



**Laboratório
Circuitos Digitais II**

Prof. Dr. Eng.
Fernando Passold

Last Updated: 30 de maio de 2023

“... acredito que a vida é uma coisa inteligente, que as coisas não acontecem por acaso.”
(Steve Jobs)

GOOD THINGS
COME TO THOSE
WHO WAIT

GOOD THINGS
COME TO THOSE
WHO WORK THEIR
ASSES OFF AND
NEVER GIVE UP

Este material foi desenvolvido para a disciplina de “Laboratório de Circuitos Digitais II” do curso de Engenharia Elétrica da UPF. Trata de experimentos práticos apenas na área de Eletrônica Digital Sequencial e Combinacional.

Este material foi criado usando software livre, multiplataforma e de domínio público: [LaTeX](#) (pacote [MiKTeX](#) / [MacTeX](#)) e editor [TeXstudio](#). Foi adotada a fonte Palatino.

G > ^ V
©Prof. Dr. Eng. Fernando Passold, 30 de maio de 2023.

Sumário

1 Biestável RS Básico com portas NAND	7
1.1 Objetivos	7
1.2 Fundamentos Teóricos	7
1.3 Parte Prática	7
1.4 Formulário à Completar	8
2 Latch e Flip-Flop tipo D.	11
2.1 Objetivos	11
2.2 Fundamentos Teóricos	11
2.3 Parte Prática	11
2.4 Questões	12
3 Contador Assíncrono usando Flip-Flop tipo D	15
3.1 Objetivos	15
3.2 Fundamentos Teóricos	15
3.3 Parte Prática	15
3.3.1 Primeira Montagem	16
3.3.2 Segunda Montagem	16
3.4 Questões	17
3.5 Sugestões	17
3.6 Anexo 1: Como usar Gerador de Sinais	19
3.7 Anexo 2: Gerador de Pulso de Clock	20
4 Contador Assíncrono Módulo-n com Reset Manual	23
4.1 Objetivos	23
4.2 Fundamentos Teóricos	23
4.3 Parte Prática	23
4.4 Questões	23
5 Contador Assíncrono $Up/Down$	27
5.1 Objetivos	27
5.2 Fundamentos Teóricos	27
5.3 Parte Prática	27

5.4	Questões	28
6	Divisor de frequência	29
6.1	Objetivos	29
6.2	Fundamentos Teóricos	29
6.3	Parte Prática	29
6.4	Questões	30
7	Aplicação de Ripple Counter	31
7.1	Objetivos	31
7.2	Fundamentos Teóricos	31
7.3	Parte Prática	31
7.4	Questões	32
8	O Contador Síncrono 74190	35
8.1	Objetivo	35
8.2	Fundamentos Teóricos	35
8.3	Parte Prática	36
8.4	Perguntas	38
9	Aplicação de Contador Síncrono 74190	43
9.1	Objetivo	43
9.2	Fundamentos Teóricos	43
9.3	Parte Prática	44
9.4	Perguntas	46
10	Máquina síncrona “Exótica”	47
10.1	Introdução	47
10.2	Projeto	47
10.3	Circuito (Parte Prática)	48
10.3.1	Módulo de Chaves	49
10.3.2	Interface de Saída	49
10.4	Formas de onda	50
10.5	Teste sugerido	50
11	Projeto Contador Síncrono [2022/1]	51
11.1	Introdução	51
11.2	Embasamento Teórico	51
11.3	Projeto	51
11.3.1	Análise do caso do estado "7"	53
11.4	Explorando Outras Soluções	54
11.5	Solução usando MUXes	54
11.6	Problema: estado inicial dos FF's	56

11.6.1 Solução à nível de Simulação	56
11.7 Possível solução para inicialização dos FF's	56
11.8 Circuito final com "Parallel Load"	56
12 Contador Johnson usando Reg. de Deslocamento 74LS164	61
12.1 Fundamentos Teóricos	61
12.2 Parte Prática	62
12.3 Questões	62
13 Contador em Anel usando Registrador de Deslocamento 74194	67
13.1 Fundamentos Teóricos	67
13.2 Parte Prática	68
13.3 Questões	68
14 Contador Johnson Bidirecional usando Registrador Bidirecional	73
14.1 Fundamentos Teóricos	73
14.2 Parte Prática	74
14.3 Perguntas	74
15 Circuito Transmissor/Receptor Serial Síncrono	77
15.1 Fundamentos Teóricos	77
15.1.1 Circuito Transmissor (TX)	77
15.1.2 Circuito Receptor (RX)	78
15.2 Parte Prática	78
15.3 Questões	78

1 | Biestável RS Básico com portas NAND

Listo de Material

- 1 CI 74LS00 (4 × NANDs (2));
- 2 led's;
- 2 resistores de 330Ω .

S(et)	R(eset)	Entradas	Saídas	
		$Q(t+1)$	$\bar{Q}(t+1)$	Comentários
0	0	$q(t)$	$\bar{q}(t)$	Mantém estado
1	0	1	0	Set
0	1	0	1	Reset
1	1	1	1	Não usar

Tabela 1.1: Tabela verdade do biestável RS.

1.1 Objetivos

Comprovar o padrão de comportamento de um circuito biestável básico do tipo RS (Set-Reset) baseado em portas NAND. Perceber que é um circuito capaz de memorizar estados.

1.2 Fundamentos Teóricos

Neste laboratório será explorada a forma como trabalha um circuito biestável digital do tipo RS (ou Set-Reset) baseado em portas NAND.

Um biestável é um circuito estável capaz de transitar entre 2 estados distintos. Notar que este circuito possui realimentações entre as suas saídas e suas entradas o que justamente o transforma num circuito capaz de assumir os 2 estados distintos: Setado (saída principal, Q em nível lógico ALTO) ou Resetado (saída principal, Q , em nível lógico BAIXO) – ver fig. 1.1. Para modificar estes 2 estados possíveis, este circuito possui uma entrada para realizar o Set, S e outra entrada para realizar o Reset, R . Este circuito compõe o bloco básico das futuras pastilhas Latchs ou Flip-Flops.

Porém, o bistável RS enfrenta de um ligeiro problema quando o usuário resolve ativar as suas 2 entradas ao mesmo tempo (comprovar em testes no laboratório o que sucede) – ver última linha da tabela verdade do biestável RS na tabela 1.1.

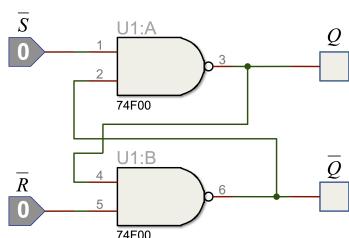


Figura 1.1: Diagrama elétrico do RS básico (portas NAND).

1.3 Parte Prática

Montar no proto-board o circuito mostrado na fig. 1.2, alimentar e comprovar a tabela verdade do biestável RS (para as 4 combinações possíveis de $R(t)$ e $S(t)$ e considerando que $Q(t)$ pode iniciar em 0 ou em 1, totalizando assim 8 situações diferentes).

Comprove o estado das 2 saídas quando as 2 entradas deste circuito são setadas. Lembrar que ambos os leds estão na configuração ATIVO BAIXO!

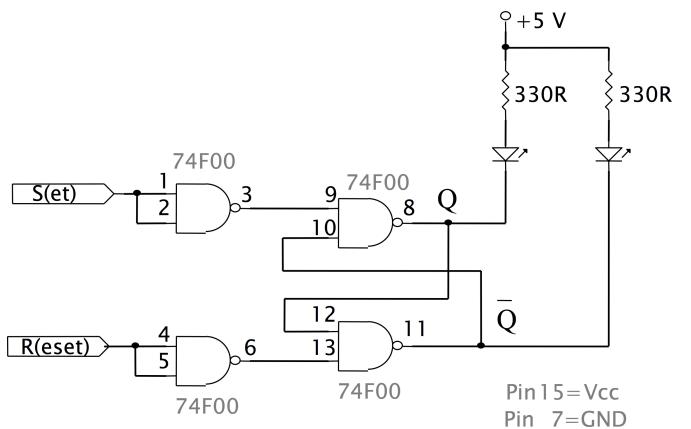


Figura 1.2: Montagem de biestável RS básico

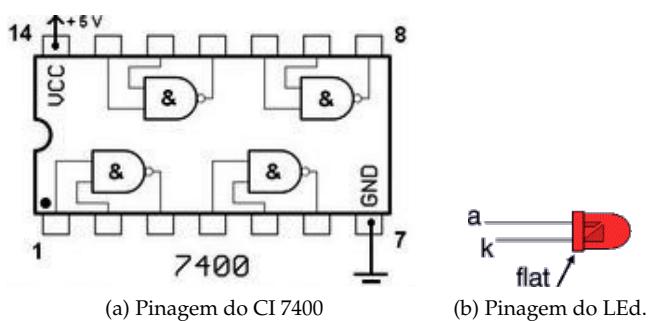


Figura 1.3: Componentes utilizados.

Observações:

- Como trata-se de um circuito sequencial, isto é, que **depende do seu estado anterior**, as 4 combinações possíveis de entrada devem ser realizadas levando-se em conta SEMPRE o estado atual (ou anterior) do circuito. Por exemplo: Se $q(t)$ se refere ao estado atual e $Q(t+1)$ se refere ao próximo estado do circuito, então se o circuito começou resetado, isto é, $q(t) = 0$ e as entradas foram modificadas para $S = 1$ e $R = 0$, o circuito deve terminar "setado" ($Q(t+1) = 1$). Se entretanto, para esta mesma configuração de entrada, isto é, $S = 1$ e $R = 0$, o circuito já iniciar setado ($q(t) = 1$), não será verificado nenhuma modificação na sua saída ($Q(t+1) = 1$), pois o circuito manterá a condição de setado.

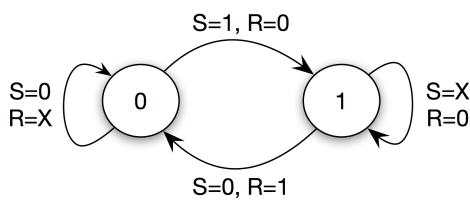


Figura 1.4: Diagrama de estados do biestável RS.

1.4 Formulário à Completar

Complete a tabela verdade 1.2 mostrada à seguir.

Note que é você quem deve indicar os níveis lógicos apresentados nas entradas S e R além de indicar o nível lógico presente nas saídas Q e \bar{Q} ANTES e DEPOIS, todas as vezes em que o nível lógico numa das entradas, R ou S for modificado.

Lembrar que os Leds mostrados na fig. 1.2 encontram-se na configuração ATIVO BAIXO.

Perceba que o circuito "não perdoa" erros, isto é, se a modificação dos níveis lógicos apresentados nas entradas R ou S foi feita de forma diferente da que estava prevista, o circuito reage em questão de nano-segundos, não sendo possível "voltar para trás". Neste caso, considere os novos níveis lógicos assumidos na saída do circuito como a nova condição inicial para o próximo teste (modificação de níveis lógicos apresentados nas entradas R ou S). Por esta razão, **não é sugerida** uma sequência de teste.

É o próprio estudante que determina a sua própria sequencia de testes, não esquecendo de anotar na tabela, a sequencia executada (ordem em que os níveis lógicos foram apresentados nas entradas e lidos nas saídas).

A título de exemplo, as linhas 0 e 1 da tabela 1.2 retratam um caso exemplo em que o circuito iniciou com ambas as entradas conectadas ao GND ("0") e por acaso iniciou resetado tão logo foi alimentado, ou seja: inciou com (estado atual): $q(t) = 0$ e $\bar{q}(t) = 1$ (notar pela tabela verdade do biestável-RS, que não ativar nenhuma das entradas, apenas significa que se pretende que as saídas do biestável mantenham o último estado assumido). Suponha que à seguir, as entradas são modificadas para $S = 1$ (ligação da entrada S ao $+V_{cc}$) e que a entrada R permaneceu em "0"; deve ser notado que imediatamente o circuito deve modificar sua saída para a condição de setado, ou seja, passa a assumir um novo estado (próximo estado): $Q(t+1) = 1$ e $\bar{Q}(t+1) = 0$. As próximas linhas da tabela devem ser completadas por cada estudante.

Espera-se que o estudante apresente no mínimo 8 transições (combinações) diferentes de níveis lógicos para S , R e $q(t)$ na tabela 1.

Atenção – Note que:

1. Não é possível modificar na prática, **ao mesmo tempo**, o nível lógico das 2 entradas, R e S . A não ser que as duas entradas sejam curto-circuitadas fisicamente no próprio proto-board.
2. Tente reproduzir na prática uma sequencia de entrada como: a) $SR = 00 \rightarrow SR = 11$, ou b) $SR = 10 \rightarrow SR = 01$. Tente repetir cada uma destas sequencias mais de 3 vezes e perceba que provavelmente não será obtida a mesma resposta (saída). Você saberia explicar por que isto acontece?
Dica: lembrar que qualquer porta lógica possui atrasos de propagação e que ele não é nulo, apesar de assumir um valor bem baixo. E lembrar que este atraso de propagação é muito mais baixo do que o período de tempo que você levaria para tentar alternar fisicamente e simultaneamente os níveis lógicos nas entradas S e R – o circuito responde muito mais rápido do que você é capaz de alterar seus níveis lógicos.
3. Será percebido que na prática o que se consegue realizar é seguir uma sequencia de código Gray para simulação de diferentes respostas para o RS.

Teste	Entradas		Estado atual		Próximo estado		Comentário
	S	R	$q(t)$	$\bar{q}(t)$	$Q(t+1)$	$\bar{Q}(t+1)$	
0	0	0	?	?	0	1	Inicia resetado
1	1	0	0	1	1	0	Set

Tabela 1.2: Tabela (formulário) à ser completado em sala de aula

2 | Latch e Flip-Flop tipo D.

Listar de Material

- 1 CI 74LS74 (Flip-Flop D duplo, borda de subida);
- 1 CI 74LS75 (Latch D de 4 bits, ativo alto);
- 2 led's;
- 2 resistores de 330Ω .
- 2 resistores de $10K\Omega$ (ou mesmo $100K\Omega$).

Biestáveis	Latch	Flip-Flop
Entrada de Controle	Enable	Clock
Sensibilidade	Nível lógico	Borda do sinal
C.I.:	74LS75 (alto)	74LS74 (subida ↑)

Tabela 2.1: Tabela comparativa Latch (CI 74LS75) × Flip-Flop (CI 74LS74).

2.1 Objetivos

- 1) Comprovar a tabela verdade e comportamentos distintos entre um Latch e Flip-Flop tipo D. Ambos são biestáveis e funcionam “copiando” o dado de entrada quando a pastilha é ativada, dai nome de “D” (Data).
- 2) Comprovar o efeito de “bounce” (ruído numa entrada de circuito digital sequencial) e testar uma solução para este problema.

2.2 Fundamentos Teóricos

Neste laboratório serão explorados os CIs: 7475 (um latch de 4 bits) e o 7474 (um flip-flop duplo). Suas pinagens aparecem na figura 2.1. Ambos são CIs biestáveis com diferentes tipos de entrada extra de controle, o que determina se são chamados de latches ou flip-flops e determina diferentes formas de comportamento – ver a tabela 2.1.

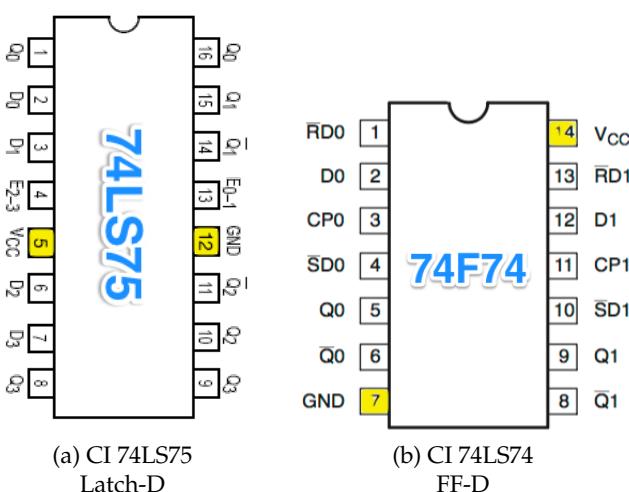


Figura 2.1: Pinagens dos CIs 74LS75 e 74LS74.

Note que num Latch, sua entrada extra de controle é sensível ao nível lógico (pino “Enable”, ativo alto no caso do 74LS75). Já num Flip-Flop, esta entrada extra é sensível à borda do sinal aplicado na mesma (pino de “Clock”, sensível a borda de subida no caso do 74LS74). Em comum, ambas pastilhas (biestáveis) “copiam” os dados de entrada quando ativadas, conforme cada caso.

2.3 Parte Prática

Monte (um de cada vez) os circuitos 2.2(b) e 2.2(c).

Observações:

1. Note que nos circuitos elétricos das figuras 2.2(b) e 2.2(c) foram usadas de forma proposital as saídas barradas (complementares) dos biestáveis (saídas \bar{Q}), conectados à Leds na configuração “ativo BAIXO”, isto significa que, quando um led se ativa (acende), o biestável está “setado” ($Q = 1$, $\bar{Q} = 0$).
2. Tanto para o caso da figura 2.2(b) ou 2.2(c) apenas uma parte do CI foi utilizada. O estudante pode selecionar outra parte do CI desde que consiga tornar operacional o biestável em questão. Note que no caso do 7475, existem 4 latches mas apenas 2 pinos de Enable. Neste caso, cada 2 latches compartilham um mesmo pino de Enable. Por exemplo: os Latch₀ e o Latch₁ compartilham o pino de Enable: E_{0-1} .
3. As entradas assíncronas \bar{R} (Reset) e \bar{S} (Set) do 7474 devem ser mantidas em nível lógico ALTO para garantir o funcionamento normal do FF-D.
4. A entrada assíncrona \bar{R} (Reset, ativo BAIXO) do 7474 pode ser usada para forçar aquele FF à iniciar no estado resetado.
5. **Tentar simular** a sequencia de níveis lógicos dos pinos de entrada dos biestáveis, conforme o que é sugerido na figura 2.4. Para o caso do circuito com Latch, a entrada de controle corresponde ao sinal de Enable, e para o caso do circuito com Flip-Flop, à entrada de Clock.

6. ANTES de proceder aos testes práticos, sugere-se que o estudante complete as formas de onda esperadas para as saídas Q do Latch e do Flip-Flop (figura 2.4). Use a figura 2.5 para anotar os resultados práticos obtidos. As formas de onda de saída deveriam coincidir entre as 2 figuras.
7. É esperada uma certa dificuldade em simular na prática o sinal de clock, conforme mostrado na figura 2.4, já que neste caso, interessa para esta pastilha em particular os instantes de tempo em que ocorrem bordas de subida de nível lógico. Note que o simples ato de desconectar o cabo de Clock do GND (terra) e deixá-lo no “ar”, já é suficiente para esta pastilha interpretar uma borda de subida ocorrendo. Lembrar que pastilhas TTL tendem a assumir nível lógico ALTO para entradas flutuando, sem sinal. Isto dificulta executar na prática a sequencia de sinal de Clock prevista na figura 2.4. Mesmo assim, tente anotar os resultados práticos obtidos na figura 2.5. Eventualmente não será possível executar a sequencia prevista na figura 2.4, principalmente para o caso do Flip-Flop.
8. Um eventual comportamento “errático” do Flip-Flop, principalmente associado com sua entrada de Clock, se deve ao efeito “bouncing”, ruído produzido pelas próprias conexões manuais, problema que continua existindo mesmo usando chaves, como mostram as figuras 2.3(a) e 2.3(b).
9. Uma vez constatada dificuldade para reproduzir o sinal de Clock com o circuito da figura 2.2(c), sugere-se que o estudante aumente este circuito de forma a incluir uma etapa de “debouncing” como sugerido na figura 2.3(c). Notar que neste caso, todo o CI 7474 será utilizado.
10. Perceber que quando o circuito de “debouncing” é adotado o Flip-Flop volta a se comportar de maneira desejada!
11. Notar pela figura 2.3(b) que o efeito “bouncing” pode durar na prática, até 150 μ s e várias bordas de su-

bida e de descida podem ser observadas dentro deste intervalo (ver também: <http://www.labbookpages.co.uk/electronics/debounce.html>, disponível em 19/08/2010).

2.4 Questões

1. Explique em poucas linhas (máximo 5 linhas) o efeito de “bounce” e faça acompanhar uma figura (diagrama no tempo) que esclareça o problema (comparando tensões analógicas × níveis lógicos).
2. Explique por que o circuito da 2.3(c) funciona como circuito de “debounce”. Dica: repare que o circuito da 2.3(c) faz uso das entradas assíncronas \bar{S} (ou *Preset*) e \bar{R} (ou *Clear*) do CI 7474 (FF-D, ↑). Explique como o biestável RS interno ao FF-D é usado como circuito de “debounce”. Complete a resposta com um diagrama no tempo esclarecendo tensões analógicas × níveis lógicos.
3. Proponha (pesquise e apresente) outros 2 circuitos de “debouncing” e acrescente uma breve explicação (máximo 3 linhas) para o seu funcionamento. Faça acompanhar diagramas no tempo esclarecendo as tensões analógicas × níveis lógicos envolvidos.

Sugere-se consultar:

http://www.nxp.com/documents/data_sheet/74F74.pdf

<https://www.ti.com/lit/ds/symlink/sn54ls75.pdf>

Continua →

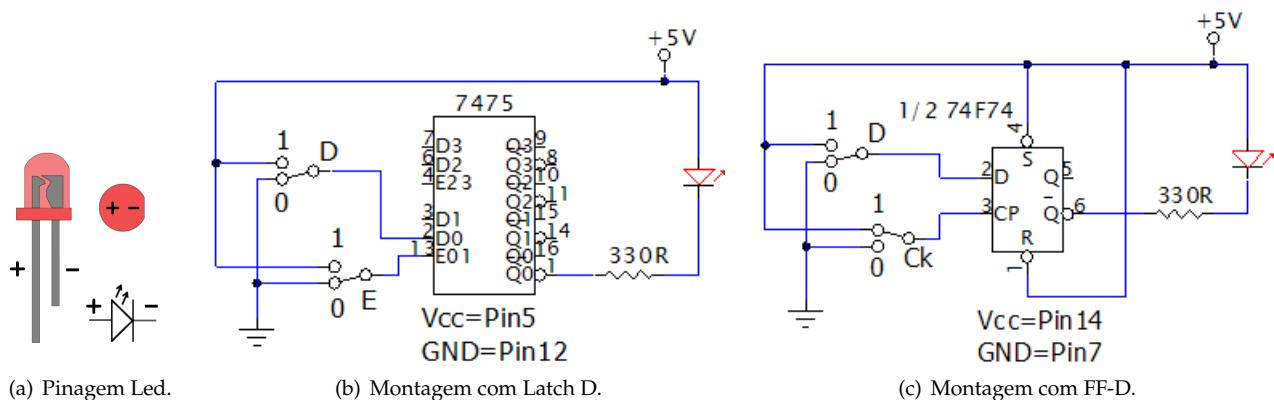


Figura 2.2: Montagens à serem realizadas.

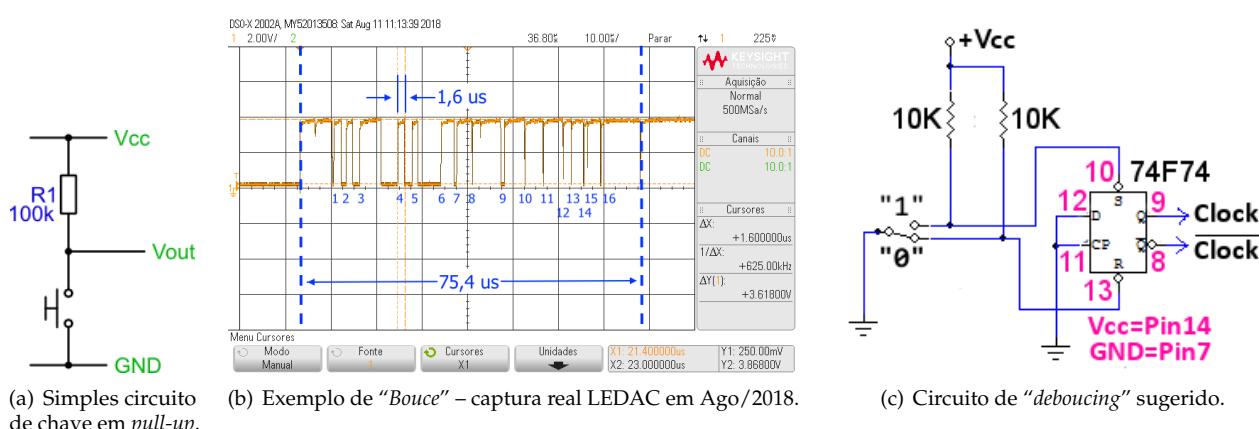


Figura 2.3: Efeito "bouncing" e circuito de debouncing.

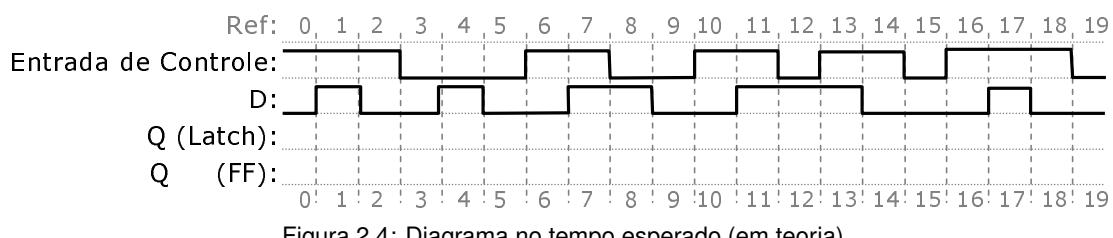


Figura 2.4: Diagrama no tempo esperado (em teoria).

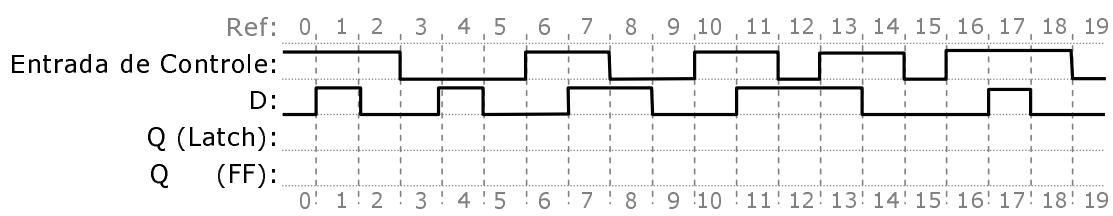


Figura 2.5: Diagrama no tempo levantado na prática (resultado real obtido).

Respostas esperadas

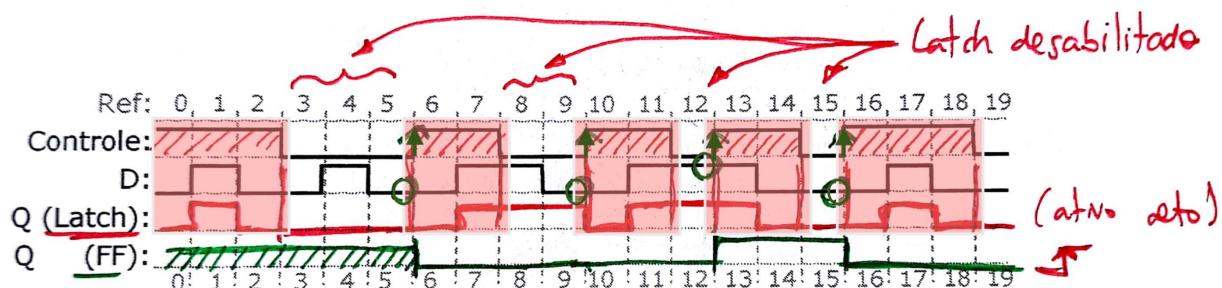


Figura 2.6: Diagrama no tempo esperado (teórico/simulação).

Questões

1. Explicar efeito "bounce":

Explicar que consiste em ruído da chave – ver figura 2.3(b). Este ruído gera vários pulsos (bordas ativas) de um sinal, problema que se agrava num circuito digital, se este tipo de sinal for usado para entrada de clock do mesmo. O que implica que, o usuário usando uma chave pode ter a intenção de gerar apenas uma borda ativa de sinal de clock (por exemplo) e na realidade, sem um circuito de debouncing, ao invés de apenas 1 borda ativa do sinal de clock ser percebida pelo circuito digital, o mesmo pode “ler” e considerar várias bordas ativas para este sinal, ao invés de uma única borda que seria o desejável.

2. Explicar por que circuito da fig. 2.3(c) resolve o problema do “bounce”:

Constar que o problema é resolvido de forma simples: basta um pulso ser considerado nível lógico baixo para uma das entradas assíncronas (ativo baixo) do FF da fig. 2.3(c) ativar a correspondente entrada (de Reset ou Set, dependendo da posição da chave). Mesmo que vários pulsos sejam aplicados à entradas assíncronas de Set ou Reset, o primeiro pulso que for considerado pelo circuito como nível lógico baixo, já basta para ativar esta entrada. Uma vez ativada esta entrada assíncrona, não importa se mais pulsos (ou níveis lógicos baixos) ingressarem numa destas entradas. Perceber também que neste circuito não há como o usuário ativar ao mesmo tempo as entradas de Set e Reset – evitando-se assim o problema de um biestável RS quando se ativam simultaneamente suas duas entradas. E notar que um simples biestável RS pode ser construído na prática usando-se apenas $2 \times \text{NAND}(2)$ ou $2 \times \text{NOR}(2)$. Notar ainda que na prática, estas entradas devem permanecer em nível lógico por no mínimo o tempo de setup (ou t_s) daquela pastilha. E na prática, o $t_s \approx 10\text{ns}$ e nota-se pela fig. 2.3(b) que o gráfico apresentado está na escala de $\mu\text{-segundos}$ ou 10^{-6} (segundos) enquanto t_s está na escala de dezenas de 10^{-9} (segundos) – então é altamente provável que um circuito digital, interprete várias sequências de níveis lógicos baixos pelo gráfico da fig. 2.3(b).

Nota: prefixos SI:

Nome	Símbolo	Escala	Equivalente numérico
nenhum	nenhum	$1 = (1 \times 10^0)$	1,0
mili-	m	10^{-3}	0,001
micro-	μ	10^{-6}	0,000 001
nano-	n	10^{-9}	0,000 000 001
pico-	p	10^{-12}	0,000 000 000 001

3. Apresentar outros circuitos de debouncing...

3 | Contador Assíncrono usando Flip-Flop tipo D

Listar de Material

- 2 × CIs 74LS74 (Flip-Flop D duplo, borda de subida);
- 1 × CI 74LS00 (4 portas NAND(2));
- 1 × módulo display de 7-Segmentos;
- Gerador de sinais (saída TTL, freq.= 1 Hz ~ 10 MHz);
- Osciloscópio de no mínimo 2 canais de entrada.

3.1 Objetivos

Comprovar o funcionamento e operação de um contador binário assíncrono baseado em FF-D.

3.2 Fundamentos Teóricos

Contadores assíncronos são aqueles que se caracterizam por empregarem FF's na configuração *toggle* (ver figura 3.1) e possuir a entrada de clock (*CLK*) do estágio seguinte (próximo FF) conectado ou à saída *Q* ou \bar{Q} do estágio anterior (ou FF anterior). Conforme esta ligação é realizada obtém-se contagem num sentido ou outro (crescente ou decrescente). Leve em consideração ainda que com determinado tipo de ligação, pode estar sendo adotado FF ativado por borda de subida (\uparrow) ou borda de descida (\downarrow) e sendo assim, teremos $2 \times 2 = 4$ possibilidades diferentes de estruturas internas de contadores assíncronos, conforme resume a figura 3.2.

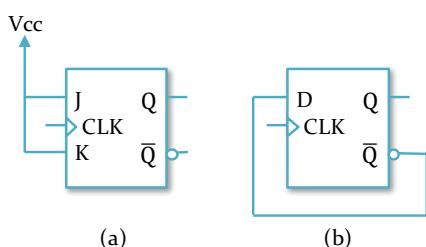


Figura 3.1: Configurações *toggle* para FF's tipo FF-JK (a) e FF-D (b).

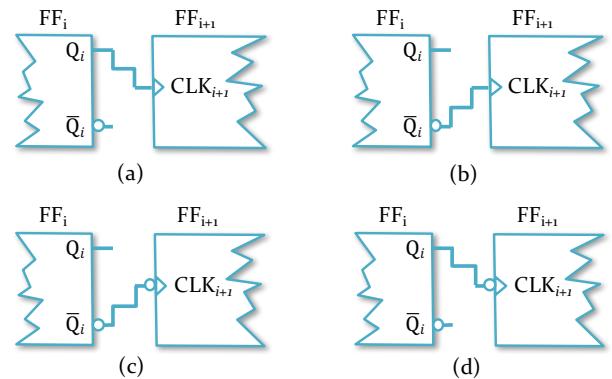


Figura 3.2: Configurações possíveis para estrutura interna de contador assíncrono.

Notar que somente a entrada de clock do primeiro FF é conectado a um gerador externo de clock (como ocorre no caso de todos os contadores assíncronos).

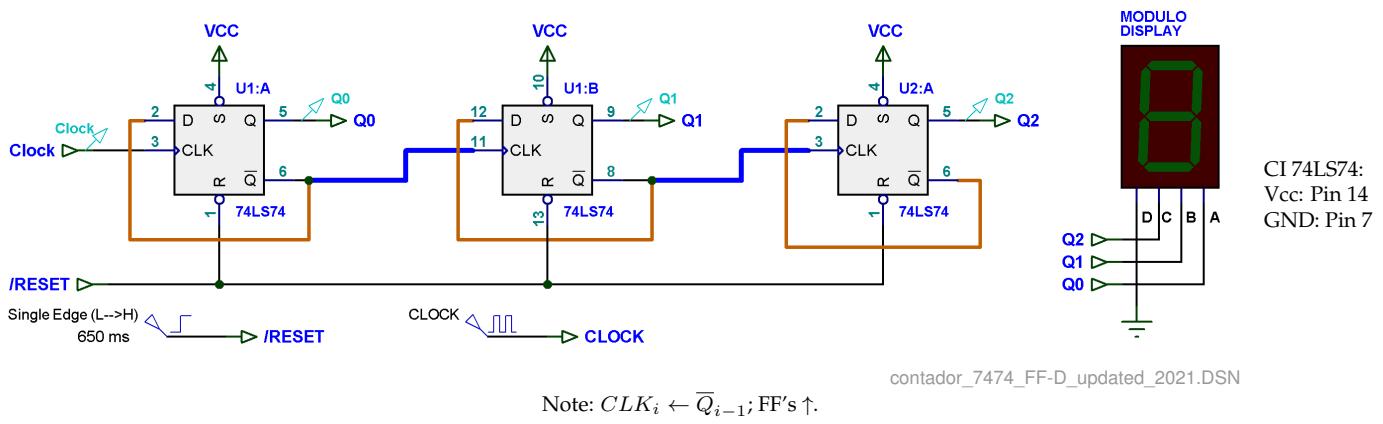
Não existe um “barramento de clock”, onde as entradas de clock de todos os FF's são interligadas num mesmo ponto em comum – por isto este tipo de contador é chamado de **contador assíncrono**.

Este tipo de estrutura de contador também é conhecido como *Ripple Counter*, uma vez que de um estágio para o próximo (FF seguinte), são acumulados atrasos de propagação e assim, o atraso total de propagação fica proporcional ao número de FF's utilizados menos um (o atraso do último FF não impacta no funcionamento da contagem).

3.3 Parte Prática

Neste laboratório será utilizado o FF duplo tipo D, CI 74LS74, ativado por **borda de subida**. **Dois circuitos serão montados** – mas as diferenças entre um e outro são mínimas como veremos à seguir.

As entradas de clock de ambos os circuitos devem ser conectadas a um gerador de sinais, onda quadrada, nível de saída padrão TTL. O gerador pode ser ajustado para operar entre 0,5 Hz à 4 Hz para facilitar que o usuário acompanhe visualmente a leitura da sequência de contagem formada. As saídas $Q_2Q_1Q_0$ devem ser conectadas respectivamente às entradas *CBA* do **módulo display de 7-Segmentos**. Notar porém que Q_0 corresponde ao bit LSB e Q_2 ao bit MSB e a entrada *D* do módulo deve ser aterrada (não é usada).

Figura 3.3: Esquema elétrico da **primeira montagem**.

3.3.1 Primeira Montagem

O circuito da **primeira montagem** é mostrado na figura 3.3.

Notar alguns detalhes:

- Na primeira montagem a entrada de clock do próximo estágio é conectada à saída \bar{Q} do estágio anterior ($CLK_i \leftarrow \bar{Q}_{i-1}$).
- É interessante se implementar um “**barramento de Reset**” interligando as entradas de *Reset* dos FF’s, assim, um único pulso de nível lógico baixo é suficiente para garantir o início de operação do circuito com todas as saídas resetadas. Naturalmente que durante a operação normal, este barramento deve permanecer em nível lógico alto (conectado ao +Vcc) – Reset desabilitado. **Já os pinos de Set (ou Preset) devem ser mantidos em nível lógico alto** para que o circuito não apresente nenhum comportamento anômalo ou fique paralisado no estado $Q_2Q_1Q_0 = 111_{(2)} = 7_{(10)}$.

- As entradas de *Reset* ou *Set* são assíncronas, ou seja, não funcionam em sincronismo com a entrada de controle deste biestável (no caso, a entrada de clock).
- Perceba como este contador conta. Complete o diagrama no tempo da fig. 4.5(a).

3.3.2 Segunda Montagem

Para a **segunda montagem**, você vai apenas trocar as conexões associadas com as entradas de Clock de cada FF. Antes: $CLK_i \leftarrow \bar{Q}_{i-1}$, agora: $CLK_i \leftarrow Q_{i-1}$. Para isto: conectar a entrada de clock do segundo FF (U1:B), pino 11, no pino 5 de U1:A (saída Q do estágio anterior), desfazendo a conexão anterior ao pino 6 (\bar{Q}). E conectar o pino 3 de U2:A ao pino 9 de U1:B, desfazendo a conexão que havia antes ao pino 8 (\bar{Q}). **Perceba que o sentido da contagem vai mudar agora**. Complete o diagrama no tempo da fig. 4.5(b).

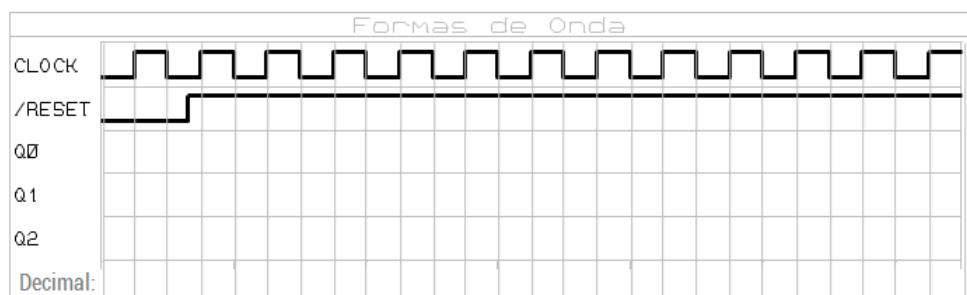
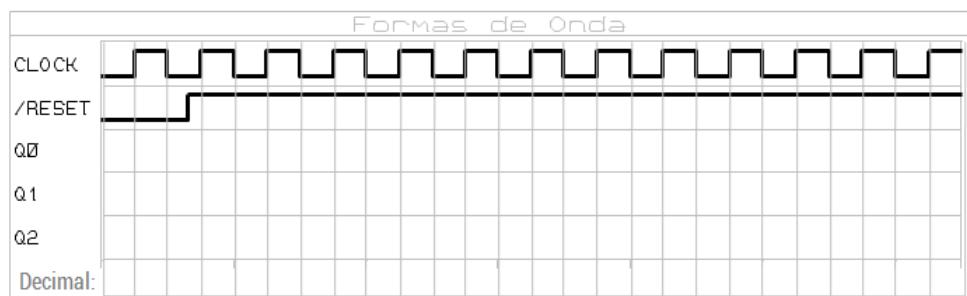
(a) Formas de onda para **primeira montagem** ($CLK_i \leftarrow \bar{Q}_{i-1}$; FF's ↑).(b) Formas de onda para **segunda montagem** ($CLK_i \leftarrow Q_{i-1}$; FF's ↑).

Figura 3.4: Formas de onda esperadas (simuladas) para as montagens deste laboratório.

3.4 Questões

- 1) Primeiramente (e durante a aula), complete os diagramas no tempo das figuras 4.5 (a) e (b).

Obs.: Completar a linha “Decimal” prevista nas figuras com o resultado da contagem exibido no módulo display. Note que a saída Q_0 corresponde à saída LSB do contador.

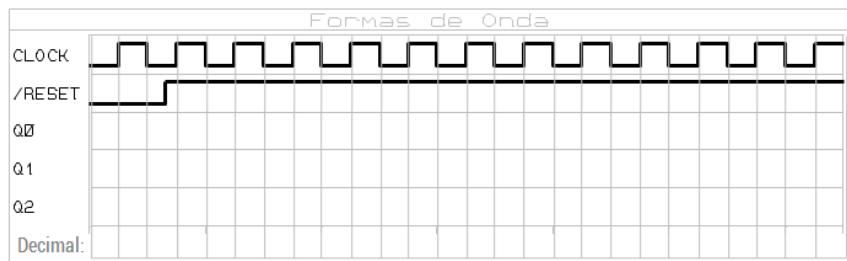
- 2) O que mudaria no caso da primeira montagem se o FF utilizado fosse ativado por **borda de descida** ao invés de borda de subida? Mostre o que ocorreria no digrama no tempo da fig. 3.5).
- 3) Apresente o diagrama elétrico de um **contador binário assíncrono crescente de módulo 16** (isto é, que possua 16 estados distintos de saída, contado de 0 à 15)? Como deveria ser expandido um dos circuitos anteriores? Complete a fig. 3.6 demonstrando como ficam as formas de onda resultantes.
- 4) **Modifique** o circuito do diagrama elétrico da fig. 3.3 para obter um **contador assíncrono crescente de módulo 5**, isto é, que conte de 0 à 4. Obs.: Se fará necessário o uso de portas lógicas básicas extras como AND, NOT ou NAND. Complete a fig 3.7.
- 5) **Modifique** o circuito do diagrama elétrico da fig. 3.3 para

obter um contador assíncrono **decrescente de módulo 5, contando de 7 à 3** (inclusive). Obs.: Será necessário o uso de portas lógicas básicas extras como AND, NOT ou NAND. Complete a fig 3.8.

3.5 Sugestões

O estudante pode conectar um osciloscópio de mais de 2 canais aos circuitos sendo testados. Neste caso, o canal 1 seria conectado à entrada de Clock (saída do gerador de sinais). O canal 2 seria conectado a saída Q_0 (do primeiro FF). O canal 3 seria conectado à saída Q_1 (segundo FF) e o canal 4, à saída Q_2 (terceiro FF). Note como os atrasos de propagação vão se acumulando do estágio inicial para o final.

Note que apenas pressionar o botão de Auto Scale do osciloscópio pode não ser suficiente para que seja mostrado de forma apropriada as 4 formas de onda, principalmente no caso do sinal de *Clock* variar numa frequência muito baixa. Neste caso é necessário que o estudante modifique a base de tempo do osciloscópio, botão Horizontal para que sua base de tempo diminua para algo na faixa de 200 ms/div. Note então que o sinal passará a ser atualizado de forma lenta na tela do osciloscópio, mas será possível visualizar mais de um ciclo completo do sinal de *Clock*. Ou então aumente a frequência do sinal de clock de entrada para algo como 1 KHz, o que deve facilitar o sincronismo de sinais no osciloscópio.



Obs.: Formas de onda para $\overline{Q}_{i-1} \rightarrow CLK_i$, usando FF's ↓ (ativados por borda de descida).

Figura 3.5: Formas de onda esperadas para questão 2.

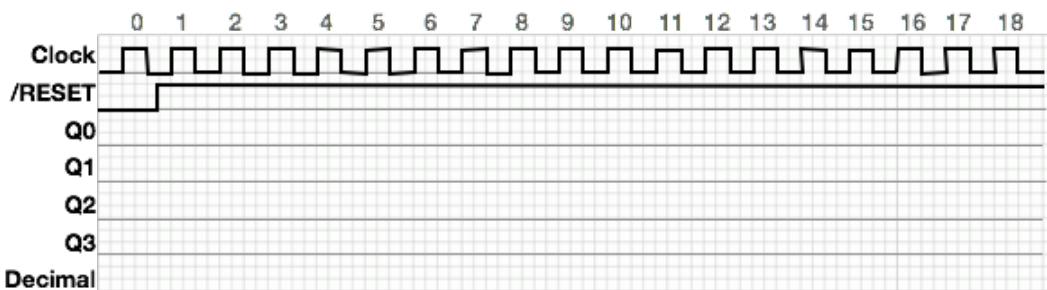


Figura 3.6: Formas de onda esperadas para a questão 3 (contador MOD-16).

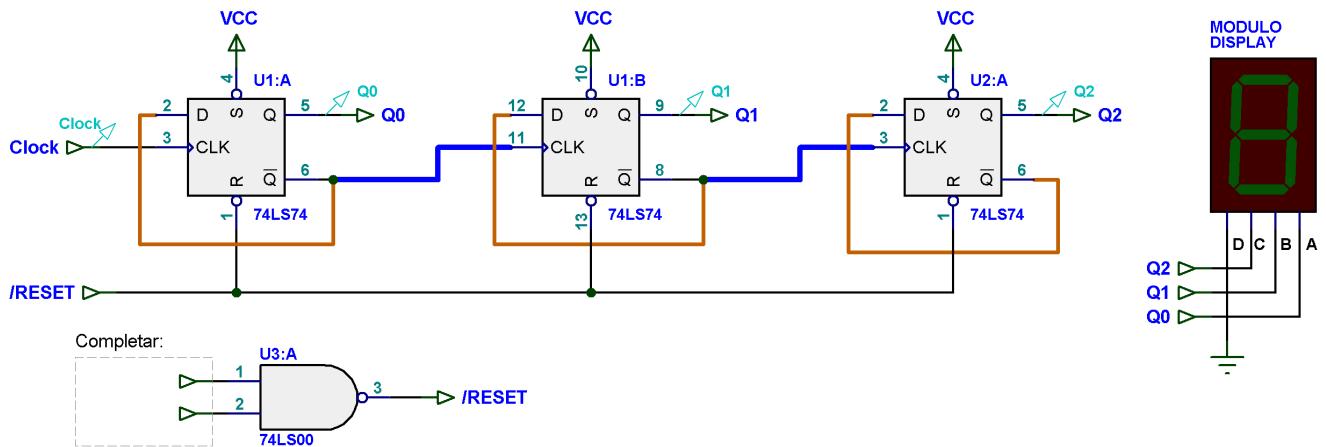


Figura 3.7: Espaço para resposta (diagrama elétrico) referente à questão 4.

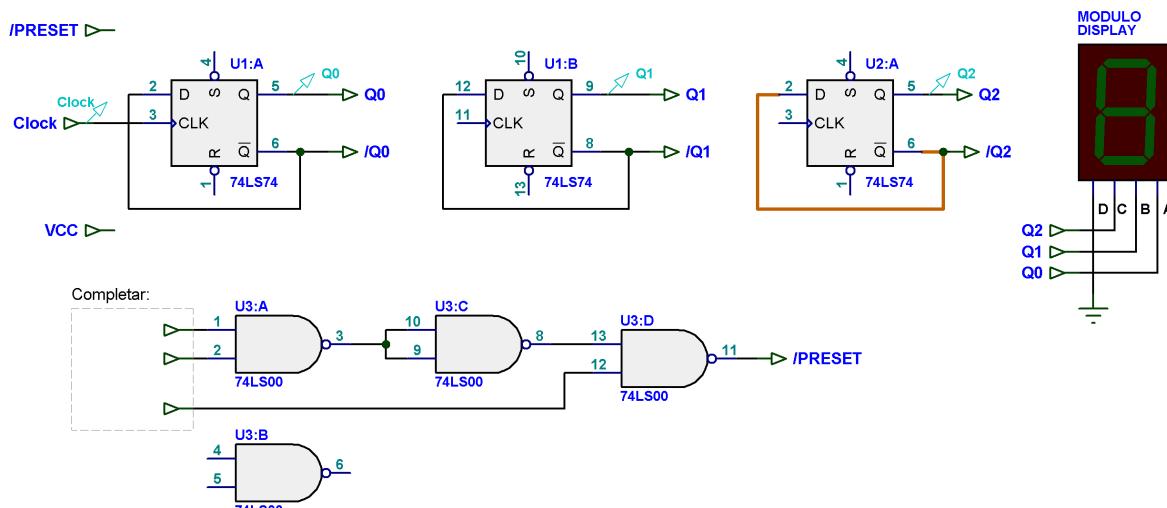


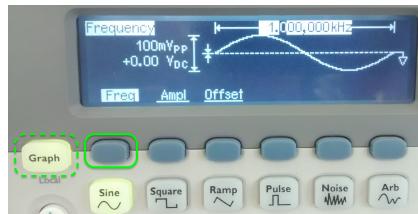
Figura 3.8: Espaço para resposta (diagrama elétrico) referente à questão 5.

3.6 Anexo 1: Como usar Gerador de Sinais

A próxima figura (3.9) ilustra como ajustar o gerador de sinais.



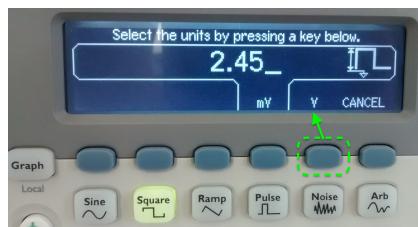
(a) Tela inicial (ajustes de fábrica). Para continuar aperte qualquer botão como **Freq**, **Ampl** ou **Offset**.



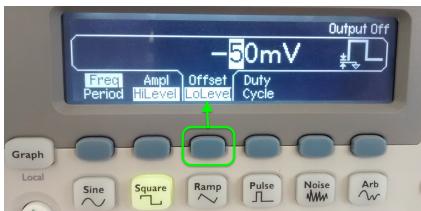
(b) Aperte o botão associado com **Freq** da figura anterior (a) para ajustar a frequência. Note que inicialmente o gerador gera um senóide de 1,0 KHz com amplitude bem baixa (100 mVpp). O botão **Graph** pode estar ativado ou não.



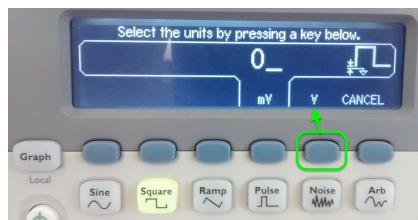
(d) Ajustando a amplitude da onda. Aperte o botão **Ampl** na tela da figura (a).



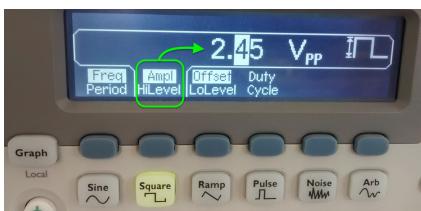
(e) Digite um valor como 2.45_ (Volts) para amplitude de pico da onda à ser gerada ($2 \times 2,45 = 4,9$ Volts). Atente para apertar o botão **V** para sair da escala de fábrica em mVolts.



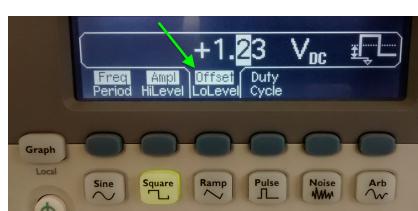
(g) Ajustando o **LoLevel** da onda. Na tela da figura (a) aperte $2 \times$ o botão **Offset**. Note que de fábrica teríamos um valor negativo de -50mV. Tensões negativas de entrada num circuito digital podem queimar componentes!



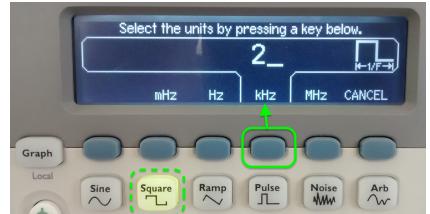
(h) Digite o valor 0_ e aperte o botão **V** para que o gerador altere este valor para 0,0 Volts.



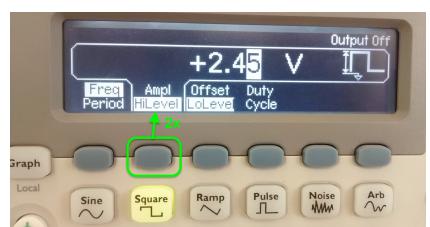
(j) Note que **Ampl** deve continuar valendo 2,45 Volts (o mesmo valor para **HiLevel**!).



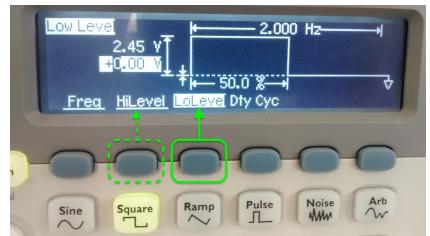
(k) Note que o valor para **OffSet** ficou em 1,23 V (devido ao bug comentado anteriormente – deveria ter ficado em $4,9/2 = 2,45$ Volts). Mas **LoLevel** permanece em 0,00 V como é o desejável.



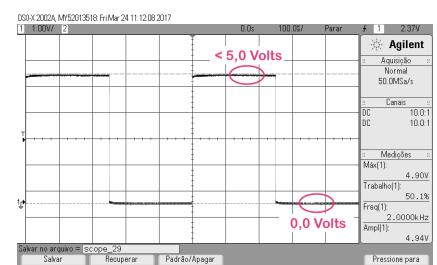
(c) Ajuste inicial de Frequência. Use o teclado numérico para digitar diretamente um valor (use 2_ **KHz** para ajustes com osciloscópio ou 2 **Hz** para uso no laboratório). Repare na escala desejada. Eventualmente o botão **Square** já pode ter sido pressionado.



(f) Repare que se o botão **Ampl** da figura (d) for pressionado $2 \times$ o valor para **HiLevel** continua sendo +2.45 V (resultado de um provável bug de software na interface do equipamento).



(i) Confirme (verifique) que **LoLevel** = 0.00 V e que **HiLevel** = 2.45 V. Para facilitar, o botão **Graph** pode estar ativado.



(l) Use o osciloscópio para confirmar a onda quadrada nível TTL sendo gerada. Não esquecer de ativar botão **Output** no gerador de sinais para liberar a saída do gerador.

Figura 3.9: Dicas para uso do gerador de sinais Agilent 33210A 10 MHz.

3.7 Anexo 2: Gerador de Pulso de Clock

Tente simular o sinal de Clock usando um botão (*push-button*) como o mostrado na fig. 3.10.

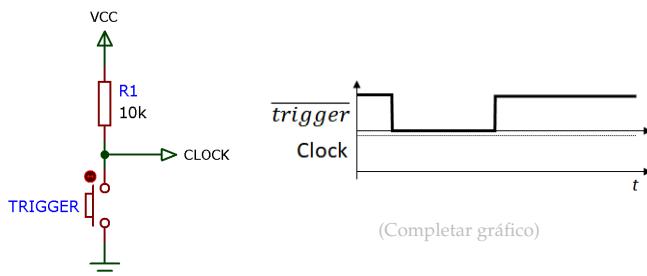


Figura 3.10: Circuito gerador de pulso de clock (errático).

Questão: – O que realmente acontece na prática?

Círculo Monoastável

Uma possível solução para o problema da geração do sinal de clock é incorporar um circuito de “monoastável”. Um monoastável é um circuito que gera um pulso em nível lógico ALTO ou BAIXO com duração de tempo “programada” usualmente via circuito RC (Resistor + Capacitor).

A figura 3.11 mostra um circuito prático de um monoastável usando apenas 2 portas NAND schmitt trigger. A seção de schmitt trigger é necessária para “digitalizar” a tensão analógica envolvida com a carga e descarga do capacitor, que foi incorporada dentro uma de uma malha de circuito digital.

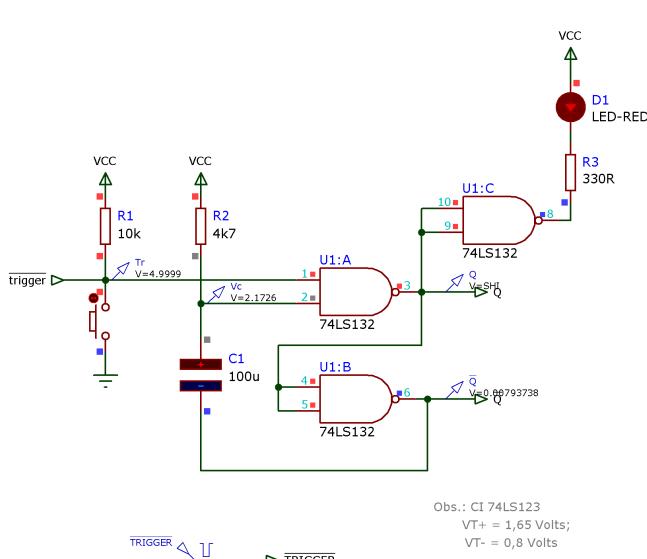


Figura 3.11: Circuito gerador de pulso de clock (monoastável).

A pinagem do CI 74LS132 é mostrada na fig. 3.12.

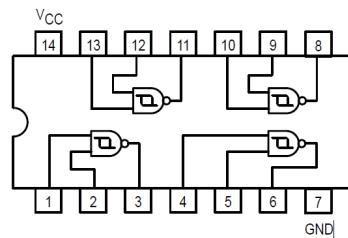


Figura 3.12: Pinagem do CI 74LS132 (quad 2-input schmitt trigger nand gate).

Para fins de simulação num soft como o Proteus, o “pressionar” do botão pode ser simulado por um pulso que comuta de nível lógico ALTO para BAIXO, no instante de tempo inicial (*Start Time*) $t = 1$ (segundo), permanece em nível lógico baixo certo período de tempo (no caso, *Pulse width*, $t_w = 125$ milissegundos) e que depois retorna para nível lógico ALTO. A figura 3.13 detalha como este sinal (*Single Pulse*) é configurado no Proteus.

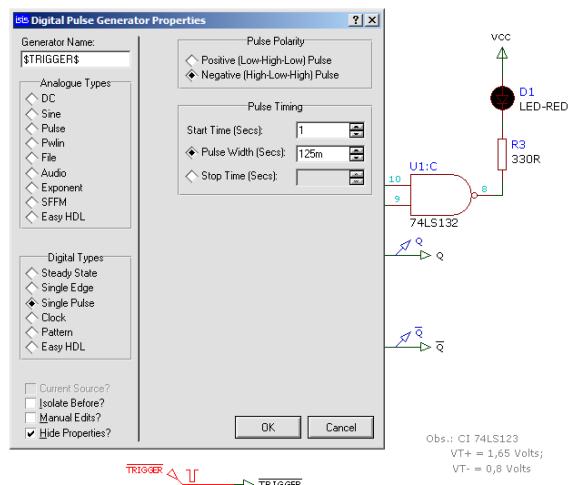


Figura 3.13: Propriedades do pulso $\overline{\text{trigger}}$ (disparo).

A figura 3.14 tenta mostrar o que ocorre quando se aciona o push-button depois de 1 segundo do circuito ter sido alimentado (desconsidere o período transitório inicial no qual o capacitor é carregado e o botão não é pressionado).

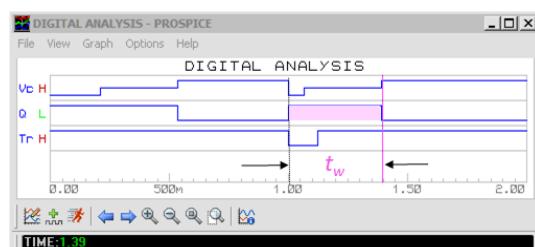


Figura 3.14: Formas de onda geradas no circuito da fig. 3.11 (simulação via Proteus).

Note que antes deste botão ser pressionado a saída Q deste circuito está em nível lógico BAIXO (sinalizado pelo Led que acompanha o circuito e que vai estar apagado). Quando este botão é acionado, no tempo $t = 1$ segundo, a saída Q comuta para nível lógico BAIXO e permanece neste nível por um período de tempo, t_w que é proporcional a rede RC presente no circuito:

$$t_w = RC \ln \left(\frac{1}{1 - \Delta v/E} \right)$$

Em teoria:

$$\begin{aligned} t_w &= 4700 \cdot 100 \times 10^{-6} \ln \left(\frac{1}{1 - 3,5/5} \right) \\ &= 0,47 \ln \left(\frac{1}{1 - 0,7} \right) \\ &= 0,47 \ln(1/0,3) \\ &= 0,16764 \text{ (segundos)} \end{aligned}$$

O Proteus, na sua simulação (fig. 3.14) obtém um $t_w \cong 390$ milissegundos. Note porém que o Proteus não pode ser considerado confiável em simulações envolvendo partes analógicas e digitais, mesmo que o capacitor C1 mostrado na fig. 3.11 seja “Animated”. Tanto que o sinal usado (pulso) para simular o pressionamento do botão não funciona para efeitos de simulação analógica, isto é, o algoritmo interno no Proteus não consegue resolver a parte transitória deste circuito envolvendo o que acontece com a carga e descarga deste capacitor quando se pressiona esta chave. Mas, para fins práticos, este circuito funciona.

4 | Contador Assíncrono Módulo- n com Reset Manual

ListadeMaterial

- 2 CI's 74LS76 (Dual Negative-Edge-Triggered J-K Flip-Flop (with Preset and Clear), \downarrow);
Obs: Na falta destes podem ser usados o 74LS112 ou 74LS73 (mas respeitando-se sua pinagem – diferente).
- 1 CI 74F02 ($4 \times \text{NOR}(2)$);
- 1 push-bottom NA miniatura;
- 1 resistor de 100Ω ;
- 1 módulo de display de 7 segmentos (preferencialmente hexadecimal).

4.1 Objetivos

Montar um circuito contador assíncrono de módulo- n crescente com circuito de Reset manual (*push-bottom*).

4.2 Fundamentos Teóricos

Trata-se de montar um contador assíncrono crescente utilizado FF-JK's ativados por borda de descida do sinal do clock e que possuem entradas assíncronas para *Set* e *Reset* – ver figura 2. Um barramento de \overline{SET} e outro (separado) de \overline{RESET} são criados. Note que o barramento de \overline{SET} é permanentemente conectado ao $+VCC$ (a princípio não é utilizado). Mas o barramento de \overline{RESET} é configurado através das portas lógicas NOR de tal forma que um nível lógico BAIXO seja gerado quando a contagem formada pelos 4 FF's atinge o número $10_{(10)} = 1010_{(2)}$ (ou seja: $Q_3 = 1$ e $Q_1 = 1$). Note porém que o sinal de \overline{RESET} também pode ser produzido em qualquer instante de tempo, quando o usuário pressionar o botão de "Reset". Note também que a porta U3:A possui seu pino 3 conectado à um resistor de *pull-down* (100Ω).

4.3 Parte Prática

O circuito a ser avaliado nesta prática aparece na figura 1 mostrada à seguir. Sugere-se que o aluno(a) divida a sua montagem em 2 partes:

1. Primeiramente monte o circuito da figura 2(a), a base do contador assíncrono crescente de módulo completo

($2^4 = 16$), ligando (inicialmente) o barramento de \overline{SET} e \overline{RESET} ao barramento $+VCC$ da sua montagem no protoboard. Para comprovar o correto funcionamento desta parte do circuito, conectar as saídas dos FF's à um módulo display (fig. 2(c));

2. Em seguida, monte o circuito da figura 2(b). Este circuito substitui a ligação anterior realizada para o barramento de \overline{RESET} . O barramento de \overline{SET} continua conectado ao $+VCC$. Alimente este circuito e comprove que um sinal de \overline{RESET} é gerado sempre que a contagem atinge o número $10_{(10)}$ ou sempre que o usuário pressiona o botão de "Reset".

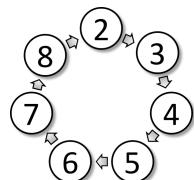
Atenção para alimentar corretamente os CI's utilizados:

	74LS76	74F02	74LS112	74LS73
Vcc	Pin 05	Pino 14	Pin 16	Pin 4
GND	Pin 13	Pino 7	Pin 8	Pin 11

Se sobrar tempo tente implementar na prática o circuito exigido no item 5 das Perguntas.

4.4 Questões

- 1) Complete a fig. 4.5 com as formas de onda para o circuito da fig. 4.4(a) – enquanto o mesmo ainda é um contador MOD-16. Suponha que o circuito inicia resetado.
- 2) Justifique porque no circuito de "Reset Manual" é adotado um **resistor de "pull down"** no valor de 100Ω . Poderia ter sido utilizado um resistor de $10K\Omega$?
- 3) **Deduza a equação** para o sinal \overline{RESET} da figura 4.4(b). No circuito ele é ativo baixo, isto é \overline{RESET} ; sugere-se raciocinar em termos de lógica "ativo alto". Note que no lugar das portas U3:B, U3:C e U3:D, poderia ter sido utilizada uma única e simples porta AND de 2 entradas. O que aconteceria se o pino 3 da porta U3:A fosse permanentemente conectado ao $+VCC$?
- 4) Complete a fig. 4.6, com as **formas de onda** para o circuito da fig. 4.4(a) incluindo o sinal de \overline{RESET} (Fig. 4.4(b)). Suponha que o circuito inicia resetado.
- 5) **Modifique** o circuito completo da figura 4.4 de forma que o mesmo realize a seguinte sequência de contagem:



Note que desta vez, uma leve modificação será necessária

no circuito da fig. 4.4. Dica: talvez seja necessário trabalhar agora com as entradas assíncronas de "Preset" ou "Set". (\bar{S}_D) de alguns FFs.

- 6) Faça acompanhar o diagrama elétrico referente ao item 5.
- 7) Complete a fig. 4.7, com as formas de onda esperadas para o item 5 das Perguntas.

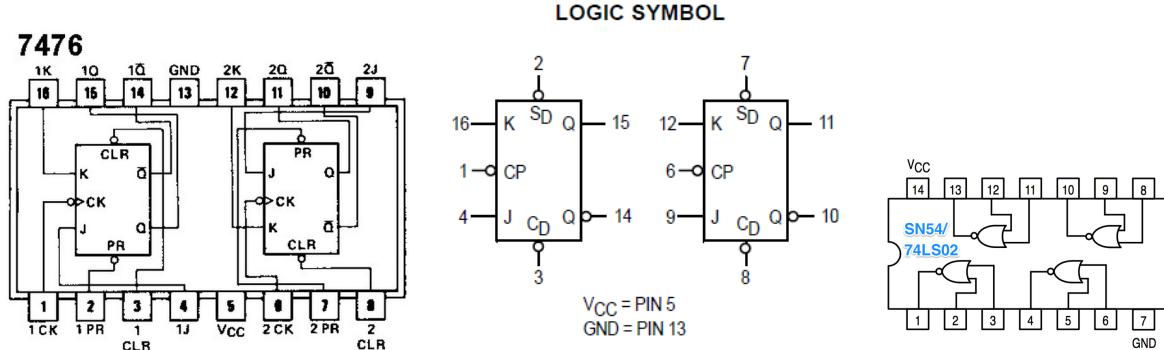


Figura 4.1: Detalhes do 74LS76 e do 74LS02.

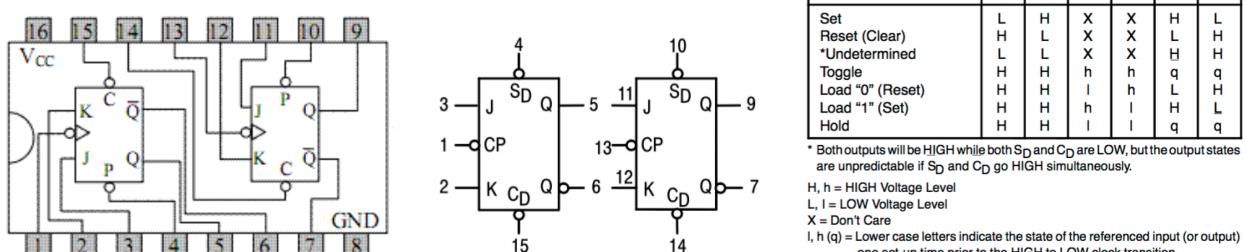


Figura 4.2: Detalhes do 74LS112 (Dual JK negative edge-triggered flip-flop).

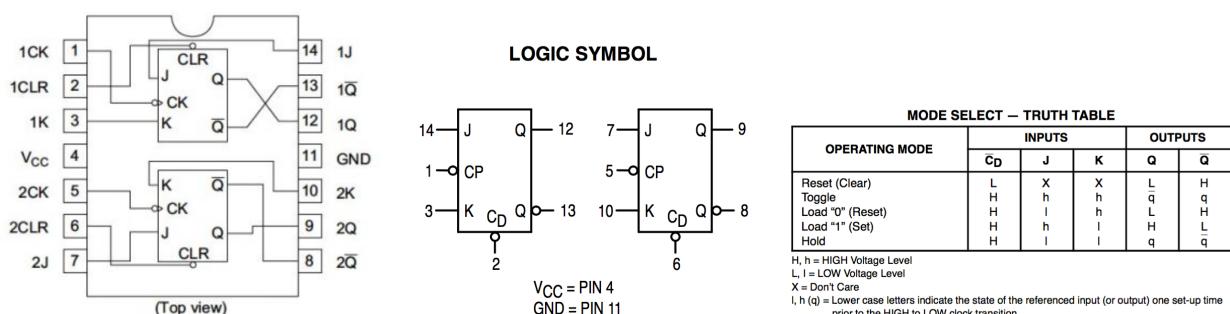
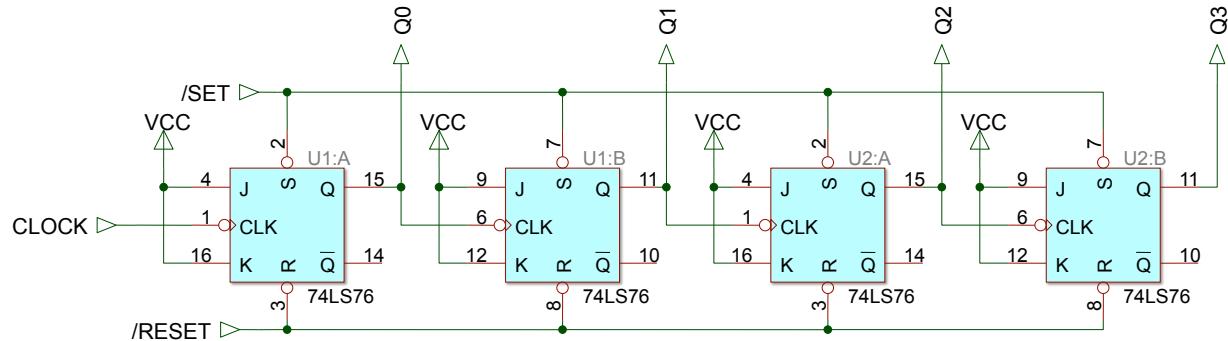
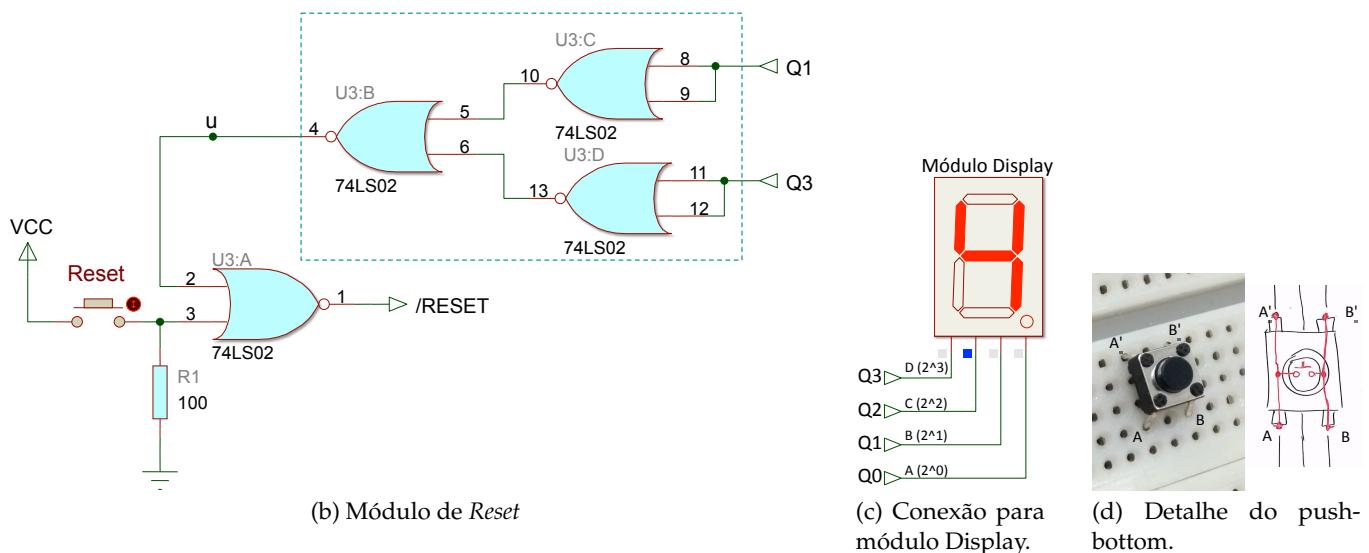


Figura 4.3: Detalhes do 74LS73 (Dual JK negative edge-triggered flip-flop).



(a) Módulo básico do contador assíncrono crescente MOD-16.



(b) Módulo de Reset

(c) Conexão para módulo Display.

(d) Detalhe do push-bottom.

Figura 4.4: Contador assíncrono com circuito de Reset.

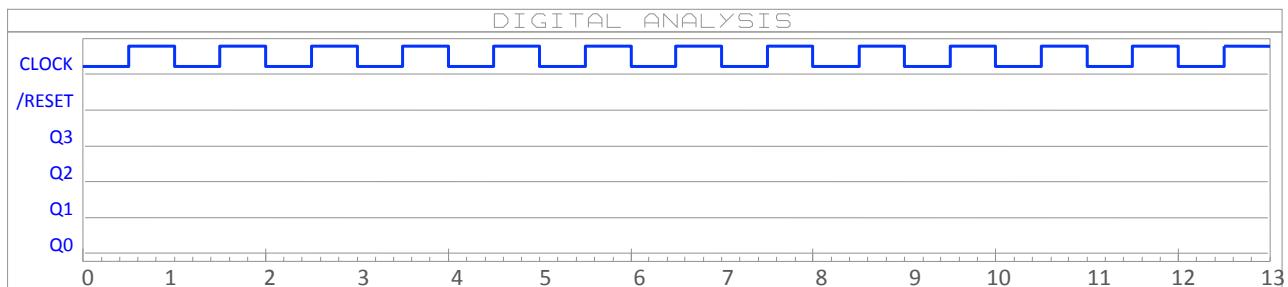


Figura 4.5: Formas de onda esperadas para circuito da fig. 4.4(a) (Contador MOD-16) – Item 1 das Questões.

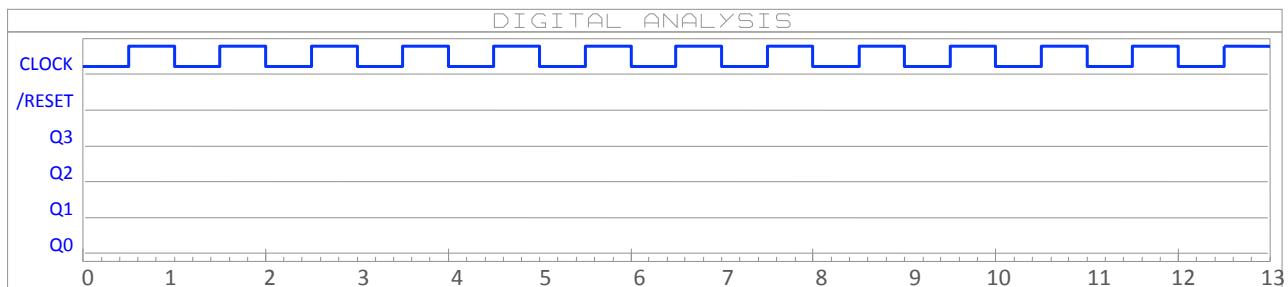


Figura 4.6: Formas de onda esperadas para circuito completo da fig. 4.4 – Item 4 das Questões.

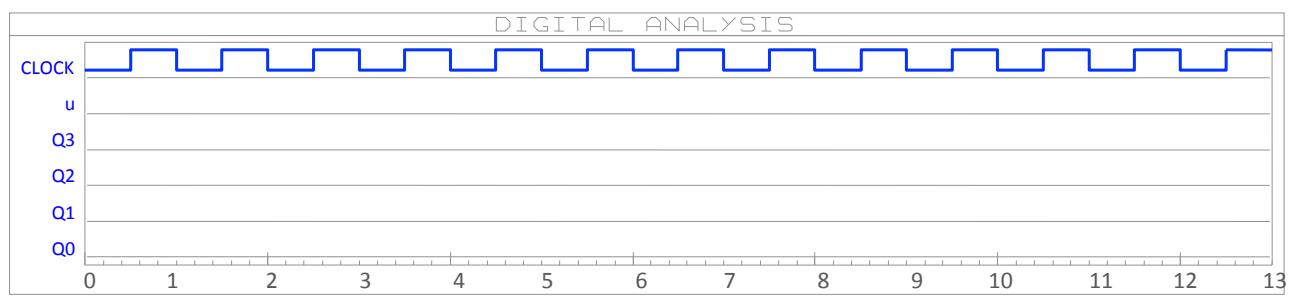


Figura 4.7: Formas de onda esperadas para circuito modificado referente ao item 7 das Questões.

5 | Contador Assíncrono $Up/Down$

Listas de Material

- 2 CI's 74LS74 (Duplo Flip-Flop D, \uparrow ;
- 1 CI 74LS157 (Quádruplo MUX de 2 canais)
- 1 módulo de display de 7 segmentos.

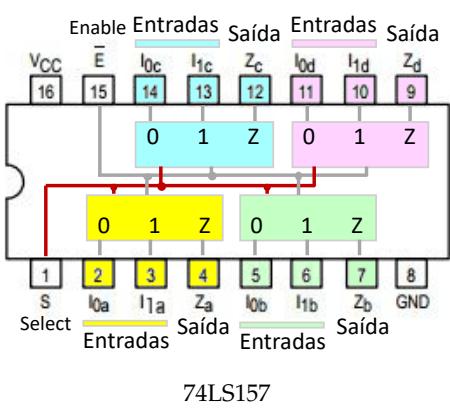
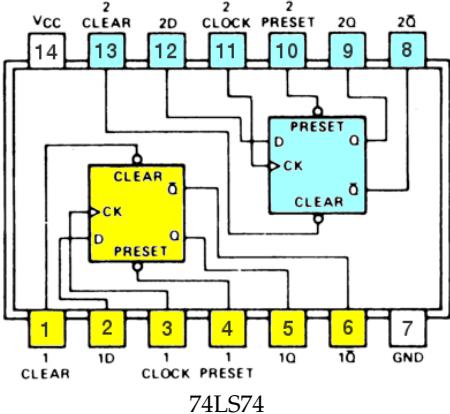


Figura 5.1: Pinagens dos CI's.

5.1 Objetivos

Montar um circuito contador assíncrono crescente e decrescente cujo modo de contagem é controlado através do nível lógico aplicado na entrada "Mode" do circuito.

5.2 Fundamentos Teóricos

Um contador assíncrono, também conhecido como contador ondulante (ou "ripple counter"), é um contador onde os flip-flops não mudam todos de estado ao mesmo tempo, ou seja, são caracterizados por seus flip-flops funcionarem de maneira assíncrona (sem sincronismo), não possuindo entradas de clock em comum. Neste tipo de circuito, a entrada clock se faz apenas no primeiro flip-flop, sendo as outras derivadas das saídas dos blocos anteriores. A saída Q ou \bar{Q} de um flip-flop é conectada na entrada de clock do próximo flip-flop. Neste caso, a saída de um flip-flop funciona como a entrada de relógio para o flip-flop seguinte.

Será utilizado um CI multiplexador de 2 canais (CI 74157) para selecionar se será a saída Q ou \bar{Q} que será propagada para a entrada de clock do estágio seguinte.

E para este laboratório serão utilizados 3 FF's do tipo D configurados como biestáveis do tipo T (*toggle*), isto é, a cada vez que entra uma borda ativa do sinal de clock, a saída de cada FF alterna seu estado. No caso do FF-D isto é facilmente obtido conectando-se a sua entrada D à sua própria saída \bar{Q} . Notar ainda que estamos trabalhando com FF-D's sensíveis à borda de subida do sinal de clock. Sendo assim, para contar de maneira crescente, a entrada de clock do estágio seguinte deve ser ligada à saída \bar{Q} do estágio anterior – ver figura 5.2 à seguir.. E para este circuito contar de maneira decrescente basta ligar a entrada de clock de próximo estágio, à saída Q do estágio anterior.

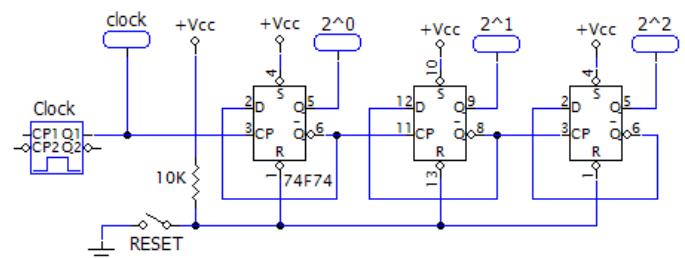
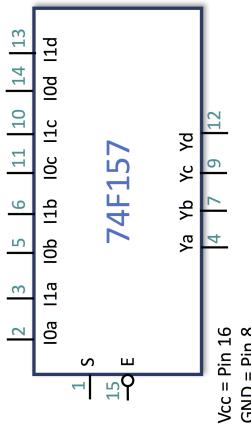


Figura 5.2: Contador assíncrono crescente usando FF-D \uparrow .

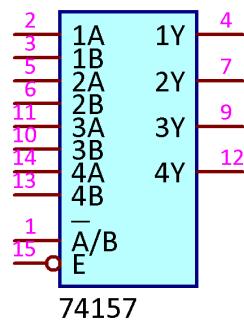
5.3 Parte Prática

Usar 2 CI's 7474 (FF-D, sensível à borda de subida) para montar um contador assíncrono de módulo 8. Note que apenas 3 FFs serão utilizados: $2^3 = 8$ estados. Depois, usando o MUX de 2 canais, CI 74157, conectar as saídas Q e \bar{Q} do estágio atual para as entradas de clock do próximo estágio, de forma a que quando a entrada de seleção do MUX ("Select") estiver em nível lógico ALTO, a contagem seja realizada de maneira crescente e quando

esta entrada estiver em nível lógico baixo, deve ser realizada uma contagem decrescente. A entrada de *Select* do MUX será chamada de entrada “*Mode*” e desta forma este contador deve ser *Up/Down*. Repare que é necessário habilitar o CI 74157 (MUX) aplicando nível lógico BAIXO na sua entrada *Enable* (pino 15), e que existe apenas 1 único pino de *Select* (*S* – pino 1).



(a) Símbologia ANSI.



(b) “Símbologia” Proteus.

5.4 Questões

- 1) Complete o diagrama elétrico final do circuito montado na prática sem se esquecer de mencionar quais são as entradas e saídas MSB e LSB – ver figura 5.4.

MUX (74LS157)	Saídas FFs (74LS74)		
Pin Name	Pin Number	Pin Name	Pin Number
$I_{0a} == 1A$	2	Q_0 (U __)	
$I_{1a} == 1B$	3	Q_1 (U __)	
$I_{0b} == 2A$	5	Q_2 (U __)	
$I_{1b} == 2B$	6	Q_3 (U __)	
$I_{0c} == 3A$	11	Q_4 (U __)	
$I_{1c} == 3B$	10	Q_5 (U __)	
$I_{0d} == 4A$	14	Q_6 (U __)	
$I_{1d} == 4B$	13	Q_7 (U __)	

(c) Tabela de “equivalência” incluindo coluna para conexões.

Figura 5.3: (a) e (b) Símbolo lógico e pinagem do 74F157; (c) Tabela para facilitar conexões entre as saídas dos FFs e as entradas dos MUXes.

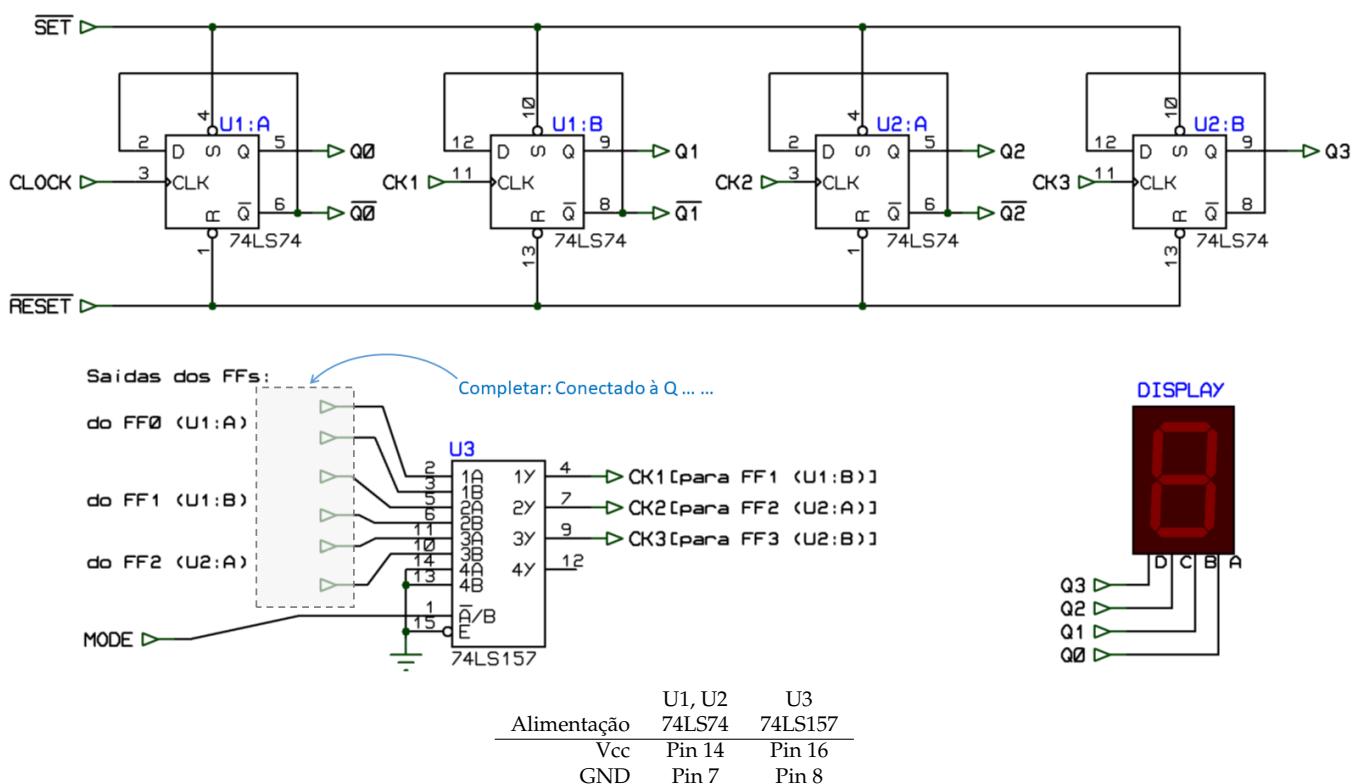


Figura 5.4: Diagrama elétrico parcial para circuito deste laboratório (à completar).

6 | Divisor de frequência

Listar de Material

- 1 × CI 7408 (4 × AND(2));
- 2 × CI 7490 ($\div 2$ e $\div 5$, \downarrow);
- 2 × módulos display de 7-segmentos;
- gerador de sinais + fonte 5 V;

6.1 Objetivos

O objetivo deste laboratório é montar um divisor de frequência por 38 baseado no contador integrado assíncrono, 7490.

6.2 Fundamentos Teóricos

Neste laboratório será explorado o CI 7490 que é um contador integrado assíncrono, disparado por borda de descida. A pinagem deste CI é mostrada na figura 6.1 abaixo.

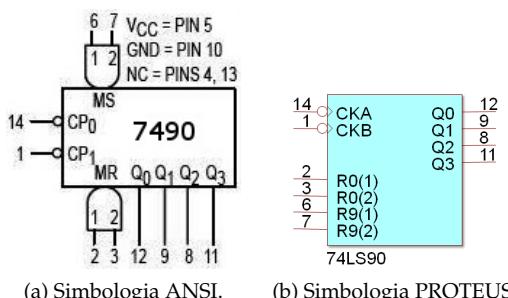


Figura 6.1: Pinagem do CI 7490.

Este CI apresenta 2 estágios internos. O primeiro formado pelos pinos CP0 e Q0 na típica configuração de circuito "togle" usando FF-JK. O segundo estágio é formado por 2 FFs-JK e 1 FF-RS configurados internamente de tal maneira a dividir a frequência na sua entrada, pino CP1, por 5. A divisão da frequência de entrada por 10 ocorre quando o primeiro e segundo estágio são interligados, conectando-se a saída do primeiro estágio, pino Q0 na entrada do segundo estágio, pino CP1. Desta forma o contador conta de 0 à 9 (em decimal).

Este CI possui ainda 2 pares de entradas assíncronas de controle: MS1 e MS2 (relacionadas com o *Master Set*) e MR1 e MR2 (relacionadas com o *Master Reset*). Para forçar as saídas deste contador para $Q_3Q_2Q_1Q_0 = 1001_{(2)} = 9_{(10)}$, os pinos MS1 e MS2 devem ir a nível lógico ALTO (enquanto as outras entradas MR1 e MR2 devem permanecer em nível lógico BAIXO). Para

zerar todas as entradas deste CI, aplica-se nível lógico ALTO às entradas MR1 e MR2 (enquanto se mantêm as outras entradas MR1 e MR2 em nível lógico BAIXO).

6.3 Parte Prática

Deseja-se nesta prática, montar um divisor em frequência por 38. Para tanto:

- 1) Primeiramente monte o circuito mostrado na figura 6.2(a). Trata-se simplesmente de um contador BCD (ou decimal) de 0 à 99, cujas saídas foram conectadas à displays de 7-segmentos com o único objeto de comprovar que a parte inicial do circuito está Ok (funcional). Na entrada de clock (F_IN) do circuito, conectar um gerador de sinais com saída ajustada para padrão TTL e onda quadrada oscilando à frequência entre 1 à 10 Hz (para que a contagem possa ser acompanhada visualmente nos displays). Repare que os pinos MS1 (*Master Set 1*) e MS2 (*Master Set 2*), ou R9(1) e R9(2) na simbologia adotada pelo Proteus, DEVEM permanecer em nível lógico BAIXO, senão o circuito não conta. Atenção também para os pinos de alimentação do CI 7490: +Vcc=pin 5 e GND=pin 10. Se a montagem foi bem sucedida, o circuito deve contar de 0 à 99 e reiniciar automaticamente.
- 2) Modifique o circuito anterior para o que é mostrado na figura 6.2(b). Supõe-se que este novo circuito deveria resultar num divisor de frequência por 38 (medido no ponto marcado como F_OUT ou saída Q1 do segundo CI 7490).
- 3) Aumente a frequência da entrada de clock (em comparação ao adotado anteriormente) para F_IN = 3800 Hz e com auxílio de um osciloscópio, comprove a frequência e ciclo de trabalho presentes na saída deste circuito.
Freq. de saída (F_OUT) = _____, Ciclo de trabalho = ____(%).
- 4) Se tudo foi montado corretamente deve ter sido percebido que o circuito da figura 6.2(b) não divide a frequência de entrada por 38. Por que não? Descubra o erro e corrija o circuito da figura 6.2(b).
Dica: O contador assíncrono 74LS90 possui 2 módulos internos: o primeiro que divide a frequência de entrada por $\div 2$ e segundo que $\div 5$; se os 2 são interligados, gera-se um circuito divisor (de frequência) por 10, ou seja, forma-se um contador decimal que gera código BCD na sua saída (e não código binário). Por isto mesmo, o resultado de 2 CIs 7490 cascataeados pode ser acompanhado em 2 módulos display de 7-segmentos (um para unidades e outros para dezenas). Repare então como é representado o número 38 no código BCD e como o mesmo é representado na base binária:
 $38_{(10)} = \underline{\hspace{2cm}}_{(2,BCD)}$ e $38_{(10)} = \underline{\hspace{2cm}}_{(2)}$.

6.4 Questões

- a) Apresente o diagrama elétrico do circuito que corrige o erro contido na figura 6.2(b).
- b) Apresente um **outro** (um segundo) diagrama elétrico, diferente do anterior, também capaz de dividir a frequência de entrada por 38 usando $2 \times$ CIs 7490.
- c) Em relação ao item (a) anterior, preveja o ciclo de trabalho presente na saída deste circuito. Apresente a dedução (cálculos). *Dica:* uma tabela da contagem realizada pelo segundo CI 7490 (o responsável pelas dezenas) pode ser usado para contabilizar o número de pulsos de clock ingressados no estágio anterior (CI 7490 responsável pelas unidades) e assim

pode-se facilmente contabilizar o número de pulsos de clock nos quais a saída Q1 do segundo CI 7490 permaneceu em nível lógico ALTO.

- d) Para o circuito **corrigido** da figura 6.2(b), introduza uma frequência de entrada de 3800 Hz e com auxílio do osciloscópio meça:
Freq. de saída: _____(Hz); e Ciclo de trabalho: _____(%).
- e) O circuito da figura 6.2(b) funcionaria sem modificações se ao invés de CIs 74LS90 fossem usados CIs 74LS93 (contador assíncrono binário de 4-bits; primeiro estágio $\div 2$ e segundo estágio $\div 8$)? Explique porque funcionaria.

Fim.

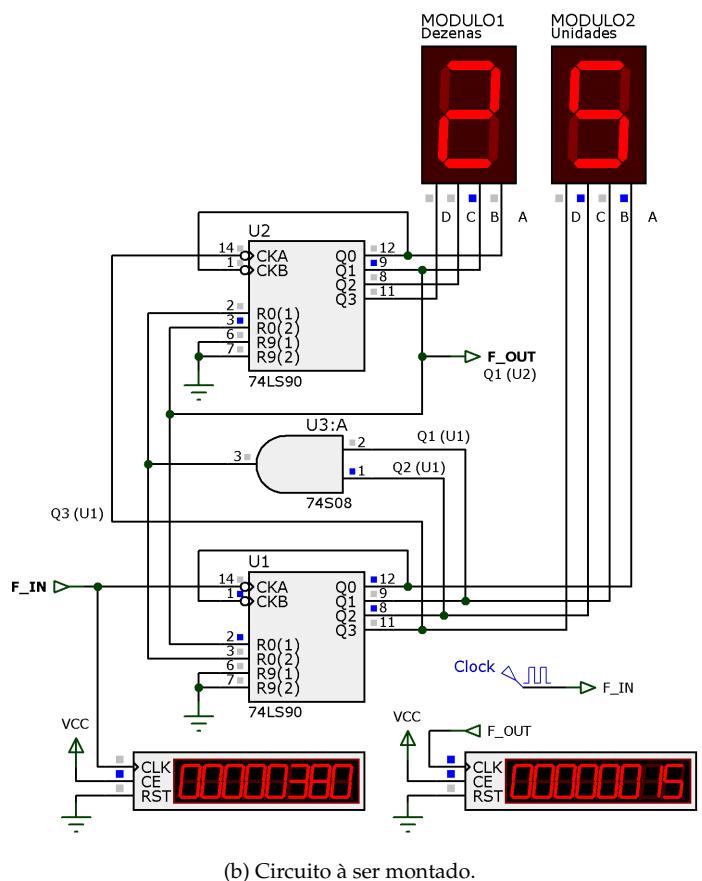
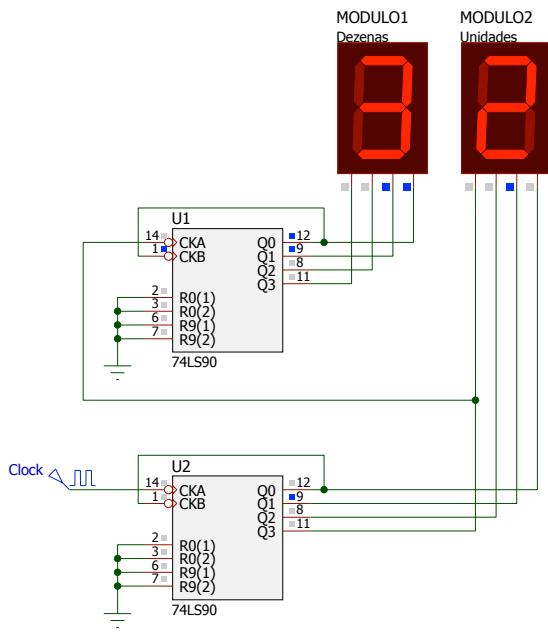


Figura 6.2: Circuitos previstos neste laboratório.

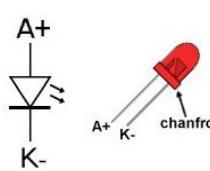
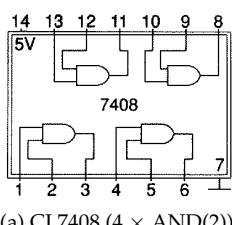


Figura 6.3: Pinagens de outros CIs e componentes auxiliares

7 | Aplicação de Ripple Counter

Listade Material

- 1× CI 74LS74 (FF-D, borda ↑);
- 1× CI 74LS93 (Contador assíncrono binário 4-bits, borda ↓);
- 1× CI 74LS00 (4× NAND(2));
- 1× Resistor de $100\ \Omega$;
- 1× Resistor de $1\ K\Omega$;
- 1× Capacitor $1\mu F @ 10\ Volts$;
- 2× push buttons NA;
- 1× Módulo Display.

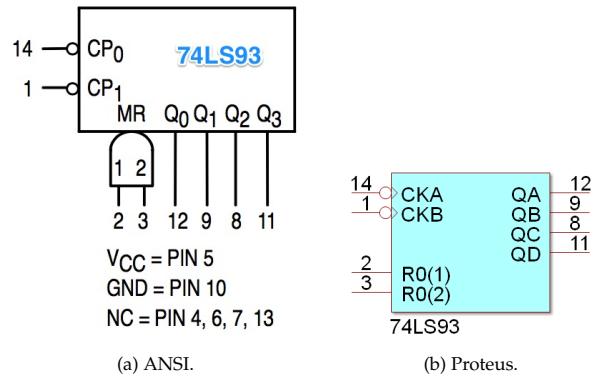


Figura 7.1: Pinagem do CI 74LS93.

7.1 Objetivos

Testar uma aplicação prática de contador assíncrono binário de 4-bits e que faz uso de portas lógicas básicas + Flip-Flop Tipo D para controlador estados distintos que o circuito contador pode assumir.

7.2 Fundamentos Teóricos

Neste laboratório será usado o CI 74LS93, um contador assíncrono binário de 4-bits. A figura 7.1 mostra o símbolo lógico desta pastilha que é formada por 2 estágios contadores assíncronos crescentes. O primeiro é um simples divisor da frequência de entrada por 2 (entrada de clock em CP0 e saída em Q0). O segundo estágio é composto por 3 flips-flops organizados numa estrutura de contador assíncrono crescente de Módulo-8 (entrada de clock em CP1 e saídas: Q3 Q2 Q1, onde Q3=MSB). Os dois estágios de contagem deste CI se encontram separados. Para a aplicação pretendida neste laboratório, os 2 estágios serão interligados, conectando-se a saída Q0 do primeiro estágio à entrada CP1 do segundo estágio, compondo assim uma estrutura que seria capaz de contar de 0 até $F_{(16)}$.

Já a pastilha 74LS74 é um simples flip-flop tipo D, ativado por borda de subida, com entradas assíncronas para forçar Set (pino \bar{S} , ativo baixo) ou Reset (pino \bar{R} , ativo baixo). Este biestável será usado para comutar entre 2 estados distintos no circuito à ser montado neste laboratório: 1) quando a saída $\bar{Q} = 1$, o mesmo permanece “parado” (estado “PAUSE”) enquanto o usuário não pressiona um botão de START e 2) quando a saída $\bar{Q} = 0$, o circuito fica liberado para contagem depois que o usuário pressiona o botão de START (estado “RUN”). Mas o circuito como um todo possui 3 estados distintos – ver figura 7.2.

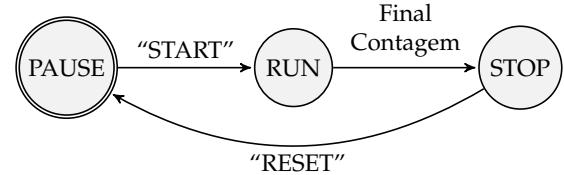


Figura 7.2: Diferentes estados do circuito de “Aplicação de Ripple Counter”.

Note que para garantir que o circuito como um todo inicie num determinado estado (no caso, “PAUSE”, com $Q3\ Q2\ Q1\ Q0 = 0000_{(2)}$) será usado um circuito RC para o “power-up reset”. Notar que quando o circuito é alimentado, o capacitor ainda se encontra descarregado e passado um curto intervalo de tempo, proporcional a constante RC, o capacitor se carrega até VCC . Com isto conseguimos gerar um pulso de subida no ponto entre o resistor e o capacitor; este ponto permanece em nível lógico baixo durante um curto espaço de tempo, tão logo o circuito é alimentado.

7.3 Parte Prática

O circuito à ser montado aparece na figura 7.3. Uma vez montado o mesmo, um sinal de clock (onda quadrada, padrão TTL) deve ser aplicada à sua entrada CLOCK.

Note que:

1. Uma vez que for estabelecida a alimentação neste circuito, o estágio de power-up reset garante que o mesmo sempre inicie no estado “PAUSE” e assim permaneça até que o usuário pressione o botão START.

2. Uma vez pressionado o botão de **START** o circuito inicia a contagem (passa para o estado “RUN”). Reparar que na transição de **PAUSE** para **RUN**, o CI 74LS93 sai do estado de **Reset Assíncrono** e que o sinal de **CLOCK** é liberado para o primeiro estágio do CI contador 74LS93, pino **CP0**.
3. Uma vez que o CI contador (74LS93) atinge determinado ponto de contagem (evento denominado “Final Contagem”), o circuito passa para o estado ‘**STOP**’. O “Final Contagem” é determinado pela lógica formada pela porta **NAND** (U2:C) que interrompe o sinal de **CLOCK** que antes estava chegando ao pino **CP0** do CI contador 74LS93, o que força este CI à deter sua contagem e permanecer no novo estado “**STOP**”.
4. O circuito permanece no estado “**STOP**” até que o usuário resolva pressionar o botão de **RESET**. O botão de **RESET** força o **reset** do FF-F (74LS74), cuja saída \bar{Q} é usada para manter (ou não) o circuito como um todo no estado “**PAUSE**” – o CI contador 74LS93 permanece no estado de **reset assíncrono** (“**PAUSE**”) ou de contagem (“**RUN**”).

7.4 Questões

Pede-se que os seguintes itens sejam respondidos em sala de aula:

- 1) Completar as formas de onda (figura 7.4) que podem ser previstas para o funcionamento deste circuito (figura 7.3).

Obs.: Notar que este circuito pode ser simulado no Proteus, incluindo o “pressionar” dos botões de **START** e **RESET** e assim o diagrama de formas de onda pode ser obtido; mas neste caso, temos que “simular” o *power-up reset*, e o pressionar dos botões “**START**” e “**RESET**”. Para isto, apenas para efeitos de simulação, foram criados portas extras de entrada no circuito: **ST**, **RST_SIM** e mais o sinal de **CLOCK**. O sinal de **CLOCK** é simplesmente conectado a gerador de sinais do tipo **DCLOCK**, cuja frequência foi ajustada para 2 Hz. Já o pressionar do botão **START** é simulado através de um sinal do tipo **DPULSE** que gera um sinal que inicia em nível lógico baixo, vai à nível lógico alto depois de 1,15 segundos, permanece neste nível lógico por 150 ms e depois este sinal

permanece o restante do tempo em nível lógico baixo. Assim, é simulado o pressionar do botão **START** passados 1,15 em que o circuito foi alimentado; e o botão teria sido acionado pelos 150 ms. Já o sinal **RST_SIM** é um pouco mais complexo de simular, exigindo o emprego de uma lógica extra, uma porta básica **AND**. Para simular o **Reset** inicial do FF-D, CI 74LS74, ocasionado pelo carregamento do capacitor C1, foi empregado o sinal do tipo **DEDGE** que inicia em nível lógico baixo nos primeiros 150 ms e depois permanece o restante do tempo em nível lógico alto. Estes 150 ms iniciais em nível lógico baixo simulam o tempo que o capacitor leva para se carregar e atingir uma tensão interpretada como nível lógico alto pelo FF-D, CI 74LS74. Também foi usado ainda um sinal do tipo **DPULSE** para simular o usuário pressionando o botão de **RESET** passados 7 segundos em que o circuito foi alimentado. Note que este sinal **DPULSE** realiza a transição: alto → baixo → alto, iniciando em nível lógico alto, permanecendo neste nível nos primeiros 7 segundos e somente então, o mesmo comuta para nível lógico baixo, mas apenas durante 150 ms simulando assim o pressionar do botão **RESET**, 7 segundos depois que o circuito foi ligado.

Na prática, notar que este circuito sempre iniciará resetado e assim permanecerá até que o usuário pressione o botão **START**. Para continuar testando este circuito, basta o usuário pressionar o botão de **START**, deixar passar um certo intervalo de tempo para perceber em que número de contagem o circuito se deterá e então, para recomeçar o ciclo, basta o usuário pressionar o botão **RESET**.

Note que este circuito não é “regatilhável”, isto é, uma vez pressionado o botão de **START** o mesmo não reinicia novamente a sua contagem. A única forma de reiniciar a contagem, em qualquer instante de tempo é pressionando o botão de **RESET**.

- 2) Uma vez que o circuito tenha sido corretamente montado e constatado seu funcionamento, alterar o mesmo para que a contagem se detenha no número 5 – apresente o novo diagrama elétrico referente à este modificação ou descreva que ligações devem ser desfeitas e que novas ligações devem ser realizadas.

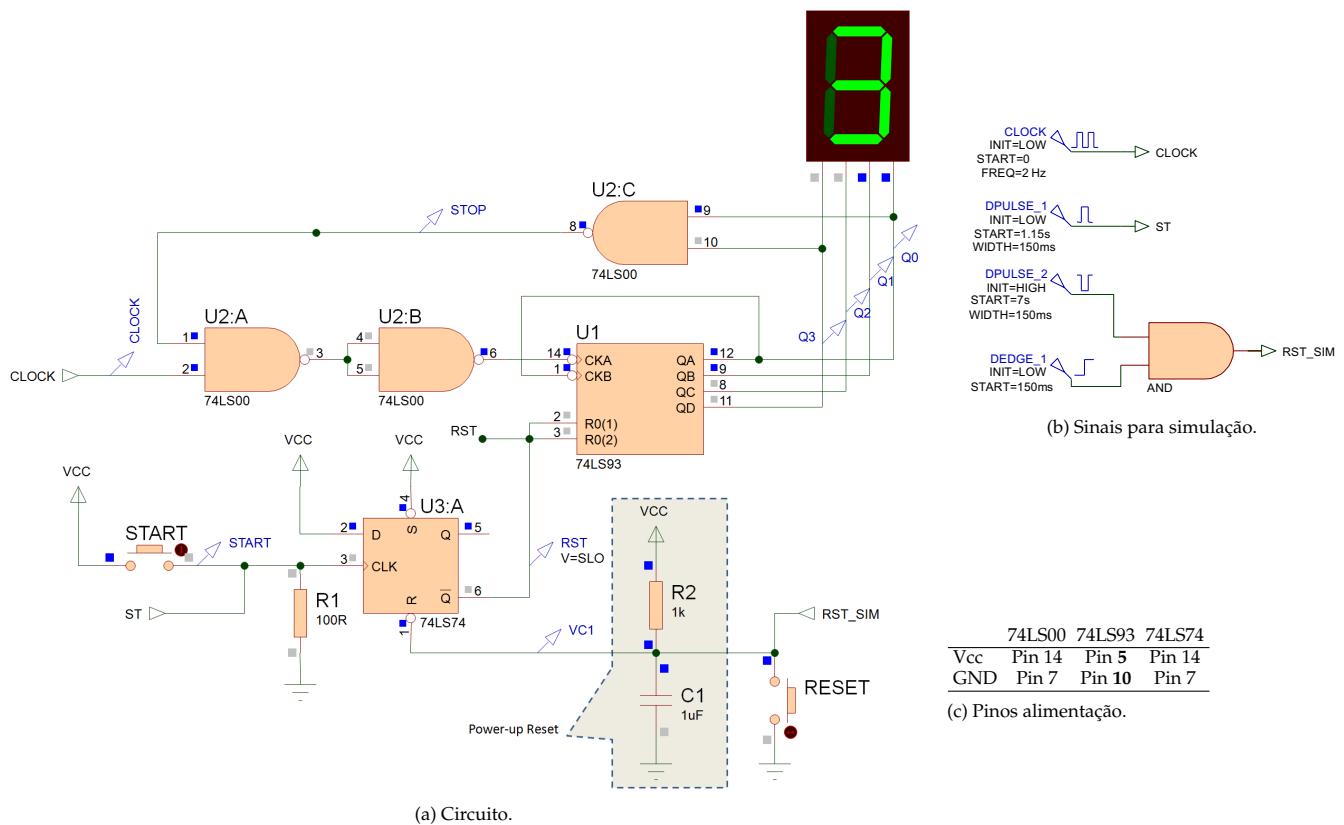


Figura 7.3: Circuito à ser montado neste laboratório.

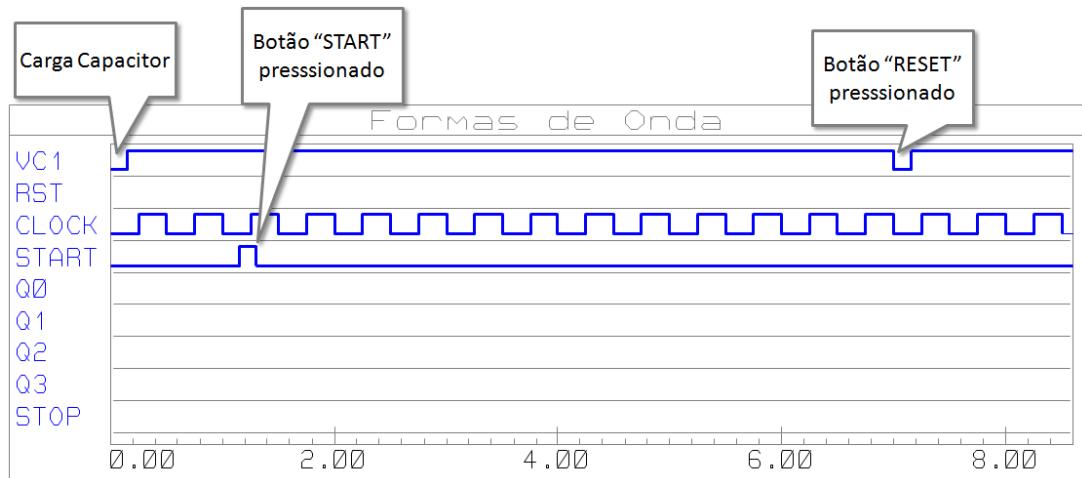


Figura 7.4: Formas de onda previstas para circuito deste laboratório (completar).

8 | O Contador Síncrono 74190

ListadeMaterial

- 1 × CI 74LS190 (contador síncrono decimal ↑);
- 1 × CI 74LS02 (4 × NOR(2));
- 1 × resistor de 10 KΩ;
- 1 × capacitor eletrolítico $10\mu\text{F} \times 10\text{V}$;
- 1 × push-button NA mini;
- 1 módulo de display de 7 segmentos;
- gerador de sinais + fonte 5 V;

8.1 Objetivo

Explorar o funcionamento e modo de operação do CI contador síncrono integrado 74LS190: seus dois modos de contagem bidirecional, sua carga paralela de dados e indicadores de final de contagem.

8.2 Fundamentos Teóricos

O CI 74190 é um contador decimal (saída BCD) **síncrono** bidirecional (up/down) de 4 bits, ativado por borda de subida. As mudanças de estado são sincronizadas com a transição BAIXO-para-ALTO aplicadas na entrada *Clock*.

Características principais do 74LS190:

- Baixo consumo (Tipicamente 90 mW de dissipação)
- Alta velocidade (frequência típica de contagem: 25 MHz)
- Contagem síncrona
- Carga paralela assíncrona
- Entradas de ajuste (*preset*) individuais
- Entradas para habilitar a contagem e controlar contar crescente ou decrescente
- Cascateável
- Diodos de saturação (*clamp diodes*) limitam os efeitos de transição de sinais em altas velocidades

A pinagem e símbolo lógico do CI 74190 aparece mostrado na fig. 8.1.

Nomes dos pinos do CI 74LS190:

\overline{CE}	Habilitação da contagem (<i>Count Enable</i>), entrada (ativo BAIXO)
CP	Entrada de Clock, ativado por borda de subida
$\overline{U/D}$	Entrada de controle da contagem (<i>Up/Down</i>)
\overline{PL}	Entrada (ativo BAIXO) assíncrona para realizar carga paralela de dados
P_n	Entradas paralelas de dados para o <i>preset</i>
Q_n	Saídas dos Flip-Flops internos do contador
\overline{RC}	Saída para <i>Ripple Clock</i>
TC	Terminal de Saída indicando fim da contagem (<i>Terminal Count</i>)

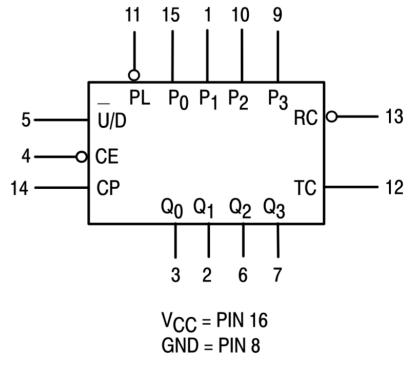


Figura 8.1: Símbolo lógico e pinagem do CI 74LS190.

A figura 8.2 mostra o diagrama de sequência de estados possíveis do CI 74LS190.

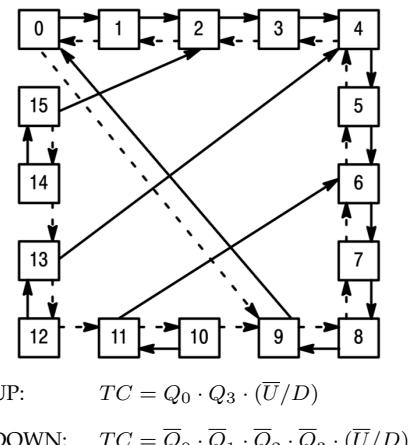


Figura 8.2: Diagrama de estados do CI 74LS190.

A entrada **assíncrona** \overline{PL} (*Parallel Load*) sobre-escreve a contagem e força a carga dos dados presentes nas entradas P_n , isto permite usar este CI como um contador programável. A

entrada \overline{CE} (Count Enable) serve como uma entrada de *carry-in*/emprestimo no caso de contadores cascadeados. A entrada de controle $\overline{U/D}$ (Up/Down) determina quando o CI conta de maneira crescente ou decrescente. A saída TC (Terminal Count) e a saída \overline{RC} (Ripple Clock) fornecem uma indicação de “overflow/underflow” e tornam possível vários métodos para geração de sinais de carry/emprestimo no caso de aplicações com contadores de diversos estágios (cascateamento).

A direção da contagem é determinada pelo sinal ($\overline{U/D}$ =Count Control Input), como indicado na tabela 8.1 (seleção do modo).

Entradas				MODE
PL	\overline{CE}	$\overline{U/D}$	CP	
1	0	0	↑	Cont Up
1	0	1	↑	Cont Down
0	X	X	X	Parallel Load (Asyn.)
1	1	X	X	No change (Hold)

Tabela 8.1: Tabela de Seleção do Modo.

O sinal \overline{CE} (Count Enable) pode baixar de nível lógico em qualquer instante do pulso de clock. Entretanto para a contagem ser corretamente desabilitada, recomenda-se que a transição BAIXO para ALTO ocorra somente quando o pulso de *Clock* estiver em nível ALTO. De maneira similar o sinal $\overline{U/D}$ somente deve ser modificado quando o pino \overline{CE} estiver em nível ALTO. Um *Clock* falso pode ocorrer se a entrada $\overline{U/D}$ mudar de estado enquanto o *Clock* estiver em nível BAIXO.

Duas saídas estão disponíveis para cascateamento: as saídas \overline{RC} =Ripple Clock Output e TC =Terminal Count Output. A saída \overline{RC} fornece um pulso negativo, com duração aproximada de um ciclo completo de clock, quando ocorre um *overflow* (após o estado 10 na sequência *Up*) ou *underflow* (após o estado 0 na sequência *Down*). A saída TC =Terminal Count, permanece normalmente em nível baixo e sobe de nível quando o circuito alcança ou o zero ou o máximo (9). Este sinal não deve ser utilizado como um sinal gerador de pulsos de clock pois está sujeito a “spikes” resultantes da decodificação interna dos pulsos de entrada de clock. O sinal TC é internamente utilizado para gerar o sinal \overline{RC} =Ripple Clock. A saída \overline{RC} permanece normalmente em nível alto. Quando \overline{CE} está em nível baixo e TC está em nível alto, a saída \overline{RC} muda para nível baixo quando o pulso de clock também passa para nível baixo e assim permanece até que o pulso de clock suba. Esta forma de operação simplifica o projeto de contadores de múltiplos estágios, como mostram as figs. 8.3 (solução assíncrona entre CIs) e fig. 8.4 (solução síncrona entre CIs).

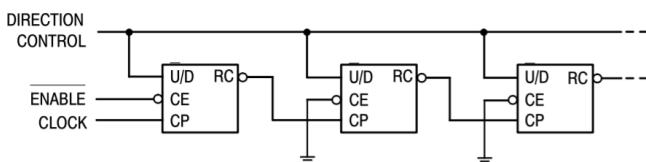


Figura 8.3: Estágio de n -ésima contagem usando pino \overline{RC} .

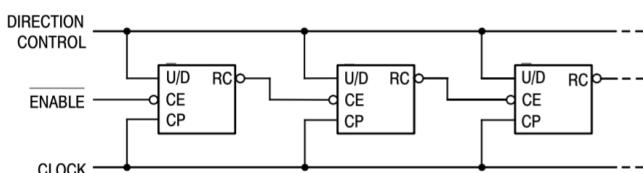
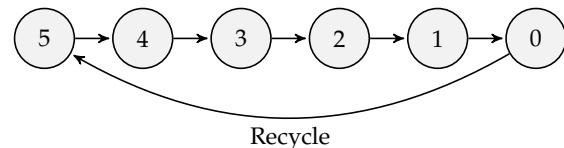


Figura 8.4: Estágio de n -ésima contagem usando pino \overline{RC} /borrow.

A figura 8.7 mostra um sequencia típica de operação do 74LS190 (extraído de material da Texas Instruments).

8.3 Parte Prática

- Ajuste o CI 74LS190 para contar de maneira crescente de 0 à 9. Apresente o diagrama elétrico deste circuito.
- Ajuste o CI 74LS190 para contar de maneira decrescente de 9 à 0. Apresente o diagrama elétrico deste circuito. Note que basta apenas modificar o nível lógico na entrada $\overline{U/D}$.
- O que acontece com o circuito (em qualquer um das versões anteriores), se o pino TC (Terminal Count) for conectado ao pino \overline{RC} Ripple Clock output?
- O que acontece se você acrescentar um circuito de *power-up* (rede RC com push-button) para forçar um *preset* no circuito em 5₍₁₀₎? Dica: Você pode ser inspirar em parte do circuito apresentado na figura 8.6 para resolver o problema.
- Monte o circuito que aparece na figura 8.6. Note que trata-se de um contador MOD-5, ou seja, capaz de dividir a frequência de entrada por 5. Inicialmente aplique uma onda quadrada de 5 Hz na entrada de *CLOCK* para comprovar o correto funcionamento do circuito. Depois aumente a frequência para 5 KHz e use um osciloscópio para medir a frequência de saída em *F_OUT* e aproveite para medir o ciclo de trabalho. Note que este circuito realiza um ciclo de contagem semelhante ao mostrado na próxima figura:



A figura 8.5 mostra num diagrama de formas de ondas, o funcionamento do circuito do item (c).

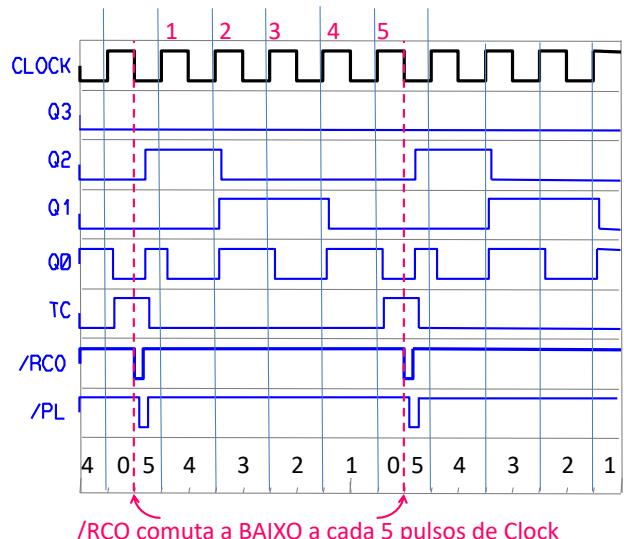


Figura 8.5: Formas de onda esperadas para circuito do item (c).

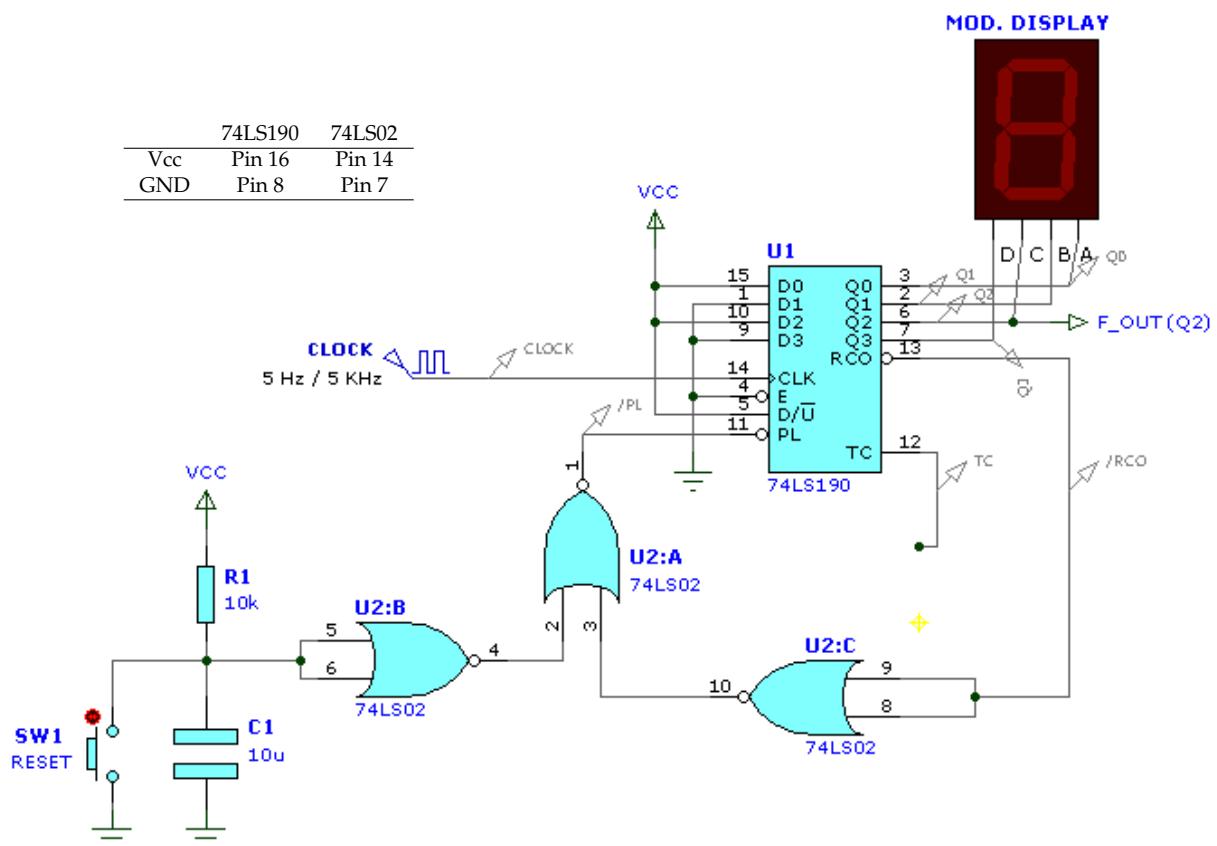


Figura 8.6: Montagem relativa ao item (c) da parte prática deste laboratório.

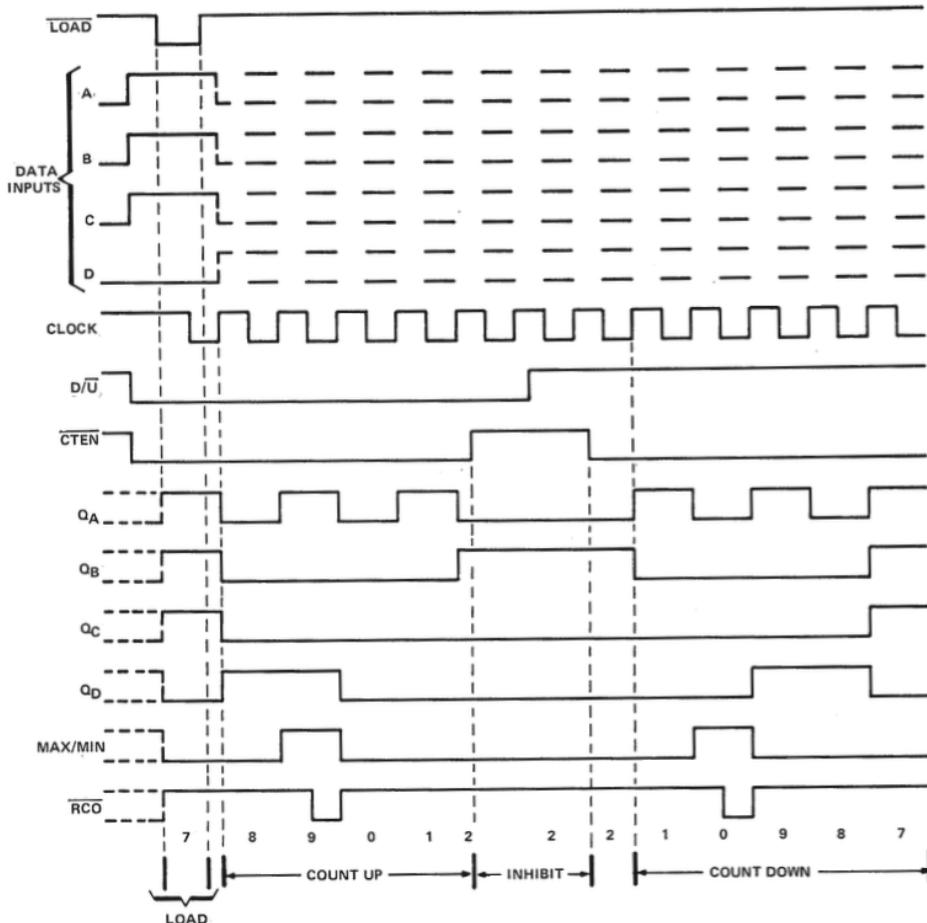


Figura 8.7: Sequencia exemplo de operação e uso do CI 74LS190.

1. Carga de dados (paralela, pino $\overline{LOAD} = 0$ ou pino \overline{PL} (Parallel Load) no Proteus) ou *preset* com código BCD igual à 7;
2. Conta crescente (pino $D/\bar{U} = 0$) até 8, nove (ponto de máximo atingido, notar mudança de nível lógico no pino MAX/MIN ou TC (Terminal Count) no caso do Proteus), 0, 1 e 2;
3. É desabilitado (pino \overline{CTEN} ou \overline{CE} no Proteus colocado em nível lógico ALTO);
4. Volta à ser habilitado e ajustado para contar decrescente (pino $D/\bar{U} = 1$) até 1, 0 (ponto de mínimo atingido), 9, 8 e 7.

Universidade de Passo Fundo
Lab. de Circuitos Digitais II

Engenharia Elétrica
Prof. Fernando Passold

Aluno(a)₁: _____ Lab 9) O Contador Síncrono 74190

Aluno(a)₂: _____ Data: ____ / ____ / ____

8.4 Perguntas

Obs.: Entregar até o final da aula:

- 1) Apresente o **diagrama elétrico** esperado para a Parte Prática, item (a).
- 2) Apresente o **diagrama elétrico** esperado para a Parte Prática, item (b).
- 3) Referente ao item (c) da Parte Prática, explique o que aconteceu quando o pino *TC* (*Terminal Count*) foi conectado ao pino *RC* (*Ripple Clock Output*)?

- 4) Apresente o **diagrama elétrico** do circuito referente ao item (d) da Parte Prática.
- 5) Explique com suas próprias palavras o funcionamento do circuito da Parte Prática, item (e).

- 6) Relativo ao item (e) da Parte Prática, complete:

1. Frequência de entrada: _____
2. Frequência de saída: _____
3. Ciclo de trabalho (em %): _____

- 7) No item (e) da Parte Prática: – Você percebeu que a transição do número 0 para o 5 ocorre ainda dentro de um ciclo do sinal de *Clock*? Isto prejudica a divisão em frequência sendo realizada?

- 8) Como o circuito apresentado na figura 8.6 deveria ser modificado para dividir a frequência de entrada por 8!? Apresente o **diagrama elétrico** deste circuito.



Respostas

- 1) Diagrama elétrico da parte prática (a): Ajustar o CI 74LS190 para contar de maneira crescente de 0 à 9.

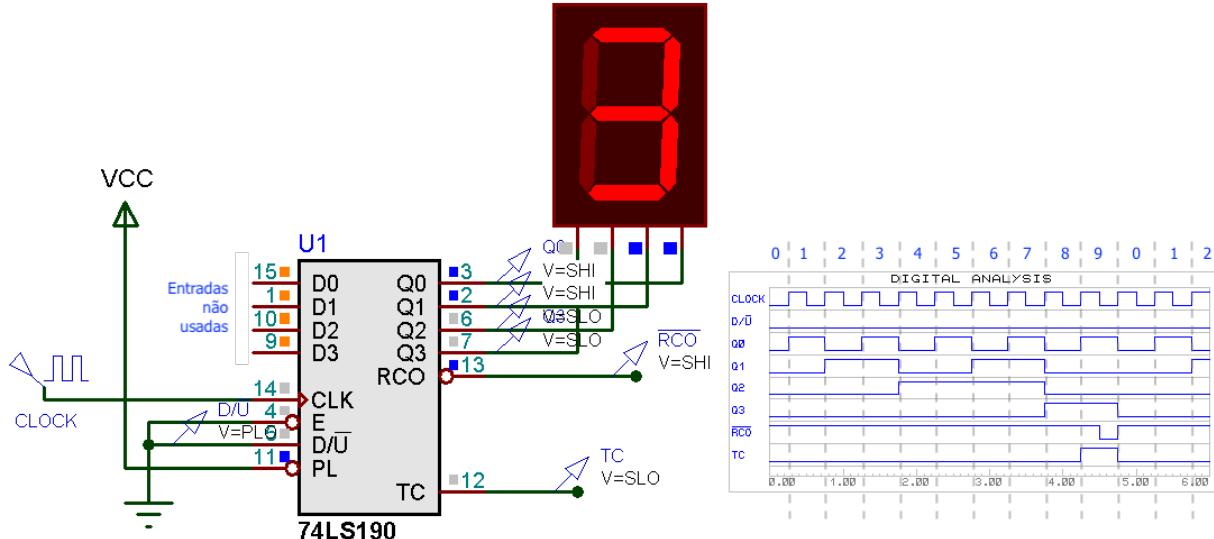


Figura 8.8: Contador 74190 configurado para contagem crescente de 0 à 9.

- 2) Diagrama elétrico da parte prática (b): Ajuste o CI 74LS190 para contar de maneira decrescente de 9 à 0.

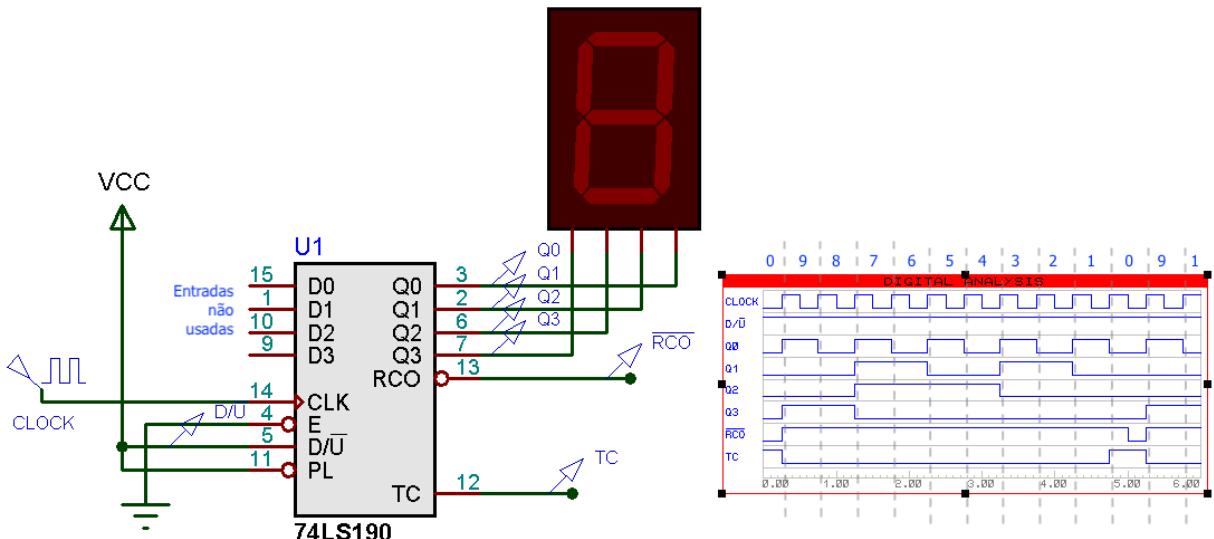


Figura 8.9: CI 74190 configurado para contar decrescente de 9 até 0.

- 3) Referente ao item (c) da parte prática, o que acontece quando o pino *TC* (*Terminal Count*) foi conectado ao pino \overline{RC} *Ripple Clock Output*?

Resp.: Nada acontece. CI permanece “parado”... Erro na formulação da questão.

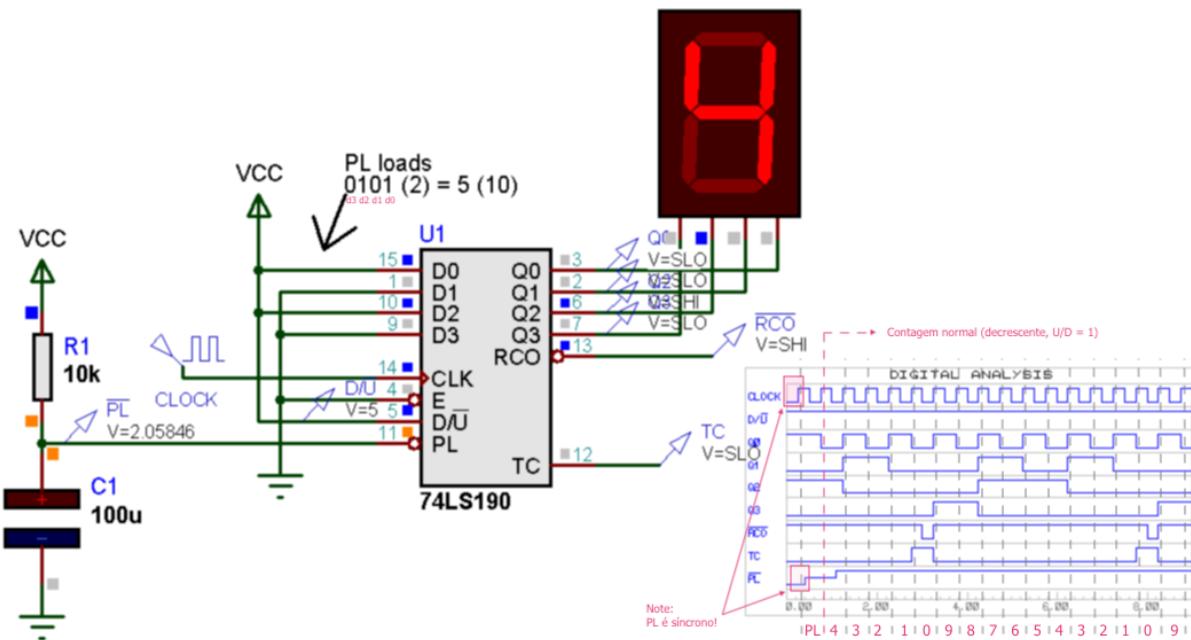
Questão Anulada

- 4) Diagrama elétrico da parte prática (d):

Questão: O que acontece se você acrescentar um circuito de *power-up* (rede RC com push-button) para forçar um *preset* no circuito em $5_{(10)}$?

Resp.: O circuito apenas inicia em $5_{(10)}$ e depois volta à sua sequência normal de contagem. No caso da figura abaixo:

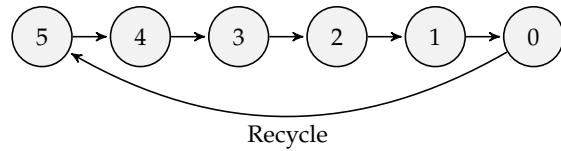
$5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 9 \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow \dots$

Figura 8.10: CI 74190 configurado para *Parallel Load* com 5.

5) Relativo ao item (e) da parte prática (circuito da 8.6):

Notas:

- Notar que trata-se de um contador MOD-5, ou seja, capaz de dividir a frequência de entrada por 5. Inicialmente aplicar uma onda quadrada de 5 Hz na entrada de *CLOCK* para comprovar o correto funcionamento do circuito.
- Depois aumentar a frequência para 5 KHz e usar um osciloscópio para medir a frequência de saída em *F_OUT*; aproveitar para medir o ciclo de trabalho.
- Note que este circuito realiza um ciclo de contagem semelhante ao mostrado na próxima figura:



Resp.: A própria figura 8.5 destaca que a cada 5 pulsos de clock, ocorre um *recycle* no contador. Circuito com formas de onda e frequências de deveriam ser medidas, aparece na figura 8.11.

Pela figura anterior, como *F_OUT* é extraído à partir de *Q₂*, percebe-se que *Q₂* = 1 durante pouco mais de 1 ciclo de clock à cada 5 pulsos de clock que ingressam no CI (quase 1,5). Então:

$$\text{Duty-cycle} = \frac{1,5}{5} = \frac{3}{10} = 0,3 \text{ ou } 30\%$$

1. Frequência de entrada: **5 KHz** (\leftarrow Notar que *F_OUT* é extraído à partir de *Q₂*)
2. Frequência de saída: **1 KHz**.
3. Ciclo de trabalho (em %): **28%**. (Saída *Q₂* fica em nível lógico alto por aprox. 280ns a cada ciclo de clock, 1 μ s)

6) Referente ao item (e) da parte prática: – Você percebeu que a transição do número 0 para o 5 ocorre ainda dentro de um ciclo do sinal de *Clock*? Isto prejudica a divisão em frequência sendo realizada?

Resp.: Não prejudica. É formado um padrão de forma de onda que se repete à cada 5 pulsos de clock, o que caracteriza a divisão da frequencia de entrada por 5.

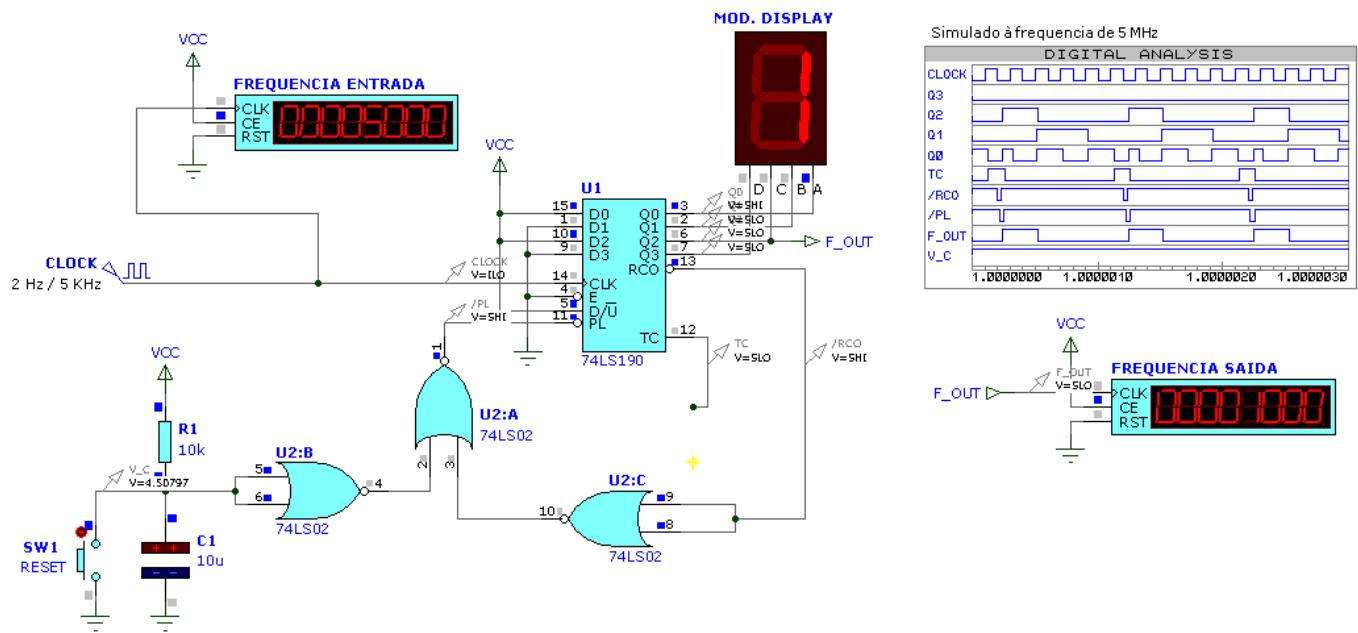


Figura 8.11: Circuito do laboratório com frequencímetros e formas de onda.

- 7) Modificação no circuito da figura 8.6 para dividir a frequência de entrada por 8. Incluir diagrama elétrico:
Resp.: Basta realizar *Parallel Load* com $8_{(10)} = 1000_{(2)}$. Ver figura 8.12 à seguir.

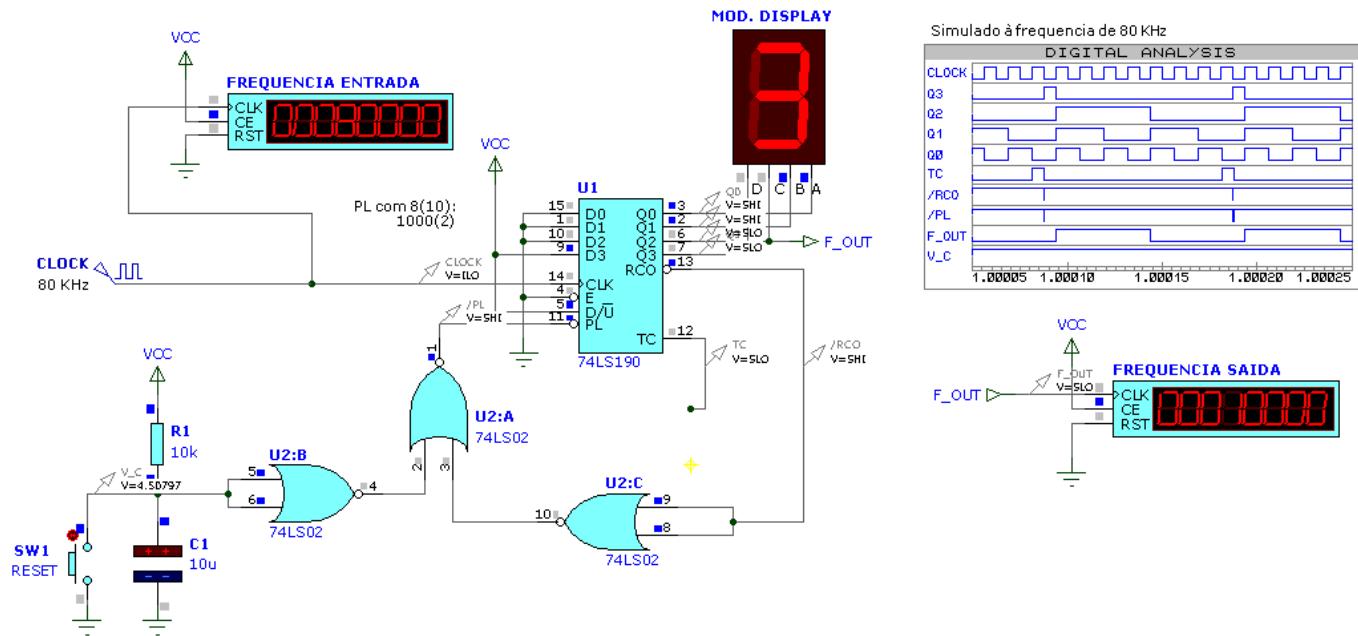


Figura 8.12: Circuito divisor de frequencia por 8, com freq. de entrada (Clock = 80 KHz).

9 | Aplicação de Contador Síncrono 74190

Listar de Material

- 1 × CI 74LS190 (contador síncrono decimal ↑);
- 1 × CI 74F14 (Schmidt-trigger);
- 1 × CI 74LS112 (2 × FF-JK ↓);
- 1 × resistor de 1 KΩ;
- 1 × capacitor eletrolítico 1μF × 10V;
- 1 módulo de display 7 segmentos;
- gerador de sinais + fonte 5 V;

9.1 Objetivo

Explorar o uso de um contador síncrono integrado (74190): seus dois modos de contagem bidirecional, sua carga paralela de dados e indicador de final de contagem.

9.2 Fundamentos Teóricos

O CI 74190 é um contador decimal (saída BCD) **síncrono** bidirecional (up/down) de 4 bits, ativado por borda de subida. Cada CI contém 4 flips-flops do tipo master-slave, com portas lógicas permitindo programação individual (pino \overline{PL} =parallel load), e contagem ascendente e descendente (pino \overline{U}/D =Up/Down). Cada circuito possui capacidade de carregamento assíncrona paralela que permite que a saída do contador seja ajustado para qualquer número desejado.

Quando a entrada *Parallel Load* (\overline{PL}) está em nível baixo, a informação presente nas entradas de dados paralelas ($P_0 - P_3$ ou $D_0 - D_3$) é carregada no contador e surge nas saídas Q_i ($i = 0 : 3$). Esta operação se sobrepõe à função de contagem, como indicado na tabela de seleção do modo (tabela 9.1). Somente nível baixo na entrada (\overline{CE} =Count Enable) habilita a contagem. A mudança de estados interna é iniciada sincronicamente na transição de subida do sinal de clock (CP). A direção da contagem é determinada pelo sinal (\overline{U}/D =Count Control Input), como indicado na tabela 9.1 (seleção do modo).

Entradas				MODE
\overline{PL}	\overline{CE}	\overline{U}/D	CP	
1	0	0	↑	Cont Up
1	0	1	↑	Cont Down
0	X	X	X	Parallel Load (Asyn.)
1	1	X	X	No change (Hold)

Tabela 9.1: Tabela de Seleção do Modo.

O sinal \overline{CE} pode baixar de nível em qualquer instante do pulso de clock. Entretanto para a contagem ser corretamente desabilitada, recomenda-se que a transição BAIXO → ALTO ocorra somente quando o pulso de clock estiver em nível ALTO. De maneira similar o sinal \overline{U}/D somente deve ser modificado quando o pino \overline{CE} estiver em nível ALTO. Um clock falso pode ocorrer se a entrada \overline{U}/D mudar de estado enquanto o clock estiver em nível BAIXO.

Duas saídas estão disponíveis para cascataamento: as saídas \overline{RC} =Ripple Clock Output e TC =Terminal Count Output. A saída \overline{RC} fornece um pulso negativo, com duração aproximada de um ciclo completo de clock, quando ocorre um *overflow* (após o estado 10 na sequência up) ou *underflow* (após o estado 0 na sequência down). A saída TC =Terminal Count, permanece normalmente em nível baixo e sobe de nível quando o circuito alcança ou o zero ou o máximo (9). Este sinal não deve ser utilizado como um sinal gerador de pulsos de clock pois está sujeito a “spikes” resultantes da decodificação interna dos pulsos de entrada de clock. O sinal TC é internamente utilizado para gerar o sinal \overline{RC} =Ripple Clock. A saída \overline{RC} permanece normalmente em nível alto. Quando \overline{CE} está em nível baixo e TC está em nível alto, a saída \overline{RC} muda para nível baixo quando o pulso de clock também passa para nível baixo e assim permanece até que o pulso de clock suba. Esta forma de operação simplifica o projeto de contadores de múltiplos estágios, como mostra as fig. 9.1 (solução assíncrona entre CIs) e fig. 9.2 (solução síncrona entre CIs).

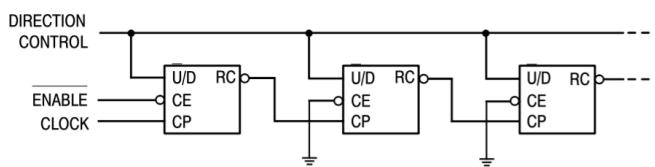


Figura 9.1: Estágio de n -ésima contagem usando pino \overline{RC} .

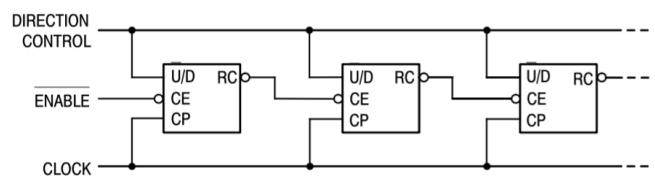


Figura 9.2: Estágio de n -ésima contagem usando pino $\overline{RC}/borrow$.

A figura 9.6 mostra um sequencia típica de operação do 74LS190 (extraído de material da Texas Instruments).

A pinagem e símbolo lógico do CI 74190 aparece mostrado na fig. 9.3.

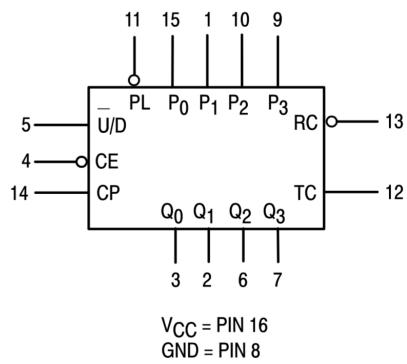


Figura 9.3: Pinagem do CI 74190.

9.3 Parte Prática

De forma a comprovar o funcionamento deste CI, será montado o circuito mostrado na figura 9.4 à seguir. Note que tal circuito faz uso de bloco responsável pela inicialização do circuito ("Power-Up reset"), constituído por um resistor e capacitor + 2 portas inversoras do tipo Schmidt-trigger. Este bloco força um estado inicial para o contador através da aplicação de um breve pulso em nível lógico baixo na entrada PL do CI 74190 justo no instante de alimentação do circuito. Outro componente auxiliar é utilizado, um FF-JK (CI 74112), na configuração *Toggle*, responsável por modificar o modo de contagem do CI 74190, toda vez que este atinge o final de contagem (indicado pelo pino TC do 74190).

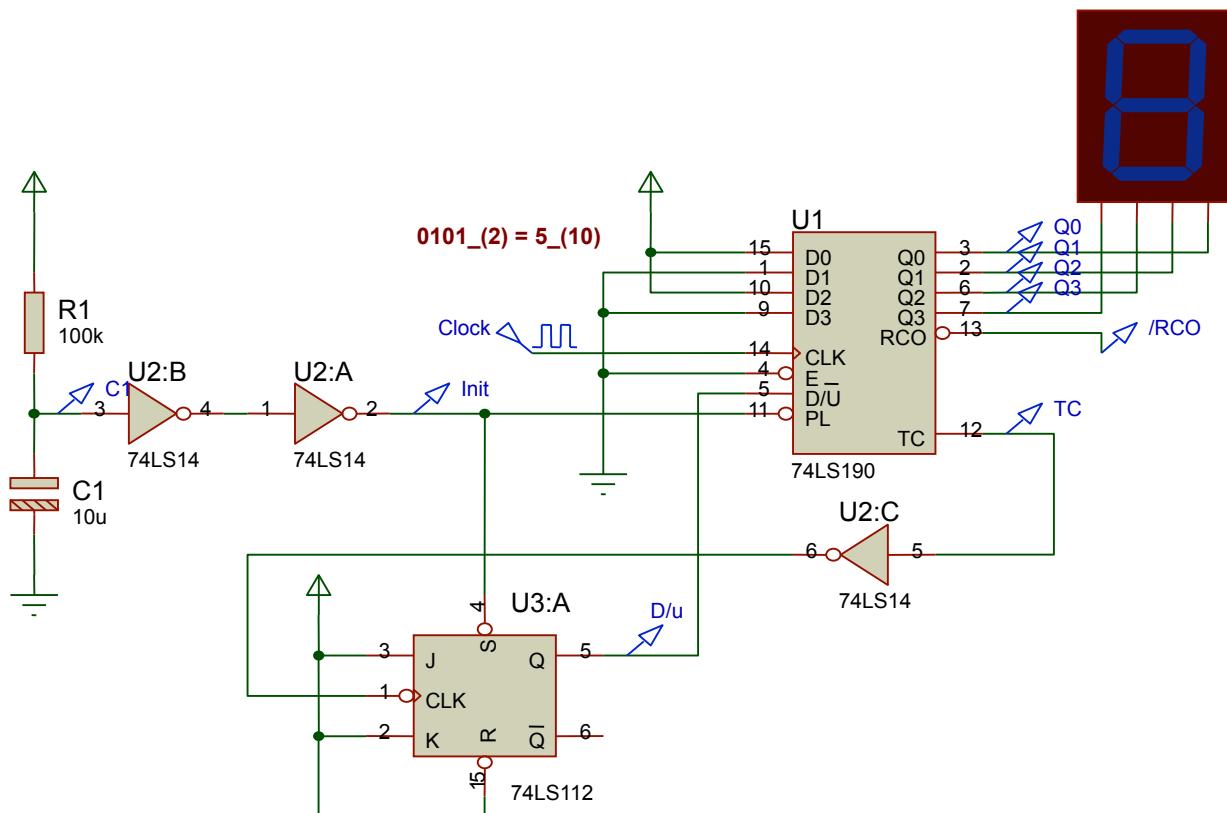


Figura 9.4: Circuito à ser montado.

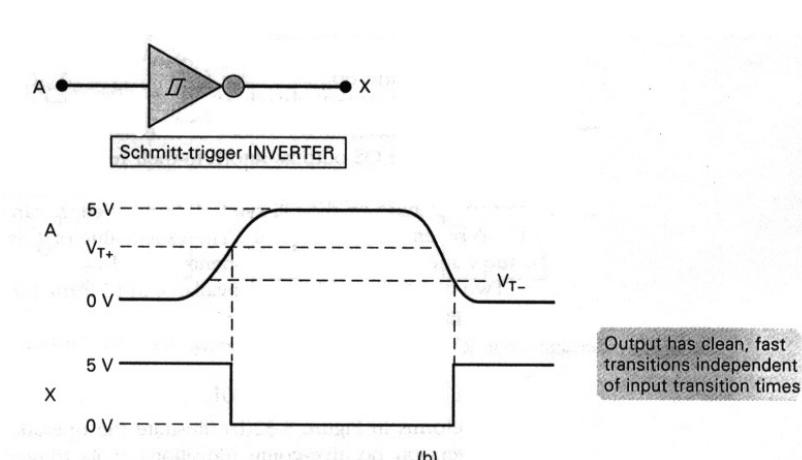
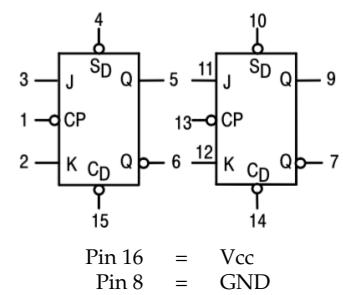
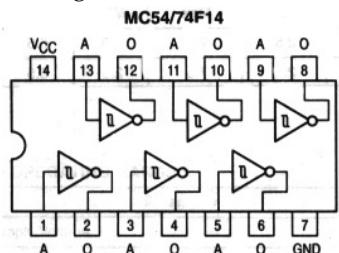


FIGURE 5-51 (a) If input transition times are too long, a standard logic device-output might oscillate or change erratically; (b) a logic device with a Schmitt-trigger type of input will produce clean, fast output transitions.

(a) Operação de um Schmitt-trigger (CI 74LS14).

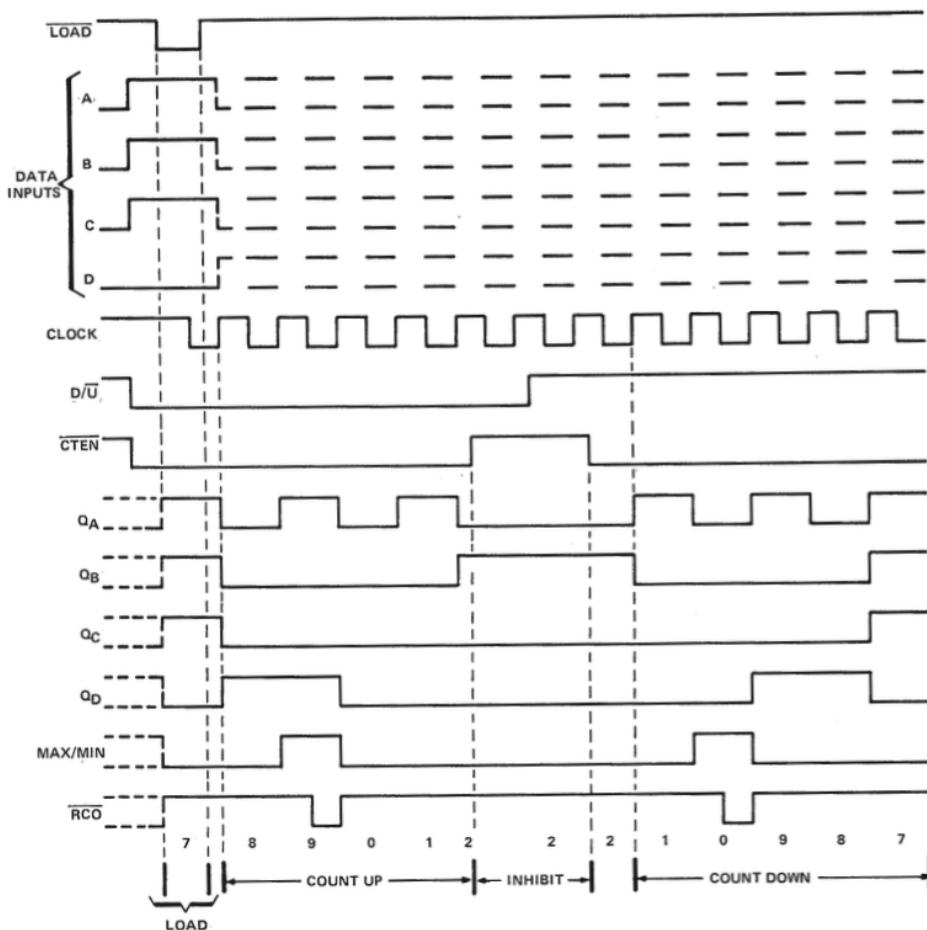


(b) Pinagem do 74LS112 (FF-JK).



(c) Pinagem do 74LS14.

Figura 9.5: Detalhes de alguns Cls.



1. Carga de dados (paralela, pino \overline{LOAD} = 0 ou pino \overline{PL} (Parallel Load) no Proteus) ou preset com código BCD igual à 7;
2. Conta crescente (pino $D/\overline{U} = 0$) até 8, nove (ponto de máximo atingido, notar mudança de nível lógico no pino MAX/MIN ou TC (Terminal Count) no caso do Proteus), 0, 1 e 2;
3. É desabilitado (pino $CTEN$ ou CE no Proteus colocado em nível lógico ALTO);
4. Volta à ser habilitado e ajustado para contar decrescente (pino $D/\overline{U} = 1$) até 1, 0 (ponto de mínimo atingido), 9, 8 e 7.

Figura 9.6: Sequencia exemplo de operação e uso do CI 74LS190.

Universidade de Passo Fundo

Engenharia Elétrica

Lab. de Circuitos Digitais II

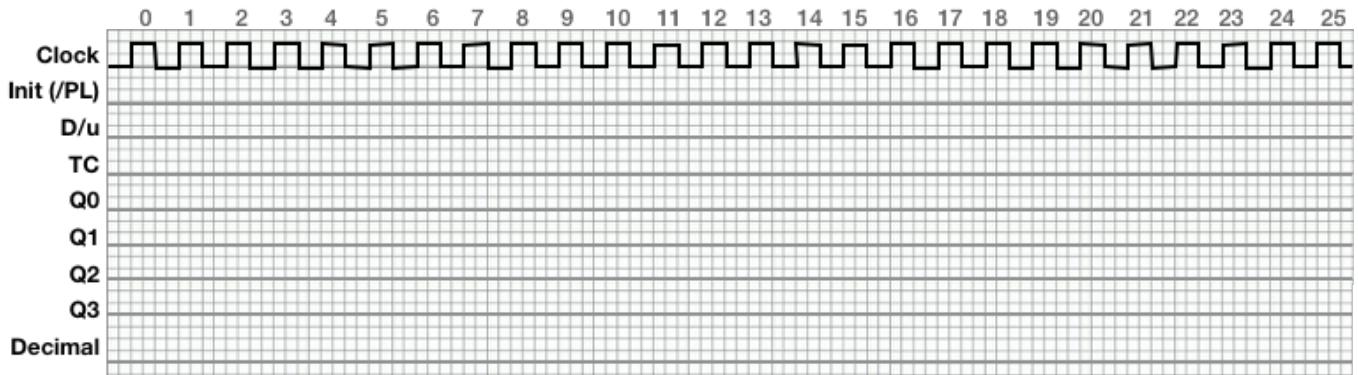
Prof. Fernando Passold

Aluno(a)₁: _____**Lab 9) Aplicação do CI 74190**Aluno(a)₂: _____ Data: ____ / ____ / ____

9.4 Perguntas

Obs.: Entregar até o final da aula:

- 1) Complete as formas de onda abaixo, mostrando todo comportamento cíclico deste circuito, desde o instante de tempo em que o mesmo é alimentado. Os seguintes sinais devem aparecer neste diagrama no tempo: *Clock*, \overline{PL} , Q_0 , Q_1 , Q_2 , Q_3 , *TC*, *D/u* e saídas Q_i do 74LS190.



Obs.: Suponha um pulso inicial de \overline{PL} (*Parallel Load*) antes dos primeiros 3 pulsos de clock.

- 2) Descreva em no máximo 3 linhas, usando suas próprias palavras a sequência de operação do circuito montado neste laboratório.

- 3) O que teria que ser adaptado no circuito montado neste laboratório para que a contagem iniciasse a partir do número 3? Mostre o diagrama elétrico desta versão.

10 | Máquina síncrona “Exótica”

Listo de Material

- U1, U2: 2× CI 74LS76 (Flip-Flop JK, borda de descida);
- U3: CI 74LS08 (4× portas AND(2));
- U4: CI 74LS32 (4× portas OR(2));
- U5: CI 74LS06 (6× portas NOT);
- D1: 1× Led comum;
- R1: 1× Resistor de 330Ω ;
- Módulo Display 7-Segmentos;
- Gerador de Sinais, saída TTL ($1 \sim 40$ Hz);
- Fonte de Alimentação TTL.

Opcionalmente, o aluno(a) pode usar ainda:

- 2× push-bottom NA;
- R2, R3: 2× Resistores de $10K\Omega$;
- R4 à R11: 8× Resistores de 330Ω ;
- D2 à D9: 8× Led's comuns;
- U6: CI 74LS138 (DEC 3/8).

10.1 Introdução

Este documento traz o projeto de uma máquina síncrona “diferente”. A figura 10.1 mostra o diagrama de estados correspondente à este circuito. Notar que existe uma entrada **Mode**, ou M , usada para definir a direção da sequência de funcionamento desta máquina.

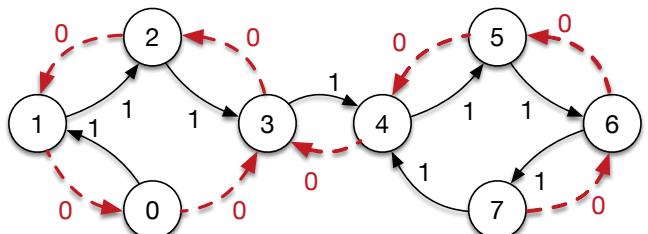


Figura 10.1: Diagrama de estados da máquina síncrona “diferente”.

10.2 Projeto

A sequência para projeto de um circuito sequencial síncrono exige que se defina o flip-flop que será adotado (ver tabela 10.1) e posteriormente, a tabela de transição do circuito completo deve ser definido com base no flip-flop sendo adotado, no caso, FFs-JK (ver tabela 10.2).

Estado Atual $Q(t)$	Próximo Estado $Q(t+1)$								
		FF-RS	FF-D	FF-JK	S	R	D	J	K
0	0	0	X	0	0	0	X		
0	1	1	X	1	1	1	X		
1	0	0	1	0	X	1	X	1	
1	1	X	0	1	X	0			

Tabela 10.1: Tabela de transição para alguns flip-flops.

Uma vez definida a tabela de transição do circuito completo (tabela 10.2), pode-se proceder ao projeto do circuito associado com cada uma das entradas dos FFs selecionados, no caso, FFs-JK. Seguem os respectivos Mapas de Karnaugh associadas com as entradas de controle de cada FF-JK e suas respectivas equações:

Ref	Estado atual				Próximo Estado			FF ₂	FF ₁	FF ₀			
	M	$Q_2(t)$	$Q_1(t)$	$Q_0(t)$	$Q_2(t+1)$	$Q_1(t+1)$	$Q_0(t+1)$	J_2	K_2	J_1	K_1	J_0	K_0
0	0	0	0	0	0	1	1	0	X	1	X	1	X
1	0	0	0	1	0	0	0	0	X	0	X	X	1
2	0	0	1	0	0	0	1	0	X	X	1	1	X
3	0	0	1	1	0	1	0	0	X	X	0	X	1
4	0	1	0	0	0	1	1	X	1	1	X	1	X
5	0	1	0	1	1	0	0	X	0	0	X	X	1
6	0	1	1	0	1	0	1	X	0	X	1	1	X
7	0	1	1	1	1	1	0	X	0	X	0	X	1
8	1	0	0	0	0	0	1	0	X	0	X	1	X
9	1	0	0	1	0	1	0	0	X	1	X	X	1
10	1	0	1	0	0	1	1	0	X	X	0	1	X
11	1	0	1	1	1	0	0	1	X	X	1	X	1
12	1	1	0	0	1	0	1	X	0	0	X	1	X
13	1	1	0	1	1	1	0	X	0	1	X	X	1
14	1	1	1	0	1	1	1	X	0	X	0	1	X
15	1	1	1	1	1	0	0	X	0	X	1	X	1

Tabela 10.2: Tabela de transição da máquina síncrona exótica.

J_2 :		$Q_1 \ Q_0$			
M	Q_2	00	01	11	10
	00	0	0	0	0
	01	X	X	X	X
	11	X	X	X	X
	10	0	0	1	0

$$J_2 = M \cdot Q_1 \cdot Q_0$$

 K_2 :

K_2 :		$Q_1 \ Q_0$			
M	Q_2	00	01	11	10
	00	X	X	X	X
	01	1	0	0	0
	11	0	0	0	0
	10	X	X	X	X

$$K_2 = \overline{M} \cdot \overline{Q_1} \cdot \overline{Q_0}$$

 J_1 :

J_1 :		$Q_1 \ Q_0$			
M	Q_2	00	01	11	10
	00	1	0	X	X
	01	1	0	X	X
	11	0	1	X	X
	10	0	1	X	X

$$J_1 = M \cdot Q_0 + \overline{M} \cdot \overline{Q_0}$$

$$J_1 = \overline{M} \oplus Q_0$$

$$J_1 = M \odot Q_0$$

 K_1 :

K_1 :		$Q_1 \ Q_0$			
M	Q_2	00	01	11	10
	00	X	X	0	1
	01	X	X	0	1
	11	X	X	1	0
	10	X	X	1	0

$$K_1 = M \cdot Q_0 + \overline{M} \cdot \overline{Q_0}$$

$$K_1 = \overline{M} \oplus Q_0$$

$$K_1 = M \odot Q_0$$

J_0 :		$Q_1 \ Q_0$			
M	Q_2	00	01	11	10
	00	1	X	X	1
	01	1	X	X	1
	11	1	X	X	1
	10	1	X	X	1

$$J_0 = 1$$

K_0 :		$Q_1 \ Q_0$			
M	Q_2	00	01	11	10
	00	X	1	1	X
	01	X	1	1	X
	11	X	1	1	X
	10	X	1	1	X

$$K_0 = 1$$

Repare que podemos reduzir as portas necessárias buscando termos em comum nas expressões anteriores:

$$J_2 = \underbrace{(M \cdot Q_0) \cdot Q_1}_{\text{t: AND(2):a}} \underbrace{\cdot \overline{\text{AND(2):c}}}_{\text{u: AND(2):b}}$$

$$K_2 = \underbrace{(\overline{M} \cdot \overline{Q_0}) \cdot \overline{Q_1}}_{\text{u: AND(2):b}} \underbrace{\cdot \overline{\text{AND(2):d}}}_{\text{t: AND(2):a}}$$

$$J_1 = \underbrace{M \cdot Q_0}_{\text{t}} + \underbrace{\overline{M} \cdot \overline{Q_0}}_{\text{u}}$$

$$K_1 = J_1$$

10.3 Circuito (Parte Prática)

A figura 10.2 apresenta o circuito capaz de reproduzir o diagrama de estados mostrado na figura 10.1. Na entrada CLOCK deve ser conectado um gerador de sinais TTL na faixa de 1 até 40 Hz.

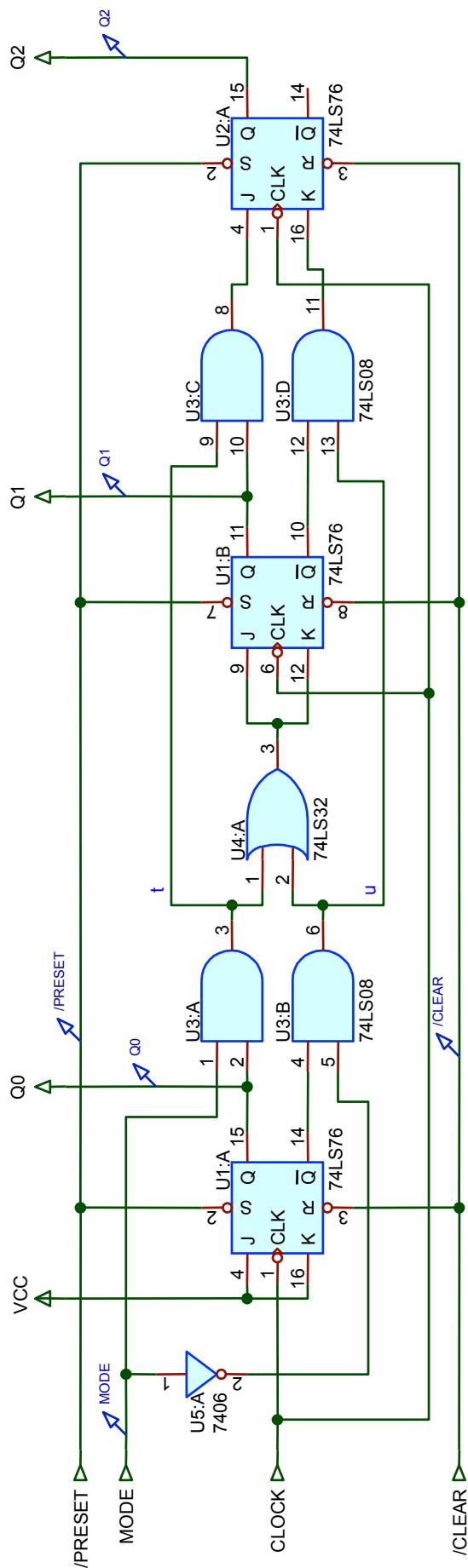


Figura 10.2: Circuito previsto para a máquina síncrona “diferente”.

10.3.1 Módulo de Chaves

Sugere-se o acréscimo do circuito da figura 10.3 conectado ao circuito da figura 10.2 para facilitar a interação com esta máquina síncrona. Desta forma, quando a chave SW1, associada com **MODE** for acionada (fechada), a saída **MODE** comuta para '0' (nível lógico baixo); e quando a chave SW1 estiver aberta: **MODE** = 1. E as chaves SW2 e SW3 permitem gerar mais facilmente na prática, pulsos de **CLEAR** e **RESET**, respectivamente.

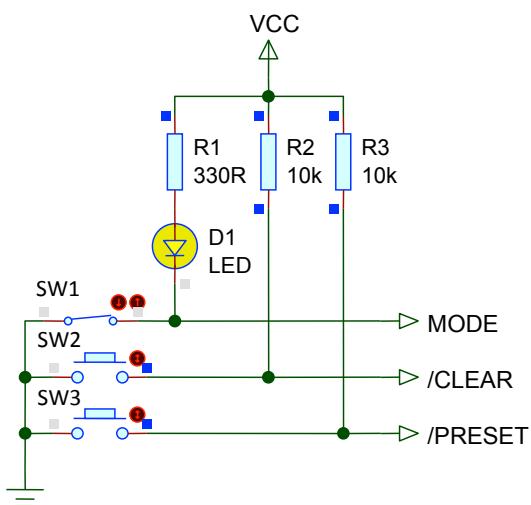


Figura 10.3: Sugestão deacomplamento de chaves à máquina síncrona “diferente”.

10.3.2 Interface de Saída

Sugere-se ainda uma interface de saída (fig. 10.4) constituída pela adição de um módulo para display de 7-segmentos conectado às saídas Q_2 (bit MSB) à Q_0 (bit LSB) do circuito da figura 10.2 incluindo leds conectados às saídas do DEC 3/8 (U6 da figura 10.4). As entradas do DEC (U6) estão conectadas às saídas Q_i dos FF's da figura 10.2. Dessa forma, o estudante pode acompanhar melhor o efeito estético e visual criado por este circuito e mesmo compreender mais claramente o funcionamento deste circuito.

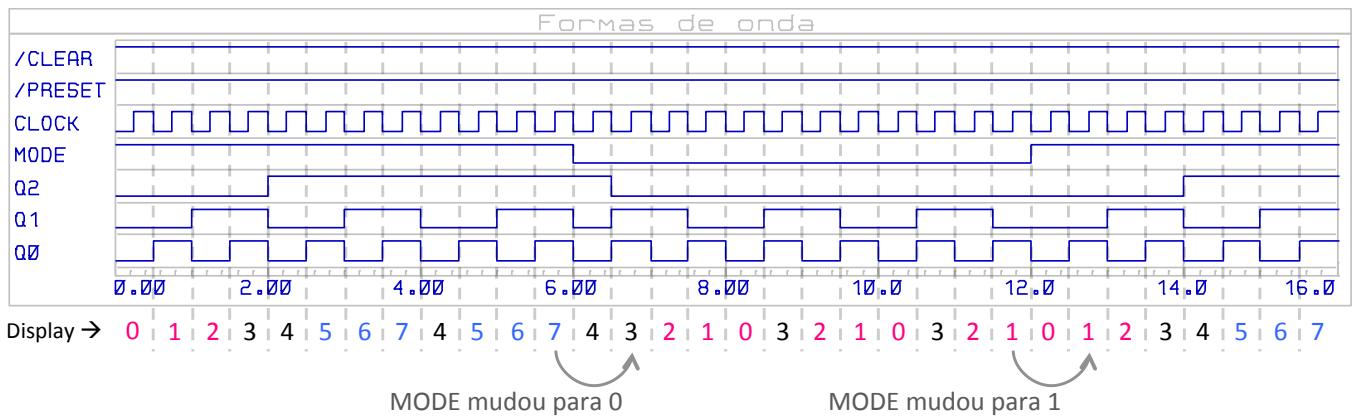


Figura 10.5: Formas de onda mostrando funcionamento do circuito.

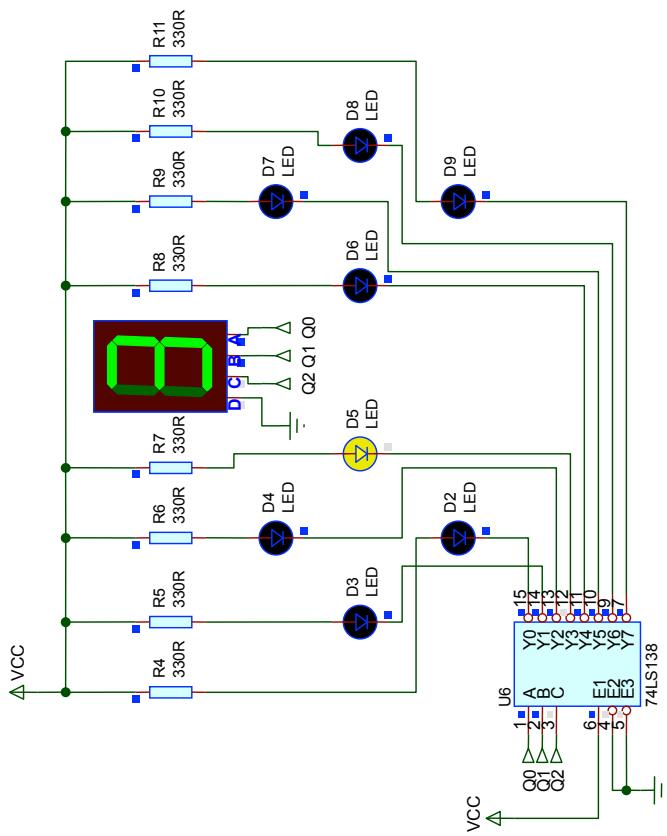


Figura 10.4: Interface de saída sugerida para o circuito da máquina síncrona "diferente".

10.4 Formas de onda

A figura 10.5 mostra formas de onda que podem ser esperadas durante o funcionamento deste circuito. Repare na

informação que apareceria no display de 7-segmentos, caso este estivesse conectado ao circuito.

10.5 Teste sugerido

O estudante é livre para testar todas as modalidades de funcionamento deste circuito, mas particularmente certa sequência está sendo sugerida. Neste caso, o estudante deve, sequencialmente, tentar realizar:

- Chavear o circuito para **MODE=1** (a respectiva chave **SW1** da figura 10.3 fica aberta e o Led D1 fica apagado);
- Pulsar a chave **SW2** para gerar um pulso de **CLEAR** no circuito, forçando-o a começar no estado '0'.
- Notar que o circuito inicia com display em '0' e assume a seguinte sequência de funcionamento: $0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7$ e então entra num "loop infinito": $\rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 4 \rightarrow 5 \rightarrow \dots$ enquanto a entrada **MODE** não mudar de estado.
- Sugere-se então que o aluno tente mudar a entrada **MODE** para '0' (implica fechar a chave **SW1** mostrada na figura 10.3 de forma a acender o Led D1), mas quando o circuito estiver no estado '6' ou '7'.
- Supondo que **MODE** foi para '0' quando o circuito estava no estado '7', a seguinte sequência deve passar a ser observada: $7 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0$ e se a entrada **Mode** continuar sendo mantida em '0' o circuito entra então num outro "looping infinito": $\rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow \dots$



11 | Projeto Contador Síncrono [2022/1]

11.1 Introdução

Segue uma rápida documentação à respeito de contador síncrono diferente criado no primeiro semestre de 2022. Este projeto foi iniciado na aula teórica de Circuitos Digitais II.

Obs.: Este documento não têm a pretensão de servir como um relatório e mais sim, como um "memorial descritivo" do projeto.

A figura 11.1 à seguir mostra o diagrama de estados especificado para este projeto.

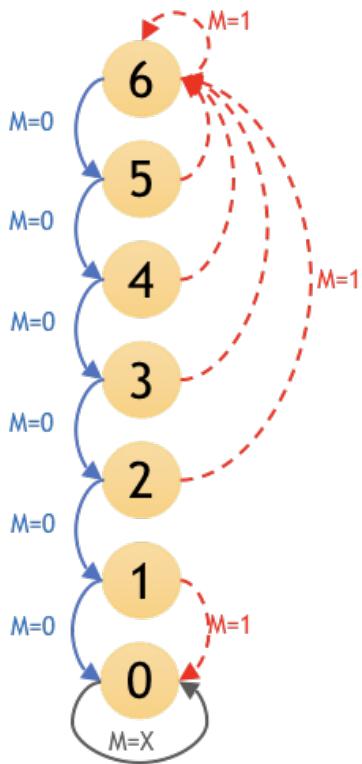


Figura 11.1: Diagrama de estados da máquina síncrona “diferente”.

Notar que esta sequencia de estados exige o uso de 3 FF's. E que uma combinação de estado não está prevista no diagrama anterior, o estado 7. O projetista deve ser capaz de especificar como o circuito evolui se os seus FF's forem configurados para este estado inicial – ver figura 11.2.

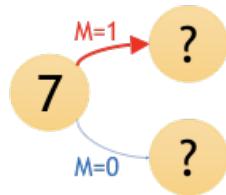


Figura 11.2: Estado não explorado – como circuito evolui?

11.2 Embasamento Teórico

O contador à ser projetado é um circuito genérico que obedece a generalização mostrada na figura 11.3.

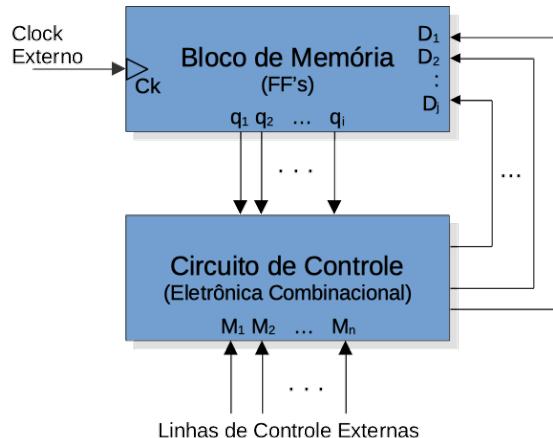


Figura 11.3: Diagrama em blocos de circuito sequencial genérico.

Na figura anterior, o Bloco de memória é o responsável por armazenar (registrar) o estado atual do circuito. É desta forma que um circuito sequencial “sabe” em que estado se encontra e qual é o próximo estado que deve assumir. Este circuito (sequencial), necessita de uma lógica de controle que leve em conta variáveis de entrada externas (no caso, a variável M), o estado atual do circuito e que então gere sinais que indiquem como o bloco de memória deve se comportar (ou “evoluir”). O sinal de clock externo permite modular (ditar) a velocidade na qual as transições de estado são executadas.

11.3 Projeto

O primeiro passo para realização deste projeto é levantar a tabela de transição do mesmo, ver tabela 11.1.

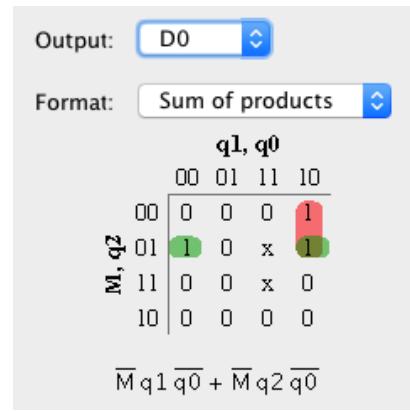
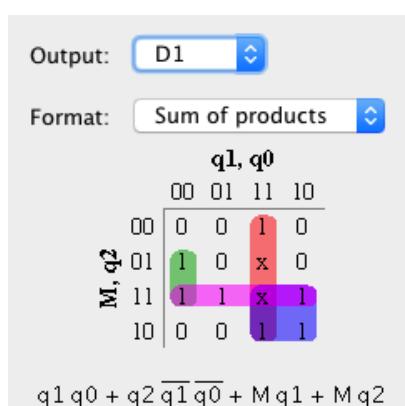
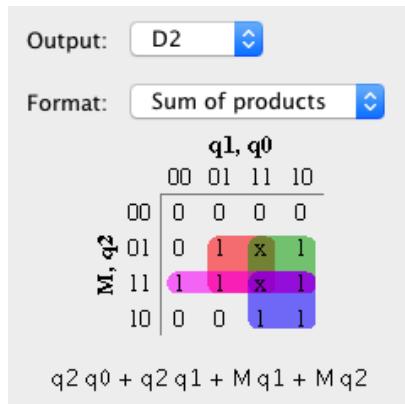
Ref	M	Estado Atual:			Próximo Estado:			Entradas Controle:		
		q2	q1	q0	Q2	Q1	Q0	D2	D1	D0
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	0	0	0
2	0	0	1	0	0	0	1	0	0	1
3	0	0	1	1	0	1	0	0	1	0
4	0	1	0	0	0	1	1	0	1	1
5	0	1	0	1	1	0	0	1	0	0
6	0	1	1	0	1	0	1	1	0	1
7	0	1	1	1	x	x	x	x	x	x
8	1	0	0	0	0	0	0	0	0	0
9	1	0	0	1	0	0	0	0	0	0
10	1	0	1	0	1	1	0	1	1	0
11	1	0	1	1	1	1	0	1	1	0
12	1	1	0	0	1	1	0	1	1	0
13	1	1	0	1	1	1	0	1	1	0
14	1	1	1	0	1	1	0	1	1	0
15	1	1	1	1	x	x	x	x	x	x

Tabela 11.1: Tabela de transição de estados.

Note que se optou por trabalhar com FF's do tipo D porque este FF apresenta apenas 1 entrada de controle (a entrada "D"). Assim, para este contador, se faz necessário especificar o circuito combinacional para as 3 entradas de controle dos 3 FF's tipo D. Se fosse optado pelo uso de FF's-JK, seriam necessárias $3 \times 2 = 6$ colunas para especificar 6 circuitos combinacionais de controle, já que este último FF apresenta 2 entradas de controle ("J" e "K").

A tabela 11.1 foi incorporada ao software LogiSim adotado como ferramenta para automatizar e facilitar o projeto do circuito combinacional necessário para as entradas D de controle dos 3 FF's.

Seguem os Mapas de Karnaugh e correspondentes equações resultantes para a entrada D de cada FF:



Usando o próprio LogiSim para projetar a eletrônica combinacional usando portas lógicas básicas, alcançamos a solução (para esta parte) mostrada na fig. 11.4.

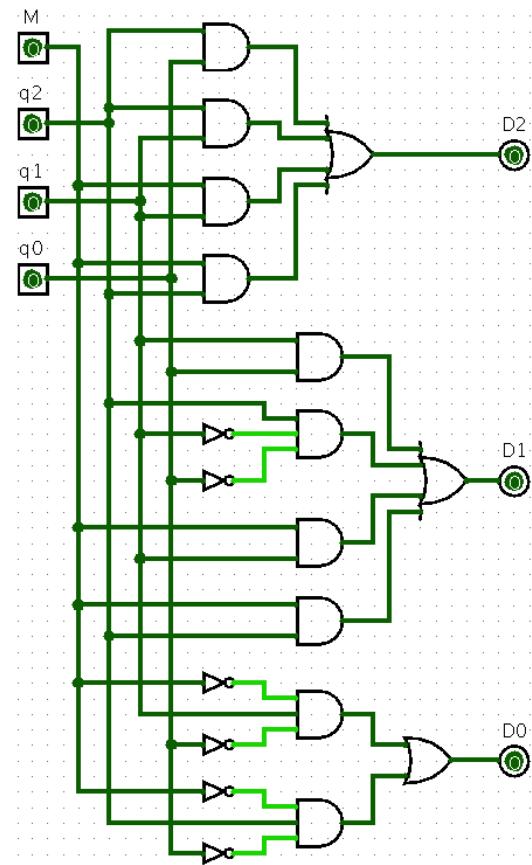


Figura 11.4: Circuito de controle dos FF usando portas lógicas básicas (síntese sugerida pelo soft LogiSim).

Uma rápida "contabilidade" sobre os CI's necessários para implementar o circuito de controle usando portas lógicas básicas revela a seguinte quantidade de CI's necessária:

- 7 x portas AND(2): 2 x CI's
 - 3 x portas AND(3): 1 x CI
 - 2 x portas OR(4): 1 x CI
 - 1 x porta OR(2): 1 x CI
 - 4 x portas NOT: 1 x CI
- total: 6 x CI's

Considerando que podemos usar o CI 74LS175 (traz dentro de si: 4 x FF's-D com Master Reset – ver fig. 11.5) para o bloco de

"memória", temos com resultado um circuito final que exigiria o uso de 7 x CIs.

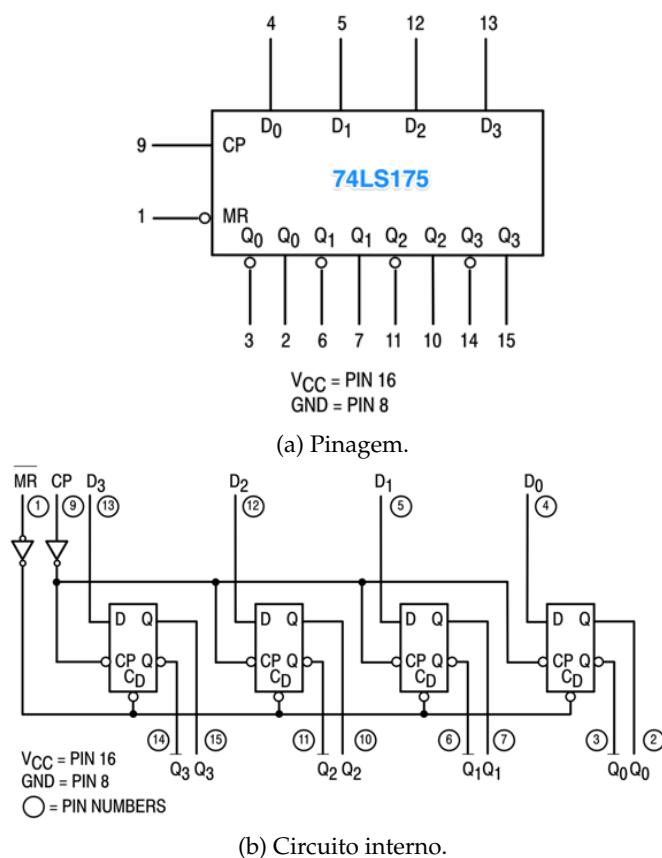


Figura 11.5: CI 74LS175: quádruplo FF tipo D com entrada de Master Reset.

A parte do circuito contendo FF's baseado no CI 74LS175 pode ser visto na fig. 11.6.

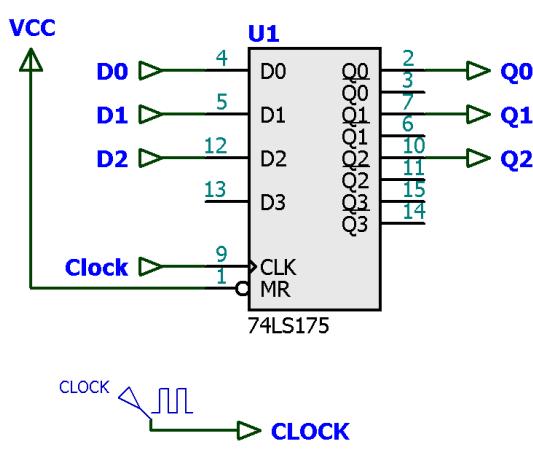


Figura 11.6: Parte do circuito contadora composta pelos FF's (CI 74LS175).

O circuito final seria montado com base nas figuras 11.4 e 11.6 e esta solução acabaria por exigir o uso de 7 pastilhas.

11.3.1 Análise do caso do estado "7"

Faltou analizar o que ocorre com o circuito caso seus FF's iniciem todos setados, ou seja, no estado "7", conforme solicitado na fig. 11.2.

Neste caso, temos que recuperar as equações booleanas encontradas para as entradas D_2 , D_1 e D_0 para poder determinar como o circuito avança caso inicie no estado "7". E para este caso, vamos considerar que o estado inicial dos FF's seja: $q_2(t) = 1$, $q_1(t) = 1$ e $q_0(t) = 1$, o que compõe o estado "7".

Iniciando pelo FF-2 temos:

$$D_2 = q_2 q_1 + q_2 q_1 + M q_1 + M q_2$$

Se $q_2(t) = 1$, $q_1(t) = 1$, $q_0(t) = 1$ e $M = 0$, então D_2 vai resultar:

$$D_2 = 1 \cdot 1 + 1 \cdot 1 + 0 \cdot 1 + 0 \cdot 1$$

$$D_2 = 1 + 1 + 0 + 0$$

$$D_2 = 1$$

Se $q_2(t) = 1$, $q_1(t) = 1$, $q_0(t) = 1$ e $M = 1$, então D_2 vai resultar:

$$D_2 = 1 \cdot 1 + 1 \cdot 1 + 1 \cdot 1 + 1 \cdot 1$$

$$D_2 = 1 + 1 + 1 + 1$$

$$D_2 = 1$$

O que significa que o próximo estado do FF-2 será: $Q_2(t+1) = 1$, independente do nível lógico da variável M .

Repetindo este raciocínio para os outros FF's obtemos:

$$D_1 = q_1 q_0 + q_2 \bar{q}_1 \bar{q}_0 + M q_1 + M q_2$$

Se $q_2(t) = 1$, $q_1(t) = 1$, $q_0(t) = 1$ e $M = 0$, então D_1 vai resultar:

$$D_1 = 1 \cdot 1 + 1 \cdot 0 \cdot 0 + 0 \cdot 1 + 0 \cdot 1$$

$$D_1 = 1 + 0 + 0 + 0$$

$$D_1 = 1$$

Se $q_2(t) = 1$, $q_1(t) = 1$, $q_0(t) = 1$ e $M = 1$, então D_1 vai resultar:

$$D_1 = 1 \cdot 1 + 1 \cdot 0 \cdot 0 + 1 \cdot 1 + 1 \cdot 1$$

$$D_1 = 1 + 0 + 1 + 1$$

$$D_1 = 1$$

Da mesma forma que ocorre para o FF-2, o próximo estado do FF-1 será: $Q_1(t+1) = 1$ independente do nível lógico da variável M .

Por fim:

$$D_0 = \bar{M} q_1 \bar{q}_0 + \bar{M} q_2 \bar{q}_0$$

Se $q_2(t) = 1$, $q_1(t) = 1$, $q_0(t) = 1$ e $M = 0$, então D_0 vai resultar:

$$D_0 = 1 \cdot 1 \cdot 0 + 1 \cdot 1 \cdot 0$$

$$D_1 = 0 + 0$$

$$D_1 = 0$$

e se $q_2(t) = 1$, $q_1(t) = 1$, $q_0(t) = 1$ e $M = 1$, então D_0 vai resultar:

$$D_0 = 0 \cdot 1 \cdot 0 + 0 \cdot 1 \cdot 0$$

$$D_1 = 0 + 0$$

$$D_1 = 0$$

Neste caso, o próximo estado de FF-0, independente do nível lógico da variável M será: $Q_0(t+1) = 0$.

Isto permite concluir que caso os FF's deste circuito contador iniciassem setados, ou seja, no estado "7", ele naturalmente iria evoluir para o estado "6", independente do nível lógico da variável "M", ou seja, algo como o mostrado na fig. 11.7.



Figura 11.7: Evolução deste contador se iniciado no estado "7".

Ref	M	q ₂ q ₁ q ₀	D ₂
0	0	0 0 0	0
1	0	0 0 1	0
2	0	0 1 0	0
3	0	0 1 1	0
4	0	1 0 0	0
5	0	1 0 1	1
6	0	1 1 0	1
7	0	1 1 1	X 1
8	1	0 0 0	0
9	1	0 0 1	0
10	1	0 1 0	1
11	1	0 1 1	1
12	1	1 0 0	1
B	1	1 0 1	0
M	1	1 1 0	1
15	1	1 1 1	X 1

11.4 Explorando Outras Soluções

Existem outras formas de implementar circuitos combinacionais, à saber:

- **Uso de MUX:** neste caso, implicaria "sacrificar" um MUX para cada entrada de controle necessária, ou seja, seriam necessários 3 x MUXes, ou um circuito final totalizando 5 x CIs (se for considerado o uso do CI 74LS175 + 1 x CI de portas inversoras que muito provavelmente será necessário).

- **Uso de DEC:** neste caso, apenas 1 x DEC resolve o problema (um de 4 entradas para 16 saídas) e seriam necessárias mais CIs de portas NAND para "coletar" os minitermos necessários para síntese de cada entrada D. Analisando a tabela de transição, se percebe:

$D_2 = \sum_m \{5, 6, 10, 11, 12, 13, 14\}$ – ou seja, implica o uso de uma porta NAND de 7 entradas (comercialmente existe de 8);

$D_1 = \sum_m \{3, 4, 10, 11, 12, 13, 14\}$ – também outra porta NAND de 7 entradas;

$D_0 = \sum_m \{2, 4, 6\}$ – ou seja, implica o uso de uma porta NAND de 3 entradas;

o que significa que além do CI 74LS175 e do DEC, seriam necessários + 4 CIs, também totalizando um circuito com 5 CIs.

Notar que a opção pelo uso do DEC implica menos "jumpers" na montagem física deste circuito, já que a solução com MUX (supondo que vamos trabalhar com MUX de 8 canais de entrada) exige que se complete conexões à todas suas 8 entradas. Como serão necessários 3 MUXes, implica: $3 \times 8 = 24$ jumpers, enquanto a solução usando DEC, implicaria, em $8 + 8 + 3 = 19$ jumpers (uma pequena diferença).

Ref	M	q ₂ q ₁ q ₀	D ₂
0	0	0 0 0	0
1	0	0 0 1	0
2	0	0 1 0	0
3	0	0 1 1	1
4	0	1 0 0	1
5	0	1 0 1	0
6	0	1 1 0	0
7	0	1 1 1	X 1
8	1	0 0 0	0
9	1	0 0 1	0
10	1	0 1 0	1
11	1	0 1 1	1
12	1	1 0 0	1
B	1	1 0 1	0
M	1	1 1 0	1
15	1	1 1 1	X 1

Ref	M	q ₂ q ₁ q ₀	D ₀
0	0	0 0 0	0
1	0	0 0 1	0
2	0	0 1 0	1
3	0	0 1 1	0
4	0	1 0 0	1
5	0	1 0 1	0
6	0	1 1 0	1
7	0	1 1 1	X 0
8	1	0 0 0	0
9	1	0 0 1	0
10	1	0 1 0	0
11	1	0 1 1	0
12	1	1 0 0	0
B	1	1 0 1	0
M	1	1 1 0	1
15	1	1 1 1	X 1

E assim finalmente podemos determinar o circuito necessário para realizar o contador especificado no diagrama de estados da fig. 11.1.

A fig. 11.6 mostra o bloco do circuito de "memória" contendo os FF's-D usando a pastilha 74LS175 tal qual já usada na solução usando portas lógicas básicas (mesmo circuito, esta parte).

A próxima figura 11.9 mostra o bloco de controle dos FF's composta por 3 MUX'es de 8 canais de entrada.

Notar que as entradas I_7 (ou X7 como usa o Proteus) dos MUXes, à princípio não deveriam possuir ligações, já que estão relacionadas com o estado atual "7", para o qual, não está previsto nenhum próximo estado (note que inicialmente estas células estão assinaladas com "X"). Mas como estamos usando MUXes, podemos nos dar "ao luxo" de programar o próximo

11.5 Solução usando MUXes

Optando-se pela solução composta por MUXes e adotando MUXes de 8 canais de entrada teremos que avaliar suas "programações":

estado do circuito, caso, o mesmo por acaso, inicie no estado "7". Neste caso, optou-se por avançar os estados da forma mostrada na figura 11.8.

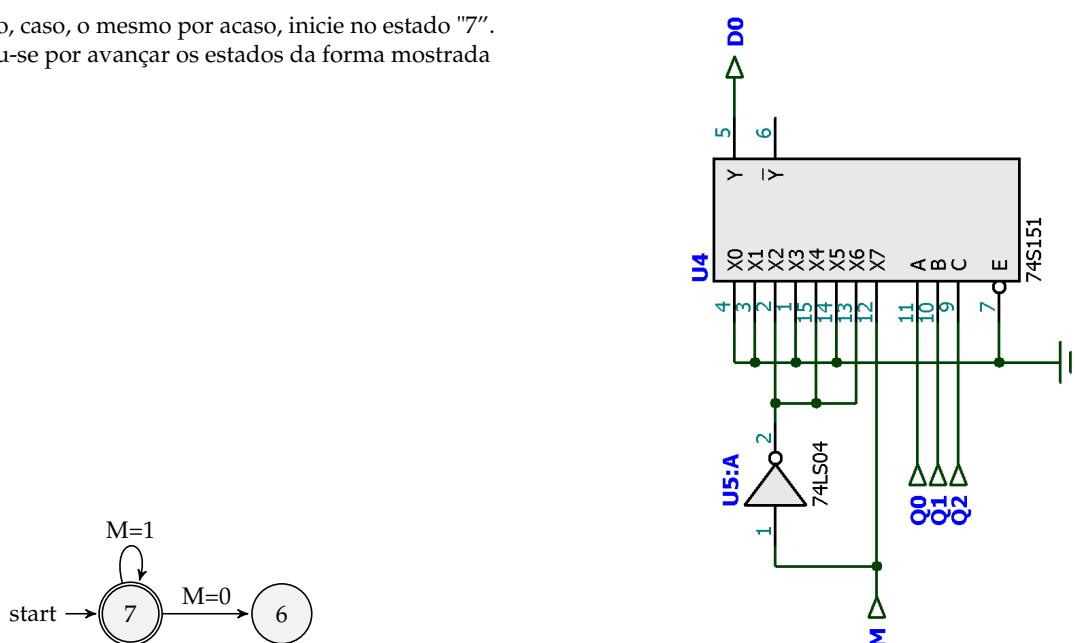
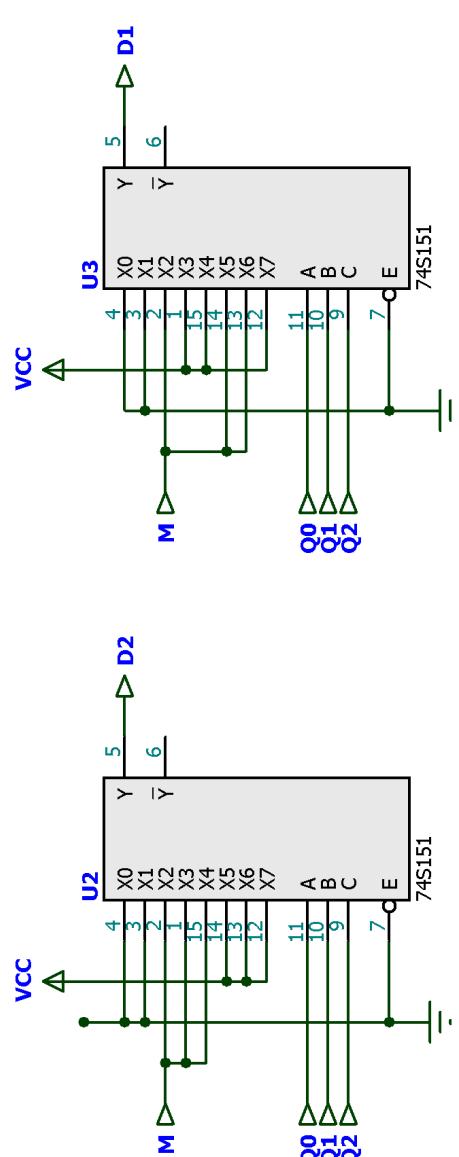


Figura 11.8: Evolução deste contador se iniciado no estado "7" para os caso da solução com MUXes.

Esta figura mostra que com os FF's no estado "7", caso a variável M se encontre no nível lógico BAIXO, estas combinações de condições implique na linha de referência 7, usada nas tabelas de "programação" mostradas anteriormente. E caso a variável M se encontre em nível lógico ALTO, isto implique na linha de referência 15. Ao final, as linhas de referência 7 e 15 das tabelas de "programação" dos MUXes, ficam como mostra a tabela 11.2 à seguir.



Ref	M	$q_2\ q_1\ q_0$	$Q_2\ Q_1\ Q_0$
7	0	1 1 1	1 1 0
15	1	1 1 1	1 1 1

Tabela 11.2: Tabela de transição de estados prevendo condição inicial "7".

Figura 11.9: Bloco de controle dos FF's, constituído por MUX'es.

E por fim, pensando numa "interface de saída" para facilitar a visualização do avanço de estados deste contador, temos o circuito mostrado na fig. 11.10.

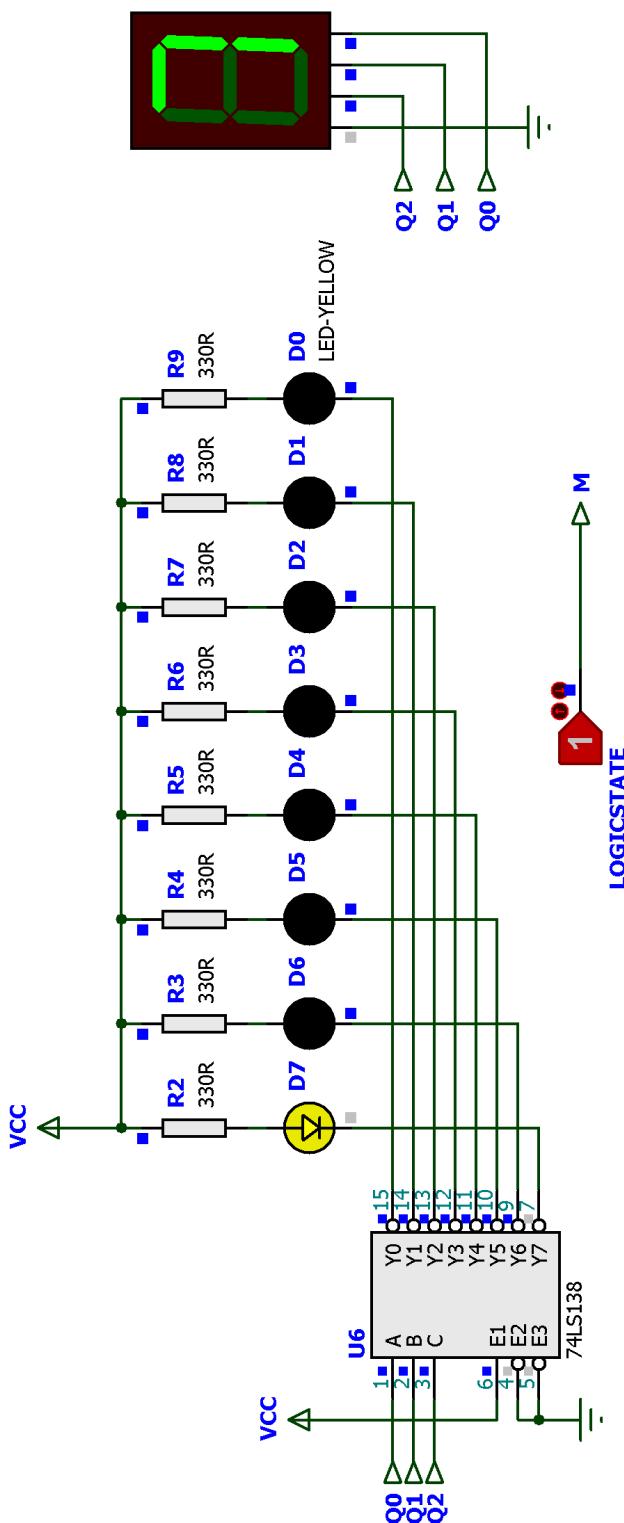


Figura 11.10: Interface de saída para este contador.

11.6 Problema: estado inicial dos FF's

Entretanto... este circuito apresenta considerações quanto ao nível lógico inicial assumido por seus FF's.

Problema:

Note que temos um detalhe associado com o bloco dos FF's. **Os FF's não podem iniciar resetados neste projeto.** Por isto, a entrada "Master Reset" não é utilizada. Se estes FF's iniciarem resetados, não importa o nível lógico da variável "M", o circuito fica paralizado no estado "0" e não será possível comprovar seu funcionamento na prática.

11.6.1 Solução à nível de Simulação

Para fins de simulação no Proteus, pode-se especificar o estado inicial de FF's. Neste caso, selecionar o CI 74LS175, clicar com o botão direito do mouse sobre o mesmo e alterar suas Propriedades da forma mostrada na fig. 11.11.

Infelizmente na prática, não será possível configurar o estado inicial deste CI desta forma.

11.7 Possível solução para inicialização dos FF's

Os FF's deste circuito não podem iniciar resetados. Como o CI 74LS175 só possui entrada de "Master Reset" mas nenhuma entrada de "Parallel Load" que permita iniciar em determinado estado, na prática, teremos que abandonar o uso desta pastilha e construir um circuito semelhante ao CI 74LS175, mas incorporando a necessária entrada de "Parallel Load". Tal circuito é mostrado na fig. 11.12.

Note que a entrada de "Parallel Load" deste circuito é assíncrona e sob lógica ATIVO ALTO. Perceba que esta entrada permite repassar de forma conveniente o nível lógico apresentado nas entadas P_i para as entradas assíncronas de Set e Reset dos FF's adotados neste caso, usando a lógica:

$$Set_i = PL \cdot P_i$$

$$\overline{Set}_i = \overline{PL} \cdot \overline{P}_i$$

$$\overline{Set}_i = NAND(PL, P_i)$$

e:

$$Reset_i = PL \cdot \overline{P}_i$$

$$\overline{Reset}_i = \overline{PL} \cdot \overline{\overline{P}_i}$$

$$\overline{Reset}_i = NAND(PL, \overline{P}_i)$$

11.8 Circuito final com "Parallel Load"

A fig. 11.13 mostra o circuito completo previsto para este contador, considerando o uso de pastilhas 74LS74 (duplo FF-D, com entradas assíncronas de Set e Reset) e circuito "extra" para o "Parallel Load".

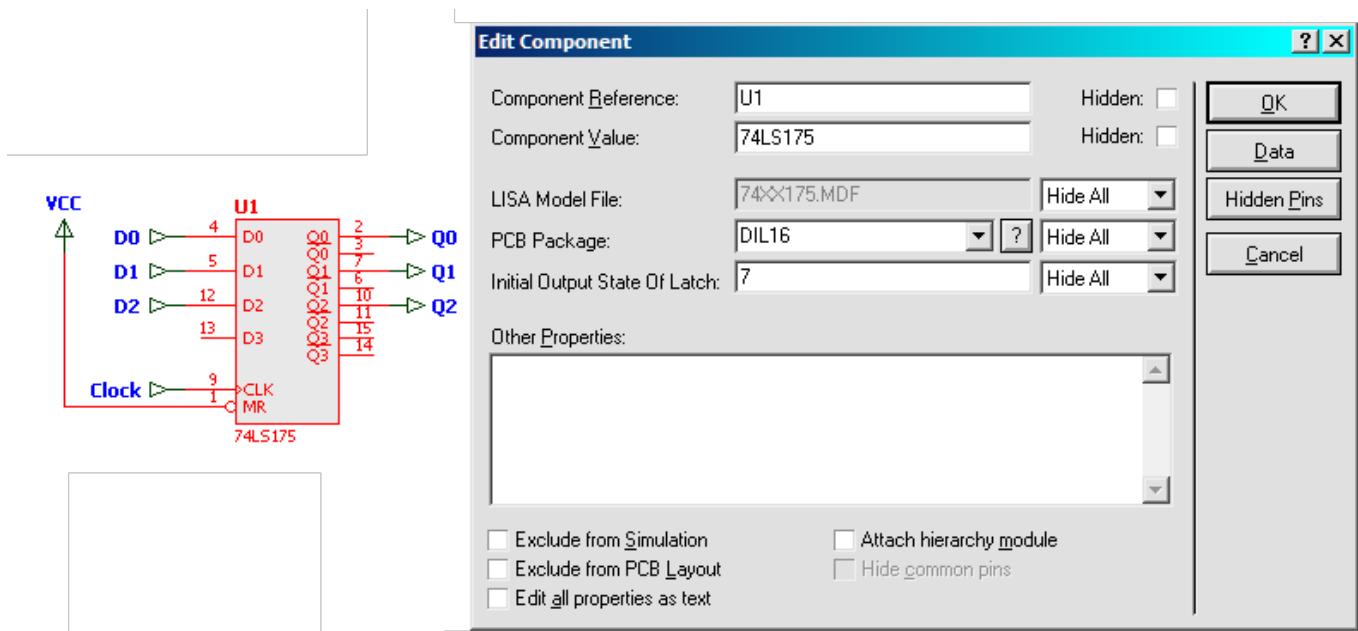


Figura 11.11: Ajuste do estado inicial dos FF's do CI 74LS175.

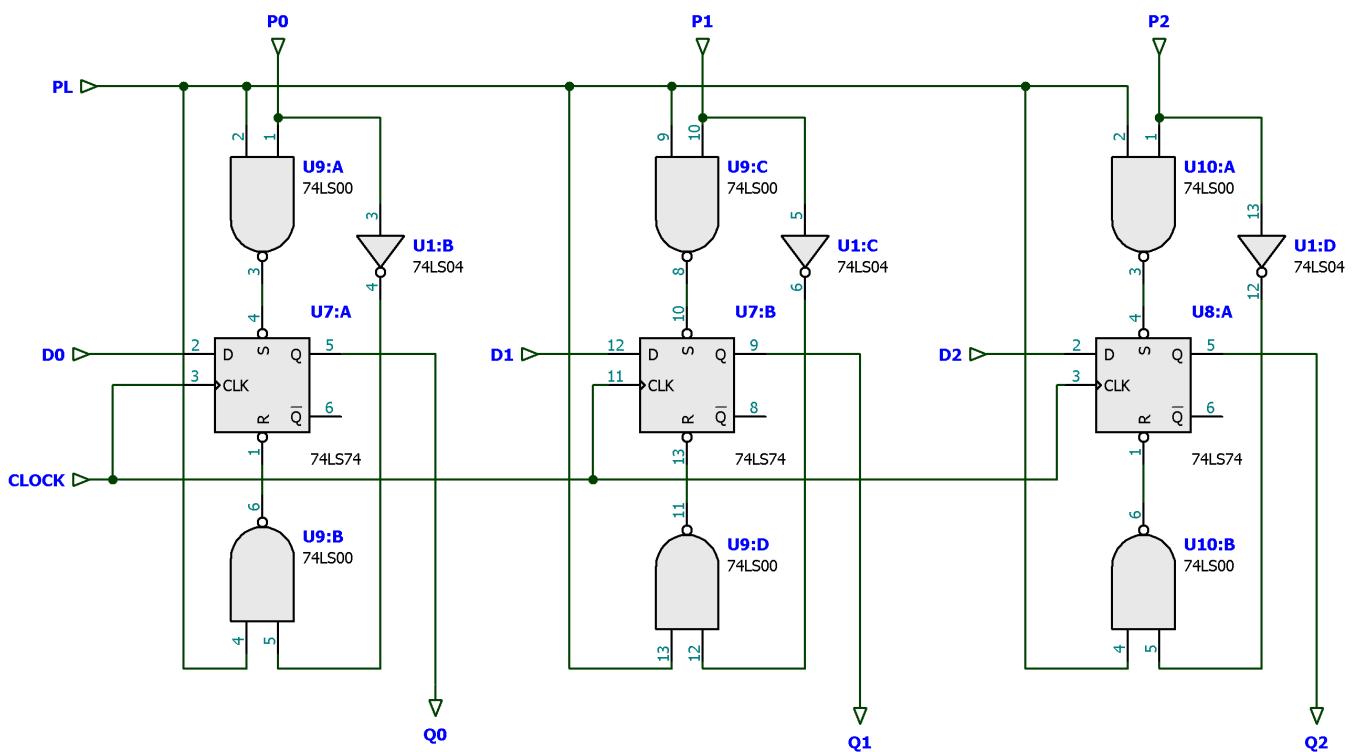


Figura 11.12: FF's-D com circuito "extra" de "Parallel Load".

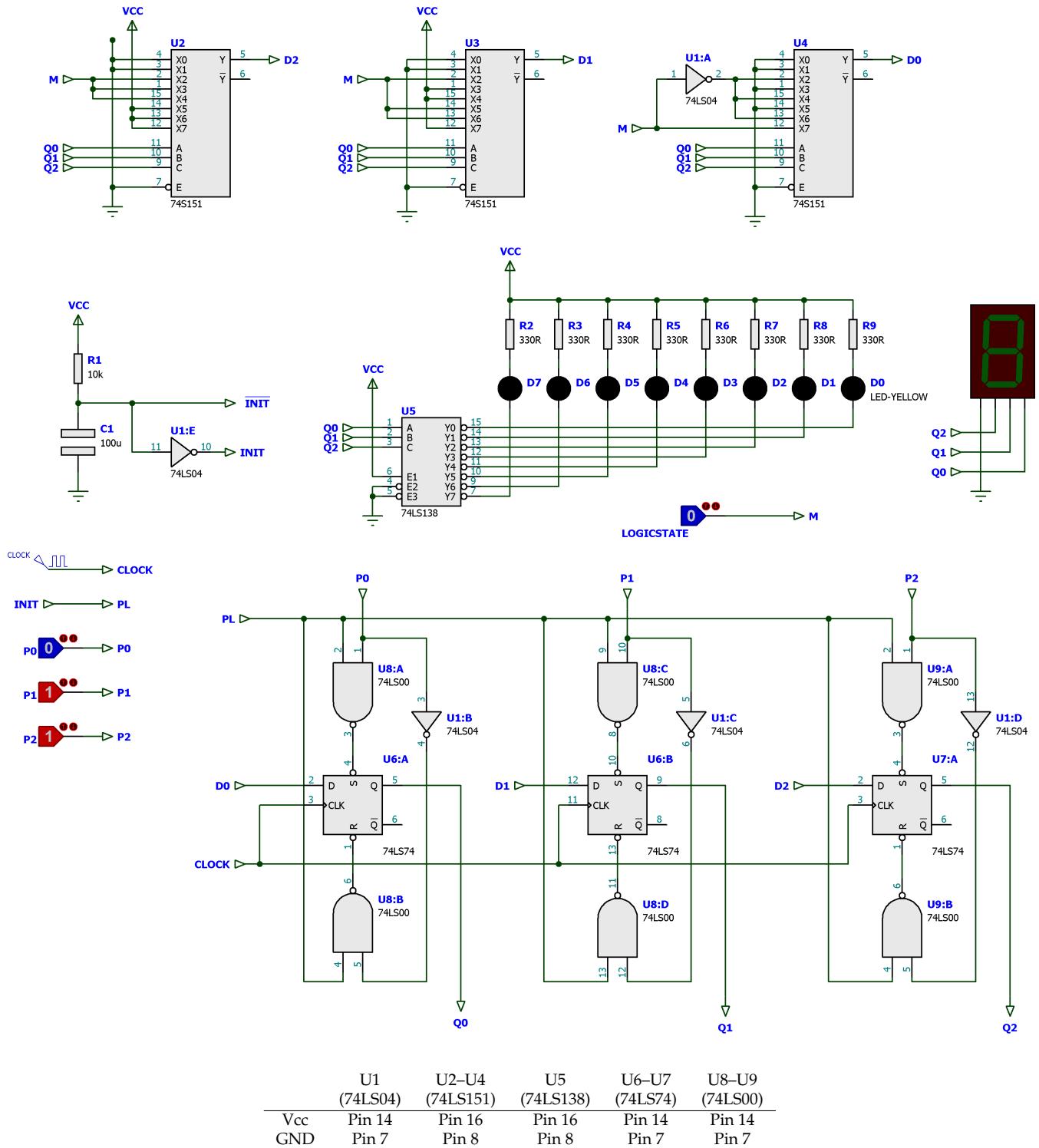


Figura 11.13: Circuito completo do contador incluindo parte "extra" de "Parallel Load". Neste caso, o circuito inicia no estado "6" com $M = 0$.

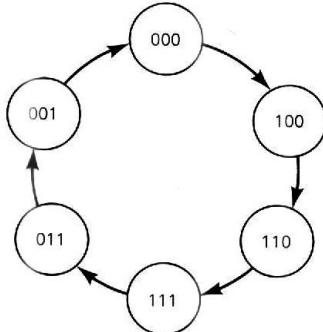
Lista de Material

1 R1 Resistor de 10 KΩ;
8 R2 – R9 Resistores de 330 Ω;
1 C1 Capacitor de 100 μ F × 10 Volts;
1 U1 CI 74LS04 (portas inversoras);
3 U2 – U4 CI 74LS151 (MUX 8 canais);
1 U5 CI 74LS138 (DEC 3/8);
2 U6 – U7 CI 74LS74 (FF-D);
2 U8 – U9 CI 74LS00 (portas NAND(2));
8 D0 – D7 Leds mini;
1 Gerador de Sinais ;
1 Fonte de Alimentação 5V, TTL .

12 | Contador Johnson usando Reg. de Deslocamento 74LS164

Listas de Material

- 1 × CI 74LS164A;
- 2 × CIs 74LS14 (6 × NOTs Schmidt-trigger);
- 2 × CIs 74LS00 (portas NAND(2));
- 9 × Leds vermelhos mini;
- 9 × resistores de $330\ \Omega \times 1/8W$;
- Gerador de funções (onda quadrada $< 17\ Hz$).



12.1 Fundamentos Teóricos

Um contador Johnson de 3 bits (6 estágios) é mostrado na figura 12.1.

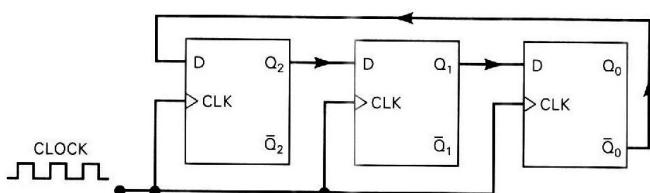


Figura 12.1: Estrutura básica de contador Johnson de módulo 6.

Note que a saída \bar{Q}_0 é conectada de volta à entrada D de Q_2 . Isto implica que o inverso do nível lógico armazenado em Q_0 será transferido para Q_2 no próximo pulso de clock – conforme pode ser percebido pela figura 12.2. Supondo que todos os FFs estejam inicialmente em 0, suas formas de onda e de estados podem ser vistas nas figuras 12.2 e 12.3 respectivamente.

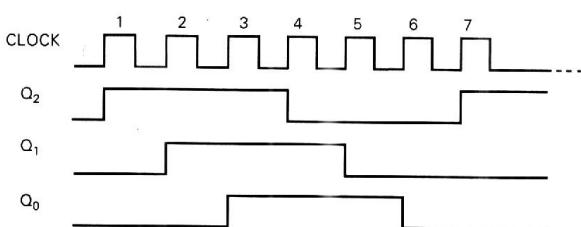


Figura 12.2: Formas de onda do contador Johnson MOD-6.

Figura 12.3: Diagrama de estados do contador Johnson MOD-6.

Já o CI 74F164A é um registrador de deslocamento de 8-bits, de entrada serial e saída paralela (*serial-in, parallel-out shift register*) de alta velocidade (90 MHz) e de 8 bits. Possui uma entrada assíncrona de reset geral (*master reset*) – pino 9, conforme demonstra a pinagem da figura 12.4. O diagrama lógico interno no CI 74LS164 é mostrado na próxima página.

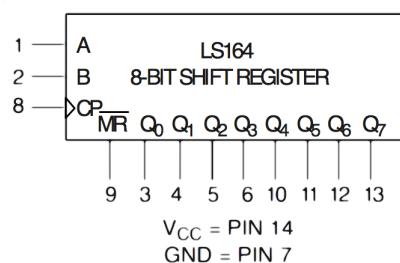


Figura 12.4: Pinagem do CI 74F164A.

Os dados seriais entram através de uma porta (interna) AND de 2 entradas: A e B , de maneira síncrona, na transição do nível lógico BAIXO-para-ALTO do sinal colocado na sua entrada de clock (CP). A tabela 12.1 mostra como opera este CI.

Modo de Operação	Entradas		Saídas		
	MR	A	B	Q_0	$Q_1 - Q_7$
Reset (Clear)	L	X	X	L	L-L
Shift	H	1	1	L	$q_0 - q_6$
	H	1	h	L	$q_0 - q_6$
	H	h	1	L	$q_0 - q_6$
	H	h	h	H	$q_0 - q_6$

Tabela 12.1: Tabela de Seleção de Modo do CI 74164.

12.2 Parte Prática

- 1) Com base no item anterior, deduza o circuito de um **contador Johnson MOD-6 (6 estágios)** usando o CI 74(LS/F)164. Você deve completar o diagrama elétrico iniciado na fig. 12.7.

Atenção: Como este CI não é capaz de drenar muita corrente em nível lógico ALTO ($I_{OH}/I_{OL} = -1\text{mA}/20\text{mA}$), se faz necessário usar portas NOT para inverter os níveis lógicos nas saída Q_i do contador e ligá-las à Leds no modo ATIVO BAIXO.

- 2) Projete e monte a lógica necessária para **decodificar as saídas do contador Johnson** montado no item anterior. Estas saídas devem ser **ATIVO BAIXO** para que drenem leds nesta configuração. Note que deverão estar presentes 6 saídas: $\bar{Y}_0 - \bar{Y}_5$. Porém, só será permitido o uso de portas NAND de 2 entradas (CI 74LS00).

Dica: Repare que é possível usar apenas 2 entradas de um porta AND ou NAND para decodificar as saídas de um contador Johnson qualquer porque durante sua transição de estados ocorrem combinações lógicas entre (saídas de) 2 FFs que não se repetem. Por exemplo, a combinação $Q_2 = Q_0 = 0$ só ocorre uma vez durante toda sua sequência de funcionamento. Esta mesma característica é compartilhada por todos os outros estados da sequência, como você poderá verificar. De fato, qualquer que seja o tamanho do contador Johnson, as portas decodificadoras conterão apenas ANDs de 2 entradas.

As figuras 12.5 e 12.6 podem auxiliar para o desenvolvimento da parte prática.

12.3 Questões

- 1) Apresentar o diagrama elétrico completo referente ao item 1 (completar a figura 12.7);
- 2) Fazer acompanhar o circuito elétrico do item 2(a);
- 3) Mostrar o diagrama elétrico para o item 2(b), acompanhado das equações booleanas para cada uma das 6 saídas decodificadas.
- 4) Mostrar um diagrama de formas de onda referente ao item 2(b).

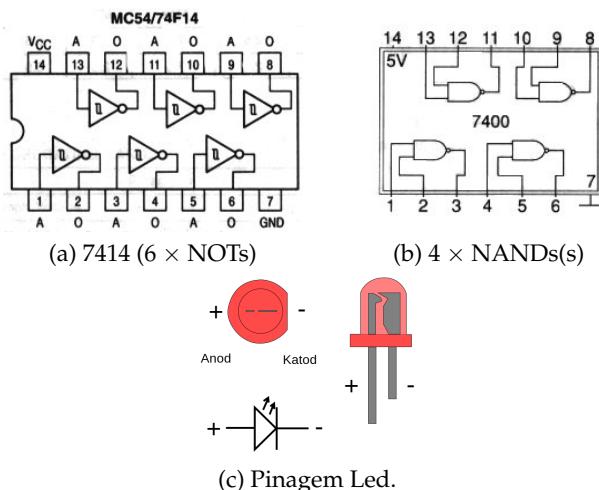


Figura 12.5: Pinagens de alguns componentes.

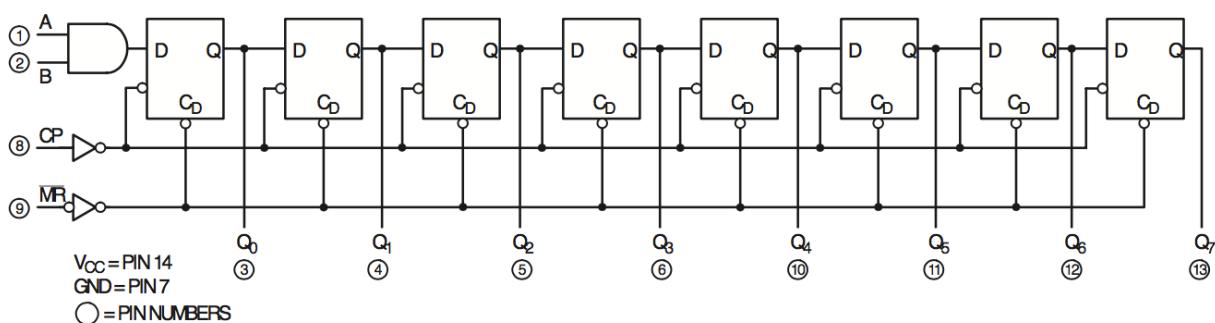
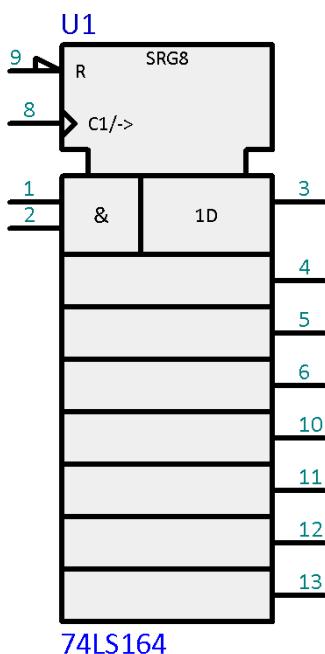


Figura 12.6: Diagrama elétrico lógico interno do CI 74LS164.

O espaço abaixo é para completar o primeiro circuito previsto para este laboratório...

1) Circuito do Contador Johnson de 6 estágios:



(Obs.: Aparece apenas o CI 74LS164 usando “padrão IEEE” do Proteus)

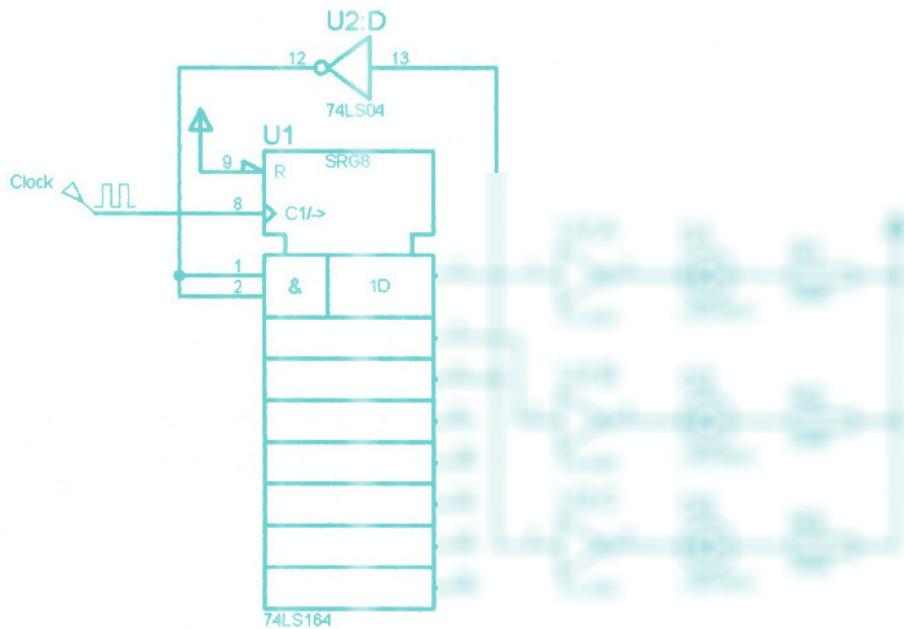
Figura 12.7: Diagrama elétrico à ser completado.

Algumas Respostas: _____

1) Circuito do contador Johnson de módulo 6:

2. Parte Prática

- 1) Montar o circuito de um contador Johnson de módulo 6 usando o CI 74LS164.



Círculo elétrico do contador Johnson.

- 2) Projete e monte a lógica necessária para decodificar as saídas do contador Johnson montado no item anterior. Note que serão geradas 6 saídas: $Y_0 - Y_5$. Porém, só será permitido o uso de portas AND de 2 entradas.

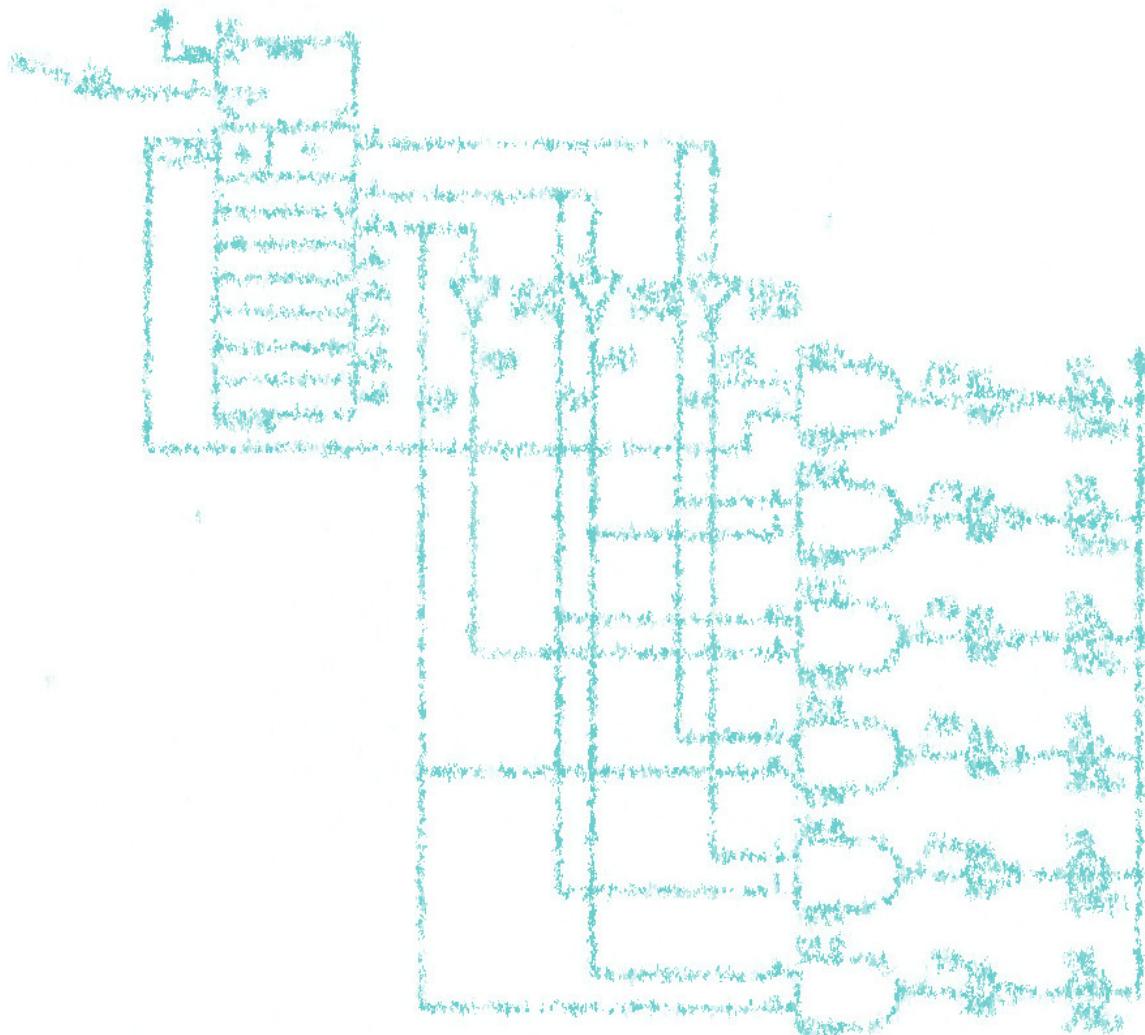
As saídas $Y_0 - Y_5$ serão barradas, pois serão utilizadas portas NAND ao invés de AND.

Tabela das Equações Booleanas				
Saída	Q_2	Q_1	Q_0	Equação Booleana
\bar{Y}_0	0	0	0	$\bar{Y}_0 = \overline{Q_0} \cdot \overline{Q_2}$
\bar{Y}_1	1	0	0	$\bar{Y}_1 = \overline{Q_0} \cdot \overline{Q_1}$
\bar{Y}_2	1	1	0	$\bar{Y}_2 = \overline{Q_1} \cdot \overline{Q_2}$
\bar{Y}_3	1	1	1	$\bar{Y}_3 = \overline{Q_0} \cdot \overline{Q_1}$
\bar{Y}_4	0	1	1	$\bar{Y}_4 = \overline{Q_0} \cdot \overline{Q_1}$
\bar{Y}_5	0	0	1	$\bar{Y}_5 = \overline{Q_1} \cdot \overline{Q_2}$

Figura 12.8: Circuito do contador Johnson de módulo 6 usando CU 74LS164.

2) Circuito do contador Johnson de módulo 6 com saídas decodificadas:

Diagrama elétrico do circuito montado:



3) Mostrar um diagrama no tempo do circuito anterior.

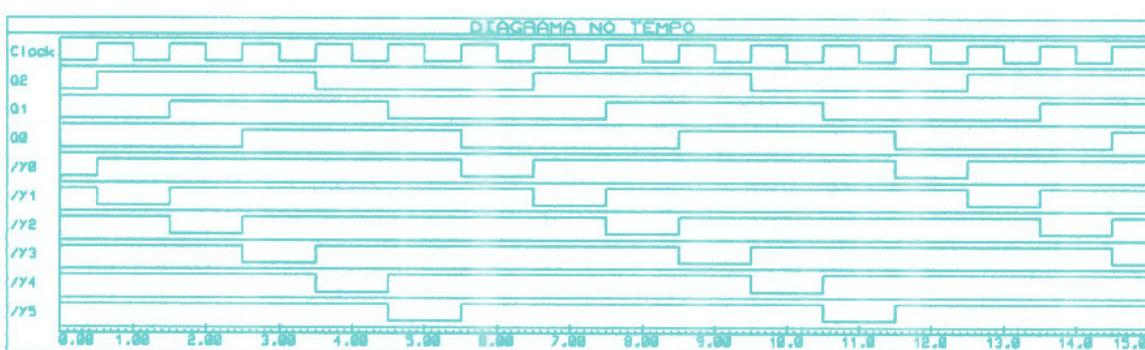
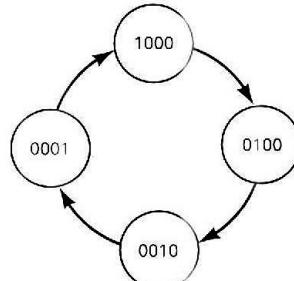
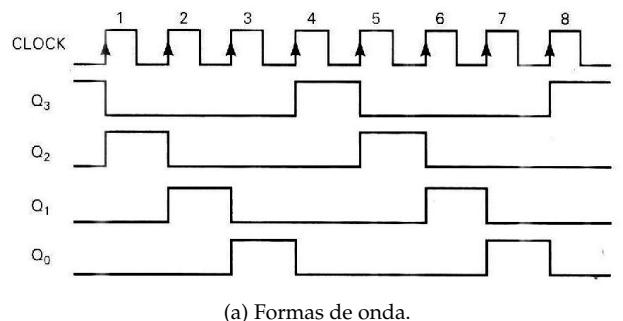


Figura 12.9: Contador Johnson de módulo 6 com saídas decodificadas.

13 | Contador em Anel usando Registrador de Deslocamento 74194

Listas de Material

- 1 CI 74LS194 (Reg. Descl. Bid.);
- 1 CI 74LS14 (Schmidt-trigger);
- 1 CI 74LS32 (portas OR);
- 1 resistor de $1\text{K}\Omega$;
- 1 capacitor de $1000\mu\text{F}$ @ 16V ;
- 1 módulo Display.



13.1 Fundamentos Teóricos

Um contador em Anel de 4 bits (4 estágios) é mostrado na fig. 13.1.

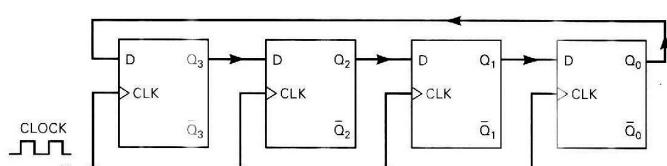


Figura 13.1: Contador em Anel de 4 estágios.

Note que a saída Q_0 é conectada de à entrada D do primeiro flip-flop. Isto implica numa recirculação de bits depois do 4º pulso de clock – conforme mostram as figs 13.2(a) e 13.2(b).

Entretanto, para gerar a sequência de estados mostrados nas figs. 13.2(a) e 13.2(b) é imprescindível que este circuito seja inicializado de uma maneira particular, conforme mostra a fig. 13.3. Assim que o circuito é ligado, um pulso de inicialização (*Power-Up Preset*) é gerado em função da constante de tempo (de carregamento) do capacitor. Este pulso de inicialização deve fazer com que o primeiro flip-flop inicie com sua saída em nível lógico ALTO e os restantes em nível lógico BAIXO.

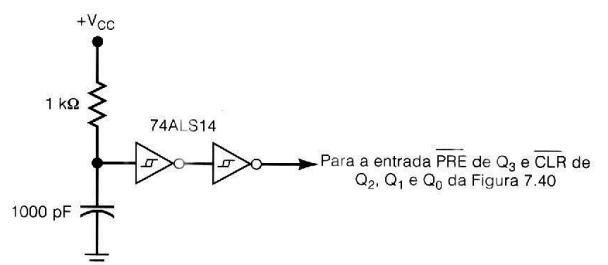
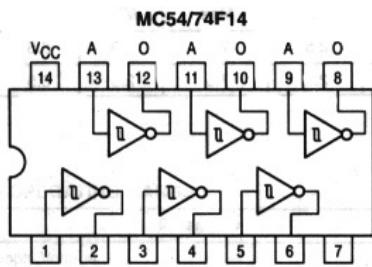


Figura 13.3: Inicialização necessária para o contador em anel.

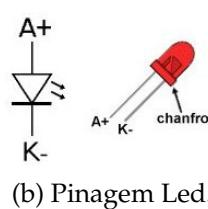
13.2 Parte Prática

Para realização deste contador neste laboratório está previsto o uso de um registrador de deslocamento universal bidirecional: o CI 74LS194. Este CI se comporta como mostrado na tabela 13.1 (próxima página). O diagrama elétrico do circuito à ser montado neste laboratório aparece na figura 13.5.

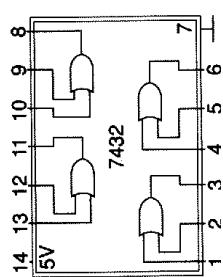
A fig. 13.4 mostra pinagem de alguns componentes.



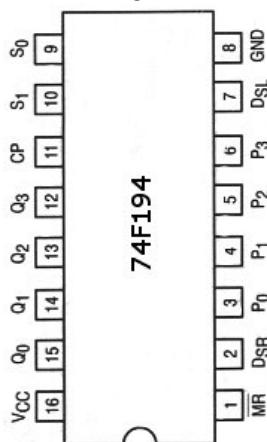
(a) 7414 (Schmidt-trigger)



(b) Pinagem Led.



(c) Pinagem do 74LS32.



(d) Pinagem do 74LS194.

Figura 13.4: Pinagens de alguns componentes.

- 1) Para que serve o circuito RC no circuito deste laboratório?
- 2) Este circuito explora 3 modos de operação do CI 74LS194. Você saberia indicar quais são os modos utilizados e quando eles são utilizados nesta montagem no laboratório?
- 3) Monte um diagrama de formas de onda para os primeiros 6 ciclos de clock considerando a entrada MODE=1 (deslocamento para direita). Inclua no gráfico, as formas de onda para as variáveis: Clock, PWR_UP, S₁, S₀, Q₃, Q₂, Q₁ e Q₀.
- 4) Idem ao item anterior, mas agora MODE=0 (deslocamento para esquerda).
- 5) Você saberia explicar porque o valor do capacitor na rede RC é elevado (1.000uF)?

Dica: repare que o “Parallel Load” (PL) do CI 74LS194 é síncrono, isto significa que o PL só ocorre na presença de uma borda ativa do sinal de clock (no caso, de subida). Se a frequência de clock à ser usada neste circuito for muito baixa, o PL pode não ser concretizado!

Obs.: Note que existem espaços para serem preenchidos nas próximas páginas em função das questões anteriores.

CP	\overline{MR}	S1	S0	D_{SR}	D_{SL}	Dn	Q0	Q1	Q2	Q3	Operação
X	0	X	X	X	X	X	0	0	0	0	Master Reset
↑	0	0	0	X	X	X	q0	q1	q2	q3	Hold
↑	1	1	0	X	0	X	q1	q2	q3	0	Shift Left (\leftarrow)
↑	1	1	0	X	1	X	q1	q2	q3	1	Shift Left (\leftarrow)
↑	1	0	1	0	X	X	0	q1	q2	q3	Shift Right (\rightarrow)
↑	1	0	1	1	X	X	1	q1	q2	q3	Shift Right (\rightarrow)
↑	1	1	1	X	X	dn	d0	d1	d2	d3	Parallel Load

Onde:
 CP = Entrada de Clock, borda de subida.
 S1, S0 = Controle do modo de operação.
 D3 – D0 = Entrada paralela de dados.
 D_{SR} = Entrada serial de dados (deslocamento para direita – Shift Right).
 D_{SL} = Entrada serial de dados (deslocamento para esquerda – Shift Left).
 \overline{MR} = Reset geral (Master Reset) assíncrono.
 Q3 – Q0 = Saídas paralelas.

Tabela 13.1: Tabela funcional do CI 74LS194.

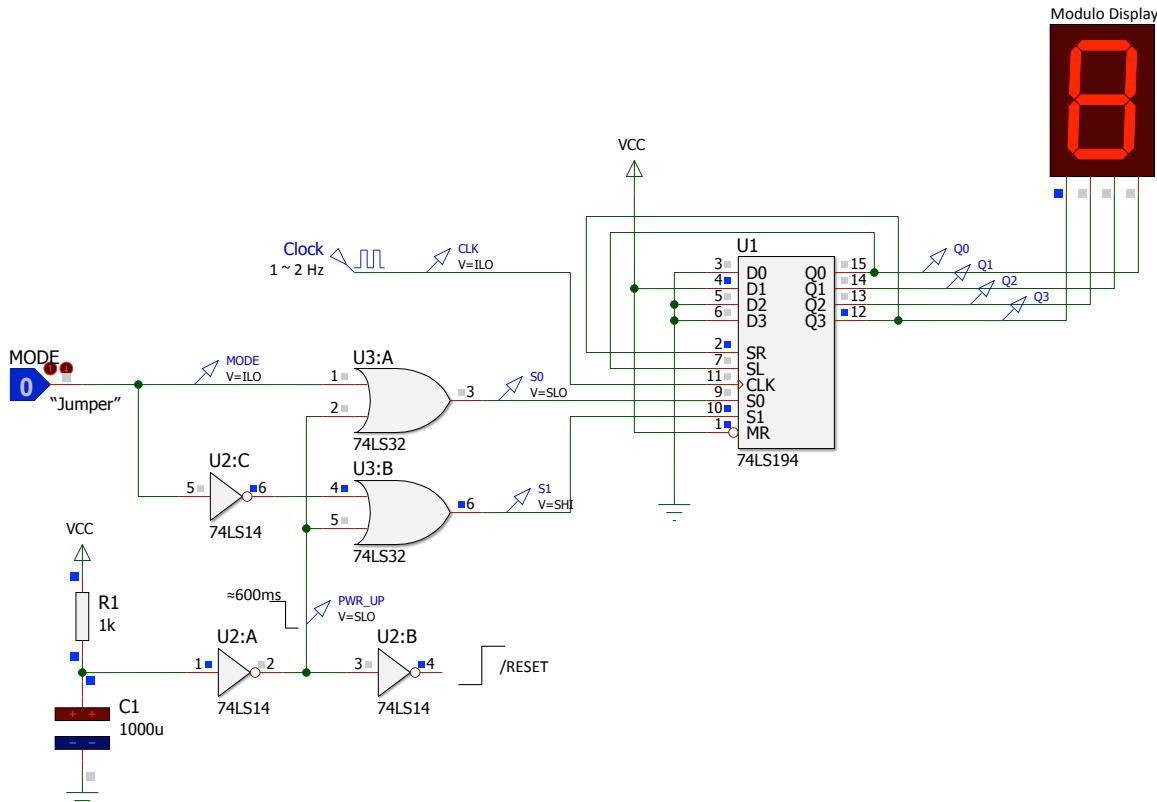


Figura 13.5: Circuito do contador em anel bidirecional à ser montado neste laboratório.

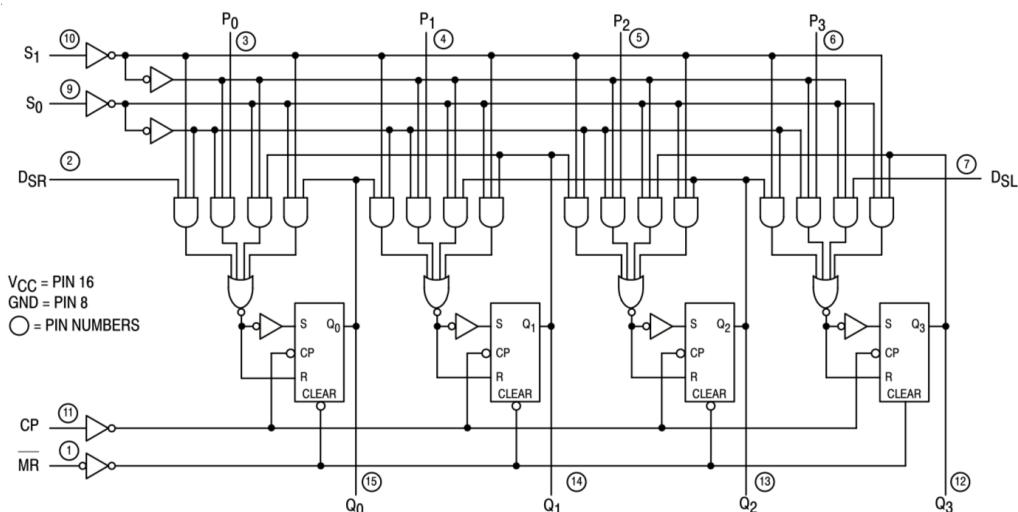
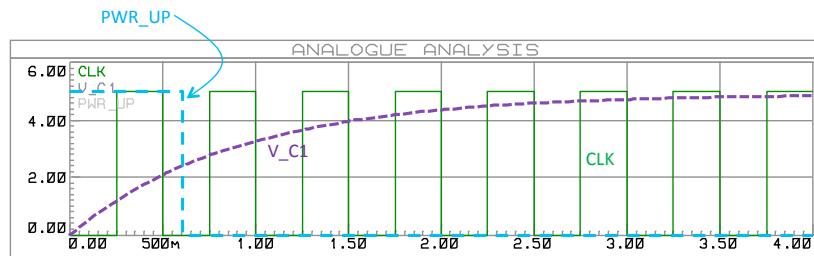


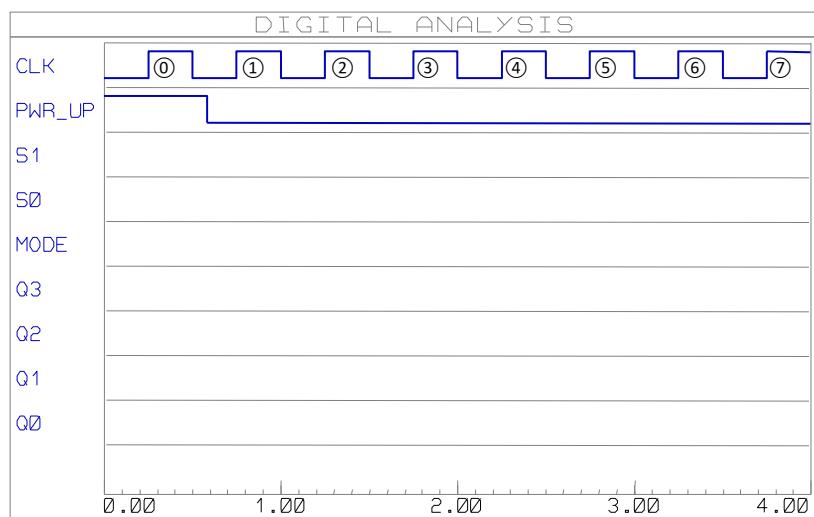
Figura 13.6: Diagrama lógico interno do 74LS194A (Motorola).

Espaço para Respostas

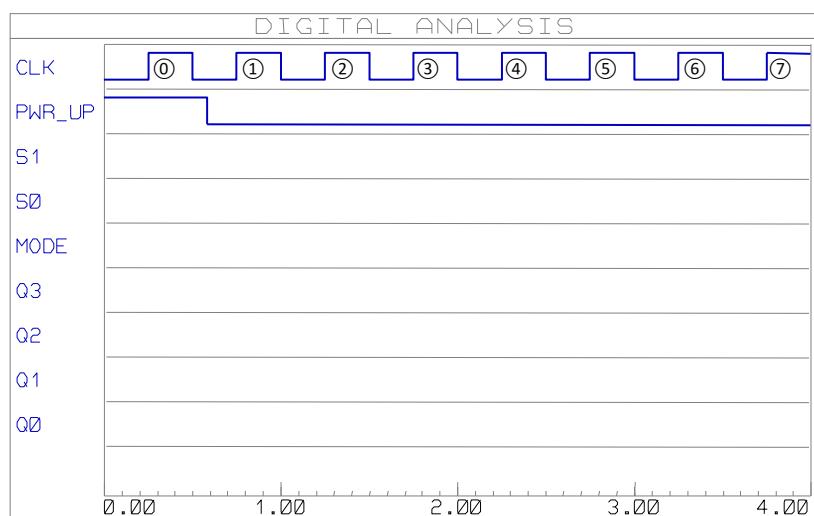
A figura à seguir mostra a forma de onda do circuito de “power-up” reset:



Questão 3) Complete as formas de onda esperadas para a entrada MODE=1:

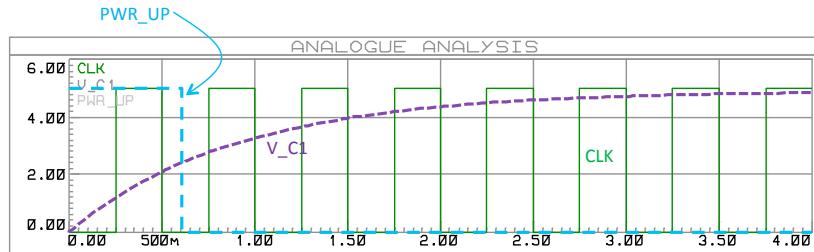


Questão 4) Complete as formas de onda esperadas para a entrada MODE=0:



Algumas Respostas: _____

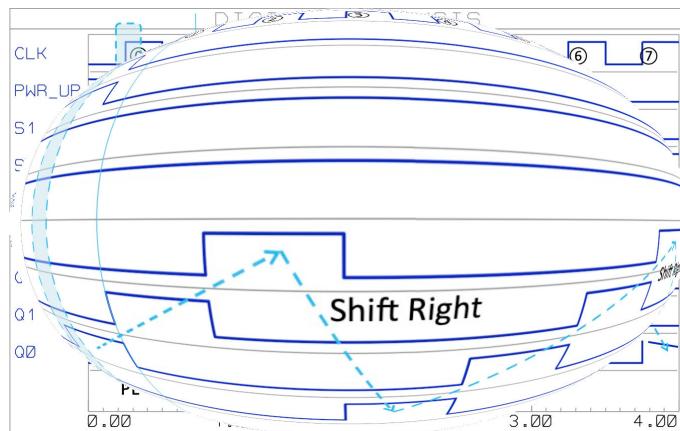
1. A rede RC é usada neste circuito para durante os primeiros 600 ms forçar a condição $S1, S0 = 11_{(2)}$ nas entradas do CI 74LS194, o que força um Parallel Load, neste caso, com o valor: $D3 - D0 = 0010_{(2)} = 2_{(10)}$. Se não for forçada uma condição inicial do circuito diferente de: $Q3 - Q0=0$, não existiria bit em nível lógico ALTO para ser circulado internamente pelo registrador de deslocamento (mesmo na presença do sinal de clock o circuito apresentaria fixamente uma saída como: $Q3-Q0=0$). Ver figura à seguir:



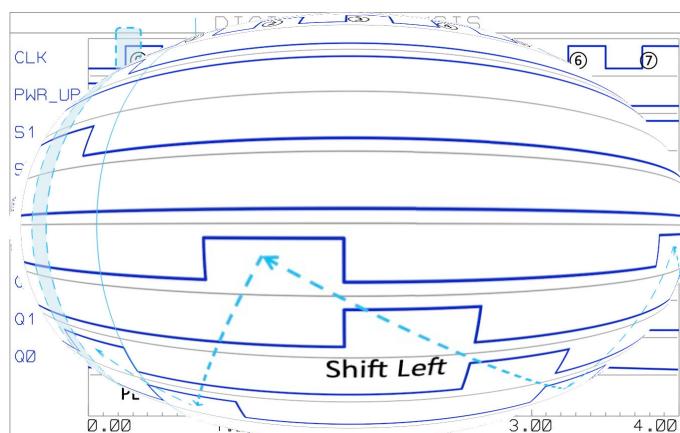
2. São explorados 3 modos de operação do CI 74LS194 conforme indica a tabela à seguir:

$S1, S0$	Modo de Operação	Observação
1 1	Parallel Load	Durante "Power-Up" (≈ 600 ms)
0 1	Shift Left	MODE=0
1 0	Shift Right	MODE=1

3. Formas de onda quando MODE=1:



4. Formas de onda quando MODE=0:



5. A rede RC é constituída por um capacitor de valor elevado, para que o tempo que o capacitor leve para se carregar seja elevado (em torno de 600ms), o que garante que o CI 74LS194 ainda vai "enxergar" (realizar) um Parallel Load mesmo no caso de frequências abaixo de $f = 1/(2 \times 600) = 0,83$ (Hz); a combinação: $S1, S0 = 11_{(2)}$ ainda persiste antes da borda de subida do sinal de clock (lembrando que o PL do 74LS194 é síncrono!).

14 | Contador Johnson Bidirecional usando Registrador Bidirecional

Listas de Material

- 1 CI 74F74 (FF-D ↑);
- 1 CI 74F14 (Schmidt-trigger);
- 1 CI 74F194 (4-Bit Bidirectional Universal Shift Register);
- 1 CI 74F08 (4 × AND(2));
- 4 Leds mini;
- 4 resistores de $330\ \Omega$;
- 2 push-buttons NA mini (padrão DPI);
- 1 resistor de $1K\Omega$;
- 2 resistores de $10\ K\Omega$;
- 1 capacitor de $1,0\ \mu F@16V$ (ou até $0,01\mu F@16V$).

Obs.: Uma barra de led's + uma rede resistiva de 330Ω até 560Ω pode ser usada no lugar os 4 leds + 4 resistores de 330Ω .

14.1 Fundamentos Teóricos

Um contador Johnson de 3 bits (6 estágios) é mostrado na fig. 14.1.

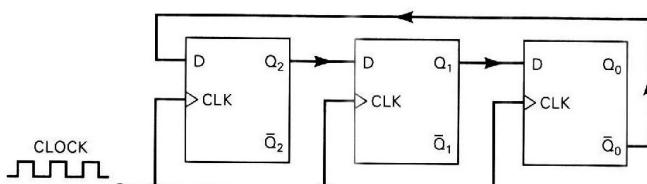
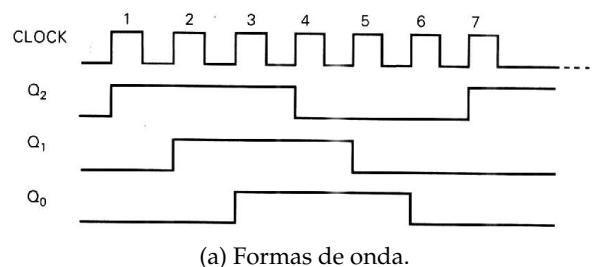
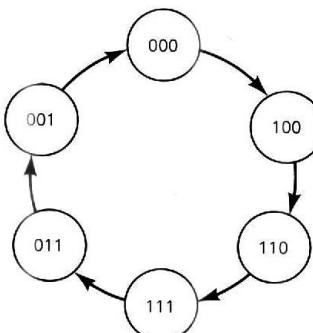


Figura 14.1: Contador Johnson de módulo 6.

Note que a saída \bar{Q}_0 é conectada de volta à entrada D de Q_2 . Isto implica que o inverso do nível lógico armazenado em Q_0 será transferido para Q_2 no próximo pulso de clock – conforme pode ser percebido pela fig. 14.1(a). Supondo que todos os FFs estejam inicialmente em 0, seu diagrama de formas de onda e de estados podem ser vistas nas figs. 14.1(a) e 14.1(b) respectivamente.



(a) Formas de onda.



(b) Diagrama de estados.

Figura 14.2: Detalhes contador Johnson MOD-6.

Uso do CI 74194

O CI 74LS194 é um registrador de deslocamento bidirecional universal de 4 bits capaz de trabalhar como registrador de entrada paralela, saída paralela, entrada serial para deslocamento para direita (*shift right*) e entrada serial para deslocamento para esquerda (*shift left*). A pinagem deste CI aparece na fig. 14.3. Seu modo de funcionamento depende da forma como estão setadas suas entradas S1 e S0 – conforme mostra a tabela 14.4 à seguir.

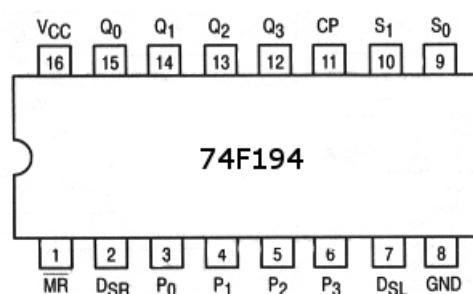


Figura 14.3: Pinagem do CI 74194.

Operating Mode	Inputs				Outputs					
	MR	S ₁	S ₀	D _{SR}	D _{SL}	P _n	Q ₀	Q ₁	Q ₂	Q ₃
Reset	L	X	X	X	X	X	L	L	L	L
Hold	H	I	I	X	X	X	q ₀	q ₁	q ₂	q ₃
Shift Left	H	h	I	X	I	X	q ₁	q ₂	q ₃	L
	H	h	I	X	h	X	q ₁	q ₂	q ₃	H
Shift Right	H	I	h	I	X	X	L	q ₀	q ₁	q ₂
	H	I	h	h	X	X	H	q ₀	q ₁	q ₂
Parallel Load	H	h	h	X	X	p _n	p ₀	p ₁	p ₂	p ₃

H (h) = HIGH Voltage Level

L (l) = LOW Voltage Level

 p_n (q_n) = Lower case letters indicate the state of the referenced input (or output) one setup time prior to the LOW-to-HIGH clock transition.

X = Immaterial

Figura 14.4: Tabela funcional do CI 74194.

Note que para fazer o CI 74194 funcionar como um contador Johnson bidirecional de 4 bits, basta enviar a saída invertida de Q_3 à entrada de deslocamento para direita (D_{SR}) para o caso de deslocamento dos bits ativos da esquerda para a direita (*shift right*, " \rightarrow "), ao mesmo tempo em que as entradas $S_1S_0 = 01$ (deslocamento para direita).

Para o caso de deslocamento dos bits ativos da direita para a esquerda (*shift left*, " \leftarrow ") é necessário enviar a saída invertida de Q_0 para a entrada de deslocamento para esquerda (D_{SL}) ao mesmo tempo em que se ajusta as entradas $S_1S_0 = 10$ (deslocamento para esquerda).

Para liberar (de forma garantida) os bits para deslocamento à esquerda ou a direita foi associada uma porta AND à cada entrada serial do 74194. Estas portas AND são comandadas pelos mesmos níveis lógicos utilizados para controlar o deslocamento para direita ou para esquerda (S_1S_0).

Para gerar os níveis lógicos para as entradas S_1 e S_0 do CI 74194, poderia ter sido usada uma simples porta

NOT (inversora), para garantir a complementaridade destas entradas, e mais uma chave de 1 pôlo \times 2 posições. Note que a posição da chave (um dos lados para *shit-left* e outro para *shit-right*) poderia ser comutada à qualquer instante de tempo. Mas como qualquer chave mecânica gera efeito *bouncing*, optou-se por incrementar um pouco o circuito e usar as entadas assíncronas de um simples FF-D (CI 7474) para gerar os sinais S_1 e S_0 para o CI 74194. Esta solução (mais "elegante"), usando botões pulsadores (*push-buttons*), ainda resolve o problema de *bouncing*. Neste caso, as saídas Q e \bar{Q} do CI 7474 foram usadas.

Como o CI 74195 possui ainda uma entrada de *Master-Reset* (pino \overline{MR}), preferiu-se incluir um circuito de inicialização baseado em rede RC para garantir que os FF's deste CI iniciem resetados.

O circuito completo é mostrado na figura 14.6.

14.2 Parte Prática

Monte o circuito mostrado na figura 14.6 e repare na sequencia de ativação dos leds conforme os botões pulsados são usados. Sugere-se montar uma pequena tabela verdade, relacionando pulsos de clock de entrada com o modo de operação do CI 74194 (entradas S_1S_0) e suas saídas ($Q_3Q_2Q_1Q_0$).

14.3 Perguntas

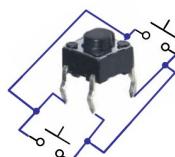
- 1) Complete a tabela verdade 14.1 associada com o circuito montado.
- 2) Monte um diagrama no tempo que retrate o comportamento observado na tabela verdade montada no item anterior.
- 3) Pergunta-se: é realmente necessário o circuito de *Reset* (inicialização) para o CI 74194?

No. Pulso de Clock	Modo S1 S0	Saídas				Entradas D_{SR} D_{SL}	Comentários
		Q3	Q2	Q1	Q0		
0	1 0						
1	1 0						
2	1 0						
3	1 0						
4	1 0						
5	1 0						
6	1 0						
7	1 0						
8	1 0						
9	1 0						
10	0 1						
11	0 1						
12	0 1						
13	0 1						
14	0 1						
15	0 1						
16	0 1						
17	0 1						
18	0 1						
19	0 1						
20	1 0						
21	1 0						

Observações:

- Considerar o pulso de clock 0 como o instante inicial em que o circuito inicia resetado.
- Preencha a tabela acima observando exatamente a sequência adotada para $S_1 S_0$.
- Notar que os bits S_1 e S_0 mudam de estado no meio do processo de deslocamento dos bits.

Tabela 14.1: Tabela verdade do circuito.



Uso do push-button.

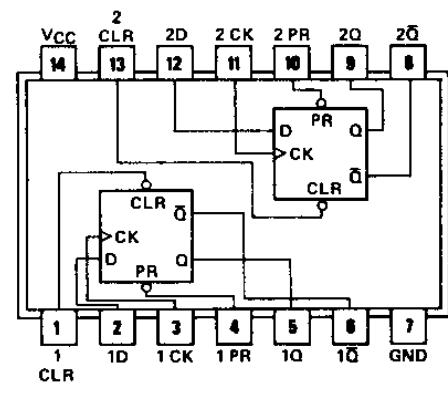
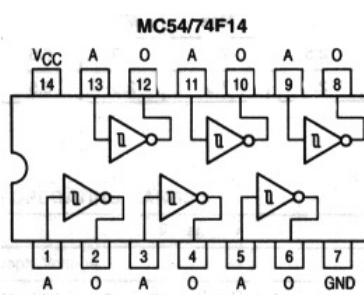
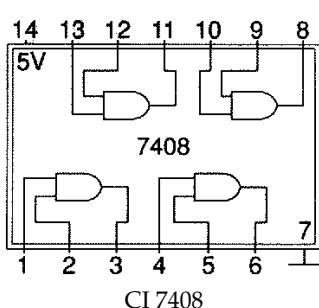


Figura 14.5: Pinagem de alguns componentes.

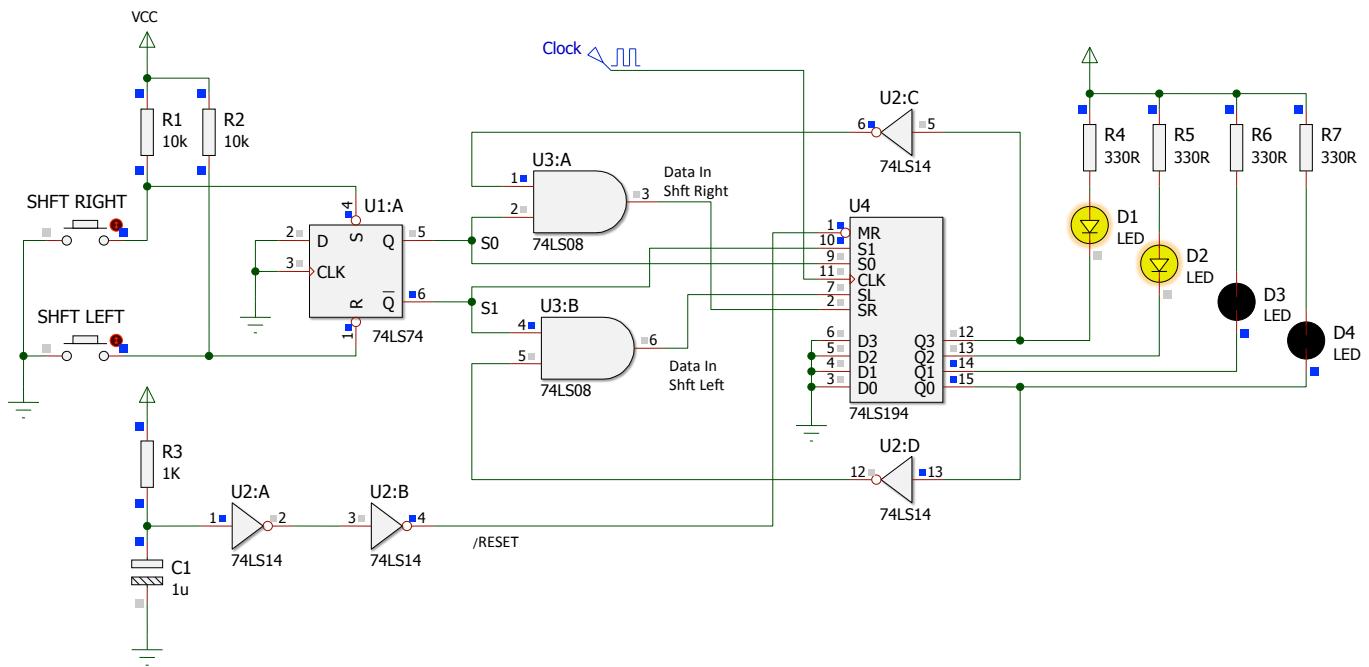


Figura 14.6: Circuito a ser montado.

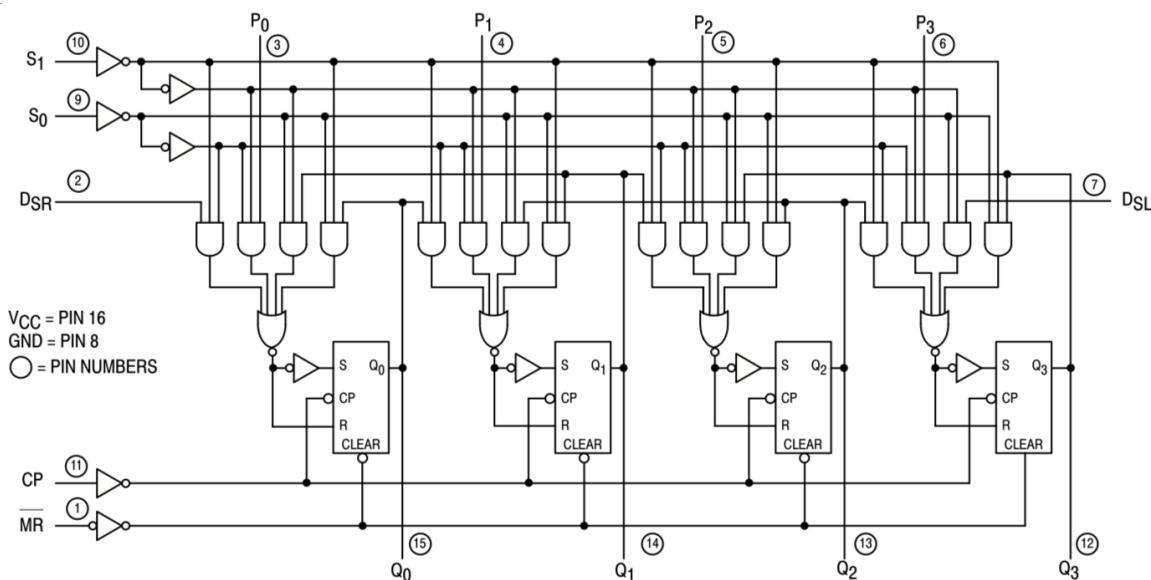


Figura 14.7: Diagrama lógico interno do 74LS194A (Motorola).

15 | Circuito Transmissor/Receptor Serial Síncrono

List de Material

Circuito Transmissor – Estudante: "TX"

- » 1 CI 74F04 (6 × NOT);
- » 1 CI 74F32 (4 × OR(2));
- » 1 CI 74LS90 (Contador Decimal);
- » 1 CI 74LS194 (4 bits Right/Left Shift Left register, ↑);
- » 1 Módulo de Display 7-segmentos (o segundo é opcional);
- » 1 Led;
- » 1 Resistor de 330 Ohms.
- » [Opcional] chave DIP 4 posições;
- » [Opcional] 4 resistores 10 KOhms.

Circuito Receptor – Estudante: "RX"

- » 1 CI 74LS90 (Contador Decimal);
- » 1 CI 74LS75 (4 bits Latch D, ALTO);
- » 1 CI 74LS164 (8 bits shift register, Serial In/Parallel Out, ↑);
- » 1 CI 74F04 (6 × NOT);
- » 1 módulo de Display 7-segmentos (o segundo é opcional);
- » 1 Led;
- » 1 Resistor de 330 Ohms.

Obs: este laboratório é para ser executado em duplas de alunos. Um dos estudantes monta o circuito transmissor enquanto o outro monta o receptor.

15.1 Fundamentos Teóricos

A ideia nesta prática é montar um **sistema de transmissão de 4 bits serial síncrono**. Para tanto necessitamos um circuito transmissor que prepare uma palavra de 4 bits para ser transmitida no formato serial (bit à bit), sincronizado com um sinal de clock. O circuito receptor deve ser capaz de receber bit à bit os dados que então recompor a palavra original de 4 bits que foi enviada.

15.1.1 Circuito Transmissor (TX)

O circuito de transmissão está baseado principalmente no CI **74LS194**, um registrador de deslocamento bidirecional e universal. Para tanto, ele dispõe de 2 entradas seriais: uma para

deslocamento para direita (*shift right* = pino 2 de entrada: D_{SR}) e outro para deslocamento para esquerda (*shift left* = pino 7 de entrada: D_{SL}). Este CI permite diferentes modos de operação podendo ser usado com registrador de entrada serial e saída serial (com deslocamento para a direita e para a esquerda), como registrador de entrada serial e saída paralela, como registrador de entrada paralela (operação de “Load” dos dados paralelos aplicados aos pinos de P_0 à P_3) e saída serial e como registrador de dados paralelo (entrada paralela/saída paralela). O modo de operação é selecionado pelos níveis lógicos aplicados aos pinos de entrada S_1 e S_0 conforme o que mostra a tabela 15.1.

Operating Mode	Inputs					Outputs				
	\overline{MR}	S_1	S_0	D_{SR}	D_{SL}	P_n	Q_0	Q_1	Q_2	Q_3
Reset	L	X	X	X	X	X	L	L	L	L
Hold	H	I	I	X	X	X	q_0	q_1	q_2	q_3
Shift Left	H	h	I	X	I	X	q_1	q_2	q_3	L
	H	h	I	X	h	X	q_1	q_2	q_3	H
Shift Right	H	I	h	I	X	X	L	q_0	q_1	q_2
	H	I	h	h	X	X	H	q_0	q_1	q_2
Parallel Load	H	h	h	X	X	P_n	p_0	p_1	p_2	p_3

H (h) = HIGH Voltage Level

L (l) = LOW Voltage Level

P_n (q_n) = Lower case letters indicate the state of the referenced input (or output) one setup time prior to the LOW-to-HIGH clock transition.

X = Immaterial

Observações:

l: significa apresentar nível lógico baixo ANTES da transição do sinal de clock de baixo para alto.

h: significa apresentar nível lógico alto ANTES da transição do sinal de clock de baixo para alto.

q_n : simboliza os estados de cada uma das saídas Q_n ANTES da transição do pulso de clock.

Tabela 15.1: Modos de operação do CI 74194.

Note que o 74LS194 trabalha com borda de subida do pulso de clock. E que para transmitirmos uma palavra de 4 bits se faz necessário primeiro realizar uma carga paralela dos bits que devem ser transferidos. Para isto, $S_1S_0 = 11_{(2)}$ (ver tabela 15.1) antes do pulso de subida de clock. Os dados que estavam nas entradas P_n vão surgir nas saídas Q_n depois da transição de nível lógico baixo para alto do sinal de clock. Depois, este registrador deve ser comutado para o modo de operação serial, fazendo-se: $S_1S_0 = 10_{(2)}$ (ver tabela 15.1) e deve permanecer neste modo nos próximos 3 pulsos de subida do sinal de clock, o que vão fazer que os bits sejam deslocados da esquerda para a direita. O primeiro flip-flop de entrada do registrador Q_0 vai

ser ajustado com o nível lógico que se encontrar na entrada D_{SR} . Por preferência pessoal esta entrada é deixada em nível lógico baixo. Assim, no 4º pulso de clock, todos os 4 primeiros FF's internos deste registrador estarão resetados.

Note que necessitamos de um circuito extra para controlar o momento de comutar o nível lógico na entrada S_0 (a única que necessita variar de nível). Para tanto, é montado um circuito contador de módulo 4 (conta de 0 à 3) que trabalhe sincronizado com o mesmo sinal de clock aplicado ao registrador 74LS194. O CI 74LS90 contador de década foi configurado como contador MOD-4 e junto às suas saídas foi adicionada uma lógica para fazer variar, no momento oportuno, o nível lógico na entrada S_0 do 74LS194. Desta forma, apenas quando as saídas do 74LS90 forem $B_A = 00_{(2)}$, surge nível lógico alto na entrada S_0 do 74LS194 (uso da porta NOR). Isto força um “parallel load” neste CI, cada vez que o contador de módulo 4 é reiniciado. Note ainda que este sinal surge instantes após a borda de descida do pulso de clock e que o registrador de deslocamento somente vai passar os dados P_n que estiverem presentes na sua entrada, para as saída Q_n no semi-ciclo seguinte de clock, quando ocorre a sua “subida” – isto garante que o registrador de deslocamento seja comutado da forma correta. Note que para outros estados de saída do CI contador, a entrada S_0 do registrador de deslocamento passa para nível lógico baixo o que faz com que o registrador de deslocamento universal passe a operar como um registrador de deslocamento serial para a direita ($S_1 S_0 = 10_{(2)}$).

15.1.2 Circuito Receptor (RX)

O circuito receptor deve trabalhar sincronizado com o sinal de clock enviado pelo circuito transmissor. Assim, à cada 4 pulsos de clock, ele deve ser capaz de mostrar num display de 7 segmentos o dado que querímos transmitir. Note que a palavra de 4 bits à ser recomposta, somente será completada no final do 4º pulso de clock, assim, também se faz necessário um circuito contador de módulo 4 a fim de indicar quando a palavra está pronta (ou quando foi enfim recebida). Este sinal pode ser usado para comandar o “enable” de um Latch de 4 bits que estaria entre o registrador de entrada serial/saída paralela (que recebe os dados), CI 74LS164, e o conjunto DEC/Display (que mostra a palavra de 4 bits recebida).

15.2 Parte Prática

A montagem do circuito será dividida entre 2 estudantes: um monta o circuito transmissor (TX) e o outro monta o circuito receptor (RX).

Nesta prática, iremos monitorar a palavra de 4 bits sendo transmitida e recebida através de Leds e módulos DEC/Display de 7 segmentos. Leds podem ser usados para verificar estados especiais do circuito de transmissão e recepção.

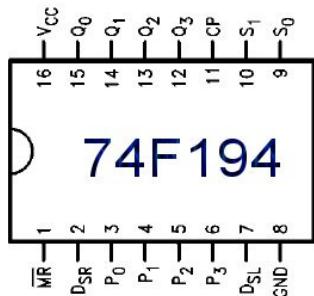
O circuito à ser montado aparece na fig. 15.2.

15.3 Questões

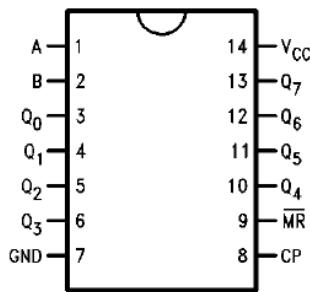
- 1) Complete um diagrama no tempo detalhando a operação de transmissão/recepção de uma palavra de 4 bits. Devem aparecer no diagrama o sinal de clock, saídas Q_0 e Q_1 do CI contador do circuito transmissor, a geração do sinal de “Parallel load” (usar a sigla “PL” simplesmente), as saídas Q_n do registrador de deslocamento universal (CI 74LS194), a representação dos bits de dados sendo enviados e lidos na entrada B no registrador de entrada serial/saída paralela (CI 74LS164) + suas saídas Q_n (Q_0 à Q_3 somente), a geração do sinal de “enable” (ou simplesmente “E”) para o Latch do circuito receptor e por fim, as saídas Q_n do Latch do circuito receptor. Suponha que se deseja transmitir o número $6_{10} = 0110_2$.

Ou seja, **resumindo, o diagrama no tempo deve mostrar** os seguintes sinais (por ordem): sinal de clock, saídas: Q_0 , Q_1 e Q_2 do 7490 do circuito transmissor, o sinal “PL” (pino S_0 do 74194), as saídas: Q_3 , Q_2 , Q_1 e Q_0 do 74194 do circuito transmissor, saídas Q_0 , Q_1 e Q_2 do 7490 do circuito receptor, a entrada “ D_{SB} ” do CI 74164 (bit sendo recebido, circuito receptor), as saídas Q_0 , Q_1 , Q_2 e Q_3 do 74164 (bits sendo deslocados) e sinal “E” (pinos E_{01} e E_{23} do CI 7475 (Latch de 4 bits) no final da recepção dos 4 bits de dados.

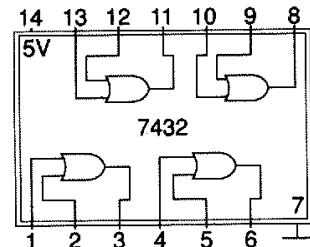
- 2) Complete o “histórico” anterior exemplificando que código decimal surge na saída do 71LS194 do circuito transmissor, a cada pulso de clock (supondo novamente que a palavra sendo transmitida fosse: $0110_2 = 6_{10}$).
- 3) Por fim, comente e justifique como funciona um circuito de transmissão/recepção serial de dados, mas assíncrono. Justifique a forma como os dados deveriam ser transmitidos (“protocolo de comunicação” envolvido no processo).



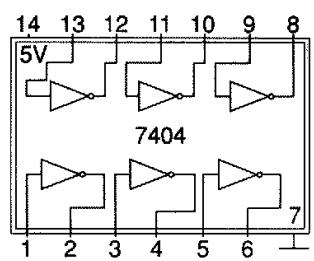
(a) Pinagem do CI 74LS194.



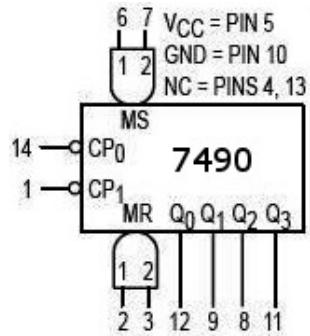
(b) Pinagem do CI 74F164A.



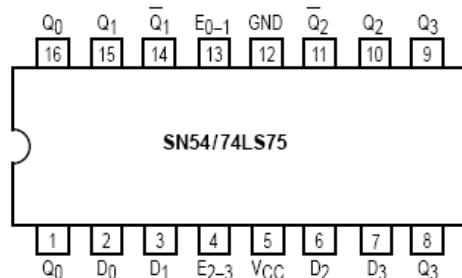
(c) Pinagem do CI 74LS32.



(d) Pinagem do CI 7404.



(e) Pinagem do CI 7490.



(f) Pinagem do CI 7475.

Figura 15.1: Pinagem dos CIs usados nesta prática.

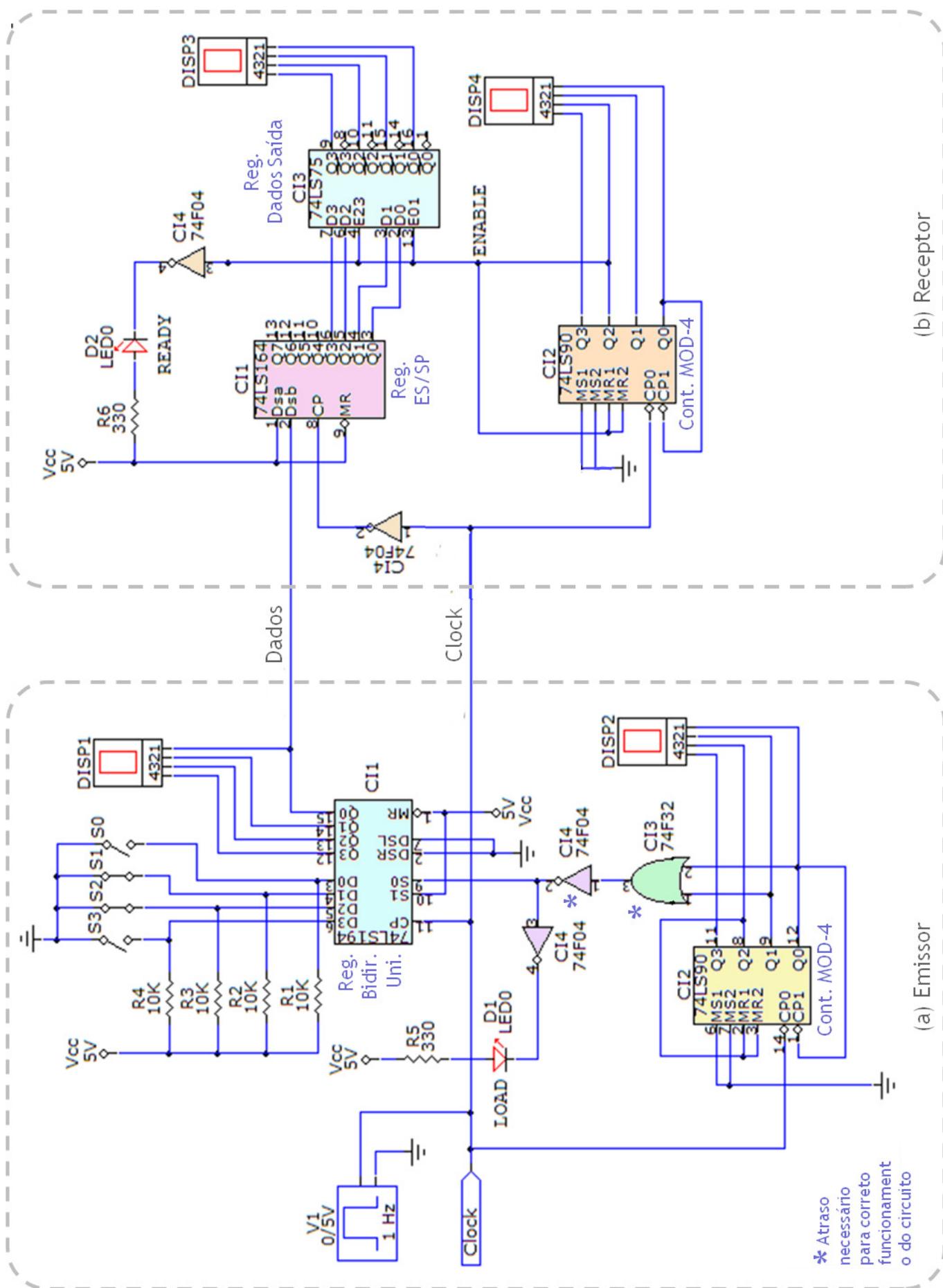


Figura 15.2: Circuito completo do transmissor/receptor serial síncrono.