

Evolution, Revolution and Convolution

Recent Progress in Field-Programmable Logic

Peter Alfke
Xilinx, Inc.

peter.alfke@xilinx.com

XILINX, Virtex, and Spartan are registered trademarks of Xilinx, Inc.
All other trademarks are the property of their respective owners

Purpose:

**Explain device capabilities and design issues
so that the users can concentrate
on the important and unique aspects
of their designs**

***“Let’s improve cost, size, and performance,
and avoid re-inventing the wheel !”***

Evolution:

**Bigger, faster, cheaper FPGAs,
Better software, fewer bugs
Faster compile times
Better support**

“The Obvious”

Revolution:

Versatile I/O for 5 V, 3.3 V, 2.5 V and 1.8 V interfaces

Controlled-Impedance I/O

drives transmission lines, simplifies pc-boards

3.125 Giga-bps serial I/O,

2.5 Gbps data + clock recovery + 8B10B + 20-bit FIFOs

Soft and hard embedded microprocessors

MicroBlaze and PowerPC

Convolution

**BlockRAM as finite state machines, counters,
asynchronous & synchr. FIFOs, code converters**

Multipliers as logic or arithmetic shifters

Triple redundancy to correct single-event upsets

Floating Point adder and multiplier

200 MHz pulse generator, variable-frequency

1 GHz frequency counter with 8-digit display

Asynchronous techniques

Evolution

Bigger, Faster, Cheaper

Supply Voltage

Capacity

Features

Routing

Clocking

Multipliers

User Expectations

- ◆ **Logic capacity at reasonable cost**
 - *100,000 to several million gates*
 - *Support for DSP (fast arithmetic, dedicated multipliers)*
 - *On-chip fast RAM*
- ◆ **Clock speed**
 - *150 MHz and above, global clocks, clock management*
- ◆ **Versatile I/O**
 - *Accommodate a variety of standards, drive transmission lines*
- ◆ **Design effort and time**
 - *Synthesis, fast compile times,*
 - *Tested and proven cores, including microprocessors*
- ◆ **Power consumption**
 - *must stay within reasonable limits*

Three Pillars of Progress

◆ Technology

- *smaller geometries, more and faster transistors*
- *better defect densities, larger chips, larger wafers, lower cost*

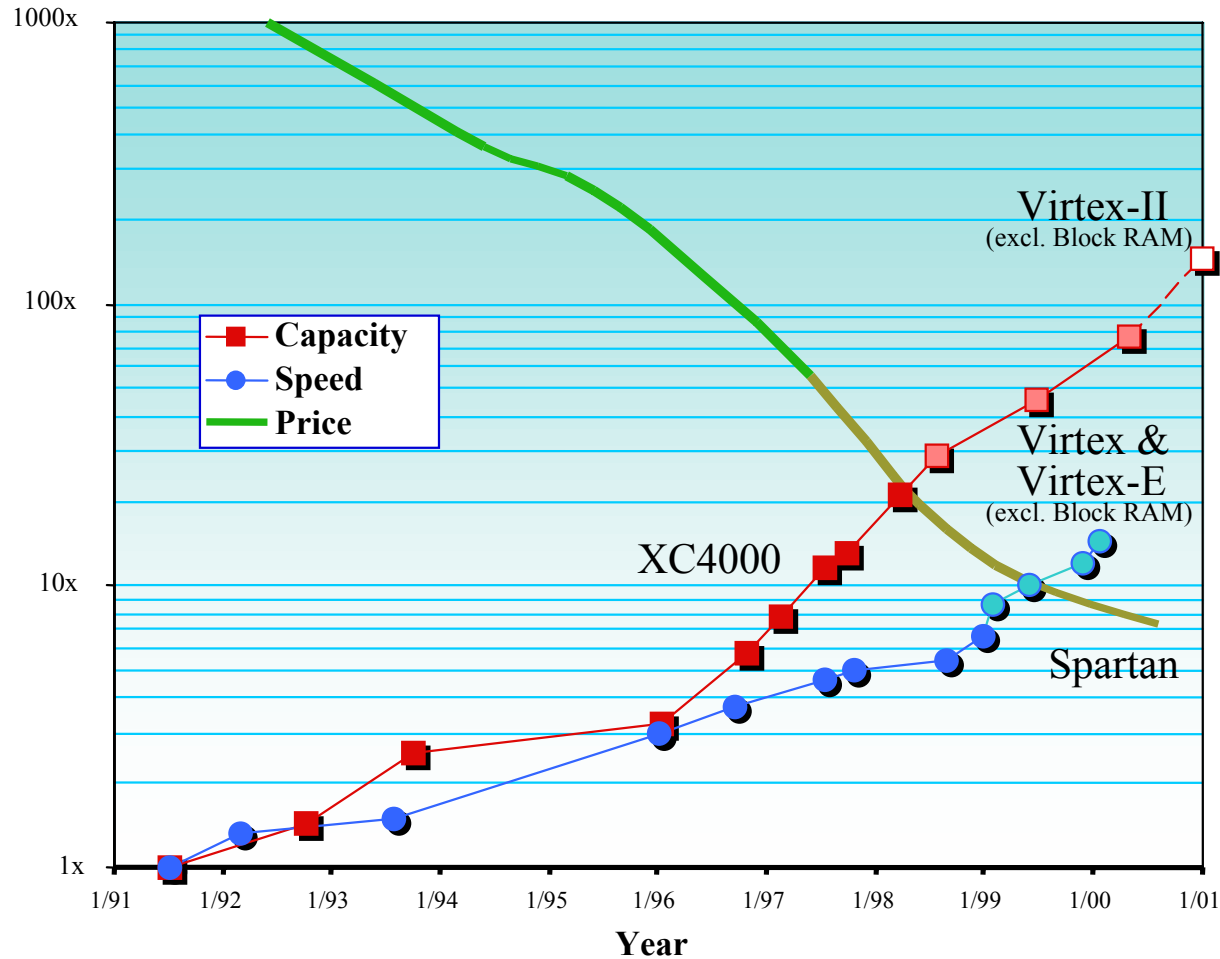
◆ Architecture

- *system features: fast carry, memory, clock management*
- *hierarchical interconnect, controlled-impedance I/O*

◆ Design Methodology

- *powerful and reliable cores, faster compilation*
- *modular, team-based design, internet-based tools*

A Decade of Progress



Response	Percentage
Yes	65%
No	30%
Don't know	5%

- P.A. Sthlm Sept 2001– 10**

Logic Capacity and Features

	<i>LUTs & FFs</i>	<i>Additional Features</i>
◆ XC4000/Spartan:	152...12,312	Carry, LUT-RAM
◆ Virtex/Spartan-II:	432...27,648	4K-BlockRAM, DLL, SRL16
◆ Virtex-E:	1,728...43,200	differential I/O
◆ Virtex-II:	512...67,548	18K-BlockRAM, Multipliers, DCM, Controlled Impedance I/O
◆ Virtex-II Pro:	2,816...45,184	PowerPC, 3.125 Gbit/sec I/O

Do Not Use for New Designs

Dead:	XC2000, XC6200
Senile:	XC3000, XC4000, XC5200, 5-V Logic
Seniors:	XC4000XL and Spartan XL, 3.3-V logic
Mid-life:	Virtex

Age Gives Way to Youth !

“One year for FPGAs = 15 years for humans”

Do Use for New Designs

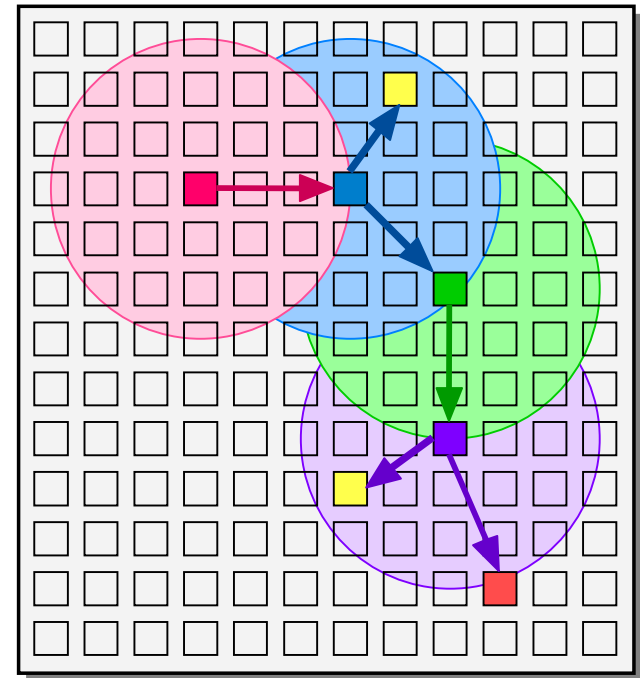
- Mature:** **Spartan-II:** inexpensive, Virtex-like, 5-V tolerant
 Virtex-E: economical, widely available
- Adolescent:** **Virtex-II:** faster, better clock management,
 larger RAMs.
 complete family by late 2001
- Embryonic:** **Virtex-II Pro:** on-chip PowerPC,
 3.125 Gbps serial I/O,
 available early 2002

***“The future belongs to the children,
and the unborn...”***

Fast Logic Needs Fast Routing

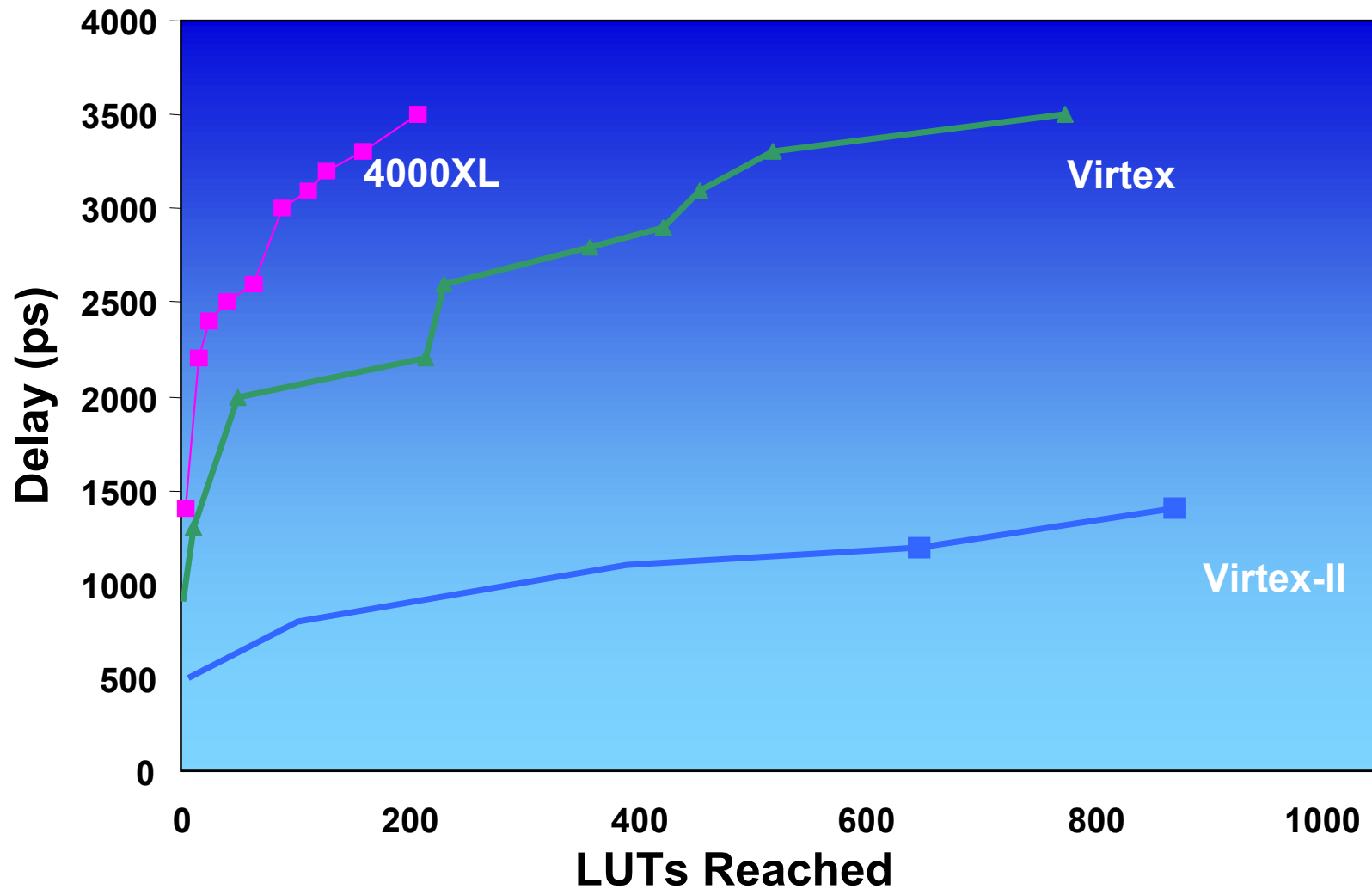
- ◆ Typical designs need a routing delay of < 1.5 ns
- ◆ Virtex delivers this performance
- ◆ Virtex-II is even faster
- ◆ Delay is independent of direction
 - *Dependably short delays provided by large numbers of short interconnect resources*

Vector-based Interconnect



The circles show 1.4-ns routing delay

Active Interconnect Technology



Global Clock Distribution

- ◆ **Global Clock with short delay**
 - *zero delay when using the Delay-Locked Loop*
- ◆ **Extremely small skew: <200 ps**
 - *while clocking up to 100,000 flip-flops*
 - *no hold-time issues ever, when using global clocks*
- ◆ **Clock gating and clock multiplexing in Virtex-II**
 - *no glitches or runt pulses, even with asynchronous control*
- ◆ **XC4000 / Spartan: 8 Global Clocks**
- ◆ **Virtex/Spartan-II: 4 Global Clocks, four DLLs**
- ◆ **Virtex-E: 4 Global Clocks, eight DLLs**
- ◆ **Virtex-II & V2Pro: 16 Global Clocks, up to 12DCM**

Global Clocks: BUFGMUX

◆ Three Modes:

— Clock Buffer

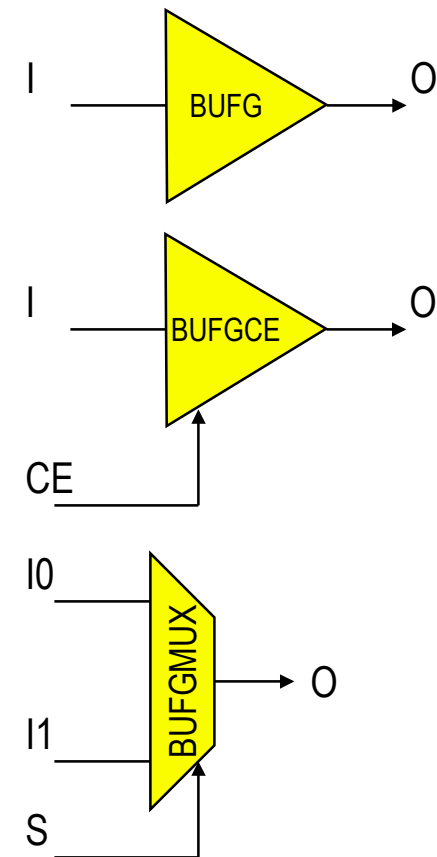
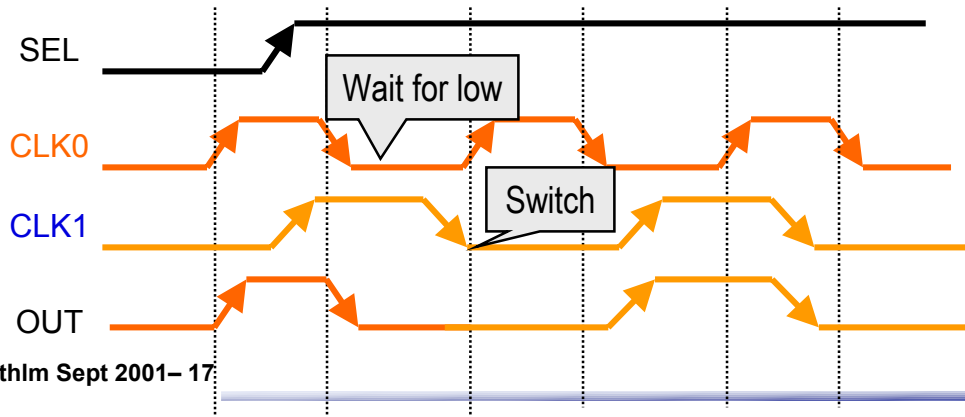
- Low skew clock distribution
- BUFG primitive

— Clock Enable

- Stop the clock High or Low
- BUFGCE (stop Low)

— Clock Multiplexer “Glitch-free”

- Switch between unrelated clocks
- BUFGMUX



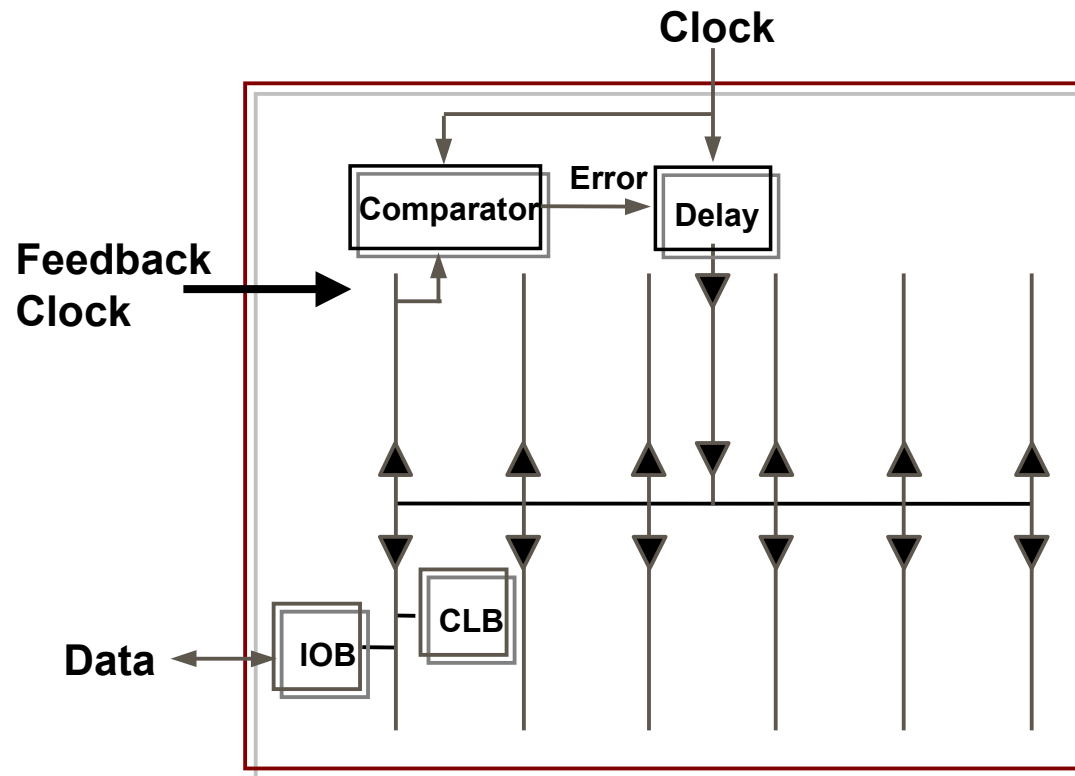
No pulse is ever shorter than the shortest input pulse

Clock Management with DLL

- ◆ **Can eliminate on-chip clock delay**
 - *can also eliminate **on-board** clock delay*
- ◆ **4 fixed-phase outputs (0, 90, 180, 270 degrees)**
- ◆ **Selectable phase shift ($n / 256$ of the period)**
 - *through configuration*
 - *or through increment/decrement*
 - *1/256 of clock period or 50 picosecond granularity*
- ◆ **Frequency synthesis (division and multiplication)**
- ◆ **Outputs are always phase-coherent**

Solves the speed problem of large chips

4 to 12 Independent DLLs



- ◆ **DLLs adjust clock delay to align internal and external clocks**
 - *Digital closed-loop control*
 - *25 to 400-MHz range, 50-picosecond resolution*

Use Phase Adjustment

- ◆ **To fine-tune input signal capture**
 - *Minimize set-up time while avoiding hold-time requirement*
 - *Center the clock in the eye of the data window*
 - Granularity: <50 ps
 - Can be feedback-controlled (servo)
- ◆ **To fine-tune output delay and hold time**
 - *Certain memories require data valid beyond the clock edge*
 - *Control timing on input and output independently*

You are now the master of nanosecond timing

Efficient Arithmetic

- ◆ **Dedicated Carry**

- *For adders, accumulators, counters, DSP applications*
- *<50 ps incremental carry delay per bit*
- *200 MHz operation over 64 bits*

- ◆ **2-s complement multipliers in Virtex-II**

- *18 x 18 bits in <7 ns, 8 x 8 in 4 ns*
 - *Faster pipelined operation will soon be supported*
- *Powerful and efficient for DSP*
- *Up to 192 independent multipliers*
 - *4 in the smallest device, XC2V40*

- ◆ **Use multipliers as shifters:**

- *17-bit arithmetic shift or 8-bit barrel shift in one multiplier*

Revolution

Beyond Bigger, Faster, Cheaper

LUT as shift register

On-chip RAM

Multi-standard I/O

Digitally-controlled impedance I/O

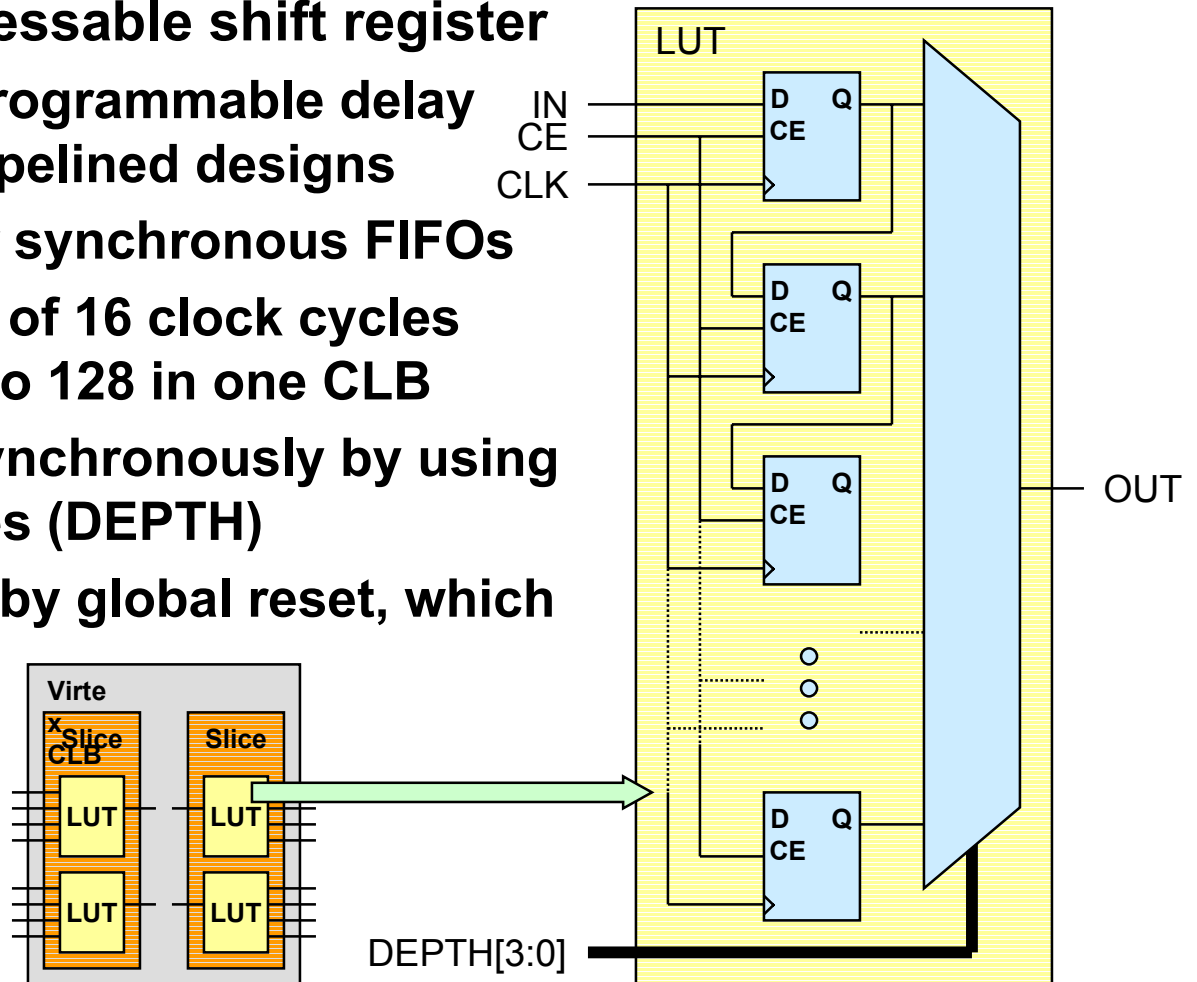
Gigabit/second serial I/O

Embedded microprocessors

Shift Register LUT (SRL16)

Dynamically addressable shift register

- ◆ Ultra-efficient programmable delay for balancing pipelined designs
- ◆ Can be used for synchronous FIFOs
- ◆ Maximum delay of 16 clock cycles in one LUT, up to 128 in one CLB
- ◆ Can be read asynchronously by using the address lines (DEPTH)
- ◆ Not being reset by global reset, which is good & bad



SRL16 Applications

- ◆ **1...16-bit shift register in one LUT**
 - *up to 128 bits in one Virtex-II CLB*
- ◆ **Pipeline compensation (different lengths per branch)**
- ◆ **FIFO, pseudo-random number generator (LFSR)**
- ◆ **Serial frame synchronizer**
- ◆ **Running average calculator**
- ◆ **Pulse generator and clock divider**
- ◆ **Pattern generator, state machine**
- ◆ **Website:** <http://support.xilinx.com/support/techxclusives/SRL16-techxclusive2.htm>

On-Chip RAM

- ◆ **Up to 120,000 Four-Input Look-Up Tables**
 - *Each 16-bit ROM, RAM or shift register*
 - 0.5 ns combin. delay, 0.5 ns set-up time, 0.5 ns clock-to-Q
 - 8 LUTs grouped in a CLB with very fast interconnect (Virtex-II)
- ◆ **Up to 192 dual-ported synchr. BlockRAMs**
 - *Each 4096 bits in Virtex,*
 - *Each 18K bits in Virtex-II, up to 36 bits wide*
 - <3 ns access time, >200 MHz operation
 - Can be used as FIFOs, state machines, counters, etc.
- ◆ **Also: Fast interface to external RAM**
 - *Up to 840 Mbp I/O data transfer rate (420 MHz DDR)*
 - *Fast bit-serial I/O, 3.125 Gbps in Virtex-IIPro (early 2002)*

Virtex-II BlockRAM

- ◆ Each 18 K bits
- ◆ Organized 16Kx1, 8Kx2, 4Kx4, 2Kx9, 1Kx18, 512x36
- ◆ Dual-ported with totally independent ports
 - *Each with its own Din, Dout, Address, Clock, Control*
 - *Each independently configurable as 16Kx1... 512x36*
- ◆ All operations are synchronous, even read
 - *Each port has its own clock*
- ◆ Reading data during write: 3 options for Dout
 - *“Write before read”, i.e. transparent read*
 - *“Read before write”, i.e. read the old content*
 - *“Do not read”, i.e. maintain old Dout, no change*

Multi-Standard I/O

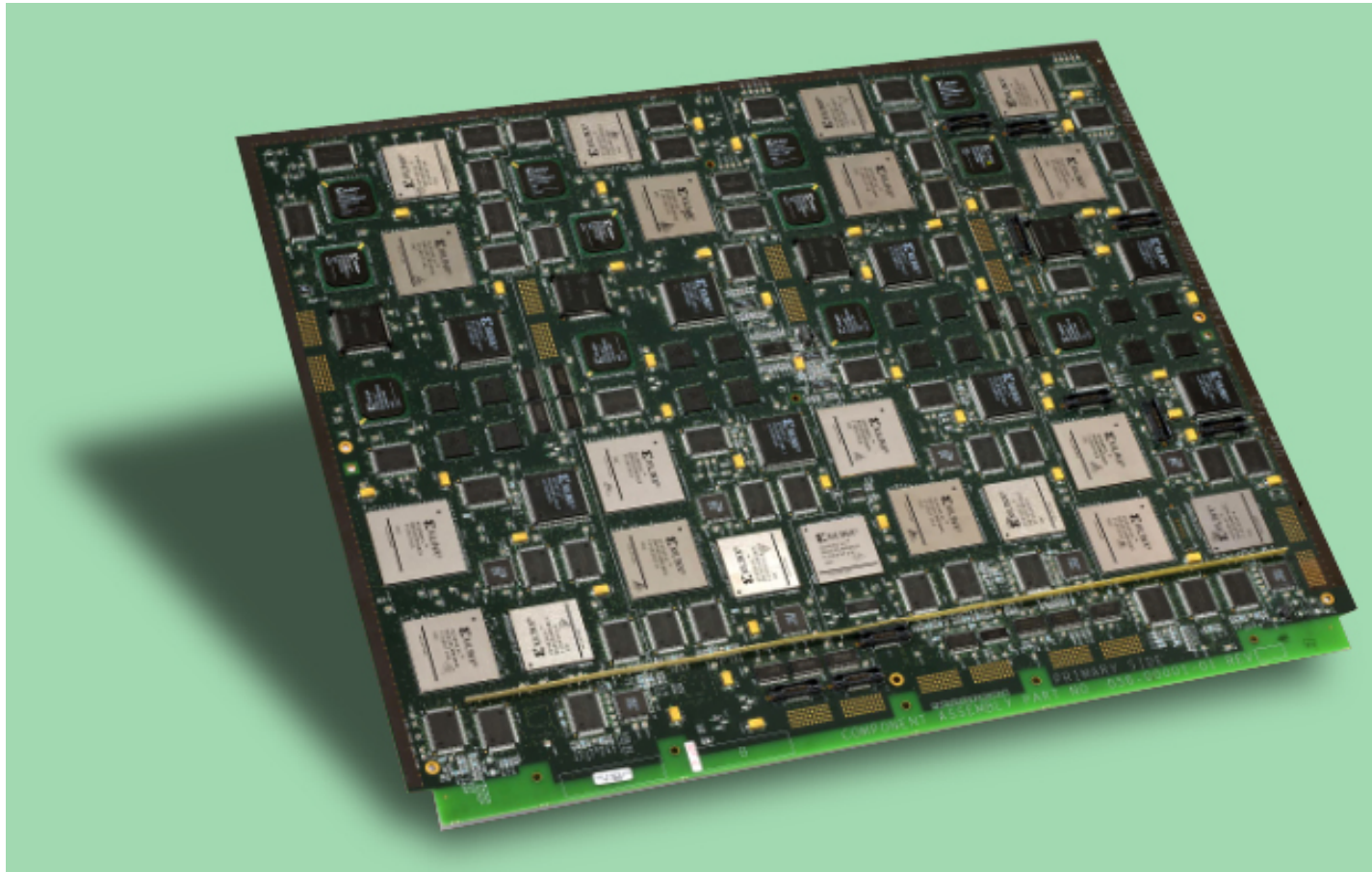
- ◆ **LV-TTL and LV-CMOS**
 - *for logic interfaces*
- ◆ **SSTL and HSTL (3.3, 2.5, 1.5 V)**
 - *for driving terminated lines*
- ◆ **GTL and GTL+**
 - *for driving double-terminated busses*
- ◆ **LVDS and LVPECL**
 - *high-speed differential signals*
- ◆ **Double-Data-Rate interfaces**
 - *for ultra-fast data transfer on rising and falling edge*

Multi-Standard I/O

- ◆ **Essential for system-level FPGAs**
 - *directly interfacing to many different circuits*
- ◆ **Essential for fast interconnects**
 - *requiring different features and trade-offs*
- ◆ **Essential for driving terminated lines**
 - *demanding by the fast transition times*
- ◆ **On-chip termination simplifies pc-boards**
 - *eliminates need for external resistor packs*

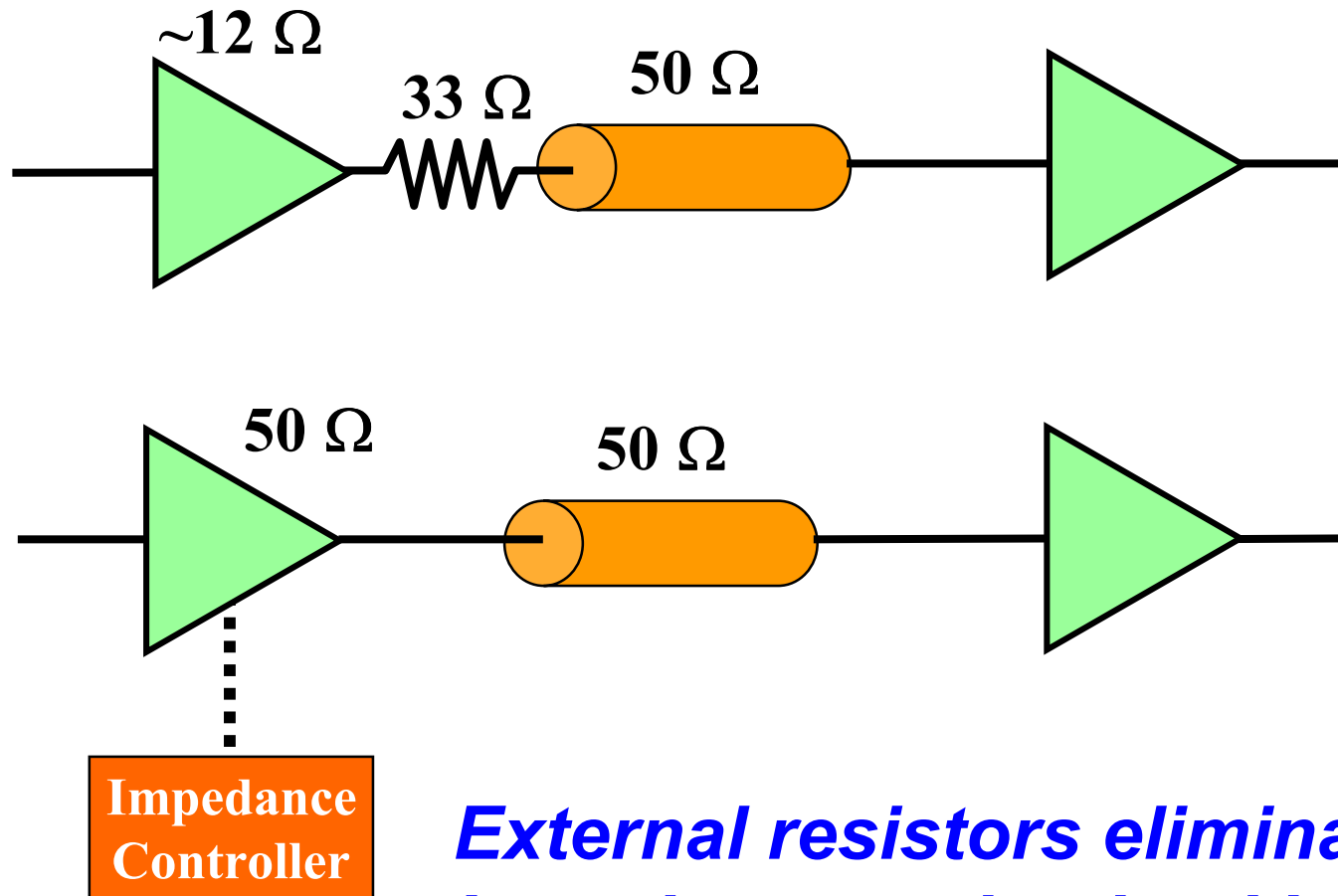
Optimized interface to any type of logic

Typical 20-Layer PCB: A Very Tough Design Problem



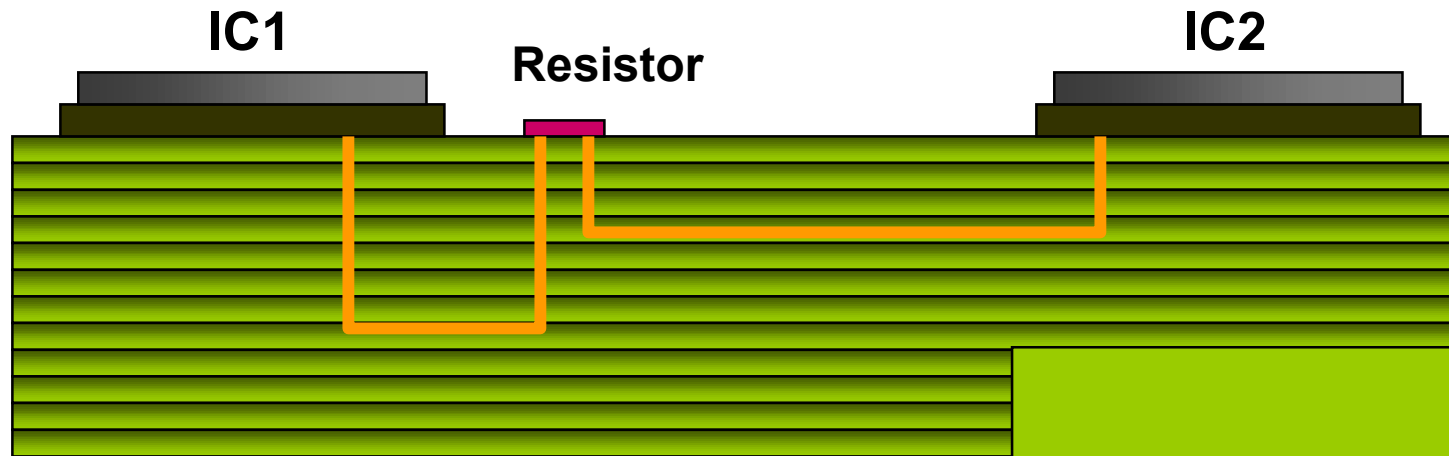
Courtesy, NetCore Systems.
P.A. Sthlm Sept 2001– 29

Digitally Controlled Impedance

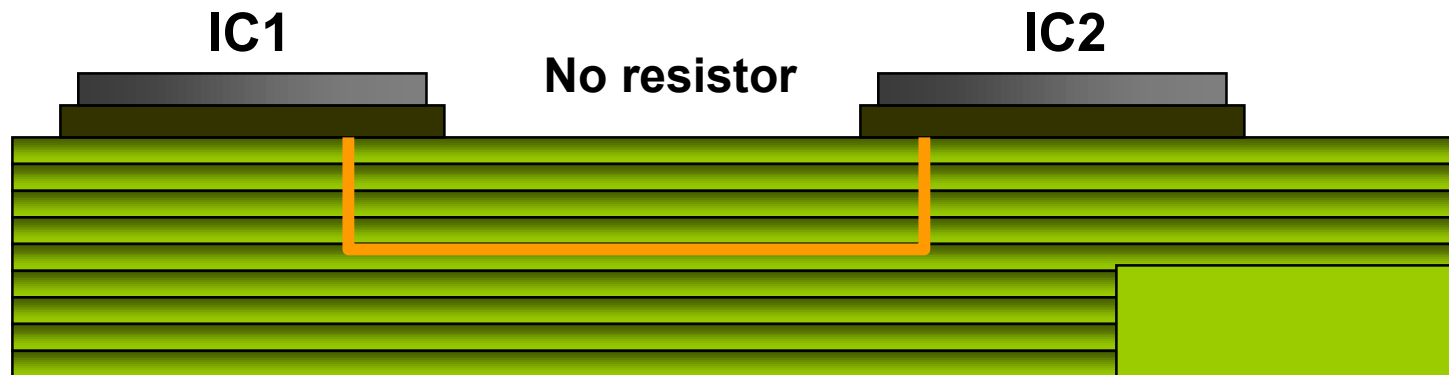


*External resistors eliminated
Impedance maintained by FPGA*

PC-Board Routing Impact



Multiply this by 1000 pins per chip, and by the N chips per board



Fewer Layers, fewer resistors, smaller board

P.A. Sthlm Sept 2001-31

© 2000, 2001 Xilinx, Inc. All Rights Reserved



Controlled-Impedance Benefits

- ◆ **Better signal integrity,**
 - *higher systems reliability*
- ◆ **Smaller PC-boards,**
 - *easier to layout,*
 - *easier to manufacture*

***XCITE I/O is the only practical way
to interconnect high pin-count
fine-pitch ball-grid packages***

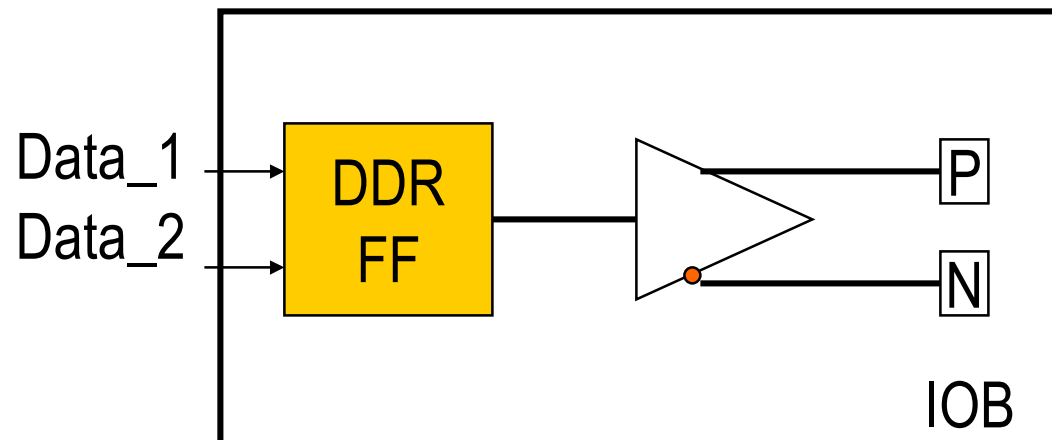
Differential Signaling: 840Mbps

- ◆ **Full LVDS Programmable Solution:**

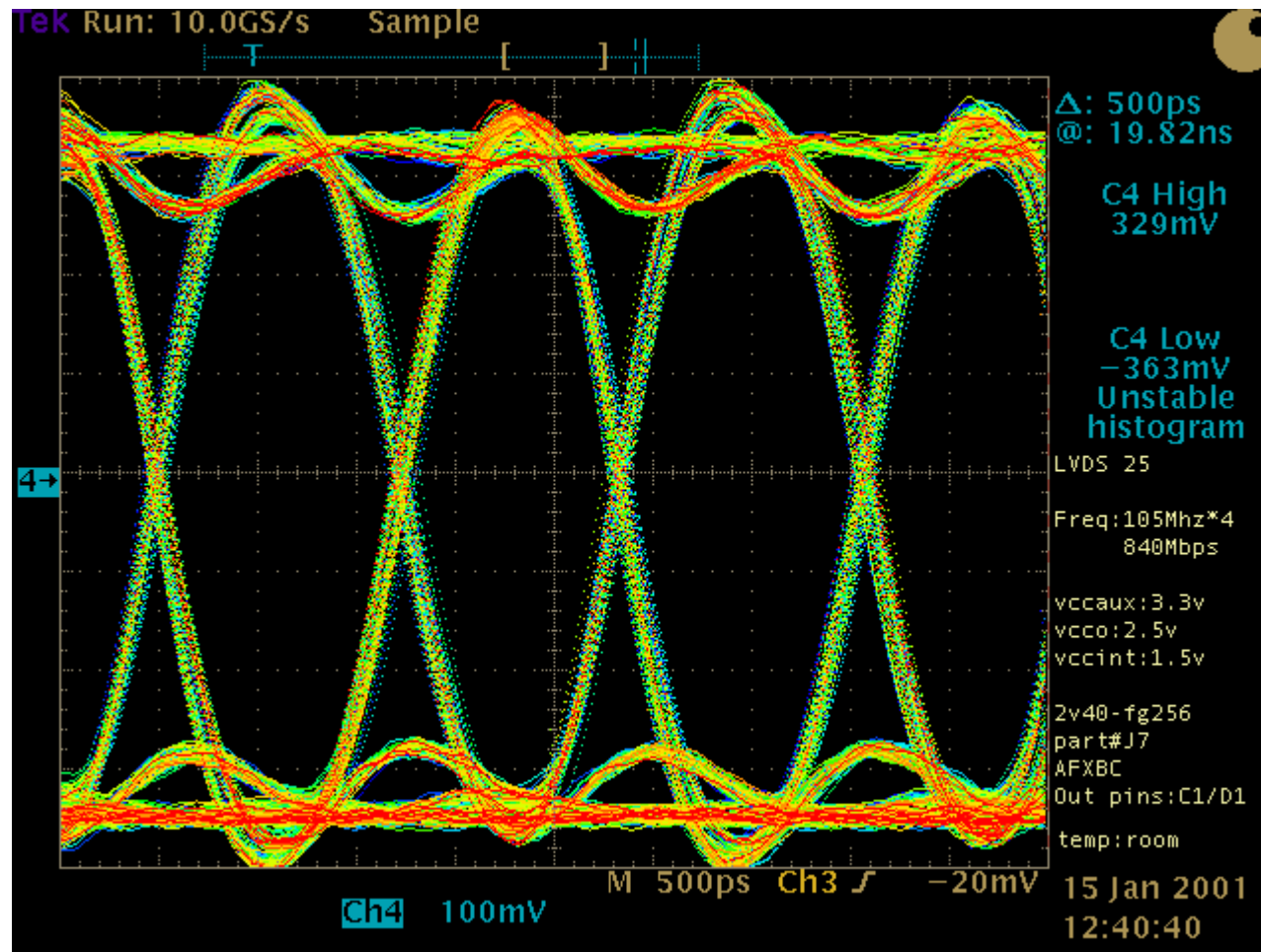
- 2.5 V : 250 mV - 400 mV
- 3.3 V : 250 mV - 400 mV
- Ext. 2.5 V : 350 mV - 750 mV
- Ext. 3.3 V : 350 mV - 750 mV

} Design
Flexibility

- ◆ Integrated current driver



840 Mbps Eye Pattern from Virtex-II LVDS



Board-Level Clock Distribution Alternatives

- ◆ ***Traditional Synchronous Design***
- ◆ ***Source-Synchronous Clock Forwarding***
- ◆ ***Bit-Serial Self-Clocking I/O***

Transmission Lines

- ◆ Long traces **are** transmission lines, they can ring
 - “*transmission line*” if round trip > transition time
 - “*lumped-capacitance*” if round trip < transition time
- ◆ Signal delay on a pc-board:
 - 140 to 180 ps per inch (50 to 70 ps per cm)
- ◆ Longest trace that behaves as a lumped-capacitance:
 - 3 inches max for a 1-ns transition time (7.5 cm)
 - 6 inches max for a 2-ns transition time (15 cm)
- ◆ Avoid reflection by terminating the line
 - either **series termination** at the source
or **parallel termination** at the destination

Evolution

	1965	1980	1995	2010 (?)
Max Clock Rate (MHz)	1	10	100	1000
Min IC Geometry (μ)	-	5	0.5	0.05
Number of IC Metal Layers	1	2	3	10
PC Board Trace Width (μ)	2000	500	100	25
Number of Board Layers	1-2	2-4	4-8	8-16

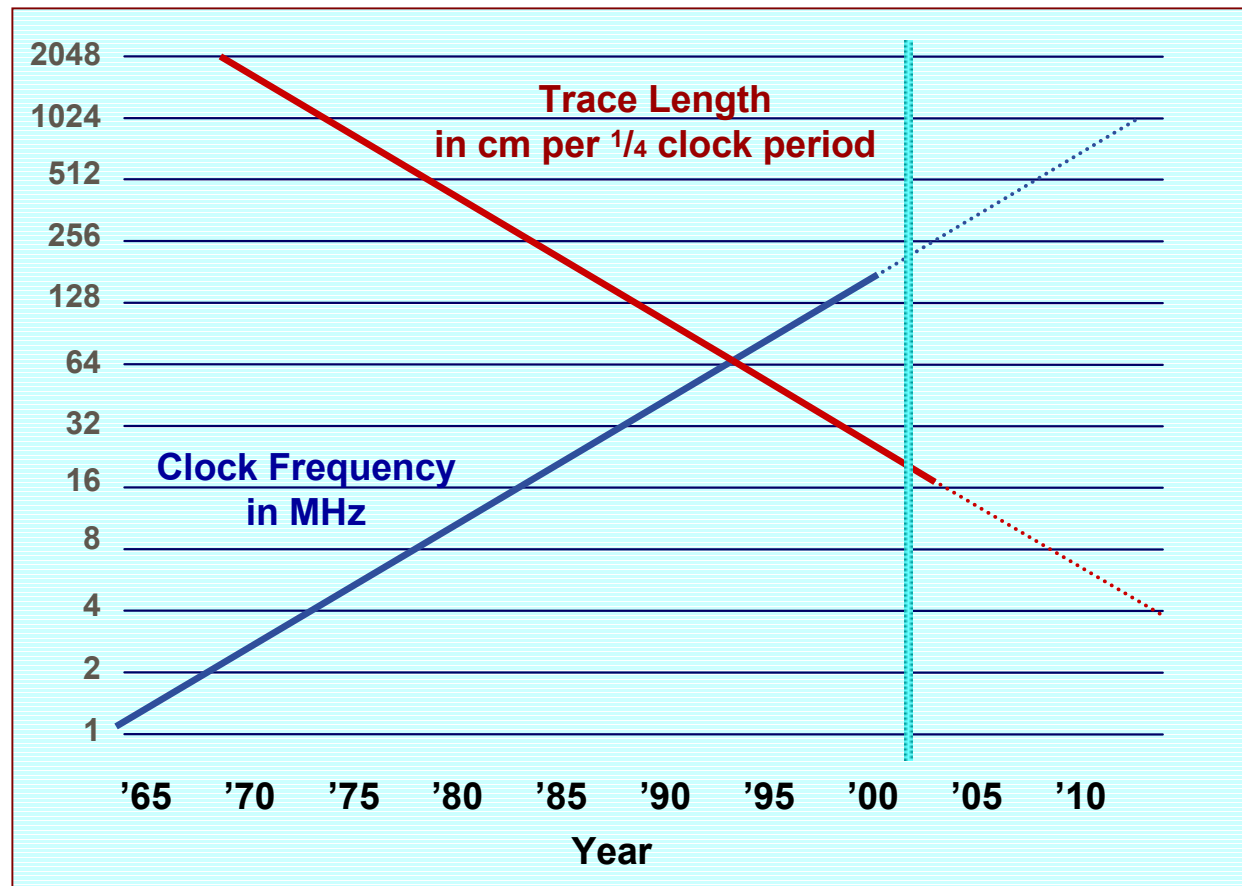
Every 5 years:

System speed doubles, IC geometry shrinks 50%

Every 7-8 years:

PC-board min trace width shrinks 50%

Moore Meets Einstein



- ◆ *Speed Doubles Every 5 Years...
...but the speed of light never changes*

Traditional Design

- ◆ **Common clock routed to all devices on the board**
- ◆ **Assumes “no board delay”**
 - *Hold-time problem if clock delay in the data direction*
 - *Performance problem if clock delay against the data direction*
- ◆ **Either rising or falling-edge clock (typically)**
 - *clock has twice the data bandwidth, clock drivers are limitation*
- ◆ **Largest distance between communicating device pins**
 - *assuming a data flight-time of 25% of the clock period:*
 - **20 cm at 250 MHz**
 - **10 cm at 500 MHz**
 - **5 cm at 1000 MHz**

Clearly out of the question above 500 MHz

Source-Synchronous Design

- ◆ **Clock intermingled with data**
 - *Uses the same type of clock driver, same routing*
- ◆ **Clock delay = data delay**
 - *No hold-time or performance problems*
- ◆ **Freedom to use double-data-rate clocking**
 - *Using both clock edges*
 - *Clock bandwidth is no higher than data bandwidth*
- ◆ **Disadvantage of source-synchronous clocking:**
 - *Works only in one direction,*
 - *Needs multiple clock signals*
 - *Uses up more device pins*

Bit-Serial Self-Clocking

- ◆ **Can achieve very high data rates, up to 10 Giga-bps**
 - *Propagation delay is irrelevant*
- ◆ **Uses one differential pair with Clock-Data Recovery**
 - *By definition, there is no skew between clock and data,*
- ◆ **Needs dedicated serializer / deserializer (SerDes)**
- ◆ **Rocket I/O solution in Virtex-II Pro (early 2002):**
 - *3.125 Gigabit/sec = 2,500 megabits per second data*
 - *dedicated clock recovery*
 - *dedicated Serdes*
 - *dedicated 8B/10B encoder/decoder*
 - *dedicated FIFOs*
- ◆ **FPGA fabric sees relatively low-speed parallel data:**
 - *8 bits @311 MHz or 16 bits @155 MHz or 32 bits @78 MHz*

On-Chip Microprocessors

Soft and Hard

◆ Soft implementation offers:

Flexibility and smaller size, and it's available today

- *But moderate performance (up to 125 MHz)*
- *Can have proprietary architecture and instruction set*
- *Example: Xilinx MicroBlaze*

◆ Hard implementation offers:

Higher performance, cache, etc. Available early 2002

- *Example: Virtex-II Pro with embedded PowerPCs*

Xilinx MicroBlaze

- ◆ 32-bit RISC architecture with Harvard buses
- ◆ 32-bit data path
- ◆ 32 general-purpose registers, 32 bit wide
- ◆ 3-operand instruction word
- ◆ Code and data access
 - *from on-chip BlockRAM or from external ROM / RAM*
- ◆ Supports IBM CoreConnect bus interface
 - *can use same IP cores as PowerPC in Virtex-II Pro*
- ◆ 125 MHz on Virtex-II
- ◆ < 900 Logic Cells, i.e. less than 10% of XC2V1000

Respectable performance and really small size

Virtex-II Pro with Embedded PowerPCs

- ◆ **Industry-standard 32-bit RISC processor**
- ◆ **300-MHz Harvard architecture core**
- ◆ **32 registers, 32 bits**
- ◆ **Separate instruction and data caches**
 - *2-way set associative*
- ◆ **MMU, variable page size (1KB to 16 MB)**
- ◆ **Dedicated on-chip memory interface (OCM)**
- ◆ **Supports IBM CoreConnect bus interface**
- ◆ **Low power consumption: 0.9 mW/MHz, small size**
- ◆ **One, two, or four PowerPC cores per Virtex-II Pro chip**
- ◆ **Available early 2002, comes with 3-Gigabit serial I/O**

Embedded Microprocessors

- ◆ **μPs excell at complex control at moderate speed**
 - *Communications protocol,*
- ◆ **□ FPGA fabric excels at ultra-fast parallel operation**
 - *Data path, DSP, Encryption/decryption*

Together they form the ideal on-chip team

Putting it all together: The Virtex-II Series Platform FPGA.

World's Fastest Logic & Routing

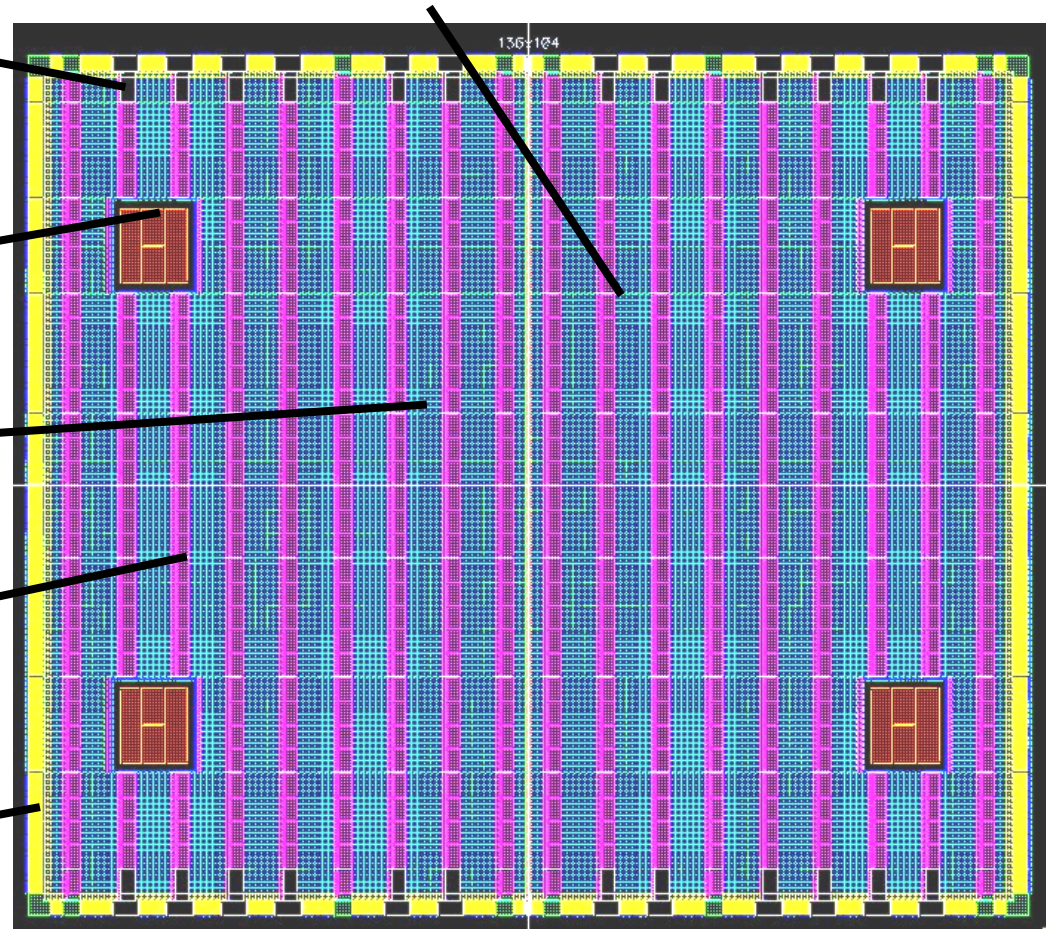
3.125Gb Serial

**IBM PowerPC[®]
RISC CPU**

XtremeDSP[™]

**Synchronous
Dual-Port RAM**

**SelectIO Ultra[™]
& SystemIO[™]**



Designing with FPGAs

- ◆ **Designing for High Performance**
- ◆ **Designing for Signal Integrity**
- ◆ **Designing for Security (Triple-DES)**
- ◆ **Designing with BlockROMs**
- ◆ **Designing for Low Power**
- ◆ **Designing for Radiation Tolerance**
- ◆ **Asynchronous Design Issues**
- ◆ **Tips and Tricks from the Xilinx Archives**



Designing for High Performance

Performance Parameters I

Parameter

Virtex-II-5 (ns)

CLB (internal):

Combinatorial LUT delay	0.39
Input set-up time through LUT	0.72
Carry delay per bit	0.045
Clock-to-Q delay	0.50

BlockRAM (internal):

Set-up time (A,D, control)	0.30
Internal clock to internal data-out	2.05

Input

Data pad to clock pad set-up time	1.60 (DCM eliminates clock delay)
Data pad to internal data-in delay	0.70

Output

Internal data to data output pad	2.63
Internal clock to data output pad	3.00
External clock pad to data out pad	2.50 (DCM eliminates clock delay)

Performance Parameters II

Internal register-to-register

16-bit adder	280 MHz
18 x 18 multiplier	110 MHz
24-bit synchronous counter	250 MHz
64-bit synchronous counter	170 MHz
DLL max output frequency	420 MHz

Virtex-II-5

Package-pad to package-pad combinatorial delays

64-bit decode,	9.3 ns
32 : 1 multiplexer	8.7 ns
One-LUT combinatorial function	5.0 ns

Virtex-II parameters are “advance” and conservative

Designing for High Speed

- ◆ **Understand the architecture, strength and limitations**

LUTs, LUT-RAMs, SRL16, Carry

Registered I/O, Output 3-state control flip-flops

Longlines, 3-state buffers, “free” multiplexers

Synchronous dual-ported BlockRAM

Global clocks with glitch-free enable and input multiplexer

DLLs, Digital Frequency Synthesizer, Phase control

Constant-coefficient multipliers in LUTs, fast and efficient

18x18 multipliers in Virtex-II, tightly coupled to BlockRAMs

The synthesizer cannot do all your homework

Provide High-Level Floorplanning

- ◆ ***Make intelligent pin assignment***
 - prevents routing congestion and poor performance
- ◆ ***Utilize the natural structure:***
 - Data flows horizontally, Control flows vertically
 - Vertical adders and counters, carry always going upwards
- ◆ ***Pick the best I/O standard, observe the banking rules***

Place & route tool should not do all your homework

Design Synchronously, Use Global Clocks

- ◆ **Up to 16 Global Clocks are available**
 - *Very low skew on these clock nets*
- ◆ **DLL eliminates clock distribution delay**
 - *Inside the chip, or even on the pc-board*
- ◆ **Do not gate the clock, use CE instead**
 - *But you may need clock gating for lowest power*
 - *Virtex-II has glitch-free clock gate and clock mux*
- ◆ **Use Carry for adders, counters and comparators**
 - *Superior speed, less logic, forces vertical orientation*
- ◆ **Use predefined cores**
 - *They have been tested and are guaranteed to work at speed*



Designing for Signal Integrity

Designing for Signal Integrity

- ◆ **Devices need good Vcc bypassing**
 - *Bypass capacitor is the only source of dynamic current*
- ◆ **Output drivers are specified by IBIS models**
 - *http://www.xilinx.com/support/troubleshoot/htm_index/sw_ibis.htm*
- ◆ **User needs understanding of transmission line effects**
 - *Characteristic impedance, reflections, dV/dt*
 - *Series termination, parallel termination, dynamic termination*
- ◆ **Model the pc-board with HyperLynx**
 - *Multi-Layer with undisturbed ground/power planes*
 - *Controlled-impedance signal lines (50 to 75 Ohms)*
- ◆ **Website:**
 - *<http://www.xilinx.com/support/techxclusives/CircuitBoard-techX6.htm>*

Signal Integrity Tools

- ◆ **IBIS models**

- http://www.xilinx.com/support/troubleshoot/htm_index/sw_ibis.htm

- ◆ **HyperLynx**

- ◆ **Fast oscilloscope and fast probes**

- *Beware of slow scopes when measuring **1 ns** rise time:*

- *A **1 GHz** scope with a 1 GHz probe displays **1.2 ns** rise time*

- *A **250 MHz** scope and probe displays: **3.0 ns** rise time*

- ◆ **Measure eye patterns**

- *Use LFSRs to generate pseudo-random sequence*

- ◆ **Spectrum analyzer**

- *Measure the effect of decoupling capacitors, etc.*

- ◆ **Website:**

- <http://www.xilinx.com/support/techxclusives/signals-techX5.htm>

Power Supply Decoupling

- ◆ **CMOS current is dynamic**
 - *I_{cc} current spike on every active clock edge*
- ◆ **Peak current can be 5x the average current**
 - *Instantaneous current peaks can only be supplied by decoupling capacitors*
- ◆ **Use one 0.1 uF ceramic chip capacitor per Vcc pin**
 - *Low L and R are more important than high C*
 - *Double up for lower L and R if necessary*
 - *Use direct vias to the supply planes, place Cs extremely close to the power-supply pins*
 - *On-chip Vcc distribution capacitance is ~0.01μF*

Tricks of the Trade

- ◆ **Reduce the output strength**
 - *LVTTL and LVCMOS offer 2, 4, 6, 8, 12, 16, and 24 mA*
- ◆ **Use SLOW attribute where available**
 - *Increases transition time especially when driving transmission lines*
- ◆ **Explore different I/O standards**
 - *Different supply voltages, input thresholds*
 - *Unidirectional, bidirectional, bus-oriented, differential...*
- ◆ **Reduce fan-out and load capacitance**
- ◆ **Add virtual ground to alleviate SSO problems**
 - *Ground output pin inside and outside, give it max strength*

Testing for Performance and Reliability

- ◆ **Manipulate circuit speed for testing purposes:**
 - *Hot and low V_{cc} = slow operation*
 - *Cold and high V_{cc} = fast operation*
- ◆ **If it fails hot: the IC is too slow**
 - *Use a faster speed grade*
 - *Modify the design, add pipelining*
- ◆ **If it fails cold: the IC is too fast**
 - *Look for signal integrity and hold time issues*
 - *Look for clock reflections on the pc-board*
 - *Look for excessive internal clock delays (hold time issues)*
 - *Look for decoding spikes driving internal clocks*
 - *Look for “dirty asynchronous tricks”*

Model and Measure

- ◆ **Model** device, package, pc-board
 - *Avoids costly pc-board re-spin*
- ◆ **Measure** performance and noise margin
 - *Avoids field disasters*
- ◆ **Do not panic:**
 - *It's only 1 and 0, High and Low that count*
 - *Noise immunity takes care of the rest*
- ◆ **References:**
 - *Classes: see www.hyperlynx.com, then go to TRAINING*
 - *Book: Johnson & Graham High-Speed Digital Design*
- ◆ **Website:**
 - *www.xilinx.com/support/techxclusives/techX-home.htm*

Designing for Data Security

- ◆ **Triple-DES encryptor / decryptor app note, XAPP270**
 - *Fastest known implementation in the world*
 - twice as fast as a dedicated Sandia ASIC of 1999
 - *>14 Gbps data rate in XC2V1000 (222 MHz clock)*
 - *>13 Gbps in XCV400E-8*
 - *High performance through massive parallelism*
 - 5000 LUTs and 6000 flip-flops working in parallel,
 - heavily pipelined, 48 stages

- ◆ **Triple-DES bitstream decryption in Virtex-II**
 - *A free option to protect the design from being copied*
 - *Dedicated to bitstream decryption, using standard algorithm*
 - External Li-battery holds key of 3 x 56 bits, 10 year life



Designing with BlockROMs

Designing with BlockRAMs

- ◆ **Dual-ported synchronous BlockRAMs**
 - *Synchronous read and write*
- ◆ **Two Ports share nothing but the common data array**
 - *Individual address, data, clock, read/write, CE*
- ◆ **Each port can be configured individually**
 - *Parallel-serial (or S-P) converter “for free”*
- ◆ **4K bits per BlockRAM in Virtex**
 - *4K, 2K, 1K or 512 deep (256 x 16 with ports combined)*
- ◆ **18K bits per BlockRAM in Virtex-II**
 - *16K, 8K, 4K, 2K, 1K or 512 deep (256 x 72 combined)*
- ◆ **Max 180 BlockRAMs in Virtex, max 192 in Virtex-II**

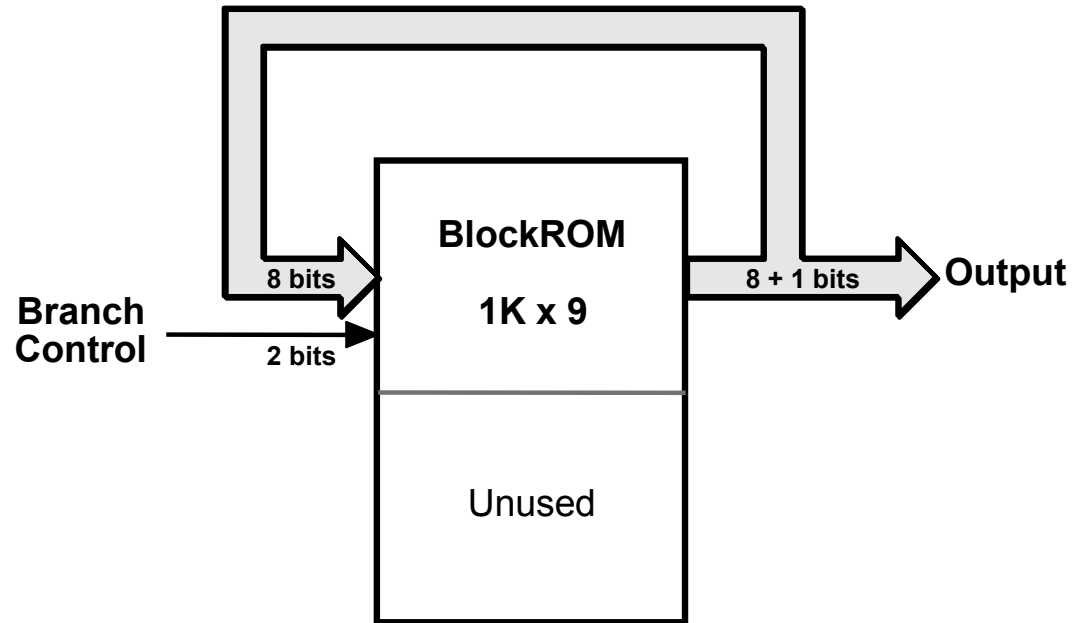
BlockROM State Machines

- ◆ **BlockRAM can be initialized as BlockROM**
- ◆ **Virtex has 4086 bits per BlockROM**
- ◆ **Counters**
 - *Two 8-bit Gray counters*
 - with additional binary outputs
 - *Two 1-digit decimal counters*
 - with 7-segment read-out
 - *16-bit up/down binary counter*
 - *4-digit BCD up/down counter*
- ◆ **Finite State Machines**
 - *Two 4-input 32-state state machines*

BlockROM State Machines

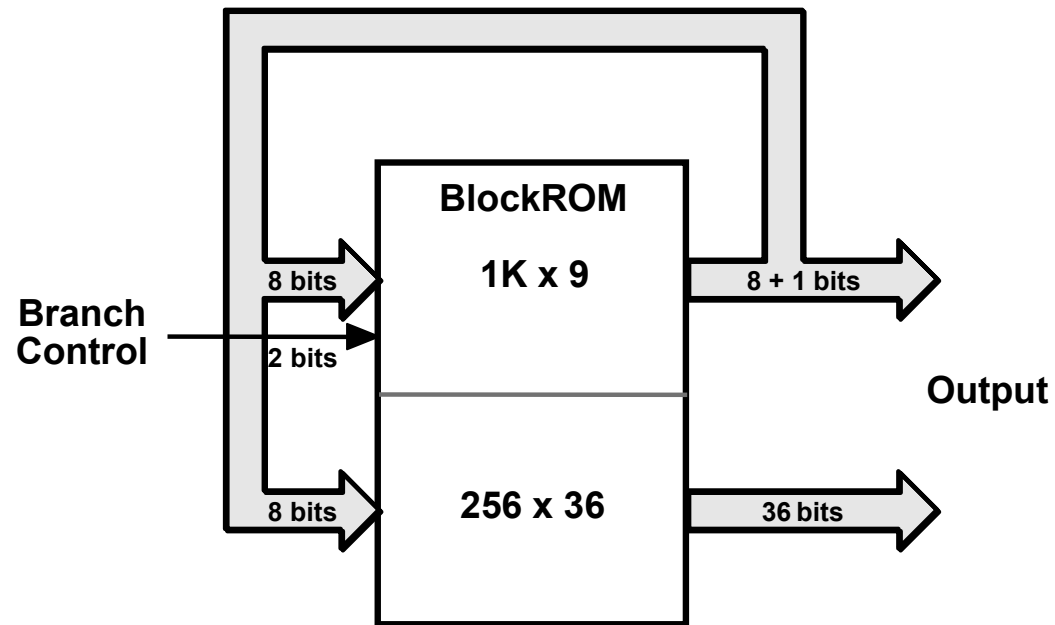
- ◆ **Virtex-II has 18K bits per BlockROM**
- ◆ **Counters**
 - *One 20-bit Gray counter*
 - *One 6-digit decimal counter*
 - using one additional CLB
 - *One 20-bit binary counter*
- ◆ **Finite State Machines (FSMs)**
 - *Two 5-input 64-state state machines*
 - *Two 4-input 128-state state machines*

Fast FSM in 1/2 BlockROM...



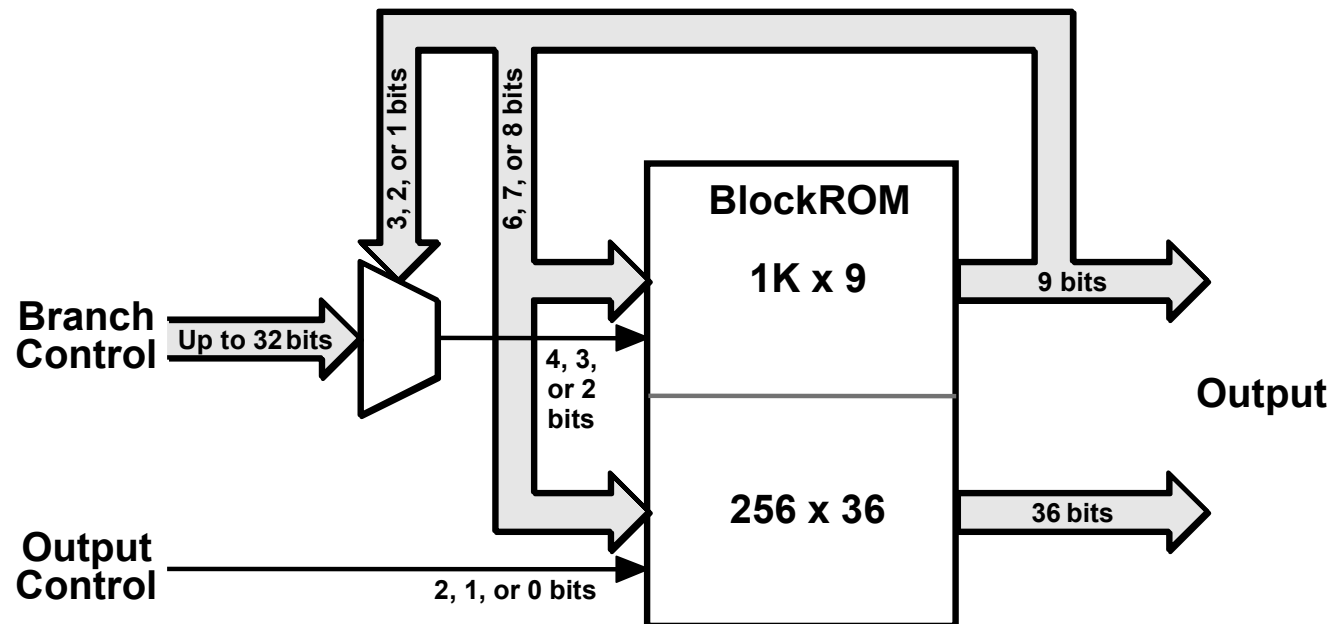
- ◆ **256 states, 4-way branch, 150 MHz operation**
 - or 128 states, 8-way branch, same speed
 - or 64 states, 16-way branch, same speed

...plus 36 Additional Outputs



- ◆ **36 additional parallel outputs**
— *from the other half of the BlockRAM (BlockROM)*

...plus Many Control Inputs



64, 128, or 256 states with multi-branch capability

36 freely assigned + 8 encoded outputs

optional multiplexed control inputs

All in one BlockRAM plus two CLBs

BlockROM Code Converters

- ◆ **Virtex-II has 18K bits per BlockROM**
- ◆ **High-resolution simultaneous Sine and Cosine table**
- ◆ **Two 11-bit binary to 4-digit BCD converters**
 - *Binary and BCD have identical LSB*
- ◆ **Two telecom 8-bit μ -law / A-law to linear converters**
- ◆ **Two 3-digit BCD to 10-bit binary converter**
 - *BCD and binary have identical LSB*
- ◆ **Wide-input Wallace-tree adder in multiple BlockROMs**



Designing for Low Power

Designing for Low Power Consumption

- ◆ To extend battery life, or
- ◆ To reduce chip temperature and cooling requirements
 - $T_{jmax} = 125 \text{ degr.C}$ (150 degree.C in ceramic)
 - Delays increase 0.35% / degr.C
above the guaranteed 85 degr.C junction temperature
- ◆ Use the free Xilinx Power Estimator
 - <http://www.xilinx.com/cgi-bin/powerweb.pl>

Power is proportional to CV^2f
Minimize all three !

Designing for Low Power

- ◆ **Clock Power + I/O Power + Logic Power**
- ◆ **Clock Power**
 - Minimize the number of high-speed clock nets
 - Use DLLs for phase-aligned sub-clocks
 - Using CE does not reduce clock power
- ◆ **I/O power**
 - Avoid wasted current in inputs driven with marginal voltage, use fast, full-swing input signals
 - Use output registers to avoid output glitches
every unnecessary transition wastes power !

Low Logic Power

- ◆ **Control Vcc tightly**
 - *Power is proportional to V_{cc}^2*
- ◆ **Minimize logic transitions and glitches**
- ◆ **Optimize counters:**
 - *Gray and Johnson are best*
 - *Binary counters double the power*
 - *Linear Feedback Shift Register are even worse*
- ◆ **Minimize internal node capacitance**
 - *Use aggressive timespecs*
 - *Design for the highest speed possible, **even if not needed !***
This assures lowest interconnect capacitance and provides the lowest power at the lower clock frequency



Designing for Radiation Tolerance

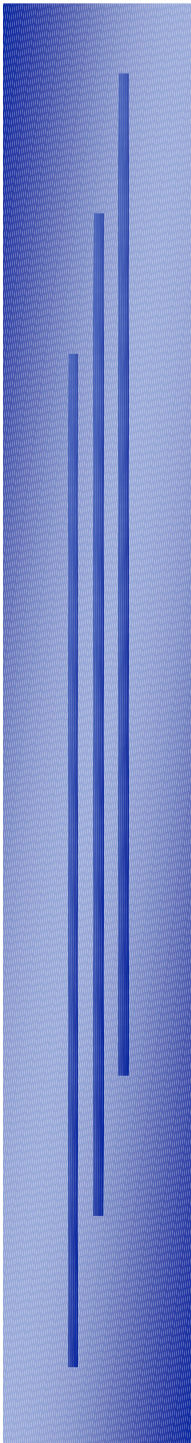
Radiation Effects

- ◆ **Latch-up**
 - *can be destructive, always requires Vcc recycling*
- ◆ **Total dose**
 - *causes premature aging*
- ◆ **Single Event Upset**
 - *tips over latch or flip-flop (soft error)*
- ◆ **Use QR-series parts: XQR4036XL or XQVR300**
 - *Functionally identical with namesake parts*
 - *Using epitaxial layer to avoid latch-up*

Designing for Radiation Tolerance

- ◆ **XQR4036XL and XQVR300 use an epitaxial layer process that provides latch-up immunity**
 - *beyond 120 MeV cm²/mg @ 125 degree C*
- ◆ **These devices tolerate total ionizing dose:**
 - *60krads(si) for XQR4036XL*
 - *100krads(si) for XQVR300*
 - *300krads(si) for XQVR300-E*
- ◆ **Single-Event Upsets increase w. smaller geometries**
 - *can be detected by readback,*
 - *mitigated by scrubbing or partial reconfiguration*
 - *mitigated by Triple Redundancy, no functional interruption*

See: www.xilinx.com/products/hirel_qml.htm



Asynchronous Issues

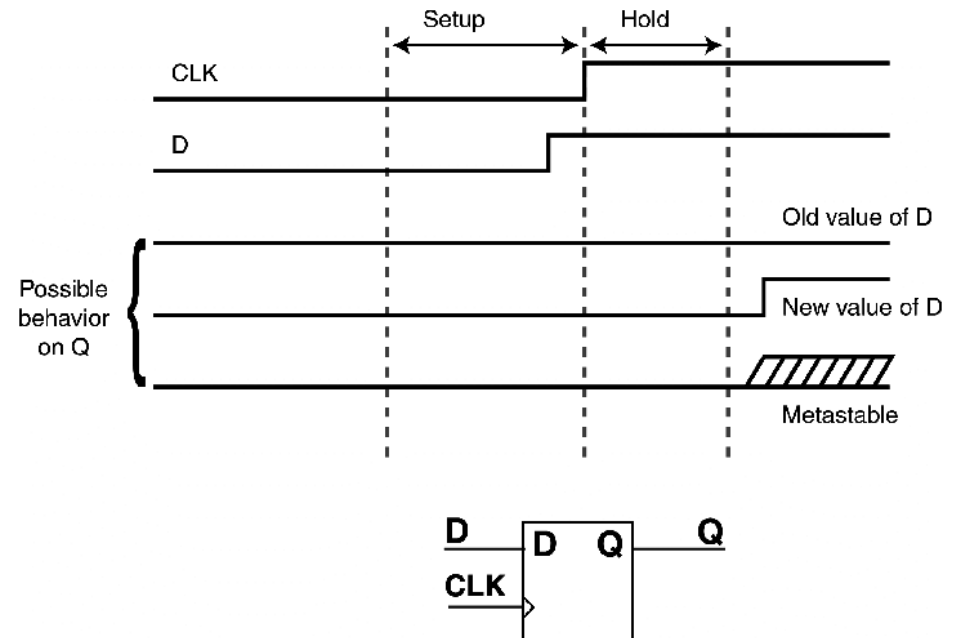
Understanding Asynchronous Design Issues

- ◆ **Most systems operate synchronously inside**
 - *But asynchronous inputs are a fact of life*
- ◆ **Occasionally, an asynchronous input will cause a flip-flop to go metastable**
 - *This is a rare, but unavoidable, probabilistic event*
- ◆ **Solution:**
 - *Faster flip-flops recover faster*
 - *Double-synchronization reduces probability*

Awareness and understanding are crucial

Setup and Hold Time Violations

- ◆ Violations occur when the flip-flop input changes too close to a clock edge
- ◆ Three possible results:
 - Flip-flop clocks in old data value
 - Flip-flop clocks in new data value
 - Flip-flop output becomes **metastable**



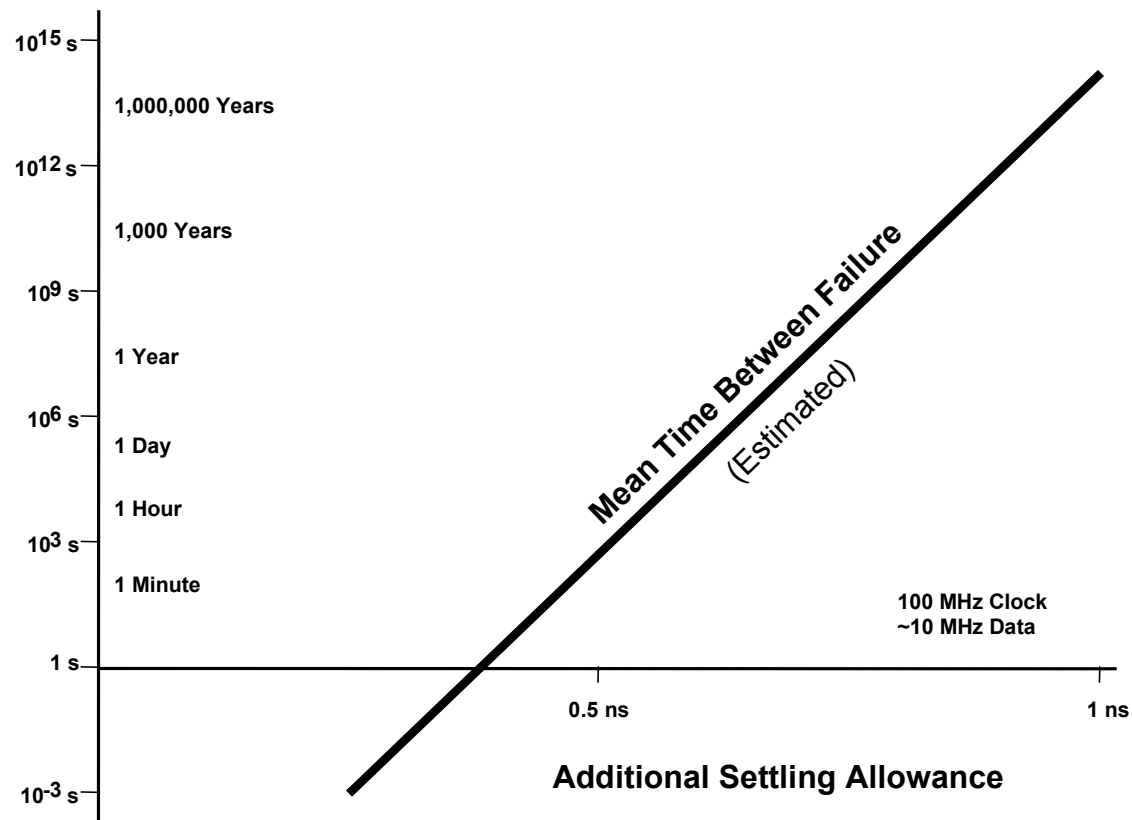
Metastability

- ◆ **Caused by asynchronous data input**
 - *Violates set-up time requirement*
 - *Usually gets synchronized in the flip-flop without problem*
- ◆ **But if data changes within a tiny set-up time window**
 - *Then the flip-flop can go metastable*
 - *Resulting in unpredictable delay to reach stable 1 or 0*
- ◆ **The 0 or 1 uncertainty is irrelevant**
 - *The slightest timing change would have given correct 1 or 0*
- ◆ **The unpredictable delay is the real problem**
 - *It can violate set-up times further down in the system, causing erratic operation or even crashes*

Mean Time Between Failure

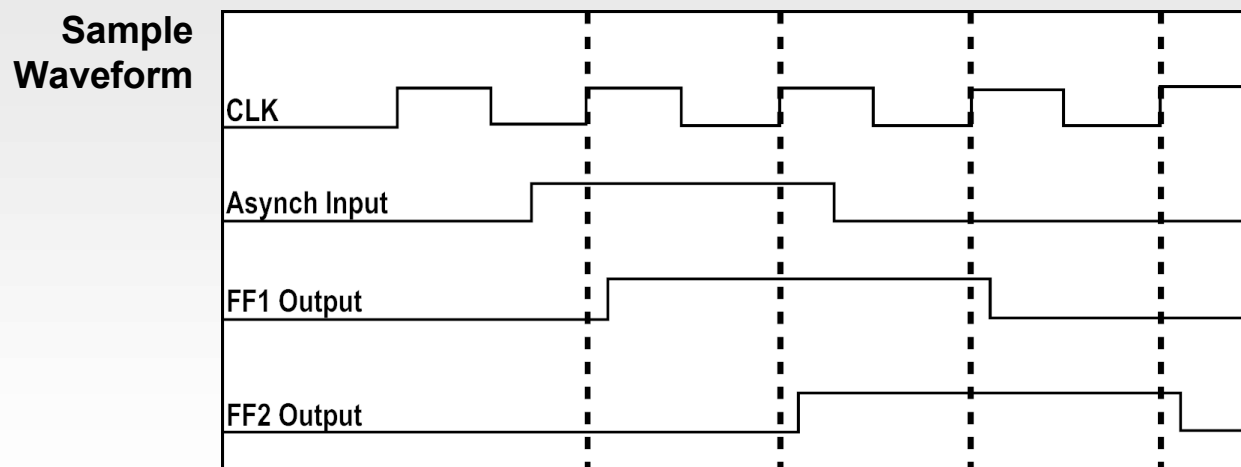
- ◆ **Measure MTBF = f (extra acceptable delay, delta t)**
 - *Assumes a given clock and data rate*
- ◆ **MTBF is exponential function of delta t**
 - *Slope determined by gain-bandwidth product in master latch*
- ◆ **Modern CMOS resolves extremely fast**
 - *but modern system also usually have little time slack*
- ◆ **The problem is as unavoidable as death and taxes**
 - *but probability can be reduced by design*

Metastability Data

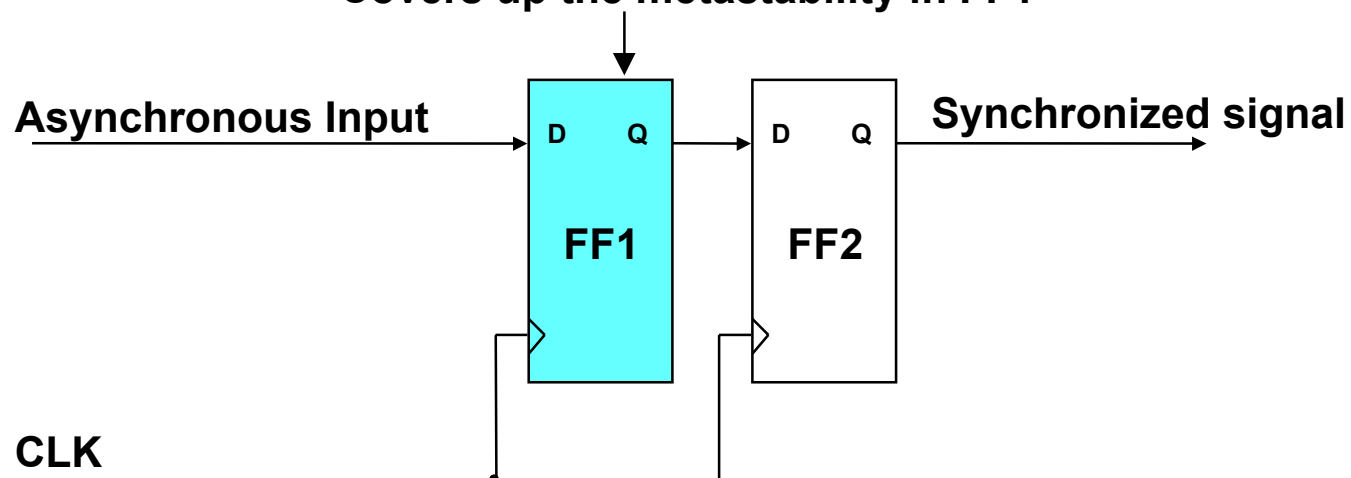


- ◆ Website (will be updated soon with Virtex-II data):
— <http://www.xilinx.com/xapp/xapp094.pdf>

Synchronization Circuit



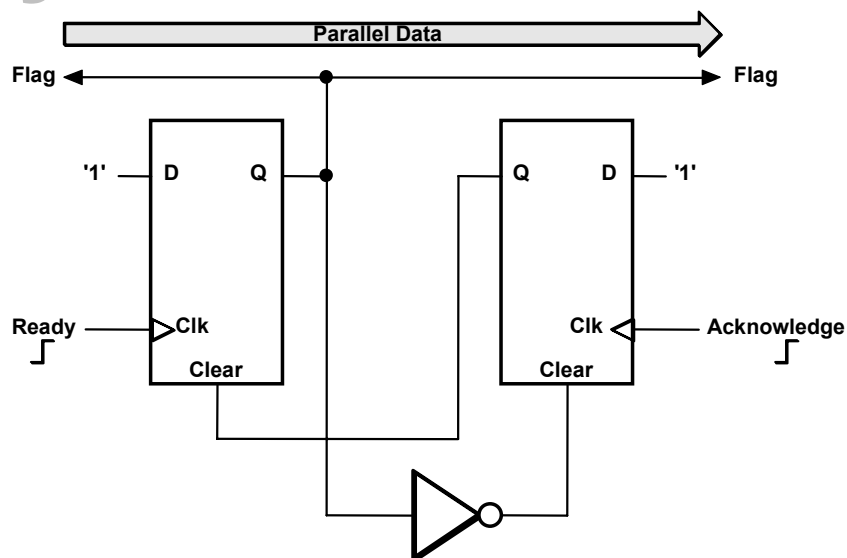
Covers up the metastability in FF1



Moving Data Across Asynchronous Clock Boundaries

- ◆ Worst-case timing happens, sooner or later
- ◆ ***Murphy never sleeps !***
- ◆ Never use multiple parallel flip-flops to synchronize an asynchronous input signal
 - *Always synchronize at a single point*
- ◆ Don't try to synchronize parallel data
 - *Use the methods described on the following slides*
 - *The problem is data corruption, not metastability*
- ◆ Use cascaded stages to combat metastability
- ◆ Website:
 - <http://www.isdmag.com/editorial/2000/design0003.html>

Moving Parallel Data with Asynchronous Handshake



- ◆ **Transmitter: Data available raises Ready, sets Flag**
 - *Receiver scans F, accepts parallel data, raises Acknowledge*
- ◆ **Acknowledge sets flip-flop, which resets Flag**
 - ***Benign race condition** between flip-flops*
- ◆ **Both sides must observe and obey the Flag**

Asynchronous FIFOs

- ◆ **Parameters: width, depth, clock frequency**
- ◆ **Data path = dual-ported BlockRAM**
- ◆ **Control = 2 addresses + Full + Empty**
- ◆ **Synchronous control is very simple:**
 - *Two counters + trivial synchronous state machines*
- ◆ **Asynchronous control is very tricky**
 - *Asynchronous addresses must control FULL and EMPTY*

BlockRAMs are Ideal for FIFOs

- ◆ **Asynchronous FIFO with Full and Empty flags**
 - *Implemented in Virtex-II*
 - *>200 MHz operation for both write and read simultaneously*
- ◆ **2K deep FIFO needs:**
 - *1 BlockRAM per 9 bit width*
 - *Grey-coded address counters in 3 CLBs*
 - *plus 3 CLBs for Full and Empty control*
- ◆ **4K deep FIFO needs:**
 - *1 BlockRAM per 4 bit width*
 - *Grey-coded address counters in 3 CLBs*
 - *plus 3 CLBs for Full and Empty control*

Full and Empty Control

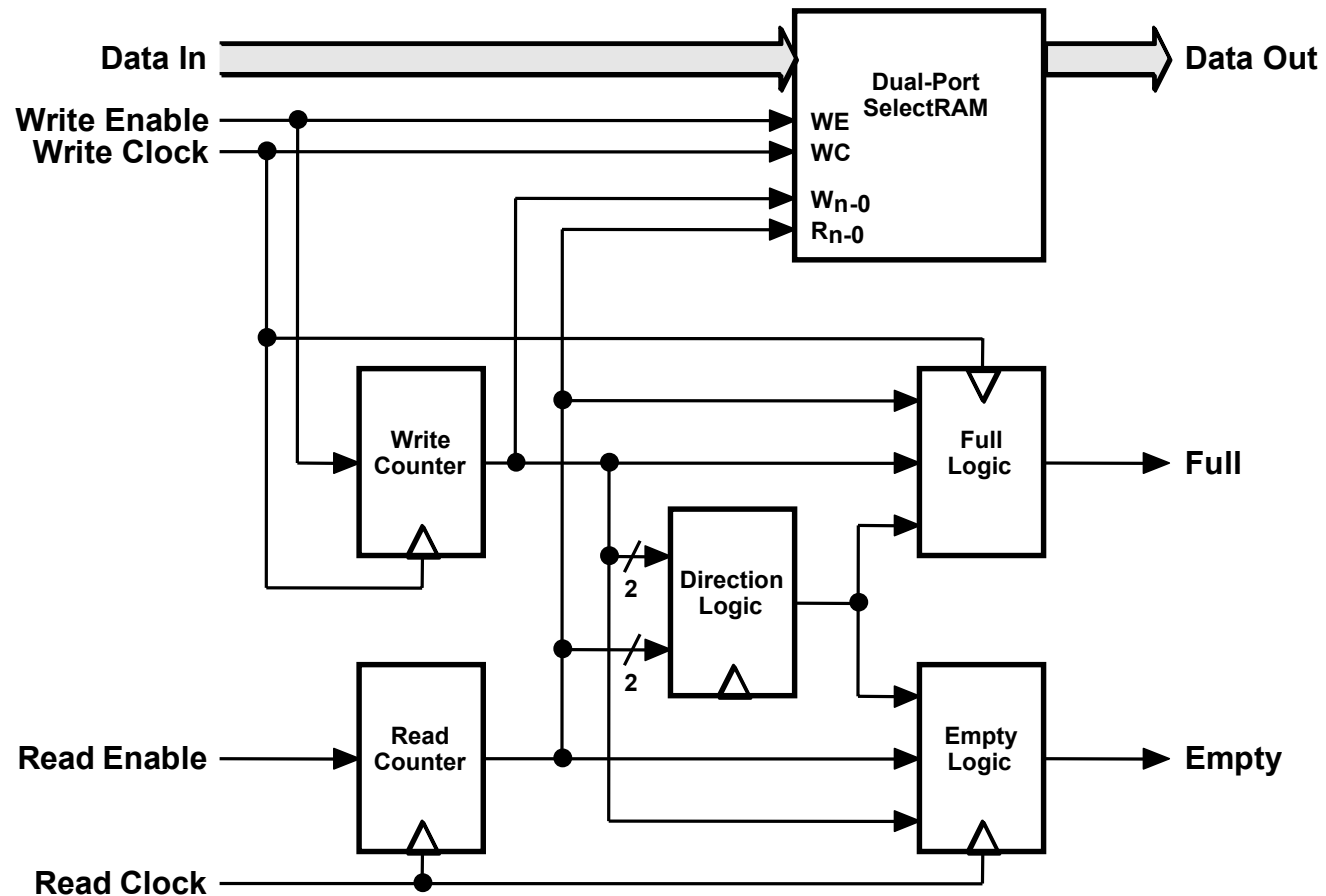
Identity-compare write and read addresses

— *identical addresses mean either Full or Empty*

Two problems:

- ♦ Comparing two asynchronously changing binary addresses can cause unpredictable glitches
- ♦ Distinguish between Full and Empty
 - *both are indicated by address identity*

FIFO Block Diagram



Gray-Coded Addresses

- ◆ **Only one bit per address changes any time**
 - *Never any glitches from the identity comparator*
- ◆ **Implementation:**
 - *Build **binary** counter*
 - *Generate XOR of two adjacent **D-inputs***
 - *Feed these XORs to a register D-input = Gray code*
 - *MSB binary = MSB Gray*
- ◆ **Advantage:**
 - *Very fast and easily expandable, binary as a bonus*
 - *Takes advantage of the fast carry structure*

No pipeline delay, but twice the binary counter cost

Separate Full from Empty

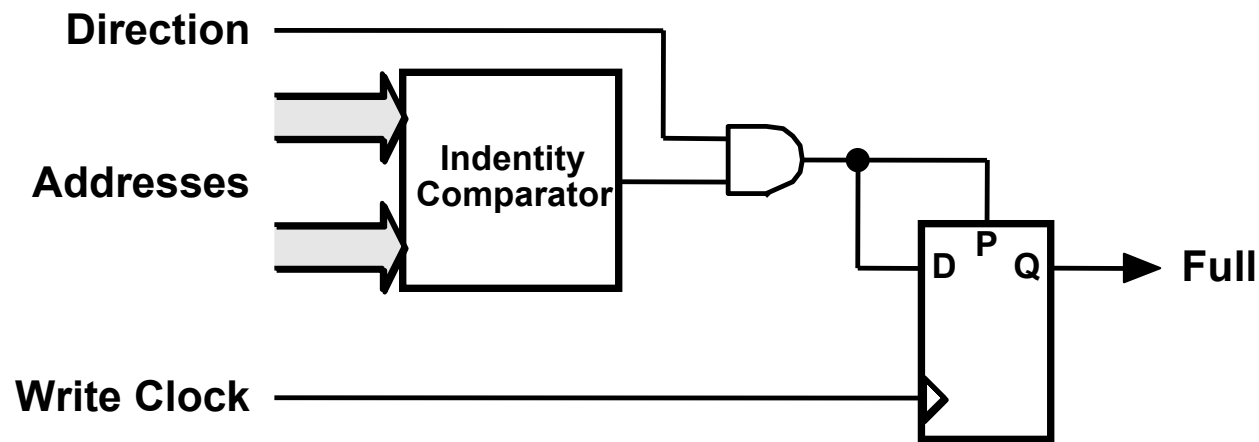
- ◆ **Divide address space into 4 quadrants, defined by the counter MSBs**
 - *This works in binary as well as in Gray*
- ◆ **Monitor the quadrant relationship of the write and read address counters**
- ◆ **Set a flag to distinguish between potentially going Full or Empty**
 - *include this in the address identity comparator*

Synchronize to the Proper Clock

- ◆ **FULL must be synchronous to write clock**
 - *Read is not concerned with fullness*
- ◆ **EMPTY must be synchronous to read clock**
- ◆ **Leading edges are naturally synchronous:**
 - *Full is the result of a write clock*
 - *Empty is the result of a read clock*

Trailing edges are caused by the other clock

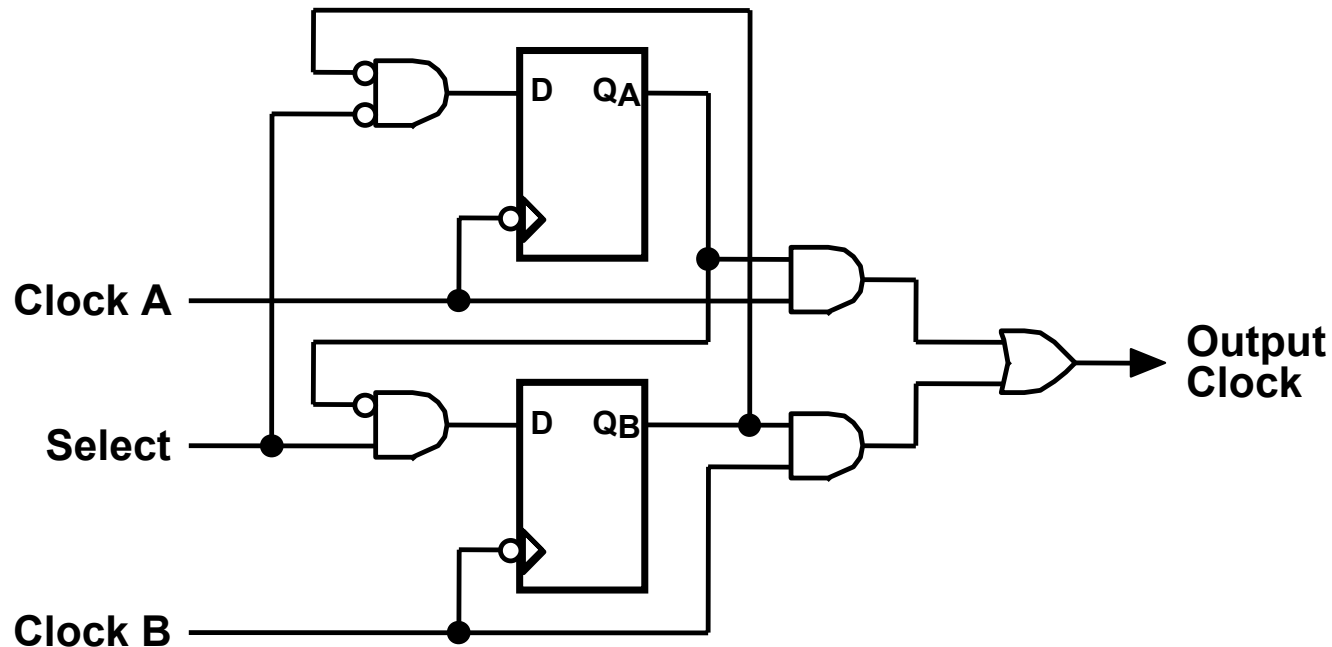
Synchronizing the Trailing Edges



- ◆ **Combinatorial FULL is the result of a write.**
 - *Use it to asynchronously preset a flip-flop.*
 - *Use it also as D-input, clocked by the write clock.*
 - *Do the symmetrically equivalent for Empty using Read Clock*

This synchronizes both edges to the proper clock.

Asynchronous Clock MUXing



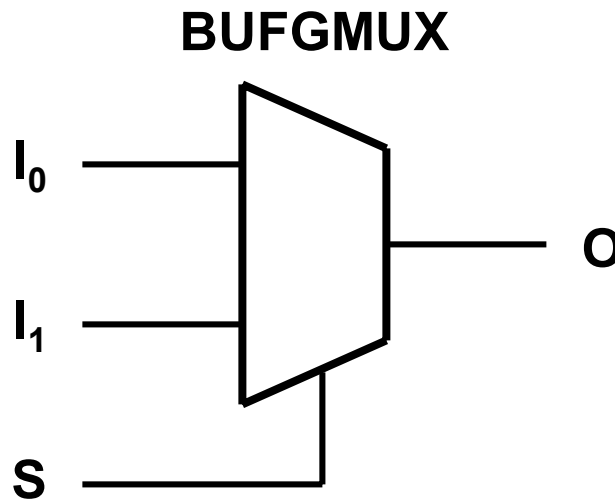
- ◆ This circuit waits for the present clock to go Low

— *Output then stays low until the new clock is Low*

Guaranteed to switch glitch-free, no runt pulses

— http://www.xilinx.com/xcell/xl24/xl24_20.pdf

Virtex-II Clock Multiplexer



- ◆ **Each global clock buffer is a mux**
 - *can switch between 2 clock sources*
 - *configured for rising or falling edge*
- ◆ **Can also do clock gating (enable)**

Dangerous stuff, but these circuits do it safely

Conclusions

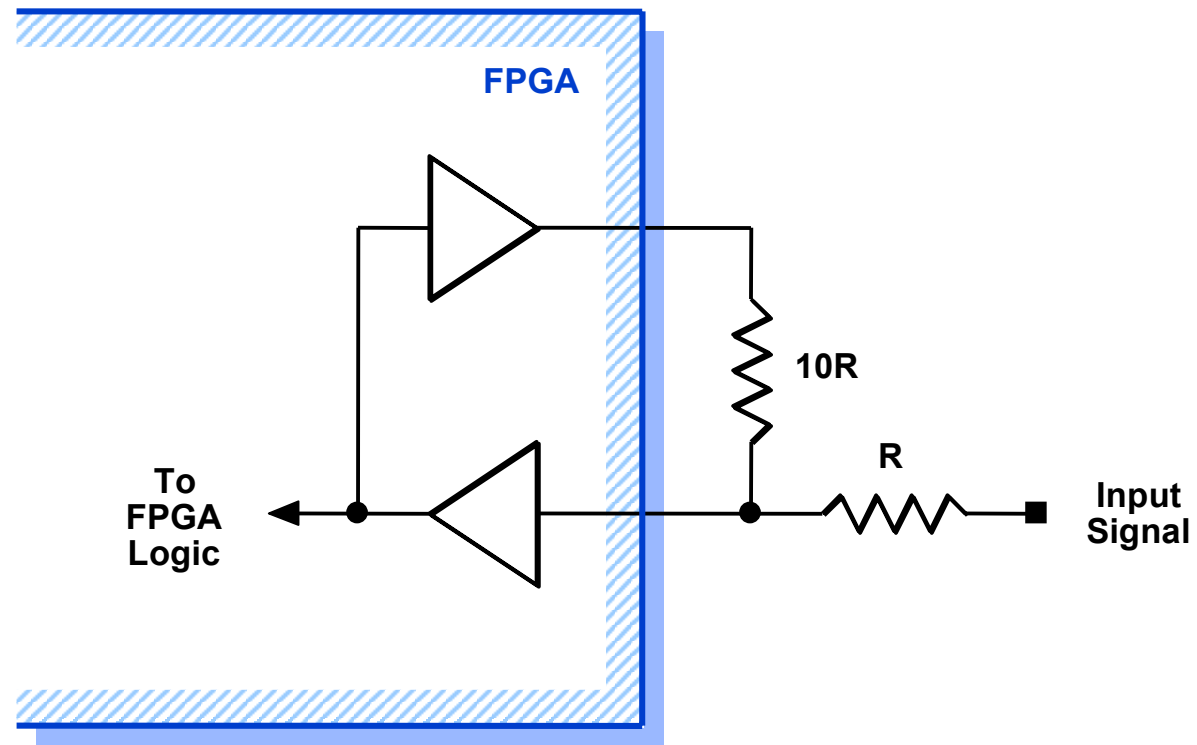
- ♦ **Asynchronous data transfer is dangerous**
 - *but not if you understand the issues and know how to design around them*
- ♦ **Clock gating is unhealthy**
 - *but not if you use smart circuits*
- ♦ **Metastability can hurt very badly**
 - *but only if inside a very tight timing budget*

Modern CMOS resolves very fast (within a few ns)



Tips and Tricks from the Xilinx Archives

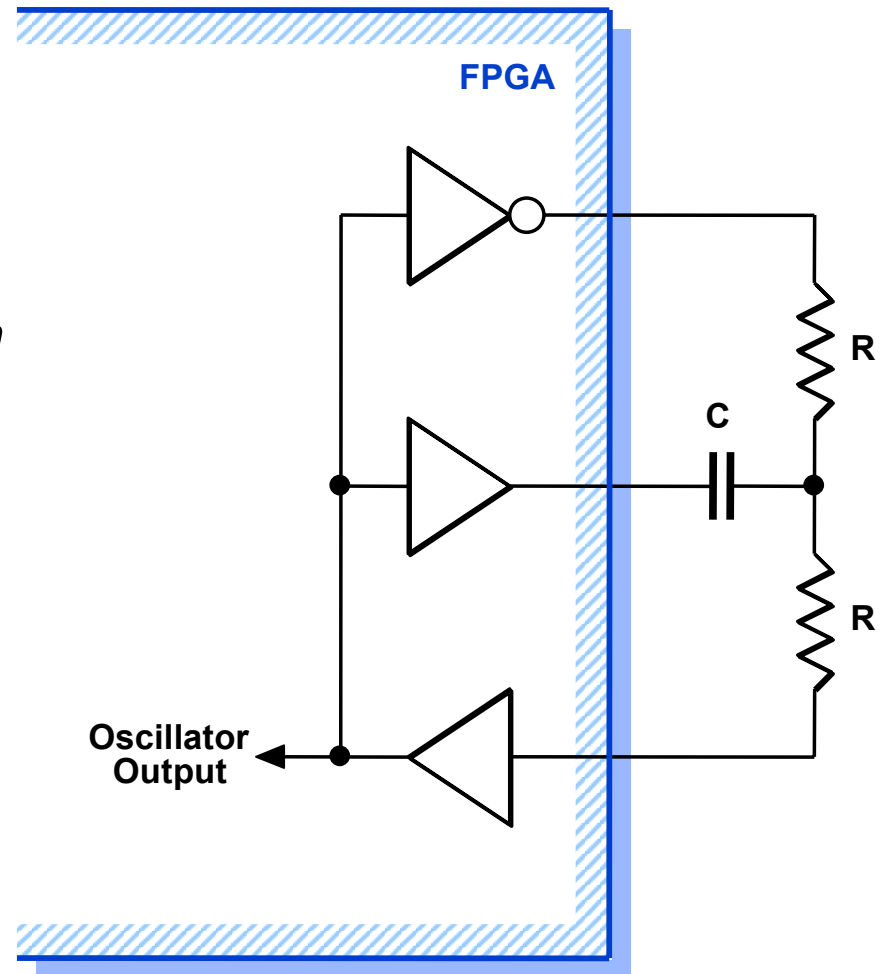
Schmitt Trigger



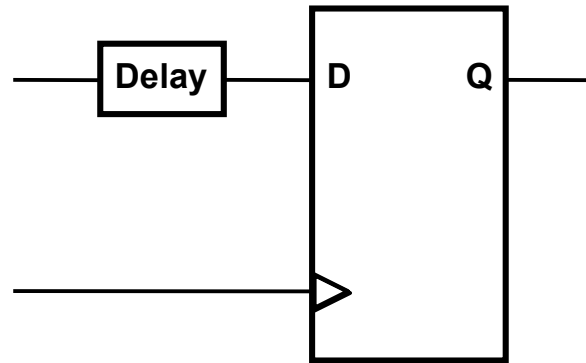
- ◆ Hysteresis = 10% of Vcc

RC Oscillator

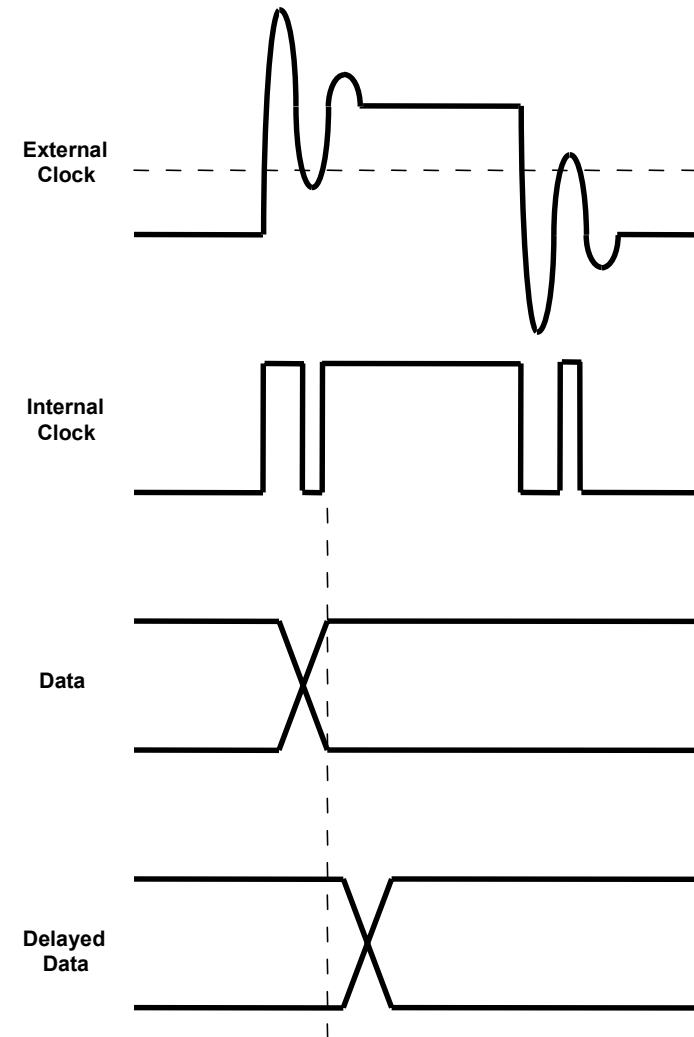
- ◆ **Wide frequency range, Hz to MHz**
 - *100 Ohm to 100 kilohm*
 - *100 pF to 1 uF*
- ◆ **Reliable start-up is absolutely guaranteed**
- ◆ **Oscillator can be started and stopped internally**



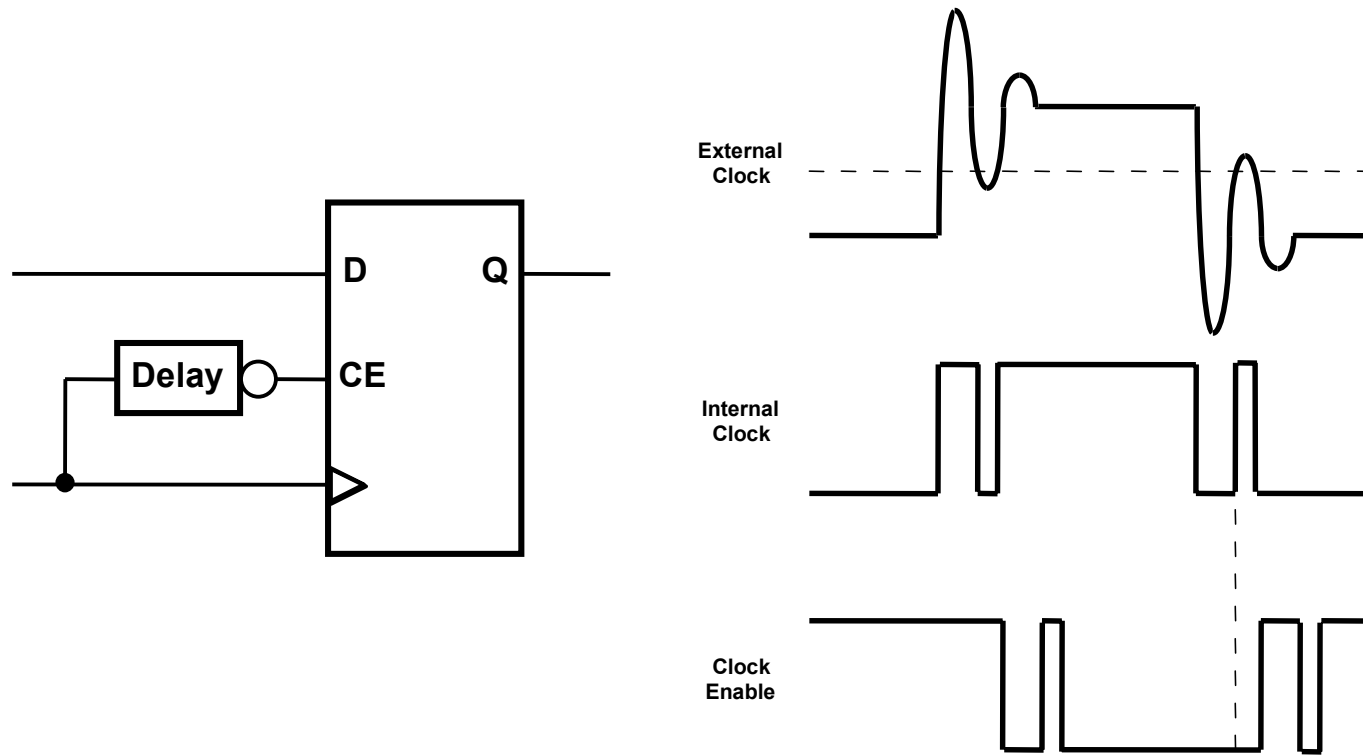
Coping with Clock Reflections



- ◆ **Problem: Double pulse on the active edge**
- ◆ **Solution: Delay D, to prevent the flip-flop from toggling soon again**



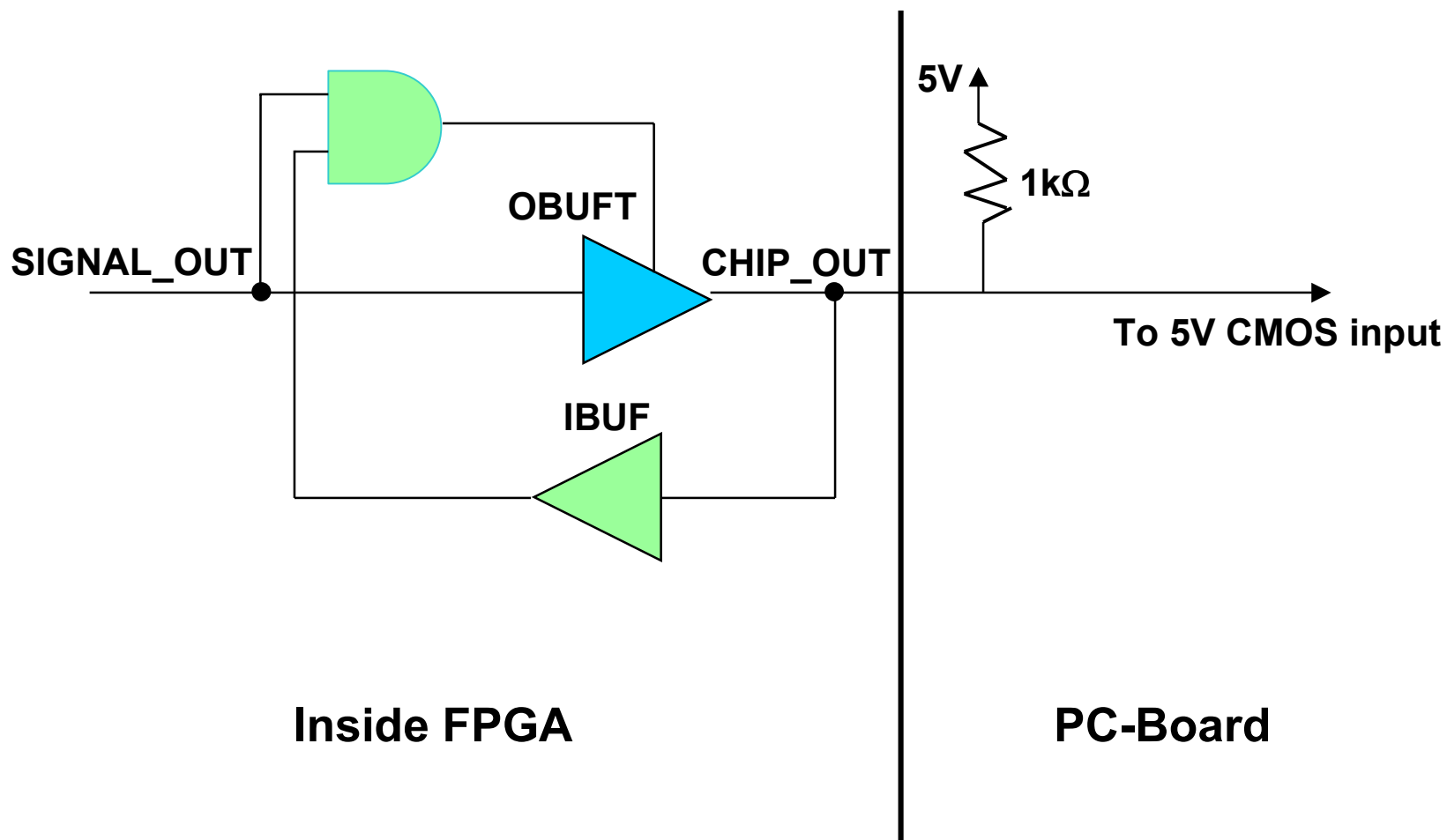
Coping with Clock Reflections



- ◆ **Problem: Double pulse on the inactive edge**
- ◆ **Solution: Disable flip-flop, by using the clock level**

— http://www.xilinx.com/xcell/xl34/xl34_54.pdf

5V-Tolerant 3.3V Output Driving 5V CMOS-Level Input





Coming Applications

Pulse Generator and Counter in one XC2V40

- ◆ **200 MHz Pulse Generator**

*Digitally-controlled Variable-Frequency Oscillator,
using DCM and partial reconfiguration of M and D values
1024 frequencies spaced between 100...200 MHz
24 binary divider stages down to 6 Hz
Controlled by two push-buttons, up and down
Frequency indicated by on-chip frequency counter*

- ◆ **1 GHz Frequency Counter**

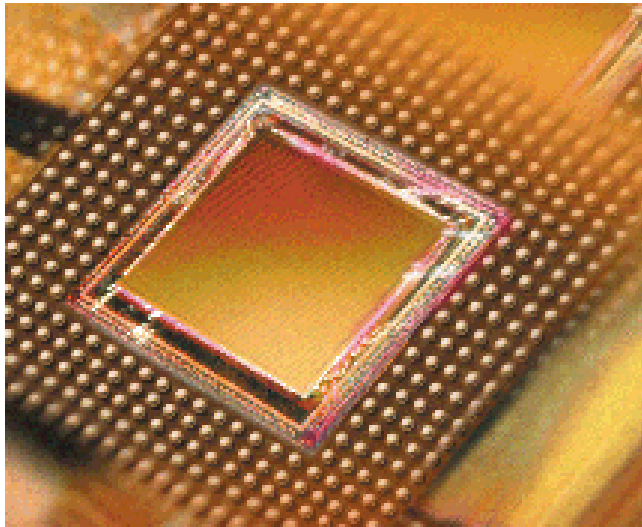
*1 ns resolution through DCM input toggle flip-flop
1 second time base, derived from 100 MHz oscillator
8 digit indication, even down to 1 Hz*

Floating-Point Adder/Multiplier

- ◆ Use the Virtex-II multipliers to multiply and also shift
- ◆ 16 x 16 bit combinatorial multipliers (40 in XC2V1000)
 - Use four multiplier for 32 x 32 bits
 - Use multipliers to perform the normalizing shift operation
- ◆ Allows non-standard optimization of cost vs speed
 - e.g. 16-bit mantissa, 4-bit exponent
 - Special code for Zero is needed, but
is “graceful underflow” really required ?

Fast floating point is now feasible in FPGAs

FPGAs circa 2005



50 Million system gates

2 Billion transistors on one chip

70-nm process technology

10-layer Cu technology

Hard and soft IP blocks

1 GHz embedded processor

Mixed-signal Intellectual Property

10-Giga-bps I/O channels

List of Good URLs

- ◆ **www.xilinx.com**
- ◆ **www.xilinx.com/support/sitemap.htm**
 - www.xilinx.com/products/virtex/handbook/index.htm
 - www.xilinx.com/support/techxclusives/techX-home.htm
 - www.xilinx.com/support/troubleshoot/psolvers.htm

General FPGA-oriented Websites:

- www.fpga-faq.com
- www.optimagic.com

Newsgroup: comp.arch.fpga

All datasheets: www.datasheetlocator.com

Search Engine (*personal preference*): www.google.com

Beyond Bigger, Faster, Cheaper ...

On-chip RAM

Efficient Arithmetic

Intelligent Clock Management

Multi-standard I/O, Built-In Termination

Giga-bit/s I/O, Embedded Microprocessors

***FPGAs have evolved from glue logic
to cost-effective system platforms***