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

ECE 120: Introduction to Computing

Another Revision of 2's Complement Overflow

Long Definition for Overflow of 2's Complement Addition

Recall the overflow condition **V** for **2's complement** addition.

Add two **N-bit 2's complement** patterns.

$$
\begin{array}{rl}
A\ a_{N-2}\ \ldots\ a_0 & \text{(sign bit is A)} \\
+\ B\ b_{N-2}\ \ldots\ b_0 & \text{(sign bit is B)} \\
\hline
S\ s_{N-2}\ \ldots\ s_0 & \text{(sign bit is S)}
\end{array}
$$

We can calculate

$$V = ABS' + A'B'S$$

Another Way to Define 2's Complement Overflow?

Other lectures saw a different condition.
Let's first name two of the carry bits.

$$
\begin{array}{rl}
C_N\ C_{N-1}\quad\quad\quad & \\
A\ a_{N-2}\ \ldots\ a_0 & \text{(sign bit is A)} \\
+\ B\ b_{N-2}\ \ldots\ b_0 & \text{(sign bit is B)} \\
\hline
S\ s_{N-2}\ \ldots\ s_0 & \text{(sign bit is S)}
\end{array}
$$

The other lectures were then told that

$$V = C_N \oplus C_{N-1}$$

Are these two expressions the same?

One Proof Strategy: Algebra

We can use Boolean algebra to prove that

$V = ABS' + A'B'S$ equals $C_N \oplus C_{N-1}$

But it's not really so fun.

Trust me, I did it.

What about brute force? (a truth table)

We can calculate **S** and C_N from **A**, **B**, and C_{N-1}, so we only have 3 variables as "inputs."

Proof by Exhaustion / Brute Force

A	B	C_{N-1}	C_N	S	V	$C_N \oplus C_{N-1}$
0	0	0	0	0	0	0
0	0	1	0	1	1	1
0	1	0	0	1	0	0
0	1	1	1	0	0	0
1	0	0	0	1	0	0
1	0	1	1	0	0	0
1	1	0	1	0	1	1
1	1	1	1	1	0	0

Always Choose the Right Proof Strategy

Always choose the **clearest and fastest proof strategy** (usually those two metrics correlate).

Using brute force for proofs doesn't make you a brute!