

Exercício Prático 01

Disciplina: Arquitetura de Computadores II

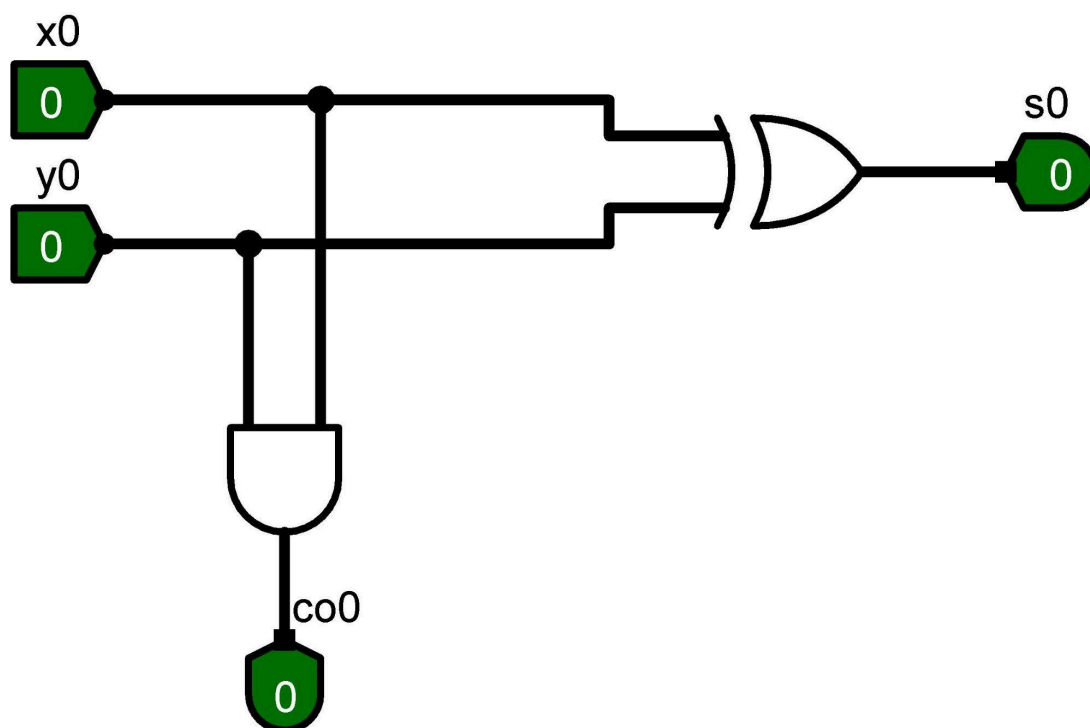
Data: 15/02/2025

Aluna: Alessandra Faria Rodrigues

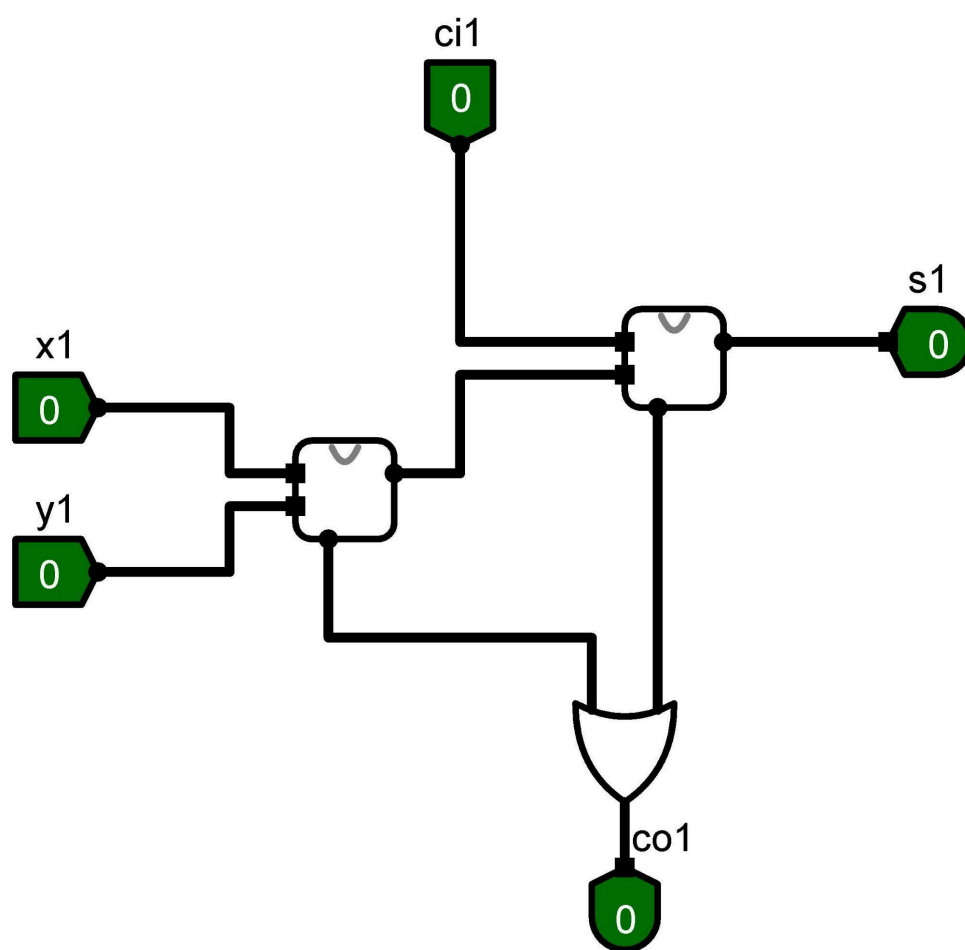
Matrícula: 828333

- Logisim

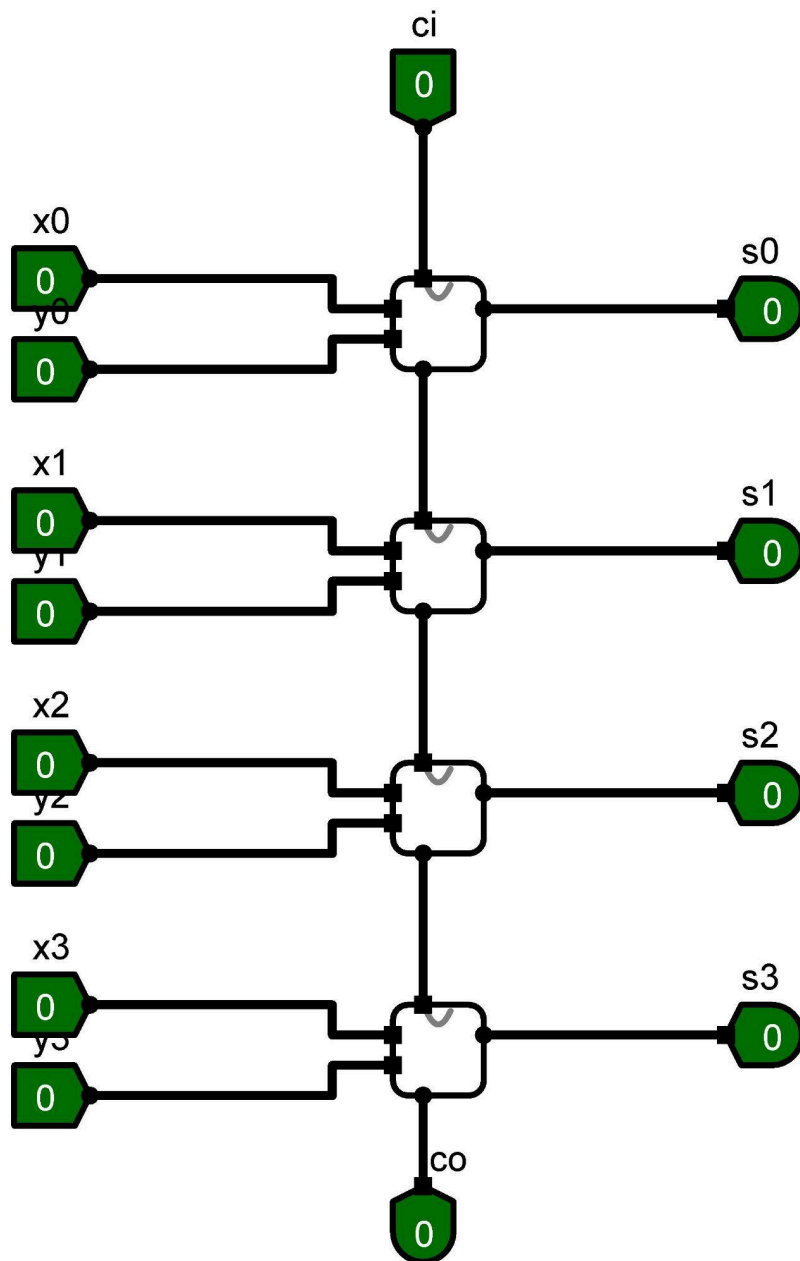
- Circuito meia soma



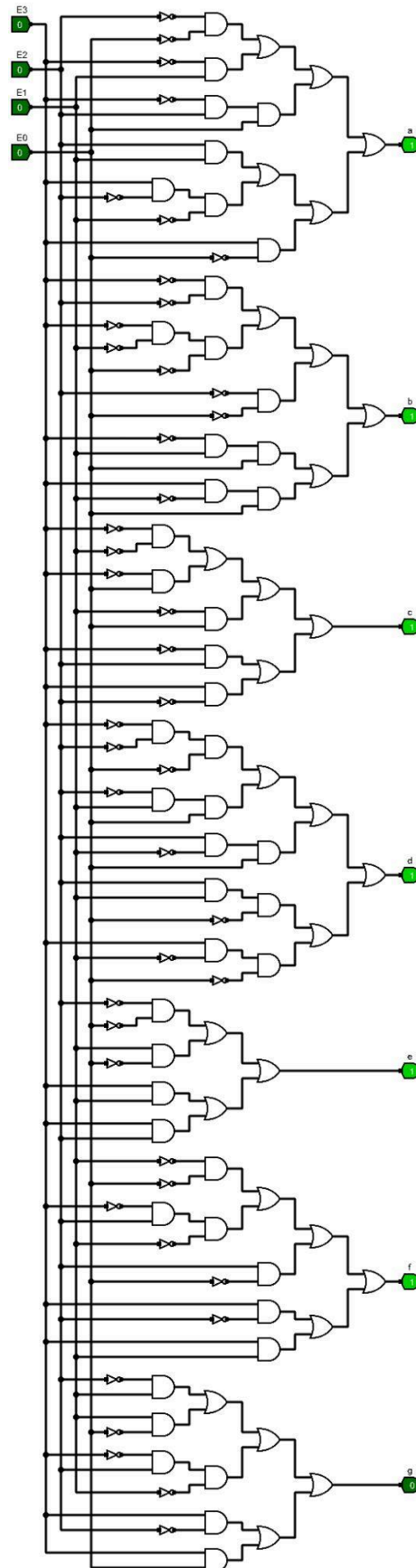
- Circuito soma completa



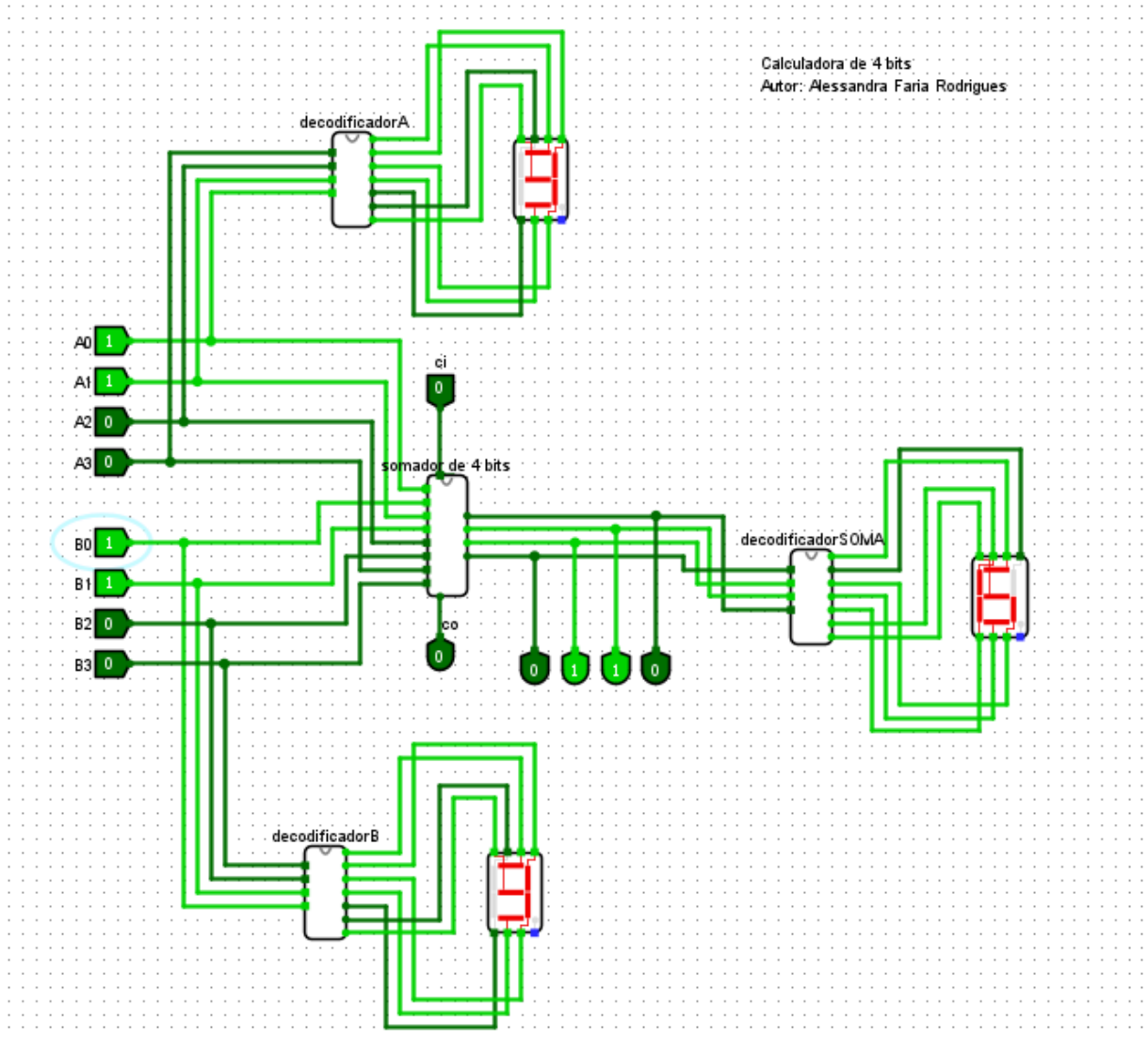
- Circuito somador de 4 bits



- Circuito decodificador

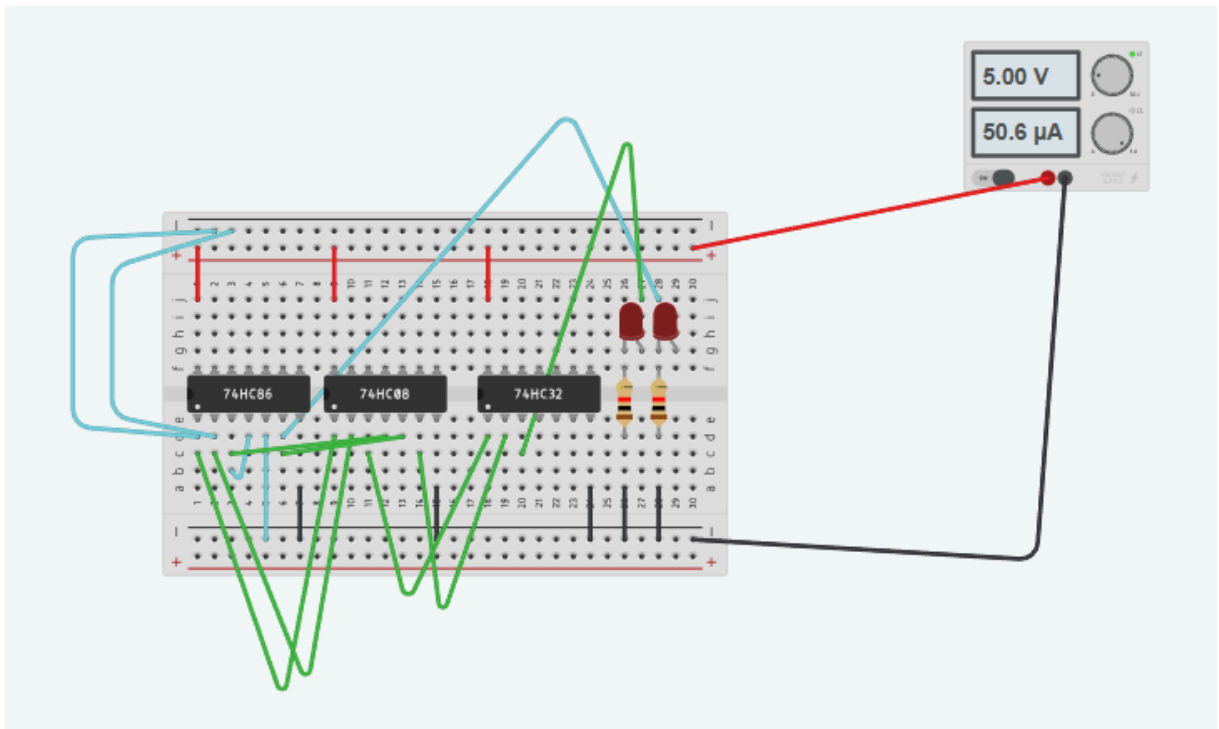


- Circuito calculadora de 4 bits

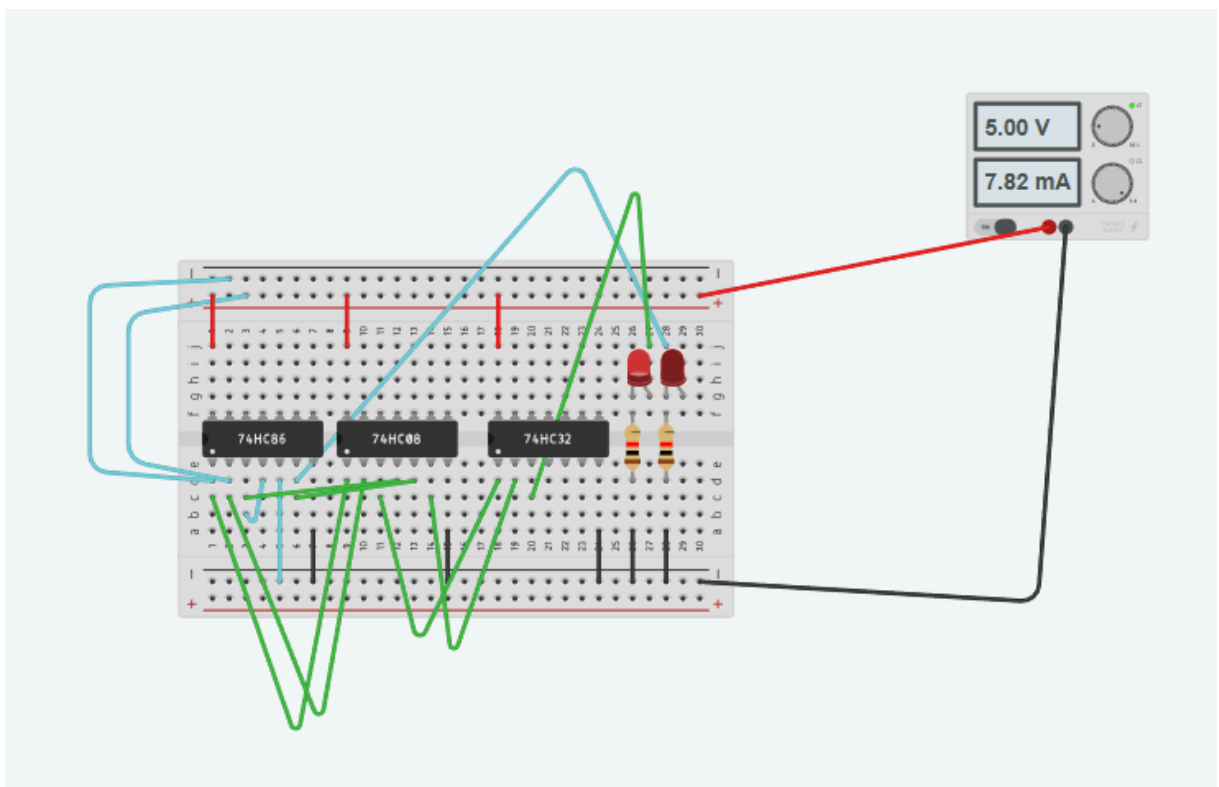


- Tinkercad

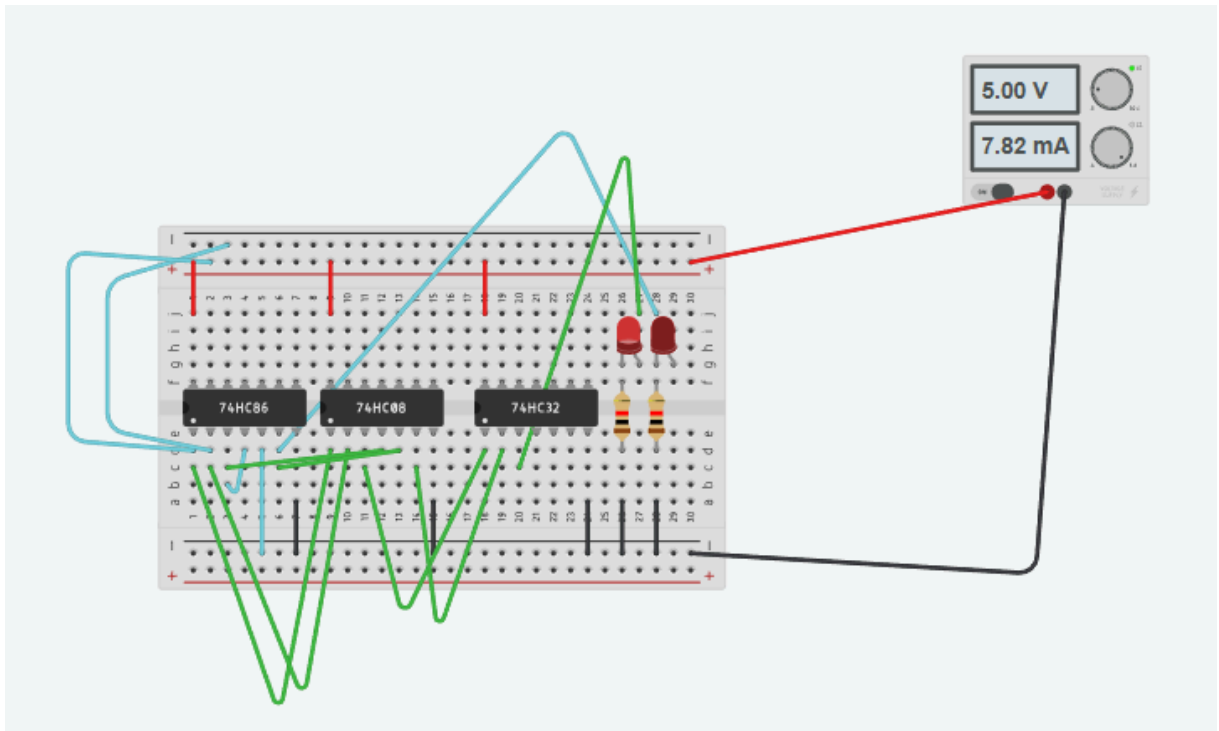
0 + 0



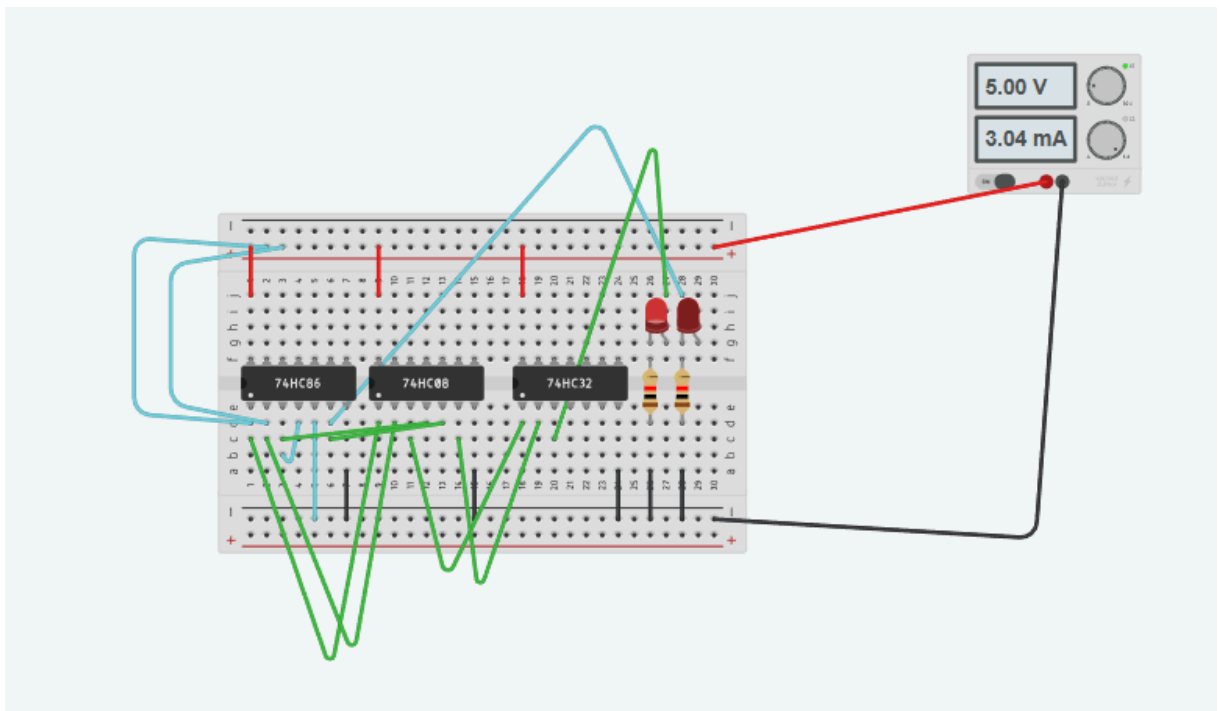
0 + 1



1 + 0



1 + 1



- Perguntas:

Pergunta 1: O que acontece se um dos terminais de entrada de uma porta lógica não estiver conectado em 0 ou 1 (eletricamente ele deverá estar flutuando, ou seja não conectado a nenhum nível lógico)

Resposta: Se uma entrada estiver flutuando, o comportamento da porta lógica será indefinido. Em circuitos digitais, os níveis lógicos precisam ser bem definidos (0 ou 1). Um terminal flutuante pode resultar em estados imprevisíveis devido a ruídos ou interferências eletromagnéticas, podendo levar a comportamentos inadequados no circuito.

Pergunta 2: Qual o problema de tempo associado a esse tipo de somador (pense no carry), considere o atraso médio de cada porta lógica de 10 ns.

Resposta: O problema de tempo está relacionado à propagação do *carry*. Em um somador onde cada bit espera o *carry* do bit anterior, o atraso total da operação é proporcional ao número de bits do somador. Como o *carry* precisa se propagar através de todas as portas lógicas, o tempo de execução aumenta consideravelmente, já que um sinal deve “esperar” o outro para se obter um resultado.

Pergunta 3: Qual o tempo necessário para a computação de uma soma e do vai um em um somador de 4 bits.

Resposta: Considerando que cada porta lógica tem um atraso médio de 10 ns, o tempo gasto pelo primeiro somador é de 30 ns.

Sinal1 -> (1 xor, 1 and) = 10 ns

Sinal2 -> (1 xor, 1 and) = 10 ns

Sinal1 or Sinal2 = +10ns

Total = 30 ns

Para um somador de 4 bits é preciso considerar o atraso do Carry Out/Carry in. O Carry in de um somador levará 20 ns para receber o sinal. Sendo assim, um somador de 4 bits leva no total 90 ns. O cálculo considera 30 ns para o primeiro somador e um atraso de 20 ns para o restante dos somadores.

Total para somador de 4 bits: $30+20+20+20 = 90$ ns

Fórmula geral: $30 + 20 \cdot (\text{quantidade de somadores} - 1)$

Pergunta 4: O que seria necessário para um somador de 32 bits ?

Resposta: Para um somador de 32 bits seria necessário 30 ns para o primeiro somador mais 20 ns multiplicado pelos somadores restantes, considerando o desenvolvimento do exercício anterior.

$$\text{Total} = 30 + 20 \cdot 31 = 650 \text{ ns}$$

Pergunta 5: Considerando esses tempos acima, calcule a frequência de operação de um somador de 32 bits.

A frequência de operação é o inverso do tempo necessário para uma soma.

Temos então: $F = 1/T$, T em ns.

$$F = 1 / (650 \cdot 10^{-9})$$

$$F = 1 / (6,5 \cdot 10^{-7})$$

$$F \approx 1,538 \text{ MHz}$$

Pergunta 6: Você consegue propor alguma forma de tornar essa soma mais veloz?

Resposta: Uma maneira de tornar a soma mais rápida é utilizar um somador de carry-lookahead (CLA). Um somador de carry-lookahead é projetado para melhorar a velocidade dos somadores tradicionais. Ele faz isso através da redução do tempo de propagação do carry, utilizando uma lógica combinacional que prevê os valores de carry sem a necessidade de esperar que cada estágio anterior seja computado.