

EP01 - Arquitetura de Computadores ||

½ somador logisim

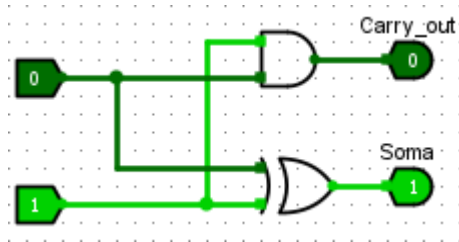


Tabela verdade ½ somador

a	b	Carry_out	Soma
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Somador completo

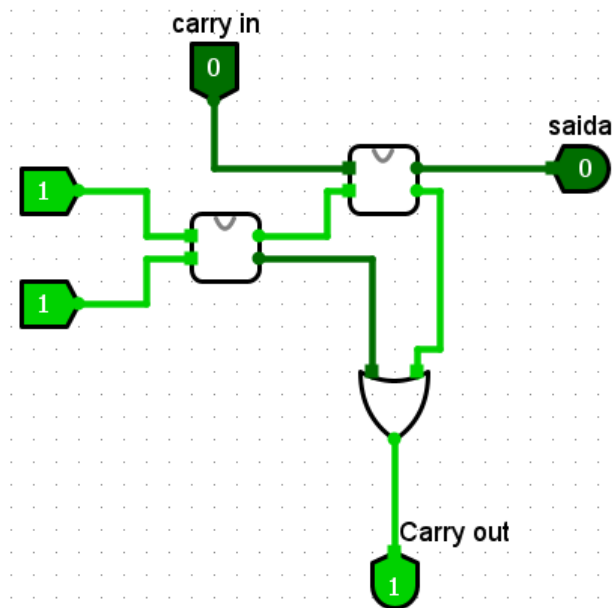
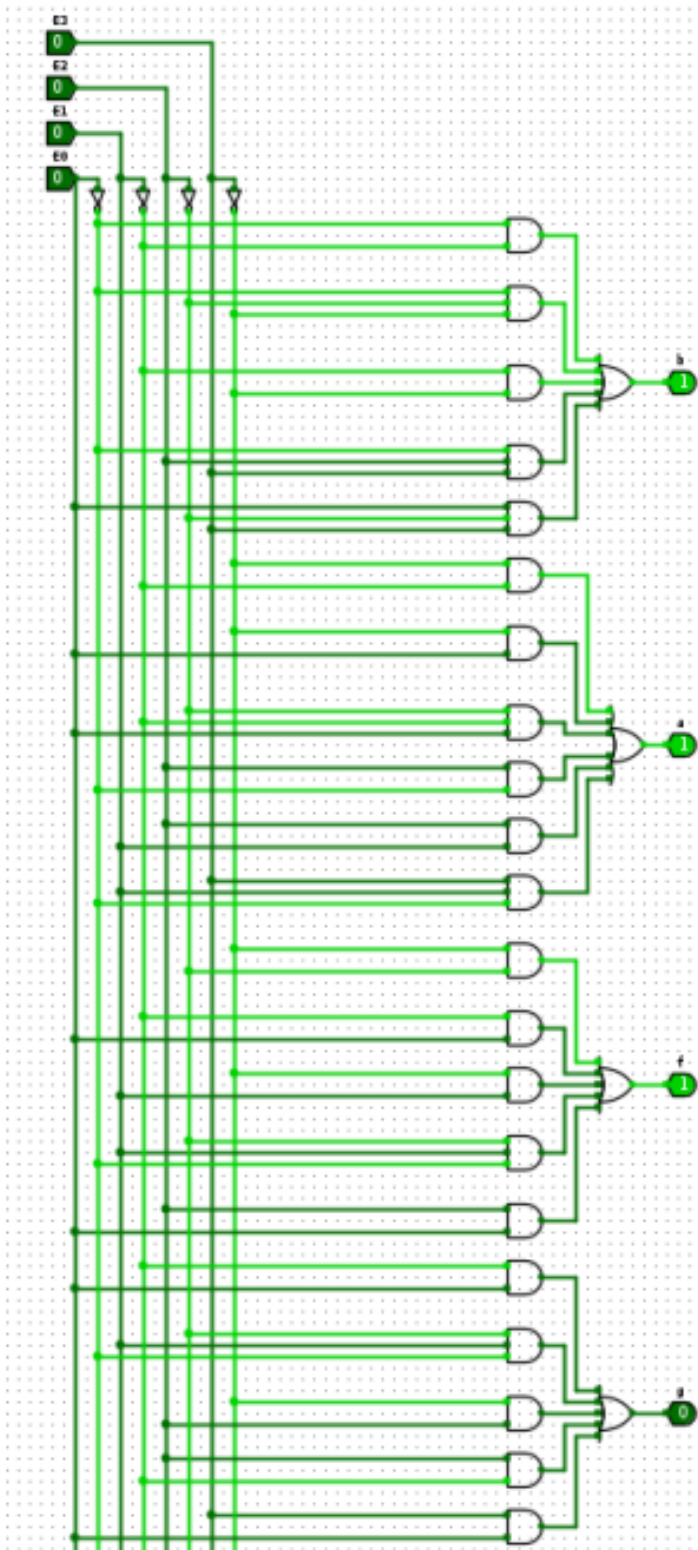


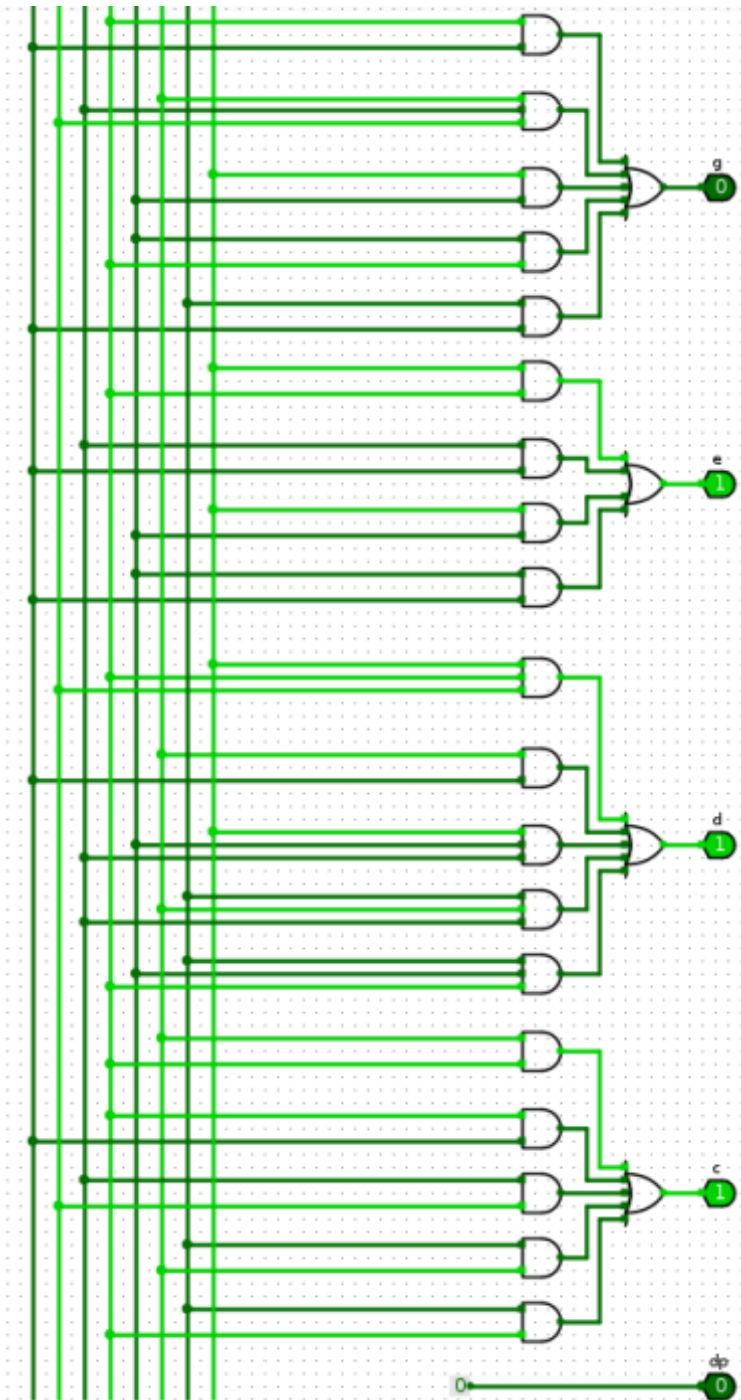
Tabela verdade somador completo

carry in	a	b	saida	Carry out
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	0	1
1	0	0	0	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	0

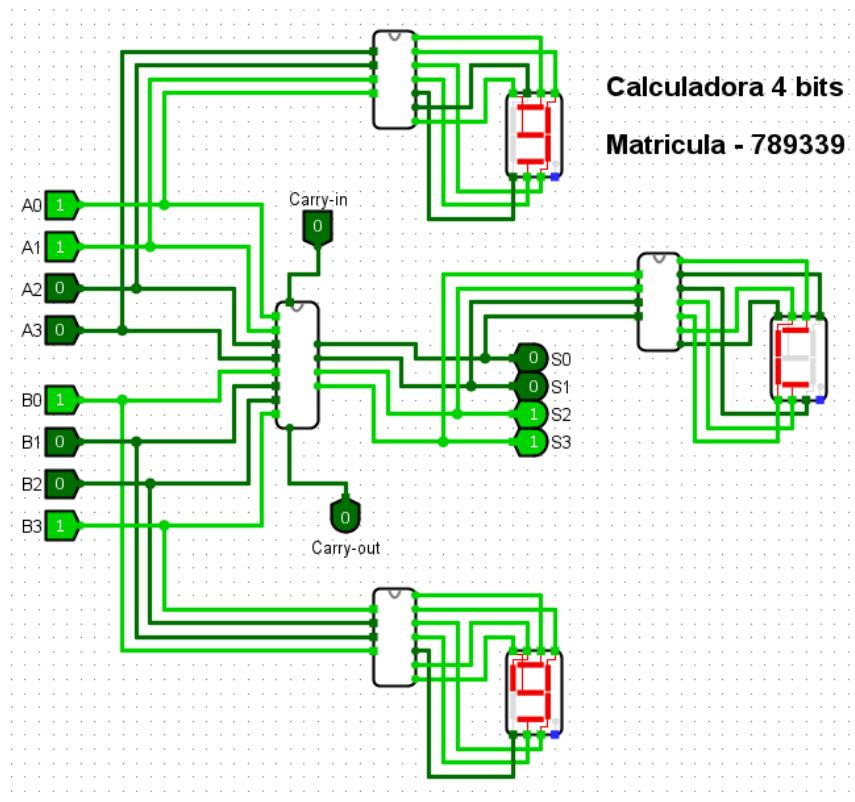
Decodificador hexadecimal



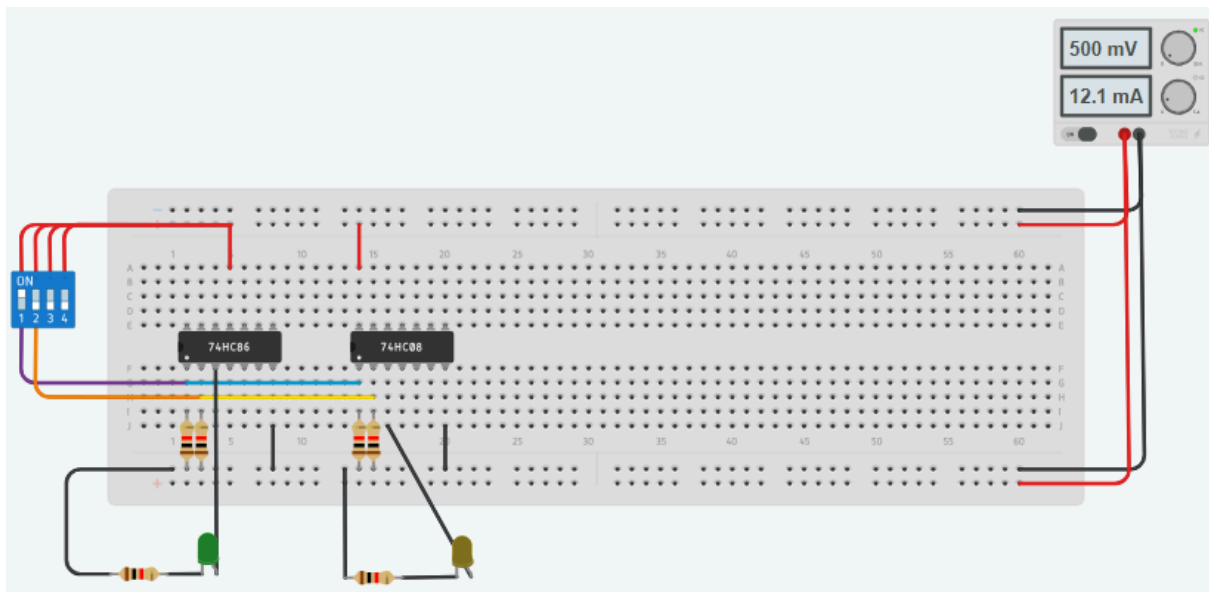
Decodificador hexadecimal



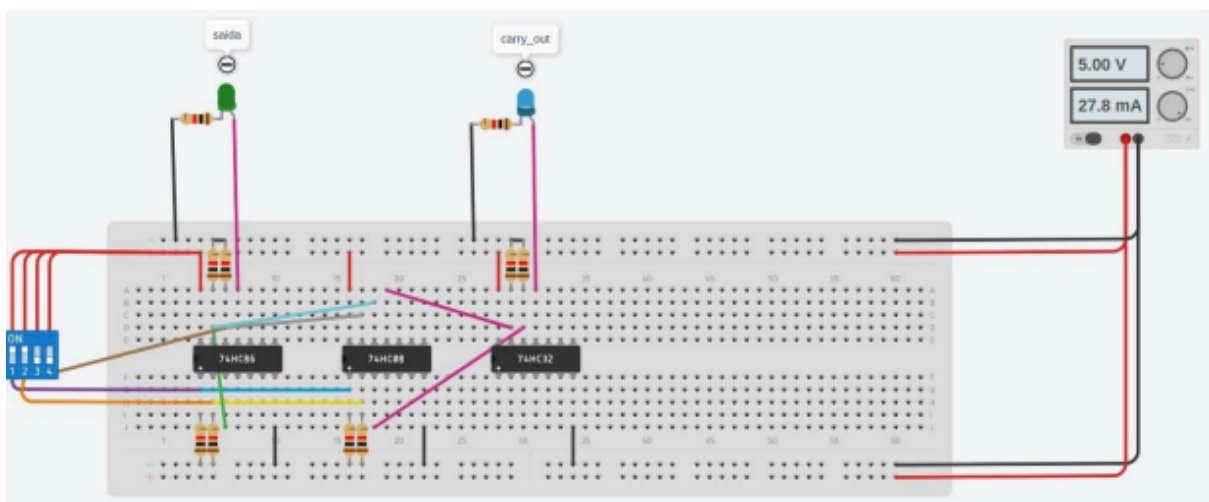
Calculadora de 4 bits Logisin



½ somador Tinkercad



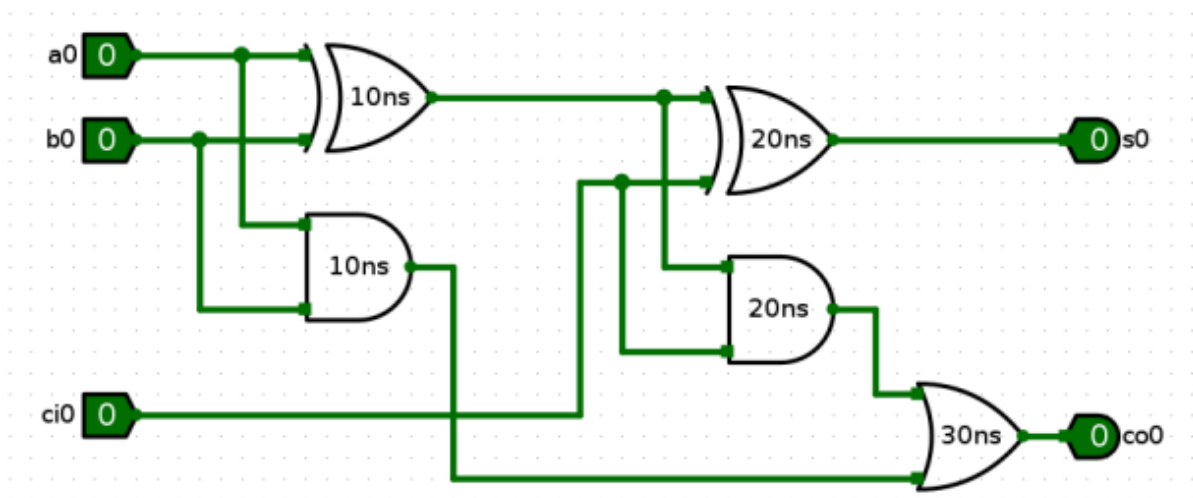
Somador completo Tinkercad



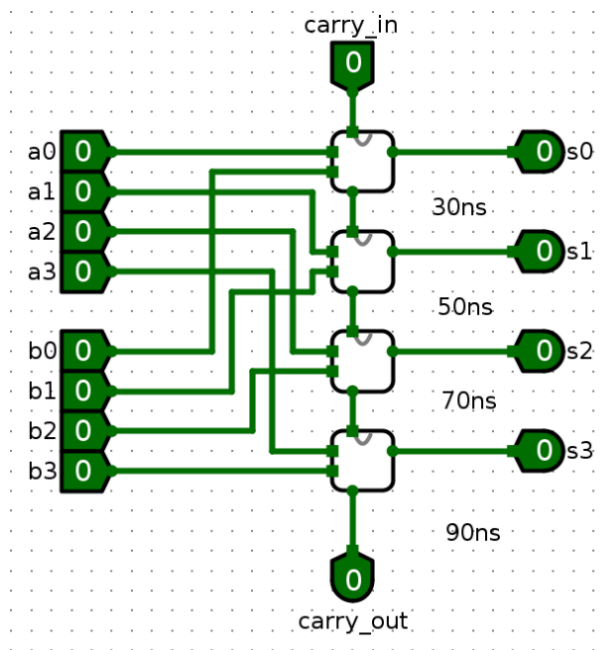
1- Cada uma das portas lógicas possui uma tabela verdadeira já pré definida para cada entrada. Porém, se um dos terminais de uma porta lógica não estiver ligado a um valor válido não sendo nem 0 nem 1, dependendo ela poderá ocorrer um erro na saída, tornando imprevisível. Isso acontece porque a porta lógica é sensível a pequenas interferências elétricas, e, portanto, estas podem influenciar e levar a resultados inesperados.

2- Para somar os 4 bits, inicialmente somente alguns valores já são conhecidos: a_0 , a_1 , a_2 , a_3 , b_0 , b_1 , b_2 , b_3 e ci_0 . Estas são as 2 palavras de 4 bits e o carry in inicial. Para passar no primeiro somador dos primeiros bits, o tempo necessário é de 30ns. Esse mesmo tempo poderia ser observado em cada um dos outros 3 somadores completos. No entanto, para os seguintes, é necessário que o carry out da porta anterior envie o sinal para o carry in da atual. Em outras palavras: o $cin+1$ depende do con . Portanto, o somador não leva 30ns para somar tudo, pois é necessário o somador anterior finalizar para completar o próximo ligado a ele.

3-



No somador para 4 bits, o processo é semelhante. Todavia, nota-se que o co_0 só esteve concluído após 30ns. Então, o sinal do co_1 só estará disponível após este tempo.



Logo, para o somador de 4 bits acima, seriam necessários 90ns para que a soma fosse efetuada por completo

4- Seguindo o padrão do somador de 4 bits, seria necessário, para o de 32 bits, 32 somadores completos. Em outras palavras:

- **32 * 2 * portas XOR = 64 XOR**

- **32 * 2 * portas AND = 64 AND**

- **32 * 1 * portas OR = 32 OR**

Além, claro, das 32 entradas e saídas, bem como o carry out e o overflow caso necessário. A operação toda de soma, neste caso, ocorreria em 650ns: 1 * 30ns para a primeira parte com co0 + 31 * 20ns para os con .

5- $F = 1 / T$ $F = 1 / 650\text{ns}$ $F = 1 / 650 * 10^{-9} \text{ s}$ $F = 10^9 / 650 \text{ Hz}$ $F \approx 1.54 * 10^6 \text{ Hz}$
 $F \approx 1.54 \text{ MHz}$, **1.54 MHz**

6- A partir dos itens anteriores, nota-se uma significativa ineficácia deste somador completo para n bits. Por se tratar de operações lógicas com portas lógicas, uma possível solução para minimizar essa perda de tempo seria simplificar o circuito, de tal sorte que não seja necessário todo o tempo para se obter o resultado esperado, utilizando técnicas de simplificação da tabela verdade e da sua equação.