SIGN EXTENSION

Sign Extension

- Needed for 2’s complement addition
- Consider case of adding two numbers of different widths

\[
\begin{array}{c}
1 \ 0 \ 1 \ 1 \\
0 \ 1 \ 0 \ 0 \ 1 \ 0 \\
\hline
0 \ 1 \ 1 \ 1 \ 0 \ 1 \\
\end{array}
\]

\(-5 + 18 = 29\!\)
Sign Extension

• Rule #1: 2’s complement input and output operands must be the same word-width because of implied zeros

\[
\begin{align*}
0 & 0 1 0 1 1 & -5 \\
0 & 1 0 0 1 0 & +18 \\
\hline
0 & 1 1 1 0 1 & +29!
\end{align*}
\]

• Rule #2: Despite a fundamental change to the number’s definition, the value of a 2’s complement number will never change due to any amount of sign extension—positive or negative

\[
\begin{align*}
1 & 0 1 1 & -5 = -8 + 2 + 1 \\
1 & 1 0 1 1 & -5 = -16 + 8 + 2 + 1 \\
1 & 1 1 0 1 1 & -5 = -32 + 16 + 8 + 2 + 1
\end{align*}
\]
Sign Extension

- Procedure:
  1) Calculate the width of the answer word so that it contains all input possibilities
     - It’s up to you to make sure the output range is sufficient
  2) Extend the inputs’ sign bits to the width of the answer
  3) Add as usual
  4) Ignore bits that ripple to the left of the answer’s MSB

\[ \begin{array}{c} 111011 \quad -5 \\ 010010 \quad +18 \\ \hline \\ 011011 \quad +13 \end{array} \]

Sign Extension

- Ignore carry bits
  - Do not spend any hardware calculating any bits to the left of the answer’s MSB

\[ \begin{array}{c} 111011 \quad -5 \\ 110010 \quad +14 \\ \hline \\ 101101 \quad -19 \end{array} \]

ignore all bits to the left of the MSB