

Experimento 4

Circuitos Combinacionais: Comparador de Palavras

Ana Caroline da Rocha Braz, 21/2008482

Kalew Silva Piveta, 18/0104071

Grupo G20

¹Dep. Ciência da Computação – Universidade de Brasília (UnB)
CIC0231 - Laboratório de Circuitos Lógicos

braz.ana@aluno.unb.br, kalew.silva@aluno.unb.br

Abstract. *This corresponds to the report of Experiment 4 on Combinational Circuits: Word Comparator. The purpose of this is to explain our resolutions and explanations about the exercises proposed in the Experiment 4 script. For this, circuits were created to measure the propagation delay and circuits with comparators using NAND gates, to compare 3-bit and 2-bit words.*

Resumo. *Este corresponde ao relatório do Experimento 4 sobre Circuitos Combinacionais: Comparador de Palavras. O objetivo deste é explicar nossas resoluções e explicações sobre os exercícios propostos no roteiro do Experimento 4. Para isso, foram criados circuitos para medir o atraso de propagação e circuitos com comparadores utilizando portas NAND, para comparar palavras de 3 e 2 bits.*

1. Introdução

Durante o semestre 2021/2, diversos experimentos da disciplina de Laboratório de Circuitos Lógicos (CIC0231) serão realizados com o intuito de apresentar os recursos e ferramentas básicas para a construção e testes de circuitos digitais, além de introduzir conceitos e métodos utilizados para circuitos digitais modernos [Lamar and Mandelli 2022].

O atraso de uma porta lógica é o intervalo de tempo entre a aplicação de um pulso na entrada e a ocorrência de um pulso resultante na saída. Existem duas maneiras de medir esse tempo: A primeira forma é a medição da saída mudando de nível alto para nível baixo. E a segunda, é a medição da saída mudando de nível baixo para nível alto [Floyd 2009, Harris and Harris 2017]. No decorrer dos experimentos será mais discutido sobre os atrasos de propagação das portas lógicas.

Já os comparadores possuem a função básica de comparar as magnitudes de dois números binários para determinar a relação comparativa entre eles. De maneira geral, um circuito comparador determina se dois números ou palavras são iguais [Floyd 2009]. Para esse experimento, os comparadores serão utilizados nos itens 2.2 e 2.3 para comparar palavras de 3 bits e 2 bits, respectivamente.

Visto isso o objetivo principal desse experimento é relatar as resoluções dos itens mencionados, discutir sobre os atrasos de propagação das portas lógicas e projetar circuitos comparadores de 3 e 2 bits, usando as técnicas de síntese de circuitos combinacionais. Para isso, foram utilizadas as portas lógicas e a ferramenta *Timing Diagram Simulation*, para medir os tempos de atraso, do *software* Deeds.

Esse relatório está organizado da seguinte forma. Na seção 2, serão apresentados os procedimentos de cada item da parte experimental. Na seção 3, serão apresentados as análises dos resultados encontrados durante os procedimentos e, logo em seguida, os links para assistir o funcionamento dos circuitos. Na seção 4, a conclusão. E, por fim, as referências e a auto-avaliação.

2. Procedimentos

Nessa seção serão apresentados os resultados dos experimentos.

2.1. Atrasos de propagação

Nesse item é pedido para que fosse implementado no Deeds o seguinte circuito da Figura 1:

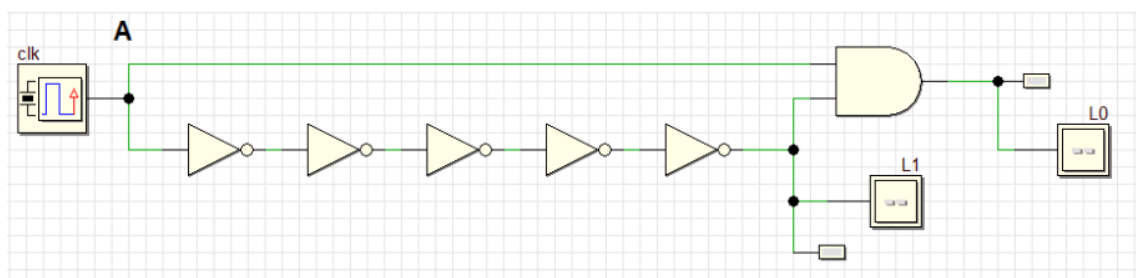


Figura 1. Circuito para verificação de atraso de propagação.

Após a implementação, o item a pede para que fosse escrito as expressões lógicas de cada saída em função da entrada A. Sendo assim, para a saída L1 temos \bar{A} e para a saída L0 temos $\bar{A}.A$. Em seguida, foi colocada a frequência do *Clock Generator* em 2Hz e iniciada a simulação. O vídeo pode ser encontrada na Seção 2.4 deste relatório.

Utilizando o *Timing Diagram Simulation*, foram obtidas as formas de onda das saídas L0 e L1, que podem ser vistas na Figura 2. No diagrama, é possível verificar que as portas possuem atrasos em suas propagações, ou seja, assim que o clock muda o seu nível lógico ainda tem um tempo até as outras portas mudarem também. Por conta da quantidade de portas NOT o atraso tende a ser maior e durante esse tempo o valor mantém em nível alto e quando passado à porta AND acontece um pulso rápido, pois ambas portas estão em níveis altos. Em relação ao tempo de atraso de propagação:

- Clock e a saída L1 é de 21n 993pS
- Clock e a saída L0 é de 26n 009pS
- Saída L1 e a Saída L0 é de 4n 000pS.

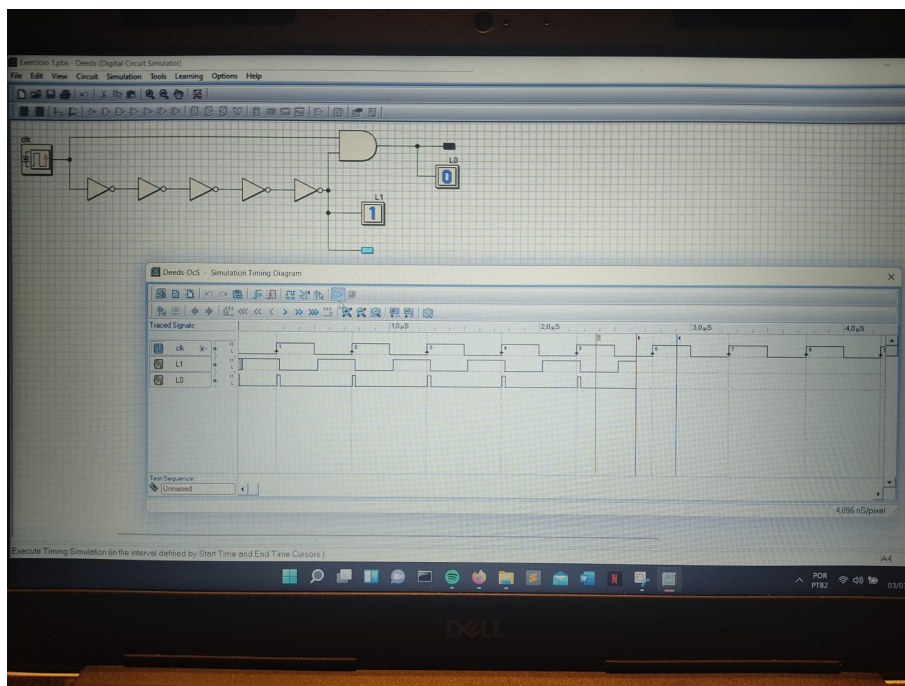


Figura 2. Diagrama de saída das saídas L0 e L1.

Posto isso, os pulsos no L0 apenas acontecem quando há a mudança do nível baixo para nível alto do clock. Isso acontece porque o nível lógico da porta L1 ainda se mantém baixo.

Não aparece pulsos em L0 no retorno de A para o nível baixo, pois a porta AND define a saída como nível alto quando suas duas entradas tem nível alto. Quando há apenas alguma das entradas com nível baixo, sua saída é automaticamente de nível baixo, ou seja, apesar do atraso fornecido pela sequência de portas NOT, elas não influenciam em um possível pulso, pois sua outra entrada já está em nível baixo.

Um número PAR de portas NOT, geram apenas um efeito de atraso na entrada, pois sua saída final é a mesma da entrada. Um efeito de pulso acontece quando, por causa do atraso, há uma divergência momentânea na saída final. Com um número PAR de portas NOT, não há essa divergência, pois as saídas são iguais a entrada, apesar do tempo de atraso, logo, não há pulso.

2.2. Comparador de 3 bits

Nesse experimento é pedido para que fosse projetado e simulado um comparador de palavras de 3 bits utilizando apenas portas NAND de duas entradas.

Primeiramente, foi construído a tabela verdade do circuito XNOR (Tabela 1) e obtido como expressão booleana $Z = \overline{A \oplus B}$.

A partir dessa tabela verdade, pode-se chegar a uma implementação apenas com portas NAND de duas entrada representada na Figura 3. Com essa implementação tornou-se possível criar um comparador apenas com portas NAND. Sendo assim, foi construído um comparador de 3 bits (Figura 4), o qual sempre que $A = B$ a saída será igual a 1. Com isso obtemos uma tabela verdade de 64 linhas, que nesse relatório foi dividido em 32 linhas (Tabela 2 e Tabela 3).

Tabela 1. Tabela da verdade do circuito XNOR.

ENTRADAS		SAÍDA
A	B	Z
0	0	1
0	1	0
1	0	0
1	1	1

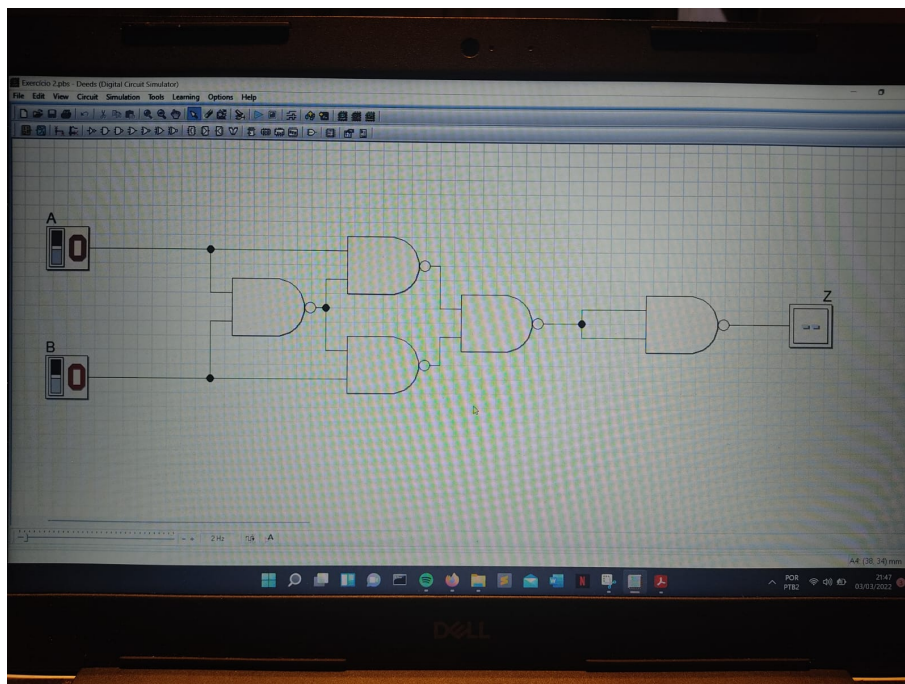


Figura 3. Circuito XNOR apenas com portas NAND

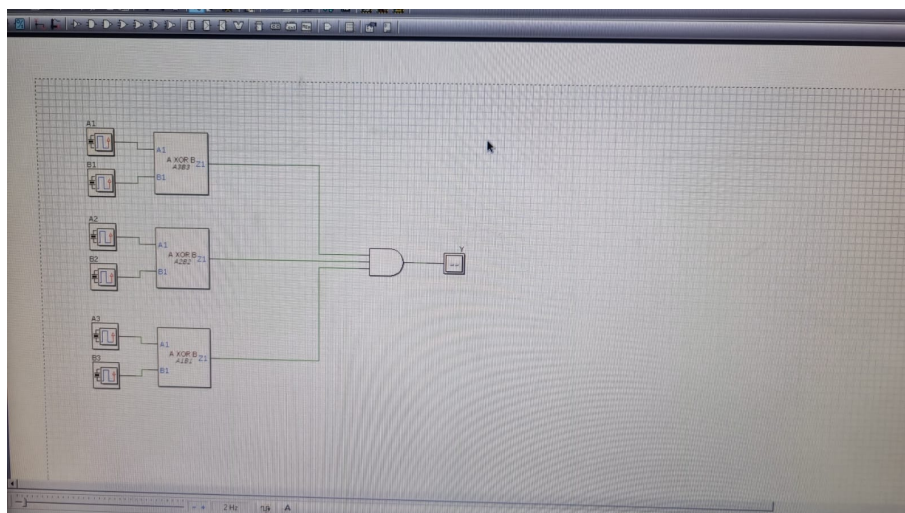


Figura 4. Circuito XNOR apenas com portas NAND

Tabela 2. Tabela verdade do circuito comparador de 3 bits: Primeira parte

ENTRADA						SAÍDA
A2	A1	A0	B2	B1	B0	A = B
0	0	0	0	0	0	1
0	0	0	0	0	1	0
0	0	0	0	1	0	0
0	0	0	0	1	1	0
0	0	0	1	0	0	0
0	0	0	1	0	1	0
0	0	0	1	1	0	0
0	0	0	1	1	1	0
0	0	1	0	0	0	0
0	0	1	0	0	1	1
0	0	1	0	1	0	0
0	0	1	0	1	1	0
0	0	1	1	0	0	0
0	0	1	1	0	1	0
0	0	1	1	1	0	0
0	0	1	1	1	1	0
0	1	0	0	0	0	0
0	1	0	0	0	1	0
0	1	0	0	1	0	1
0	1	0	0	1	1	0
0	1	0	1	0	0	0
0	1	0	1	0	1	0
0	1	0	1	1	0	0
0	1	0	1	1	1	0
0	1	1	0	0	0	0
0	1	1	0	0	1	0
0	1	1	0	1	0	0
0	1	1	0	1	1	1
0	1	1	1	0	0	0
0	1	1	1	0	1	0
0	1	1	1	1	0	0
0	1	1	1	1	1	0

Tabela 3. Tabela verdade do circuito comparador de 3 bits: Segunda parte

ENTRADA						SAÍDA
A2	A1	A0	B2	B1	B0	A = B
1	0	0	0	0	0	0
1	0	0	0	0	1	0
1	0	0	0	1	0	0
1	0	0	0	1	1	0
1	0	0	1	0	0	1
1	0	0	1	0	1	0
1	0	0	1	1	0	0
1	0	0	1	1	1	0
1	0	1	0	0	0	0
1	0	1	0	0	1	0
1	0	1	0	1	0	0
1	0	1	0	1	1	0
1	0	1	1	0	0	0
1	0	1	1	0	1	1
1	0	1	1	1	0	0
1	0	1	1	1	1	0
1	1	0	0	0	0	0
1	1	0	0	0	1	0
1	1	0	0	1	0	0
1	1	0	0	1	1	0
1	1	0	1	0	0	0
1	1	0	1	0	1	0
1	1	0	1	1	0	1
1	1	1	0	0	0	0
1	1	1	0	0	1	0
1	1	1	0	1	0	0
1	1	1	0	1	1	0
1	1	1	1	0	0	0
1	1	1	1	0	1	0
1	1	1	1	1	0	0
1	1	1	1	1	1	1

Ao realizar a simulação em forma de ondas do circuito comparador de 3 bits foi visto que, cada bloco comparador de 1 bit faz 6 vezes mais transições que o seu anterior. Isso gera um tempo de atraso maior do que o convencional, porém, ao analisar as transições que geram uma saída de nível alto, é possível averiguar, com uma certa facilidade, a coerência dos resultados.

2.3. Comparador de 2 bits

Nesse experimento pede-se para projetar e implementar um comparador de duas palavras de 2 bits, com 3 saídas, tal que $Y1 = 1$ se $A > B$, $Y2 = 1$ se $A = B$ e $Y3 = 1$ se $A < B$.

Para que pudesse ser implementado esse comparador, primeiro foi feito um comparador de 1 bit, que pode ser visto na Figura 5, tendo como expressão minimizada:

- Para $A > B$: $A \cdot \bar{B}$
- Para $A < B$: $\bar{A} \cdot B$
- Para $A = B$: $\overline{A \oplus B}$

Nesse comparador, portas XOR, NOT e AND foram utilizadas e com isso foi possível obter a Tabela 4.

Tabela 4. Tabela verdade do circuito comparador de 1 bit

ENTRADA		SAÍDA		
A	B	$A > B$	$A = B$	$A < B$
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

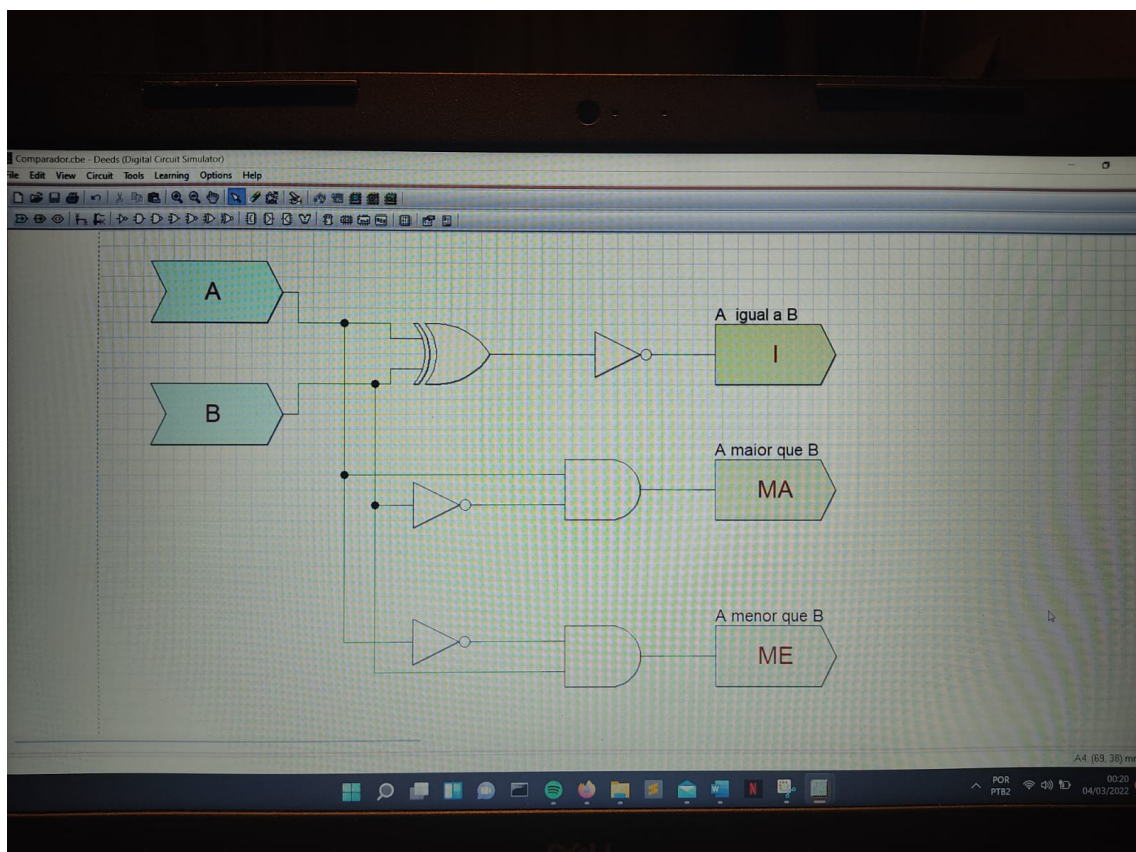


Figura 5. Comparador de 1 bit.

Já para a construção do comparador de 2 bits, foram utilizadas 2 comparadores de 1 bit, implementados anteriormente, portas AND e OR. Esse circuito pode ser visto na Figura 6 e sua tabela verdade pode ser vista logo abaixo (Tabela 5):

Tabela 5. Tabela verdade do circuito comparador de 2 bits

ENTRADA				SAÍDA		
A1	A0	B1	B0	A > B	A = B	A < B
0	0	0	0	0	1	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	1	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	1	0

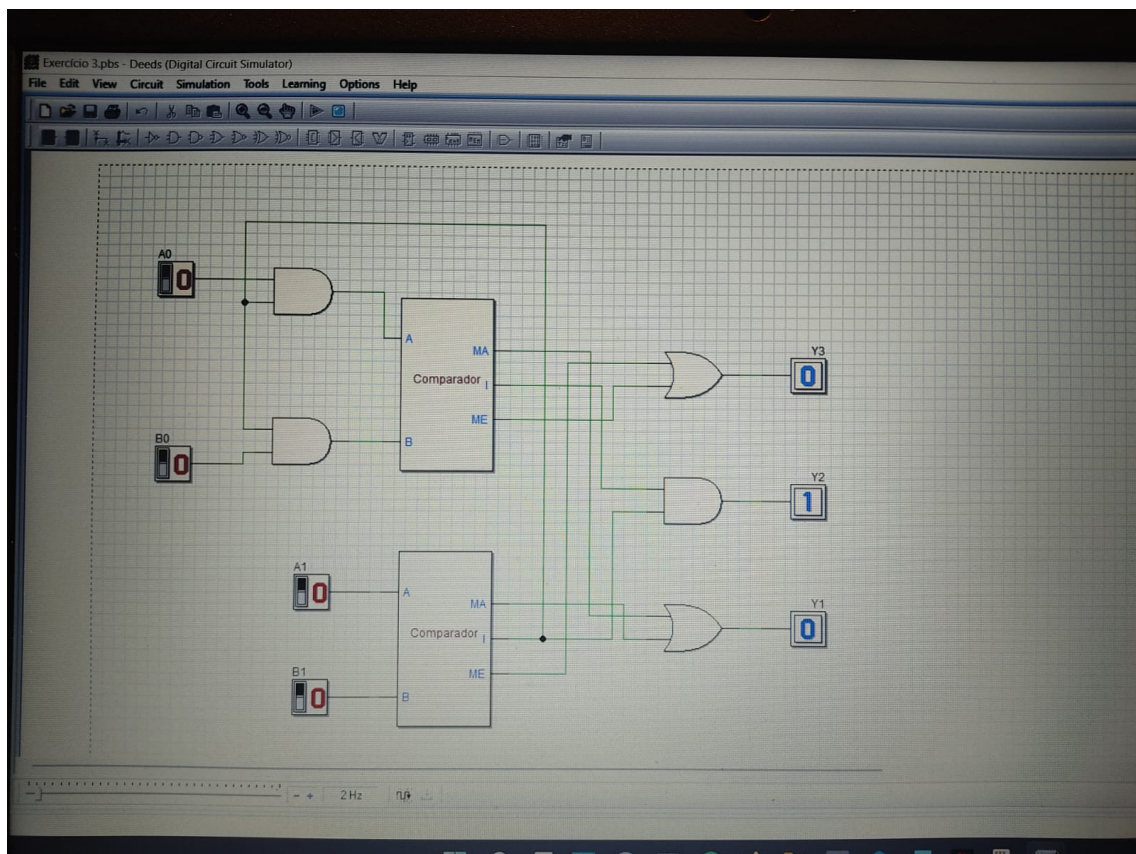


Figura 6. Comparador de 2 bits.

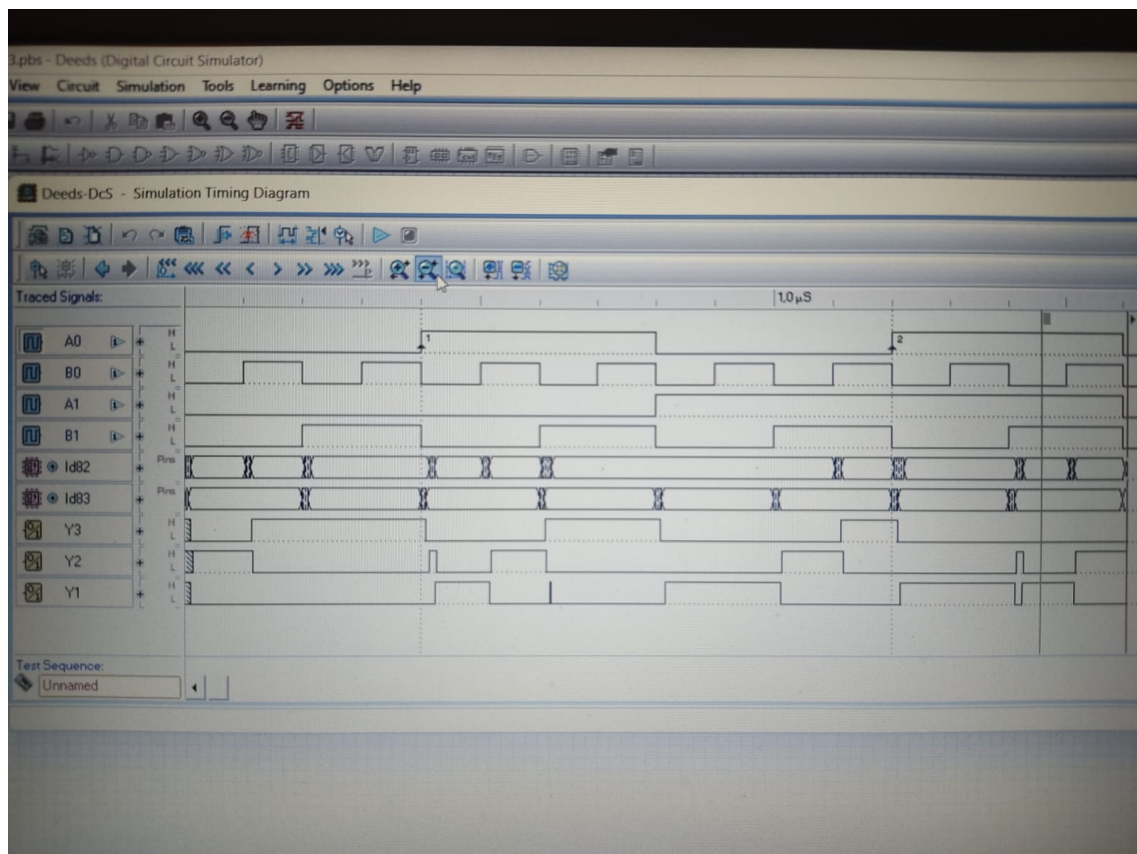


Figura 7. Diagrama de tempo de atraso do comparador de 2 bits.

Ao analisar o diagrama de tempo de atraso do comparador de 2 bits, é possível visualizar que algumas saídas do circuito influenciam outras entradas do mesmo, impondo assim um tempo de atraso dependente de outros processos. Entende-se como um atraso sequencial entre as saídas, pois uma depende da outra, de certa forma, pelo modo que o circuito foi construído.

2.4. Links para vídeo no Youtube

Para assistir o funcionamento dos circuitos desse relatório, basta clicar nos seguintes links:

- Link para assistir o funcionamento da Figura 1: [Circuito para verificação de atraso de propagação.](#)
- Link para assistir o funcionamento da Figura 3: [Circuito XNOR apenas com portas NAND](#)
- Link para assistir o funcionamento da Figura 4: [Comparador de 3 bits.](#)
- Link para assistir o funcionamento da Figura 5: [Comparador de 1 bit.](#)
- Link para assistir o funcionamento da Figura 6: [Comparador de 2 bits.](#)

3. Análise dos Resultados

Analisando os resultados separadamente temos:

- Experimento 1: Como esperado houve um tempo de atraso de propagação nas portas, sendo do clock e a saída L0 maior por ter uma porta a mais antes do

resultado final. Seguindo a teoria, quando mais portas maior será o tempo de atraso de propagação [Floyd 2009, Harris and Harris 2017].

- Experimento 2: Foi construído um circuito XOR usando apenas portas NAND e o resultado chegou ao que se esperava. Toda tabela verdade do circuito NAND coincidiu com a tabela verdade do circuito XOR. Após essa verificação foi implementado um comparador de 3 bits utilizando apenas o circuito NAND em forma de blocos e, assim, chegamos ao resultados da Tabela 2 e Tabela 3.
- Experimento 3: Nesse experimento, primeiramente foi feito um comparador de 1 bit utilizando apenas portas XOR, NOT e AND, e com esse comparador foi construída um comparador de 2 bits. Ambos comparadores também chegaram ao resultado esperado das suas tabelas verdade.

4. Conclusão

Nesse trabalho foi apresentado os resultados e análises dos exercícios propostos do Experimento 4 da matéria de Laboratório de Circuitos Lógicos - CIC0231.

Os resultados mostram que a prática representa exatamente o que está na teoria. Os atrasos das portas representam exatamente o intervalo de tempo entre a aplicação de um pulso na entrada e a ocorrência de um pulso resultante na saída e eles irão ser maiores cada vez que uma porta nova é adicionada ao circuito, como foi encontrado nos experimentos realizados. Já os comparadores, cumprem sua função básica de comparar as magnitudes de dois números binários, mostrado com os comparadores de 3 e 2 bits.

Referências

- Floyd, T. (2009). *Sistemas digitais: fundamentos e aplicações*. Bookman Editora.
- Harris, D. and Harris, S. L. (2017). *Digital design and computer architecture*. Morgan Kaufmann.
- Lamar, M. V. and Mandelli, M. G. (2022). Plano de ensino 2021/2.

Auto-Avaliação

1. b
2. d
3. d
4. a
5. d