(1)

a	b	c_{in}	Sum	c_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

C_{out}: MSOP: $\Sigma m(3, 5, 6, 7)$

ab \ c_{in}	0	1
00	0	0
01	0	1
11	1	1
10	0	1

$\Rightarrow ab + bc_{in} + ac_{in}$

Sum: MSOP: $\Sigma m(1, 2, 4, 7)$

ab \ c_{in}	0	1
00	0	1
01	1	0
11	0	1
10	1	0

$\Rightarrow \bar{a}b\bar{c_{in}} + a\bar{b}\bar{c_{in}} + \bar{a}\bar{b}c_{in} + abc_{in}$

(2) Sum using XOR

Use 2 XOR: $x = a \oplus b = \bar{a}b + a\bar{b}$

$Sum = x + c_{in} = a \oplus b \oplus c_{in}$

$Sum = \bar{a}\bar{b}c_{in} + \bar{a}b\bar{c_{in}} + abc_{in} + a\bar{b}\bar{c_{in}}$

$= c_{in}(\bar{a}\bar{b} + ab) + \bar{c_{in}}(\bar{a}b + a\bar{b})$

$= c_{in}\overline{(a \oplus b)} + \bar{c_{in}}(a \oplus b)$

$= a \oplus b \oplus c_{in}$

C_{out} implemented NAND - NAND

$$C_{out} = ab + bc_{in} + ac_{in}$$
$$= \overline{(\overline{ab})(\overline{bc_{in}})(\overline{ac_{in}})}$$

(3) for a full adder, $C_{out} = 1$ when at least 2 of 3 inputs (a, b, c_{in}) are 1

$\Rightarrow (a, b, c_{in}) = (0, 1, 1)$

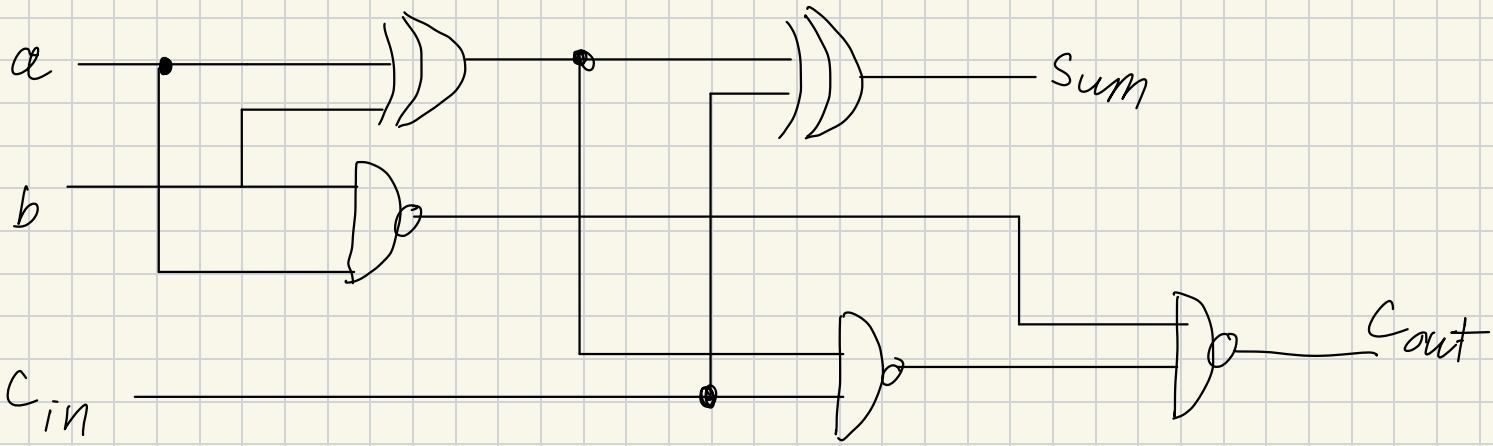
$(a, b, c_{in}) = (1, 0, 1)$

$(a, b, c_{in}) = (1, 1, 0)$

$(a, b, c_{in}) = (1, 1, 1)$

$$\Rightarrow C_{out} = ab + c_{in}(a \oplus b)$$
$$= ab + c_{in}(\bar{a}b + a\bar{b})$$
$$= \overline{(\overline{ab})(\overline{a \oplus b\, c_{in}})}$$

$$Sum = a \oplus b \oplus c_{in}$$



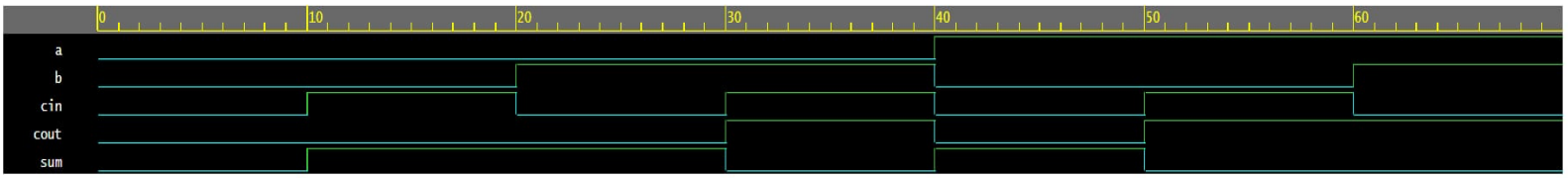
https://www.edaplayground.com/home

playground | New | Run | Save* | KnowHow WEBINARS | Productivity in AMD Vivado with SystemVerilog FREE • 1 Hour FEB 18, 2026 | REGISTER NOW | Resources | Community | Help | Playgrounds | Profile

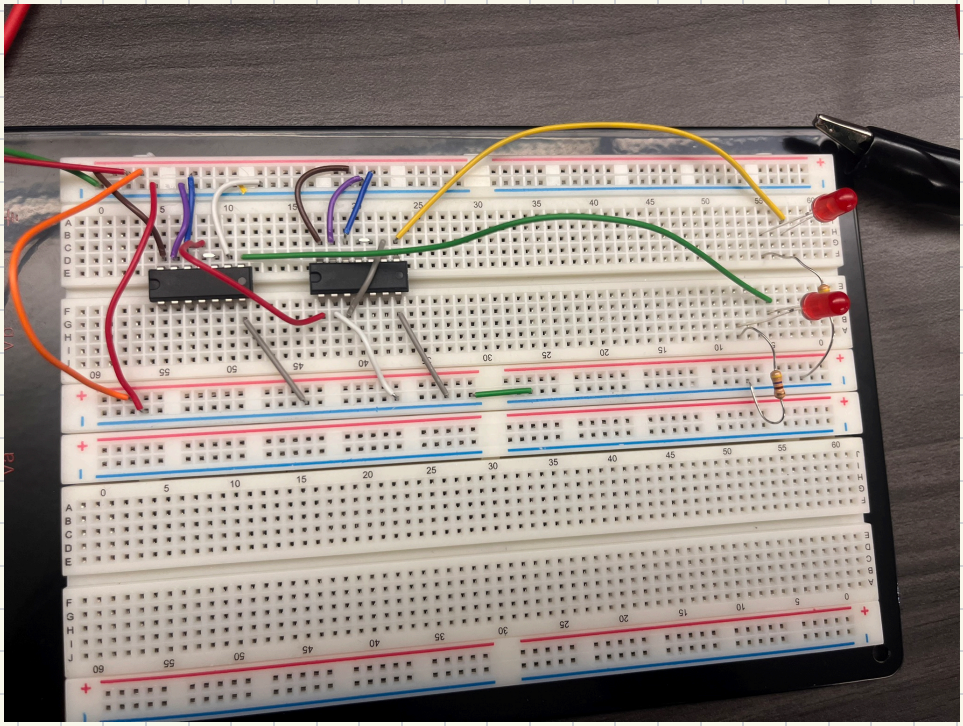
EPWave

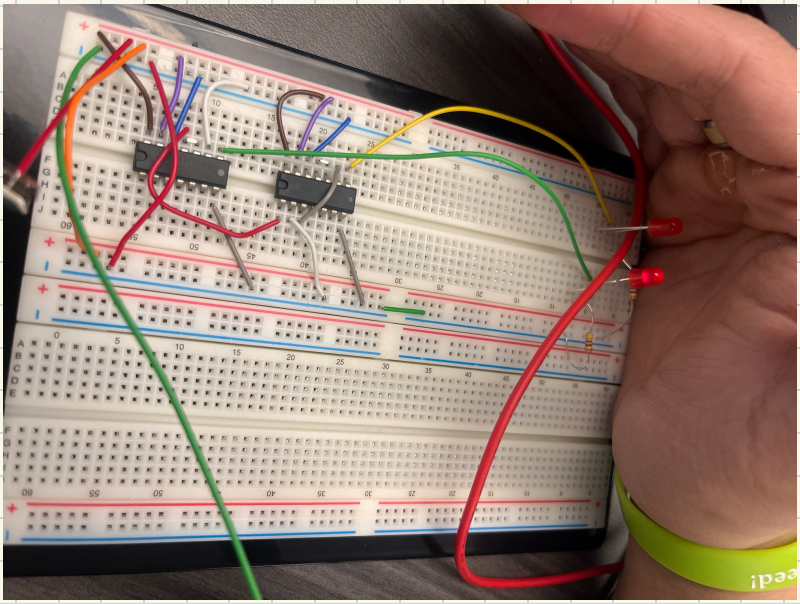
From: 0ns To: 70ns

Get Signals | Radix | 100%



Note: To revert to EPWave opening in a new browser window, set that option on your profile page.





a = 1, b = 0, cin = 0

a = 1, b = 0, cin = 1