



Relatório – Máquina de estados finitos

Disciplina: ELE0518 – Laboratório de Sistemas Digitais

Alunos: Bruno Matias de Sousa

Data: 12/04/2019

Levy Gabriel da Silva Galvão

Pedro Henrique de Souza Fonsêca dos Santos

1. Introdução

Dentro dos circuitos lógicos, usamos expressões booleanas, que nada mais são que expressões matemáticas, para descrever o comportamento de circuitos combinacionais. Porém, em circuitos sequenciais, temos a incorporação do tempo e da memória, o que torna necessário uma nova forma matemática de descrição dos eventos dentro do circuito.

Dessa forma, é incorporado o conceito de máquinas de estados finitos (FSMs, de *Finite State Machines*) que nos traz a facilidade de implementar elementos de memória com circuitos combinacionais comuns por meio de equações booleanas que satisfazem a tabela das transições que ocorrem no circuito.

Ainda é possível, por meio dessa tabela, escolher qual flip-flop se deseja utilizar como armazenador de memória, de forma a escolher o que se torna mais confortável para o projetista.

2. Referencial teórico

A resolução dos problemas desta prática necessitou dos conceitos que envolve a Máquina de Estados Finita (FSM) e dos conceito do uso dos dispositivos Flip-Flop. Primeiramente a Máquina de estado Finita do inglês FSM - Finite State Machine nada mais é de que uma diagrama de fluxo usado para representar um circuito lógico, seja ele simple ou complexo. A FSM podem modelar um grande número de problemas, entre os quais a automação de design eletrônico, projeto de protocolo de comunicação, análise e outras aplicações de engenharia. Com isso é possível representar os estados atuais e futuros do circuito e com isso retirar informações acerca da tabela verdade (tabela de estados) e obtenção do circuito equivalente.

Na FSM, os estados são representados por círculos, os círculos são ligados por uma flecha que indica a ação que aquele determinado estado deve realizar, nas flechas sempre tem indicado a variável de entrada (seletor) na qual o fluxo vai seguir, além disso se tiver uma saída, esta é representada em cima do círculo dos estado. Abaixo teremos uma figura que representa uma máquina de estado.

Para encontrar a máquina devemos seguir alguns passos:

1. Listar e Codificar os estados;
2. Através do fluxo, montar o diagrama de estados;
3. Utilizar o diagrama para montar a tabela de estados;
4. Fazer o circuito através da tabela de estados.

Os circuitos lógicos feitos até agora são circuitos combinacionais, na quais a saídas, em qualquer instante de tempo, dependem do níveis lógicos presentes na entrada, ou seja é um circuito sem memória. Para guardar a informação em circuitos lógicos precisamos de

elementos de memória. O elemento de memória mais importante é o Flip-Flop que é feito de uma configuração de portas lógicas. Esses circuitos que dependem do tempo são chamados de circuitos lógicos sequenciais. Na família de Flip-Flops temos o Tipo T, SR, D e o JK.

O Flip-Flop D, é um flip-flop que armazena o bit de entrada, do inglês Date (Dado) possui uma entrada, que é ligada diretamente à saída quando o clock é mudado. Independentemente do valor atual da saída, ele irá assumir o valor 1 se D=1 quando o clock for mudado ou o valor 0 se D=0 quando o clock for mudado. Este flip-flop pode ser interpretado como hold, visto que a informação é colocada na saída um ciclo depois de ela ter chegado na entrada. O flip-flop pode ser utilizado para armazenar um bit, ou um dígito binário de informação. A informação armazenada em um conjunto de flip-flops pode representar o estado de um sequenciador, o valor de um contador.

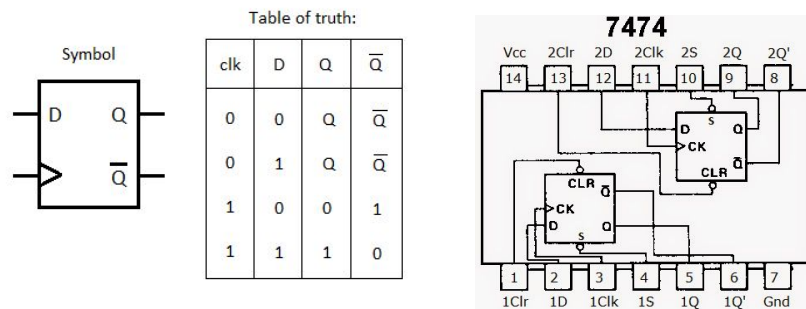


Figura 1 - Simbologia, tabela verdade e pinagem do Flip-Flop D.

O flip-flop JK funciona com a combinação J=1, K=0 é um comando para ativar (set) a saída do flip-flop, já combinação J=0, K=1 é um comando para desativar (reset) a saída do flip-flop alternando a condição inicial e a combinação J=K=1 é um comando para inverter o flip-flop, trocando o sinal de saída pelo seu atualizado, já se J=K=0 o comando mantém o estado anterior (hold). Fazendo J = K o flip-flop JK se torna um flip-flop T.



Figura 2 - Simbologia, tabela verdade e pinagem do Flip-Flop JK.

3. Metodologia

O presente projeto fez uso dos seguintes equipamentos:

- Protoboard;
- Fonte de tensão DC;
- Fios e conexões;
- Gerador de funções;
- Três resistores de 220Ω;
- Três LEDs vermelhos;

- Um CI 7404 (NOT);
- Um CI 7432 (OR);
- Dois CI's 7408 (AND);
- Um CI 7473 (flip-flop JK);
- Um CI 7474 (flip-flop D).

O projeto proposto visa produzir a sequência de saída abaixo:

000 → 100 → 110 → 111

O objetivo é que, quando o seletor H for 1 a sequência deve ser seguida e continuar repetindo, até que o seletor seja alterado para 0, permitindo que volte ao estado inicial imediatamente.

Com base nessa proposta, fora feito o diagrama de transição de estados para o circuito sequencial com a devida codificação de cada estado. Vide abaixo a máquina de estados e a codificação com a devida saída.



Figura 3 - Diagrama da máquina de estados finitos, à esquerda, seguido da codificação de cada um dos estados e respectiva saída, à direita.

Com isso feito, pode-se determinar a tabela de transição de estados, como na figura abaixo. Vale destacar que os estados estão codificados em dois bits e com a entrada H do seletor, implica em uma entrada de 3 bits para o circuito, com oito situações possíveis. A codificação dos estados futuros é facilmente identificada por meio da FSM da figura anterior. Os estados futuros também representam a entrada D dos flip-flops D. As saídas xyz são aquelas que devem seguir para alimentar os LEDs. Por fim, as colunas FF1 e FF0 representam as entradas Je K dos flip-flops JK.

ESTADO ATUAL		IN	ESTADO FUTURO		SAÍDAS			FF 1		FF 0	
s_1	s_0	H	n_1	n_0	x	y	z	J_1	K_1	J_0	K_0
0	0	0	0	0	0	0	0	0	X	0	X
0	0	1	0	1	0	0	0	0	X	1	X
0	1	0	0	0	1	0	0	0	X	X	1
0	1	1	1	0	1	0	0	1	X	X	1
1	0	0	0	0	1	1	0	X	1	0	X
1	0	1	1	1	1	1	0	X	0	1	X
1	1	0	0	0	1	1	1	X	1	X	1
1	1	1	0	0	1	1	1	X	1	X	1

Figura 4 - Tabela de estados com as entradas dos flip-flops JK e D.

Assim, a partir da tabela se pode determinar a equação booleana de cada um dos estados futuros e das saídas por meio do método desejado. O método utilizado foi o do mapa K, mas o processo não será detalhado aqui. As equações simplificadas resultantes estão dispostas logo abaixo.

$$D_1 = n_1 = s'_1s_0H + s_1s'_0H$$

$$D_0 = n_0 = s'_0H$$

$$J_1 = s_0H$$

$$K_1 = s_0 + H'$$

$$J_0 = H$$

$$K_0 = 1$$

$$x = s_0 + s_1$$

$$y = s_1$$

$$z = s_1s_0$$

Por meio destas equações, permite-se que construa o circuito, primeiramente em simulação de computador por meio do software Proteus e, depois, aplicado em prática com CIs e protoboard.

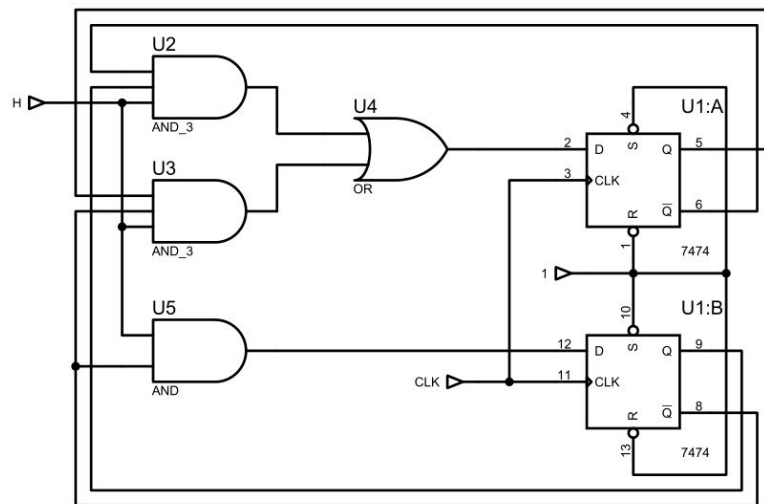


Figura 5 - Circuito sequencial implementado com flip-flops D.

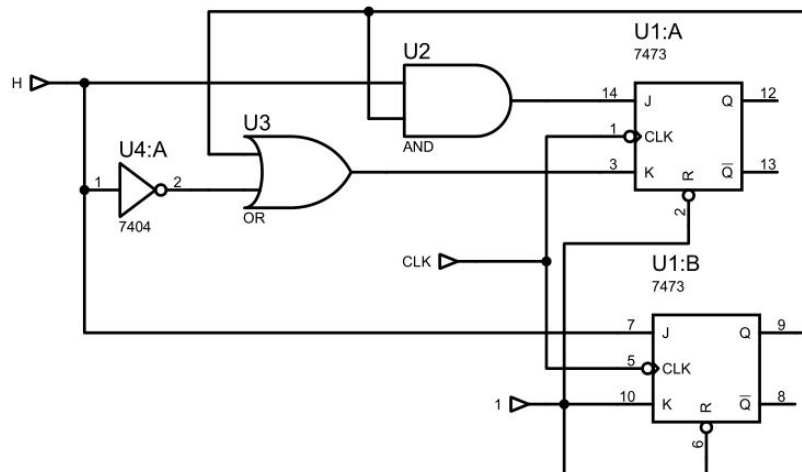


Figura 6 - Circuito sequencial implementado com flip-flops D.

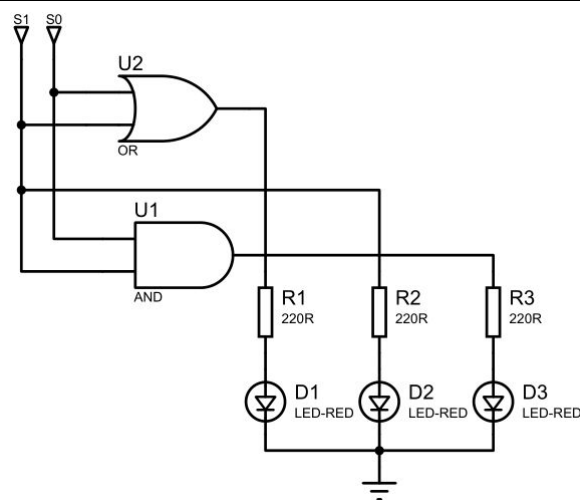


Figura 7 - Como o circuito para a implementação das saídas é comum para as duas aplicações (com JK e D), ele foi reservado nesta figura. Observa-se que as saídas x, y e z estão representadas, respectivamente, nos LEDs D1, D2 e D3.

O funcionamento e montagem em protoboard dos circuitos podem ser conferidos nos links para os vídeos do YouTube. O funcionamento do circuito com o flip-flop D pode ser conferido no link a seguir: <<https://youtu.be/wbqyYQqZfVk>>. Já o funcionamento com o flip-flop JK pode ser visto no link a seguir: <<https://youtu.be/r5fgteRvsfw>>. O clock utilizado possui 5V de amplitude e frequência variável e foi oriundo do gerador de funções.

4. Resultados práticos

Baseado na máquina de estados e na tabela de transição feita, foi montado um circuito com uma entrada H e três saídas x, y e z. O circuito funcionava de forma que sempre que H fosse 1, ele saía do estado inicial e ligava o primeiro LED, enquanto H continuasse 1, saía do segundo estado e ligava o segundo LED em conjunto e, por último, com H ainda em 1, ligava o terceiro LED junto aos outros antes de voltar para o estado inicial. Ele continua nesse loop até que H seja zero novamente. Em caso de H ir para zero no meio do loop, o circuito volta para o estado inicial no próximo clock.

Entre os dois flip-flops utilizados, foi possível ver uma diferença em dois pontos. O primeiro foi a tabela de transições que, no flip-flop D ficou bem mais simples, já que para cada saída, tínhamos apenas uma entrada, enquanto que para o flip-flop JK temos duas entradas para cada saída, o que trouxe o dobro de expressões para calcular em relação a entrada. Porém, o segundo ponto é exatamente o tamanho dessas expressões. Como a tabela de transição do flip-flop JK traz bem mais simplificações, as expressões diminuíram significativamente.

Vale lembrar que essa mudança nos flip-flops só trazem mudanças nas transições da entrada. As saídas que vão para x, y e z tinham as mesmas equações nos dois casos.

5. Conclusão

Com os testes feitos, o resultado satisfaz o esperado. A máquina de estados finitos é uma ótima forma matemática para facilitar a implementação de circuitos sequenciais, já que o tornam basicamente em circuitos combinacionais com elementos de memória em conjunto, como dito anteriormente.

6. Referências Bibliográficas

ABNT, Associação Brasileira de Normas Técnicas. **NBR 10719 – Apresentação de relatórios técnico-científicos**. Rio de Janeiro: ABNT, Copyright © 1989.

Fairchild Semiconductor, “**Dual Master-Slave J-K Flip-Flops with Clear and Complementary Outputs**,” DM7473 datasheet, Sep. 1986 [Revised Feb. 2000].

Fairchild Semiconductor, “**Dual Positive-Edge-Triggered D-Type Flip-Flops with Preset, Clear and Complementary Outputs**,” DM7474 datasheet, Sep. 1986 [Revised Jul. 2001].

MARCONI, Marina de A. & LAKATOS, Eva M. **Fundamentos de metodologia científica**. 5 ed. Editora Atlas. São Paulo, 2003.

VAHID, Frank. **Sistemas Digitais: Projetos, Otimização e HDLs**. 1 ed. Editora Bookman, 2008.