

Infraestrutura de Hardware

Aula 02 - Circuitos Digitais com Memória

2016.2

João Marcelo Teixeira
joao.marceloteixeira@ufrpe.br

DEINFO - UFRPE

2 Circuitos Digitais com Memória

Este é o Segundo de dois capítulos que contêm uma revisão de projeto de circuitos digitais:

- Circuitos combinatórios (sem memória) no capítulo 1
- Circuitos sequenciais (com memória) no capítulo 2

Tópicos neste capítulo

2.1 Latches, Flip-Flops, and Registradores

2.2 Máquinas de Estado Finito

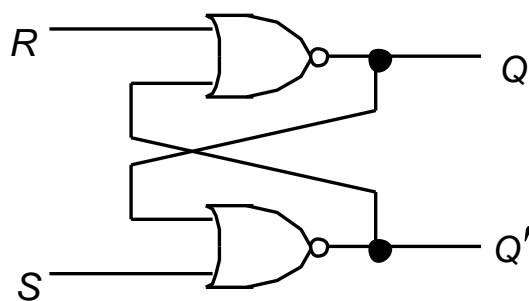
2.3 Projeto de Circuitos Sequenciais

2.4 Elementos Sequenciais Úteis

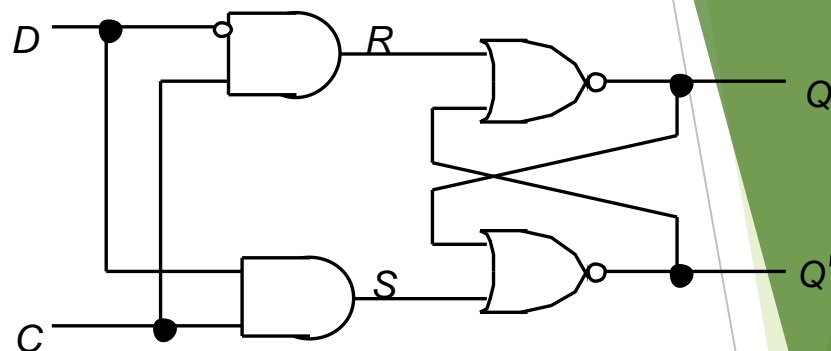
2.5 Elementos Sequenciais Programáveis

2.6 Clocks e Temporização de Eventos

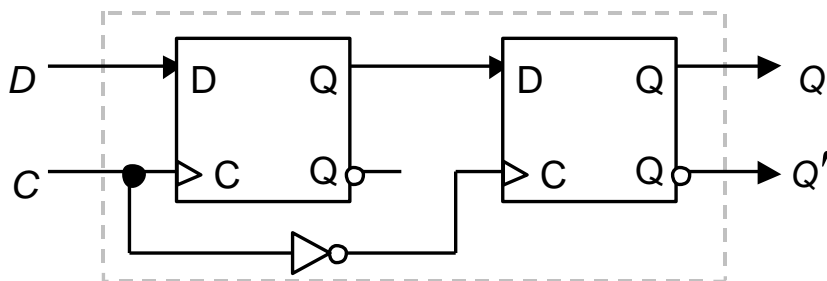
2.1 Latches, Flip-Flops e Registradores



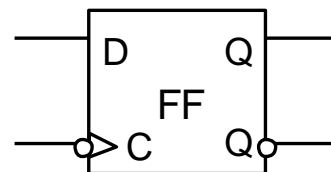
(a) SR latch



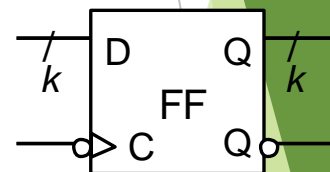
(b) D latch



(c) Master-slave D flip-flop



(d) D flip-flop symbol



(e) k -bit register

Figura 2.1 Latches, flip-flops e registradores.

Latches vs Flip-Flops

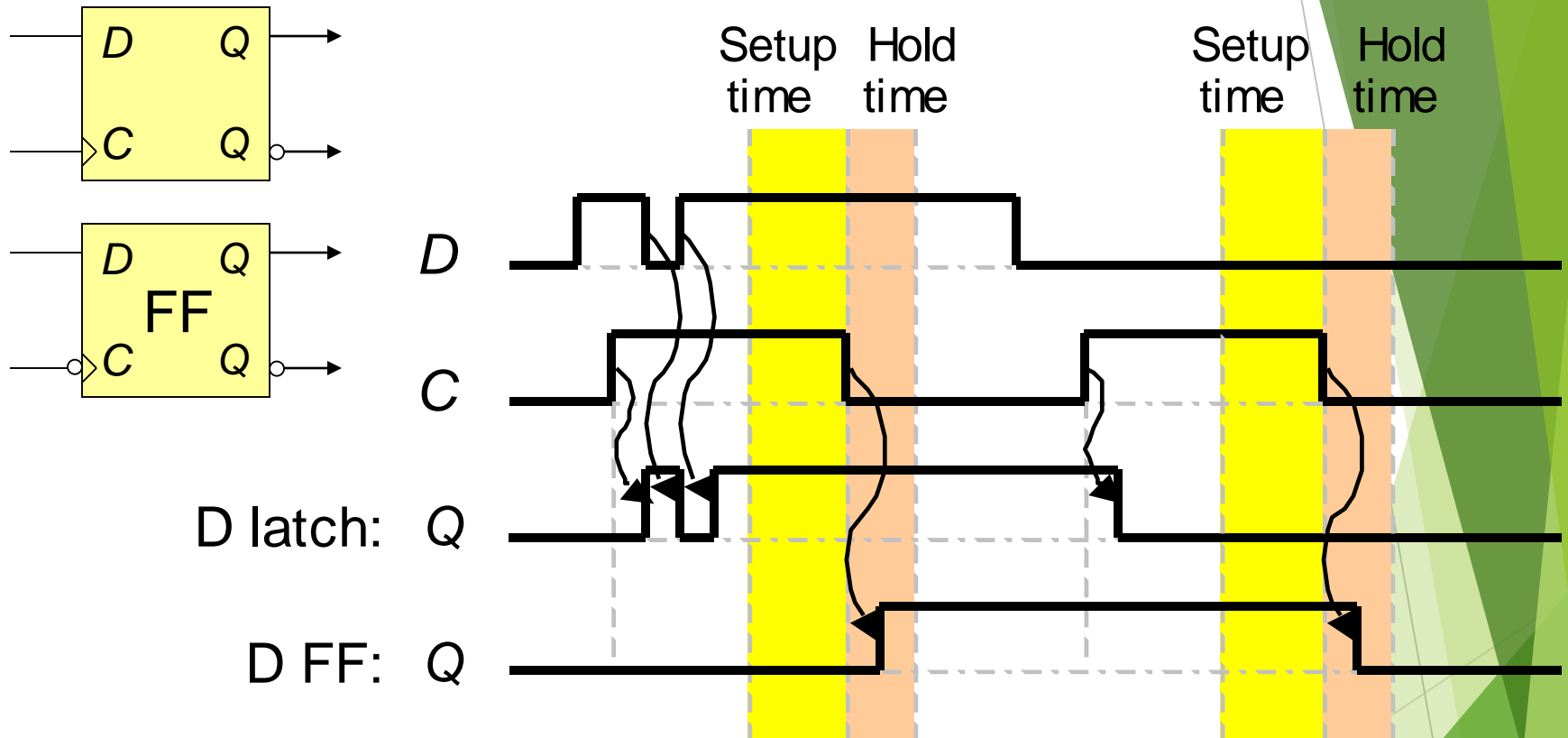


Figura 2.2 Operações de um latch D e um flip-flop D disparado por borda negativa.

Lendo e Modificando Flip-flops no Mesmo Ciclo

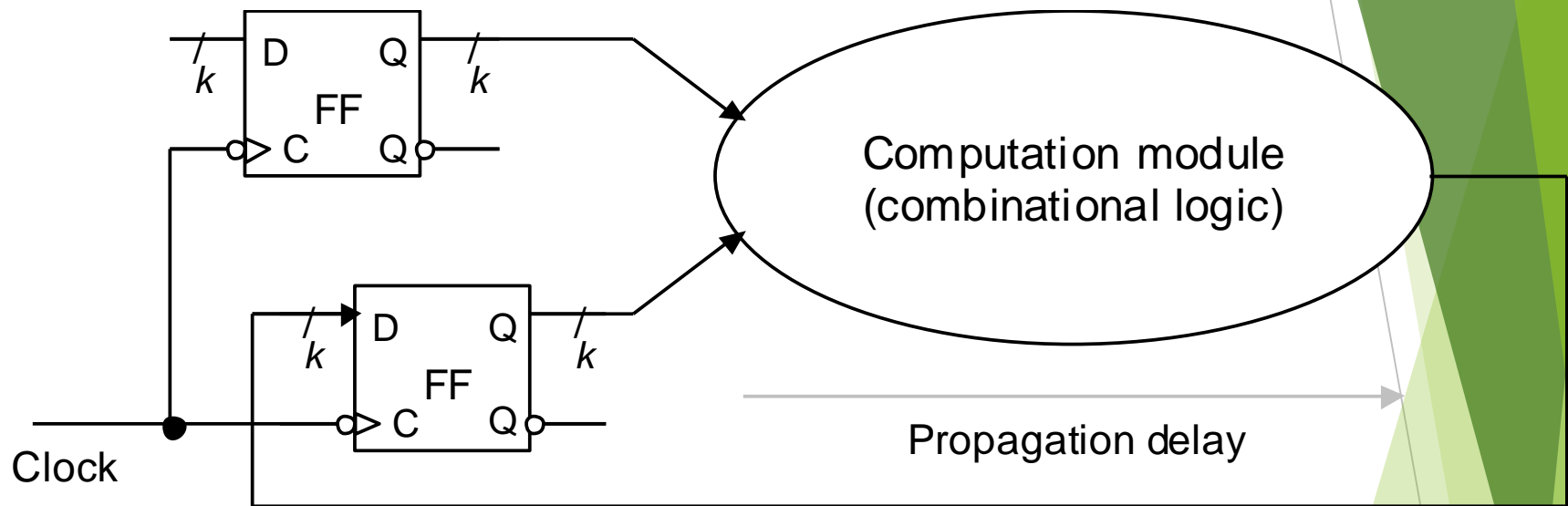


Figura 2.3 Operação registrador-para-registrador com flip-flops disparados por borda.

2.2 Máquinas de Estado Finito

Exemplo 2.1

Current state ↓	----- Input -----		
	Dime	Quarter	Reset
S ₀₀	S ₁₀	S ₂₅	S ₀₀
S ₁₀	S ₂₀	S ₃₅	S ₀₀
S ₂₀	S ₃₀	S ₃₅	S ₀₀
S ₂₅	S ₃₅	S ₃₅	S ₀₀
S ₃₀	S ₃₅	S ₃₅	S ₀₀
S ₃₅	S ₃₅	S ₃₅	S ₀₀
Next state			

S₀₀ is the initial state
S₃₅ is the final state

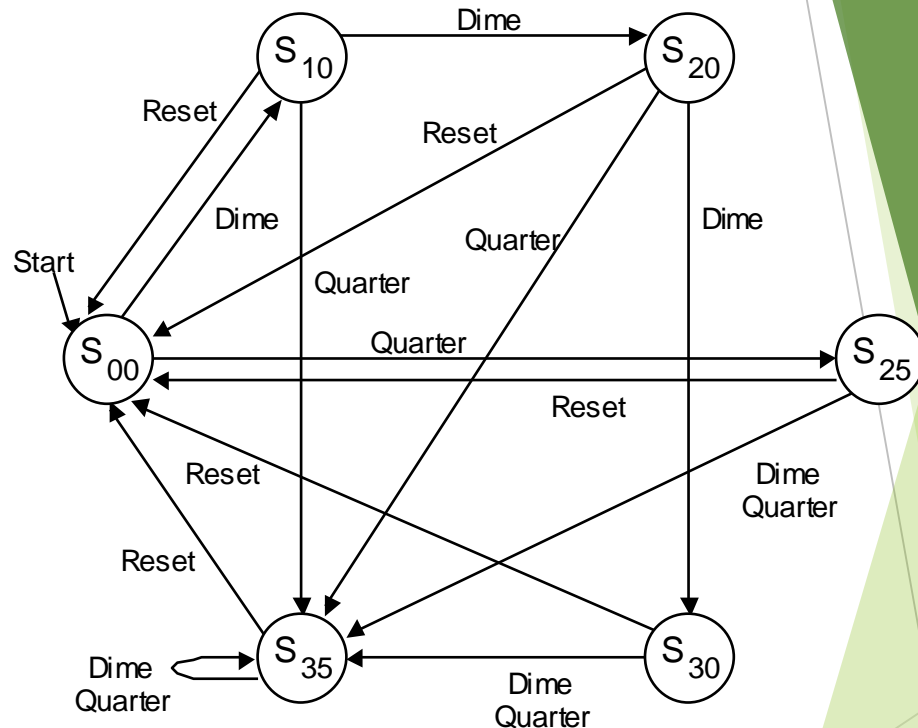


Figura 2.4 Tabela de estados e diagram de estados para a unidade de recepção de moedas de uma máquina de venda.

Implementação de uma Máquina Sequencial

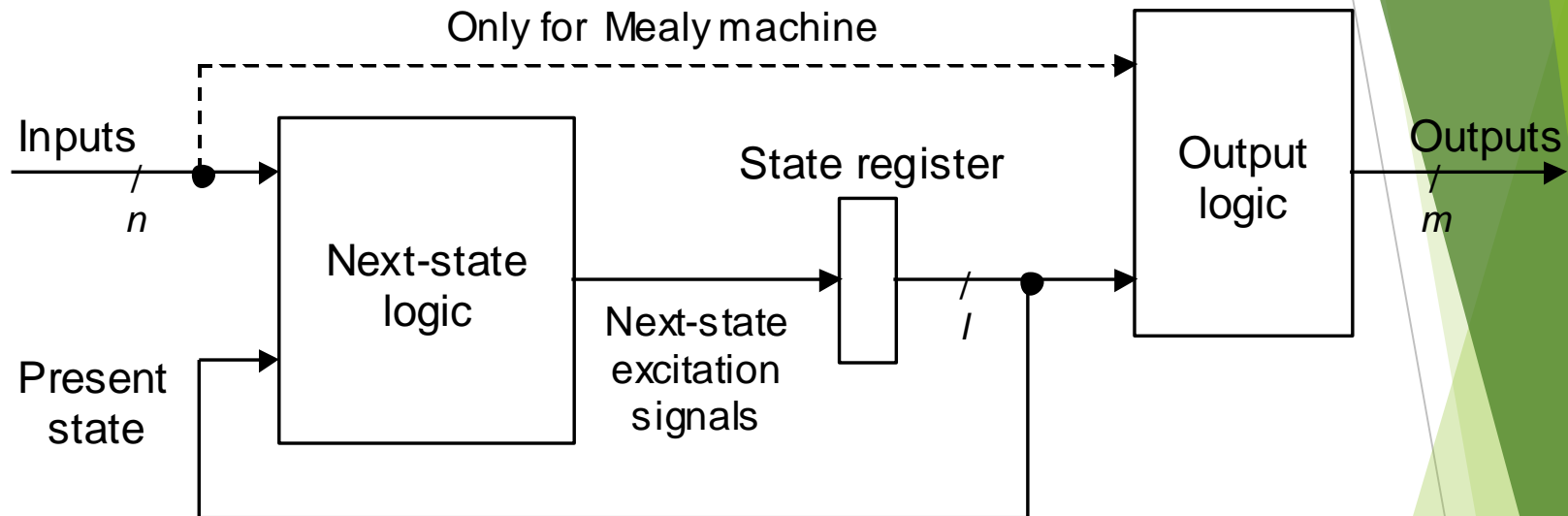


Figura 2.5 Implementação em hardware de máquinas sequenciais de Moore e Mealy.

ciais

Output
e

O estado
final é 1XX

idade de recepção

8
BParhami

Exemplo 2.3

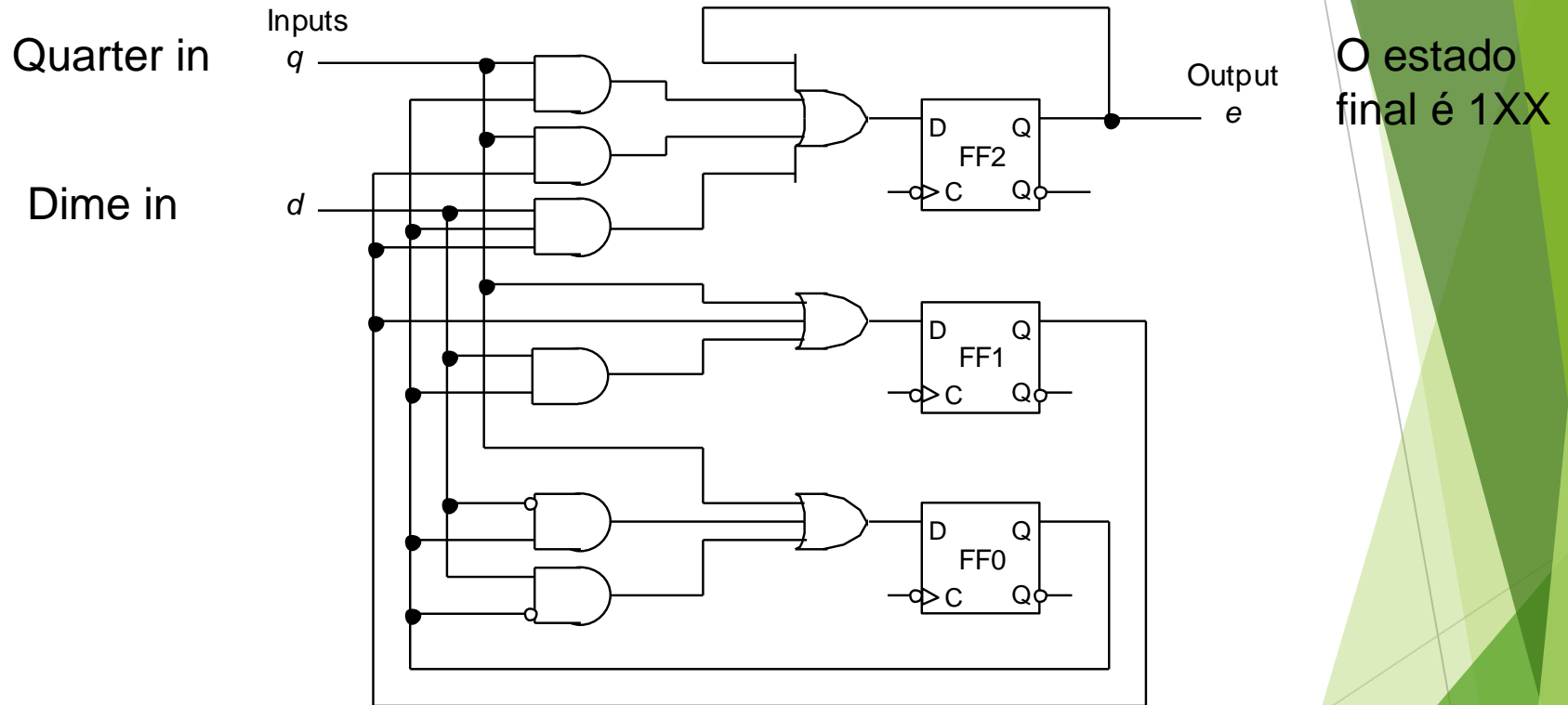


Figura 2.7 Implementação em hardware de uma unidade de recepção de moedas (Exemplo 2.3).

2.4 Elementos Sequenciais Úteis

- Blocos de construção de alto nível
 - Assemelham-se a partes pré-fabricadas de uma casa
 - Outros componentes de memória serão detalhados no capítulo 17 (SRAM, DRAM e Flash)
 - Neste capítulo serão descritos os seguintes elementos: registrador de deslocamento, arquivo de registradores (base da SRAM), contador

Registrador de Deslocamento

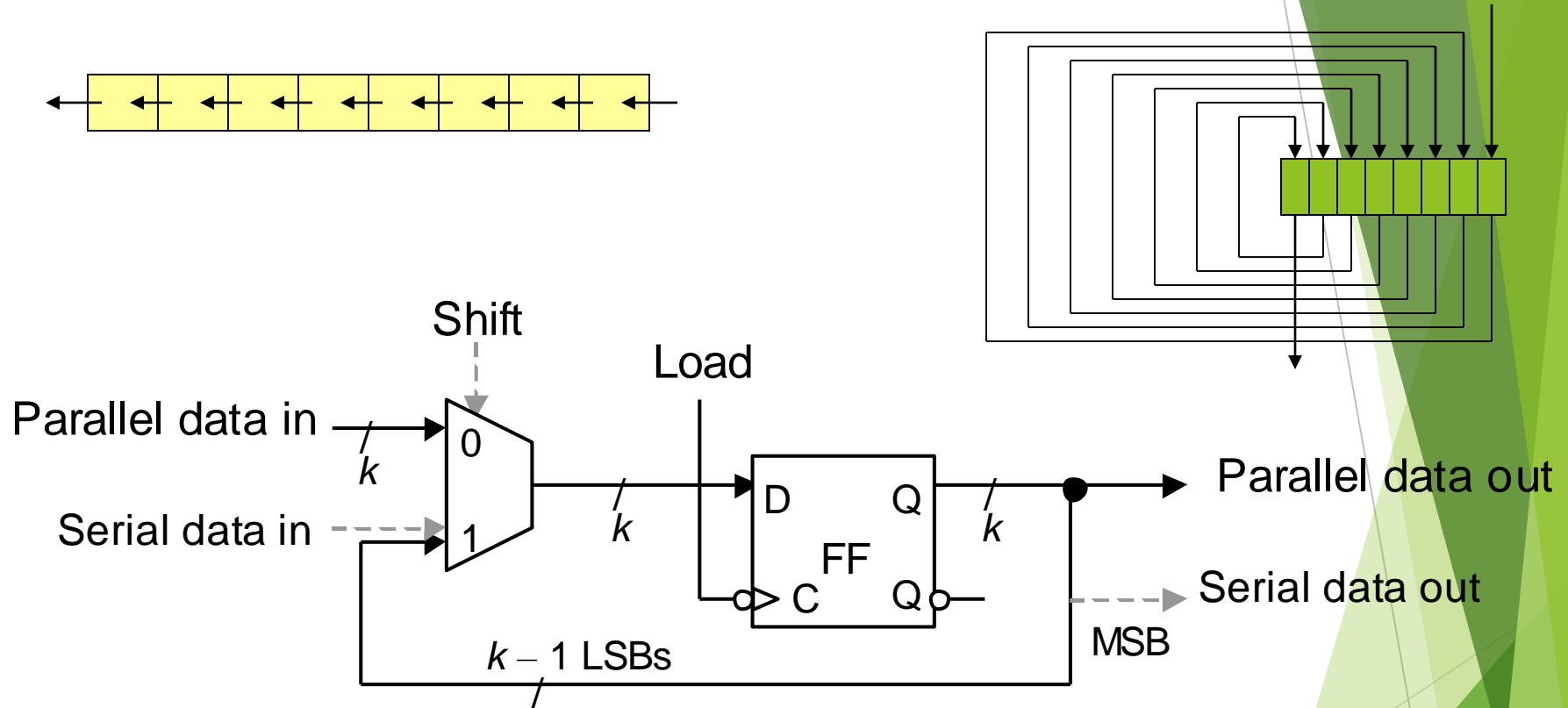
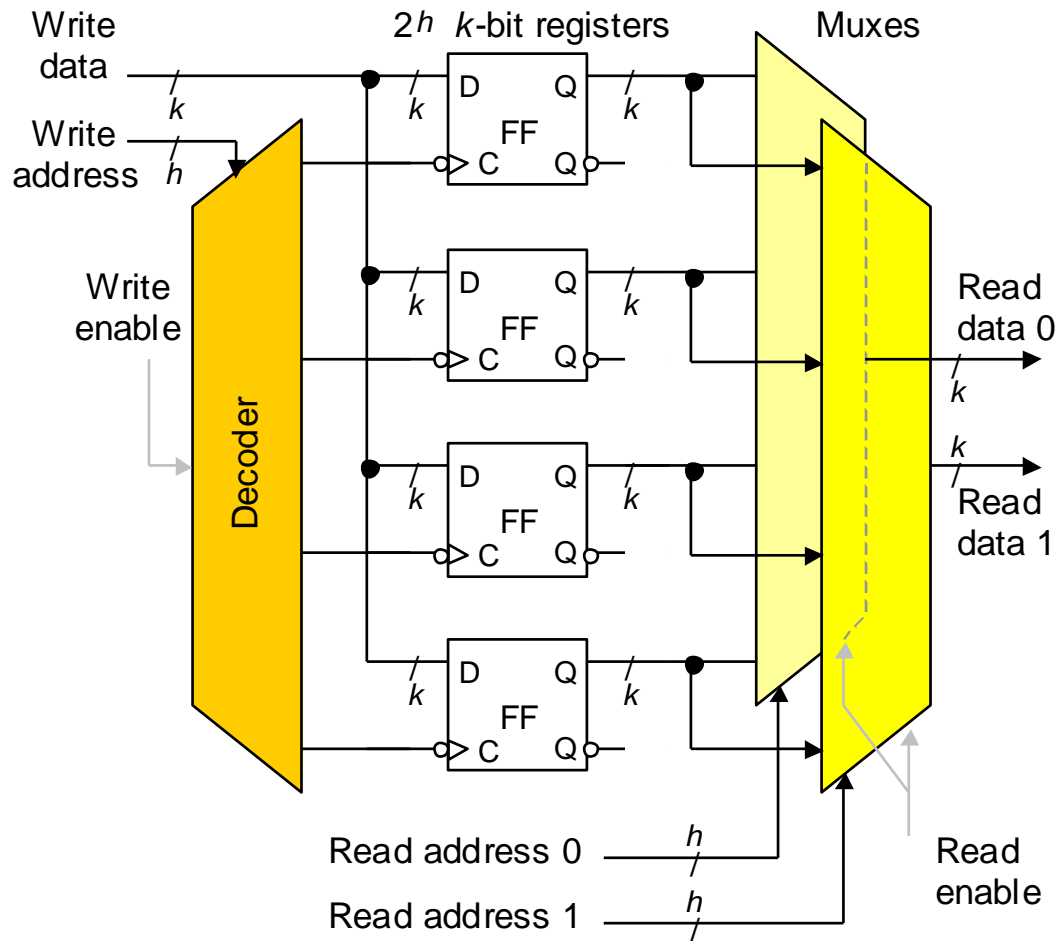
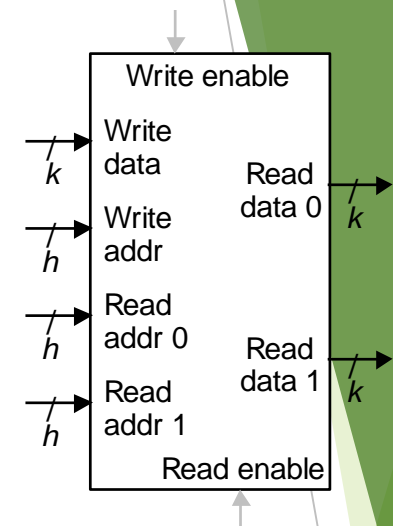


Figura 2.8 Registrador de deslocamento à esquerda de um único bit e capacidade de carga paralela. Para deslocamento lógico à esquerda, a linha de dado de entrada serial é conectada ao 0.

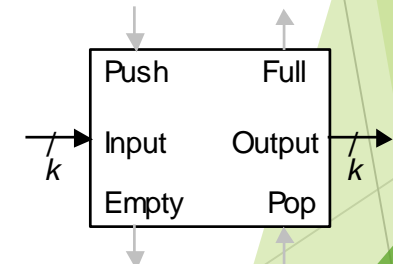
Arquivo de Registradores e FIFO



(a) Register file with random access



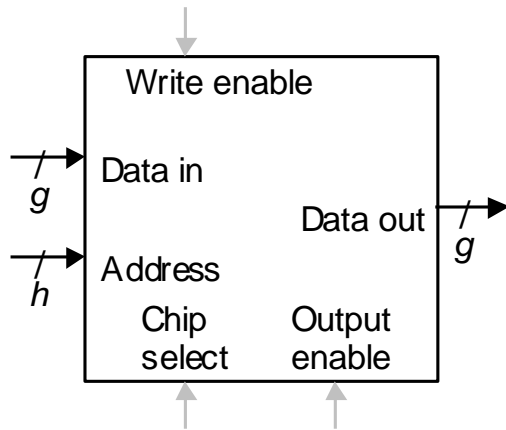
(b) Graphic symbol for register file



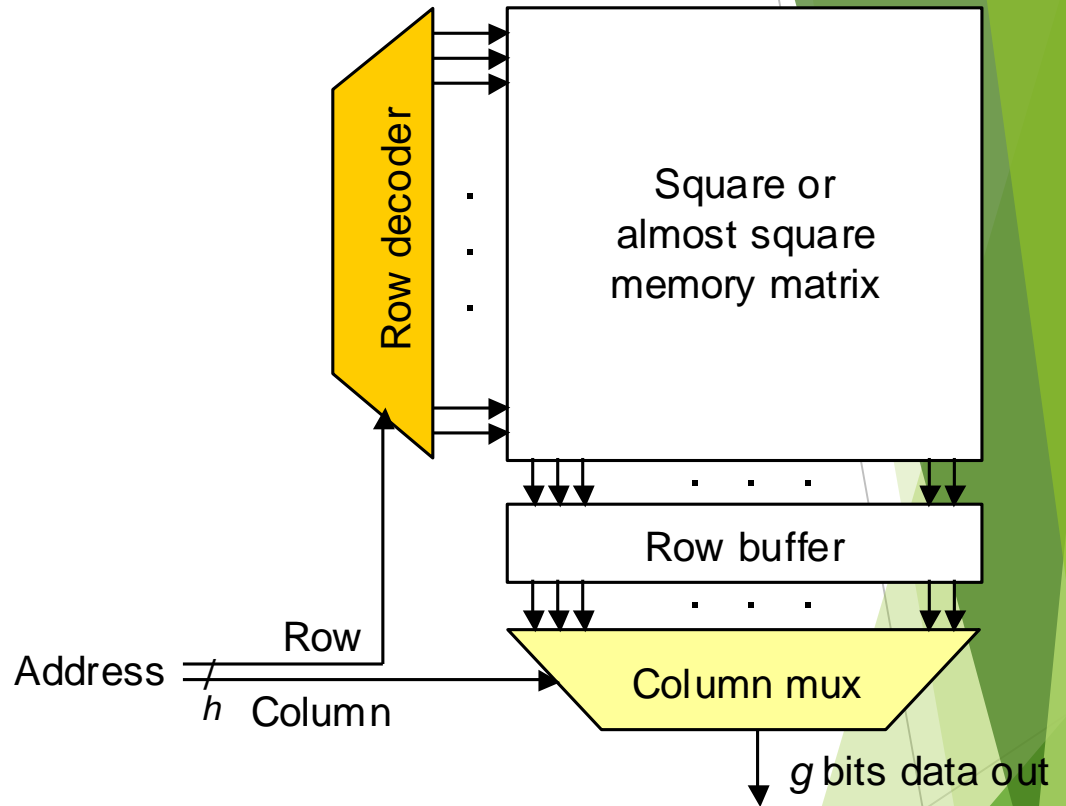
(c) FIFO symbol

Figura 2.9 Arquivo de registradores com acesso aleatório e FIFO.

SRAM



(a) SRAM block diagram



(b) SRAM read mechanism

Figura 2.10 A memória SRAM é simplesmente um arquivo de registradores grande com uma única porta.

Contador Binário

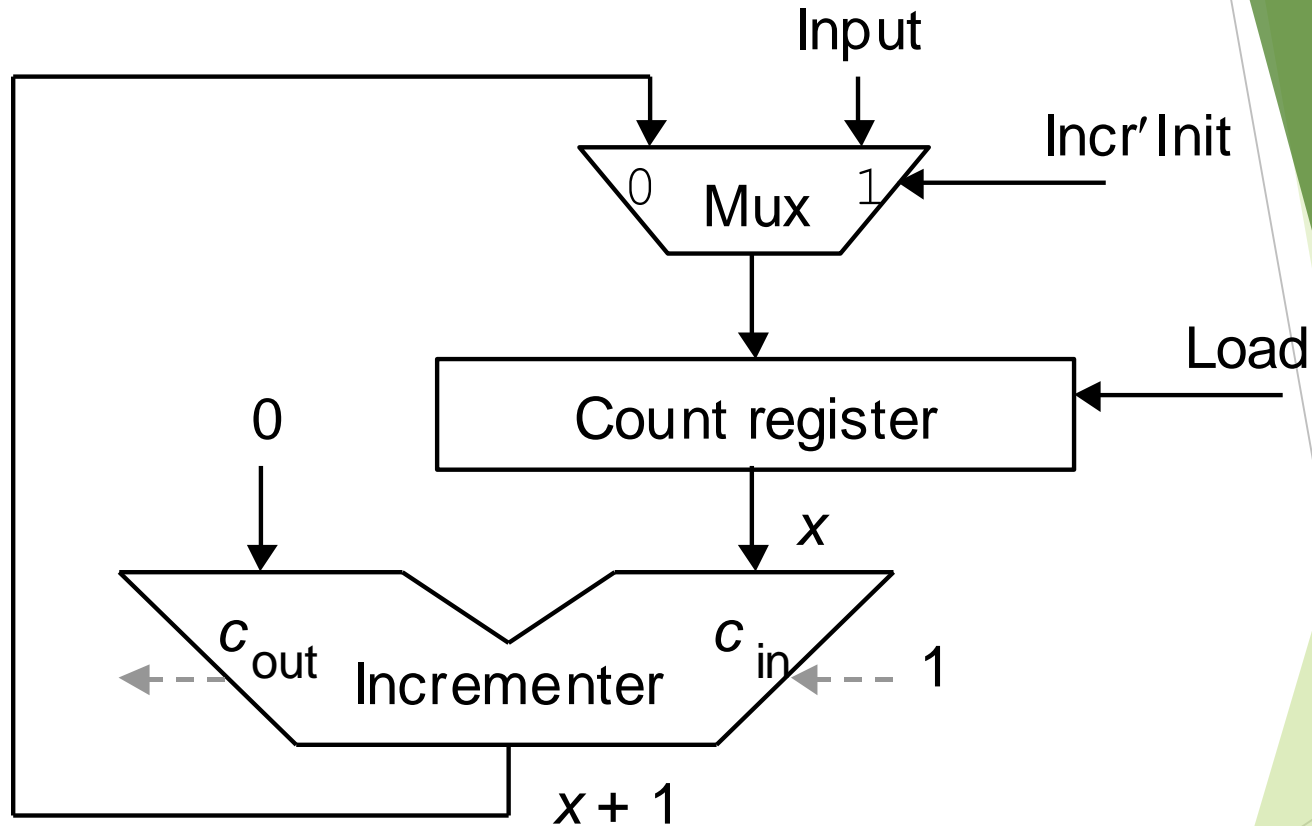


Figura 2.11 Contador binário síncrono com capacidade de inicialização.

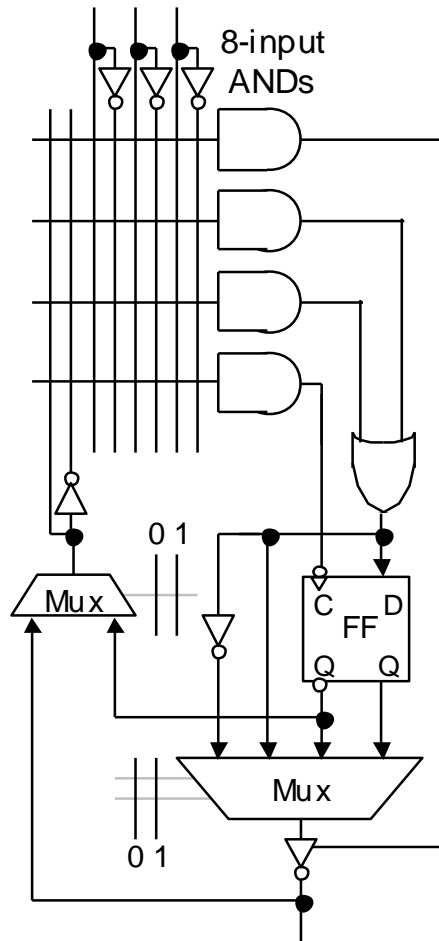
2.5 Elementos Sequenciais Programáveis

Um elemento sequencial programável contém portas e elementos de memória

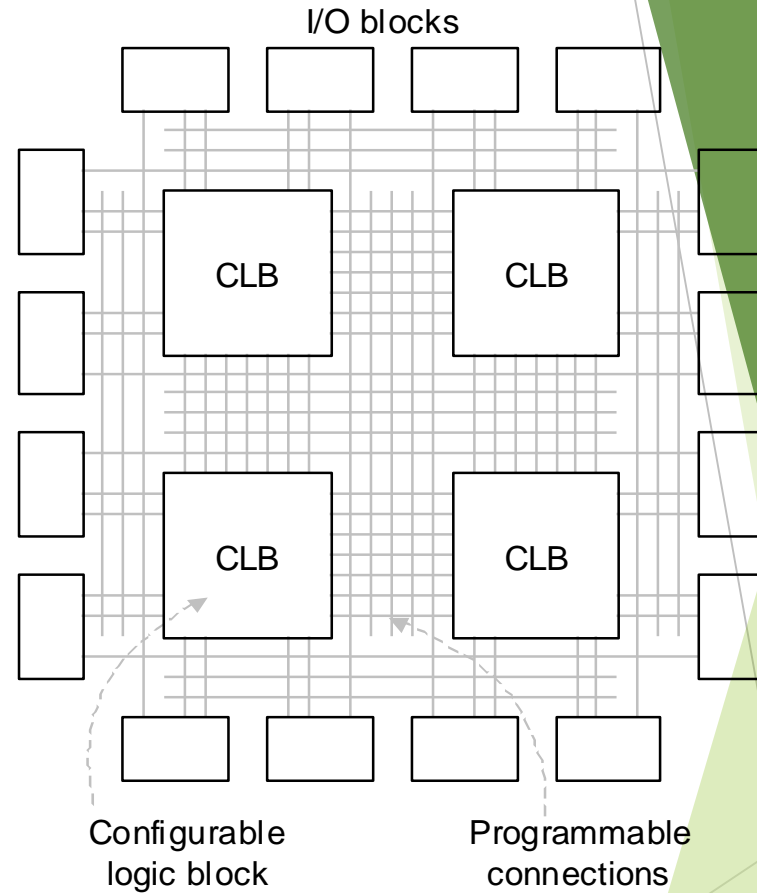
São programados através do corte de conexões existentes (*fusíveis*) ou estabelecimento de novas conexões (*anti-fusíveis*)

- Programmable array logic (PAL)
- Field-programmable gate array (FPGA)
- Ambos possuem macrocélulas e interconexões

PAL e FPGA



(a) Portion of PAL with storable output



(b) Generic structure of an FPGA

Figura 2.12 Exemplos de lógica sequencial programável.

2.6 Clocks e Temporização de Eventos

O clock é um sinal periódico: taxa do clock = frequência do clock. O inverso da taxa do clock é o período do clock:

1 GHz \leftrightarrow 1 ns

Restrição: Período do clock $\geq t_{\text{prop}} + t_{\text{comb}} + t_{\text{setup}} + t_{\text{skew}}$

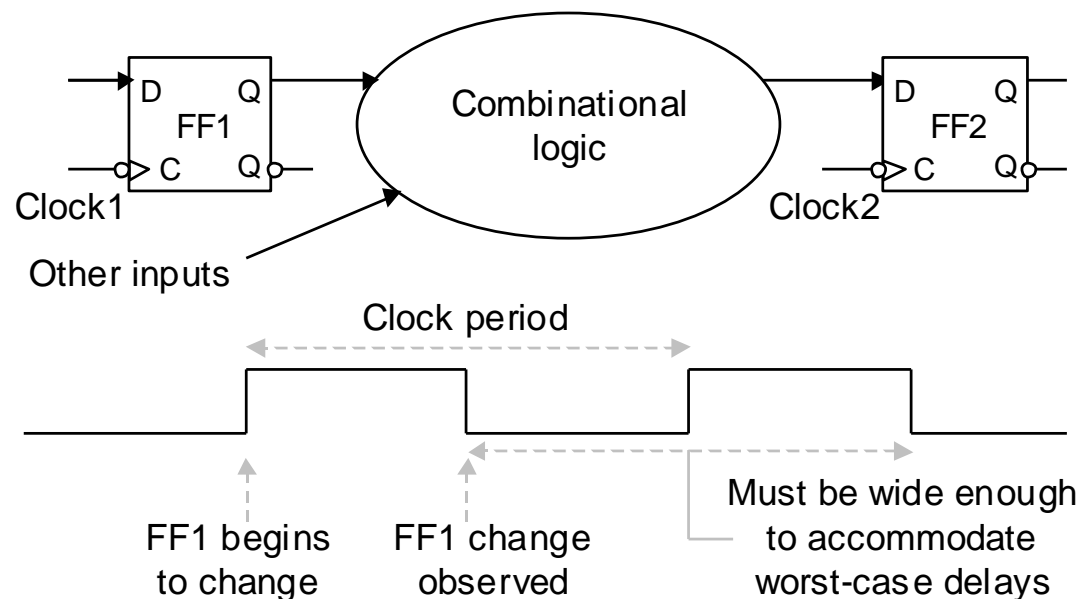
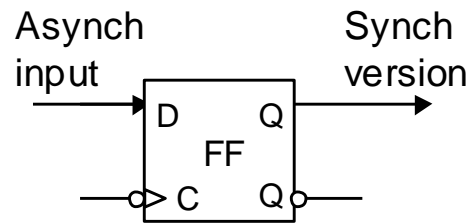
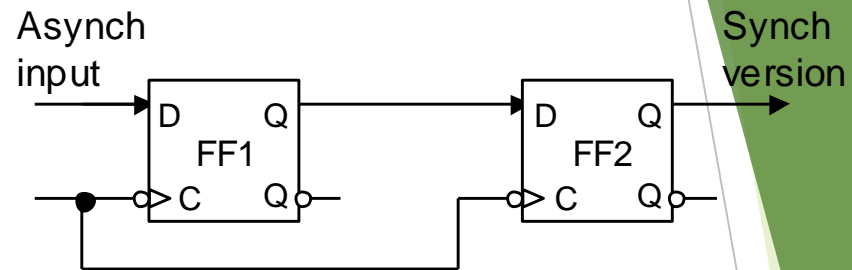


Figura 2.13 Determinando o tamanho necessário do período do clock.

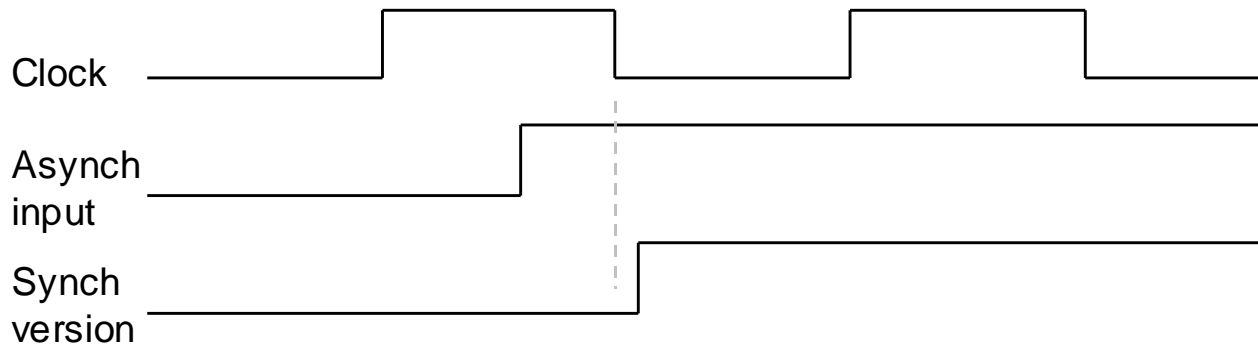
Sincronização



(a) Simple synchronizer



(b) Two-FF synchronizer



(c) Input and output waveforms

Figura 2.14 Sincronizadores são usados para prevenir problemas de temporização que podem ocorrer a partir de mudanças inesperadas em sinais assíncronos.

Operações Sensíveis ao Nível

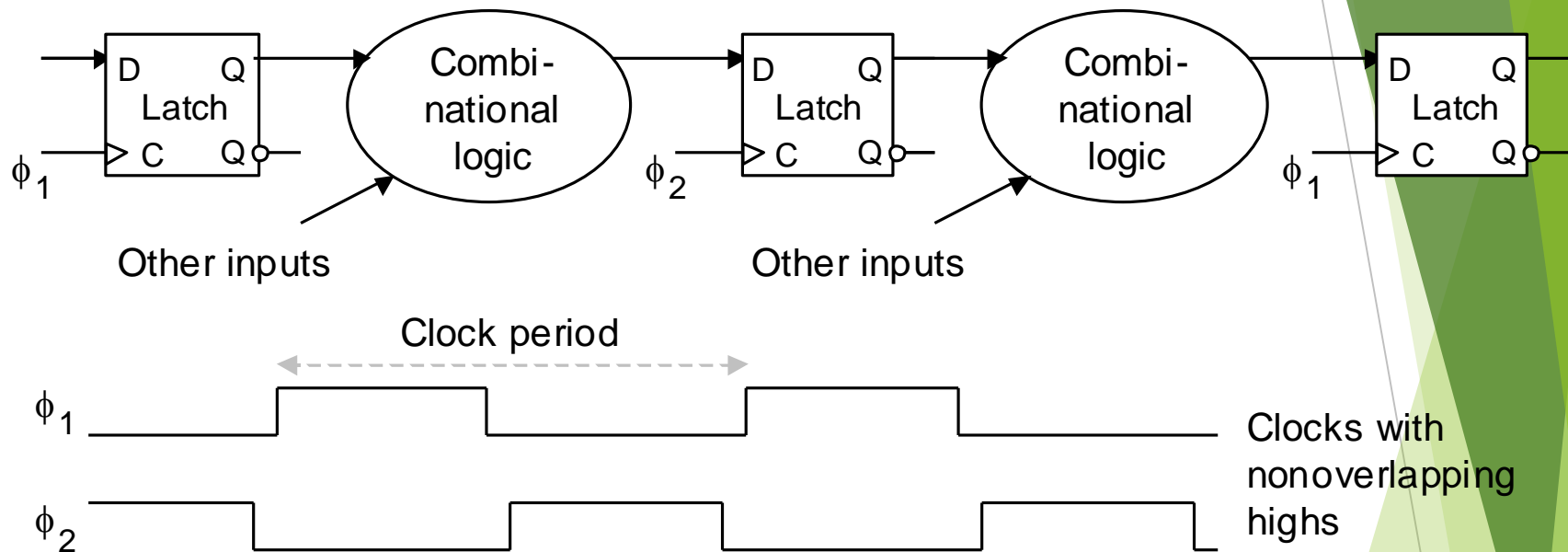


Figura 2.15 Temporização com clock de duas fases e sinais não sobrepostos.