

Experimento 1

INTRODUÇÃO ÀS MÁQUINAS DE ESTADOS SÍNCRONAS

	<i>Turmas A e C</i>	<i>Turmas B e D</i>
<i>Pré Relatório</i>	25/08/2015	27/08/2015
<i>Visto</i>	25/08/2015	27/08/2015
<i>Relatório</i>	01/09/2015	03/09/2015

I. OBJETIVO

Introduzir o conceito de máquinas de estados síncronas, focando nas etapas de projeto do esquemático a partir das equações booleanas, simulação, montagem do circuito, testes e correção de erros de projeto e montagem.

II. DESCRIÇÃO DO PROBLEMA

Considere o problema de um moedeiro para uma máquina de refrigerantes. O refrigerante custa R\$ 1,00, e a máquina aceita somente moedas de R\$ 0,50 e R\$ 1,00.

O usuário vai colocando moedas na máquina até atingir o valor desejado. Assim que o usuário tiver inserido o valor correto (no caso, moedas equivalentes a R\$ 1,00), a máquina deverá indicar que o valor correto foi atingido e colocar um refrigerante na bandeja. No entanto, se o usuário tiver inserido um valor maior que o preço do refrigerante, a máquina deverá indicar um erro e devolver ao usuário as moedas que ele inseriu. O usuário não pode abortar o processo no meio – a máquina vai continuar a receber moedas enquanto uma das duas condições não for alcançada.

III. PROJETO DA MÁQUINA DE ESTADOS

Antes de começarmos a projetar nossa máquina de estados, convém pensarmos nas entradas e saídas do circuito. Podemos definir:

ENTRADAS:

M – Moeda (0 – R\$ 0,50, 1 – R\$ 1,00)

C – Clock (isto é, o ato de empurrar a moeda para dentro da máquina).

SAÍDAS:

R – Indica o retorno do refrigerante.

E – Indica o erro.

Ou seja, o usuário irá definir a moeda que foi colocada através da entrada M e irá confirmar sua escolha através da entrada C. Podemos pensar que M é a saída de um subsistema que detecta qual moeda foi inserida, e trabalharmos apenas com o valor binário de qual moeda foi inserida. A saída R irá indicar à máquina que um refrigerante (previamente escolhido pelo usuário) deve ser colocado na bandeja, e a saída E indica à máquina que um erro foi detectado e, portanto, todas as moedas inseridas pelo usuário devem ser devolvidas. Ao fim do processo, tanto em caso de sucesso ou erro, uma nova batida do clock deverá retornar a máquina ao estado inicial.

A partir disso, podemos pensar nos estados que vamos gerar para a nossa máquina. Claramente, temos apenas 2 casos de sucesso: uma moeda de R\$ 1,00 foi inserida, ou duas moedas de R\$ 0,50 foram inseridas. Da mesma forma, temos apenas um caso de erro: uma moeda de R\$ 0,50 foi inserida e, então, uma moeda de R\$ 1,00 foi inserida (se a moeda de R\$ 1,00 fosse inserida primeiro, isso consistiria em um caso de sucesso, e a máquina não aceitaria a segunda moeda). Podemos então gerar a seguinte tabela de estados, vista na Tabela 1, com o seu respectivo diagrama de estados, visto na Figura 1. O significado de cada estado é o número e a ordem de moedas inseridas até aquele momento.

Significado	Estado	Próximo Estado se $M = 0$	Próximo Estado se $M = 1$	R	E
Estado Inicial (nenhuma moeda inserida)	E_0	E_1	E_4	0	0
{0,50}	E_1	E_3	E_2	0	0
{0,50 , 1,00}	E_2	E_0	E_0	0	1
{0,50 , 0,50}	E_3	E_0	E_0	1	0
{1,00}	E_4	E_0	E_0	1	0

Tabela 1 - Tabela de Estados e Saídas

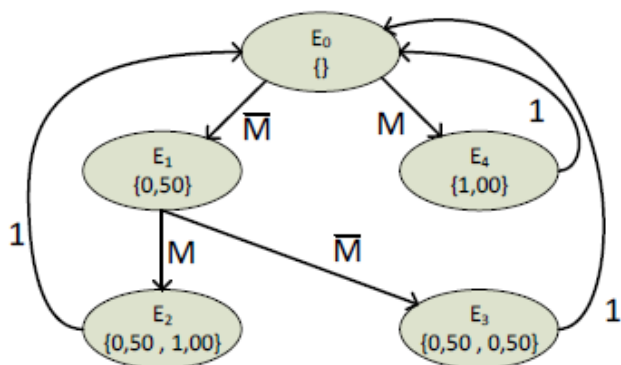


Figura 1 - Diagrama de Estados

Significado	Estado	Próximo Estado se $M = 0$	Próximo Estado se $M = 1$	R	E
Estado Inicial (Quantia = 0,00)	E_0	E_1	E_3	0	0
{0,50}	E_1	E_3	E_2	0	0
{1,50}	E_2	E_0	E_0	0	1
{1,00}	E_3	E_0	E_0	1	0

Tabela 2 - Tabela de estados e saídas reduzida

Podemos agora montar o diagrama de estados para o nosso projeto:

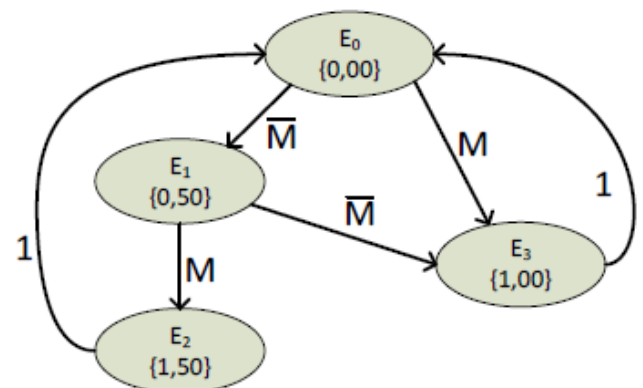


Figura 2 - Diagrama de Estados simplificado

Podemos observar que os estados E_3 e E_4 são redundantes, afinal, eles possuem a mesma saída ($R=1$ e $E = 0$), e levam ao mesmo próximo estado para as mesmas entradas M . Isso poderia ser deduzido pois, para a nossa máquina, a *ordem* em que as moedas são inseridas não importa, apenas a quantia total que foi inserida naquele momento. Logo, podemos juntar os estados E_3 e E_4 em um mesmo estado (denominado E_3). Note que não podemos juntar estes estados com o estado E_2 , pois, apesar de o próximo estado ser o mesmo, as saídas são diferentes.

Logo, podemos montar uma nova tabela de estados, onde desta vez o significado de cada estado é a quantia em dinheiro depositada até ali, conforme mostra a Tabela 2.

O próximo passo é designar códigos para os nossos estados. Como temos 4 estados, precisamos apenas de 2 bits e, portanto, podemos armazenar nosso estado com apenas 2 flip-flops. Podemos agora gerar a tabela de Transições e Saídas:

Estado	Código (Q_1Q_0)	Próximo Estado se $M = 0$	Próximo Estado se $M = 1$	R	E
E_0	00	01	11	0	0
E_1	01	11	10	0	0
E_2	10	00	00	0	1
E_3	11	00	00	1	0

Tabela 3 - Tabela de transições e saídas

Supondo agora o uso de flip-flops tipo D, podemos agora montar os mapas de Karnaugh para as variáveis de entrada desses flip-flops, D_1 e D_0 , onde D_1 é a entrada do flip-flop que gera Q_1 e D_0 a entrada do flip-flop que gera Q_0 .

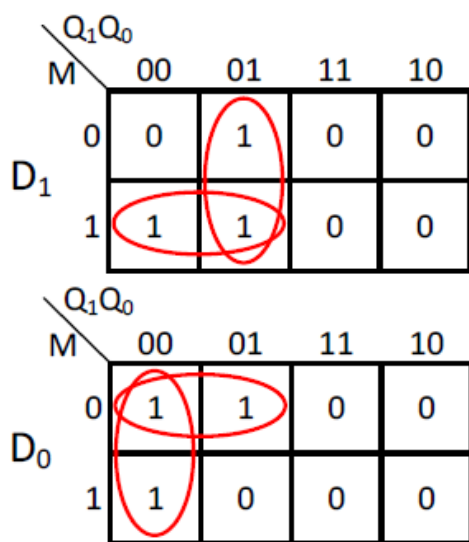


Figura 3 - Mapas de Karnaugh p/ FF D

Minimizando os termos, temos:

$$D_1 = M\bar{Q}_1 + \bar{Q}_1Q_0$$

$$D_0 = \bar{M}\bar{Q}_1 + \bar{Q}_1\bar{Q}_0$$

É fácil observar que as saídas são função apenas dos estados da máquina, não das entradas (isto é, a máquina é uma máquina do tipo Moore). Portanto:

$$R = Q_1Q_0$$

$$E = Q_1\bar{Q}_0$$

Outro flip-flop bastante utilizado é o flip-flop JK. É possível obter os mapas de Karnaugh correspondentes às entradas dos flip-flops — J_1 e K_1 , J_0 e K_0 — a partir dos mapas de Karnaugh das variáveis D_1 e D_0 . Isso é feito com base na tabela de transição apresentada na Figura 4.

transição de estado	entrada de dados	
	flip-flop D	flip-flop JK
$Q_n \rightarrow Q_{n+1}$	D	J K
0 → 0	0	0 x
0 → 1	1	1 x
1 → 0	0	x 1
1 → 1	1	x 0

Tabela 4 - Tabela de transição D → JK

Utilizando flip-flops JK, podemos escrever os mapas de Karnaugh:

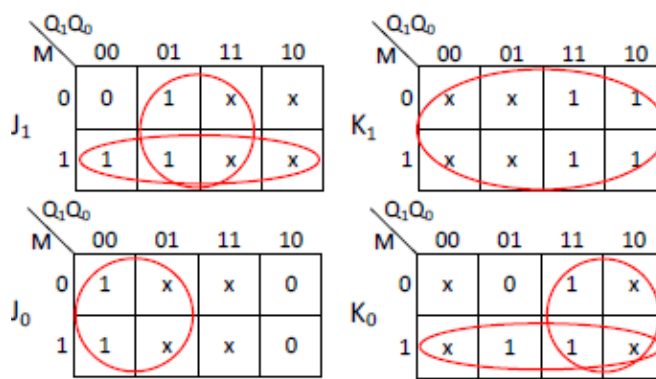


Figura 4 - Mapas de Karnaugh JK

Obtemos então as equações:

$$J_1 = M + Q_0 \quad K_1 = 1$$

$$J_0 = \bar{Q}_1 \quad K_0 = M + Q_1$$

As equações das saídas são as mesmas independente do flip-flop utilizado.

IV. PARTE EXPERIMENTAL

Monte a máquina de estados projetada na seção anterior, usando flip-flops JK. Ligue uma chave às entradas de CLEAR dos flip-flops. Basta uma única chave para todos os flip-flops (ver Figura 6). Essa chave funcionará como o botão de RESET do sistema, para que este volte ao estado inicial, 00.

Lembre-se que tanto a entrada CLEAR quanto a entrada PRESET são normalmente ativas em nível baixo. Portanto, a chave de RESET deve ser mantida em nível alto para operação normal do circuito.

Quanto à entrada PRESET, ela não será utilizada, mas deve-se ligá-la a um sinal de +5V para garantir que os flip-flops operem normalmente. Faça isso para os dois flip-flops! (Ver Figura 6).

IV.1 – Circuito de Debouncing

Debouncer é um circuito projetado para eliminar a trepidação da chave de duas posições. Tal circuito pode ser muito útil quando se utiliza uma chave de duas posições como relógio, de modo a controlar manualmente as transições de estado, para testes e debug do circuito. No caso de trepidação, um simples levantar ou abaixar de chave pode gerar múltiplas bordas de subida e descida na entrada de relógio dos flip-flops, fazendo com que a máquina de estado passe por múltiplas transições indesejadas.

Um debouncer simples pode ser implementado usando um flip-flop tipo D e uma onda quadrada de baixa frequência (Figura 5). Quanto mais baixa a frequência, maior a capacidade desse circuito de eliminar uma trepidação no sinal na entrada D do flip-flop. Por outro lado, uma frequência menor que 5 Hz é indesejável, uma vez que tornaria o processo de teste da máquina de estados muito lento. Infelizmente, o painel lógico utilizado no laboratório disponibiliza somente duas opções de ondas quadradas: 1 Hz e 100 kHz. Nesse caso, recomenda-se utilizar a onda de 100 kHz. Felizmente, esse valor de frequência é baixo o suficiente para eliminar a maior parte das trepidações geradas pelas chaves do painel.

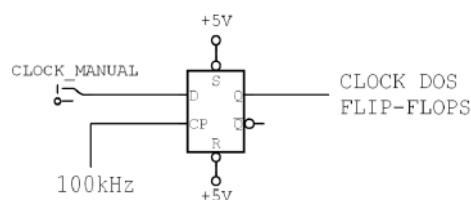


Figura 5 - Circuito de Debouncing

Como o mesmo sinal de relógio é utilizado para todos os flip-flops da máquina de estado, um único circuito de debouncing é necessário (Figura 6). Note que não há necessidade de eliminar trepidação na variável M, uma vez, que o conteúdo desta variável é lido somente nos instantes de transição do relógio, durante os quais essas variáveis devem ser mantidas estáveis.

IV.2 – Dicas para teste e debug do sistema

Ligue LEDs nas saídas (Q) dos flip-flops. Assim, será possível saber em qual estado a máquina está e monitorar o seu funcionamento, percorrendo o diagrama de estados.

Ao constatar que uma transição não está ocorrendo conforme esperado, tente identificar qual dos flip-flops não está atuando corretamente. Por exemplo: se a máquina está no estado 00 e deveria ir para o estado 01, mas ao invés disso ela vai para o estado 11, então o segundo flip-flop está mudando de estado corretamente. Portanto, o problema está no primeiro flip-flop: ele deveria estar fazendo a transição 0→0, mas está fazendo a transição 0→1.

Como a entrada K não tem efeito nessas duas transições (é don't care), então o problema deve estar na entrada J. Verifique, portanto, a lógica que gera essa entrada. Tente usar esse tipo de abordagem para testar e corrigir todas as transições de estado possíveis do circuito.

V. INSTRUÇÕES PARA A REALIZAÇÃO DO EXPERIMENTO

V.1 Pré-relatório

O pré-relatório deste experimento consiste em projetar o circuito que realiza as equações vistas na Seção 3, usando tanto flip-flops tipo D e usando flip-flops tipo JK (ou seja, projete dois circuitos). Utilizando flip-flops tipo D, seu circuito deve utilizar no máximo 5 CIs, e utilizando flip-flops tipo JK, seu circuito deve utilizar no máximo 3 CIs (em ambos os casos, sem contar o circuito de debouncer).

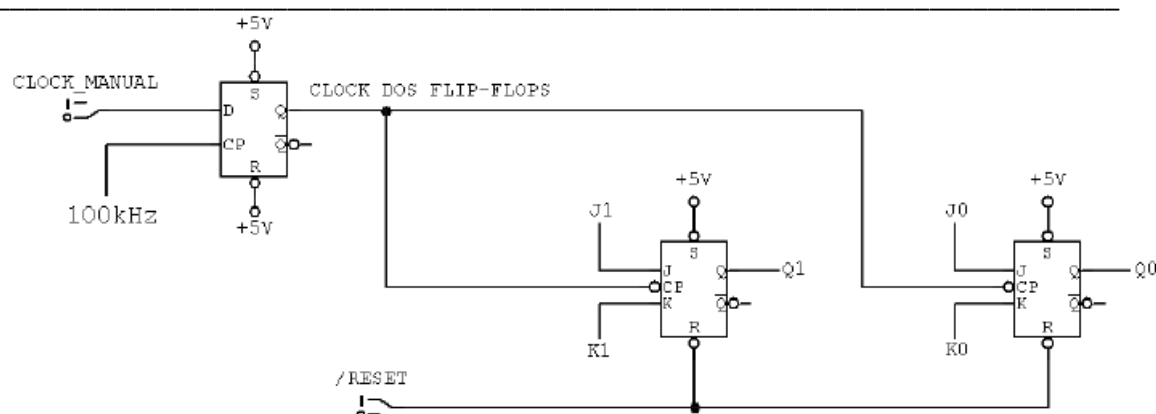


Figura 6 - Montagem do Circuito com o Debouncer

Traga o arquivo contendo este projeto para o laboratório – será muito mais fácil debugar o circuito se você tiver acesso à sua simulação. Além disso, utilizando o Circuito Maker, imprima a pinagem do circuito que você projetou (novamente, será mais fácil montar o circuito no protoboard utilizando essa pinagem).

V.2 - Realização do experimento (Visto)

Cada grupo deve seguir o procedimento apresentado no item IV deste roteiro. O circuito da máquina de estados vale 100% da nota do visto. Só serão aceitos circuitos que estejam funcionando integralmente. Ao terminar, chame o professor ou o monitor para verificar o funcionamento do circuito e anotar o visto.

V.3 - Relatório

O relatório é individual, deve ser feito à mão, e consiste em responder ao questionário abaixo. Não é necessário entregar um relatório formal, com introdução, metodologia, resultados, etc.

- 1) Explique **detalhadamente** o funcionamento do circuito de debouncing e qual a utilidade do mesmo no projeto. (2 pontos)
- 2) Explique como a utilização de LEDs adicionais pode ser útil na detecção de erros no circuito. (1 ponto)

- 3) Ao testar uma máquina de estados, Joãozinho esperava que a máquina alternasse do estado 01 para o estado 11 na subida do clock, mas a máquina foi para o estado 10. Explique em que parte do circuito está o erro, e como Joãozinho pode achá-lo e corrigi-lo rapidamente. Suponha primeiro o uso de flip-flops do tipo D e a seguir do tipo JK. (2 pontos)
- 4) Deseja-se adicionar um sistema de troco na máquina de refrigerantes. A máquina também passa a aceitar moedas de R\$ 0,25, R\$ 0,50, R\$ 1,00 e notas de até R\$ 2,00. O novo preço do refrigerante passa a ser de R\$ 1,50. Além das duas saídas já existentes, foi adicionada a saída que indica o troco do cliente. A saída de erro agora indica a inserção, a qualquer momento, de uma nota superior a R\$ 2,00. O sub-circuito que detecta as moedas, agora tem 6 opções: nenhum valor inserido, detectou-se moeda de R\$ 0,25, detectou-se moeda de R\$ 0,50, detectou-se moeda de R\$ 1,00, detectou-se uma nota de R\$ 2,00 e detectou-se uma nota superior a R\$ 2,00 (observe que será necessária mais de uma chave para todas estas opções). Caso uma nota superior a R\$ 2,00 seja inserida a qualquer momento, a máquina vai para um estado de erro e retorna automaticamente ao estado inicial. Monte a tabela de estados mínima para esta máquina, detalhe o significado de cada estado e desenhe o diagrama de estados para esta tabela. (5 pontos)