

Relatório

Heloisa Benedet Mendes
Bacharelado em Ciência da Computação
Universidade Federal do Paraná – UFPR
Curitiba, Brasil
heloisamendes@ufpr.br

Resumo—Detalhamento da sincronização de semáforos em um cruzamento rodoviário utilizando Máquina de Moore.

Index Terms—sincronização, semáforo, clock, botão, circuito, FSM, estado, saída.

I. INTRODUÇÃO

Este relatório irá detalhar a construção do projeto de sincronização de semáforos, feito a partir do simulador Digital. O trabalho em questão tem como objetivo a sincronização de 5 semáforos em um cruzamento rodoviário - 3 destinados aos carros e 2 aos pedestres - de forma que um pedestre possa apertar o botão implementado no sistema e atravessar a rua.

A sincronização é feita por meio da modelagem de uma Máquina de Estados Finitos ou *Finite State Machine* (FSM) [1], mais precisamente uma Máquina de Moore [2]. A FSM possui ao todo 7 estados, e terá um detalhamento mais aprofundado ao longo do documento.

O circuito criado foi feito com flip-flops do tipo D e conta com diversos circuitos integrados que promovem organização e facilitam sua visualização, bem como circuitos destinados apenas ao visual final do trabalho.

II. ENTENDIMENTO

De início, deve-se enumerar os semáforos a serem sincronizados de acordo com a figura 1, nota-se, então, que

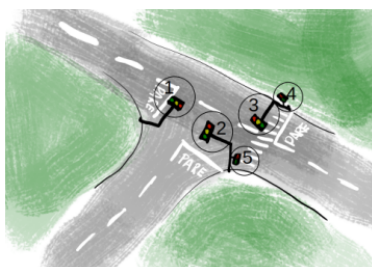


Figura 1.

os semáforos 1 e 3 são essencialmente iguais, e, por consequência, se comportam de forma contrária ao 2, bem como é descrito nas especificações do trabalho. Estes possuem as cores vermelho, amarelo e verde, a cor muda de acordo com o estado da máquina, portanto, a cada ciclo do clock integrado ao circuito.

Logicamente, os semáforos 4 e 5, destinados aos pedestres, também são idênticos, e mostram apenas as cores vermelho e

verde. Eles, no entanto, só devem apresentar a cor verde caso os semáforos 1, 2, e 3 estejam no estado vermelho, ocasião esta proporcionada unicamente pelo pressionamento do botão para pedestres.

Dessa forma, constrói-se a ideia de como deve ser pensado este sistema, e como deve ser construído o circuito, suas equivalências e seus detalhes.

III. MODELAGEM

O primeiro passo a ser dado neste projeto é a modelagem da FSM, construída como uma Máquina de Moore de 8 estados esquematizada na figura 2.

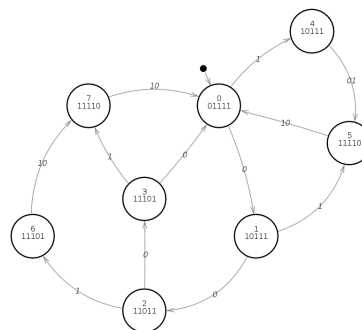


Figura 2.

A máquina sincroniza os 5 semáforos, e, como conclui-se da seção II, apenas 3 saídas são necessárias, levando em conta as equivalências próprias deste trabalho. Cada estado, portanto, possui 3 saídas divididas da seguinte forma: os 2 primeiros bits são destinados aos semáforos 1 e 3; os próximos 2 dizem respeito ao semáforo 2; e, por fim, o bit restante se refere aos semáforos 4 e 5.

Assim, é preciso determinar o que cada saída significa. A tabela I e II identificam as cores mostradas nos semáforos de acordo com as saídas:

Tabela I

Carros	
Saída	Cor
01	Verde
10	Amarelo
11	Vermelho

Tabela II

Pedestres	
Saída	Cor
0	Verde
1	Vermelho

A máquina construída opera da seguinte forma: parte do seu estado inicial (0) e recebe uma entrada 1 ou 0 - esta entrada depende do botão sincronizado [3] para pedestres, caso ele seja pressionado o sinal dado é 1, caso contrário, 0. Em seu ciclo normal, no qual o botão nunca é pressionado, a ordem é 0, 1, 2, 3, 0 ... onde todos os semáforos 1, 2 e 3 passam pelos estados verde, amarelo e vermelho, e ambos 4 e 5 permanecem unicamente no estado vermelho.

Caso o botão seja pressionado, em qualquer um dos estados, os semáforos terminarão seu ciclo atual e permanecerão no estado vermelho por um período de clock a mais, de forma que todos fiquem vermelhos ao mesmo tempo, é neste momento que o sinal para pedestres fica verde. Durante este mesmo período de clock, os pedestres irão atravessar a rua, quando o período chegar ao fim, o cruzamento voltará ao seu estado inicial.

IV. EXPRESSÕES

Para a montagem do circuito que sincroniza o cruzamento e obedece todas as especificações anteriormente citadas, necessita-se primordialmente construir uma tabela verdade para a FMS já apresentada.

E0n	E1n	E2n	B	E0...	E1...	E2...	S0	S1	S2	S3	S4
0	0	0	0	0	0	1	0	1	1	1	1
0	0	0	1	1	0	0	0	1	1	1	1
0	0	1	0	0	1	0	1	0	1	1	1
0	0	1	1	1	0	1	1	0	1	1	1
0	1	0	0	0	1	1	1	1	0	1	1
0	1	0	1	1	1	0	1	1	0	1	1
0	1	1	0	0	0	0	1	1	1	0	1
0	1	1	1	1	1	1	1	1	1	0	1
1	0	0	0	1	0	1	1	0	1	1	1
1	0	0	1	1	0	1	1	0	1	1	1
1	0	1	0	0	0	0	1	1	1	1	0
1	0	1	1	0	0	0	1	1	1	1	0
1	1	0	0	1	1	1	1	1	1	0	1
1	1	0	1	0	1	1	1	1	1	0	1
1	1	1	0	0	0	0	1	1	1	1	0
1	1	1	1	0	0	0	1	1	1	1	0

Figura 3.

$E0^{n+1} = (\overline{B} \wedge \overline{E0^n}) \vee (\overline{B} \wedge E0^n \wedge \overline{E2^n}) \vee (\overline{B} \wedge E1^n \wedge \overline{E2^n})$
 $E0^{n+1} = (\overline{B} \wedge \overline{E0^n}) \vee (\overline{B} \wedge E0^n \wedge \overline{E2^n}) \vee (E0^n \wedge E1^n \wedge \overline{E2^n})$
 $E1^{n+1} = (\overline{B} \wedge \overline{E0^n} \wedge \overline{E1^n} \wedge \overline{E2^n}) \vee (\overline{B} \wedge \overline{E0^n} \wedge E1^n) \vee (E1^n \wedge \overline{E2^n})$
 $E2^{n+1} = (\overline{B} \wedge \overline{E2^n}) \vee (\overline{B} \wedge E0^n \wedge \overline{E2^n}) \vee (E0^n \wedge \overline{E2^n})$
 $S0 = E0^n \vee E1^n \vee E2^n$
 $S1 = (E0^n \wedge \overline{E2^n}) \vee (E0^n \wedge E2^n) \vee E1^n$
 $S2 = E0^n \vee E1^n \veve E2^n$
 $S3 = (E0^n \wedge \overline{E2^n}) \vee (E0^n \wedge E2^n) \vee E1^n$
 $S4 = E0^n \vee E2^n$

Figura 4.

A tabela verdade ilustrada na figura 3 tem como base as entradas e saídas detalhadas anteriormente, e, a partir dela, utilizando mapas de Karnaugh e Álgebra Booleana, são criadas expressões matemáticas para cada uma das saídas.

Com base nas expressões obtidas, pode-se, efetivamente, iniciar o processo de montagem do circuito.

V. MONTAGEM

A construção do circuito em questão foi dividida em circuitos integrados, cada qual direcionado a uma parte específica do projeto. O circuito integrado *semaforos*, possui, por sua vez, subcircuitos nomeados de acordo com a simplificação booleana respectiva – figura 4 – cada qual construído a partir de portas lógicas, principalmente ANDs e ORs.

Os circuitos de nome $eXn+1$ tratam do próximo estado a partir do estado atual (EX), portanto, a saída do circuito $e0n+1$ deve ser a entrada D do flip-flop E0, por exemplo. Já os de nome SX, são as saídas do circuito, calculadas também com base no estado atual da FSM.

Ademais, existem também os circuitos específicos para a transformação das saídas SX em uma versão simplificada de semáforo, *cores* e *coresp* são ligados em LEDs de cores diferentes – de acordo com o semáforo em questão – e constroem a porção visual do projeto.

Todos estes circuitos compõem um circuito principal, o *semaforos*, e são sincronizados por um único clock - com a frequência de 0.5Hz, alcançada por meio de um flip-flop tipo JK - e pelo mesmo botão, que, por sua vez, é sincronizado por meio de um latch SR, no qual a saída S é ligada ao botão e R tem sua própria saída no circuito principal - o reset receberá sinal lógico 1 quando os semáforos para pedestres entrarem no estado verde, e 0 durante o restante do ciclo.

VI. CONCLUSÃO

Em conclusão, este é um trabalho de sincronização de 5 semáforos de um cruzamento de trânsito, produzido inteiramente no simulador Digital. O projeto foi construído com base em uma FSM do tipo Máquina de Moore, e em seu todo possui flip-flops do tipo D, principalmente, e do tipo JK, um latch SR e portas lógicas. Todos estes componentes estão distribuídos em diferentes circuitos integrados no trabalho.

A sincronização do circuito principal se faz por meio de um clock de 0.5Hz de frequência, e um botão sincronizado, que será pressionado quando o pedestre desejar atravessar, ele manterá os semáforos destinados aos carros no estado vermelho e permitirá a travessia.

REFERÊNCIAS

- [1] "Finite State Machine - circuito que passa por uma sequência finita de estados pré-determinados. a mudança de estados é controlada por um clock ou algum outro sinal de entrada." UFPR. [Online]. Available: <https://ufprvirtual.ufpr.br/mod/resource/view.php?id=877671redirect=1>
- [2] "Máquina de Moore - em uma máquina de moore, a saída depende apenas do estado atual." UFPR. [Online]. Available: <https://ufprvirtual.ufpr.br/mod/resource/view.php?id=877671redirect=1>
- [3] "Sincronização de botão," UFPR. [Online]. Available: <https://www.youtube.com/watch?v=e9rVVKUvj78>