

Sistemas Lógicos

(parte 3.1)

Dep.Armas e Electrónica- Escola Naval

V.2.0 V.Lobo 2000

Circuitos Sequenciais

Sistemas Lógicos (1)

- **Circuitos Combinatórios**
 - As saídas (em cada instante) dependem UNICAMENTE das entradas (presentes nesse instante)
 - Não têm memória
- **Circuitos Sequenciais**
 - As saídas dependem das entradas e do ESTADO do circuito
 - O ESTADO do circuito depende das entradas que teve em instantes anteriores
 - Têm MEMÓRIA

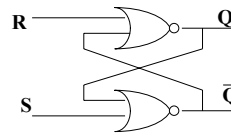
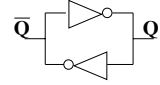


1

Elemento básico de memória

Sistemas Lógicos (1)

- Como memorizar um bit usando apenas portas lógicas ?
 - Um par de negações realimentadas mantêm o estado
 - Não é fácil mudar o estado memorizado
- **LATCH S-R (Set-Reset)**
 - A entrada S (Set) força a saída a 1
 - A entrada R (Reset) força a saída a 0



Posso descrever um LATCH SR através de uma tabela de verdade ?

2

Descrição de circuitos sequenciais

Sistemas Lógicos (1)

- **Tabelas de verdade**
 - O estado anterior conta como mais uma entrada
 - Pode-se abreviar a tabela de verdade, especificando a saída como função das entradas.

S	R	Q	Q _(t+1)
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	x
1	1	1	x

S	R	Q _(t+1)
0	0	Q
0	1	0
1	0	1
1	1	x

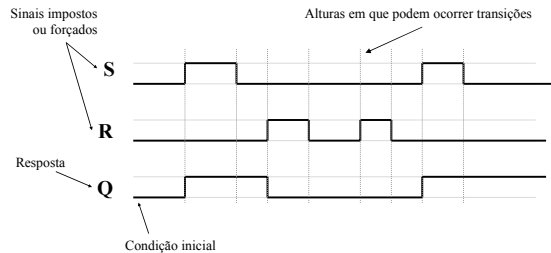
S,R	00	01	11	10
Q	0	0	x	1
Q-bar	1	0	x	1

3

Descrição de circuitos sequenciais

Sistemas Lógicos (1)

- **Diagramas temporais**
 - Mostram o comportamento no tempo dos circuitos, quando pomos à entrada um determinado sinal
 - Não definem completamente o circuito (são só a representação de uma situação possível)



4

Descrição de circuitos sequenciais

Sistemas Lógicos (1)

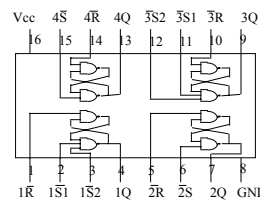
- **Circuitos Assíncronos**
 - Assim que as entradas mudam, as saídas podem mudar
 - Podem ter reacções muito rápidas, mas são (hoje) pouco usados por serem difíceis de projectar e sofrerem problemas com ruído e "corridas"
 - > Futuro promissor: viadé Proceedings of the IEEE, February 1999
- **Circuitos Síncronos**
 - Há um sinal de sincronismo (chamado CLOCK) que regula todas as transições
 - SÓ há transições nos FLANCOS DO CLOCK.
 - Nos diagramas temporais, basta analisar o que acontece nos flancos do CLK.
 - O ESTADO SEGUINTE é função apenas das entradas presentes ANTES do flanco do CLK !
 - Após um flanco de CLK, em que potencialmente há uma série de sinais a mudar, há um período de "relaxação" em que os sinais estabilizam antes de vir o CLK seguinte.
 - São os mais usados hoje em dia

5

Latches RS

Sistemas Lógicos (1)

- **Na família TTL**
 - 74279

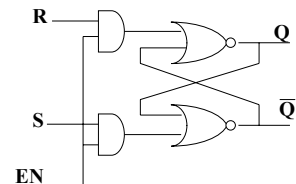


74279

Nota: Todas as entradas são activas a 0

- **GATED LATCH**

- Tem um sinal de ENABLE
- Quando o ENABLE é 0, mantêm o estado anterior. Quando é 1, reage às entradas S e R.



6

Sistemas Lógicos

(parte 3.1)

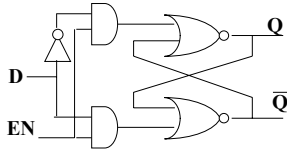
Dep.Armas e Electrónica- Escola Naval
V.2.0 V.Lobo 2000

LATCH tipo D

Sistemas Lógicos (3)

- Tem apenas uma entrada D (de Delay), e um sinal de Enable.
- É o elemento básico de memória
 - A saída está sempre disponível
 - EN=1 ⇒ Guarda o que estiver na entrada
 - EN=0 ⇒ Mantém o que tinha guardado anteriormente

Este circuito é síncrono ?



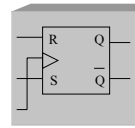
74373 tem 8 Latches

7

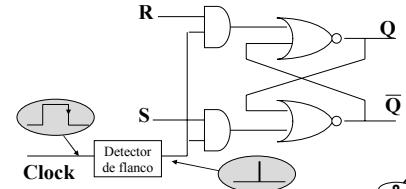
FLIP-FLOPs

Sistemas Lógicos (3)

- FLIP-FLOP
 - Só muda as saídas quando vem um flanco de CLOCK
 - Mudam as saídas só NUM DOS FLANCOS de clock (ou são Flip-flops de flanco ASCENDENTE, ou Flip-flops de flanco DESCENDENTE)
- FLIP-FLOP EDGE TRIGGERED
 - São sensíveis às entradas que têm no momento em que vem o CLK
 - A "Janela de sensibilidade" é muito estreita



Significa Edge-Triggered



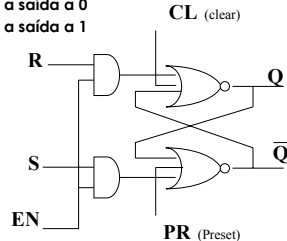
8

Flip-Flop Edge-Triggered

Sistemas Lógicos (3)

- Um flip-flop pode ter ENTRADAS ASSÍNCRONAS, que actuam independentes do CLK

- CLEAR - força a saída a 0
- PRESET - força a saída a 1



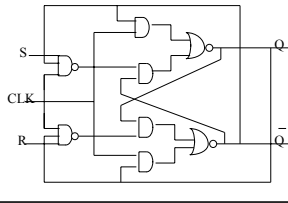
- São usadas normalmente para fazer RESETS ou impôr condições especiais (por exemplo, estados iniciais)

9

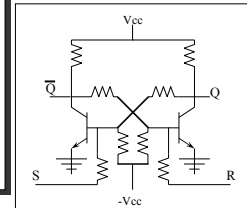
Outras implementações

Sistemas Lógicos (3)

Uma montagem alternativa para a implementação um circuito Edge-Triggered



- O modo concreto de implementar um circuito pode variar...



Implementação física de um LATCH RS (usando apenas 2 transistores)

10

Tipos de entradas : D,RS,JK,T

Sistemas Lógicos (3)

- Flip-flops TIPO D (delay)
 - Memorizam e atrasam a entrada
- Flip-flops TIPO RS (Set-Reset)
 - São "ligados" e "desligados" por S e R
- Flip-Flops TIPO JK
 - Quase iguais aos RS, mas resolvem a ambiguidade de R=S=1, fazendo neste caso um TOGGLE (invertendo o estado anterior)
 - O J faz Set, e o K faz Reset
 - Os mais usados na prática
- Flip-Flops TIPO T (Toggle)
 - Invertem o estado quando a entrada é 1

D	Q
0	0
1	1

S	R	Q
0	0	Q
0	1	0
1	0	1
1	1	?

J	K	Q
0	0	Q
0	1	0
1	0	1
1	1	Q-bar

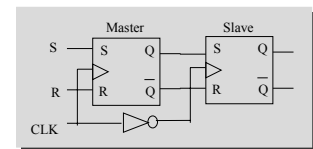
T	Q
0	Q
1	Q-bar

11

FLIP-FLOP Master-Slave

Sistemas Lógicos (3)

- Memoriza o que acontece quando o CLK está activo, mas só produz efeitos quando vem o flanco
- Tem uma concepção mais simples que os Edge-Triggered
- É composto por dois LATCHES em cadeia
 - Quando um está "transparente", o outro está "fechado"
 - Nunca há um caminho directo entre a entrada e saída
 - O Primeiro (Mestre) está ligado à entrada, e fornece dados ao Segundo (o Escravo)
- Tem "1's catching"
 - "Apanha" os picos



12

Sistemas Lógicos

(parte 3.1)

Dep.Armas e Electrónica- Escola Naval
V.2.0 V.Lobo 2000

Problemas simples

Sistemas Lógicos (1)

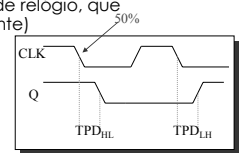
- Num automóvel, um dos pisca-piscas tem uma frequência de 2Hz. Pretende-se diminuir essa frequência para 0,5Hz. Projecte o circuito diminui a frequência do pisca-pisca.
- Pretende-se desenhar um alarme contra ladrões, que tenha uns sensores (por hipótese uns feixes de laser infra-vermelho) que enviam um sinal 1 quando não está ninguém presente, e 0 quando está (quando está alguém presente o feixe é interrompido). Existem 4 desses sensores, e quando qualquer um deles é activado, o alarme deverá começar a tocar. O alarme só deverá parar quando alguém premir o botão de "reset" que se encontra na sua caixa. Projecte este sistema.
- Desenhe um sistema para actuar sobre semáforos que põe o semáforo encarnado quando recebe um 1 e verde quando recebe um 0. Nas transições de verde para vermelho, E VICE-VERSA, deverá 2 segundos no laranja.

13

Flip-Flops - Temporizações

Sistemas Lógicos (1)

- Os flip-flops podem ser sensíveis ao flanco **Ascendente ou Descendente** (neste caso representados com uma bola no sinal de relógio, que significa que este é negado internamente)

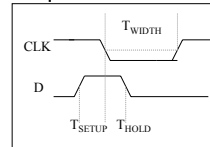


- Propagation Delay

- Tempo entre o clock e a reacção do integrado
- Pode ser diferente para transições H-L ou L-H

- Set-Up Time

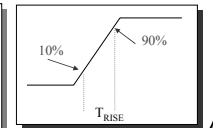
- Antecedência com que é necessário actuar nas entradas



- Hold Time

- Rise Time

- Pulse Width



14

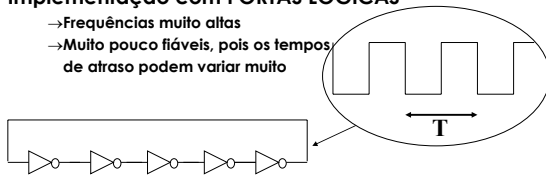
Temporizadores

Sistemas Lógicos (1)

- Geradores de Clock ou **Multivibradores Astáveis**
 - Geram ondas quadradas com uma dada frequência

- Implementação com PORTAS LÓGICAS

- >Frequências muito altas
- >Muito pouco fiáveis, pois os tempos de atraso podem variar muito



$$T \cong 2 \times n \times t_{pd}$$

$$f = \frac{1}{\text{Período}}$$

$$5 \text{ gates com } t_{pd} = 9\text{ns} \Rightarrow f = 11\text{MHz}$$

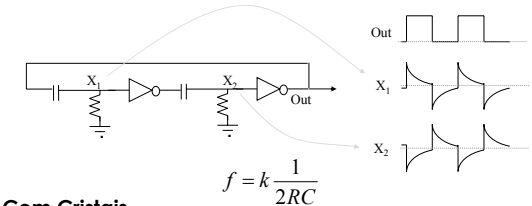
15

Temporizadores

Sistemas Lógicos (1)

- Com Circuitos RC e portas lógicas

- Facilmente ajustáveis
- Pouco precisos (dependentes das tolerâncias das resistências e condensadores)



- Com Cristais

- Muito precisos
- Só para frequências razoavelmente altas
- Normalmente usam-se integrados dedicados para gerar o clock a partir de um cristal de quartzo

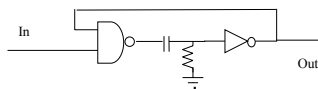
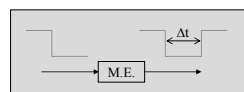
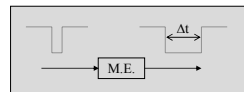
16

Temporizadores Monoestáveis

Sistemas Lógicos (1)

- Mono-estáveis

- Também chamados "one-shot"
- Geram pulsos de **largura fixa**
- Podem ser usados para
 - >Gerar atrasos fixos
 - >"Rectificar" picos



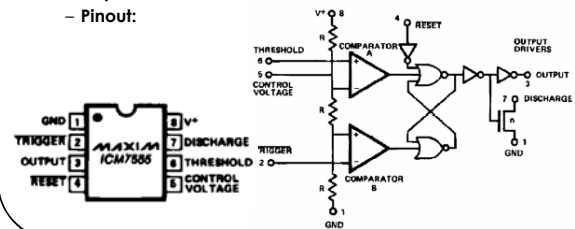
17

Integrados para temporizadores dedicados

Sistemas Lógicos (1)

- 555

- Integrado muito flexível para circuitos temporizadores
- Pode ser usado como monoestável ou gerador de clock
- O "duty-cycle" pode ser regulado
- É muito usado, e há bastantes "clones".
- Esquema:
- Pinout:



18

Sistemas Lógicos

(parte 3.1)

Dep.Armas e Electrónica- Escola Naval
V.2.0 V.Lobo 2000

Integrados para temporizadores dedicados

Montagens típicas de 555

A) Clock com duty-cycle de 50%

$$f = 1 / (1,4 RC)$$

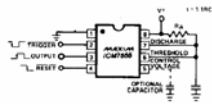
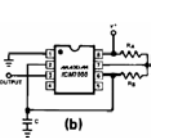
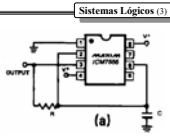
B) Clock com duty-cycle variável

$$f = 1,46 / ((Ra+2Rb) C)$$

$$d = Rb / (Ra+2Rb)$$

C) Monoestável

$$T_{pulsos} = 1,1 RC$$

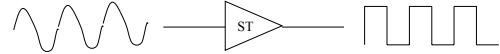


19

Schmitt-Triggers

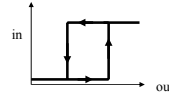
Podem receber sinais analógicos à entrada

- Convertem sinais analógicos em sinais digitais com 2 níveis apenas
- São usados para regenerar sinais digitais



Têm um ciclo de histerese

- Para passar de 0 para 1 é necessário que a entrada vá francamente a 1, mas para voltar a 0 tem que ir francamente a 0.
- São imunes a ruído de "cross-over"



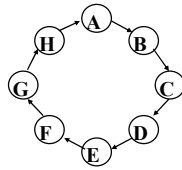
Que tipos de problemas podem ser resolvidos com estes circuitos?

20

Contadores

O que é um contador?

- É um sistema que passa ciclicamente por uma série de estados
- Exemplos de sequências:
 - 0,1,2,3,,0,1,2,3,0,1,2,3...
 - 9,7,5,3,1,9,7,5,3...
- Contador de binário natural
 - Corresponde ao nosso conceito "intuitivo de contador"
 - Conta 0,1,2,3...(2ⁿ-1)
- Contador em anel
 - Conta 100,010,001...100,010....



Módulo de um contador

- N° de estados que atravessa antes de voltar ao inicial

21

Contador em Anel

N flip-flops, dos quais um está a 1 e os outros a 0

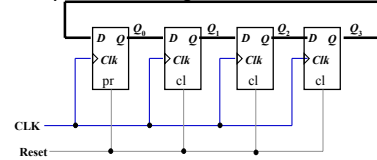
O flip-flop activo passa o seu 1 ao flip-flop seguinte

- É um shift-register com a saída ligada à entrada

N flip-flops ⇒ N estados diferentes

Exemplo de um contador em anel de 4 bits

- Sequência de contagem:



1000
0100
0010
0001
1000
0100
...

22

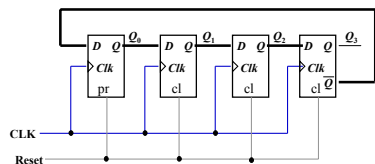
Contador Johnson

Parecido com o contador em anel, mas realimenta-se a negação do último bit (em vez de realimentar o último bit)

N flip-flops ⇒ 2N estados diferentes

Exemplo de um contador Johnson de 4 bits

- Sequência de contagem:



1000
1100
1110
1111
0111
0011
0001
0000
1000
1100
...

23

Contador Binário Síncrono (mód 2ⁿ)

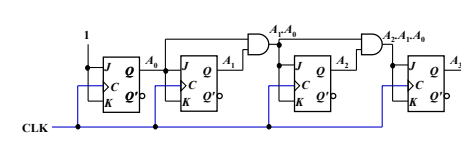
Conta em binário natural

- O bit menos significativo faz sempre toggle
- O bit mais significativo só faz toggle se todos os menos significativos forem 1

N flip-flops ⇒ 2^N estados diferentes

Exemplo de um contador binário síncrono de 4 bits

- Sequência de contagem:



A₃A₂A₁A₀
0000
0001
0010
0011
0100
0101
0110
0111
1000
1001
1010
1011
1100
1101
1110
1111
0000
0001

24

Sistemas Lógicos

(parte 3.1)

Dep.Armas e Electrónica- Escola Naval
V.2.0 V.Lobo 2000

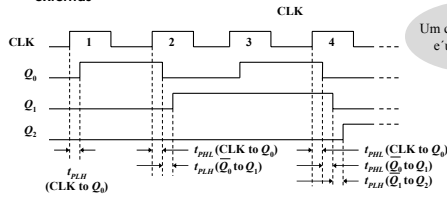
Contador Binário Assíncrono (ripple-mód 2ⁿ)

O clock não é comum a todos os FF

- A saída de um FF serve de clock ao FF seguinte
- Há ligeiros atrasos na mudanças dos diversos bits

São mais fáceis de construir

- Não necessitam de gates externas



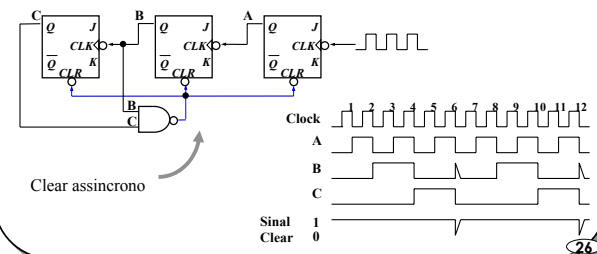
Nota: Um contador binário é um divisor de frequência

25

Contadores binários de módulo arbitrário

Ideia chave → Fazer reset ao contador para voltar o início

- Detectar o primeiro estado "não desejado", e fazer um clear assíncrono
- Gates AND
- O estado "não desejado" existe duante um curto espaço de tempo



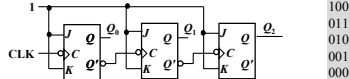
26

Contadores UP/DOWN

Contador para baixo (DOWN)

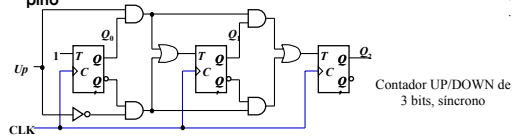
- Basta usar NOT(Q) em vez de Q

Contador DOWN de 3 bits, assíncrono



Contadores UP/DOWN

- Podem contar para cima (0,1,2,...) ou para baixo (9,8,7,...)
- A função (UP ou DOWN) é normalmente controlada por um pino

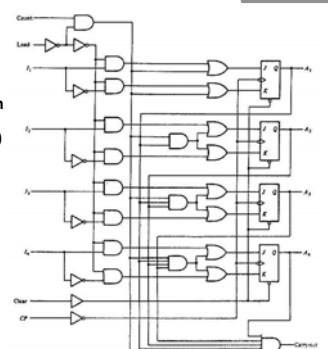
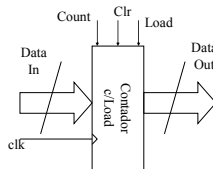


27

Contadores com carregamento paralelo

Contadores com carregamento paralelo

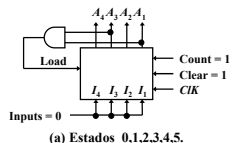
- Em vez de ter um reset a forçar o estado 0, podem ser inicializados (ou seja, carregados em paralelo) com qualquer valor



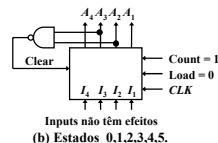
28

Um exemplo de um contador mód 6

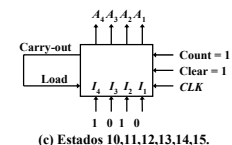
- 4 maneiras diferentes de o construir um contador mód 6 com um contador binário mód 16 com "parallel load" (síncrono) e clear assíncrono



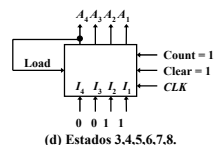
(a) Estados 0,1,2,3,4,5.



(b) Estados 0,1,2,3,4,5.



(c) Estados 10,11,12,13,14,15.



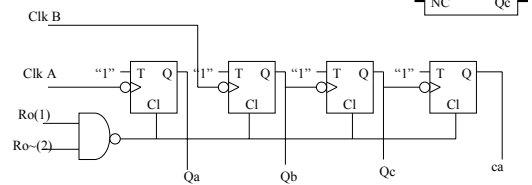
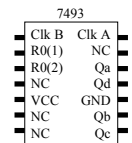
(d) Estados 3,4,5,6,7,8.

29

Contadores - Integrados disponíveis

7493

- Contador binário assíncrono (ripple-counter) de 4 bits
- 1º bit é independente dos outros
- Tem de ser ligado externamente



30

Sistemas Lógicos

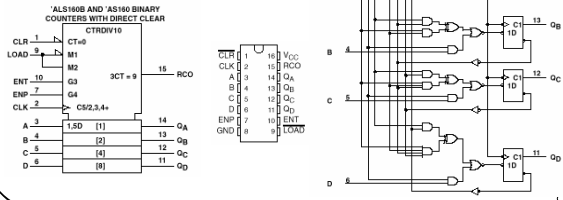
(parte 3.1)

Dep.Armas e Electrónica- Escola Naval
V.2.0 V.Lobo 2000

Contadores - Integrados disponíveis

Sistemas Lógicos (1)

- **74163**
 - Contador binário síncrono de 4 bits, com pre-load e enable
- **Outros**
 - **74160** - Contador BCD síncrono
 - **74190** - Contador BCD up/down síncrono com LOAD

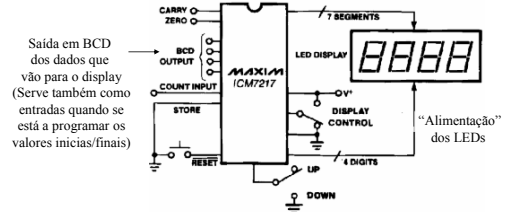


31

Contadores - Integrados mais complexos

Sistemas Lógicos (1)

- **Maxim 7217**
 - Contador up/down de 4 dígitos decimais
 - Saídas directas para 4 displays de 7 segmentos (anodo ou cátodo, freq. variável)
 - Valor de início e fim de contagem programável
 - Saída carry para ligar em cascata
 - Permite circuitos muito compactos

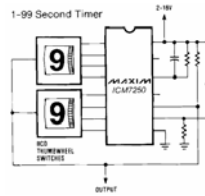


32

Contadores - Integrados mais complexos

Sistemas Lógicos (1)

- **7250**
 - Temporizador programável digitalmente de 0-99
 - Unidade de atraso definida um circuito RC
 - Pode gerar temporizações desde microsegundos até dias



Exercícios sobre contadores

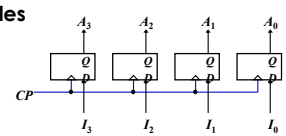
Por uma questão de segurança, pretende-se impedir que entrem mais de 230 pessoas nos caçilheiros. A entrada para o átrio há umas barras que contam as pessoas que aí passam (saída COUNT), e que podem ser bloqueadas (através da entrada BLOCK). As barras deverão bloquear quando tiverem entrado 230 pessoas, e só deverão ser desbloqueadas quando o caçilheiro partir. Quando o caçilheiro partir é também necessário reinicializar o contador. Para detectar a partida, há um sensor no portão de acesso ao barco que envia um pulso a 1 quando o portão fecha. Projecte este sistema.

33

Registos

Sistemas Lógicos (1)

- Como guardar informação que ocupe mais que 1 bit ?
 - Usar um conjunto de Flip-flops
 - É necessário dar alguma ordem a esse conjunto de Flip-flops
- **Registos**
 - São basicamente uma colecção de Flip-flops
 - Diferem no modo como os bits são "carregados" para dentro dos Flip-flops, no modo como são "passados para fora", e, eventualmente no modo como os bits passam de um FF para outro
- **Registo paralelo simples**
 - Podem fazer operações Parallel-in/Parallel-out

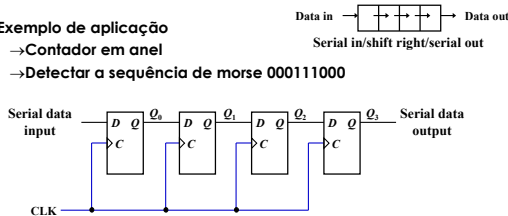


34

Registo de deslocamento

Sistemas Lógicos (1)

- **Registo de deslocamento (shift-registers)**
 - São malhas de atraso
 - Têm uma entrada Serial-in e uma saída Serial out
 - Os dados internos são normalmente acessíveis
 - Pode ser usado para fazer conversões série/paralelo
 - Há registos que fazem shift right ou shift left
- Exemplo de aplicação
 - Contador em anel
 - Detectar a sequência de morse 000111000



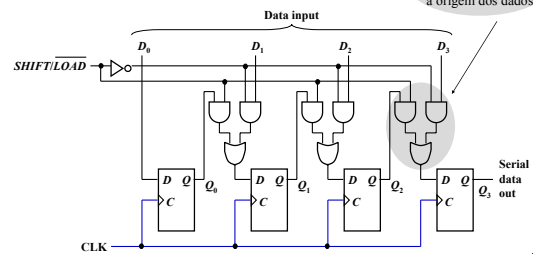
35

Registos multifunção

Sistemas Lógicos (1)

- **Registos parallel-in/serial-out**
 - Conversões paralelo/serie
 - Tem uma entrada que selecciona a função a realizar:
 - Fazer um deslocamento (shift) dos dados
 - Carregar um novo dado

Isto é basicamente um MUX que selecciona a origem dos dados



36