Lecture 13: Memory and Programmable Logic

Syed M. Mahmud, Ph.D
ECE Department
Wayne State University

Aby K George, ECE Department, Wayne State University
Contents

• Introduction
• Random Access Memory
• Memory Decoding
• Read Only Memory
• Programmable Logic Array
• Programmable Array Logic
• Sequential Programmable Devices
Introduction

• Memory unit: A device to which binary information is stored, and from which information is retrieved when needed for processing.

• Two types of Memories
  • Random Access Memory (RAM)
  • Read Only Memory (ROM)

• RAM can perform read and write operations

• ROM is a programmable logic device (PLD)

• Other types of PLDs
  • Programmable Logic Array (PLA)
  • Programmable Array Logic (PAL)
  • Field Programmable Gate Array (FPGA)
Random Access Memory (RAM)

- Information can be selectively retrieved from any of its internal location (any random location).
- Binary information is stored in groups of words.
- For $k$ address lines, $2^k$ words are possible.
- 1 byte = 8 bits
Random Access Memory (RAM)

• The address line select one particular word.

• Each word in memory is assigned an identification number called address. For $k$ address lines the address will be from 0 to $2^k - 1$.

1 byte (B) = 8 bits (b)
1 Kilobyte (K / KB) = $2^{10}$ bytes = 1,024 bytes
1 Megabyte (M / MB) = $2^{20}$ bytes
1 Gigabyte (G / GB) = $2^{30}$ bytes
1 Terabyte (T / TB) = $2^{40}$ bytes

1K X 16 Memory
Write and Read Operations in RAM

• Write operation
  • Apply the binary address of the desired word to the address lines.
  • Apply the data bits that must be stored in memory to the data input lines.
  • Activate the write input.

• Read Operation
  • Apply the binary address of the desired word to the address lines.
  • Activate the read input.

<table>
<thead>
<tr>
<th>Control Inputs to Memory Chip</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory Enable</td>
</tr>
<tr>
<td>----------------</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>1</td>
</tr>
</tbody>
</table>
Types of RAMs

• Static RAM (SRAM)
  • Consists of internal latches.
  • Information is available as long as the power is applied.
  • Have shorter read/write cycles.

• Dynamic RAM (DRAM)
  • Stores information in the form of electric charges on capacitors.
  • Stored charge in capacitor tends to discharge with time, and the capacitors must be periodically recharged by refreshing the dynamic memory.
  • Reduced power consumption and larger storage capacity.
RAM – Memory Cell

• Binary cell is modeled by an SR Latch with associated gates to form D Latch.

• The Binary cell stores 1-bit data.
Design of a 4 x 4 RAM

- 4 – Address Lines
- 4 – Outputs

$2^m \times n$ RAM consists of $m \times 2^m$ decoder, $n$ input lines and $n$ output lines
Decoder for RAM

- \( k \) inputs -> \( 2^k \) outputs
- \( 2^k \) - AND gates with \( k \) inputs
- The total number of gates and the number of inputs per gate can be reduced by using two decoders in a two-dimensional selection scheme.
- 10 X 1024 decoder using two 5 X 32 decoders.
Read Only Memory (ROM)

- Permanent storage of binary information.

- 5 x 32 Decoder
- 8-OR gates with 32 inputs for each
- 256 internal connections
- Connections are programmable
- Simplest way is to use fuses.
- High Voltage is applied to blown the fuse.

![Diagram of a 32 x 8 ROM](image.png)

Internal logic of a 32 x 8 ROM
# Read Only Memory

## ROM Truth Table (Partial)

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>( l_4 )</td>
<td>( l_3 )</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Combinational Circuit Implementation in ROM

• $F_1 = \Sigma(0,1,2,4,7)$
• $F_2 = \Sigma(1,2,4,6,7)$
Types of ROMs

• Paths in ROM can be programmed in four ways

• Mask Programming
  • Done during fabrication.
  • Costly method and useful when large quantity of the same ROM is required.

• Programmable Read Only Memory (PROM)
  • Fuses are blown by high voltage pulses.
  • PROM is programmable in the library.

• Erasable PROM (EPROM)
  • Once programmed, EPROM can be restructured to the initial state.
  • UV light is used for erasing.

• Electrically Erasable PROM (EEPROM)
  • Programmed connections can be erased with an electrical signal.
Programmable Logic Devices

(a) Programmable read-only memory (PROM)

(b) Programmable array logic (PAL)

(c) Programmable logic array (PLA)
Programmable Logic Array (PLA)

- $F_1 = AB' + AC + A'BC'$
- $F_2 = (AC + BC)'$

<table>
<thead>
<tr>
<th>Product Term</th>
<th>Inputs</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$A$</td>
<td>$B$</td>
</tr>
<tr>
<td>$AB'$</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$AC$</td>
<td>2</td>
<td>-</td>
</tr>
<tr>
<td>$BC$</td>
<td>3</td>
<td>-</td>
</tr>
<tr>
<td>$A'BC'$</td>
<td>4</td>
<td>0</td>
</tr>
</tbody>
</table>
Programmable Logic Array (PLA)

\[ F_1 = \Sigma(0,1,2,4) \]
\[ F_2 = \Sigma(0,5,6,7) \]

The combination that gives minimum number of product terms is \( F_1' \) and \( F_2 \)

\[ F_1 = A'B' + A'C' + B'C' \]
\[ F_1' = AB + BC + AC \]

\[ F_2 = AB + AC + A'B'C' \]
\[ F_2' = A'B + A'C + AB'C' \]
Sequential Programmable Devices

• Sequential Programmable Logic Device (SPLD)
  • Includes AND-OR array (PAL or PLA) and flip-flops

• Complex Programmable Logic Device (CPLD)
  • Collection of PLDs on a single integrated circuit and I/O blocks.

• Field Programmable Gate Array (FPGA)
  • Consists of lookup tables, multiplexers, gates and flip-flops.

• The design with PLD, CPLD, or FPGA requires extensive computer-aided design (CAD) tools to facilitate the synthesis procedure.
Summary

• How to implement a combinational function in a ROM?
• What is the difference between ROM, PLA and PAL?
• What are the different types of ROMs?
• How to implement a combinational function in a PLA?
• What are the common sequential programmable devices?
Homework – 7

• 7.1, 7.19, 7.20, 7.21
• Design a Full Adder using ROM and PLA
• Design a 5X2 RAM using D Flip-flop