



# Computer Architecture

With Simulation Exercises

# Computer Arithmetic



- Carry Lookahead Addition
- Adder-Subtractor Design
- Shift-Add Multiplication
- Booth Multiplication
- Restoring Division
- Non-Restoring Division

# Exercises: Adder I



## ■ Create 4-Bit Propagate/Generate Units

- $P1, P2, P3, P4$
- $G1, G2, G3, G4$

## ■ Create a 4-Bit Carry Lookahead Unit

- $C1 = C0 \cdot P1 + G1$
- $C2 = C0 \cdot P1 \cdot P2 + G1 \cdot P2 + G2$
- $C3 = C0 \cdot P1 \cdot P2 \cdot P3 + G1 \cdot P2 \cdot P3 + G2 \cdot P3 + G3$
- $GP = P1 \cdot P2 \cdot P3 \cdot P4$
- $GG = G1 \cdot P2 \cdot P3 \cdot P4 + G2 \cdot P3 \cdot P4 + G3 \cdot P4 + G4$

# Exercises: Adder II



- Create 4-Bit Adder Units
  - $S = A \oplus B \oplus C$
- Test All Four-Bit Units Independently
- Combine All Four-Bit Units into One File
- Use Hierarchical Design to Create a 16-Bit Carry-Lookahead Adder
- Test Adder Using a Robust Set of Tests

# Exercises: Adder Subtractor



- Create a 4-Bit Add/Subtract Select Unit
  - $B' = B \oplus C0$
- Add This Unit to Adder
- Replace All references to B With B'
- C0 Becomes the Add/Subtract Selector
- Test The Adder/Subtractor with a Robust Set of Test Vectors

# Exercises: Shift-Add Multiplier




- Add Registers
- Solidify All Control Details
- Special Care is Required for Overflow Control
- Design ROM Format
- Design ROM Sequencer
- Create ROM Code
- Test with Robust Set of Inputs

# Difficulties



- Microprogrammed Algorithms Are Extremely Time Consuming
- The Details of the Algorithm Get Lost in the Details of Constructing the Control
- The Point of Shift-Add Multiplication is to Teach Microprogrammed Control
- Completing a Partial Design May be More Desirable

# Other Algorithms

- 
- Supply the Student With A Pre-Constructed Data Path
  - Include Definitions for ROM Sequencer and ROM Format
  - Allow Students to Implement Other Algorithms by Microprogramming Instructor-Provided Hardware



# Other Exercises



## ■ Hardwired Control

- Direct State-Machine Implementation
- PLA Control

## ■ CPU Implementation

- PDP-8 Scale
- Some Instructor-Supplied Design Elements
- Modification of Existing Designs

# The Second Semester



- Pipelined Data Paths
- Parallel Processors
- Super-Scalar Machines
- Cache Implementation