MODULE 3: Boolean Algebra and Digital Logic

Lecture 3.5
Programmable Logic

Prepared By:
- Scott F. Midkiff, PhD
- Luiz A. DaSilva, PhD
- Kendall E. Giles, PhD

Electrical and Computer Engineering
Virginia Tech

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech
Invent the Future®

Lecture 3.5 Objectives

- Describe the motivation for using programmable logic
- Define the major types of programmable logic and identify key features of each
- Describe the implementation of two-level AND/OR (sum-of-products) logic expressions using an AND/OR programmable logic array (PLA)

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech
Invent the Future®

Programmable Logic (1)

- Programmable logic is widely used to implement higher level logic functions
  - Combinational only
  - Mixed combinational and sequential

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech
Invent the Future®

Programmable Logic (2)

- Common hardware (the programmable logic) can be customized to implement many different logic functions
  - Reduces hardware development cost, usually at the expense of per unit cost
  - Attractive where a small to medium number of parts are needed for a particular design
  - Custom logic or application specific integrated circuits (ASICs) may be more attractive for high volume parts or where maximum performance and/or density is required

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech
Invent the Future®

Programmable Logic (3)

- Designs can be implemented and modified quickly
  - Reduces design time and, thus, time to market for a new part
  - Some programmable logic allows field modifications and upgrades
- Detailed design process is usually implemented using computer-aided design (CAD) tools
  - Programmable logic customization file created based on high level input such as truth table, logic expressions, or gate diagram

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech
Invent the Future®

Programmable Logic (4)

- Potential disadvantages
  - Higher per unit cost
  - Lower performance (operating frequency, power consumption)
  - Lower density (fewer devices per area)
  - Structure constrains design optimization

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech
Invent the Future®

CHECK POINT

As a checkpoint of your understanding, please pause the video and make sure you can do the following:

- Describe the motivations for using programmable logic

If you have any difficulties, please review the lecture video before continuing.

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech Invent the Future®

Types of Programmable Logic (1)

- The programmable logic market is very active — new devices are introduced frequently
- Programmable logic arrays (PLAs)
  - Implement two-level combinational logic functions
  - Usually part of a more complex programmable logic device or used as part of an application specific integrated circuit
  - Often used to implement general control logic on microprocessors

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech
Invent the Future®

Types of Programmable Logic (2)

- Programmable logic devices (PLDs)
  - Implement sequential and/or combinational logic—combine programmable and combinational logic devices with flip-flops to hold state variables
  - Often include programmable input/output pins
  - Highly constrained circuit topology
- Erasable programmable logic devices (EPLDs)
  - Like PLDs, except allow the user to erase and “re-program” the device to change the logic function

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech
Invent the Future®

Types of Programmable Logic (3)

- Programmable gate arrays (PGAs)
  - Include combinational logic blocks that can be interconnected to create reasonably complex circuits
  - Topology is relatively unconstrained, at least compared to PLDs
  - Often include flip-flops or other memory elements to enable implementation of sequential circuits
  - Customized as a final step in manufacturing
- Field Programmable Gate Arrays (FPGAs)
  - Like PGAs, except programmed by the end-user (in the field)

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech
Invent the Future®

Types of Programmable Logic (4)

- Run-time reconfigurable (RTR) field programmable gate arrays
  - Like regular FPGAs, except they can be quickly programmed in place (while still in the circuit)
  - Allows a computational device to use the same logic for different functions
  - Research area—goal is to achieve hardware performance with software flexibility

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech
Invent the Future®

CHECK POINT

As a checkpoint of your understanding, please pause the video and make sure you can do the following:

- Define the major types of programmable logic and identify key features of each

If you have any difficulties, please review the lecture video before continuing.

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech Invent the Future®

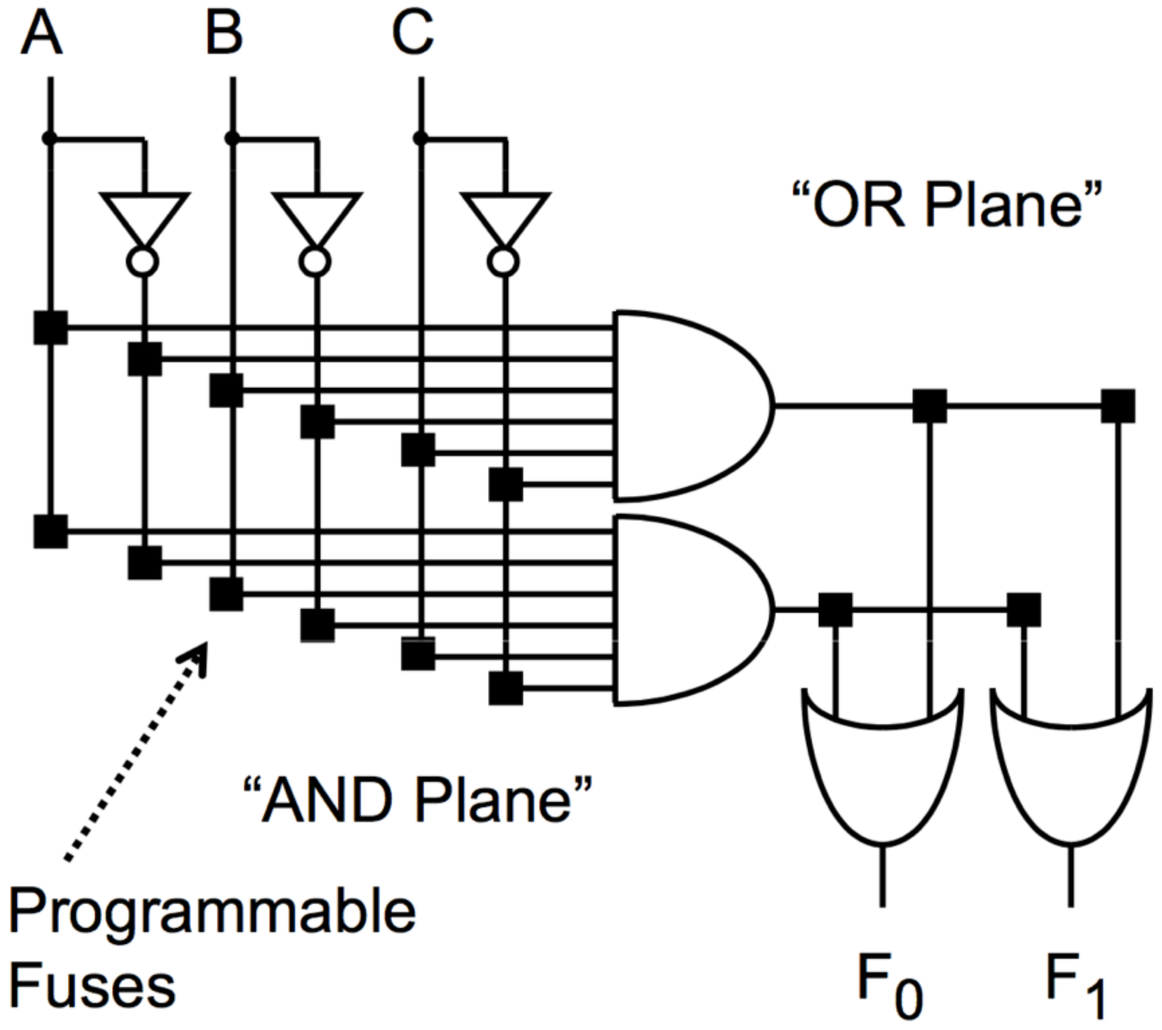
Programmable Logic Arrays (PLAs)

- PLAs are widely used in processor and other hardware and are often a core element in other types programmable logic
- Provide a regular structure for implementing two-level combinational logic expressions, such as AND/OR or NAND/NAND expressions
- Realizing a customized PLA:
  - PLA may be programmed by making or breaking connections as a final step in manufacturing or as an end-user operation
  - PLA may be custom designed, for example by a CAD tool, for use as part of a VLSI integrated circuit

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech Invent the Future®

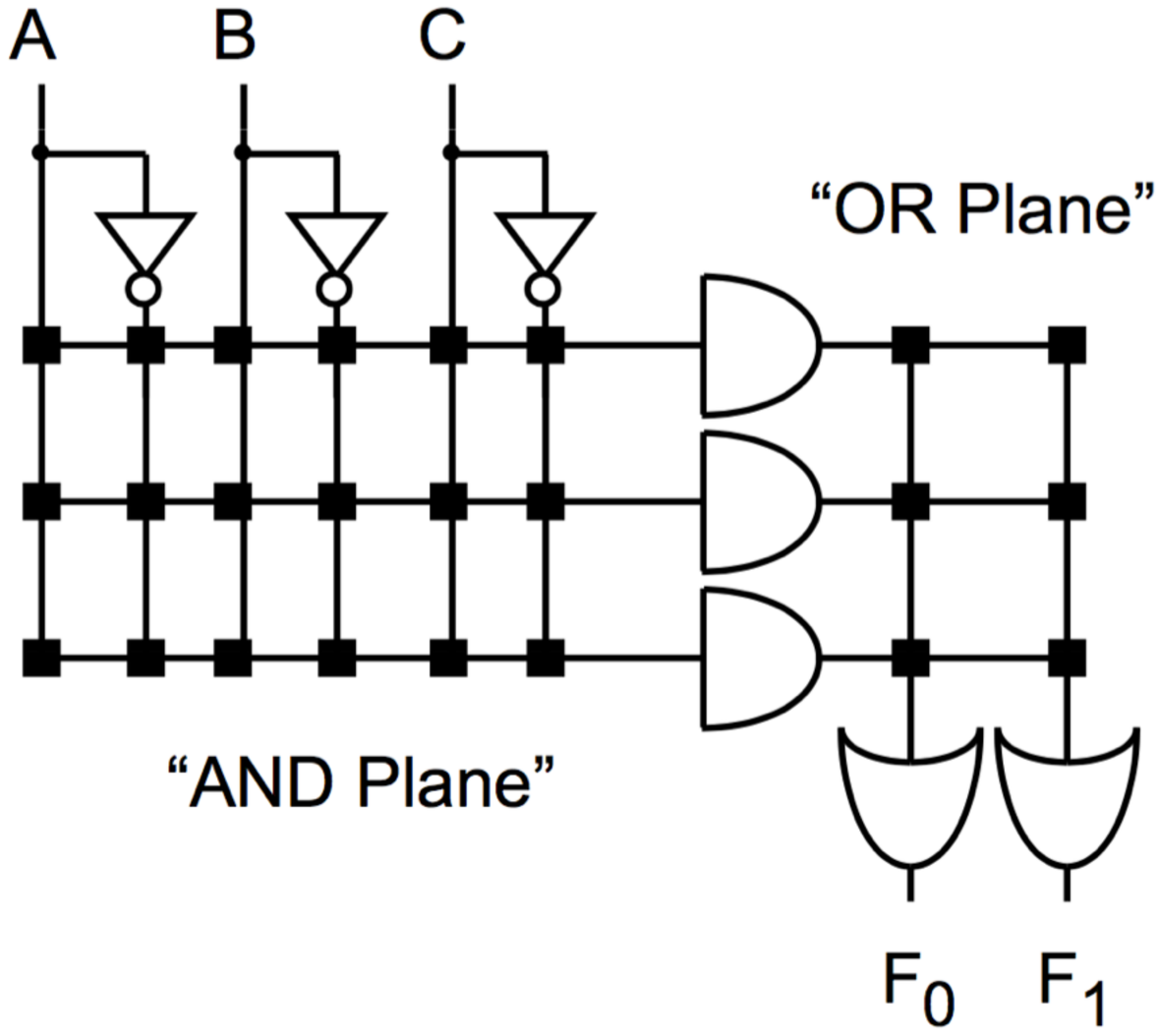
AND/OR Programmable Logic Array



© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech
Invent the Future®

Simplified Diagram of a PLA



Programmed by making or breaking fused connections

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech
Invent the Future®

PLA Example: Truth Table

- Design a PLA to implement functions F_0 and F_1 specified by this truth table
- Assume an AND/OR PLA, as shown previously
- Expressions
- $F_0 = A'B' + AB'C'$
- $F_1 = A'B' + AB'C$
  - Note that the term A'B' is common to both functions

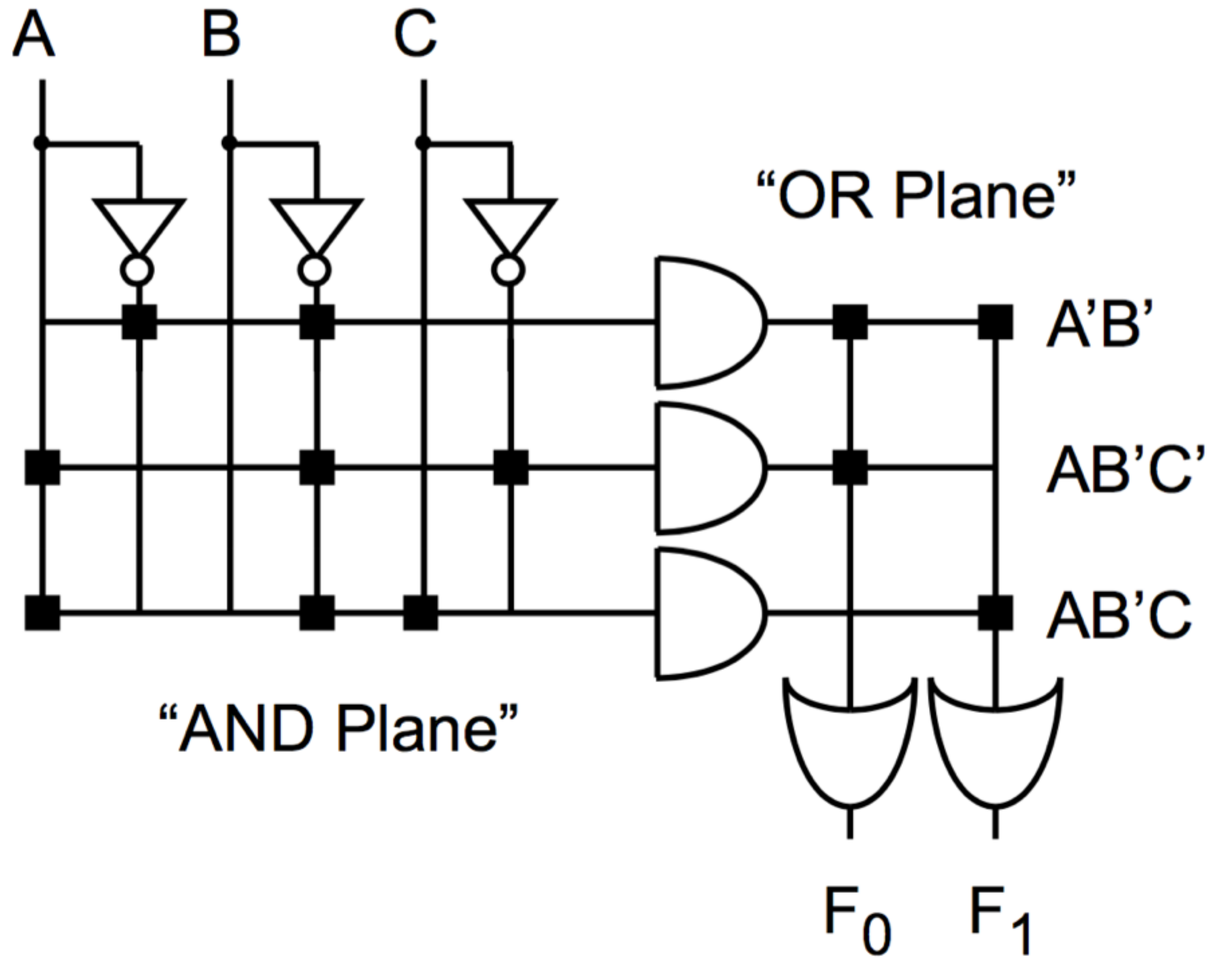
A	B	C	F_0	F_1
0	0	0	1	1
0	0	1	1	1
0	1	0	0	0
0	1	1	0	0
1	0	0	1	0
1	0	1	0	1
1	1	0	0	0
1	1	1	0	0

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech Invent the Future®

PLA Example: PLA "Programming"

$F_0 = A'B' + AB'C'$
$F_1 = A'B' + AB'C$



© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech
Invent the Future®

CHECK POINT

As a checkpoint of your understanding, please pause the video and make sure you can do the following:

- Validate that the functions for F_0 and F_1 on slide 16 conform to the given truth table.

If you have any difficulties, please review the lecture video before continuing.

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech Invent the Future®

Summary

- For many types of designs, programmable logic offers significant advantages in cost, flexibility, and simplicity
- There are many types of programmable logic, such as PLAs, PLDs, and FPGAs
- PLAs provide a generic two-level logic structure that can be customized to implement a set of functions

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech
Invent the Future®

MODULE 3: Boolean Algebra and Digital Logic

Lecture 3.5
Programmable Logic

Prepared By:
- Scott F. Midkiff, PhD
- Luiz A. DaSilva, PhD
- Kendall E. Giles, PhD

Electrical and Computer Engineering
Virginia Tech

© S. F. Midkiff, L. A. DaSilva, and K. E. Giles

VirginiaTech
Invent the Future®