

Experimento 7

Latches e Flip-Flops: RS e JK

Matheus Cardoso de Souza, 202033507
Ualiton Ventura da Silva, 202033580
Grupo G42

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

matheus-cardoso.mc@aluno.unb.br, 202033580@aluno.unb.br

Abstract. *The current report explores the construction of different models of Latches and Flip-flops, showing their specificities, utilities and inherent limitations. Such circuits, so widely used to build up computational circuits, show themselves as vital for the modern society, and, therefore, it's thorough study and understanding becomes of crucial importance for Computer Science students.*

Resumo. *O presente relatório aborda a construção de Latches e Flip-flops de diferentes modelos, explorando suas especificidades, utilidades e limitações inerentes. Tais circuitos, tão amplamente utilizados para a construção de circuitos computacionais atuais, mostram-se de vital importância para a sociedade moderna, e, portanto, seu estudo minucioso apresenta-se como um imperativo para estudantes de Ciências da Computação.*

1. Introdução

A princípio existem variados tipos de métodos utilizados para o registro de memória, mas os experimentos realizados visam descrever e utilizar aqueles que surgem do uso de semicondutores.

Com o uso de portas lógicas podem haver outras maneiras além da que será descrita no presente relatório, assim como um mesmo sistema de memória poderá ser descrito com diferentes portas lógicas. Por exemplo, um Latch RS poderá ser representado tanto com o uso de portas *NAND* como *NOR*, contudo, nos experimentos realizados optou-se pelo uso de portas *NAND*.

1.1. Objetivos

Apesar da existência de outros sistemas de memória, focou-se na elaboração e análise de Latch RS Simples, Latch RS Engatilhado, Flip Flop RS e Flip Flop JK.

É necessário notar que diante das análises de cada circuito de memória, alguns possuem estados proibidos e estes mesmos estados serão explorados mais a diante.

1.2. Materiais

Em função da natureza do ensino a distância, os presentes experimentos não foram realizados usando-se materiais e equipamentos físicos, mas sim emulados por meio do software Deeds.

A seguir estão enumerados os materiais utilizados:

- Software Deeds
- Portas lógicas
 - *OR*
 - *NOT*
 - *NANDs* de 2 e 3 entradas
- *Clocks*
- *LEDs*

2. Procedimentos

Passaremos a apresentar os experimentos requeridos.

2.1. Latch RS simples com NANDs

Para esta primeira parte do experimento, desejamos implementar um circuito que seja semelhante ao diagrama da figura 1.

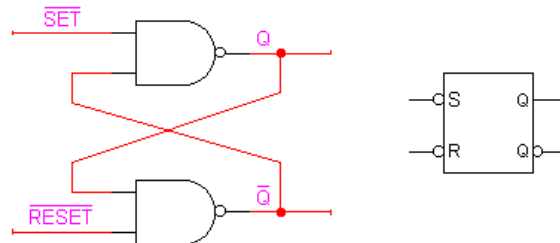


Figura 1. Latch RS (implementado com *NANDs*)

Com base em [Mandelli 2021c] sabemos que a tabela verdade para o circuito de uma *Latch RS* deve ser igual à expressa na tabela 1. Assim, temos como proposta de implementação o circuito demonstrado em 2.

Tabela 1. Tabela Verdade para *Latch RS*

Entradas		Saídas	
A (\overline{RESET})	B (\overline{SET})	L0 (Q)	L1 (\overline{Q})
0	0	1	1
0	1	1	0
1	0	0	1
1	1	Q_n	$\overline{Q_n}$

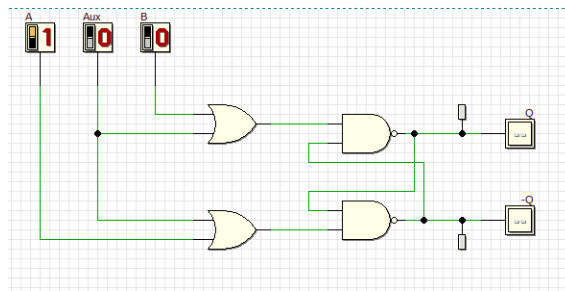


Figura 2. Latch RS (implementado com *NANDs*)

É interessante notar que o circuito em 2 não é exatamente igual ao do diagrama em 1 pois a passagem dos estados $11 \rightarrow 00 \rightarrow 11$ seria difícil de reproduzir manualmente. Dessa forma, foi adicionado um input adicional e duas portas *OR* para permitir uma transição mais exata entre esses estados.

Para a visualização do funcionamento deste circuito, confira no seguinte link: <https://youtu.be/RH6w3QnfUhA>

2.2. Latch RS engatilhado

Utilizando os modelos de circuitos apresentados temos que sua implementação através do software Deeds será:

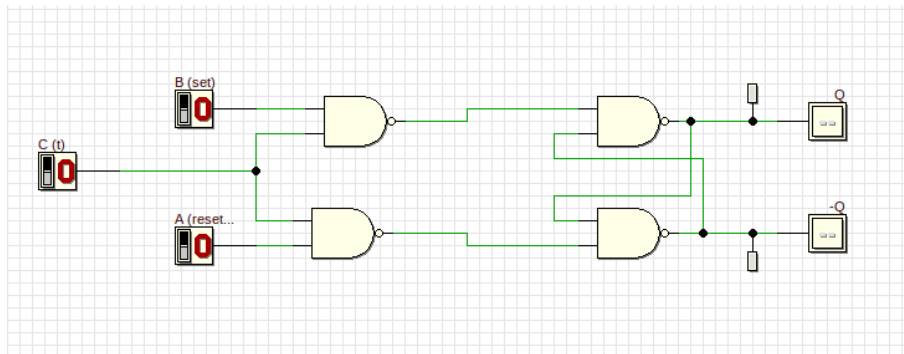


Figura 3. Latch RS Engatilhado (implementado com *NANDs*)

Sendo que a tabela verdade utilizada será:

Tabela 2. Tabela Verdade para *Latch RS Engatilhado*

Entradas			Saídas	
C (<i>TRIGGER</i>)	A (<i>RESET</i>)	B (<i>SET</i>)	L0 (Q_{n+1})	L1 ($\overline{Q_{n+1}}$)
0	X	X	Q_n	$\overline{Q_n}$
1	0	0	Q_n	$\overline{Q_n}$
1	0	1	0	1
1	1	0	1	0
1	1	1	1	1

A utilização de um gatilho permite definir quando poderá ocorrer o registro de bits ou não, portanto para $T = 0$, temos que o estado das saídas não poderá ser modificado, caso contrário temos a situação em que $T = 1$, sendo que este caso descrito possuirá o mesmo comportamento ao utilizar um Latch RS comum.

Para a visualização do funcionamento deste circuito, confira no seguinte link: <https://youtu.be/VNXkBLNcpbo>

2.3. Flip-flop RS

Implementando o circuito através do software Deeds, temos:

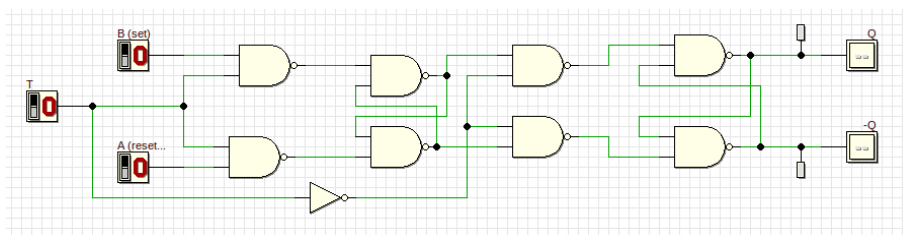


Figura 4. Flip Flop RS (implementado com NANDs)

Simulando tem-se o seguinte comportamento: <https://youtu.be/qPIkTOfm8uI>

Assim, sua tabela verdade será descrita por:

Tabela 3. Tabela Verdade para Flip-flop RS

Entradas			Saídas	
C (<i>TRIGGER</i>)	A (<i>RESET</i>)	B (<i>SET</i>)	L0 (Q_{n+1})	L1 ($\overline{Q_{n+1}}$)
0	X	X	Q_n	$\overline{Q_n}$
Clk_H	0	0	Q_n	$\overline{Q_n}$
Clk_H	0	1	0	1
Clk_H	1	0	1	0
Clk_H	1	1	1	1

Em questão de saídas poderá não parecer perceptível a diferença entre Latches e Flip Flops, contudo, um fator imprescindível entre ambos é o fato de que Latches para que possam criar sua mudança de estado, basta que o nível mude para um que possibilite esta mudança. Já Flip Flops, possuem a característica de que para criar suas mudanças será possível somente nas bordas da onda, ou seja, um flip flop sensível a onda de subida somente modificará seu estado no momento em que ocorre a transição de nível lógico baixo para alto, assim, define-se que Latches são sensível ao estado lógico, diferentemente de flip flops que são sensíveis às bordas.

2.4. Flip-flop JK

Temos que seu circuito será descrito no software Deeds como:

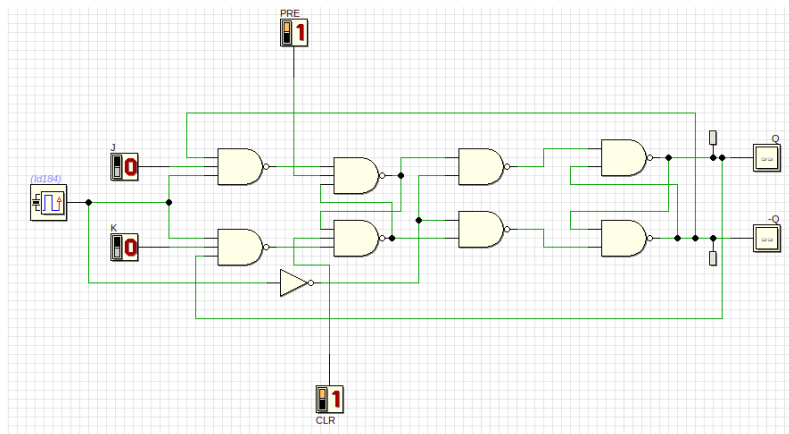


Figura 5. Flip Flop JK (implementado com *NANDs*)

Observe que para sua implementação fez-se necessário o uso de entradas de *PRE-SET* e *CLEAR*, por não haver estado inicial definido. Contudo, não utilizou-se a entrada de *CLEAR*, mas definiu-se para ser possível a criação de um modelo mais completo. Simulando obtemos o seguinte funcionamento:

<https://youtu.be/QCF_M5kqSQE>

Este mesmo *Flip Flop* poderá ser utilizado como um divisor de frequência de maneira que possa diminuir a frequência de um determinado clock. Segue abaixo o vídeo que descreve seu comportamento:

<https://youtu.be/00iOIssj00k>

Analisando de maneira geral, medindo 1 pulso de *Q*, obtemos:

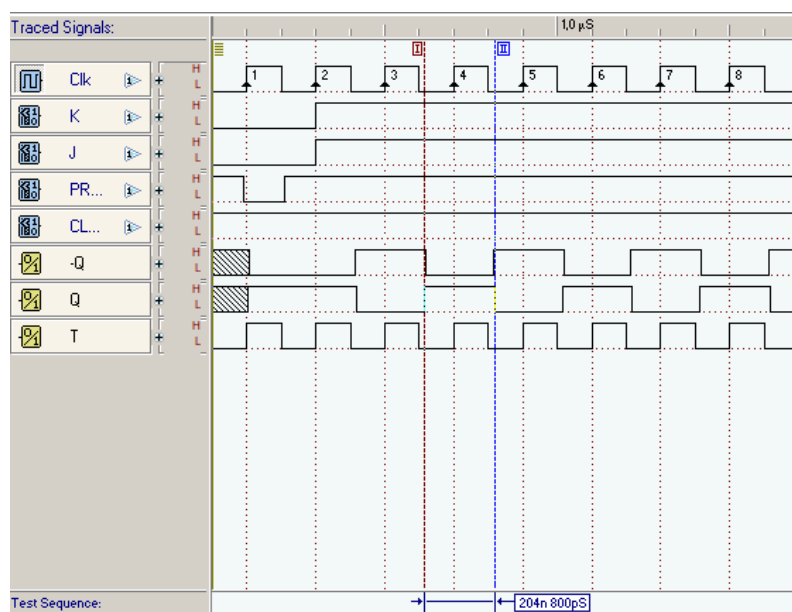


Figura 6. Medida de Saída em Q

Analisando 1 pulso de *T*, obtemos:

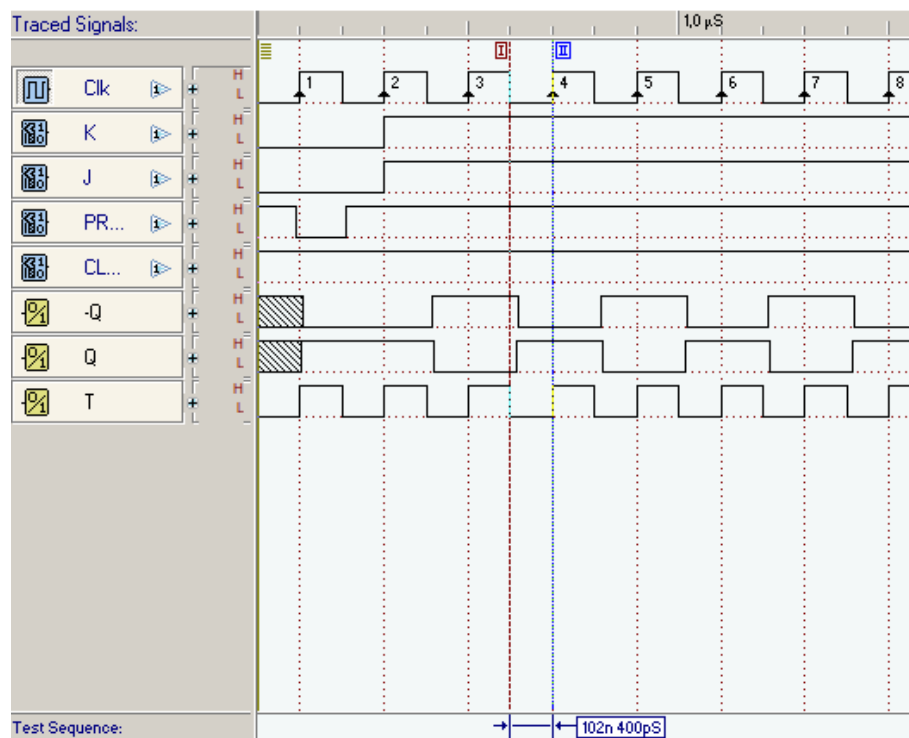


Figura 7. Medida de Saída em T

Como 1 pulso nos exemplos utilizados descrevem a metade de um período, temos então que seus períodos serão aproximadamente:

$$P_Q \approx 200ns$$

$$P_T \approx 100ns$$

Logo:

$$F_Q \approx \frac{1}{P_Q} \approx 0.005(ns^{-1})$$

$$F_T \approx \frac{1}{P_T} \approx 0.01(ns^{-1})$$

Calculando suas razões:

$$\frac{F_Q}{F_T} = 0,5$$

Assim, concluímos que a frequência de Q é metade da de T.

3. Análise dos Resultados

Passaremos a analisar individualmente cada um dos tópicos anteriores, levantando observações pertinentes para cada um deles.

3.1. Análise do tópico 2.1

No tópico 2.1 podemos fazer uma observação que se repetirá nos tópicos 2.2 e 2.3, que é referente a questão de *estados proibidos*. Como é possível perceber no vídeo deste experimento, quando há a transição do estado 00 para o estado 11, o software *Deeds*, que simula os circuitos digitais, levanta um erro de “loop infinito”. Esse comportamento

ocorre pois, no estado 00 o circuito entra no *estado proibido* pois, em um circuito real, após sair desse estado, a transição se dará de forma não determinística, sendo dependente de atrasos causados por fios e portas lógicas que compõem o *Latch RS* em específico. No caso do simulador *Deeds*, como não há atrasos de portas lógicas sendo simulado, o programa acaba por emular ambos Q e \overline{Q} indo para o estado oposto quando *SET* e *RESET* são postos em 1. Dessa forma, haverá uma eterna mudança entre os estados $\{Q = 1, \overline{Q} = 1\}$ e $\{Q = 0, \overline{Q} = 0\}$. Dessa forma, o programa para a execução levantando o erro de “loop infinito”.

Tal problema não aconteceria com outros circuitos, como por exemplo *Latches D*, assim como explicado em [Mandelli 2021c], nem em *Flip-flops D/JK* como mencionado em [Mandelli 2021b] e nem em *Flip-flops T* assim como ensinado em [Mandelli 2021a].

3.2. Análise do tópico 2.2

Semelhantemente ao experimento feito em 2.1, temos que o *estado proibido* está presente caso *SET* e *RESET* assumam o valor 1 ao mesmo tempo. E, dessa forma, ao mudar ambos para o valor 0 ao mesmo tempo, o circuito entra em estado não determinístico. E, uma vez que o *Deeds* não simula atraso aleatório inerente à portas lógicas e cabos usados para construção do circuito, o programa entra num estado de “loop infinito”, alternando entre estados de 0 e 1. Dessa forma, ele interrompe a execução da emulação levantando um erro.

3.3. Análise do tópico 2.3

A utilização de um Flip Flop e Latch RS possibilitou observar as diferenças contrastantes entre Flip Flops e Latches. E pode-se comprovar seus comportamentos.

3.4. Análise do tópico 2.4

Com o Flip Flop JK descrito, foi possível não somente a análise de seu funcionamento como o esperado assim como também a sua utilização para a criação de um divisor de frequência, podendo ser feita uma análise que buscava comprovar e evidenciar o comportamento divisor.

Deve-se atentar ao fato de que para aumentar as divisões bastaria criar um comportamento em série de vários Flip Flops JK.

4. Conclusão

Através dos experimentos realizados pudemos descrever parcialmente o comportamento de circuitos utilizados para o armazenamento de memória, assim como também pudemos analisar o erro associado aos mesmos.

Pode-se observar que circuitos sequencias podem ser utilizados não somente para o armazenamento de dados, mas também como controle de pulsos de clock e ainda serem empregados em circuitos como de divisores de frequências.

Referências

[Mandelli 2021a] Mandelli, M. G. (2021a). Flip-flop t e preset-clear. <https://aprender3.unb.br/mod/resource/view.php?id=386092>.

[Mandelli 2021b] Mandelli, M. G. (2021b). Flip-flops rs, d e jk. <https://aprender3.unb.br/mod/resource/view.php?id=386087>.

[Mandelli 2021c] Mandelli, M. G. (2021c). Latches rs e d simples. <https://aprender3.unb.br/mod/resource/view.php?id=386082>.

Auto-Avaliação

Respostas:

Questão	Resposta
1	D
2	A
3	B
4	D
5	C
6	D
7	D