University of Illinois at Urbana-Champaign
Dept. of Electrical and Computer Engineering

ECE 120: Introduction to Computing

A Comparator for 2's Complement

Comparing 2's Complement Is Different from Unsigned

Let's design a comparator for **2's complement** numbers.

Is the function the same as with **unsigned** (like addition)?

For **unsigned**, **1001 > 0101**.

Is the same true with 2's complement?

No.

Should we just start over?

Start with the Sign Bits

Let's try a little harder first…

If we compare two non-negative numbers,
- the approach IS the same.
- Right?

Maybe we can just use some extra logic to handle the sign bits?

Consider All Possible Combinations of Sign Bits

Let's make a table based on the sign bits:

A_s	B_s	interpretation	solution
0	0	$A \geq 0$ AND $B \geq 0$	use unsigned comparator
0	1	$A \geq 0$ AND $B < 0$	$A > B$
1	0	$A < 0$ AND $B \geq 0$	$A < B$
1	1	$A < 0$ AND $B < 0$	unknown

Interpret 2's Complement as Unsigned

Remember our "simple" rule for translating **2's complement** bit patterns to decimal?

The pattern $\mathbf{A = a_{N-1}a_{N-2} \ldots a_1a_0}$

has value $\mathbf{V_A = -a_{N-1}2^{N-1} + a_{N-2}2^{N-2} + \ldots + a_0 2^0}$

Let **A** be negative ($\mathbf{a_{N-1} = 1}$).

Interpreted as **unsigned**, the same bits have value $\mathbf{V_A + 2^N}$.*

*The statement is true by definition of 2's complement, actually.

Negative Numbers Can be Compared Directly

What happens if we feed two negative 2's complement numbers into our unsigned comparator?

We compare $\mathbf{V_A + 2^N}$ with $\mathbf{V_B + 2^N}$.

And we get an answer: <, =, or >.

Let's say that we find $\mathbf{V_A + 2^N < V_B + 2^N}$.

In that case, $\mathbf{V_A < V_B}$, so **we have the right answer for 2's complement**.

The same result holds for other answers.

We Need Special Logic for the Sign Bits

Now we can complete our table:

A_s	B_s	interpretation	solution
0	0	A ≥ 0 AND B ≥ 0	use unsigned comparator
0	1	A ≥ 0 AND B < 0	A > B
1	0	A < 0 AND B ≥ 0	A < B
1	1	A < 0 AND B < 0	use unsigned comparator

Simply Flip the Wires on the Most Significant Bit

Can we just flip the wires on the sign bits?

For $A_s = 0$ and $B_s = 1$,
- we feed in $\mathbf{A_{N-1} = 1}$ and $\mathbf{B_{N-1} = 0}$, and
- the unsigned comparator produces **A > B**.

For $A_s = 1$ and $B_s = 0$,
- we feed in $\mathbf{A_{N-1} = 0}$ and $\mathbf{B_{N-1} = 1}$, and
- the unsigned comparator produces **A < B**.

What about when $A_s = B_s$?

Flipping the bits then has no effect!

Answers are also correct in those cases.

One Comparator with a Control Signal can Do Both

Can we use a single comparator to perform both kinds of comparisons?

Yes, if we
- add a control signal **S**
- to tell the comparator whether to do **unsigned** (**S=0**) or **2's complement** (**S=1**) comparison.

Simply **XOR'ing the most significant bits of A and B with S suffices**.
- This approach leverages flexibility in the problem to reduce the logic needed.
- Analyze the design to understand how it works.

