**Figure 3.1** Logic values as voltage levels

<table>
<thead>
<tr>
<th>Voltage</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{DD}$</td>
<td>Logic value 1</td>
</tr>
<tr>
<td>$V_{DD}$</td>
<td>Undefined</td>
</tr>
<tr>
<td>$V_{SS}$</td>
<td>Logic value 0</td>
</tr>
</tbody>
</table>

**Figure 3.2** NMOS transistor as a switch

(a) A simple switch controlled by the input

(b) NMOS transistor

(c) Simplified symbol for an NMOS transistor

**Figure 3.3** PMOS transistor as a switch

(a) A switch with the opposite behavior of Figure 3.2a

(b) PMOS transistor

(c) Simplified symbol for an PMOS transistor

**Figure 3.4** NMOS and PMOS transistors in logic circuits

(a) NMOS transistor

(b) PMOS transistor

**Figure 3.5** A NOT gate built using NMOS technology

(a) Circuit diagram

(b) Simplified circuit diagram

(c) Graphical symbols
Figure 3.6 NMOS realization of a NAND gate

Figure 3.7 NMOS realization of a NOR gate

Figure 3.8 NMOS realization of an AND gate

Figure 3.9 NMOS realization of an OR gate

Figure 3.10 Structure of an NMOS circuit

Figure 3.11 Structure of a CMOS circuit
Figure 3.12 CMOS realization of a NOT gate

Figure 3.13 CMOS realization of a NAND gate

Figure 3.14 CMOS realization of a NOR gate

Figure 3.15 CMOS realization of an AND gate

Figure 3.16 A CMOS complex gate

Figure 3.17 A CMOS complex gate
Figure 3.18 Voltage levels in a CMOS circuit

(a) Circuit
(b) Voltage levels

Figure 3.19 Interpretation of voltage levels

(b) Positive logic truth table and gate symbol
(c) Negative logic truth table and gate symbol

Figure 3.20 Interpretation of voltage levels

(a) Voltage levels
(b) Positive logic
(c) Negative logic

Figure 3.43a NMOS transistor when turned off

(a) When $V_{GS} = 0$ V, the transistor is off

Figure 3.43b NMOS transistor when turned on

(b) When $V_{GS} = 5$ V, the transistor is on

Figure 3.44 Current-voltage relationship in the NMOS transistor
Figure 3.45 Voltage levels in the NMOS inverter

(a) NMOS NOT gate  
(b) $V_x = 5$ V

Figure 3.46 Voltage transfer characteristics for the CMOS inverter

(a) NMOS NOT gate driving another NOT gate

(b) The capacitive load at node A

Figure 3.47 Parasitic capacitance in integrated circuits

Figure 3.48 Voltage waveforms for logic gates

(a) Small transistor  
(b) Larger transistor

Figure 3.49 Transistor sizes
Figure 3.50: Dynamic current flow in CMOS circuits

Figure 3.51: Poor use of NMOS and PMOS transistors

(a) NMOS transistor  (b) PMOS transistor

Figure 3.52: Poor implementation of a CMOS AND gate

(a) An AND gate circuit  (b) Truth table and voltage levels

Figure 3.53: High fan-in NMOS NAND gate

Figure 3.54: High fan-in NMOS NOR gate
Figure 3.55: The effect of fan-out on propagation delay

Figure 3.56: A noninverting buffer

Figure 3.57: Tri-state buffer

Figure 3.58: Four types of tri-state buffers

Figure 3.59: An application of tri-state buffers

Figure 3.60: A transmission gate
Figure 3.61  
(a) Truth table  
<table>
<thead>
<tr>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$f = x_1 \oplus x_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

(b) Graphical symbol  

(c) Sum-of-products implementation  

$\sum_{x_1, x_2} f = x_1 \lor x_2$  

(d) CMOS implementation  

Figure 3.61b  
CMOS Exclusive-OR gate

Figure 3.62  
A 2-to-1 multiplexer built using transmission gates

Figure 3.21  
(a) Dual-inline package  
(b) Structure of 7404 chip  
(c) 7400-series chip

Figure 3.22  
Implementation of $f = x_1 x_2 + x_1 x_3$

Figure 3.23  
The 74244 buffer chip
Logic gates and programmable switches

Inputs (logic variables)

Outputs (logic functions)

Figure 3.24 Programmable logic device as a black box

Figure 3.25 General structure of a PLA

Figure 3.26 Gate-level diagram of a PLA

Figure 3.27 Customary schematic of a PLA

(a) Programmable NOR-plane

(b) A programmable switch

(c) EEPROM transistor
Figure 3.65 A programmable version of a NOR-NOR PLA

Figure 3.66 A NOR-NOR PLA used for sum-of-products

Figure 3.28 An example of a PAL

Figure 3.67 PAL programmed to implement two functions

Figure 3.29 Output circuitry

Figure 3.30 A PLD programming unit
Figure 3.31 A PLCC package with socket

Figure 3.32 Structure of a CPLD

Figure 3.33 A section of a CPLD

Figure 3.34 CPLD packaging and programming

Figure 3.35 Structure of an FPGA

Figure 3.36 A two-input lookup table
Custom Chips

- Created from scratch.
- Designer selects number, placement, and connections for each and every transistor.
- Are most dense and highest speed.
- Requires a substantial design effort.
- Used only when high performance and density is required: processors/memories.
Standard-Cell Chips

- Gates prebuilt and stored in a library.
- Gates needed for a design are selected and placed, and wires are routed between them.
- Standard-cell chips often called application specific integrated circuits (ASICs).
- Saves time since gates are reused.
- CAD tools exist to place and route gates.

Gate-Arrays

- Parts of chip are prefabricated (transistors).
- Parts of chip are custom fabricated (wires).
- Provides cost savings since all template wafers are identical.
- Many variants exist.

Concluding Remarks

- Introduced basics of using transistors for digital logic.
- Described several types of IC chips:
  - Standard chips from the 7400 series.
  - Various types of PLDs.
  - Custom and semi-custom chips.
Figure P3.11 Circuit for problem 3.55