

Relatório do experimento 2: contadores

Henrique Koji Miyamoto (RA 169614) e Pedro Luís Azevedo Costa (RA175857)

1 INTRODUÇÃO

Nesse experimento, foram projetados contadores binários, a partir do conceito de máquina de estados finitos, e usando três propostas diferentes: um contador assíncrono (baseado em divisor de frequências), um síncrono com portas lógicas (utilizando flip-flops JK) e um síncrono com ROM. Todas elas têm a mesma função (contar três bits, isto é, de 0 a 7 em decimal), no entanto, seu desempenho - e, conseqüentemente, o contexto de aplicação - varia de acordo com a implementação.

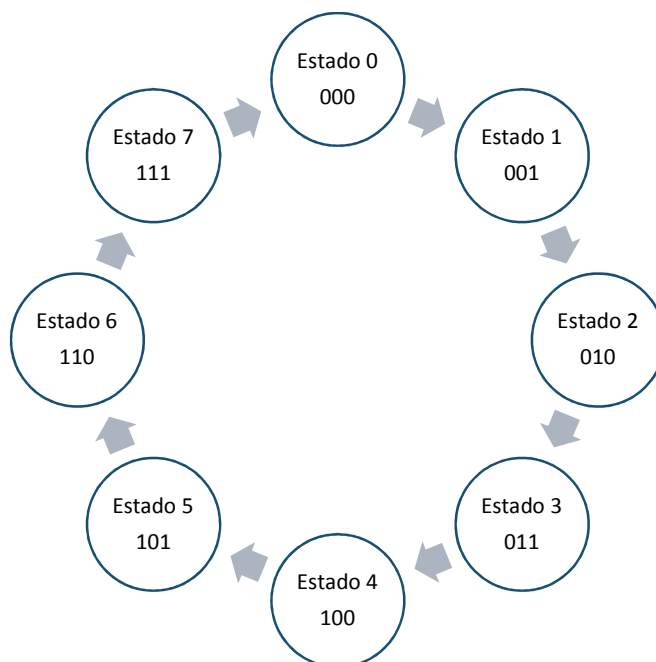


Figura 1. Representação de estados para contador de 3 bits

2 MATERIAIS E PROCEDIMENTOS

Nesse experimento, foi utilizado o programa *Quartus II versão 13.0 Web Edition*, um ambiente de prototipação e simulação de circuitos lógicos, para montar e testar os circuitos. O programa permite a simulação funcional e temporal: aquela considera as portas lógicas como ideias, isto é, sem atrasos, enquanto esta considera os atrasos de propagação.

2.1 CONTADOR ASSÍNCRONO

O contador assíncrono é baseado em um divisor de frequências. Na nossa implementação, utilizamos flip-flops D cuja entrada é a saída negada, de modo que a frequência do *clock* seja reduzida à metade em cada flip-flop do circuito. Foram utilizados três flip-flops D, um para cada bit da contagem, e três portas inversoras, conforme diagrama a seguir.

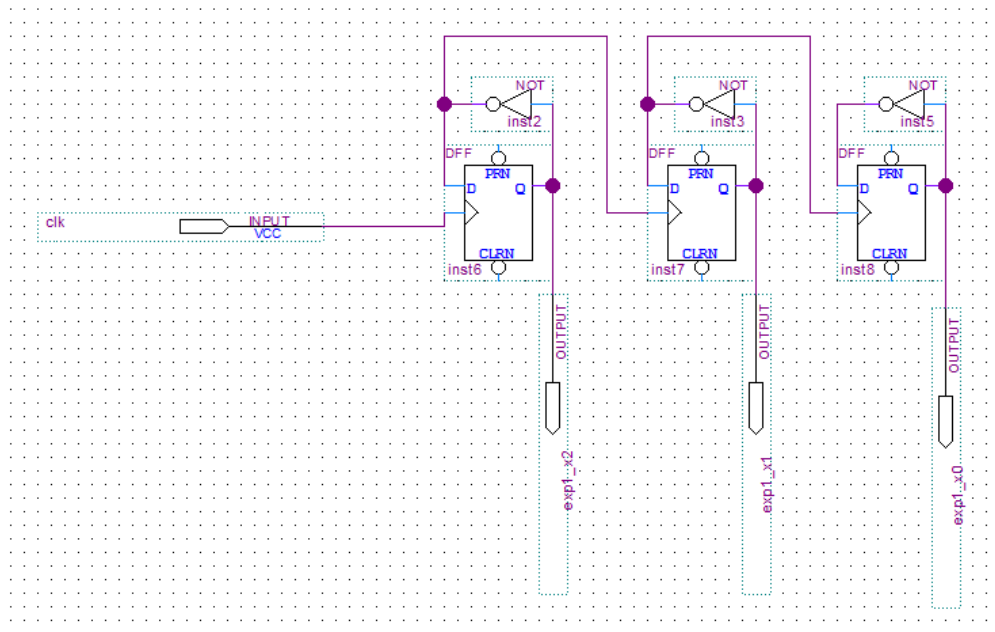


Figura 2. Diagrama do contador assíncrono

Essa implementação é assíncrona porque cada flip-flop recebe o sinal do *clock* em um instante diferente, devido ao atraso de propagação de um flip-flop para o outro.

2.2 SÍNCRONO COM PORTAS LÓGICAS

Na segunda proposta implementação, montamos um contador a partir de um circuito combinacional com flip-flops JK. Diferentemente, da implementação anterior, nessa, todos os flip-flops estão ligados diretamente ao sinal do *clock*, portanto, trata-se de um contador síncrono.

Os flip-flops JK seguem a seguinte lógica interna:

Tabela 1. Lógica de funcionamento do flip-flop JK

J	K	$Q(t+1)$
0	0	$Q(t)$
0	1	0
1	0	1
1	1	$\overline{Q(t)}$

Desse modo, a minimização que realizamos foi de acordo com a seguinte tabela-verdade:

Tabela 2. Tabela-verdade montada para minimização do circuito

y_2	y_1	y_0	Y_2	Y_1	Y_0	J_2	K_2	J_1	K_1	J_0	K_0
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	0	0	X	1	X	X	1
0	1	0	0	1	1	0	X	X	0	1	X
0	1	1	1	0	0	0	X	X	1	X	1
1	0	0	1	0	1	X	0	0	X	1	X
1	0	1	1	1	0	X	0	1	X	X	1
1	1	0	1	1	1	X	0	X	0	1	X
1	1	1	0	0	0	X	1	X	1	X	1

y: estado atual, Y: próximo estado, X: estado *don't care*

E os mapas de Karnaugh utilizados para minimização foram os seguintes:

J_2	$x_1'x_0'$	$x_1'x_0$	x_1x_0	x_1x_0'
x_2'	0	0	1	0
x_2	X	X	X	X

K_2	$x_1'x_0'$	$x_1'x_0$	x_1x_0	x_1x_0'
x_2'	X	X	X	X
x_2	0	0	1	0

J_1	$x_1'x_0'$	$x_1'x_0$	x_1x_0	x_1x_0'
x_2'	0	1	X	X
x_2	0	1	X	X

K_1	$x_1'x_0'$	$x_1'x_0$	x_1x_0	x_1x_0'
x_2'	X	X	1	0
x_2	X	X	1	0

J_0	$x_1'x_0'$	$x_1'x_0$	x_1x_0	x_1x_0'
x_2'	1	X	X	1
x_2	1	X	X	1

K_0	$x_1'x_0'$	$x_1'x_0$	x_1x_0	x_1x_0'
x_2'	X	1	1	X
x_2	X	1	1	X

Figura 3. Mapas de Karnaugh para minimização das funções de Boole

De modo que o resultado da minimização foi:

$$J_2 = K_2 = x_1x_0$$

$$J_1 = K_1 = x_0$$

$$J_0 = K_0 = 1$$

Portanto, projetamos o seguinte circuito:

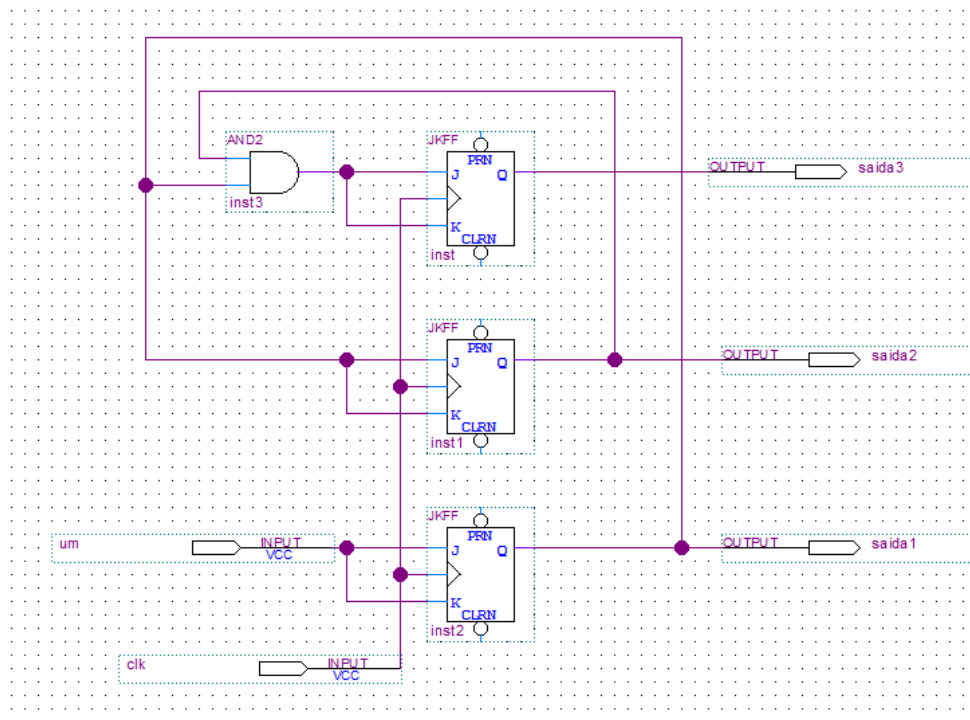


Figura 4. Diagrama da implementação síncrona com portas lógicas

2.3 SÍNCRONO COM ROM

A terceira implementação utiliza uma memória somente de leitura (*read only memory*, ROM). Essa memória corresponde, segundo Ercegovac (2000), “a uma representação tabular de um sistema combinacional em que o vetor de bits de entrada é usado para identificar qual linha da matriz é acessada”.

A essa memória ROM está associada a uma tabela-verdade como a seguinte, que relaciona cada entrada (endereço) com sua saída (conteúdo):

Tabela 3. Tabela-verdade da ROM

Entrada	Saída
000	001
001	010
010	011
011	100
100	101
101	110
110	110
110	000

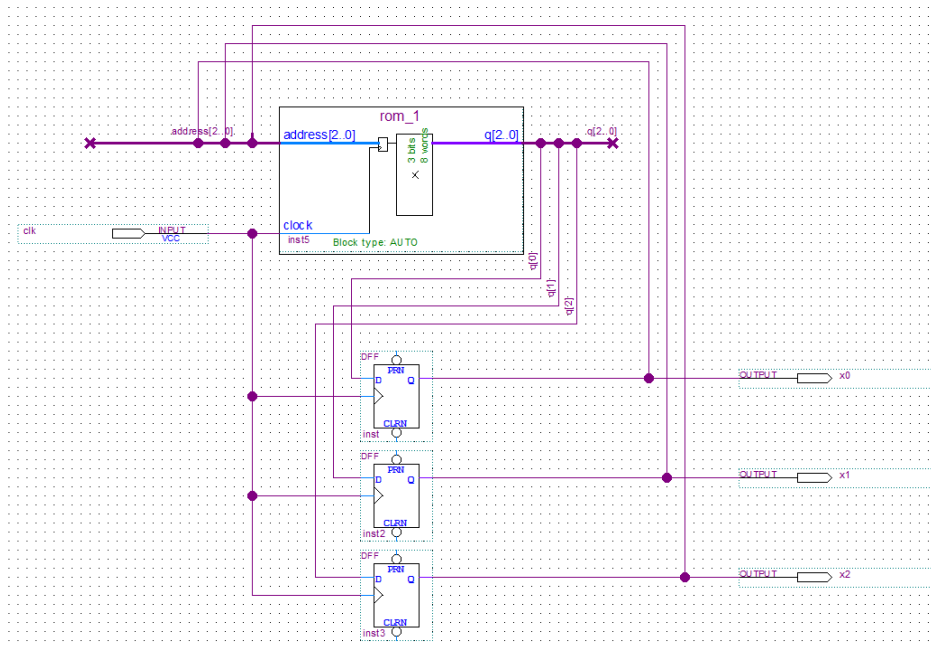


Figura 5. Diagrama da implementação síncrona com portas lógicas

3 RESULTADOS

A seguir, apresentamos e comentamos os resultados das simulações dos nossos circuitos. Em cada imagem, a primeira linha se refere ao contador assíncrono; a segunda, ao síncrono com portas lógicas (flip-flop JK); e a terceira, ao síncrono com ROM. A primeira imagem apresenta o resultado da simulação funcional, isto é, sem levar em conta os atrasos de propagação.

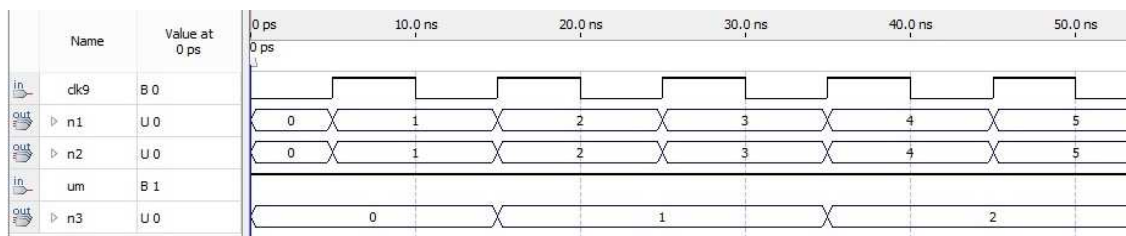


Figura 6. Resultado da simulação funcional para as três implementações

A segunda imagem mostra o resultado da simulação temporal, isto é, considera os atrasos do arranjo lógico do circuito.

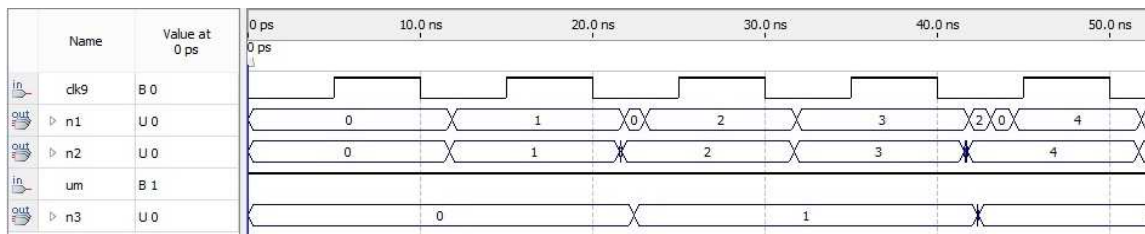


Figura 7. Resultado da simulação temporal para as três implementações

Notamos que não há grande diferença entre a primeira e a segunda implementação quanto à velocidade de contagem – que é menor que no circuito com ROM. No entanto, é notado que, na primeira, temos mais problemas com números indesejáveis, que aparecem momentaneamente entre dois números da contagem. Isso se deve ao atraso de propagação do sinal do *clock*, o que é muito mais perceptível no contador assíncrono, pois o sinal que entra nos dois últimos dispositivos vem da saída dos anteriores.

Em verdade, o fenômeno acima descrito ocorre em todas as implementações, mas em menor escala (números indesejáveis aparecem menos vezes e por menos tempo). A figura a seguir, mostra em detalhe um trecho da simulação temporal, de forma a evidenciar a consequência do atraso na contagem pelo segundo método, que é, no entanto, mais eficiente nesse aspecto, pois todos os flip-flops recebem o sinal do *clock* simultaneamente.

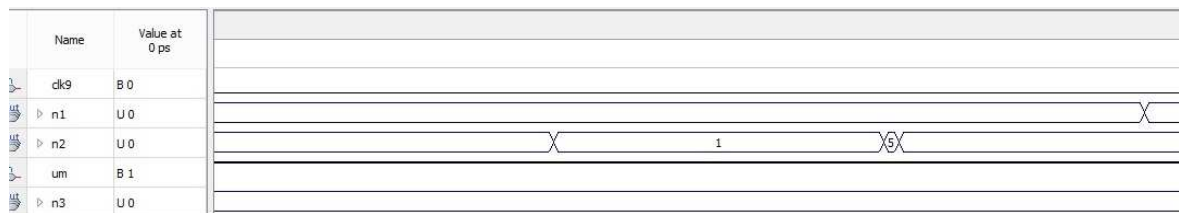


Figura 8. Detalhe da simulação temporal para as três implementações

4 DISCUSSÃO

Eis aqui algumas considerações sobre a realização dos experimentos.

- Na primeira vez que simulamos o contador assíncrono, nos deparamos com o problema de ele contar regressivamente. No entanto, esse problema pôde ser facilmente resolvido: sua causa era as saídas estarem conectadas entre o inversor e a entrada D dos flip-flops; então mudamos o arranjo de modo que estivessem conectadas entre a saída Q e a porta NOT, o que resolveu o problema.
- Não tivemos problemas significativos com a implementação do circuito síncrono com portas lógicas, no entanto foi o circuito mais trabalhoso de projetar dentre os três, já que envolveu uma minimização através de mapas de Karnaugh e planejamento do uso de flip-flops JK. Isso possibilitou obter um circuito mais simples do que obteríamos usando flip-flops D.
- O projeto síncrono com ROM foi o de mais difícil execução. Isso porque não conhecíamos o funcionamento de tal dispositivo e seu uso no aplicativo requer configurações um pouco mais complicadas, que demoramos para compreender.

Podemos, também, discutir em que situações seria apropriado usar cada uma das técnicas utilizadas.

- A simulação temporal do circuito assíncrono mostra que ele é o menos eficiente, no sentido de que os atrasos de propagação levam ao surgimento de números diferentes da sequência de contagem por um intervalo de tempo relativamente significativo (o maior dentre as três implementações). Desse modo, esse arranjo não é recomendado para circuitos que operam rapidamente ou que tenham muitos bits de contagem, pois cada flip-flop D a mais aumenta o atraso de propagação. No entanto, trata-se da técnica de mais rápida e fácil implementação, logo pode ser usado em casos de pouca complexidade.
- O contador síncrono com portas lógicas tem uma velocidade de contagem muito próxima à do contador assíncrono, porém os atrasos têm menor efeito no funcionamento do circuito. O intervalo de tempo em que os bits são diferentes da sequência é bem menor, de modo que talvez o não seja grande o suficiente para afetar as saídas no circuito real.
- O contador síncrono com ROM tem uma velocidade de contagem que é duas vezes menor que a das outras implementações. Talvez isso se deva a uma característica interna dele, como maior necessidade de processamento. Entretanto, quase não há problemas devido ao atraso e sua implementação requer poucos componentes: basicamente a ROM, o que torna a implementação mais fácil (por exemplo, não há necessidade de simplificações como na técnica anterior).

Desse modo, o experimento realizado possibilitou projetar diferentes técnicas para a realização da mesma tarefa e, ao comparar o desempenho de cada uma delas, pudemos entendê-las melhor.

5 REFERÊNCIAS

ERCEGOVAC, Milos D. ; LANG, Tomás ; MORENO, Jaime H. *Introdução aos sistemas digitais*. Tradução: José Carlos Barbosa dos Santos. Porto Alegre: Bookman, 2000.