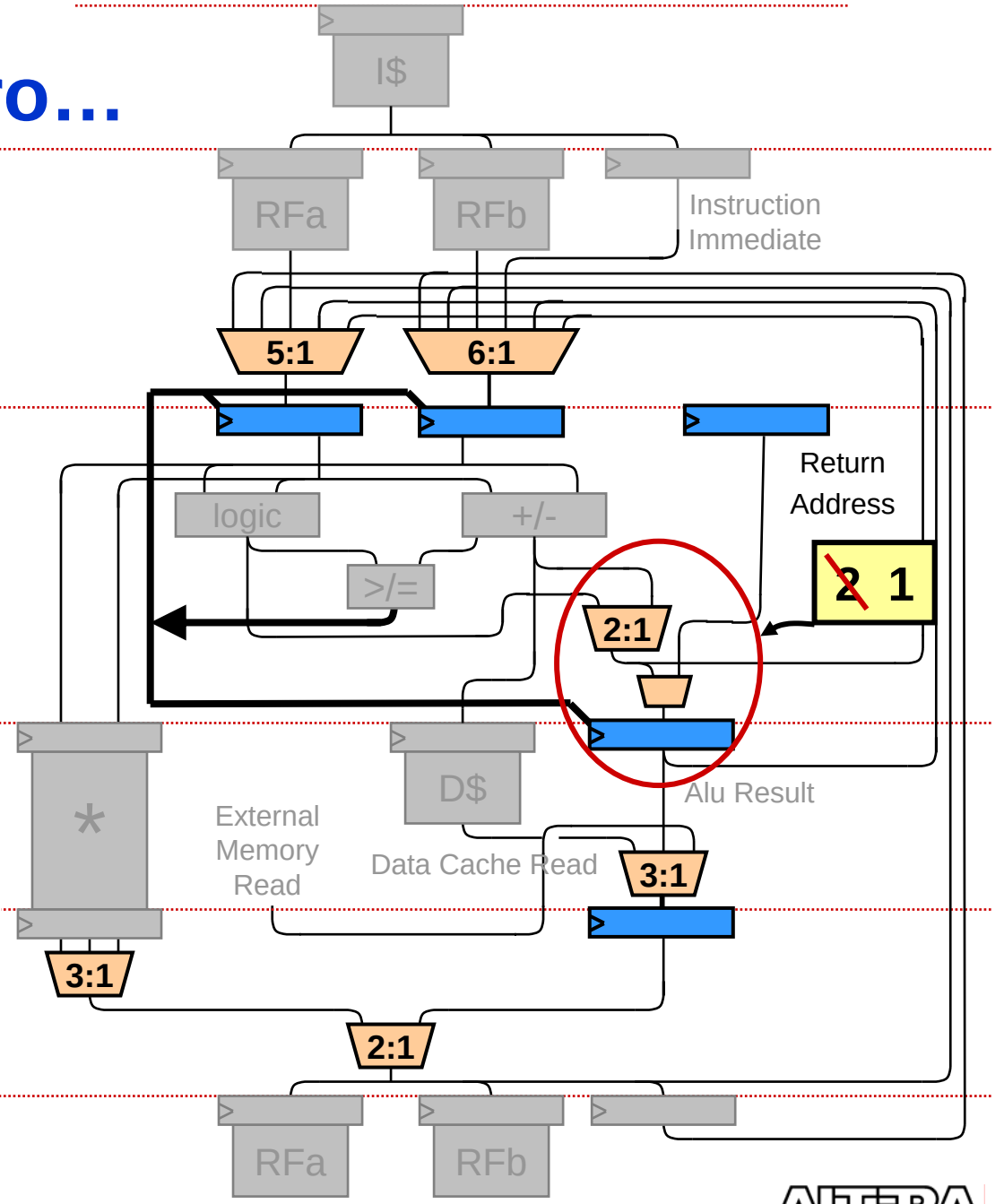
Forwarding Zero…

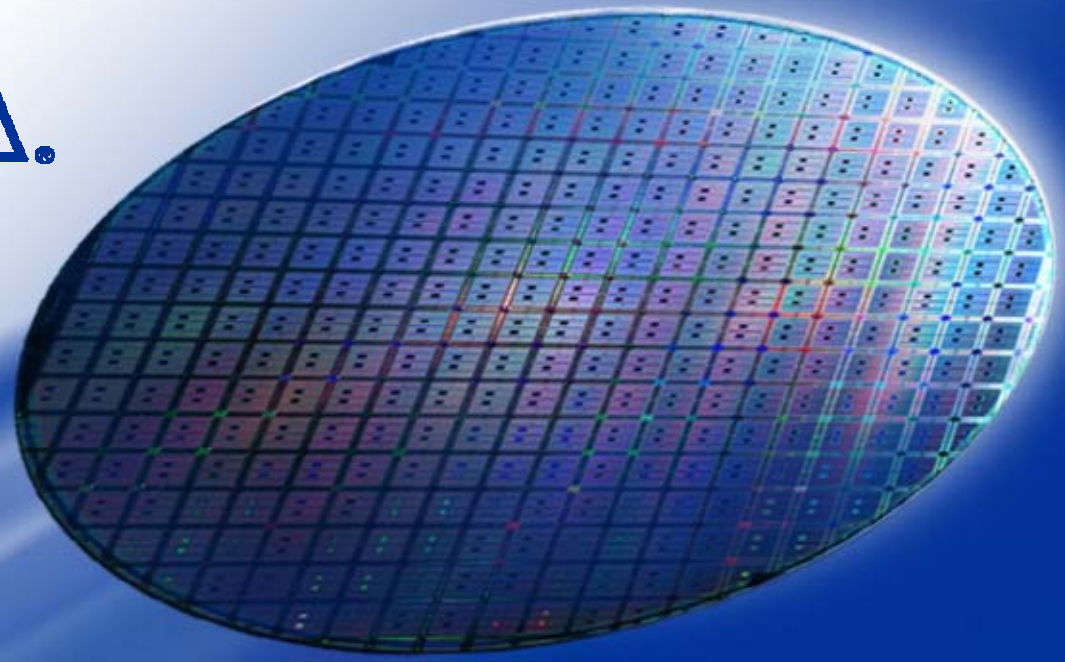
CMP.op r3, r2, r1

IF (**r2 op r1**)
THEN **R3** = **0x00000001**
ELSE **R3** = **0x00000000**

Mostly 0's

Can use Synchronous Reset instead of multiplexer input.

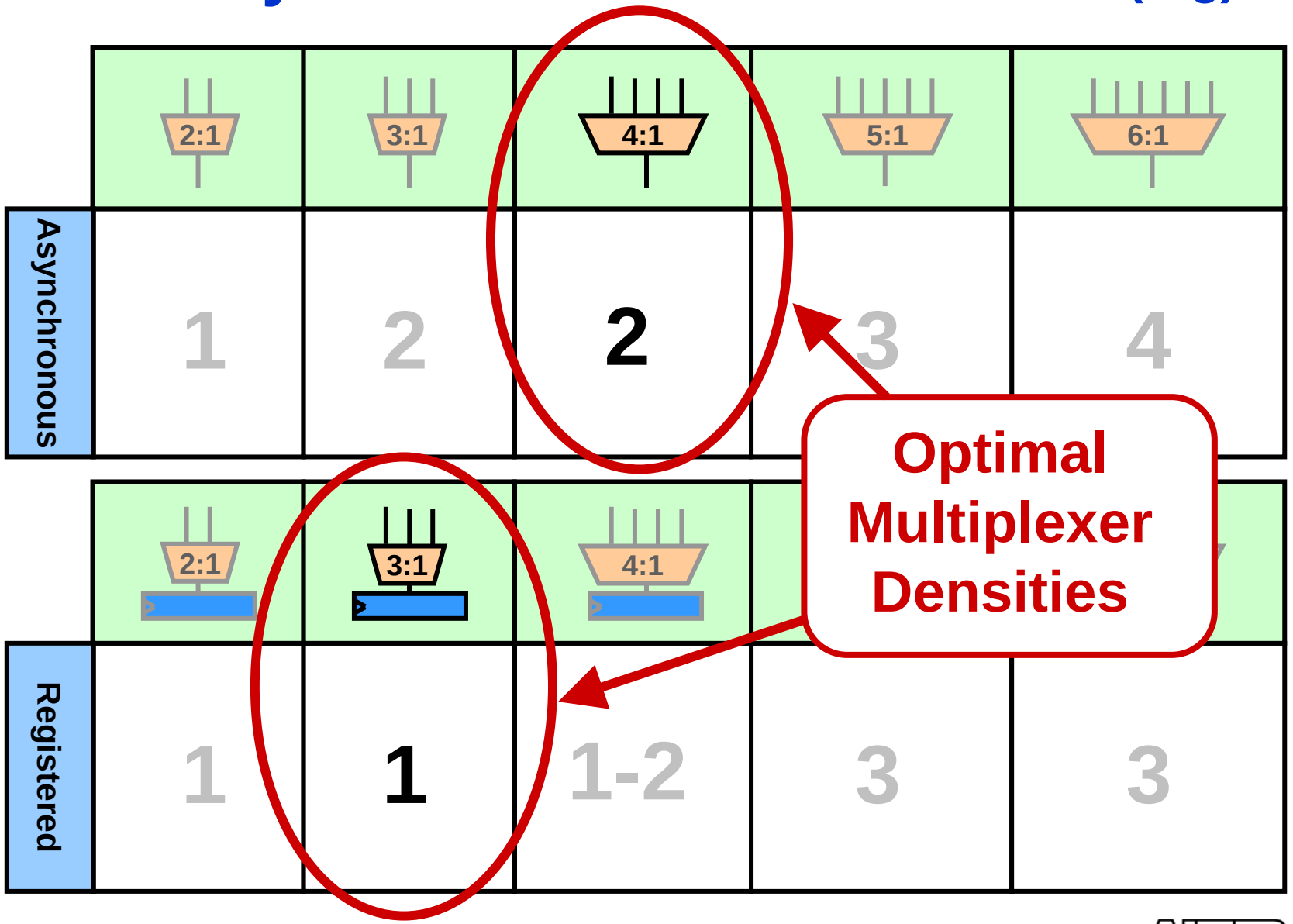




Optimizing Multiplexers in FPGA

Summary

Summary: Restructure to 4:1 or 3:1(reg)

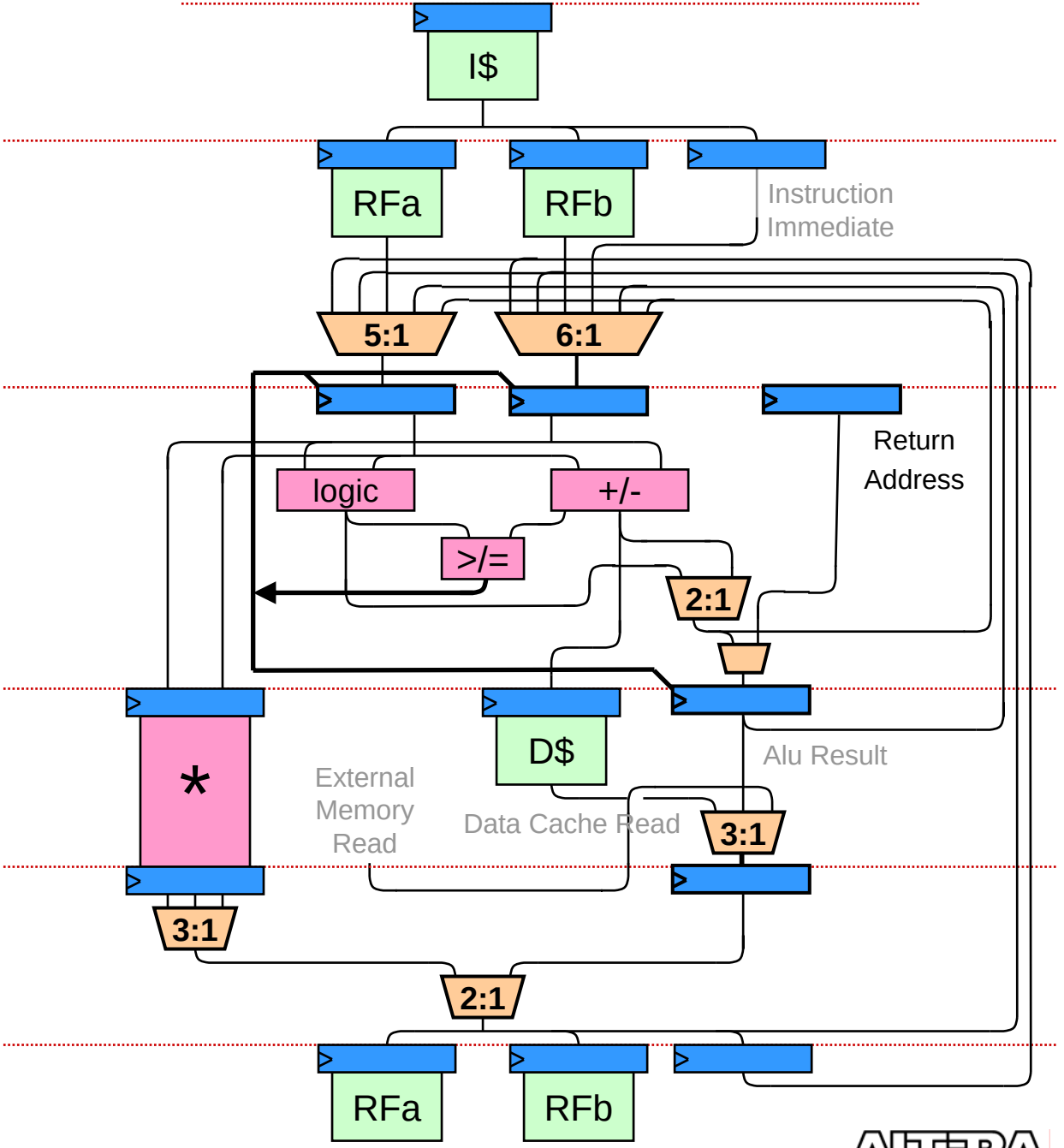


Summary

544 LEs
(17 x 32bits)

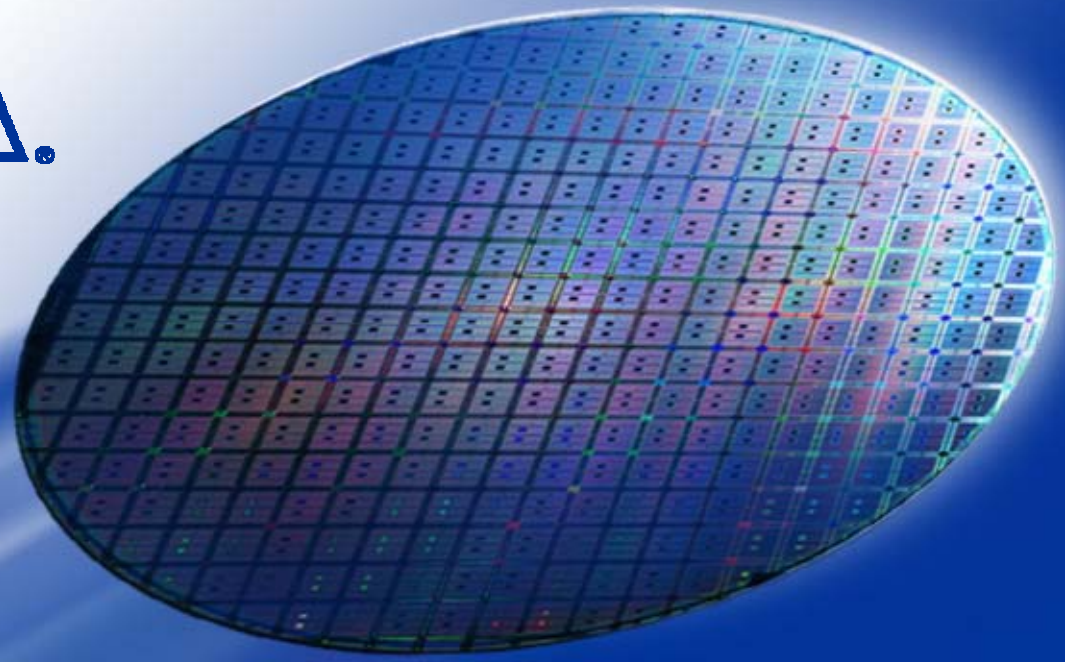
- 42%

320 LEs
(10 x 32bits)



Techniques Extend to Real Designs…

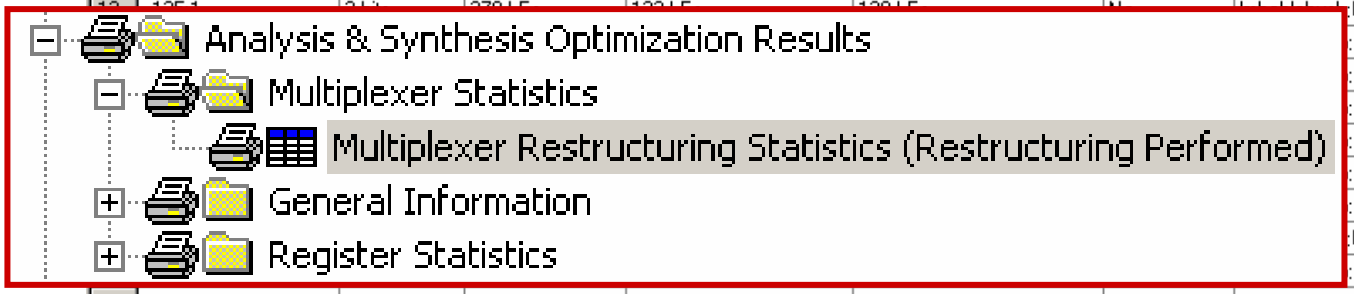
	Original		Optimized	
	Size	Speed	Size	Speed
A	2,400	40 MHz	-50%	2.5x
B	7,373	77 MHz	-77%	2.0x
C	13,500	50 MHz	1c12 fit	1.5x
D	13,472	67 MHz	-60%	unchng
E	1,925	75 MHz	-27%	unchng
Others	…	…	…	…



Optimizing Multiplexers in FPGA

Support in Quartus Synthesis

New Multiplexer Report:



Multiplexer Restructuring Statistics (Restructuring Performed)

	Multiplexer Inputs	Bus Width	Baseline Area	Area if Restructured	Saving if Restructured	Registered	Example Multiplexer Output
1	143:1	9 bits	855 LEs	405 LEs	450 LEs	No	|mbcb|cbusb:U0|dto[24]~56
2	2:1	224 bits	224 LEs	0 LEs	224 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalme:stupealmc|almmsk0[1]
3	2:1	224 bits	224 LEs	0 LEs	224 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalms:stupdalmc|almmsk2[18]
4	2:1	192 bits	192 LEs	0 LEs	192 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalme:stupealmc|almmsk1[0]
5	2:1	192 bits	192 LEs	0 LEs	192 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalms:stupfalmc|almmsk3[16]
6	34:1	7 bits	154 LEs	7 LEs	147 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupfc|ppflgi[4]
7	34:1	7 bits	154 LEs	7 LEs	147 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupgc|ppflgi[6]
8	34:1	7 bits	154 LEs	7 LEs	147 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupec|ppflgi[2]
9	34:1	7 bits	154 LEs	7 LEs	147 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupfc|ppflgi[0]
10	34:1	7 bits	154 LEs	7 LEs	147 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupfc|ppflgi[5]
11	34:1	7 bits	154 LEs	7 LEs	147 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupfc|ppflgi[1]
							U0|dto[29]~61
							:u1|datadn:datadncont|dndatac:dataoutcont|actboutdt[5]
							:u1|datadn:datadncont|dndatac:dataoutcont|matedndt[3]
							:u1|stup:stupcont|stupalms:stupcalmc|almmsk0[35]
							:u1|stup:stupcont|stupalms:stupaalmc|almmsk2[36]
							:u1|stup:stupcont|stupalms:stupcalmc|almmsk1[32]
							:u1|stup:stupcont|stupalms:stupcalmc|almmsk3[32]
							U0|dto[25]~57
							:u1|datadn:datadncont|dndatac:dataoutcont|actboutdt[4]
21	136:1	2 bits	180 LEs	90 LEs	90 LEs	No	|mbcb|cbusb:U0|dto[28]~60
22	17:1	7 bits	77 LEs	0 LEs	77 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupec|ppflgi[3]
23	17:1	7 bits	77 LEs	0 LEs	77 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupbc|ppflgi[7]
24	64:1	2 bits	84 LEs	8 LEs	76 LEs	Yes	|mbcb|serialb:u1|datadn:datadncont|dndatac:dataoutcont|matedndt[5]
25	148:1	4 bits	392 LEs	328 LEs	64 LEs	No	|mbcb|cbusb:U0|dto[8]~40
26	2:1	56 bits	56 LEs	0 LEs	56 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalms:stupbalmc|almbuf[5]
27	2:1	56 bits	56 LEs	0 LEs	56 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalmg:stupgalmc|almbuf[10]
28	2:1	56 bits	56 LEs	0 LEs	56 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalms:stupcalmc|almbuf[20]
29	2:1	56 bits	56 LEs	0 LEs	56 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalme:stupealmc|almbuf[30]

(Table is always produced after Analysis & Synthesis, *even if optimizations are disabled*)

New Multiplexer Report:

Multiplexer Restructuring Statistics (Restructuring Performed)

	Multiplexer Inputs	Bus Width	Baseline Area	Area if Restructured	Saving if Restructured	Registered	Example Multiplexer Output
1	143:1	9 bits	855 LEs	405 LEs	450 LEs	No	|mbcb|cbusb:U0|dto[24]~56
2	2:1	224 bits	224 LEs	0 LEs	224 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalme:stupealmc|almmsk0[1]
3	2:1				224 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalms:stupdalmc|almmsk2[18]
4	2:1				192 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalme:stupealmc|almmsk1[0]
5	2:1				192 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalms:stupfalmc|almmsk3[16]
6	34:1				147 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupfc|ppflgi[4]
7	34:1				147 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupgc|ppflgi[6]
8	34:1				147 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupec|ppflgi[2]
9	34:1				147 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupfc|ppflgi[0]
10	34:1				147 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupfc|ppflgi[5]
11	34:1				147 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupfc|ppflgi[1]
12	135				138 LEs	No	|mbcb|cbusb:U0|dto[29]~61
13	66:1				120 LEs	Yes	|mbcb|serialb:u1|datadn:datadncont|dndatac:dataoutcont|actboutdt[5]
14	64:1				114 LEs	Yes	|mbcb|serialb:u1|datadn:datadncont|dndatac:dataoutcont|matedndt[3]
15	2:1				112 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalms:stupcalmc|almmsk0[35]
16	2:1	112 bits	112 LEs	0 LEs	112 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalms:stupaalmc|almmsk2[36]
17	2:1	96 bits	96 LEs	0 LEs	96 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalms:stupcalmc|almmsk1[32]
18	2:1	96 bits	96 LEs	0 LEs	96 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalms:stupcalmc|almmsk3[32]
19	142:1	2 bits	188 LEs	92 LEs	96 LEs	No	|mbcb|cbusb:U0|dto[25]~57
20	67:1	3 bits					dt[4]
21	136:1	2 bits					
22	17:1	7 bits					
23	17:1	7 bits					
24	64:1	2 bits					dt[5]
25	148:1	4 bits					
26	2:1	56 bits					
27	2:1	56 bits	56 LEs	0 LEs	56 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalmg:stupgalmc|almbuf[10]
28	2:1	56 bits	56 LEs	0 LEs	56 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalms:stupcalmc|almbuf[20]
29	2:1	56 bits	56 LEs	0 LEs	56 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalme:stupealmc|almbuf[30]

Multiplexer Inputs	Bus Width
143:1	9 bits
2:1	224 bits
2:1	224 bits
2:1	192 bits
2:1	192 bits
34:1	7 bits

- **Number of Unique (or Constant) Inputs**
- **Number of busses with identical structure**

(Table is always produced after Analysis & Synthesis, *even if optimizations are disabled*)

New Multiplexer Report:

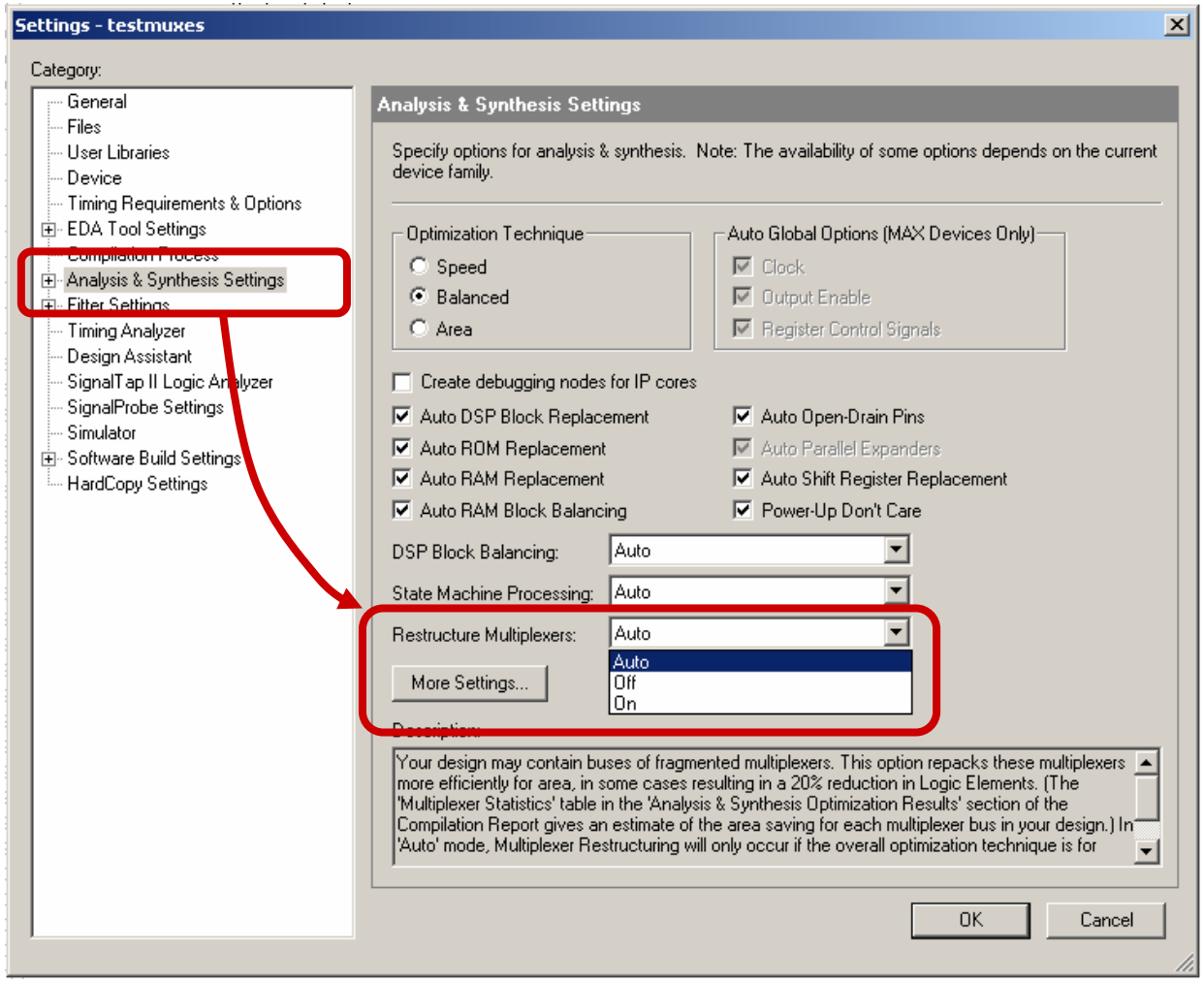
Multiplexer Restructuring Statistics (Restructuring Performed)

	Multiplexer Inputs	Bus Width	Baseline Area	Area if Restructured	Saving if Restructured	Registered	Example Multiplexer Output
1	143:1	9 bits	855 LEs	405 LEs	450 LEs	No	|mbcb|cbusb:U0|dto[24]~56
2	2:1	224 bits	224 LEs	0 LEs	224 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalme:stupealmc|almmsk0[1]
3	2:1	224 bits	224 LEs	0 LEs	224 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalms:stupdalmc|almmsk2[18]
4	2:1	192 bits	192 LEs	0 LEs	192 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalme:stupealmc|almmsk1[0]
5	2:1	192 bits	192 LEs	0 LEs	192 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalms:stupfalmc|almmsk3[16]
6	34:1	7 bits	154 LEs	0 LEs	154 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupfc|ppflgi[4]
7	34:1	7 bits	154 LEs	0 LEs	154 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupgc|ppflgi[6]
8	34:1	7 bits	154 LEs	0 LEs	154 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupec|ppflgi[2]
9	34:1	7 bits	154 LEs	0 LEs	154 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupfc|ppflgi[0]
10	34:1	7 bits	154 LEs	0 LEs	154 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupfc|ppflgi[5]
11	34:1	7 bits	154 LEs	0 LEs	154 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupfc|ppflgi[1]
12	135:1	3 bits	270 LEs	132 LEs	138 LEs	No	|mbcb|cbusb:U0|dto[29]~61
13	66:1	4 bits	176 LEs	56 LEs	120 LEs	Yes	|mbcb|serialb:u1|datadn:datadncont|dndatac:dataoutcont|actboutdt[5]
14	64:1	3 bits	126 LEs	12 LEs	114 LEs	Yes	|mbcb|serialb:u1|datadn:datadncont|dndatac:dataoutcont|matedndt[3]
15	2:1	112 bits	112 LEs	0 LEs	112 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalms:stupcalmc|almmsk0[35]
16	2:1	112 bits					ont|stupalms:stupaalmc|almmsk2[36]
17	2:1	96 bits					nt|stupalms:stupcalmc|almmsk1[32]
18	2:1	96 bits					nt|stupalms:stupcalmc|almmsk3[32]
19	142:1	2 bits					
20	67:1	3 bits					adncont|dndatac:dataoutcont|actboutdt[4]
21	136:1	2 bits	180 LEs	90 LEs	90 LEs	No	|mbcb|cbusb:U0|dto[28]~60
22	17:1	7 bits	77 LEs	0 LEs	77 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupec|ppflgi[3]
23	17:1	7 bits	77 LEs	0 LEs	77 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupin:stupbc|ppflgi[7]
24	64:1	2 bits	84 LEs	8 LEs	76 LEs	Yes	|mbcb|serialb:u1|datadn:datadncont|dndatac:dataoutcont|matedndt[5]
25	148:1	4 bits	392 LEs	328 LEs	64 LEs	No	|mbcb|cbusb:U0|dto[8]~40
26	2:1	56 bits	56 LEs	0 LEs	56 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalms:stupbalmc|almbuf[5]
27	2:1	56 bits	56 LEs	0 LEs	56 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalmg:stupgalmc|almbuf[10]
28	2:1	56 bits	56 LEs	0 LEs	56 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalms:stupcalmc|almbuf[20]
29	2:1	56 bits	56 LEs	0 LEs	56 LEs	Yes	|mbcb|serialb:u1|stup:stupcont|stupalme:stupealmc|almbuf[30]

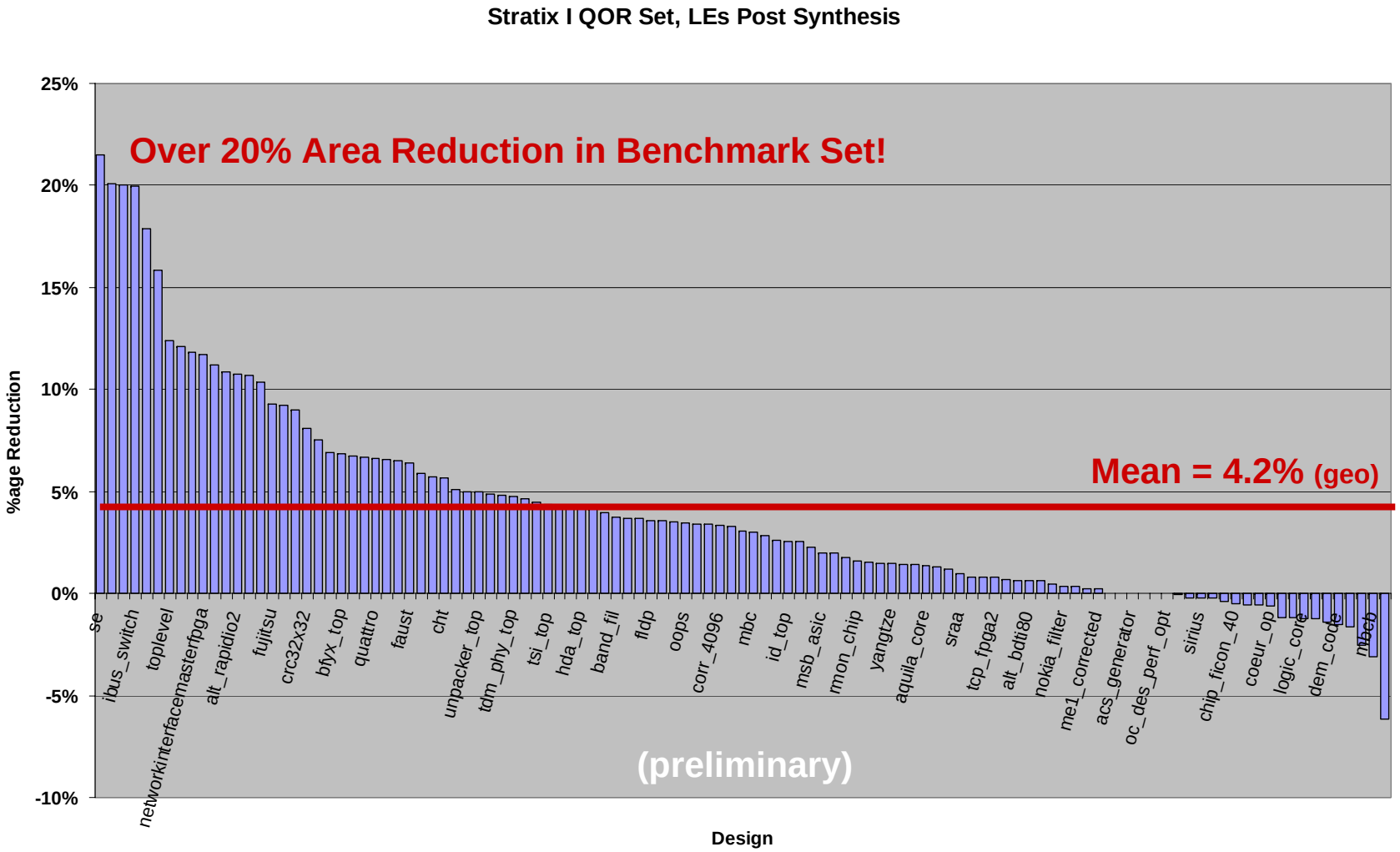
– Estimate of Area Inefficiency

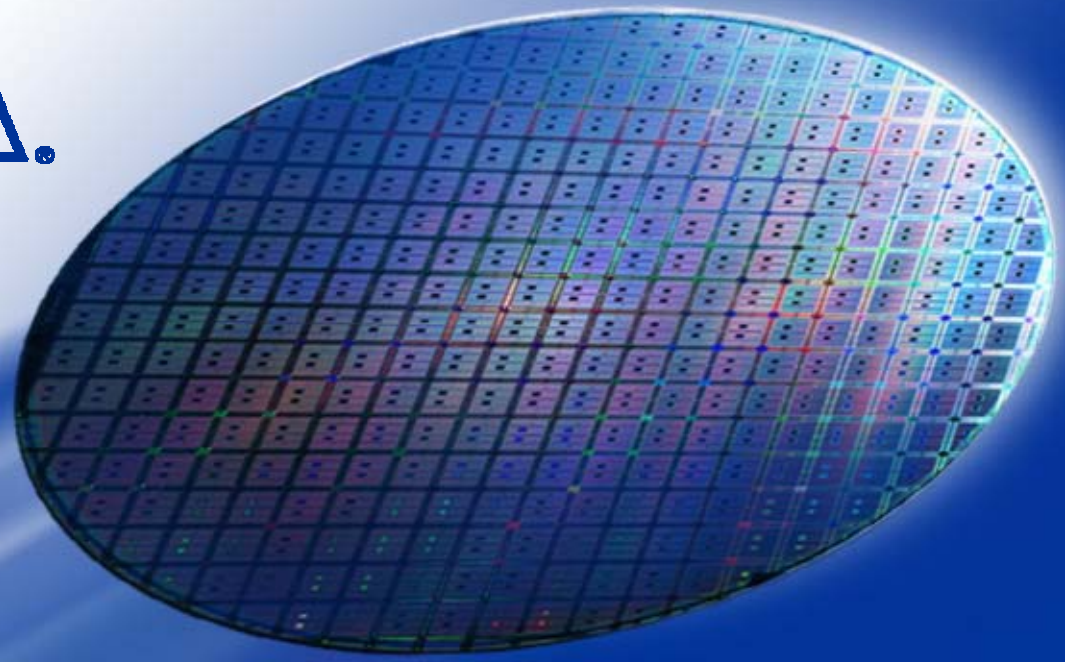
(Table is always produced after Analysis & Synthesis, *even if optimizations are disabled*)

New Synthesis Option:



Results: (Stratix I: Logic Reduction)

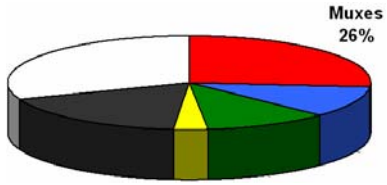




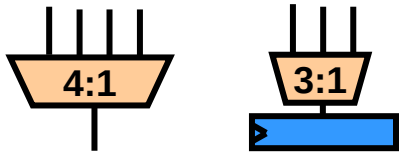
Summary

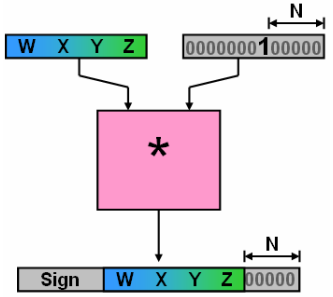
Summary

- System Design on FPGAs
  - Low cost easy-to-use tools with Time-to-Market advantage

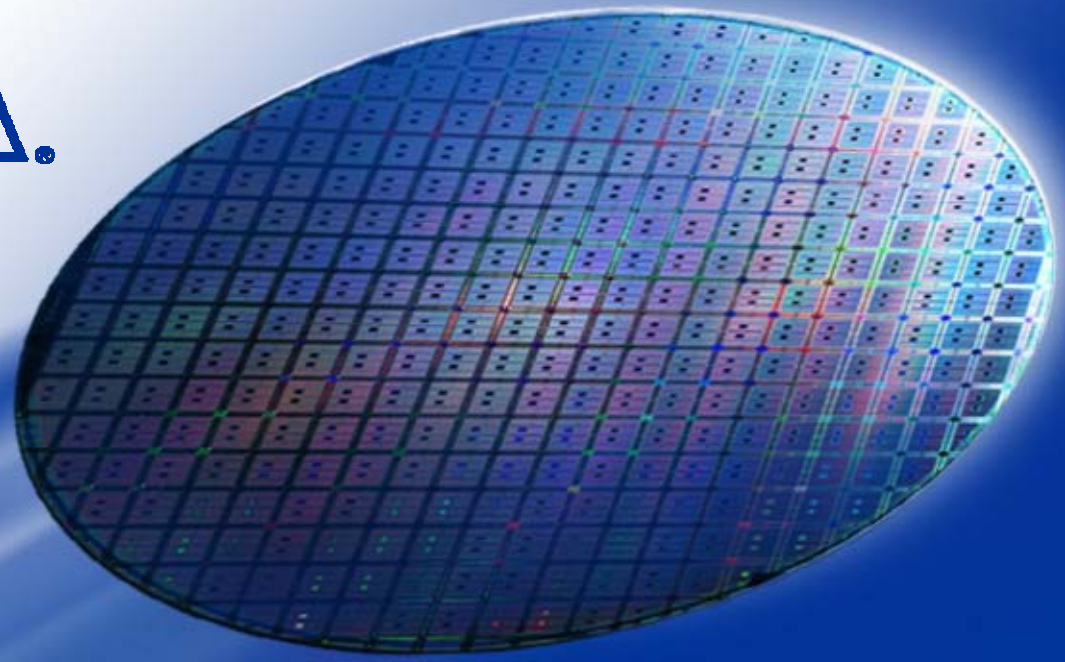


- Architecting Designs for FPGAs
  - **Multiplexer Costs can dominate in FPGAs**
    - 25% of the area on average
    - Significant in Processor / Busses
  - **FPGA Multiplexer Costs do not scale linearly**
    - best to map to 4:1 or 3:1(reg)
    - Registers can reduce multiplexer costs!
  - **The Cheapest Multiplexers are those not implemented in Logic!**
    - Eg: By using a multiplier





- Synthesis Tools assist in Optimization Process
  - But the Designer still has a huge influence on QoR



The End.

Questions?