ECE253—Digital Systems

Jonah Chen

September 23, 2021

Contents

1 Number Conversions

- Computers use binary. We use hexadecimal to make it less error-prone writing binary numbers.
- Convert binary to hex: grouping four bits together makes the conversion easier. $0101\,1110 = 5e$.
- Converting hex to binary: $a6 = 1010\,0110$
- Converting binary to decimal: find the bit positions for all the "1"s. $101\,0111 = 2^6 + 2^4 + 2^2 + 2^1 + 2^0 = 87$.
- Converting decimal to binary: Repeatedly divide by two and get quotient and remainder. The remainders from the binary digits from least significant to most significant bits.

$$
\begin{aligned}
76/2 &= 38\\
38/2 &= 19\\
19/2 &= 9 + 1/2\\
9/2 &= 4 + 1/2\\
4/2 &= 2\\
2/2 &= 1\\
1/2 &= 0 + 1/2
\end{aligned}
$$

  Thus, $76 = 1001100$.

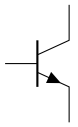
- Converting hex to decimal: $3e = 3 \times 16 + 14 = 62$.
- Converting decimal to hex: We can use algorithmic way, which is repeatedly dividing by sixteen and take the remainders to extract the hex digits but dividing by 16 is very difficult. So, first convert to binary then convert to hex. $96 = 110\,0000 = 60$

2 Binary addition

- Each step of computation has three inputs and two outputs.

3 Primative logic gates

x	y	$x+y$	xy	$\overline{x}$	$\overline{y}$
0	0	0	0	1	1
0	1	1	0	1	0
1	0	1	0	0	1
1	1	1	1	0	0



Example 1

Say X is a 3-bit number, with bits $x_2x_1x_0$. Design a circuit with X as input and one output f. f should be 1 if $X \geq 5$, otherwise f should be 0.

	x_2	x_1	x_0	f
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	0
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

$f = x_2(x_1 + x_0)$

- Equation
-
- Timing Diagram

4 Boolean Algebra

Definition:

- A literal is a variable or not a variable
- A product term is a term that contains the "and" of literals
- A minterm is a product term that contains all the literals. Because there is a minimum set of input combinations (exactly 1 row of truth table) that cause the minterm to be true.
- A maxterm is a sum term that contains all the literals. Because there is a maximum set of input combinations (all but 1 truth table) that cause the maxterm to be true.

x_2	x_1	x_0	minterm	label	maxterm	label
0	0	0	$\overline{x_2}\,\overline{x_1}\,\overline{x_0}$	m_0	$x_2 + x_1 + x_0$	M_0
0	0	1	$\overline{x_2}\,\overline{x_1}x_0$	m_1	$x_2 + x_1 + \overline{x_0}$	M_1
0	1	0				
0	1	1				
1	0	0				
1	0	1				
1	1	0	$x_2x_1\overline{x_0}$	m_6	$\overline{x_2} + \overline{x_1} + x_0$	M_6
1	1	1	$x_2x_1x_0$	m_7	$\overline{x_2} + \overline{x_1} + \overline{x_0}$	M_7

Example 2

x_2	x_1	x_0	f	label
0	0	0	0	m_0
0	0	1	0	m_1
0	1	0	1	m_2
0	1	1	1	m_3
1	0	0	1	m_4
1	0	1	0	m_5
1	1	0	1	m_6
1	1	1	1	m_7

Canonical SOP form:

$$\begin{aligned}
f &= \sum m(2,3,5,6,7) && (4.1)\\
&= \overline{x_2}x_1\overline{x_0} + \overline{x_2}x_1x_0 + x_2\overline{x_1}\,\overline{x_0} + x_2x_1\overline{x_0} + x_2x_1x_0 && (4.2)\\
&= \overline{x_2}x_1(\overline{x_0} + x_0) + x_2x_1(\overline{x_0} + x_0) + x_2\overline{x_1}\,\overline{x_0} && (4.3)\\
&= \overline{x_2}x_1 + x_2x_1 + x_2\overline{x_1}\,\overline{x_0} && (4.4)\\
&= x_1 + x_2\overline{x_1}\,\overline{x_0} && (4.5)\\
&= x_1 + x_2\overline{x_0} && (4.6)
\end{aligned}$$

Canonical POS form:

$$\begin{aligned}
f &= \prod M(0,1,5) && (4.7)\\
&= (x_2 + x_1 + x_0)(x_2 + x_1 + \overline{x_0})(\overline{x_2} + x_1 + \overline{x_0}) && (4.8)
\end{aligned}$$

Proposition:

$$a + bc = (a+b)(a+c) \tag{4.9}$$

Proof.

$$\begin{aligned}
(a+b)(a+c) &= a + ab + ac + bc && (4.10)\\
&= a(1+b) + a(1+c) + bc && (4.11)\\
&= a + bc && (4.12)
\end{aligned}$$

□

Proposition: [De-Morgan's Law]

$$\overline{ab} = \overline{a} + \overline{b} \tag{4.13}$$

$$\overline{(a+b)} = \overline{a}\,\overline{b} \tag{4.14}$$

Note this law is the means of translation between "and" and "or".

Example 3

Gumball Factory: three sensors

- s_2: ball too large
- s_1: ball too small
- s_0: ball too light

Design logic circuit with output one if ball is too large or it's too small and too light.

Theorem: [Consensus]

$$xy + \overline{x}z + yz = xy + \overline{x}z \tag{4.15}$$

Proof. Use the property $1f = f$, $a + \overline{a} = 1$.

$$\begin{aligned} xy + \overline{x}z + yz = xy + \overline{x}z &= xy + \overline{x}z + (x + \overline{x})yz && (4.16) \\ &= xy + xyz + \overline{x}z + \overline{x}yz && (4.17) \\ &= xy + x\overline{z} && (4.18) \end{aligned}$$

□

Example 4

$$\begin{aligned} f &= \overline{a}b + \overline{a}c + \overline{b}c && (4.19) \\ &= \overline{a}b + \overline{a}c(b + \overline{b}) + \overline{b}c && (4.20) \\ &= \overline{a}b + \overline{a}cb + \overline{a}c\overline{b} + \overline{b}c && (4.21) \\ &= \overline{a}b(1 + c) + \overline{b}c(\overline{a} + 1) && (4.22) \\ &= \overline{a}b + \overline{b}c && (4.23) \end{aligned}$$

4.1 Karnaugh maps

5 Multiplexers

- Recall 2-to-1 multiplexer $f = a\overline{s} + bs$.
- 4-to-1 multiplexer can be constructed with three 2-to-1 multiplexer.