Today

• Binary addition
• Representing negative numbers
Binary Addition

Consider the following binary numbers:

\begin{align*}
0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 & 1
\end{align*}

How do we add these numbers?
Binary Addition

0 0 1 0 0 1 1 0
0 0 1 0 1 0 1 1

1
Binary Addition

0 0 1 0 0 1 1 0
0 0 1 0 1 0 1 1
\[\downarrow\]
0 1

And we have a carry now!
Binary Addition

\[
\begin{array}{cccccccc}
0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\
\end{array}
\]

And we have a carry again!
Binary Addition

0 0 1 0 0 1 1 0
0 0 1 0 1 0 1 1

\[ \begin{array}{cccccccc}
0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\
\end{array} \]

0 0 0 1

and again!
Binary Addition

\[
\begin{array}{cccccccc}
0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\
\end{array}
\]

\[\downarrow\]

1 0 0 0 1
Binary Addition

\[
\begin{array}{cccccc}
0 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 1 & 1 \\
\end{array}
\]

One more carry!
Binary Addition

0 0 1 0 0 1 1 0
0 0 1 0 1 0 1 1
\[\downarrow\downarrow\]
0 1 0 1 0 0 0 1
Binary Addition

Behaves just like addition in decimal, but:

• We carry to the next digit any time the sum of the digits is 2 (decimal) or greater
Negative Numbers

So far we have only talked about representing non-negative integers.

• What can we add to our binary representation that will allow this?
Representing Negative Numbers

One possibility:

• Add an extra bit that indicates the sign of the number

• We call this the “sign-magnitude” representation
Sign Magnitude Representation

+12 0 0 0 1 1 0 0
Sign Magnitude Representation

+12  0 0 0 0 1 1 0 0

-12  1 0 0 0 1 1 0 0
Sign Magnitude Representation

+12 0 0 0 0 1 1 0 0

-12 1 0 0 0 1 1 0 0

What is the problem with this approach?
Sign Magnitude Representation

What is the problem with this approach?

• Some of the arithmetic operators that we have already developed do not do the right thing
Sign Magnitude Representation

Operator problems:
• For example, we have already designed a counter (that implements an ‘increment’ operation)

-12  1 0 0 0 1 1 0 0
Sign Magnitude Representation

Operator problems:

-12 \quad 1 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0

Increment
Sign Magnitude Representation

Operator problems:

-12

1 0 0 0 1 1 0 0

Increment

1 0 0 0 1 1 0 1
Sign Magnitude Representation

Operator problems:

-12 $\overset{1}{\overset{1}{0}}\overset{0}{\overset{0}{0}}\overset{1}{\overset{1}{1}}\overset{0}{\overset{0}{0}}$

-13 $\overset{1}{\overset{1}{0}}\overset{0}{\overset{0}{0}}\overset{1}{\overset{1}{1}}\overset{0}{\overset{1}{0}}$

!!!
Representing Negative Numbers

An alternative:

• When taking the additive inverse of a number, invert all of the individual bits
• The leftmost bit still determines the sign of the number
One’s Complement Representation

12  0 0 0 0 1 1 0 0

-12  1 1 1 1 0 0 1 1

Invert
One’s Complement Representation

12

-12

0 0 0 0 1 1 0 0

Invert

1 1 1 1 0 0 1 1

Increment

1 1 1 1 0 1 0 0
One’s Complement Representation

12  
\[0\quad 0\quad 0\quad 0\quad 1\quad 1\quad 0\quad 0\]

\[\text{Invert}\]

-12  
\[1\quad 1\quad 1\quad 1\quad 0\quad 0\quad 1\quad 1\]

\[\text{Increment}\]

-11  
\[1\quad 1\quad 1\quad 1\quad 0\quad 1\quad 0\quad 0\]
One’s Complement Representation

What problems still exist?
One’s Complement Representation

What problems still exist?

• We have two distinct representations of ‘zero’:

\[
\begin{array}{c}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{array}
\]
One’s Complement Representation

What problems still exist?

• We can’t directly add a positive and a negative number:

\[
\begin{array}{c}
12 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
+ & & & & & & + \\
-5 & 1 & 1 & 1 & 1 & 0 & 1 & 0
\end{array}
\]
One’s Complement Representation

12         0 0 0 0 1 1 0 0
+         +
-5         1 1 1 1 1 0 1 0

0 0 0 0 0 1 1 0
One’s Complement Representation

12  0 0 0 0 1 1 0 0
+    +
-5  1 1 1 1 1 0 1 0

6  0 0 0 0 1 1 0
    !!!!
Representing Negative Numbers

An alternative:
(a little intuition first)

0 0 0 0 0 0 0 0 0

Decrement
Representing Negative Numbers

An alternative:

(a little intuition first)

0 0 0 0 0 0 0 0

Decrement

1 1 1 1 1 1 1 1
Representing Negative Numbers

An alternative:
(a little intuition first)

```
0   0   0   0   0   0   0   0
-1  1   1   1   1   1   1   1
```

Define this as

```
-1  1   1   1   1   1   1   1
```

Decrement
Representing Negative Numbers

A few more numbers:

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>-2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>-3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
Two’s Complement Representation

In general, how do we take the additive inverse of a binary number?
Two’s Complement Representation

In general, how do we take the additive inverse of a binary number?

- Invert each bit and then add ‘1’
Two’s Complement Representation

Invert each bit and then add ‘1’

5 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1

-5 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 \quad 1

Two’s complement
Two’s Complement Representation

Now: let’s try adding a positive and a negative number:

\[
\begin{array}{c}
12 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
+ & + & & & & & & \\
-5 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\
\end{array}
\]
Two’s Complement Representation

Now: let’s try adding a positive and a negative number:

\[
\begin{array}{c}
12 \\
+ \\
-5 \\
\end{array}
\begin{array}{c}
00001100 \\
+ \\
11111011 \\
\end{array}
\begin{array}{c}
00000111 \\
\end{array}
\]
Two’s Complement Representation

Now: let’s try adding a positive and a negative number:

\[
\begin{array}{c}
12 \\
+ \\
-5
\end{array} \quad \begin{array}{c}
0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0 \\
+ \\
1 \ 1 \ 1 \ 1 \ 1 \ 0 \ 1 \ 1
\end{array}
\]

\[
7 \quad \begin{array}{c}
0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1
\end{array}
\]
Two’s Complement Representation

Two’s complement is used for integer representation in today’s processors
Two’s Complement Representation

Two’s complement is used for integer representation in today’s processors

One oddity: we can represent one more negative number than we can positive numbers
Implementing Subtraction

How do we implement a ‘subtraction’ operator?
(e.g., A – B)
Implementing Subtraction

How do we implement a ‘subtraction’ operator?
(e.g., A – B)

• Take the 2s complement of B
• Then add this number to A