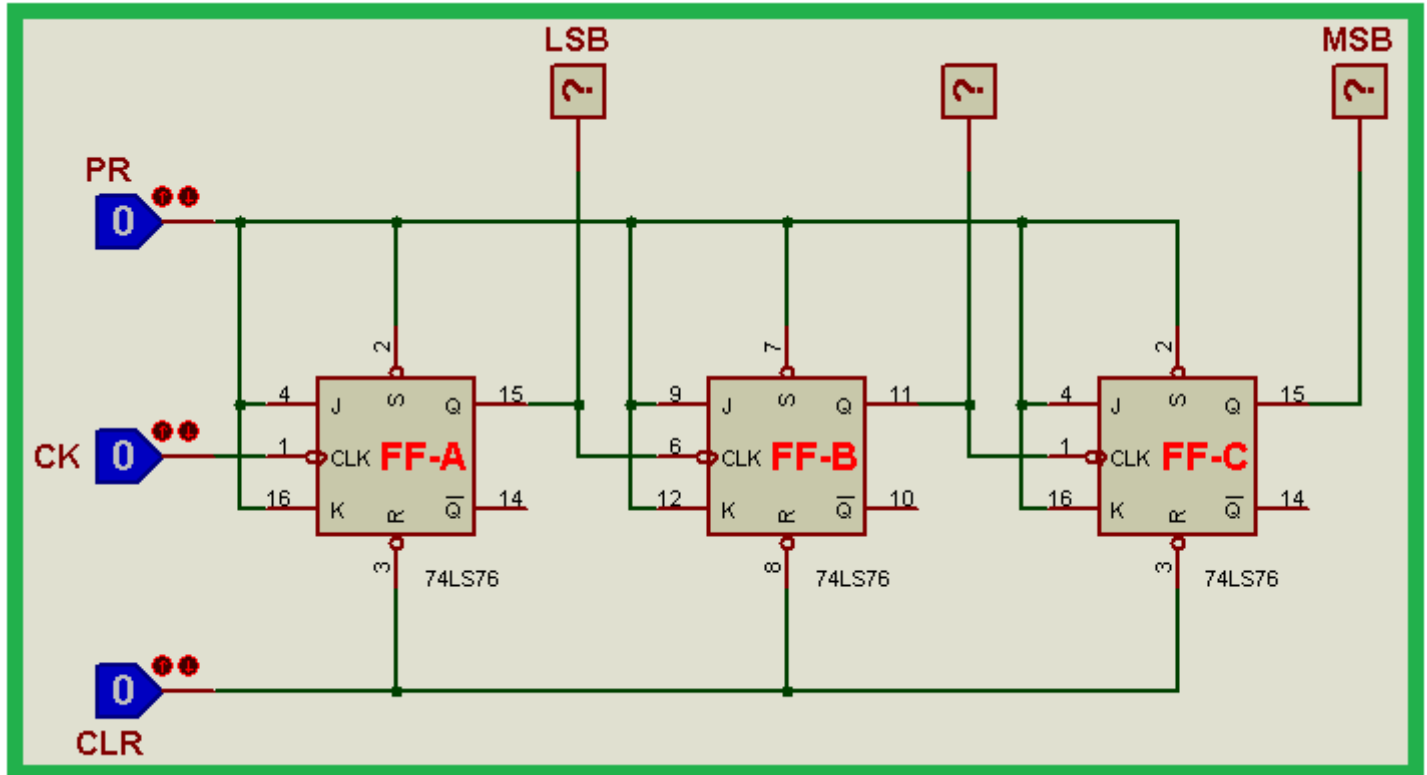


CONTADORES ASSÍNCRONOS

Implementação com FFs JK – configurado em tipo "T"

Os contadores assíncronos caracterizam-se por não possuírem entradas comuns de clock, ou seja, o pulso de clock é aplicado apenas no primeiro FF.

A figura abaixo ilustra um contador assíncrono de 3 bits, implementado com FF tipo T, onde se observa que o pulso de clock é aplicado apenas no primeiro FF.



Observações Importantes:

- 1 – Todos os FFs estão configurados no modo de operação "Toggle".
- 2 – Entradas assíncronas são ativas em "0", portanto, para liberar o FF ambas devem estar em nível lógico "1".
- 3 – Os FFs são ativados na transição negativa do pulso do clock, ou seja, na transição H-L (nível alto para nível baixo).

Em se tratando de um contador de 3 bits (cada bit é representado por um FF), a contagem é de 8 dígitos, ou seja, de 0 a 7 e a partir do 8º pulso de clock o contador é zerado e a contagem recomeça a partir do 9º pulso de clock.

Como um determinado pulso é transmitido de um FF para outro, esse contador recebe também o nome de "contador de pulsos" ou "contador de ripple".

A tabela a seguir mostra o comportamento das saídas de cada FF na ocorrência dos pulsos de clock, lembrando que a ativação dos FFs ocorre na transição H-L.

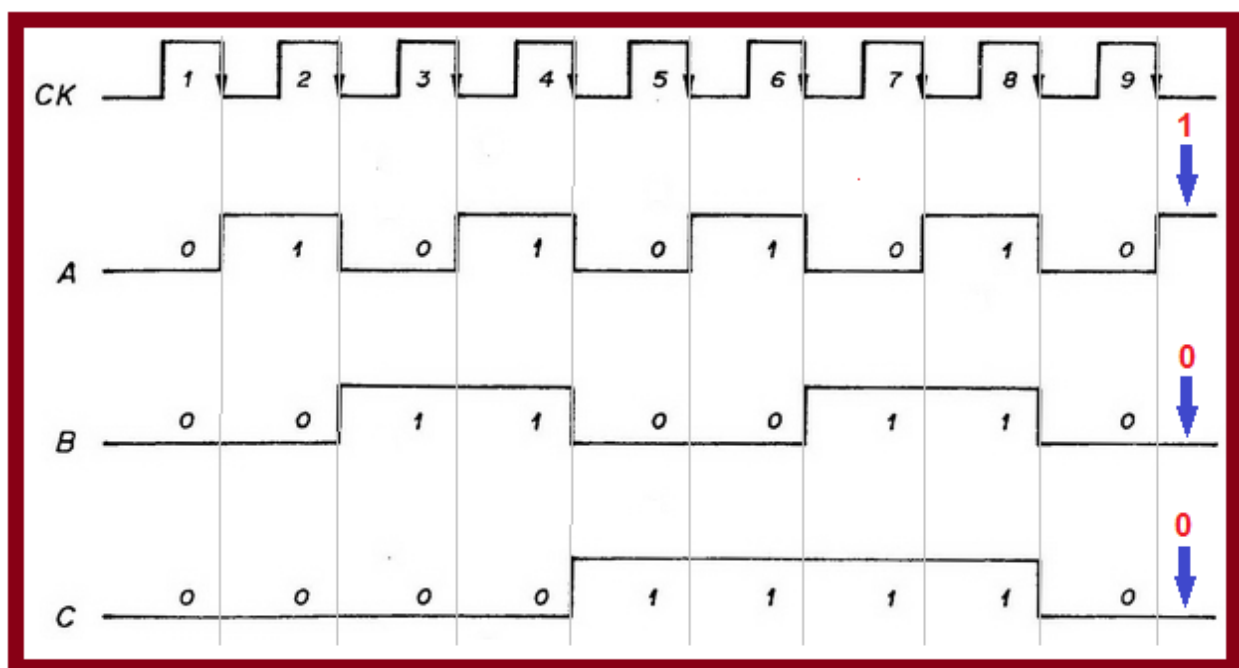
Para melhor entendimento do funcionamento desse tipo de contador os FFs foram identificados com as letras de A (LSB) até C (MSB).

CK	Saídas dos FFs			Saída binária/decimal
	C	B	A	
ligado	0	0	0	000 / 0
1	0	0	1	001 / 1
2	0	1	0	010 / 2
3	0	1	1	011 / 3
4	1	0	0	100 / 4
5	1	0	1	101 / 5
6	1	1	0	110 / 6
7	1	1	1	111 / 7
8	0	0	0	000 / 0
9	0	0	1	000 / 1

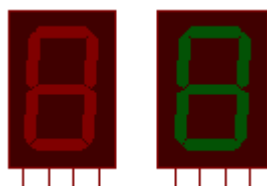
Veja abaixo as formas de onda (gráfico) correspondentes a tabela.

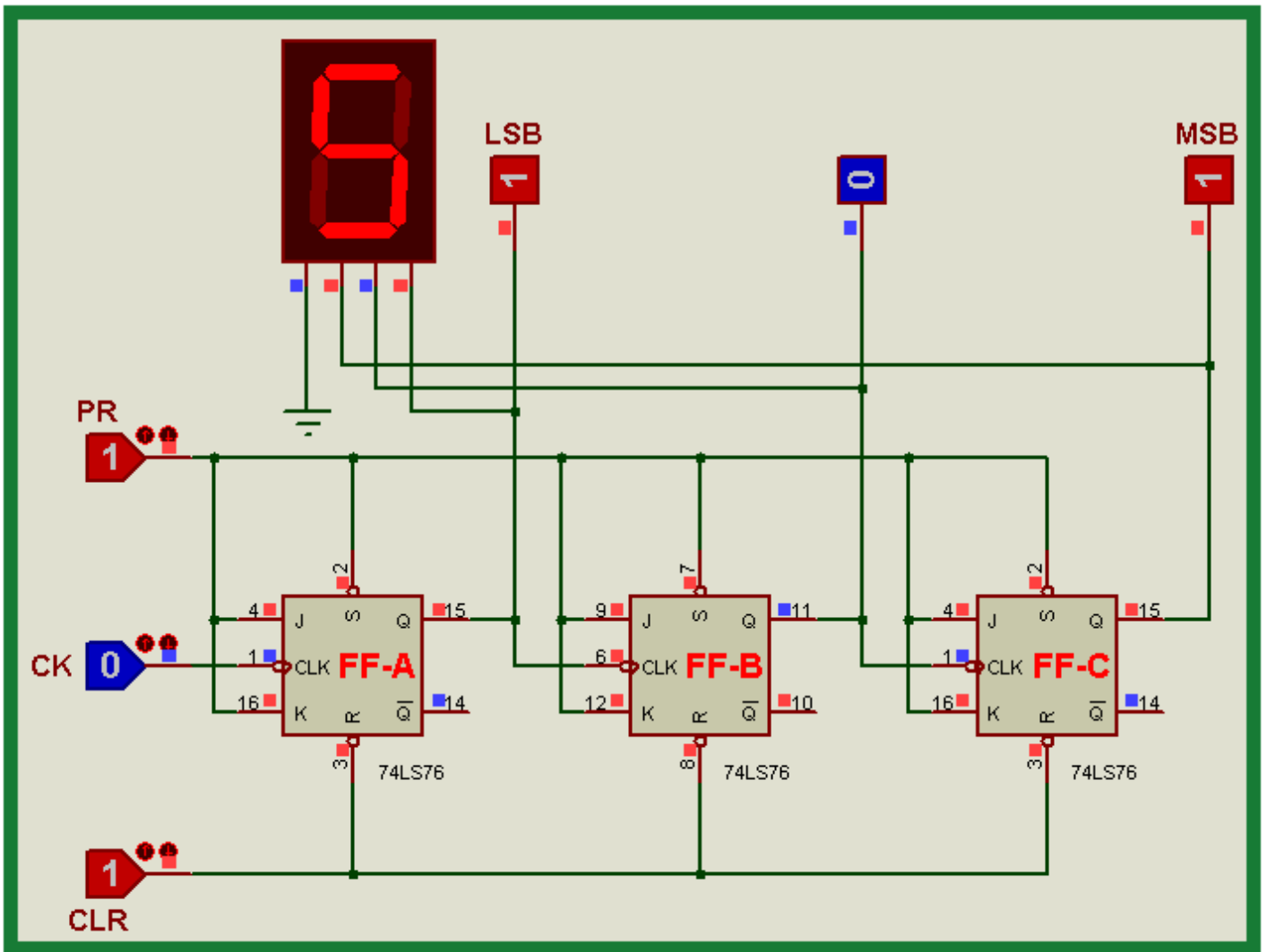
No 8º pulso de clock o contador é zerado.

A partir do 9º pulso de clock a contagem reinicia.



A figura a seguir mostra o mesmo circuito com um display BCD-7 segmentos interligado às saídas. O referido display já possui internamente um decodificador BCD 8421 para 7 segmentos, que permite mostrar os caracteres decimais de 0 a 9 e os hexadecimais de A até F.





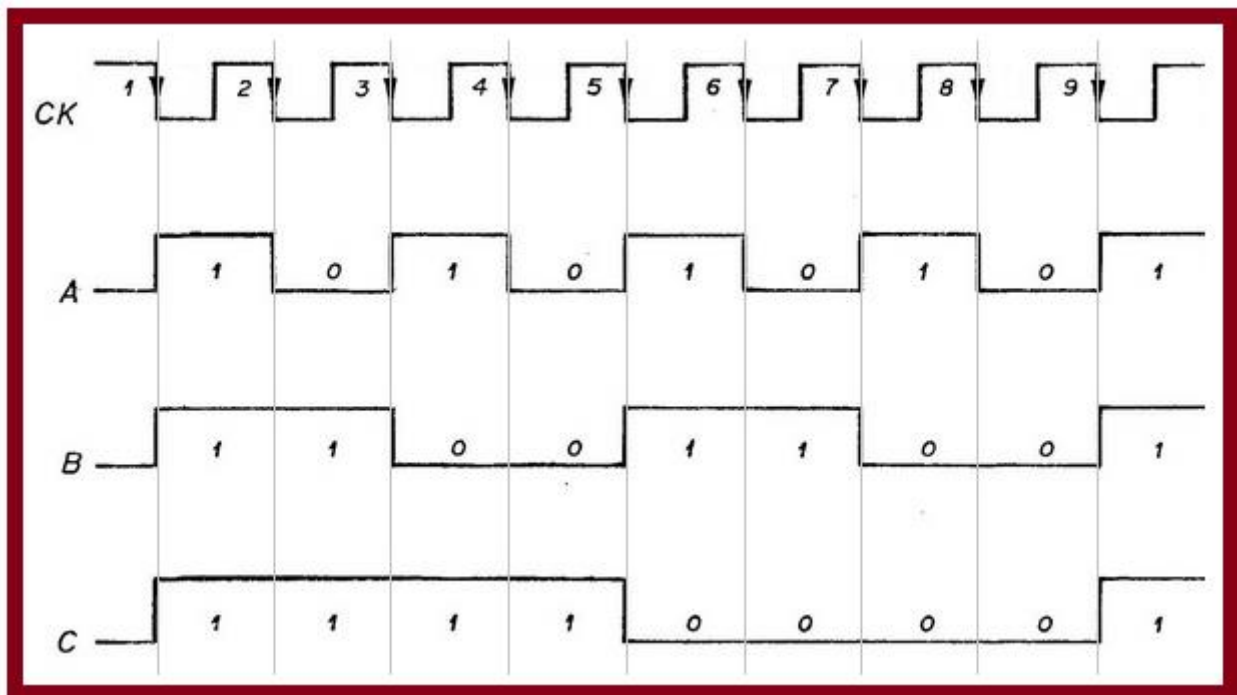
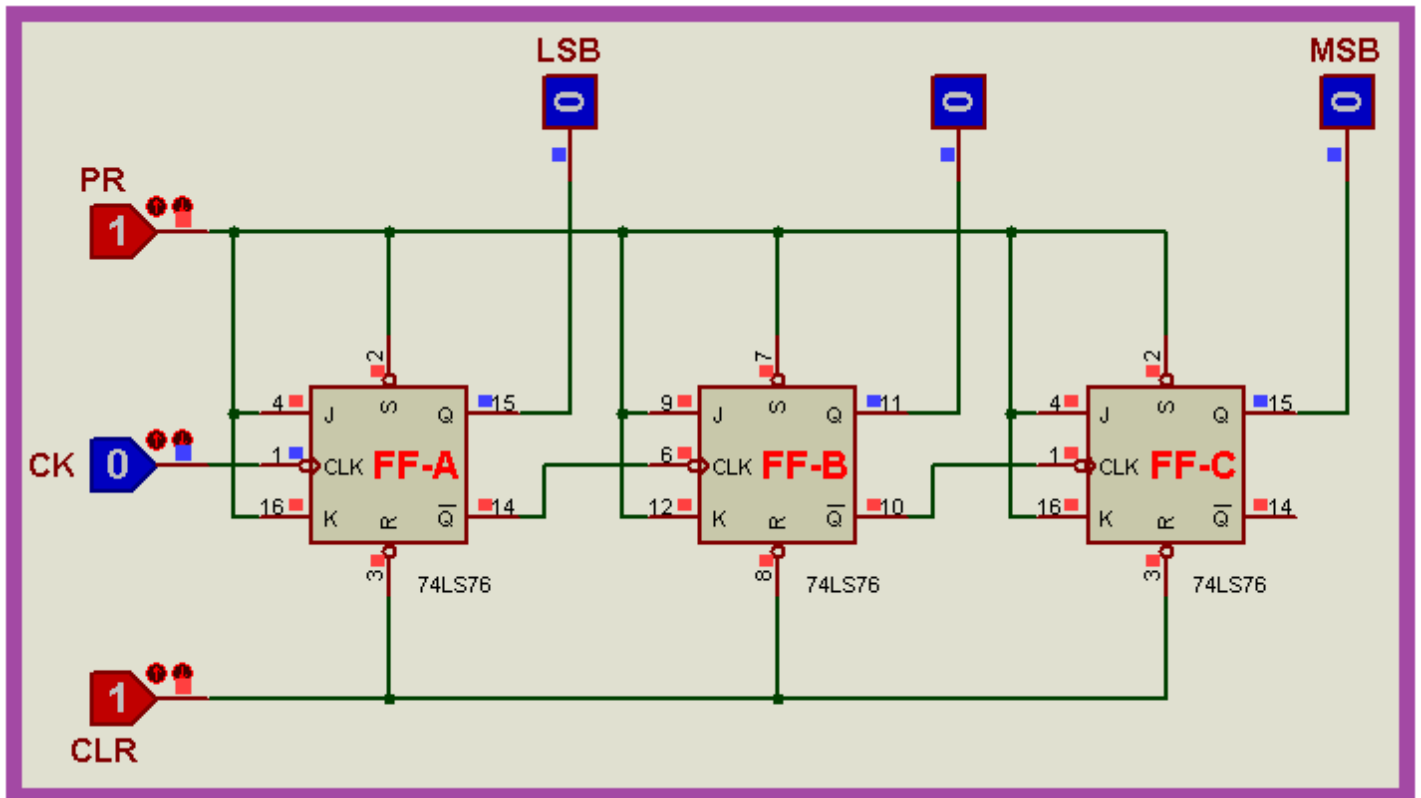
Os contadores que foram mostrados são do tipo **progressivo ou crescente**.

Para construir um contador decrescente ou regressivo, existem 3 formas bastante utilizadas:

1) Inserir no clock do FF seguinte a saída complementada de cada FF.

Veja como fica a tabela e as modificações feitas no circuito:

CK	Saídas dos FFs			Saída binária/decimal
	C	B	A	
ligado	0	0	0	000 / 0
1	0	0	1	111 / 7
2	0	1	0	110 / 6
3	0	1	1	101 / 5
4	1	0	0	100 / 4
5	1	0	1	011 / 3
6	1	1	0	010 / 2
7	1	1	1	001 / 1
8	0	0	0	000 / 0
9	0	0	1	111 / 7

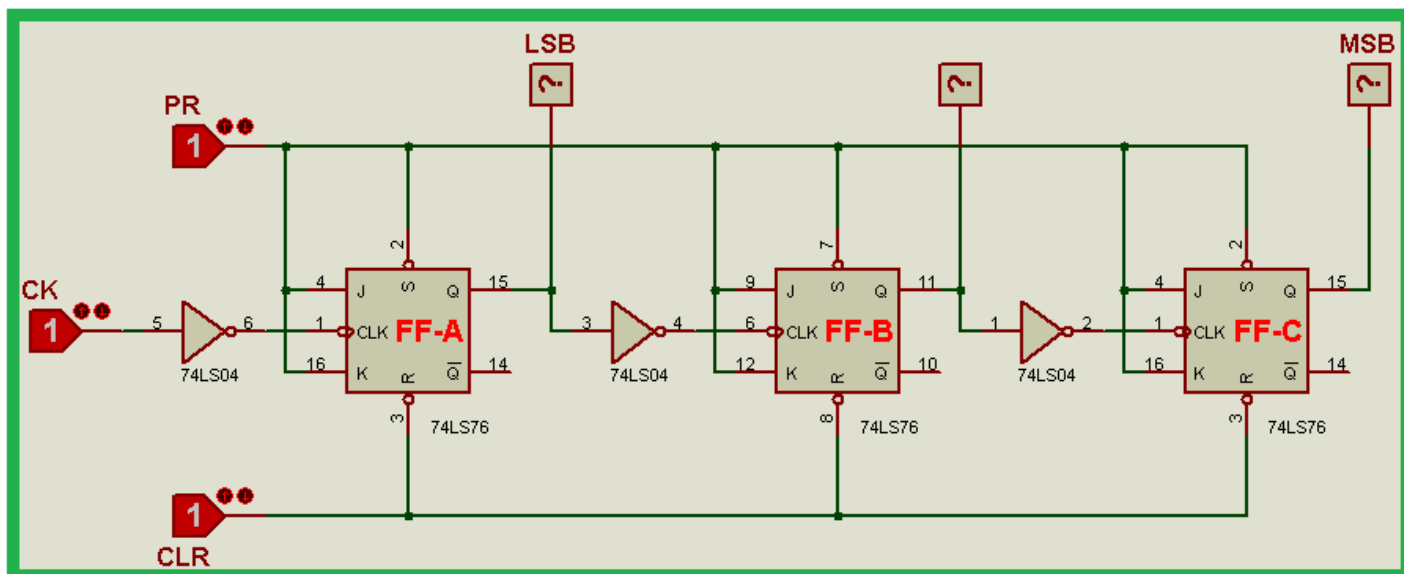


Observe no gráfico acima que o 1º pulso de clock mostra o binário 111 que corresponde ao decimal 7. O 8º pulso de clock zera o contador e a partir do 9º pulso de clock é reiniciada a contagem decrescente.

2) Outra opção para a construção de um contador assíncrono decrescente é a inserção de um inversor em cada entrada do pulso de clock, conforme mostra a figura a seguir.

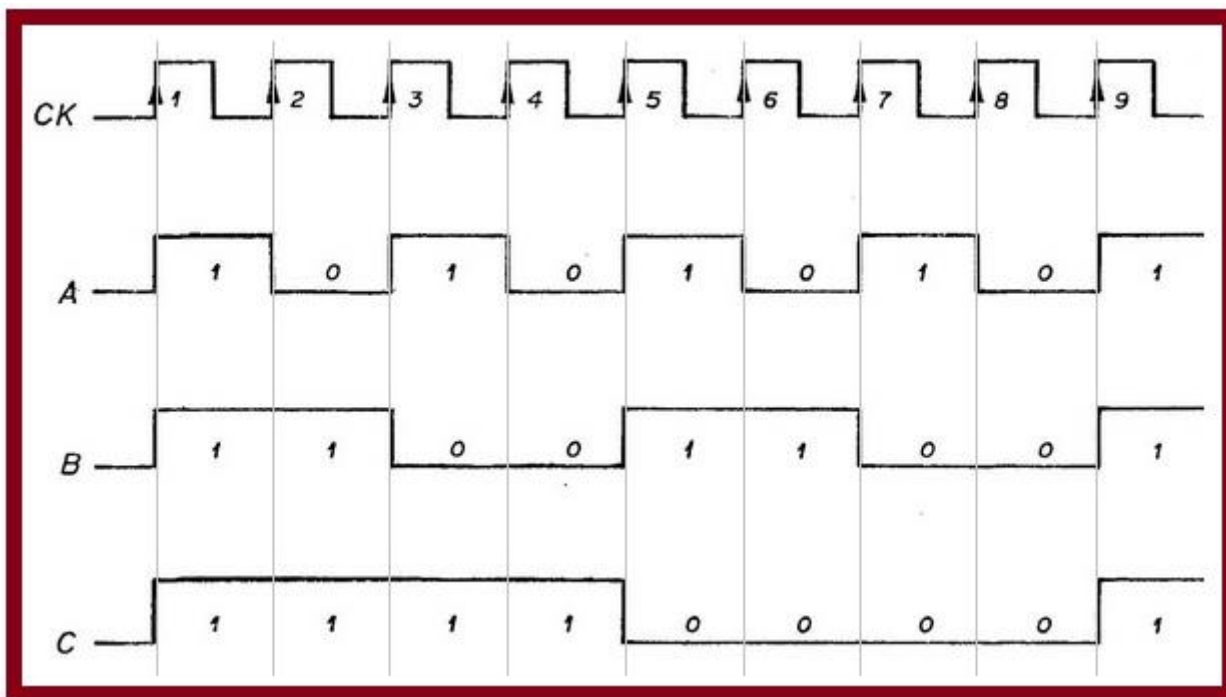
Observe que a única modificação ocorrida foi a inserção de um inversor da saída do FF à entrada de clock do próximo FF, além é claro, da inserção de um inversor na entrada do pulso de clock do FFA.

Neste caso, a ativação de cada FF se dará por transição positiva, ou seja, de nível baixo para nível alto (L-H).



Observe no gráfico a seguir, que representa o desempenho do contador no sentido decrescente que a transição de pulso de clock para ativar o FF é por borda positiva, ou seja, L-H.

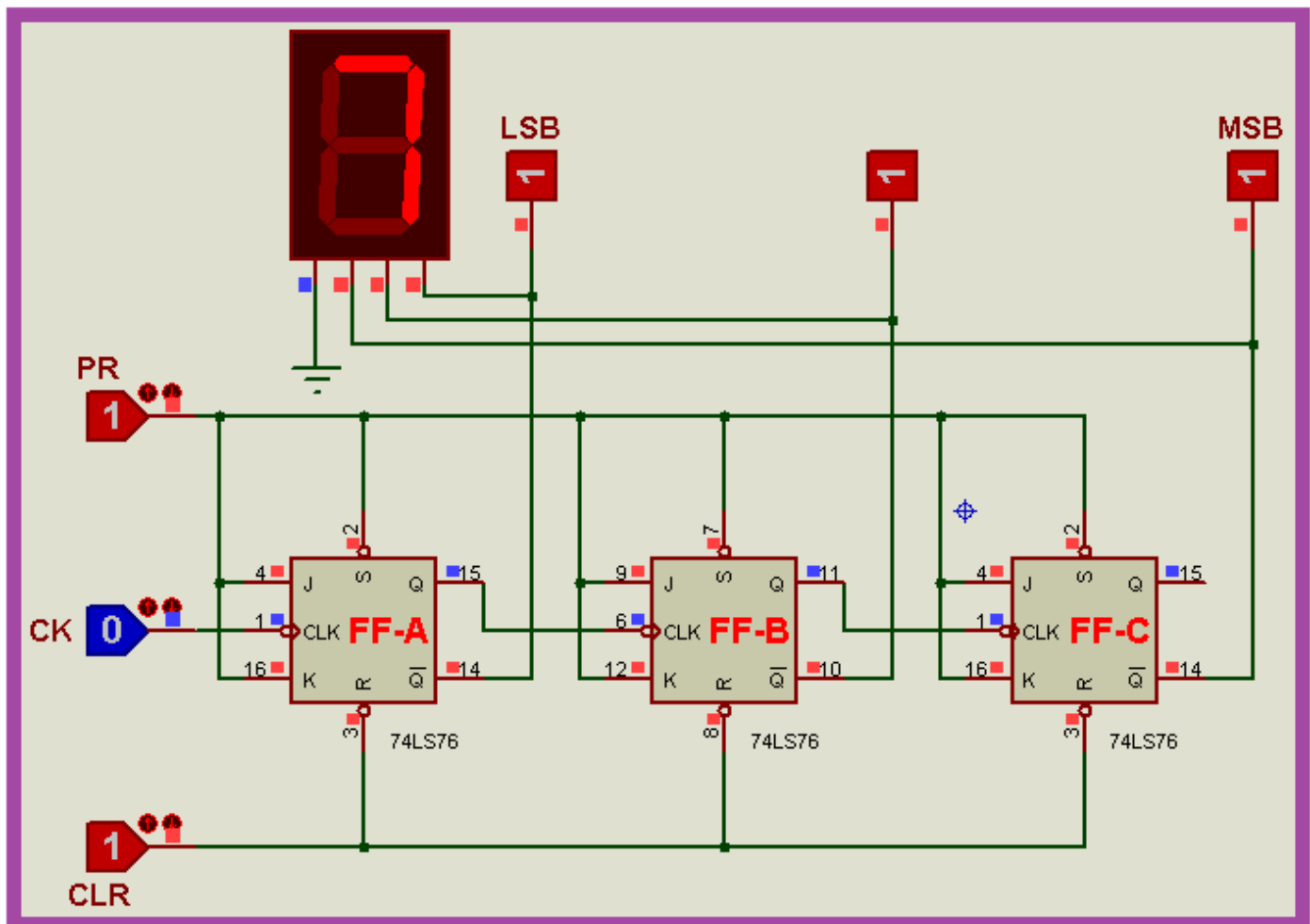
Assim, o 8º pulso de clock zera o contador e a partir do 9º pulso de clock a contagem é reiniciada.



3) Como terceira opção, podemos obter as saídas para decodificação a partir das saídas complementadas de cada FF conforme sugere o layout.

Neste caso, quando o circuito é ligado as saídas serão representadas pelo binário 111 que corresponde ao decimal 7.

Veja a seguir o circuito com o display BCD e o comportamento do mesmo de acordo com a tabela apresentada.



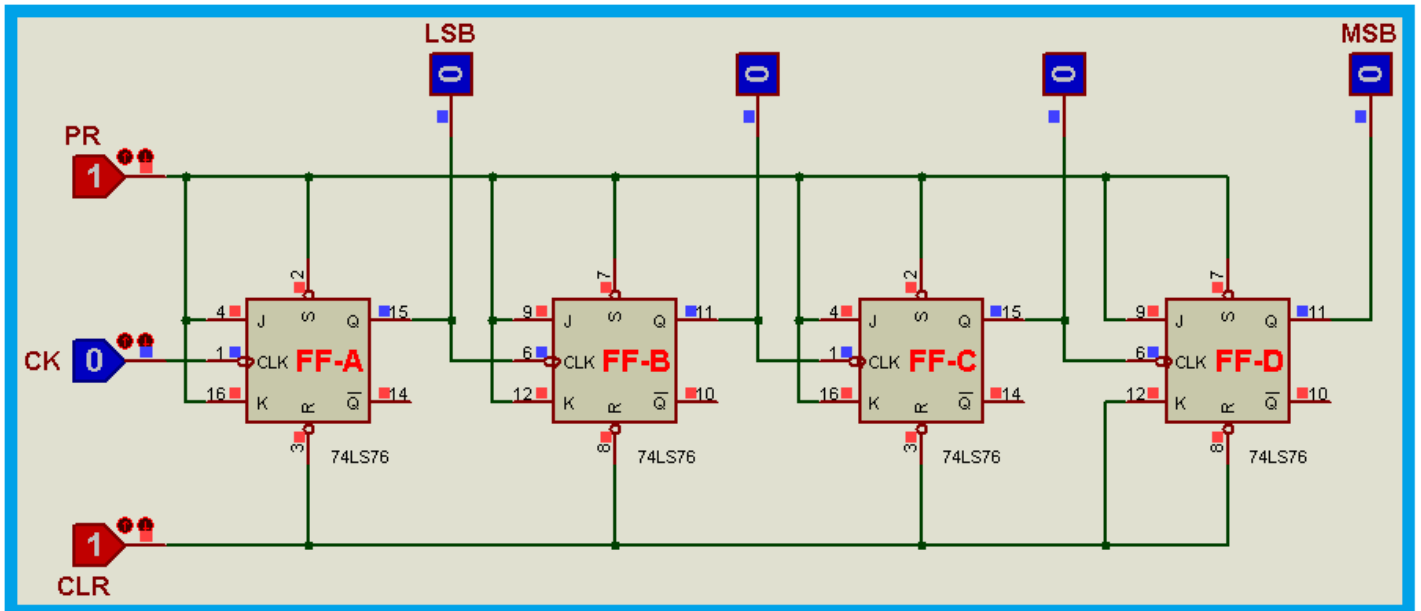
CK	Saídas dos FFs			Saída binária/decimal
	C	B	A	
ligado	1	1	1	111 / 7
1	1	1	0	110 / 6
2	1	0	1	101 / 5
3	1	0	0	100 / 4
4	0	1	1	011 / 3
5	0	1	0	010 / 2
6	0	0	1	001 / 1
7	0	0	0	000 / 0
8	1	1	1	111 / 7
9	1	1	0	110 / 6

CONTADOR ASSÍNCRONO PROGRESSIVO DE 4 BITS

Para construir um contador assíncrono progressivo ou regressivo de 0 até F, ou seja, correspondente decimal de 0 a 15, necessitamos de 4 FFs (4 bits).

Neste caso a contagem se iniciará em 0000 (binário) que corresponde ao decimal 0, até 1111 (binário) que corresponde ao hexadecimal F ou decimal 15.

Assim, o contador será zerado no 15º pulso de clock e a partir do 16º de clock a contagem será reiniciada, conforme mostra a tabela.



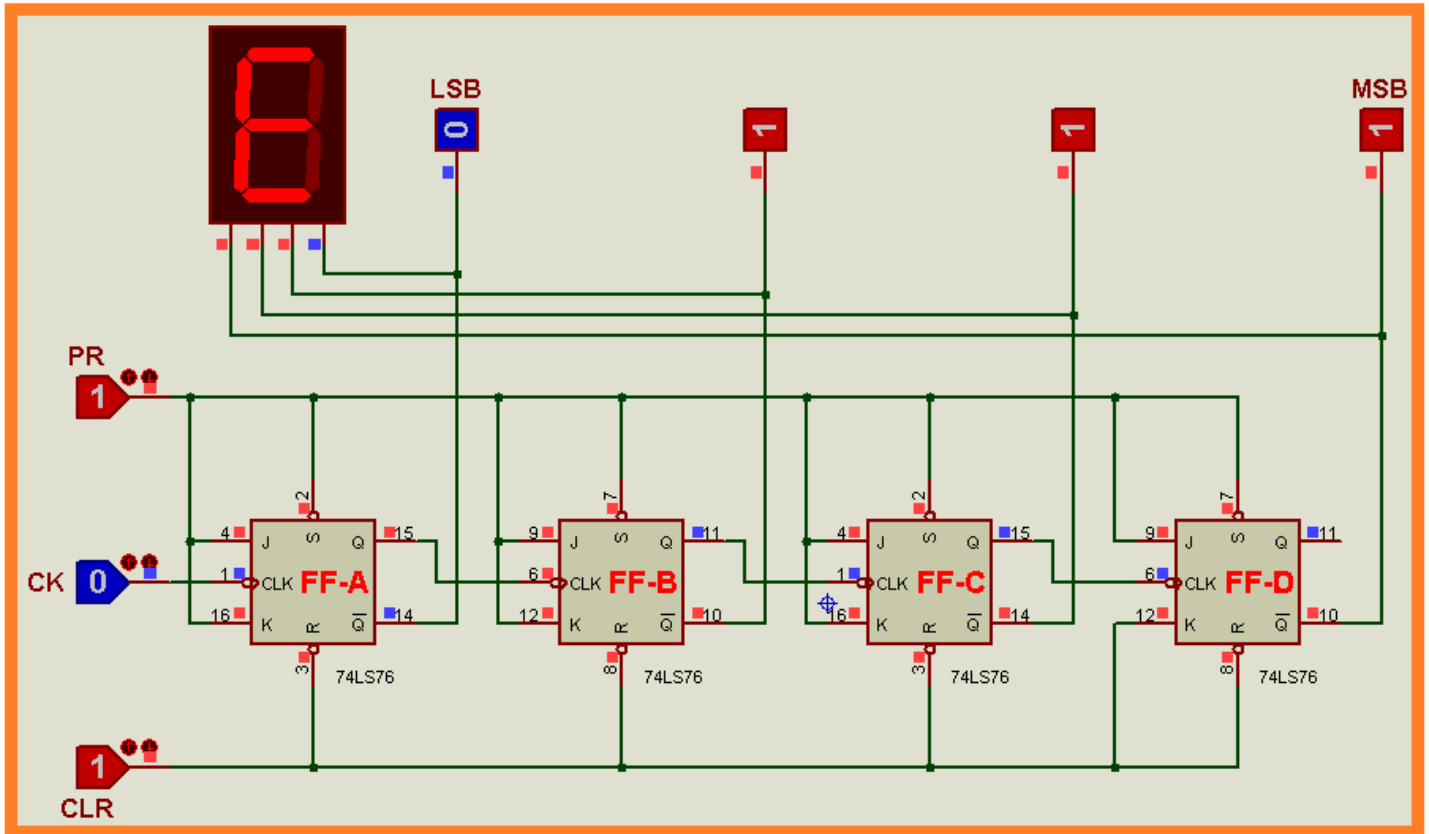
CK	Saídas dos FFs				Saída binária / decimal
	D	C	B	A	
ligado	0	0	0	0	0000 / 0
1	0	0	0	1	0001 / 1
2	0	0	1	0	0010 / 2
3	0	0	1	1	0011 / 3
4	0	1	0	0	0100 / 4
5	0	1	0	1	0101 / 5
6	0	1	1	0	0110 / 6
7	0	1	1	1	0111 / 7
8	1	0	0	0	1000 / 8
9	1	0	0	1	1001 / 9
10	1	0	1	0	1010 / 10 – A
11	1	0	1	1	1011 / 11 – B
12	1	1	0	0	1100 / 12 – C
13	1	1	0	1	1101 / 13 – D
14	1	1	1	0	1110 / 14 – E
15	1	1	1	1	1111 / 15 – F
16	0	0	0	0	0000 / 0

A construção de um contador assíncrono regressivo de 4 bits obedece ao mesmo procedimento adotado para os contadores de 3 bits.

Como exemplo, a figura a seguir ilustra o projeto de um contador assíncrono regressivo de 4 bits, onde se observa que as saídas complementares de cada FF indicarão a contagem de F até 0.

Observe que na tabela mostrada que corresponde a essa montagem, quando o circuito é ligado as saídas estarão em 1111 que representa o decimal 15 ou o hexadecimal F.

A partir do 15º pulso o contador é zerado, recomeçando a contagem regressiva a partir do 16º pulso.



CK	Saídas dos FFs				Saída binária / decimal
	D	C	B	A	
ligado	1	1	1	1	1111 / 15 - F
1	1	1	1	0	1110 / 14 - E
2	1	1	0	1	1101 / 13 - D
3	1	1	0	0	1100 / 12 - C
4	1	0	1	1	1011 / 11 - B
5	1	0	1	0	1010 / 10 - A
6	1	0	0	1	1001 / 9
7	1	0	0	0	1000 / 8
8	0	1	1	1	0111 / 7
9	0	1	1	0	0110 / 6
10	0	1	0	1	0101 / 5
11	0	1	0	0	0100 / 4
12	0	0	1	1	0011 / 3
13	0	0	1	0	0010 / 2
14	0	0	0	1	0001 / 1
15	0	0	0	0	0000 / 0
16	1	1	1	1	1111 / 15 - F