

Lecture 9: ROMs, PLAs and PALs, Oh My

CSE 370, Autumn 2007
Benjamin Ylvisaker

Where We Are

- Last lecture: Combinational Verilog
- This lecture: ROMs, PLAs and PALs, oh my
- Next lecture: Quine-McCluskey Minimization
- Homework 3 due Wednesday
- Read lab 3

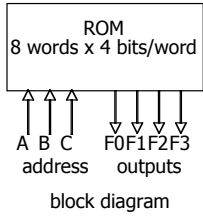
Discrete Gates are a Pain

- Building circuits with discrete chips, like what you've done in lab, is only tractable up to modestly sized circuits
- Alternatives:
 - ROMs
 - PLAs
 - PALs
 - FPGAs (later)

Read-Only Memories

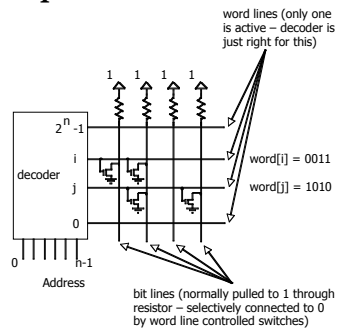
- Direct implementation of a truth table

| A | B | C | F0 | F1 | F2 | F3 |
|---|---|---|----|----|----|----|
| 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 1 | 0 | 1 |



ROM Implementation

- “words”
- “word-size”
- “address”

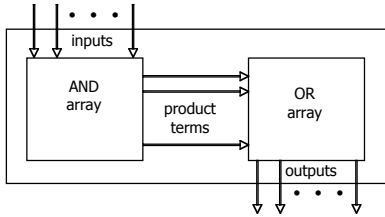


ROM Pros and Cons

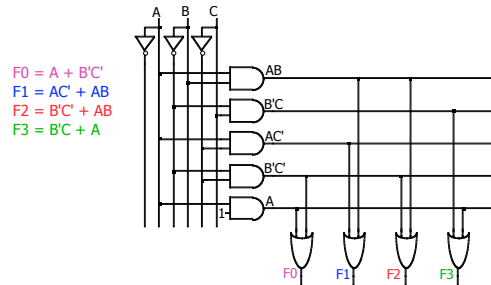
- Pros
 - Simple to use (just build a truth table for your functions)
 - Guaranteed to be able to implement any functions of a given number of inputs
- Cons
 - Large: Size is proportional to $2^{\#}$ of inputs
 - Inefficient: can't exploit don't cares or “structure” of a function

PLAs and PALs

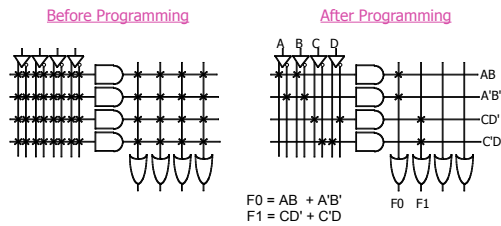
- “Programmable Logic Array” and “Programmable Array Logic”



Program with Wire Connections



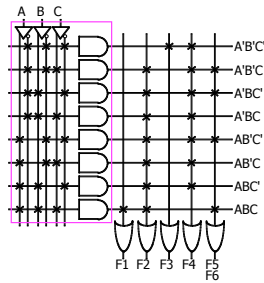
Drawing PLA “Programs”



Bigger PLA Example

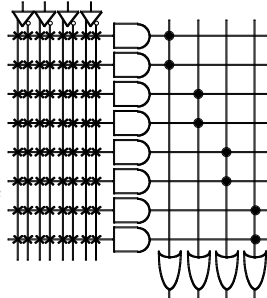
$F1 = ABC$
 $F2 = A + B + C$
 $F3 = A' B' C'$
 $F4 = A' + B' + C'$
 $F5 = A \text{ xor } B \text{ xor } C$
 $F6 = A \text{ xnor } B \text{ xnor } C$

| A | B | C | F1 | F2 | F3 | F4 | F5 | F6 |
|---|---|---|----|----|----|----|----|----|
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |

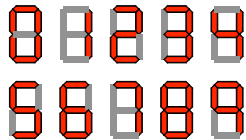
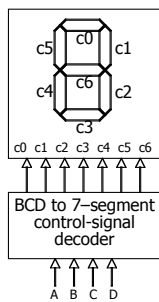


The Difference Between PLAs and PALs

- PLAs
 - Programmable AND and OR
- PALs
 - Only programmable ANDs
 - ORs hard-wired



BCD to SSD Converter



BCD to SSD Truth Table

| A | B | C | D | f0 | f1 | f2 | f3 | f4 | f5 | f6 |
|---|---|---|---|----|----|----|----|----|----|----|
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | - | - | - | - | - | - | - | - |
| 1 | 1 | - | - | - | - | - | - | - | - | - |

PLAs and PALs

- Pros
 - More efficient than ROMs for many “realistic” functions
 - Still quite flexible
- Cons
 - Cannot implement all functions of N inputs
 - Requires logic minimization for efficiency

Additional PLA Features

- Feedback terms
 - Used to implement multi-level logic
- Registered outputs
 - Used to implement memory/sequential logic
- Tri-state outputs
 - Used to allow multiple circuits to share a single output wire (very carefully)

Thank You for Your Attention

- Read lab 3
- Continue homework 3
- Continue reading the book