Half adder (without carry as input)

(a) The four possible cases

(b) Truth table
Full Adders (W/ carry as input) – Truth Table

Addition at the i-th bit:

<table>
<thead>
<tr>
<th>$c_i$</th>
<th>$x_i$</th>
<th>$y_i$</th>
<th>$c_{i+1}$</th>
<th>$s_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0 0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0 1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0 1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1 0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1 0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1 1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1 1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

(a) Truth table

\[ s_i = x_i \oplus y_i \oplus c_i \]

\[ c_{i+1} = x_i y_i + x_i c_i + y_i c_i \]

(b) Karnaugh maps
Full Adders (FA) – Circuit

\[ s_i = x_i \oplus y_i \oplus c_i \]

\[ c_{i+1} = x_i y_i + x_i c_i + y_i c_i \]

(b) Karnaugh maps

(c) Circuit
Alternative Implementation of Adders

Figure 5.5. A decomposed implementation of the full-adder circuit.

(a) Block diagram based on expressions of $S_i$ and $C_{i+1}$

(b) Detailed diagram
Example for multi-bit addition

\[ X = x_4 x_3 x_2 x_1 x_0 = 01111 \quad (15)_{10} \]
\[ + Y = y_4 y_3 y_2 y_1 y_0 = 01010 \quad (10)_{10} \]
\[ = 1110 \quad \text{Generated carries} \]
\[ S = s_4 s_3 s_2 s_1 s_0 = 11001 \quad (25)_{10} \]

Figure 5.3. An example of addition.
Ripple Carry Adders

Figure 5.6. An $n$-bit ripple-carry adder.
5.3. Formats for representation of integers

(a) Unsigned number

(b) Signed number

0 denotes +
1 denotes −
5.3.1 Negative Number Representations

- $b_{n-1} = 0$ denote “+”; $b_{n-1} = 1$ denote “-”

1. Sign-and-magnitude representation
   - $b_i \rightarrow b_i$ for $i = 0 \sim (n-2)$

2. 1’s complement representation
   - $b_i \rightarrow \text{not } (b_i)$ for $i = 0 \sim (n-2)$

3. 2’s complement representation
   - $b_i \rightarrow \text{not } (b_i)$ for $i = 0 \sim (n-2)$, then $+1$ at $b_0$
Interpretation of four-bit (n=4) signed integers

<table>
<thead>
<tr>
<th>$b_3b_2b_1b_0$</th>
<th>Sign and magnitude</th>
<th>1’s complement</th>
<th>2’s complement</th>
</tr>
</thead>
<tbody>
<tr>
<td>0111</td>
<td>+7</td>
<td>+7</td>
<td>+7</td>
</tr>
<tr>
<td>0110</td>
<td>+6</td>
<td>+6</td>
<td>+6</td>
</tr>
<tr>
<td>0101</td>
<td>+5</td>
<td>+5</td>
<td>+5</td>
</tr>
<tr>
<td>0100</td>
<td>+4</td>
<td>+4</td>
<td>+4</td>
</tr>
<tr>
<td>0011</td>
<td>+3</td>
<td>+3</td>
<td>+3</td>
</tr>
<tr>
<td>0010</td>
<td>+2</td>
<td>+2</td>
<td>+2</td>
</tr>
<tr>
<td>0001</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
</tr>
<tr>
<td>0000</td>
<td>+0</td>
<td>+0</td>
<td>+0</td>
</tr>
<tr>
<td>1000</td>
<td>-0</td>
<td>-7</td>
<td>-8</td>
</tr>
<tr>
<td>1001</td>
<td>-1</td>
<td>-6</td>
<td>-7</td>
</tr>
<tr>
<td>1010</td>
<td>-2</td>
<td>-5</td>
<td>-6</td>
</tr>
<tr>
<td>1011</td>
<td>-3</td>
<td>-4</td>
<td>-5</td>
</tr>
<tr>
<td>1100</td>
<td>-4</td>
<td>-3</td>
<td>-4</td>
</tr>
<tr>
<td>1101</td>
<td>-5</td>
<td>-2</td>
<td>-3</td>
</tr>
<tr>
<td>1110</td>
<td>-6</td>
<td>-1</td>
<td>-2</td>
</tr>
<tr>
<td>1111</td>
<td>-7</td>
<td>-0</td>
<td>-1</td>
</tr>
</tbody>
</table>
1’s complement addition (examples)

\[
\begin{align*}
(+5) \quad &\quad 0101 \\
(+2) \quad &\quad + \quad 0010 \\
\hline
(+7) \quad &\quad 0111
\end{align*}
\begin{align*}
(-5) \quad &\quad 1010 \\
(+2) \quad &\quad + \quad 0010 \\
\hline
(-3) \quad &\quad 1100
\end{align*}
\begin{align*}
(+5) \quad &\quad 0101 \\
(-2) \quad &\quad + \quad 1101 \\
\hline
(+3) \quad &\quad 10010
\end{align*}
\begin{align*}
(-5) \quad &\quad 1010 \\
(-2) \quad &\quad + \quad 1101 \\
\hline
(-7) \quad &\quad 10111
\end{align*}
\begin{align*}
\text{carry} \quad &\quad 1 \quad \rightarrow 1 \\
\hline
0011
\end{align*}
\begin{align*}
\text{carry} \quad &\quad 1 \quad \rightarrow 1 \\
\hline
1000
\end{align*}
### 2’s complement addition (examples)

<table>
<thead>
<tr>
<th>Operation</th>
<th>5</th>
<th>0 1 0 1</th>
<th>2’s Complement</th>
<th>Operation</th>
<th>11</th>
<th>1 0 1 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>(+ 5)</td>
<td>+ (+ 2)</td>
<td>0 0 1 0</td>
<td>(+ 7) 0 1 1 1</td>
<td>+ (+ 2)</td>
<td>0 0 1 0</td>
<td>(+ 3) 1 0 0 1 1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Operation</th>
<th>5</th>
<th>0 1 0 1</th>
<th>2’s Complement</th>
<th>Operation</th>
<th>7</th>
<th>1 0 1 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>(+ 5)</td>
<td>+ (-2)</td>
<td>1 1 1 0</td>
<td>(+ 3) 1 0 0 1 1</td>
<td>+ (-2)</td>
<td>1 1 1 0</td>
<td>(+ 3) 1 1 0 0 1</td>
</tr>
</tbody>
</table>

Ignore: ignore
Arithmetic Overflows when result goes beyond the finite range: $-2^{(n-1)}$~$2^{(n-1)}$ - 1

<table>
<thead>
<tr>
<th></th>
<th>0 1 1 1</th>
<th></th>
<th>1 0 0 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>(+ 7)</td>
<td>0 1 1 1</td>
<td>(+ 9)</td>
<td>1 0 0 1</td>
</tr>
<tr>
<td>+ (+ 2)</td>
<td>+ 0 0 1 0</td>
<td>+ (+ 2)</td>
<td>+ 0 0 1 0</td>
</tr>
<tr>
<td></td>
<td>1 0 0 1</td>
<td>(+ 5)</td>
<td>1 0 1 0 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(−5)</td>
<td>1 0 1 1</td>
</tr>
</tbody>
</table>

$c_4 = 0$
$c_3 = 1$

$c_4 = 0$
$c_3 = 0$

<table>
<thead>
<tr>
<th></th>
<th>0 1 1 1</th>
<th></th>
<th>1 0 0 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>(+ 7)</td>
<td>0 1 1 1</td>
<td>(+ 5)</td>
<td>1 0 1 0 1</td>
</tr>
<tr>
<td>+ (−2)</td>
<td>+ 1 1 1 0</td>
<td>+ (−2)</td>
<td>+ 1 1 1 0</td>
</tr>
<tr>
<td></td>
<td>1 0 1 0 1</td>
<td>(−9)</td>
<td>1 0 1 1 1</td>
</tr>
</tbody>
</table>

$c_4 = 1$
$c_3 = 1$

$c_4 = 1$
$c_3 = 0$

**Overflow** = $C_{n-1} \ XOR \ C_n$
2’s complement subtraction

\[
\begin{align*}
\text{(+5)} + (\text{+2}) & \quad 0101 - 0010 \quad \Rightarrow \quad 0101 + 1110 = 10011 \\
\text{(+3)} + (\text{+2}) & \quad 0101 - 0010 \quad \Rightarrow \quad 0101 + 1110 = 10011
\end{align*}
\]

\[
\begin{align*}
\text{ignore} & \\
\text{ignore}
\end{align*}
\]

\[
\begin{align*}
\text{(-5)} + (\text{+2}) & \quad 1011 - 0010 \quad \Rightarrow \quad 1011 + 1110 = 11001 \\
\text{(-7)} + (\text{+2}) & \quad 1011 - 0010 \quad \Rightarrow \quad 1011 + 1110 = 11001
\end{align*}
\]

\[
\begin{align*}
\text{ignore} & \\
\text{ignore}
\end{align*}
\]
Multipliers for unsigned numbers

(a) Multiplication by hand
Multipliers for unsigned numbers

Multiplicand M (11) 1 1 1 0
Multiplier Q (14) 1 0 1 1

Partial product 0 1 1 1 0
+ 1 1 1 0

Partial product 1 1 0 1 0 1
+ 0 0 0 0

Partial product 2 0 1 0 1 0
+ 1 1 1 0

Product P (154) 1 0 0 1 1 0 1 0

(b) Multiplication for implementation in hardware (4bit x 4 bit = 8 bit)
Multipliers for signed numbers

\[
\begin{array}{c}
\text{Multiplicand M} \quad (+14) \\
\text{Multiplier Q} \quad (+11) \\
\hline
0 0 1 1 1 0 \\
0 1 0 1 1
\end{array}
\]

\[
\begin{array}{c}
\text{Partial product 0} \\
0 0 1 1 1 0 \\
+ 0 0 1 1 1 0
\end{array}
\]

\[
\begin{array}{c}
\text{Partial product 1} \\
0 0 1 0 1 0 1 \\
+ 0 0 0 0 0 0
\end{array}
\]

\[
\begin{array}{c}
\text{Partial product 2} \\
0 0 1 0 1 0 0 \\
+ 0 0 1 1 1 0
\end{array}
\]

\[
\begin{array}{c}
\text{Partial product 3} \\
0 0 1 0 0 1 1 \\
+ 0 0 0 0 0 0
\end{array}
\]

\[
\begin{array}{c}
\text{Product P} \quad (+154) \\
0 0 1 0 0 1 1 0 1 0
\end{array}
\]

(a) Positive multiplicand
Multipliers for signed numbers

<table>
<thead>
<tr>
<th>Multiplicand M</th>
<th>$(-14)$</th>
<th>(10010)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiplier Q</td>
<td>$(+11)$</td>
<td>(01011))</td>
</tr>
<tr>
<td>Partial product 0</td>
<td>(1110010)</td>
<td>(+110010)</td>
</tr>
<tr>
<td>Partial product 1</td>
<td>(1101011)</td>
<td>(+000000)</td>
</tr>
<tr>
<td>Partial product 2</td>
<td>(1110101)</td>
<td>(+110010)</td>
</tr>
<tr>
<td>Partial product 3</td>
<td>(1101100)</td>
<td>(+000000)</td>
</tr>
<tr>
<td>Product P</td>
<td>$(-154)$</td>
<td>(1101100110)</td>
</tr>
</tbody>
</table>

(b) Negative multiplicand
IEEE Standard floating-point formats: 
\[ \text{mantissa} \times R^{(\text{exponent})} \text{ where } R=\text{radix} \]

(a) Single precision

(b) Double precision
Chapter 7 – Sequential circuits

- Combinational circuits studied so far in previous chapters have their inputs only depending on the present inputs.
- Sequence circuits’ output depend on NOT only the present inputs, BUT ALSO the previous STATE of the circuits.
- Sequence circuits introduce the two new concepts:
  - Memory/storage elements saving the states
  - State/State-Transitions and Sequences
Control of an alarm system (example)

Figure 7.1. Control of an alarm system.
Memory element

Figure 7.2. A simple memory element.
7.1 Basic Latches

Figure 7.4. A memory element with NOR gates.
Redrawing the memory element in cross-connected style without changing its topology, we obtain the SR latch.
Figure 7.5. A latch built with NOR gates -- SR Latch

(a) SR Latch Circuit

(b) Truth/Characteristic Table

<table>
<thead>
<tr>
<th>S</th>
<th>R</th>
<th>Q_a</th>
<th>Q_b</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0/1</td>
<td>1/0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

(c) Timing diagram

Figure 7.5. A latch built with NOR gates.
Gated SR Latch

(a) Circuit

(b) Characteristic table

<table>
<thead>
<tr>
<th>Clk</th>
<th>S</th>
<th>R</th>
<th>(Q(t + 1))</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>x</td>
<td>x</td>
<td>(Q(t)) (no change)</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>(Q(t)) (no change)</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>(x)</td>
</tr>
</tbody>
</table>
Gated SR Latch

(a) Circuit

Time
Gated SR latch with NAND gates

Figure 7.7. Gated SR latch with NAND gates.
Gated D latch

(a) Circuit

<table>
<thead>
<tr>
<th>Clk</th>
<th>D</th>
<th>Q(t+1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>x</td>
<td>Q(t)</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

(b) Characteristic table

(c) Graphical symbol
Gated D latch

(a) Circuit

(d) Timing diagram
Setup and hold times

Figure 7.9. Setup and hold times.
Master-slave D flip-flop

(a) Circuit

(b) Timing diagram

(c) Graphical symbol
A positive-edge-triggered D flip-flop

Figure 7.11. A positive-edge-triggered D flip-flop.
Master-slave D flip-flop with *Clear* and *Preset*

(a) Circuit

(b) Graphical symbol
Synchronous reset for a D flip-flop

Figure 7.15. Synchronous reset for a D flip-flop.
T flip-flop

(a) Circuit

(b) Truth table

<table>
<thead>
<tr>
<th>T</th>
<th>Q(t + 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Q(t)</td>
</tr>
<tr>
<td>1</td>
<td>\overline{Q(t)}</td>
</tr>
</tbody>
</table>

(c) Graphical symbol
T flip-flop

(a) Circuit

(b) Timing diagram
JK flip-flop

(a) Circuit

(b) Truth table

<table>
<thead>
<tr>
<th>J</th>
<th>K</th>
<th>Q(t+1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Q(t)</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Q(t)</td>
</tr>
</tbody>
</table>

(c) Graphical symbol