

LIÇÃO 9

OS CONTADORES DIGITAIS

Na lição anterior analisamos o princípio de funcionamento de um dos mais importantes blocos da Eletrônica Digital, o *flip-flop*. Vimos que estes blocos poderiam ter diversos tipos de comportamento e que, quando reunidos, poderiam apresentar comportamentos interessantes como, por exemplo, a capacidade de dividir frequências, de armazenar informações (bits), além de outras. Nesta lição vamos nos dedicar justamente a uma das funções mais importantes dos *flip-flops* que é a de fazer a contagem do número de pulsos, o que corresponde em última análise a contagem de bits. A partir desta contagem podemos usar estes circuitos para a realização de operações mais complexas como somas, manipulação de dados etc.

9 - OS TIPOS DE CONTADORES

Em Eletrônica Digital devemos separar os circuitos lógicos sem sincronismo daqueles que possuam algum tipo de sincronismo externo, ou seja, que usam um sinal de *CLOCK*.

Existem aplicações em que tudo o que importa para o circuito é fazer uma operação com determinados níveis lógicos aplicados à sua entrada, quando eles estão presentes, não importando quando isso ocorra. Tais circuitos não precisam de sincronismo algum e são mais simples de serem utilizados.

No entanto, com circuitos muito complexos, como os utilizados em computadores e em muitos outros

casos, o instante em que uma operação deve ser realizada é muito importante e isso implica em que os circuitos devam ser habilitados no instante em que os níveis lógicos são aplicados em sua entrada.

Isso significa que tais circuitos devem ser sincronizados por algum tipo de sinal vindo de um circuito externo. E este circuito nada mais é do que um oscilador que produz um sinal de *clock* ou relógio.

Os circuitos que operam com estes sinais são denominados circuitos com lógica sincronizada.

Para os contadores temos então diversas classificações que levam em conta estes e outros fatores, por exemplo:

a) Classificação quanto ao sincronismo:

Os contadores podem ser ASSÍNCRONOS, quando existe o sinal de *clock* aplicado apenas ao primeiro estágio. Os estágios seguintes utilizam como sinal de sincronismo a saída de cada estágio anterior. Estes contadores também são denominados *Ripple Counters*.

Os contadores também podem ser SÍNCRONOS, quando existe um sinal de *clock* único externo aplicado a todos os estágios ao mesmo tempo.

b) Classificação quanto ao modo de contagem

Os contadores podem ser PROGRESSIVOS ou CRESCENTES, quando contam numa sequência de

números crescentes, ou seja, dos valores mais baixos para os mais altos, como (1,2,3,4...). São também chamados pelo termo inglês de *UP COUNTERS*.

Os contadores podem ser REGRESSIVOS ou DECRESCENTES, quando a contagem é feita dos valores mais altos para os mais baixos como (4,3,2,1...). O termo inglês é *DOWN COUNTERS*.

Se bem que possamos fazer contadores usando funções lógicas comuns e mesmo *flip-flops* discretos, podemos contar na prática com circuitos integrados em lógica TTL ou CMOS que já possuam contadores completos implementados.

9.1 - CONTADOR ASSÍNCRONO

Conforme explicamos, neste tipo de contador, o sinal de *clock* é aplicado apenas ao primeiro estágio, ficando os demais sincronizados pelos estágios anteriores.

Na figura 1 temos a estrutura básica de um contador deste tipo usando *flip-flops* do tipo J-K.

Usamos três estágios ou três *flip-flops* ligados de tal forma que a saída Q do primeiro serve de *clock* para o segundo, e a saída Q do segundo serve de *clock* para o terceiro.

Sabemos que os *flip-flops* ligados da forma indicada funcionam como divisores de frequência.

Assim, o sinal de *clock* aplicado ao primeiro tem sua frequência dividida por 2.

A frequência estará dividida por 4

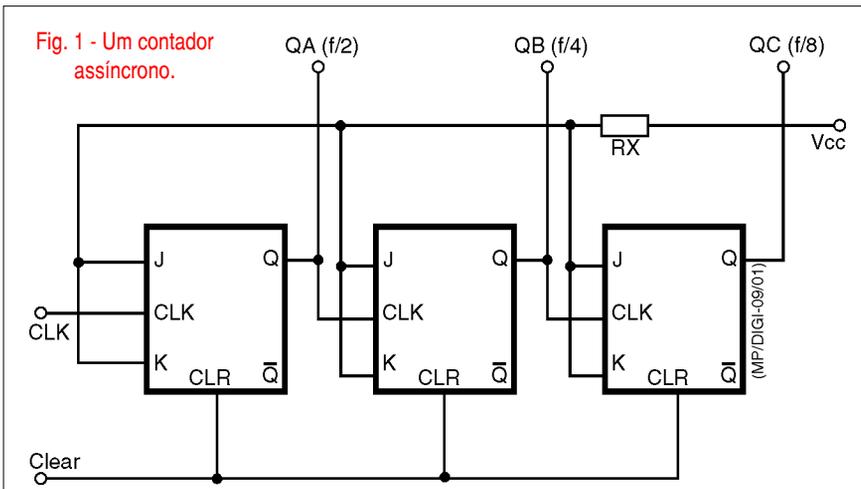


Fig. 1 - Um contador assíncrono.

Entrada	QA	QB	QC	Valor Binário
0	1	1	1	7
1	1	1	0	6
2	1	0	1	5
3	1	0	0	4
4	0	1	1	3
5	0	1	0	2
6	0	0	1	1
7	0	0	0	0

Portanto, este contador fornece em sua saída valores binários que correspondem à contagem decrescente dos pulsos de entrada, partindo de 7. Trata-se de um contador decrescente ou *DOWN COUNTER*.

na saída do segundo e por 8 na saída do terceiro.

Tudo isso pode ser visualizado pelo diagrama de tempos mostrado na figura 2.

Mas, se elaborarmos uma tabela verdade com os níveis lógicos obtidos na saída de cada um dos *flip-flops*, a cada pulso do *clock* aplicado, a partir do instante em que todas as saídas sejam zero, teremos algo interessante a considerar:

Entrada	QC	QB	QA
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Veja que a sequência de valores obtidos 000, 001, 010, 011, 100, 101, 110 e 111 corresponde justamente à contagem em binário dos pulsos de 0 a 7! Em outras palavras, este circuito conta os pulsos de entrada e fornece saídas que são a representação binária desta contagem.

Veja também que ele faz a contagem crescente, ou seja, de 0 até 7.

Se, em lugar de três *flip-flops*, usarmos quatro, no circuito mostrado na figura 3, teremos a contagem de 0000 a 1111, ou seja, uma contagem crescente de 0 a 15 pulsos.

Oito desses *flip-flops* ligados em série podem contar até 256 pulsos e com isso fornecer uma saída de 8 bits ou 1 byte.

O circuito apresentado comuta na transição negativa do sinal de *clock*.

Vamos supor agora que em lugar de usarmos como saídas de contagem as saídas Q de cada *flip-flop*, usássemos as saídas complementares /Q, conforme a figura 4. É fácil perceber que, partindo da situação em que todos os *flip-flops* estejam ressetados, a tabela verdade obtida terá nas saídas os complementos da tabela anterior. Esta tabela será:

Como no caso anterior, se tivermos mais *flip-flops*, podemos contar a partir de valores mais altos. Com 4 *flip-flops* podemos partir a contagem de 15 e com 8 *flip-flops*, de 255. Veja que a quantidade máxima que podemos contar com um contador deste tipo depende da quantidade de *flip-flops* usados.

Um problema que ocorre com este tipo de *flip-flop* é que cada um precisa de um certo tempo para mudar de

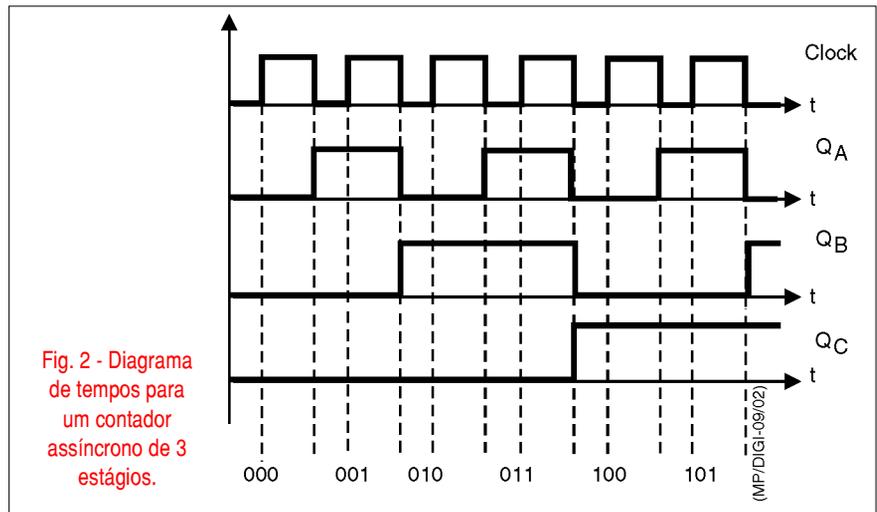


Fig. 2 - Diagrama de tempos para um contador assíncrono de 3 estágios.

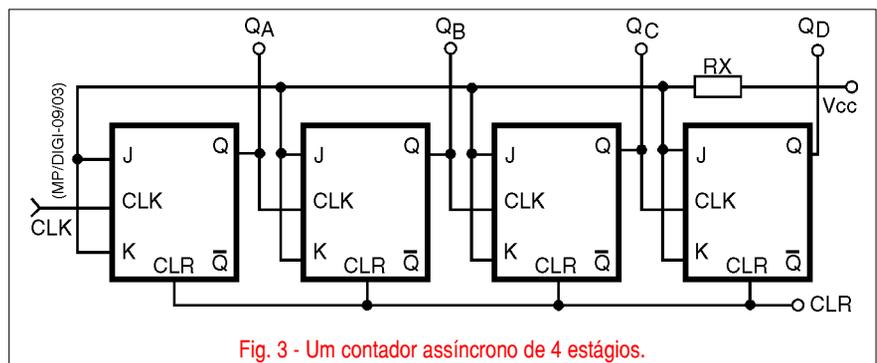


Fig. 3 - Um contador assíncrono de 4 estágios.

estado. Isso significa que à medida que usamos mais *flip-flops* em sequência num contador, os tempos de mudança de estado são somados e o conjunto precisa cada vez de mais tempo para chegar ao estado final desejado.

Se aplicarmos um novo pulso de *clock* para contagem à entrada do circuito, antes de ocorrer a mudança de estado do conjunto, pode ocorrer um funcionamento errático. Assim, a frequência máxima de operação de um contador é dada pelo tempo necessário para cada estágio mudar de estado multiplicado pelo número de estágios usados no contador.

9.2 - CONTAGEM PROGRAMADA

Conforme vimos, os ciclos de contagem dos circuitos dados como exemplos no item anterior são sempre potências de 2, ou seja, são circuitos que contam até 2, 4, 8, 16, 32 etc. O que fazer se precisarmos de um circuito que tenha um ciclo de contagem diferente desses valores, que não seja uma potência de 2?

Devemos levar em conta dois fatores:

Podemos usar a entrada *CLEAR* para reiniciar a contagem, zerando-a, quando chegar ao valor desejado. Por exemplo, podemos reiniciar a contagem depois do 5 se quisermos um contador que conte de 0 a 5, ou seja, que tenha 6 estados de saída, conforme a tabela verdade dada a seguir:

Entrada	QC	QB	QA
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	0	0	0

estado instável

No sexto pulso que corresponde ao estado 110, o circuito vai a um estado que ativa a entrada *CLEAR* e leva todos os *flip-flops* a serem ressetados.

Para este circuito a solução é simples. Veja que a situação em que devemos ter a volta a zero da contagem

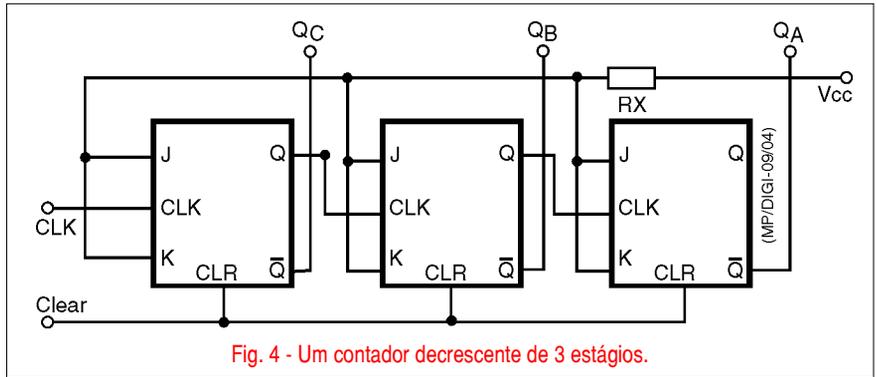


Fig. 4 - Um contador decrescente de 3 estágios.

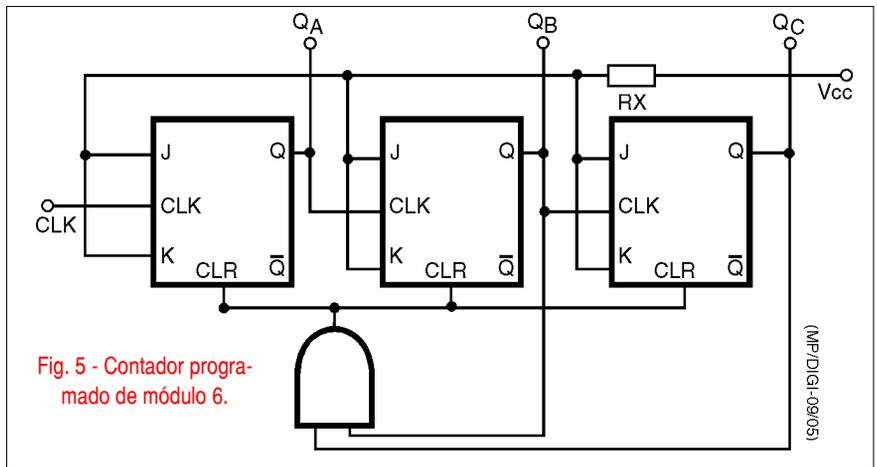


Fig. 5 - Contador programado de módulo 6.

e portanto, a ativação da linha CLR (*clear*) ocorre com uma única combinação de sinais: QA e QB no nível alto.

Se usarmos *flip-flops* que tenham entradas *CLEAR* ativadas pelo nível alto, basta usar uma porta AND de duas entradas com as entradas ligadas nas saídas QB e QC e a saída na linha comum de *CLEAR* de todos os *flip-flops*, conforme a figura 5. Se os *flip-flops* usados tiverem um *CLEAR* ativado no nível baixo como o 7476 (TTL), basta usar uma porta NAND em lugar de AND.

Se quiséssemos um contador até 4, por exemplo, o estado em que deveria ocorrer a ativação da entrada *CLEAR* ocorreria com a quinta combinação de saídas, ou seja, 101, o que significa QC=1 e QA=1. Bastaria então ligar as entradas da porta AND nessas saídas, conforme a figura 6.

Um diagrama de tempos pode mostrar exatamente o que ocorre com o contador elaborado desta maneira. Este diagrama é apresentado na figura 7. Observe que, quando as saídas chegarem ao estado 110, que

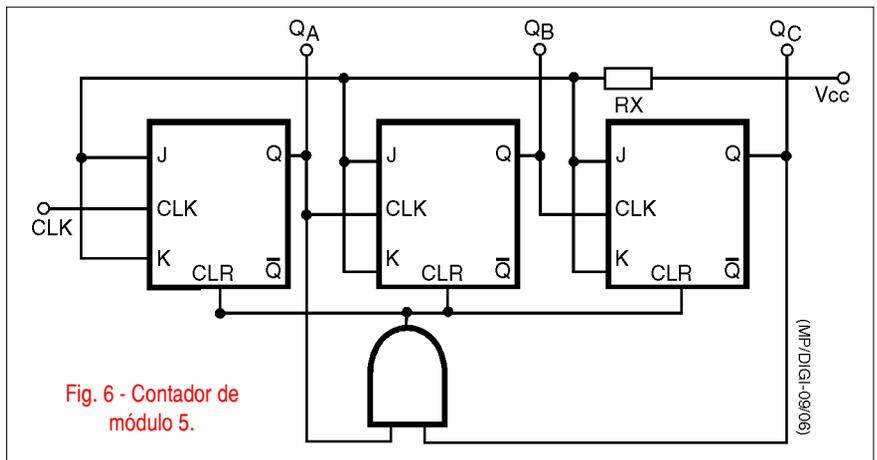


Fig. 6 - Contador de módulo 5.

seria a contagem do quinto pulso no circuito da figura 6, um pulso de reset de curta duração é produzido. Esta curta duração é dada justamente pelo tempo que os *flip-flops* demoram para mudar de estado ressetando, pois eles “realimentam” as entradas da porta AND.

Nos exemplos dados fizemos a programação da contagem usando as entradas de *CLEAR* de cada *flip-flop*.

Uma outra maneira de projetarmos um contador consiste em usarmos as entradas *PRESET* em lugar de *CLEAR*.

Para isso fazemos com que, no momento em que for atingida a contagem do valor imediatamente anterior àquele em que deve ocorrer a volta a zero, ou seja, $n-1$, em lugar de termos a comutação dos *flip-flops*, tenhamos a ativação das entradas de *PRESET*. Desta forma, no pulso seguinte de *clock* (n) teremos a volta a zero (reset) do contador.

Para um contador de 6 estados, que depois do quinto pulso resseta, teremos a seguinte tabela verdade.

Pulsos	QC	QB	QA
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
o PRESET é ativado			
x x x x			
volta a zero na transição do clock			
6	0	0	0
7	0	0	1
8	0	1	0

Um circuito usando uma porta NAND é mostrado na figura 9.

Veja que a detecção da condição de produção do pulso de PRESET deve ser reconhecida com os níveis 101 nas saídas dos estágios dos contadores e com o pulso indo ao nível alto na entrada de contagem.

Para obtermos a configuração 1111 que nos permitiria usar uma porta AND de quatro entradas, basta levar em conta a saída /QB em lugar de QB.

Assim, basta usar a porta AND e ligá-la nas entradas de *PRESET* (PR) dos *flip-flops*.

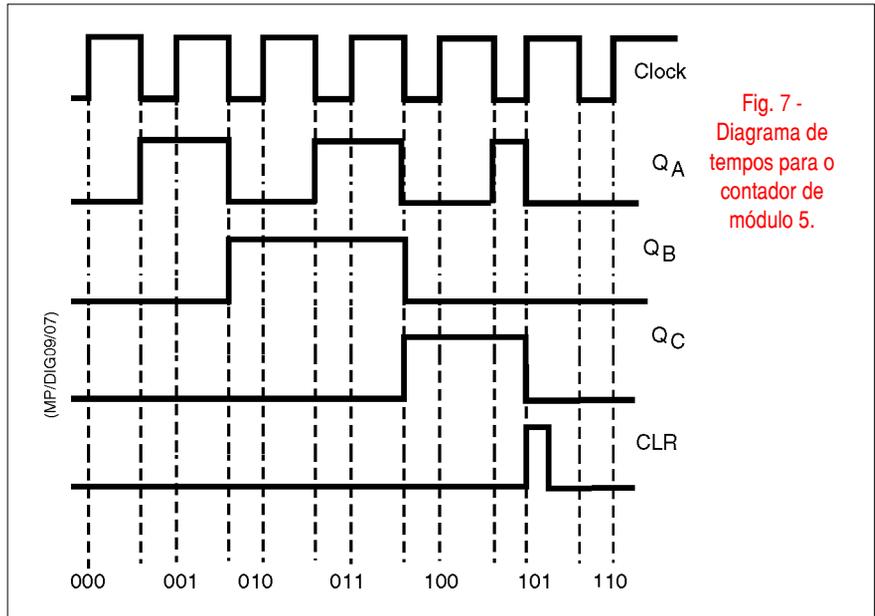


Fig. 7 - Diagrama de tempos para o contador de módulo 5.

Se as entradas forem ativadas no nível baixo (/PR), basta trocar a porta AND por uma porta NAND de quatro entradas.

9.3 - CONTADORES UP/DOWN (PROGRESSIVOS E REGRESSIVOS)

Usando alguns artifícios, como por exemplo, portas apropriadas, é possível programar um contador de modo que ele tanto conte progressivamente como regressivamente.

Usando 3 estágios, podemos ter um contador UP/DOWN, conforme a figura 10. Uma entrada (UP/DOWN) pode ser usada para determinar o sentido da contagem.

Trata-se de uma entrada seletora de dados ou DATA SELECTOR, que pode ser usada para mudar o modo de funcionamento dos estágios deste circuito.

Funcionamento: conforme vimos nesta lição, se usarmos as saídas Q dos *flip-flops* de um contador, a contagem será crescente, mas se usarmos as saídas /Q, a contagem será

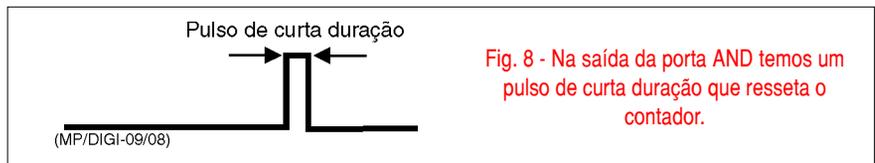


Fig. 8 - Na saída da porta AND temos um pulso de curta duração que resseta o contador.

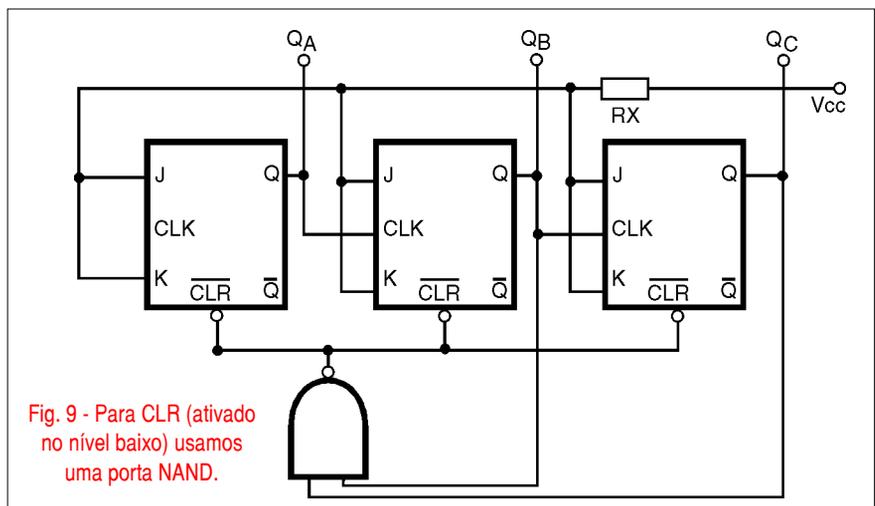


Fig. 9 - Para CLR (ativado no nível baixo) usamos uma porta NAND.

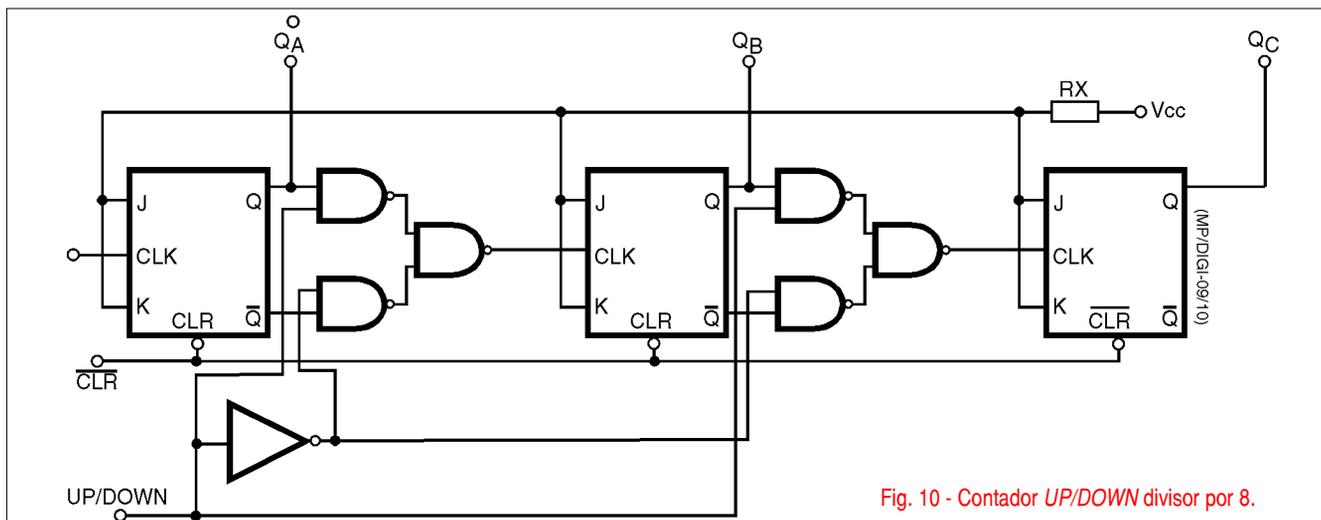


Fig. 10 - Contador UP/DOWN divisor por 8.

decrecente. Assim, o que fazemos é colocar um circuito seletor nessas saídas, de tal modo que ele coloque a saída Q de cada flip-flop na entrada de clock do seguinte, quando a contagem deve ser progressiva, e coloque a saída /Q na entrada do seguinte, quando na contagem decrescente. Três portas NAND para cada estágio podem fazer isso a partir do sinal de comando UP/DOWN.

9.4 - CONTADORES SÍNCRONOS

Sincronizar a contagem por um clock único aplicado a todos os estágios não é apenas uma necessidade dos circuitos mais complexos, principalmente, os usados em Informática e Instrumentação.

O sincronismo de todos os estágios pelo mesmo clock tem ainda vantagens operacionais importantes. Conforme vimos, nos contadores

assíncronos, os tempos de comutação dos flip-flops influem no funcionamento final do circuito, pois eles são cumulativos.

Em outras palavras, cada estágio precisa esperar o anterior completar a operação antes de iniciar a sua.

Usando lógica sincronizada, ou seja, um contador em que todos os estágios são sincronizados por um clock único, este problema não existe e podemos ter contadores muito mais rápidos, na verdade, contadores cuja velocidade independe do número de etapas.

Para mostrar como isso pode ser feito, vamos tomar como exemplo o circuito da figura 11.

Este circuito utiliza flip-flops tipo J-K ligados de uma forma denominada PARALLEL CARRY.

Nesta forma de ligação, J e K do primeiro flip-flop são mantidas no nível alto por meio de um resistor ligado ao positivo da alimentação (Vcc).

Assim, o primeiro flip-flop muda de estado a cada pulso de clock. No entanto, J do segundo flip-flop está ligado à saída Q do primeiro. Isso significa que o segundo flip-flop só mudará de estado quando o primeiro flip-flop for ressetado, ou seja, a cada dois pulsos de clock.

Da mesma forma, com o uso de uma porta AND, o terceiro flip-flop só vai mudar de estado quando as saídas Q do primeiro e segundo flip-flop forem ao nível 1, ou seja, a cada 4 pulsos de clock.

Para 4 bits, utilizando 4 estágios, podemos usar o circuito mostrado na figura 12.

Um problema que ocorre com este tipo de configuração é que a partir de 3 estágios, a cada estágio que acrescentamos no contador devemos adicionar uma porta AND cujo número de entradas vai aumentando.

Assim, para 4 estágios, a porta deve ter três entradas, para 5 estágios, 4 entradas e assim por diante.

Uma maneira de não termos este problema consiste em usar uma configuração diferente de contador apresentada na figura 13 e denominada RIPPLE CARRY.

Neste circuito as portas usadas sempre precisam ter apenas duas entradas, o que é importante para a implementação prática do contador.

No entanto, como desvantagem deste circuito, temos uma limitação da velocidade de operação, pois como o sinal para os estágios vem da porta anterior, temos de considerar seu atraso.

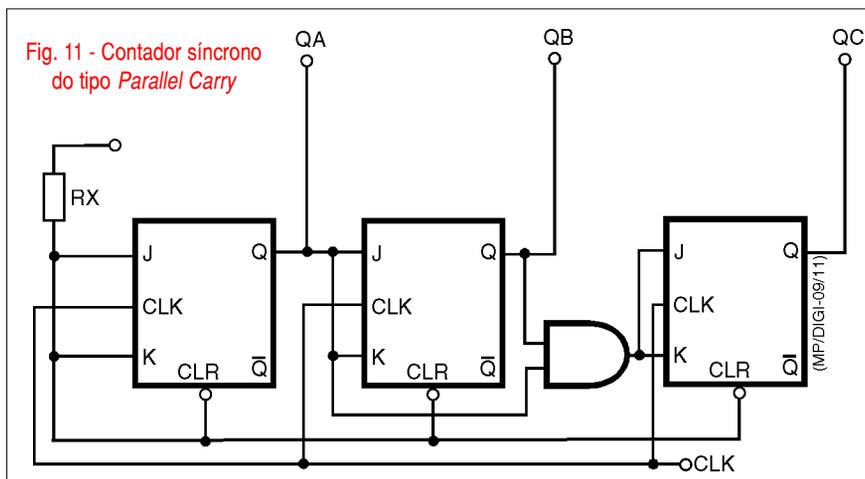


Fig. 11 - Contador síncrono do tipo Parallel Carry

9.5 - CONTADORES SÍNCRONOS PROGRAMÁVEIS

Da mesma forma que no caso dos contadores assíncronos, também é necessário, em determinadas aplicações, fazer a contagem até valores que não sejam potências de 2.

A divisão ou contagem por outros valores pode ser feita com a ajuda de portas ligadas de modo a “sentir” quando um determinado valor é alcançado, ressetando então todos os *flip-flops*.

9.6 - CONTADORES TTL

Utilizando portas lógicas e *flip-flops* podemos implementar contadores que contemham ou façam a divisão de um sinal de entrada por qualquer valor. No entanto, na prática, podemos contar com muitos circuitos integrados em tecnologia TTL que já contemham estes circuitos completos num único chip e até com recursos que permitam alterar seu funcionamento de modo a ser obtida a contagem até um determinado valor.

A seguir veremos alguns dos principais circuitos integrados contadores em tecnologia TTL.

a) 7490 - Contador de Década

Este é um dos mais populares dos contadores TTL e contém em seu interior quatro *flip-flops* já interligados de modo a funcionar como divisores por 2 e por 5. Isso significa que esses divisores podem ser usados para resultar num contador até 2 e num contador até 5, e em conjunto, num contador até 10.

Na figura 14 temos a disposição dos terminais deste circuito integrado.

Este circuito pode ser usado de três formas diferentes, sempre com as entradas R0(1), R0(2), R9(1) e R9(2) aterradas:

Quando ligamos a entrada B à saída QA e aplicamos o sinal de *clock* à entrada A, o circuito funciona como um contador BCD, ou seja, conta até 10, com as saídas em decimal codificado em binário apresentadas nos pinos QA, QB, QC e QD. Esta ligação é mostrada na figura 15.

A tabela verdade para os pulsos aplicados na entrada neste modo de funcionamento será:

Pulso	QD	QC	QB	QA
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

Quando ligamos a saída QD à entrada A e aplicamos o sinal de *clock* à entrada B, teremos o circuito funcionando como um divisor de frequência por 10 simétrico. Teremos na saída QA um sinal quadrado (ciclo ativo de 50%) com 1/10 da frequência do *clock*.

Este modo de funcionamento tem as ligações mostradas na figura 16.

Finalmente, quando quisermos usar o circuito como divisor por 2 ou por 5, independentes, não é preciso ligação externa alguma.

O sinal aplicado em CLK1 tem a frequência dividida por 2 e o sinal

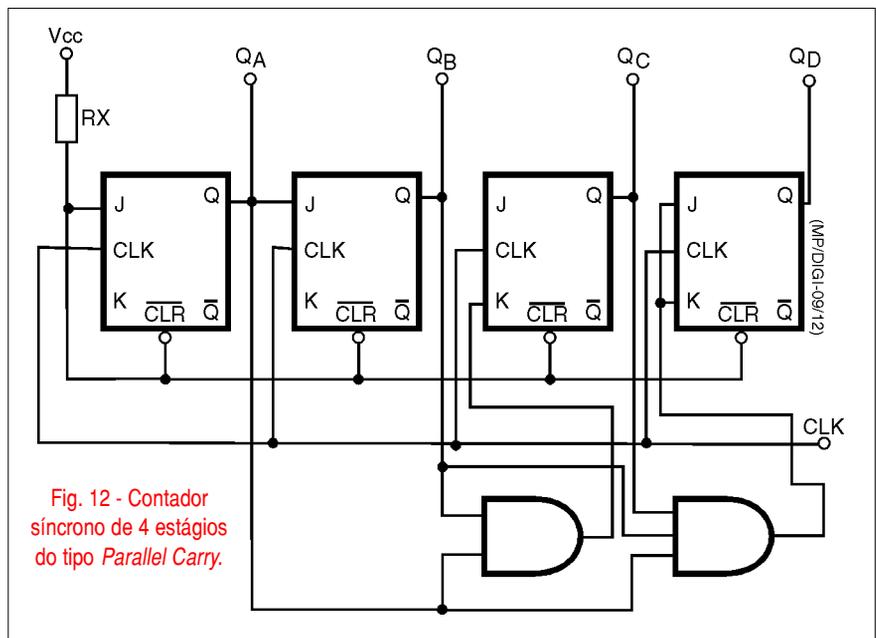


Fig. 12 - Contador síncrono de 4 estágios do tipo Parallel Carry.

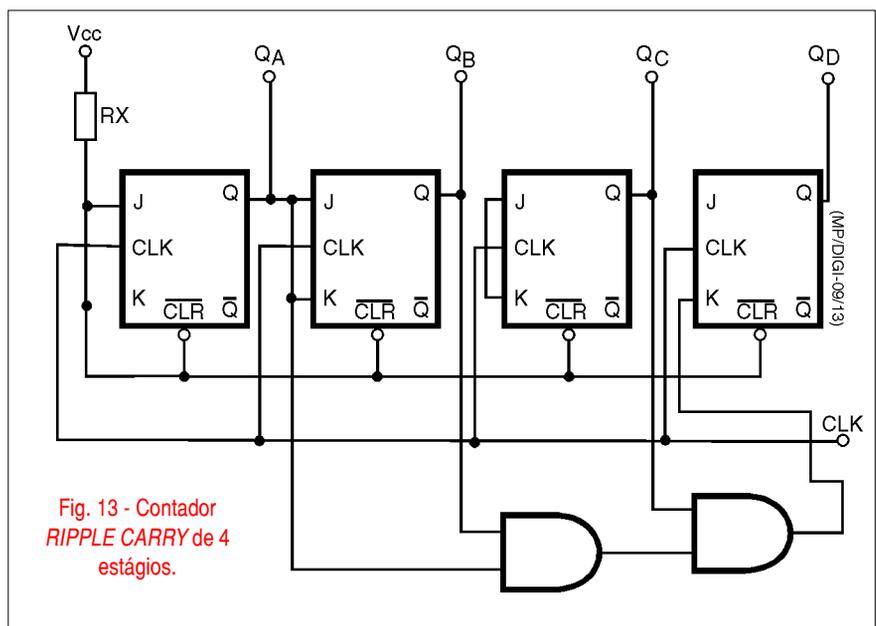


Fig. 13 - Contador RIPPLE CARRY de 4 estágios.

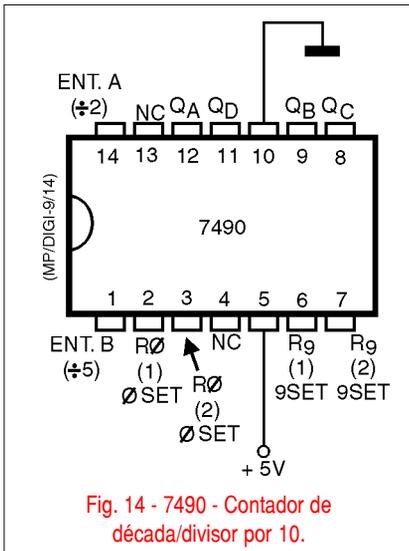


Fig. 14 - 7490 - Contador de década/divisor por 10.

aplicado no CLK2 tem a frequência dividida por 5. Na operação normal as entradas R0(1) e R0(2) devem ser mantidas no nível baixo.

b) 7492 - Contador-Divisor por 12

Este circuito integrado contém quatro *flip-flops* ligados como um divisor por 2 e um divisor por 6 que podem ser usados de maneira independente.

A pinagem deste circuito integrado TTL é mostrada na figura 17.

O disparo dos *flip-flops* ocorre na transição do sinal de *clock* do nível alto para o nível baixo. Para ressetar o contador para 0000, basta aplicar o nível lógico 1 nas entradas R0.

Existem três modos de operação para este circuito integrado:

Como contador até 12, basta ligar a saída QA à entrada B. O sinal de *clock* é aplicado à entrada A.

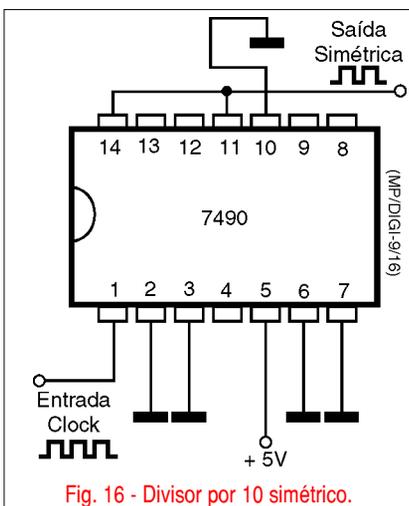


Fig. 16 - Divisor por 10 simétrico.

A tabela verdade para este modo de operação será:

Entrada	QD	QC	QB	QA
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1

Na segunda forma de operação, ligamos a saída QD à entrada A. O circuito funcionará como um divisor simétrico de frequência. A frequência do sinal de *clock* aplicado à entrada B será dividida por 12 e o sinal terá um ciclo ativo de 50%.

Na operação sem nenhuma ligação externa, o sinal aplicado à entrada A terá sua frequência dividida por 2 e o sinal aplicado na entrada B terá sua frequência dividida por 6.

9.7 - CONTADORES E DIVISORES CMOS

Temos ainda diversos circuitos integrados em tecnologia CMOS contendo contadores e divisores.

A seguir veremos um dos mais importantes.

Na operação normal, contando até 10, as entradas RST e EN devem ser mantidas no nível baixo.

Levando-se a entrada RST ao nível alto, o contador é ressetado. Se a entrada EN for levada ao nível alto, a contagem é paralisada.

Na figura 18 temos as formas de onda deste contador, mostrando de que forma em cada instante temos sempre apenas uma saída no nível alto.

Como em todos os circuitos CMOS, a frequência máxima de contagem depende da tensão de alimentação. Para 10 V, a frequência máxima é da ordem de 5 MHz.

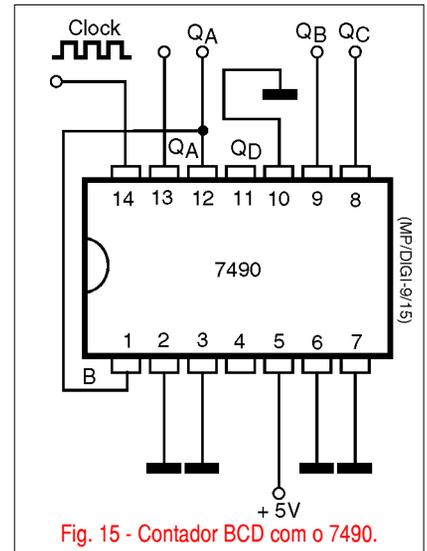


Fig. 15 - Contador BCD com o 7490.

4018 - Contador/Divisor Por N

Este circuito integrado, que será melhor analisado na próxima lição, pode fazer a divisão ou contagem de pulsos em valores até 10 programados pelas ligações externas.

Sua pinagem é mostrada na figura 19 e seu uso será visto posteriormente.

QUESTIONÁRIO

1. Que tipo de contador tem cada estágio controlado pelo anterior, com o sinal de *clock* aplicado apenas ao primeiro estágio?

- a) Síncrono
- b) Assíncrono
- c) *Ripple Counter*
- d) Contador de década

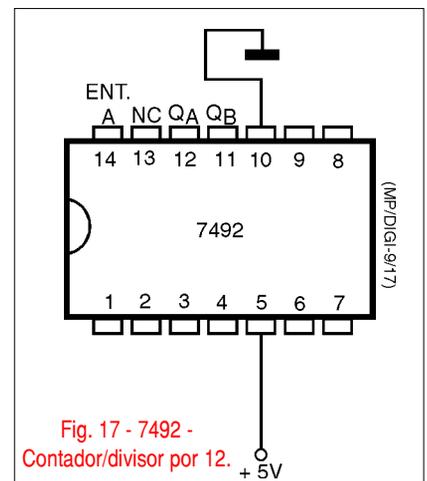


Fig. 17 - 7492 - Contador/divisor por 12.

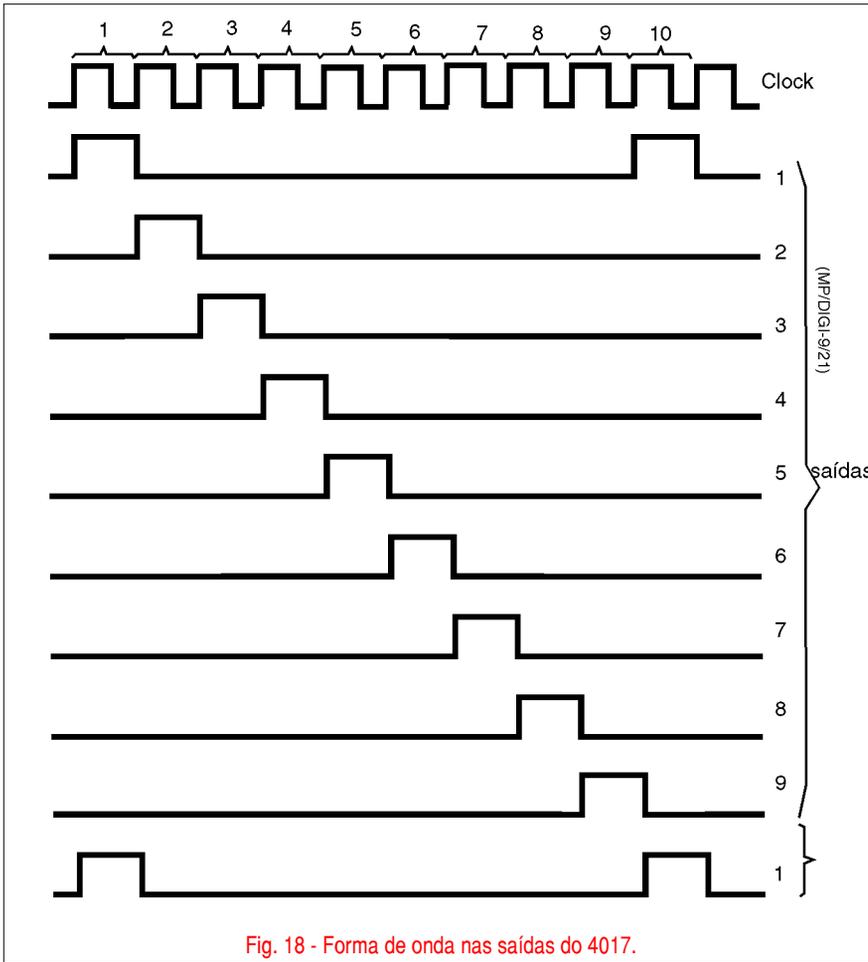


Fig. 18 - Forma de onda nas saídas do 4017.

2. Qual é o valor máximo de contagem de um contador que use 4 *flip-flops*?

- a) 4
- b) 8
- c) 16
- d) 10

3. Para um contador de 4 estágios, um do tipo síncrono e outro assíncrono, qual é o mais rápido?

- a) O contador síncrono
- b) O contador assíncrono
- c) Ambos têm a mesma velocidade
- d) Depende do modo como são usados

4. Podemos fazer a contagem até valores que não sejam potências de 2 usando que tipos de circuitos?

- a) Contadores comuns sozinhos
- b) Contadores comuns e funções lógicas
- c) Somente funções lógicas complexas

d) Não é possível fazer isso

5. Qual dos contadores/divisores abaixo relacionados tem saídas do tipo 1-de-10?

- a) 7400 b) 7490
- c) 74190 d) 4017

Resp.: 1-b, 2-c, 3-a, 4-b, 5-d

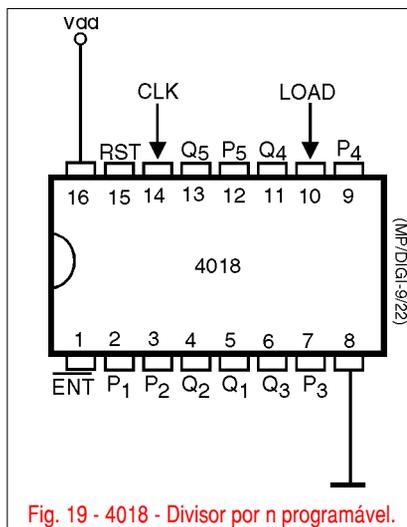


Fig. 19 - 4018 - Divisor por n programável.