Lecture 19 (chap. 3)

Introduction to Reconfigurable Hardware (FPGA).

Khaza Anuarul Hoque
ECE 4250/7250

Field-Programmable Gate Arrays (FPGAs): Introduction

- ICs that contain an array of identical logic blocks with programmable interconnections.
- Three major programmable elements in FPGAs: the **logic block**, **the interconnect** (routing), and the **input/output block**.
- Programmable logic blocks are created by using multiplexers, look-up tables, and AND-OR or NAND-NAND arrays.
- Have revolutionized the way prototyping and designing is done in the world due to the flexibility offered as it is reprogrammable.
- Vendors: Xilinx, Altera, Lattice Semiconductor, and Microsemi.

FPGAs: Introduction (continued)

- Advantages:
  - Reduction in manufacturing time as one adopts FPGAs instead of MPGAs.
  - Easier design iterations.
  - Less costly to correct design mistakes or specification changes.
  - Reduced prototyping costs.
  - At low volumes, FPGAs are cheaper than MPGAs.

FPGAs vs MPGAs

A **mask-programmable gate array** uses one (or more) metal mask in the IC manufacturing process, to achieve the intended functionality.

In an **FPGA**, the user programs a standard off-the shelf chip, either by loading a bitstream into latches, or into Flash cells, or by programming antifuses.

FPGAs: Introduction (continued)

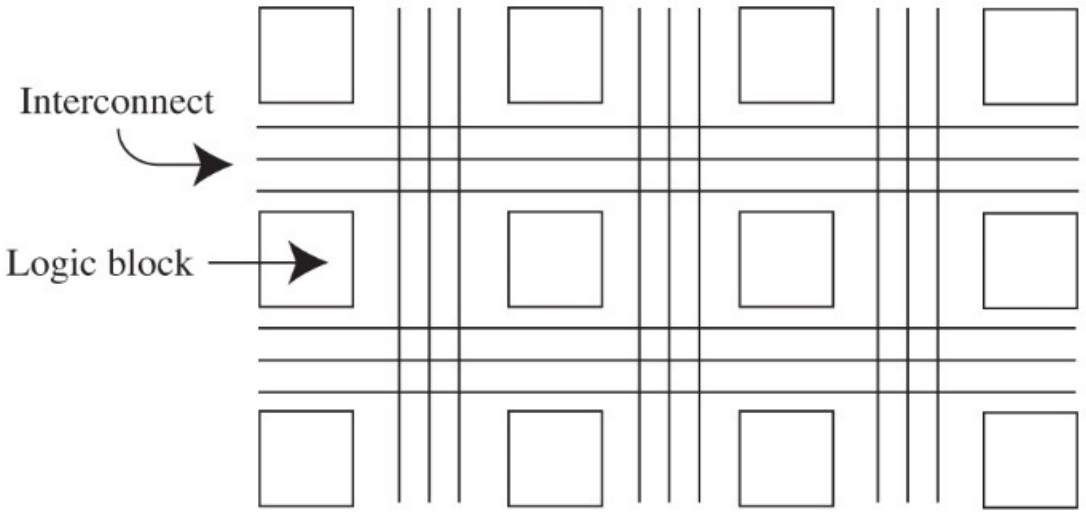
- Disadvantages:
  - Are less dense than traditional gate arrays (MPGAs).
  - A lot of resources are needed to achieve programmability.
  - MPGAs have better performance than FPGAs.
  - Interconnection delays are unpredictable in FPGAs.
  - PLDs such as PALs and GALs are simple and inexpensive.
  - CPLDs are faster, cheaper, and more predictable in timing than FPGAs.

FPGAs: Organization

- Matrix-based (symmetrical array):
  - Manufacturer: Most Xilinx FPGAs.
  - Large granularity: capable of implementing 4-variable functions or more.
  - Typically contain 8 x 8 arrays in the smaller chips and 100 x 100 or larger arrays in the bigger chips.
  - Routing: two-dimensional channeled (horizontal and vertical).

FPGAs: Organization (continued)

- Matrix-based (symmetrical array):



FPGAs: Organization (continued)

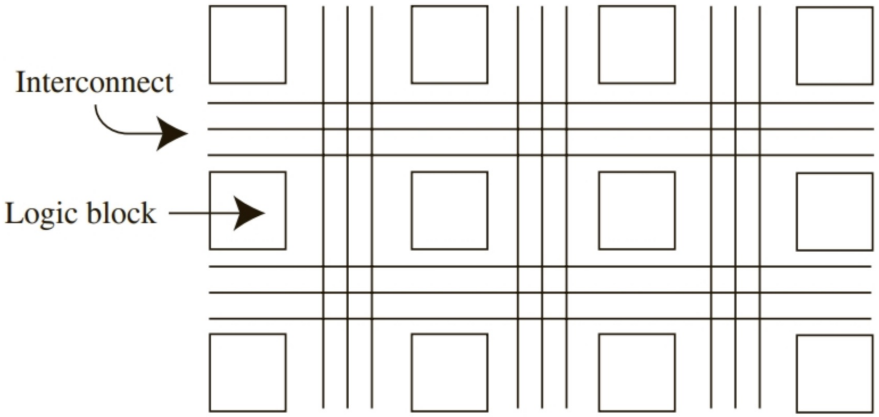
- Row-based:
  - Inspired by traditional gate arrays.
  - Traditional mask-programmable gate arrays use very similar architectures.
  - Routing: one-dimensional channeled routing, as the routing resources are located as a channel in between rows of logic resources.
  - Manufacturer: some Microsemi FPGAs employ this architecture.

FPGAs: Organization (continued)

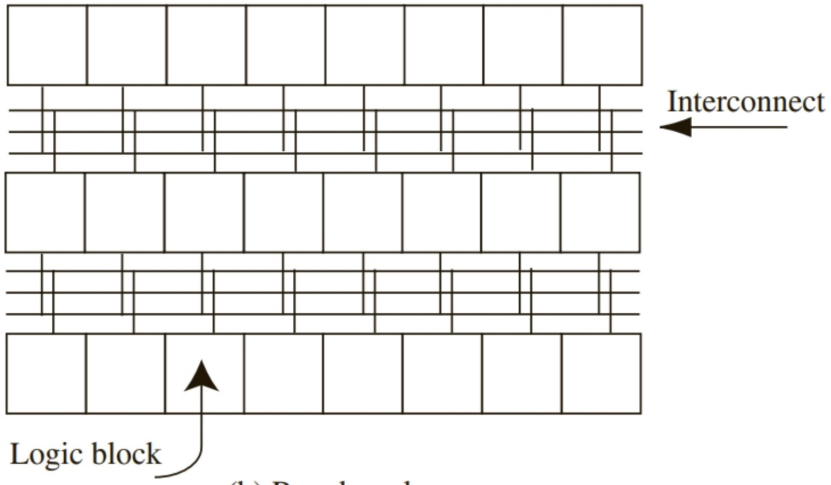
- Hierarchical:
  - Blocks of logic cells are grouped together by a local interconnect, and several such groups are interconnected by another level of interconnect.
  - Manufacturer: Altera APEX20 and APEX II.

FPGAs: Organization (continued)

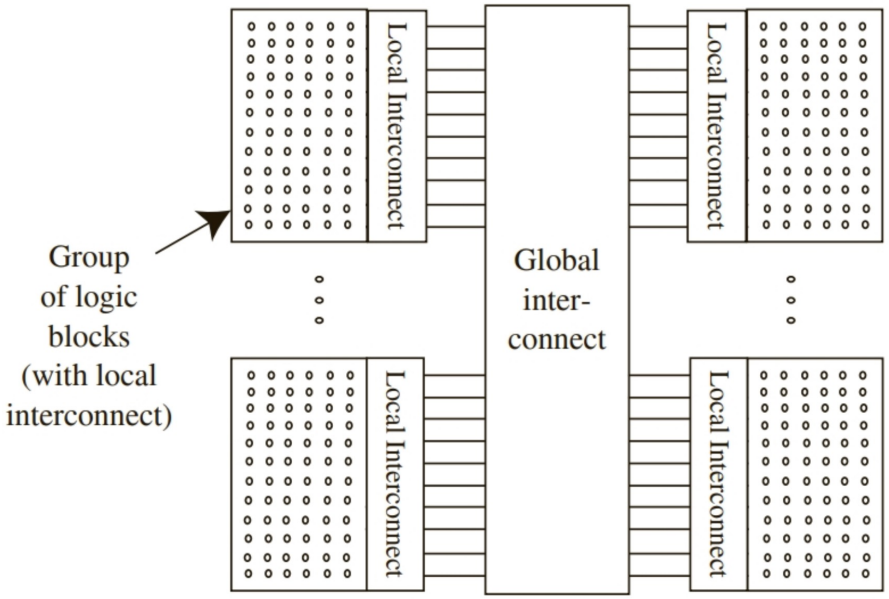
- Sea-of-gates:
  - Consists of a large number of gates with an interconnect superimposed on the sea of gates.
  - Manufacturers: Plessey – mid-1990's (sea-of-gates), Microsemi Fusion (sea of tiles).



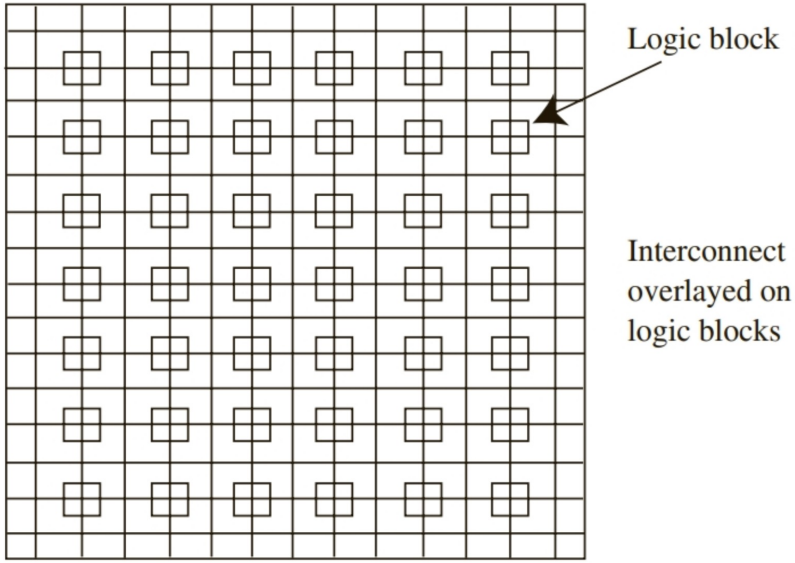
(a) Matrix based (symmetrical array)



(b) Row based



(c) Hierarchical



(d) Sea of gates

FPGAs: Programming Technologies

- FPGAs consist of a large number of logic blocks interspersed with a programmable interconnect.
- Programmable:
  - The logic block: the same building block can be "programmed" or "configured" to create any desired circuitry.
  - Interconnections between the logic blocks.

FPGAs: Programming Technologies (continued)

- StaticRAM (SRAM):
  - Involves creating reconfigurability by bits stored in SRAM cells.
  - Logic blocks, I/O blocks, and interconnects can be made programmable by using configuration bits stored in SRAM.
  - Reconfigurable logic blocks can easily be implemented as look-up tables (LUTs).
  - Bits that are stored in the SRAM for deciding the LUT functionality or interconnection are called **configuration bits**.

FPGAs: Programming Technologies (continued)

- SRAM:
  - A cell usually takes six transistors.
  - Write Operation: when the *Word Line* is set to high, the values on the *Bit Line* will be latched into the cell.
  - Read operation: done by precharging the *Bit Line* and *Bit Line* to a logic 1 and then setting *Word Line* to high. The contents stored in the cell will then appear on the *Bit Line*.

FPGAs: Programming Technologies (continued)

- SRAM Advantages:
  - Volatile: new contents can be written again and again.
  - Fabrication steps are the same for making SRAM cells and for making logic.
- SRAM Disadvantages:
  - Cost: five or six transistors are used for every SRAM cell.
  - Volatile: another device (such as an EPROM) is needed to store the configuration bits.

FPGAs: Programming Technologies

- EPROM/EEPROM:
  - Used to control programmable connections.
  - Compared with SRAMs:
    - Are slower than SRAMs; SRAMS can be programmed faster.
    - EPROMs also require more processing steps than SRAM.
  - EEPROM is similar to EPROM, but removal of the gate charge can be done electrically.

FPGAs: Programming Technologies (continued)

- Flash memory :
  - A form of EEPROM.
  - Allows multiple locations/segments to be erased in one operation by pulling electrons off.
  - Stores information in floating gate transistors as in EPROM.
  - Cell is read by placing a specific voltage on the control gate.

FPGAs: Programming Technologies (continued)

- Antifuse:
  - Programming: changes from high resistance (open) to low resistance (closed) when a high voltage is applied to it.
  - Normally OFF.
  - One-time programmable.
- Antifuse Advantages:
  - Area consumed is small.
  - Connections are faster than SRAM and EEPROM.
- Antifuse Disadvantages:
  - Not reprogrammable.

FPGAs: Programming Technologies (continued)

- Comparison of technologies:

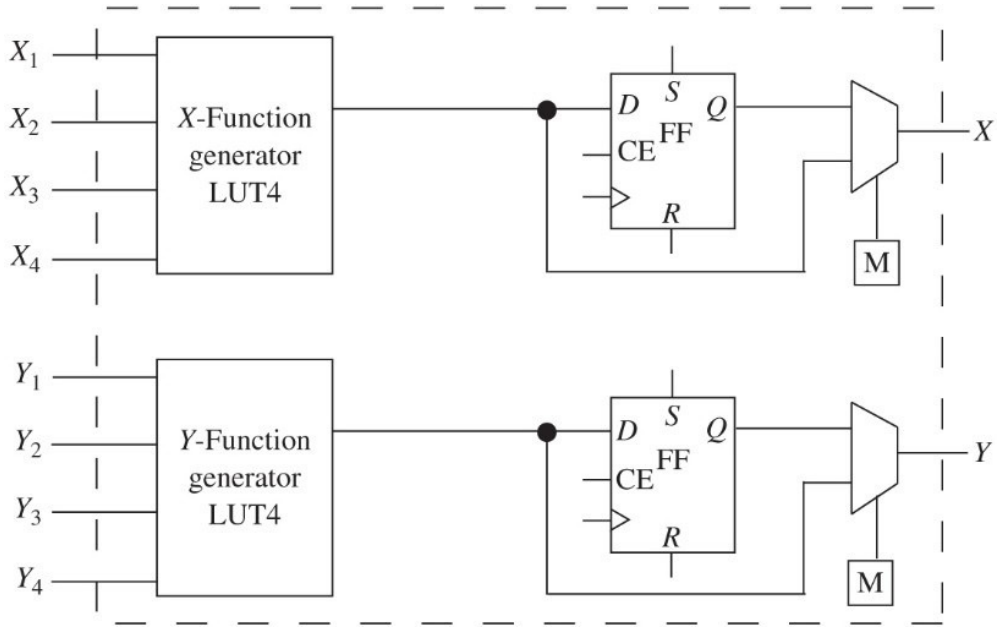
Programming Area Technology	Volatility	Programmability	Overhead	Resistance	Capacitance
SRAM	Volatile	In-circuit reprogrammable	Large	Medium to high	High
EPROM	Nonvolatile	Out-of-circuit reprogrammable	Small	High	High
EEPROM/Flash	Nonvolatile	In-circuit reprogrammable	Medium to high	High	High
Antifuse	Nonvolatile	Not reprogrammable	Small	Small	Small

FPGAs: Programmable Logic Block Architectures

- Look-Up-Table–Based Programmable Logic Blocks:
  - Many LUT–based FPGAs use a 4-variable look-up table (LUT4) plus a flip-flop as the basic element and then combine them in different ways.
  - Very common for Xilinx and Altera.

FPGAs: Programmable Logic Block Architectures (continued)

- Look-Up-Table–Based Programmable Logic Blocks:

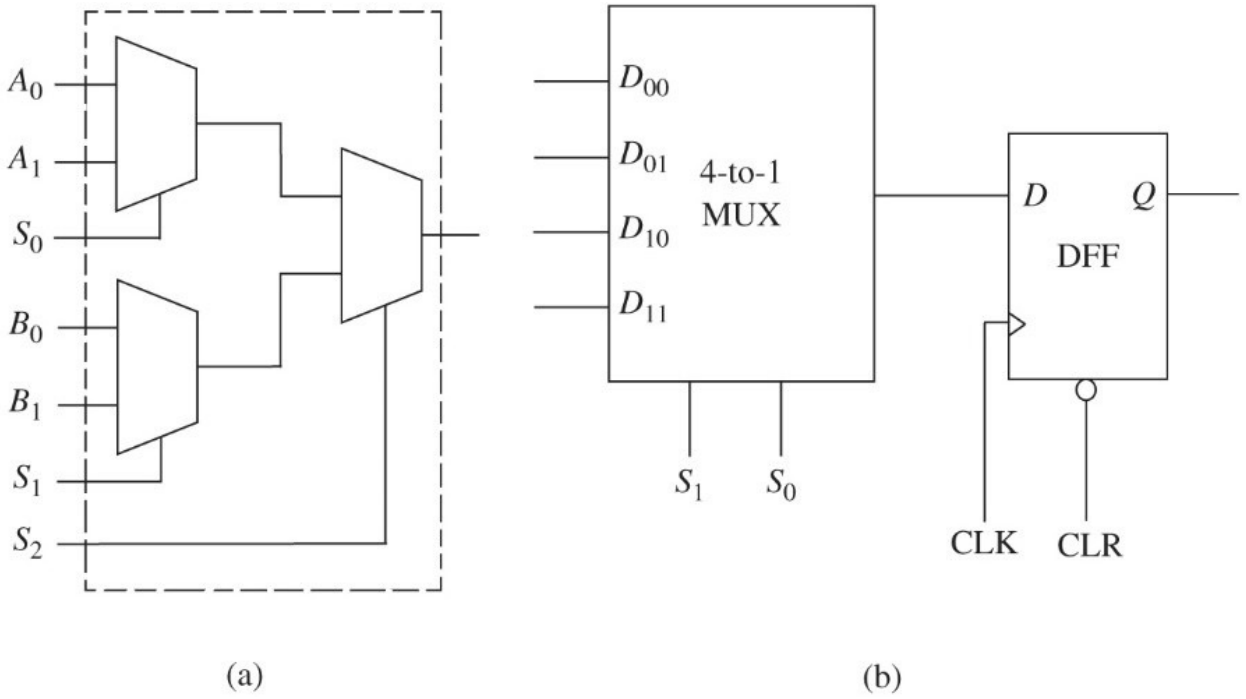


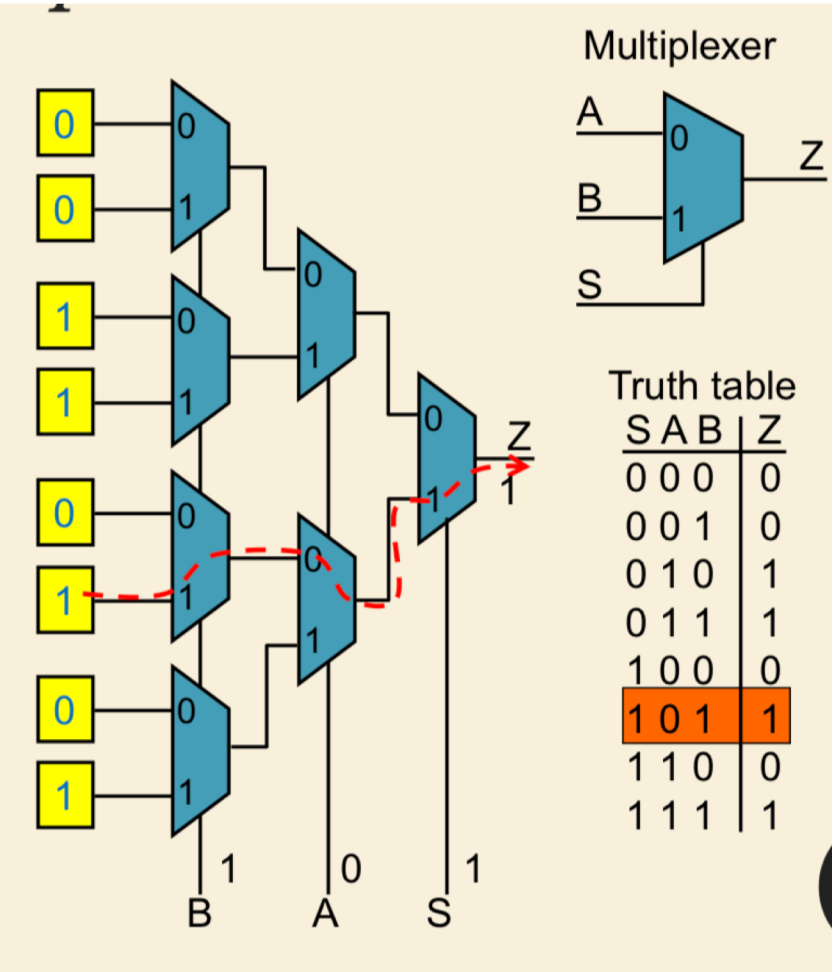
FPGAs: Programmable Logic Block Architectures (continued)

- Logic Blocks Based on Multiplexers and Gates:
  - Any combinational function can be implemented using multiplexers alone.
  - Microsemi is a manufacturer of multiplexer-based FPGAs.

FPGAs: Programmable Logic Block Architectures (continued)

- Logic Blocks Based on Multiplexers and Gates:




Multiplexer
A
B
S
0
1
Z
Truth table
S A B | Z
0 0 0 | 0
0 0 1 | 0
0 1 0 | 1
0 1 1 | 1
1 0 0 | 0
1 0 1 | 1
1 1 0 | 0
1 1 1 | 1
0
0
1
1
0
1
0
1
0
1
Z
1
1
0
1
B
A
S

FPGAs: Programmable Interconnects

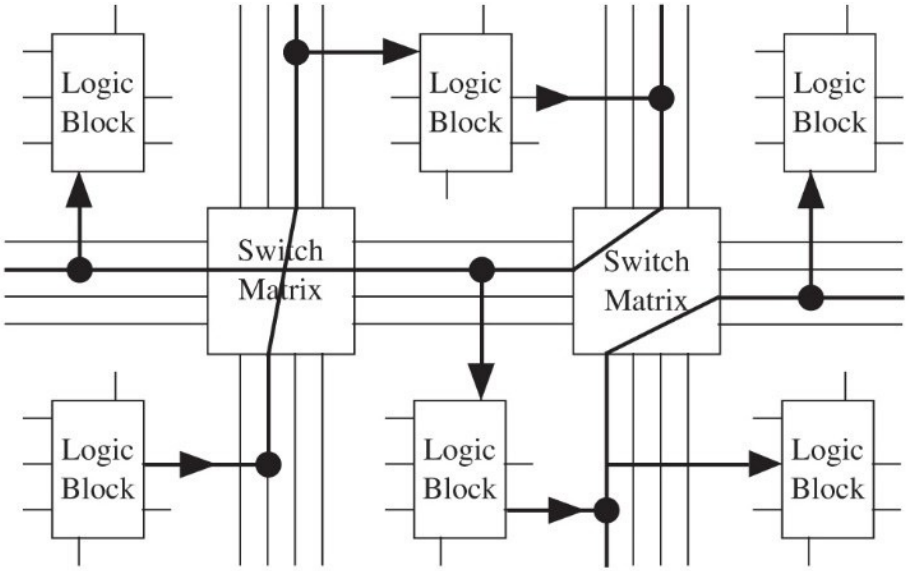
- There are different types of interconnection resources in all commercial FPGAs. Every vendor has its own specific names for the different types of interconnects in their FPGA.
- Main types:
  - Interconnects in Symmetric Array
  - Row-based

FPGAs: Programmable Interconnects (continued)

- Interconnects in Symmetric Array FPGAs:
  - General-purpose
  - Direct interconnects
  - Global lines

FPGAs: Programmable Interconnects (continued)

- General-purpose:
  - Use switch matrices that provide interconnections between routing wires connected to the switch matrix.

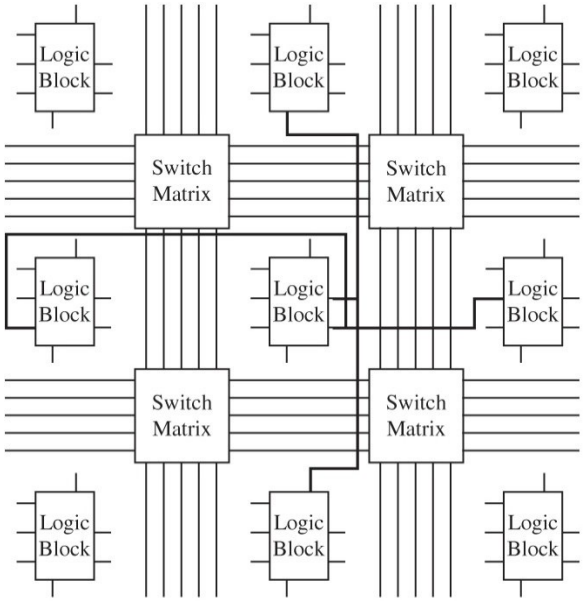


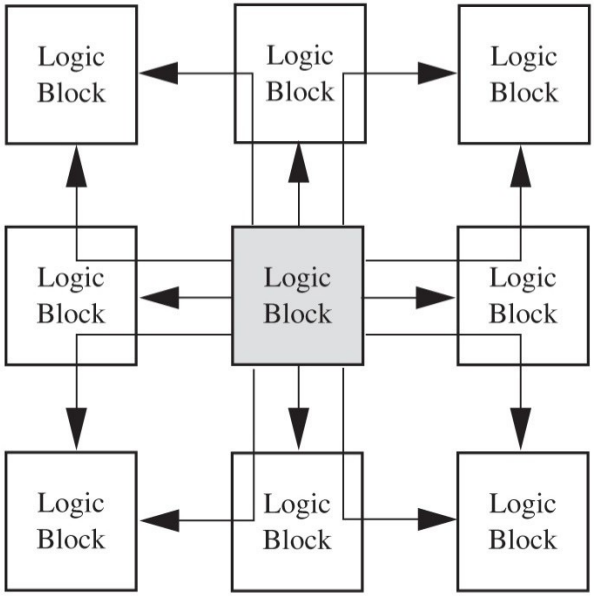
FPGAs: Programmable Interconnects (continued)

- Direct interconnects:
  - Special connections between adjacent logic blocks. Interconnects are fast because they do not go through the routing matrix.
  - Direct interconnections are to the four nearest neighbors or, in some cases, to eight neighboring blocks.

FPGAs: Programmable Interconnects (continued)

- Direct interconnects:
  - Nearest 4 and 8 neighboring blocks



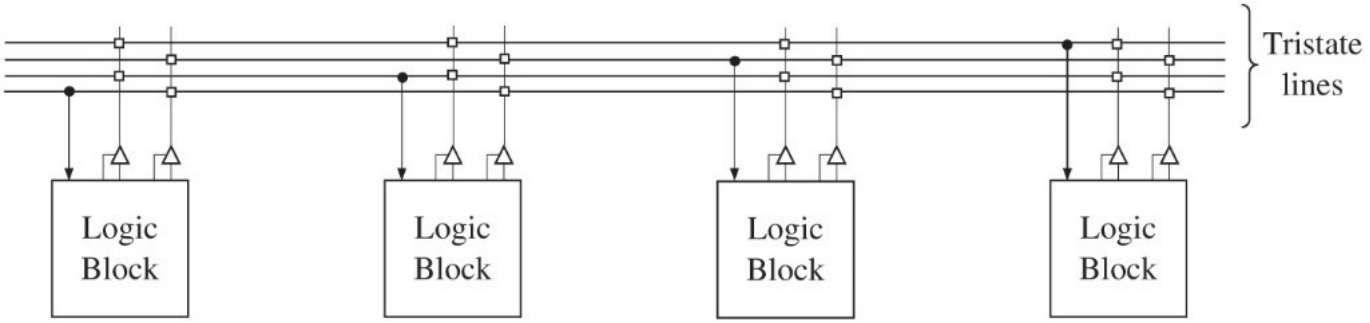


FPGAs: Programmable Interconnects (continued)

- Global lines:
  - Routing lines span the entire width/height of device.
  - A limited number (two or four) of such global lines are provided by many FPGAs in horizontal and vertical directions.

FPGAs: Programmable Interconnects (continued)

- Global lines:



FPGAs: Programmable Interconnects (continued)

- Row-Based:
  - Has rows of logic blocks and channels of switches to enable connections between the logic blocks.
  - Routing resources in these are similar to routing in traditional gate arrays.
  - Interconnects are either nonsegmented or segmented.
    - Nonsegmented has a high area overhead compared to segmented.

FPGAs: Programmable I/O Blocks

- I/O blocks on modern FPGAs allow use of the pin as input and/or output, in direct (combinational) or latched forms, in tristate true or inverted forms, and with a variety of I/O standards.

FPGAs: Programmable I/O Blocks (continued)

- I/O Standards:
  - LVTTL: low-voltage transistor-transistor logic; 3.3-V standard that can tolerate 5-V signals.
  - PCI: peripheral component interconnect; has 5-V and 3.3-V versions.
  - LVCMOS: low-voltage complementary metal-oxide semiconductor; LVCMOS2, a 2.5-V standard that can tolerate 5-V signals.
  - LVPECL: low-voltage positive emitter-coupled logic
  - SSTL: stub-series terminated logic
  - AGP: advanced graphics port
  - CTT: center tap terminated
  - GTL: gunning transceiver logic
  - HSTL: high-speed transceiver logic

FPGAs: Dedicated Specialized Components

- Dedicated Memory:
  - A key feature of modern FPGAs is the embedding of dedicated memory blocks (RAM) onto the chip.
  - Modern FPGAs include 16K to 10M bits of memory. The width of the embedded RAM often can be adjusted.

FPGAs: Dedicated Specialized Components (continued)

- Dedicated Arithmetic Units:
  - Used to implement arithmetic logic.
  - Implementation generally takes more area and power and is slower than custom implementations. It is therefore of benefit to provide dedicated fast-carry logic to create fast adders.
  - Many FPGAs also contain dedicated multipliers that are more efficient than those implemented using the programmable logic in the FPGA.

FPGAs: Dedicated Specialized Components (continued)

- Digital Signal Processing (DSP) Blocks:
  - Dedicated multipliers help DSP applications.
    - Example: Xilinx Virtex 5
  - Carry chains to facilitate addition.
    - Example: Altera Stratix IV
  - Also provided: DSP building blocks such as hardware for fast Fourier transforms (FFTs), finite impulse response (FIR) filters, infinite impulse response (IIR) filters, encryption/ decryption, compression/decompression, and security functions.

FPGAs: Dedicated Specialized Components (continued)

- Embedded Processors:
  - Many modern FPGAs contain an entire processor core which is useful when designers use hybrid solutions.
  - Some also include: the core of a small MIPS processor (i.e., MIPS R 4000), an embedded version of the IBM PowerPC processor, or custom processors (i.e., MicroBlaze from Xilinx).

FPGAs: Applications

- Rapid prototyping:
  - FPGAs can contain 5 million or more gates, so many large real-world systems can be prototyped using a single FPGA.
  - Multiple FPGAs can be interconnected to realize large systems.
  - Accomplished by using boards with multiple FPGAs and plugging multiple boards into a backplane (motherboard).

FPGAs: Applications (continued)

- As Final Product in Medium-Speed Systems:
  - When 150-200 MHz is sufficient, FPGAs can be used as the final product.
  - Enhancements to the system can be done as software updates rather than as hardware changes.

FPGAs: Applications (continued)

- Reconfigurable Circuits and Systems:
  - Can build dynamically reconfigurable circuits and systems.
  - SRAM-based FPGAs make it possible to implement “soft” hardware.
  - Used to design circuits and systems that need multiple functionalities at various times.
  - Example: reprogrammable Tomahawk missile.

Tomahawk !

FPGAs: Applications (continued)

- Glue Logic:
  - New interface logic can be implemented on the same FPGA as in a software update.
- Hardware Accelerators/Coprocessors:
  - An FPGA can be used to implement the key kernel, as different kernels can dynamically be programmed into the FPGA.
  - Examples: computer architecture simulator acceleration, emulation boards, and hardware test/verification.

FPGAs: Design Flow

- Automatic synthesis tools are available that will take a VHDL description of the system as an input and generate an interconnection of gates and flip-flops to realize the system.
- Behavioral models can be translated into design implementations reasonably efficiently.

FPGAs: Design Flow (continued)

- Design steps for a digital system:
  - 1. Create a behavioral, RTL, or structural model of the design in a hardware description language such as Verilog or VHDL.
  - 2. Simulate and debug the design.
  - 3. Synthesize the design targeting the desired device.
  - 4. Run a mapping/partitioning program.
  - 5. Run an automatic place and route program.
  - 6. Run a program that will generate the bit pattern necessary to program the FPGA.
  - 7. Download the bit pattern into the internal configuration cells in the FPGA, and test the operation of the FPGA.