CSE 453

The FPGA

The Field Programmable Gate Array (FPGA)

- Types
  - SRAM based
  - Antifuse based
  - Flash based
- Elements in an FPGA
  - Combinational Logic
  - Interconnections
  - I/O Pins

SRAM Based FPGAs

- Configuration held in static memory (SRAM)
  - SRAM output continuously drives circuit
  - DRAM not used due to required refresh
- Advantages
  - Easily Reprogrammable
    - Ideal for prototyping
  - Dynamically Reconfigurable
    - Can be reprogrammed while system is in operation
  - Fabrication uses standard VLSI processes
- Disadvantages
  - Power Consumption
  - Bits Susceptible to Theft

History of the FPGA

- Inventor - Ross Freeman
- Early FPGA Manufacturers
  - Xilinx
  - Altera
  - Actel
- Early Uses of the FPGA
  - Glue Logic
  - Prototyping Devices
- Today's Uses of the FPGA
  - Wide Variety of Digital Systems
    - High speed telecommunications equipment
    - Video accelerators in home personal video recorders

FPGA Characteristics

- Standard Parts
  - Not designed for particular function
  - Programmed by customer for particular purpose

- Implement Multilevel Logic
  - ☞ Logic blocks can be connected in networks of arbitrary depth
    - ↳ Compare to PLDs
      - ✓ Two levels of NAND/NOR

FPGA Programming

- FPGA "*program*" is know as a "*personality*" interwoven into a logic structure
  - ☞ Programming directly implements logic functions & interconnects
  - ☞ Does NOT fetch instructions

- Compare to Stored Program Computer

Programming the FPGA

- Permanently Programmed vs. Reprogrammable
- Reprogrammable devices are know as reconfigurable devices
  - ☞ Ideal for
    - ↳ Prototyping
    - ↳ Machines that serve different functions at different times

FPGA Logic

- Fine-Grained Logic
  - ☞ Logic elements implement a smaller function (with a handful of gates) & a register
- Coarse-Grained Logic
  - ☞ Logic elements implement a larger function (such as an ALU) & a register
- Platform FPGAs
  - ☞ Newer category
  - ☞ Incorporates several different types of structures
  - ☞ Large systems can be implemented by mapping parts on type best suited for it
  - ☞ Typically includes CPU so some functions can be run in software
    - ↳ May also include specialized bus logic

FPGA vs. Custom VLSI

- Application Specific Integrated Circuit (ASIC)
  - ☞ The alternative to the FPGA
  - ☞ Uses predesigned layout as opposed to a full custom layout
    - ↳ Few large digital chips other than microprocessors require a significant amount of custom layout
- Disadvantage of ASICs
  - ☞ Time
    - ↳ Requires several months before it can even be tested
  - ☞ Cost
    - ↳ More expensive when manufactured in low quantities

- Advantages of ASICs
  - Faster*
  - Power Consumption*
  - Cost
    - Less expensive when manufactured in high quantities

* Faster & more energy efficient since they are designed for a particular purpose

Trends Resulting in Use of FPGAs Over ASICs for Custom Logic

- Moore's Law
  - Using additional transistors, design process can be simplified
- Skyrocketing Mask Costs

Technology	Mask Cost
0.09 µm	$1,000,000
0.18 µm	$250,000
0.25 µm	$120,000
0.35 µm	$60,000

Table 1-1, page 13, Wayne Wolf, ***FPGA-Based System Design***, Pearson Education, Inc. (Prentice Hall), 2004

  - FPGA allows cost to be absorbed across a larger population

FPGA-Based System Design

- Goals & Techniques
  - Performance
    - Latency, Throughput
  - Power/Energy
    - Power Budget
      - ✓ Battery Powered Systems
        - Battery Capacity
      - ✓ Power-Grid Powered Systems
        - Cooling
  - Design Time
    - FPGA Allows Quick Prototyping
    - Use of FPGA Final Design
  - Design Cost
    - Reduction in Design Time Reduces Cost
    - Required Support Tools Less Expensive Than Custom VLSI Tools
  - Manufacturing Costs
    - Cost of replicating system many times
    - Generally FPGA is more expensive than ASIC
      - ✓ Due to overhead of programming
    - The fact that FPGAs are standard parts helps to reduce costs

FPGA Fabrics

- *aka*, FPGA Structures
- Major Elements
  - Combinational Logic
  - Interconnects
  - I/O Pins
- General Structure

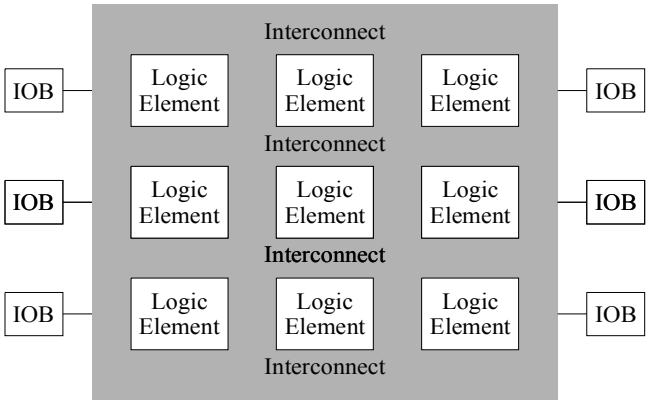

Figure 3-1, page 106, Wayne Wolf, ***FPGA-Based System Design***, Pearson Education, Inc. (Prentice Hall), 2004

  - Logic Element (LE)
    - *aka*, Combinational Logic Block (CLB)
  - I/O Block (IOB)
    - Consists of I/O Blocks & Pins
- Interconnects
  - Connections made between LEs and wires
  - Wires organized into "*wiring channels*" or "*routing channels*" which run vertically & horizontally through the chip
    - Segments of varying length vs. offset segments

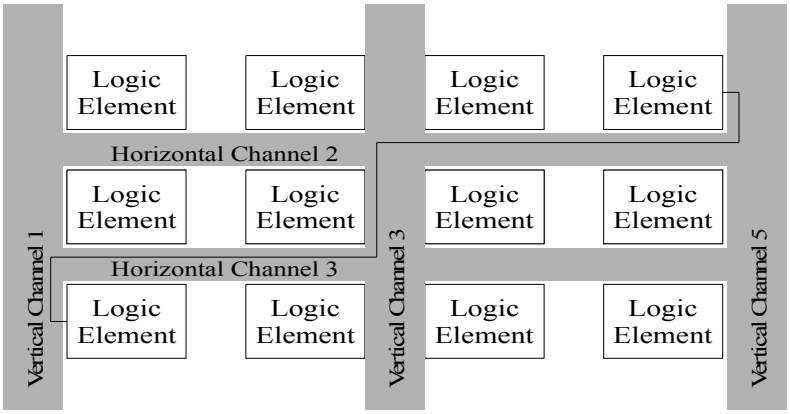

Figure 3-2, page 107, Wayne Wolf, ***FPGA-Based System Design***, Pearson Education, Inc. (Prentice Hall), 2004

- Interconnects

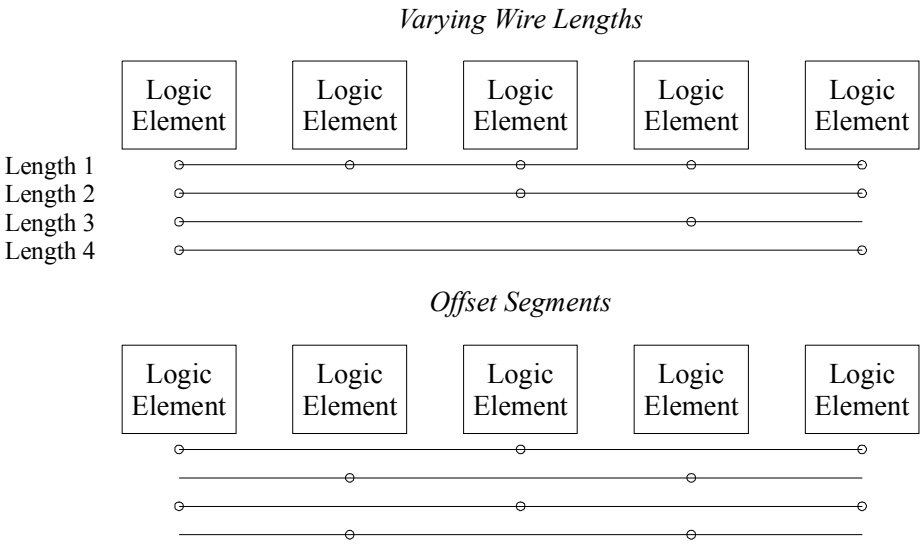

Figure 3-3, page 108, Wayne Wolf, ***FPGA-Based System Design***, Pearson Education, Inc. (Prentice Hall), 2004

FPGA Configuration

- FPGA must be "*programmed*" or "*configured*"
  - All major elements must be configured
    - Logic
    - Interconnects
    - I/O Pins
- Three Major Circuit Technologies for Configuring FPGA
  - SRAM
  - Anitfuse
  - Flash
- Consider a system designer who wishes to use an FPGA
  - What questions need be addressed?
    - How much logic can fit on the FPGA?
      - ✓ Depends on FPGA architecture, logic, design process
    - How many I/O pins are available?
      - ✓ Fairly straightforward answer
    - How fast will the design run?
      - ✓ Depends on FPGA architecture, logic, design process
  - These questions lead to the following questions when selecting an FPGA:
    - How many logic elements should the FPGA have?
    - How large should each element be?
    - How much interconnect should it have?
    - How many types of interconnect structures should it have?
    - How long should each type of interconnect be?
    - How many pins should the FPGA have?

The SRAM-Based FPGA

- State of memory continuously and directly controls circuit being configured
- Advantages of Using SRAM
  - Easily reprogrammable
    - Hence, common choice for prototyping
  - Dynamically reconfigurable systems
    - Can be reprogrammed during system operation
  - FPGA circuits can be fabricated with standard VLSI processes
  - Does not need refreshing
    - Unlike DRAM
- Disadvantages of Using SRAM
  - Power dissipation
  - Bits susceptable to theft
  - Large number of bits required to program FPGA
    - Each LE requires many bits
    - Each programming interconnection point requires own bit
- Logic Elements
  - Basic method used to build CLB is a lookup table (LUT)
  - Lookup Table (LUT)
    - SRAM used to implement a truth table
    - Address represents a combination of inputs
    - *n*-input function requires 2^n locations
    - 2^{2n} functions can be implemented
    - Delay is static
      - ✓ Example
        - 4-input XOR has same delay as a 4-input NAND
      - ✓ Note that static logic gate is generally faster than the LE
    - Memory
      - ✓ Latch or flip-flop is often included in LE

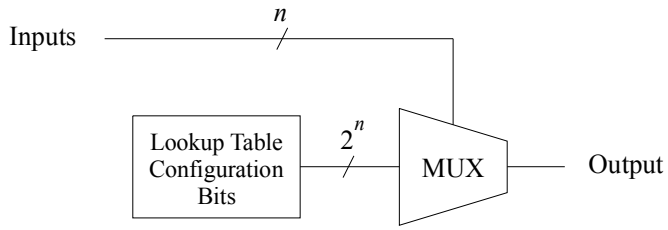

Figure 3-4, page 111, Wayne Wolf, ***FPGA-Based System Design***, Pearson Education, Inc. (Prentice Hall), 2004

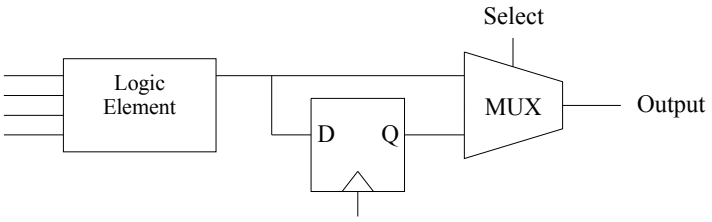

Figure 3-5, page 112, Wayne Wolf, ***FPGA-Based System Design***, Pearson Education, Inc. (Prentice Hall), 2004

- Complex Logic Elements
  - Many LEs also contain special circuitry for addition
    - Example
      - ✓ Including circuitry to implement carry chain for addition improves performance over standard lookup

- Spartan II CLB
  - Two identical slices, each with an LUT
  - Each slice includes two "*logic cells*" (LCs)

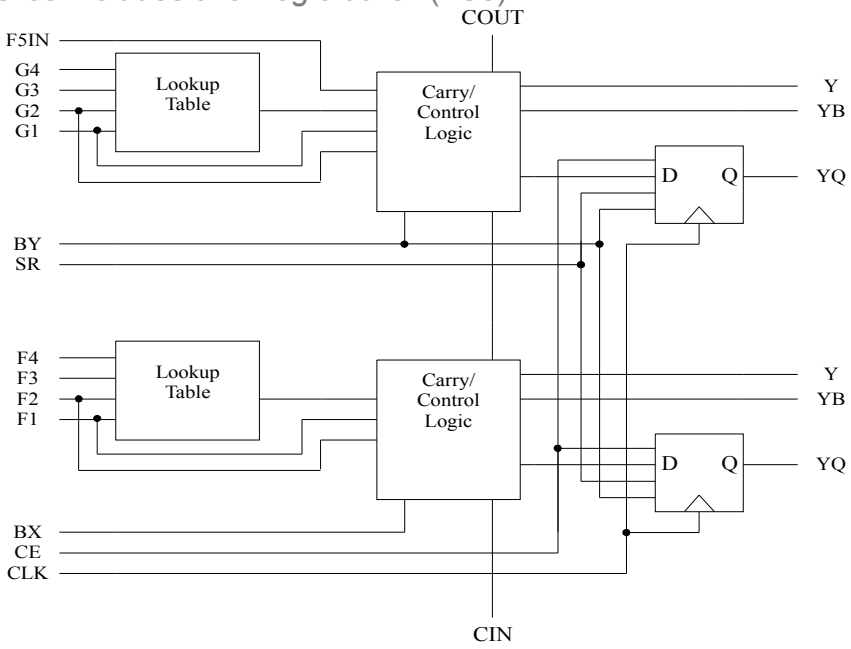

Figure with Example 3-1, page 113, Wayne Wolf, ***FPGA-Based System Design***, Pearson Education, Inc. (Prentice Hall), 2004

  - Building an Adder
    - SUM Generation
      - ✓ Carry/Control Logic also includes an XOR gate
      - ✓ Used to generate SUM
    - Carry Computation
      - ✓ LUT computes carry
  - Each slice includes multiplexor to combine results of 2-function generators in slice
    - Another multiplexor combines outputs of multiplexors in 2 slices, generating result for entire CLB
  - Registers can be configured as D-type flip-flops or as RS latches
  - Each register has clock & clock enable signals
  - Each CLB contains two, 3-state drivers (BUFTs) that can be used to drive on-chip busses

- Interconnection Networks
  - An interconnection point controlled by SRAM cell
    - Bidirectional Interconnection Point:

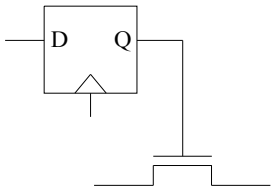

Figure 3-6, page 118, Wayne Wolf, ***FPGA-Based System Design***, Pearson Education, Inc. (Prentice Hall), 2004

      - ✓ Note the CMOS transistor acts as a pass transistor
  - "*Interconnection Point*" *aka*, "*Connection Box*"
  - Pass transistors are slow
    - Alternative programmable interconnection points exist which offer higher performance
      - ✓ Cost = Additional Chip Area
  - Performance
    - Programmable interconnects are slower than interconnects on a custom chip
      - ✓ Why?
        - Pass Transistor
        - Longer Wire Lengths
  - Wire Categories
    - Short wires used to connect local LEs
    - Global wires used for long distances
      - ✓ Fewer connection points reduce impedance
      - ✓ Repeaters can reduce effects of delay
    - Special Wires
      - ✓ Dedicated to clock distribution or other register control signals
- Spartan II Interconnect System
  - Interconnect Types
    - Local
    - General Purpose
    - I/O
    - Global
    - Clock
  - Local Interconnects
    - Connects LUTs, flip-flops, & acts as a general purpose interconnect
    - Internal CLB feedback
    - Direct paths for high speed communication between horizontally adjacent CLBs

- General Purpose
  - Bulk of routing
  - Types of general purpose interconnects
    - ✓ General Routine Matrix (GRM)
      - Switch matrix used to connect horizontal & vertical routing channels, connections between CLBs & routing channels
    - ✓ 24 Single Length Lines
      - Connects each GRM to 4 nearest GRMs to left, right, above, & below
    - ✓ 96 Hex Lines
      - Route GRM signals to GRMs 6 blocks away
      - Provide longer interconnects (contain buffering to increase drive capability)
      - 1/3 are bidirectional
      - 2/3 are unidirectional
    - ✓ 12 Long Lines
      - Span entire chip, both vertically & horizontally
- I/O
  - Run around edge of chip to allow interconnections to pins
- Global
  - Designed to distribute high-fanout signals (both clock & logic)
  - Primary global routing network
    - ✓ 4 dedicated global nets with dedicated input pins
    - ✓ Each net can drive all CLB, I/O registers, & block RAM clock pins
    - ✓ Clock distribution network is buffered to provide low delay & low skew

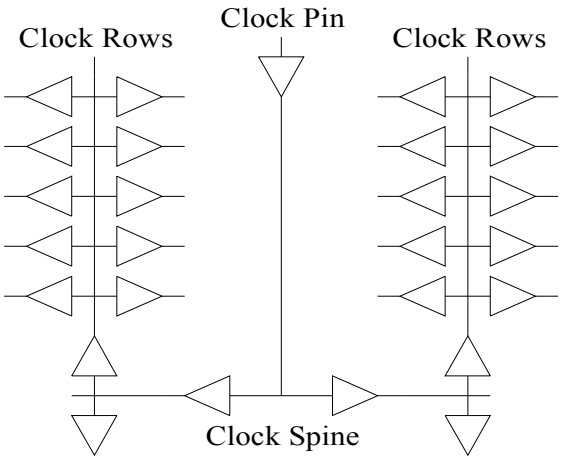

Figure with Example 3-3, page 122, Wayne Wolf, ***FPGA-Based System Design***, Pearson Education, Inc. (Prentice Hall), 2004

- Secondary Global Routing Network
  - 24 backbone lines
    - ✓ 1/2 along top
    - ✓ 1/2 along bottom
  - Delay-locked loop (DLL) used to regulate internal clock

- Configuration
  - Reconfigured by changing SRAM contents
  - Lines allocated for configuration
    - Permanent or released to general purpose I/O after configuration
    - Usually bit serial (since reconfiguration is not frequently done)
      - ✓ Could be parallel if configuration time is important
  - Programmed by
    - PC with download cable
    - PROM (programmable read only memory) on printed circuit board with FPGA
      - ✓ FPGA on power-up runs through protocol on configuration pins
  - SRAM in FPGA
    - More conservative than those used in bulk SRAM
      - ✓ Why?
        - Must be as immune as possible to power supply noise
      - ✓ Result
        - Slower to read/write than bulk SRAM cells, but memory is more stable
  - Scan Chains & JTAG
    - Chips & boards must be tested to ensure both were manufactured properly
    - JTAG (Joint Test Action Group)
      - ✓ JTAG standard created to allow chips on boards to be more easily tested
      - ✓ Boundary Scan
        - Design to scan pins at boundary between chip & board during testing
      - ✓ During testing pins are decoupled from their normal values & used as a shift register
      - ✓ Allows input values to be presented & outputs validated
        - Process controlled by *test access port* (TAP) controller
      - ✓ JTAG is built into most FPGAs
    - TAP Controller Pins
      - ✓ TDI
        - Shift Register Input
      - ✓ TDO
        - Shift Register Output
      - ✓ TCK
        - Test Clock
      - ✓ TMS
        - Test Mode Select
      - ✓ TRST
        - Test Reset Pin
        - Optional
      - ✓ Instruction Register (IR)
        - Determines actions taken by TAP
        - State transition graph of TAP defined by JTAG standard

- ✓ Bypass Register (BP)
  - Allows bits to be shifted into IR or IR to be left intact
- ✓ Each pin on chip is modified to include JTAG shift register logic
  - Outside unit controls & observes all pins on chip

JTAG Architecture

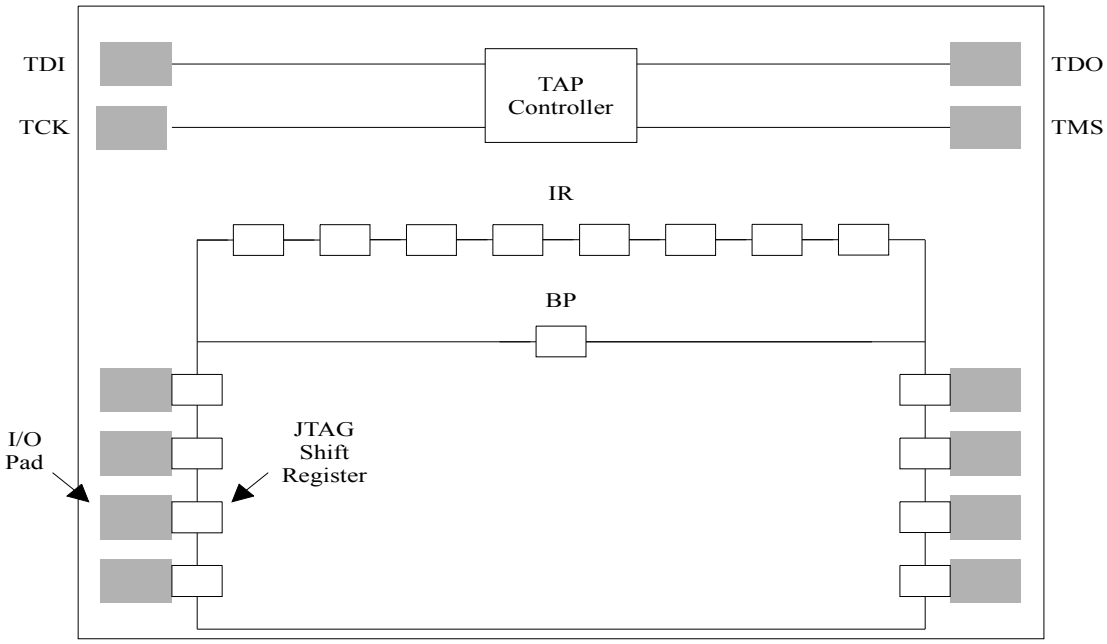

Figure 3-7, page 126, Wayne Wolf, ***FPGA-Based System Design***, Pearson Education, Inc. (Prentice Hall), 2004

- Spartan-II Configuration
  - Requires about 200,000 to 1.3 million bits
    - Depends on size of chip
  - Configured on one of several modes
    - Master Serial Mode
      - ✓ Chip is first in chain
      - ✓ Configuration loaded from EPROM or download cable
    - Slave Serial Mode
      - ✓ Configuration obtained from another slave serial mode chip or the master serial mode chip in chain
    - Slave Parallel Mode
      - ✓ Provides fast 8-bit wide configuration
    - Boundary Scan Mode
      - ✓ Utilizes standard JTAG pins

- Pins Dedicated to Configuration
  - PROGRAM'
    - ✓ Used to initiate configuration
  - M0, M1, M2
    - ✓ Control configuration mode
  - DONE
    - ✓ Signals configuration has finished
  - TDI, TDO, TMS, TCK
    - ✓ Boundary scan pins
    - ✓ Used to configure without using dedicated configuration pins
- Spartan 3
  - Optimized for High Density & High Pin Count
  - Ideal for highly integrated data processing applications
- Spartan 3E
  - Ideal for logic integration, DSP co-processing, & embedded control which require significant logic and processing resources
- Spartan 3A
  - Ideal for bridging, differential signaling & memory interfacing functions, which require wide or multiple interfaces

- Spartan-3E
  - XC3S100E
  - CP132
    - 8x8 mm
  - System Gates
    - 100K
  - Logic Cells
    - 2,160
  - Dedicated Multipliers
    - 4
  - Block RAM Bits
    - 72K
  - Distributed RAM Bits
    - 15K
  - Digital Clock Managers (DCMs)
    - 2
  - Max User I/O
    - 108
  - Max Diff. I/O Pairs
    - 40

The Permanently Programmed FPGA

- Do NOT have to be configured at power-up
  - Only need to be configured once
- Implementations
  - Antifuses
  - Flash

- Antifuse Configuration
  - Fabricated as normally open
  - Programming accomplished by applying a voltage across the antifuse to connect *Metal 1* to *Metal 2*
  - Resistance = Order of 100 Ω
    - More resistance than standard via
  - Advantages Over a Fuse
    - Most connections in FPGA should be open
      - ✓ Most programming points left in initial state

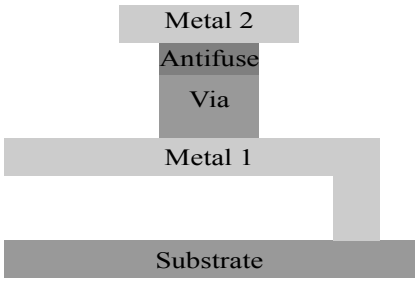

Figure 3-8, page 128, Wayne Wolf, ***FPGA-Based System Design***, Pearson Education, Inc. (Prentice Hall), 2004

- Flash Configuration
  - Flash
    - High-quality programmable read-only memory
    - Low-leakage capacitor holds voltage to control transistor gate
    - Programmed Flash Switch

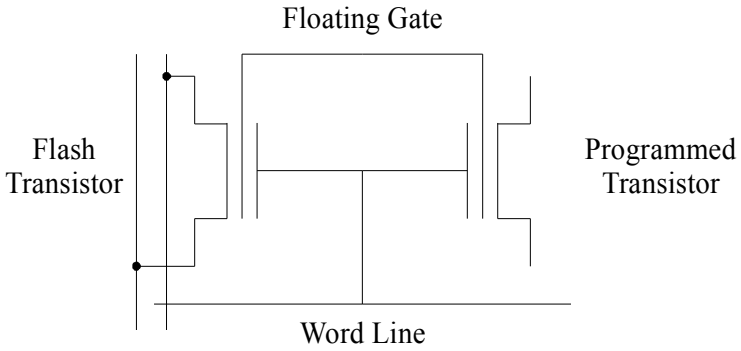

Figure 3-9, page 129, Wayne Wolf, ***FPGA-Based System Design***, Pearson Education, Inc. (Prentice Hall), 2004

References

- Wayne Wolf, ***FPGA-Based System Design***, Pearson Education, Inc. (Prentice Hall), 2004
- ***Spartan-3 Generation FPGAs - The Ultimate Low-Cost Applications Platform***, Xilinx Inc., PN 0010855-1, 2006
  - http://www.xilinx.com/publications/prod_mktg/pn0010983.pdf
- ***Xilinx Spartan-3E FPGA Family: Data Sheet***, Xilinx Inc., August 26, 2009
  - http://www.xilinx.com/support/documentation/data_sheets/ds312.pdf