

論理回路入門 (13)

同期式N進カウンターの設計

李 亜民

2022年12月20日(火)

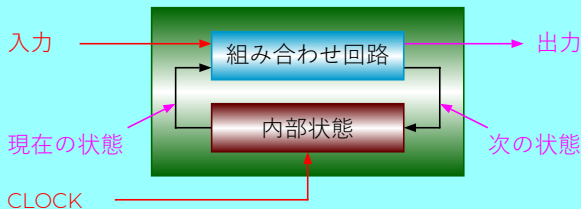
同期式N進カウンターの設計

ポイント

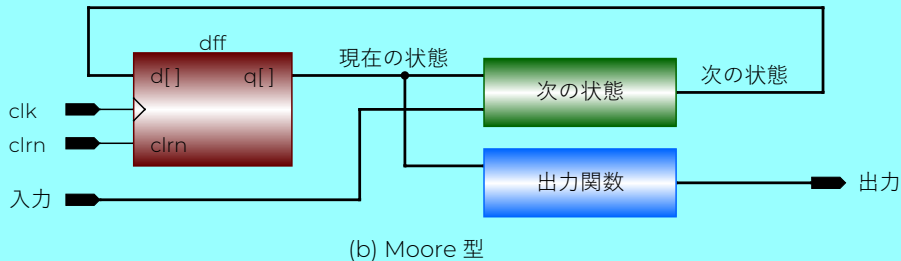
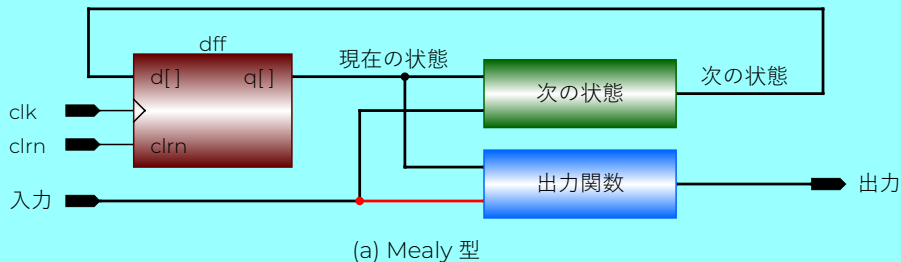
- 同期式N進カウンター
 - ▶ DFF で同期式N進カウンター
 - ▶ JKFF で同期式N進カウンター
 - ▶ TFF で同期式N進カウンター
- 非同期式カウンター
- 非同期式 FIFO

順序回路

- 順序回路は内部状態と入力信号で出力が決まる回路である。
- 順序回路の構成:
 - ① 記憶素子 (D フリップフロップなど)
 - ★ 現在の状態を記憶しておくため
 - ② 組み合わせ回路
 - ★ 状態遷移関数 (次の状態)
 - ★ 出力関数



順序回路 — Mealy 型と Moore 型



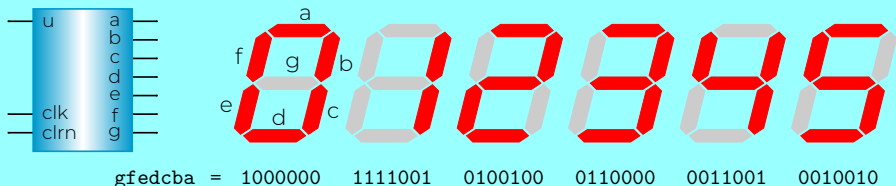
順序回路設計の手順

- 1 問題を理解する。
- 2 状態遷移図をつくる。
- 3 FF の数を決める ($n = \lceil \log_2 N \rceil$ 、 N は状態の数)。
- 4 各状態に n ビットの番号を付け、真理値表をつくる。
 - ▶ 次の状態の真理値表をつくる。
 - ▶ 出力関数の真理値表をつくる。
- 5 真理値表から論理式をつくる (どの条件で出力が 1 になるか)。
 - ▶ カルノー図を用いて論理式を簡単化する。
- 6 論理式から回路をつくる。
- 7 テストベンチをつくる (シミュレーションするため)。
- 8 回路をシミュレーションする (回路設計の正当性検証)。
- 9 与えられた回路の動作を理解する (波形の説明)。

同期式 N 進カウンターの設計

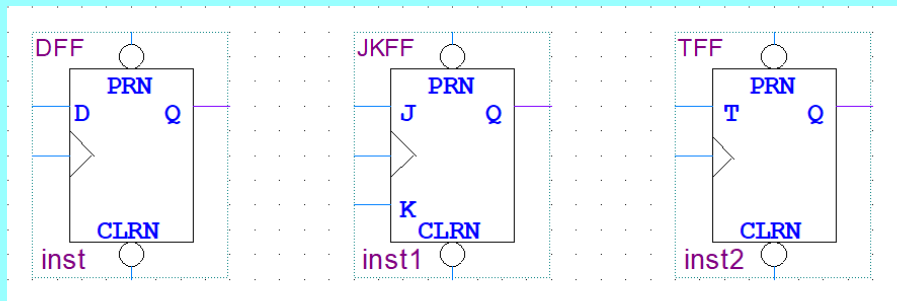
問題: 7セグメント LED を使いカウントアップ / カウントダウンする回路を作成しよう。

- カウント範囲: 0 ~ 5
- $u = 1$ (カウントアップ): 0, 1, 2, 3, 4, 5, 0, 1, 2, 3, ...
- $u = 0$ (カウントダウン): 0, 5, 4, 3, 2, 1, 0, 5, 4, 3, ...
- 7セグメント LED で表示する (0: 点灯; 1: 消灯)



DFF で同期式 N 進 カウンターの設計

フリップフロップ



DFF : CLK の立ち上がりで、 $Q_{\text{next}} = D$

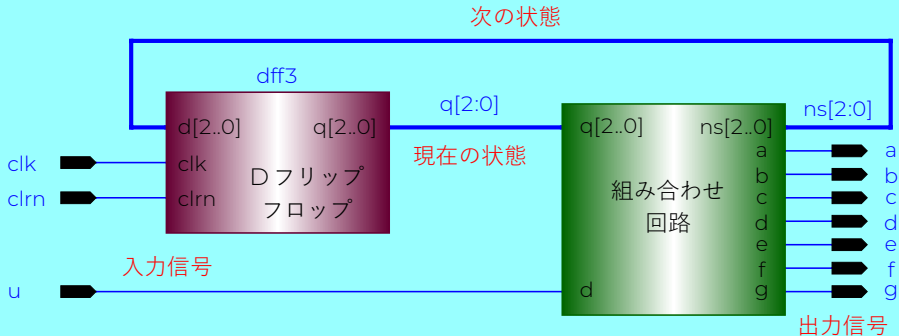
JKFF : CLK の立ち上がりで、 $Q_{\text{next}} = J \bar{Q} + \bar{K} Q$

TFF : CLK の立ち上がりで、 $Q_{\text{next}} = T \bar{Q} + \bar{T} Q$

DFF で同期式 N 進カウンターの設計

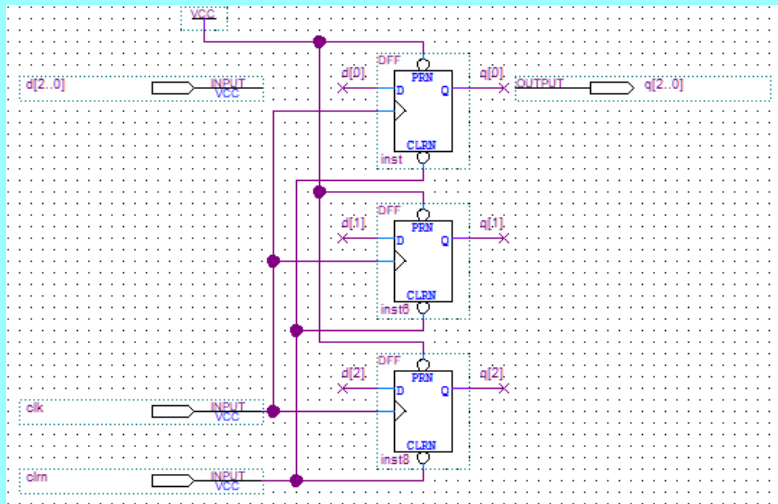
0 ~ 5 カウンター:

- 状態数: 6 (0, 1, 2, 3, 4, 5)
- 3 ビット DFF を使う ($\lceil \log_2 6 \rceil = 3$)
- カウント全体の回路:



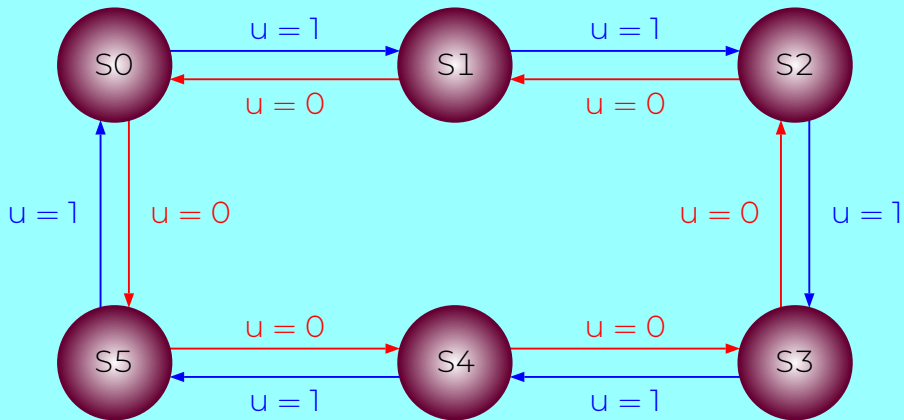
DFF で同期式N進カウンターの設計

0～5 カウンター: 3 ビット DFF 回路:



DFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 状態遷移図:



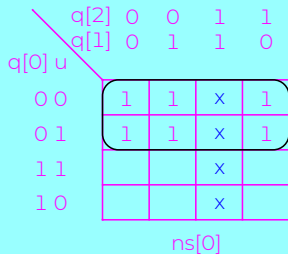
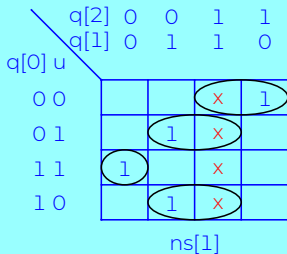
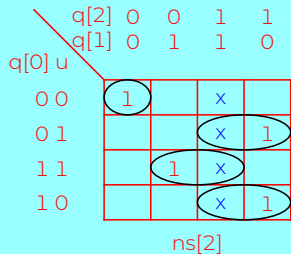
DFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 状態遷移表:

	現在の状態			入力		次の状態		
	q[2]	q[1]	q[0]			u	ns[2]	ns[1]
S0	0	0	0	1	S1	0	0	1
				0	S5	1	0	1
S1	0	0	1	1	S2	0	1	0
				0	S0	0	0	0
S2	0	1	0	1	S3	0	1	1
				0	S1	0	0	1
S3	0	1	1	1	S4	1	0	0
				0	S2	0	1	0
S4	1	0	0	1	S5	1	0	1
				0	S3	0	1	1
S5	1	0	1	1	S0	0	0	0
				0	S4	1	0	0
入力として	1	1	0	X	Don't care	X	X	X
存在しない	1	1	1	X	ドントケア	X	X	X

DFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 次の状態:



DFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 次の状態:

		q[2]			
		0	0	1	1
q[0] u		q[1]			
		0	1	1	0
00	1		x		
01			x	1	
11		1	x		
10			x	1	

ns[2]

		q[2]			
		0	0	1	1
q[0] u		q[1]			
		0	1	1	0
00			x	1	
01		1	x		
11	1		x		
10		1	x		

ns[1]

		q[2]			
		0	0	1	1
q[0] u		q[1]			
		0	1	1	0
00	1	1	x	1	
01	1	1	x	1	
11			x		
10			x		

ns[0]

$$ns[2] = \overline{q[2]} \overline{q[1]} \overline{q[0]} \bar{u} + q[1] q[0] u + q[2] \overline{q[0]} u + q[2] q[0] \bar{u}$$

DFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 次の状態:

q[2]	0	0	1	1
q[1]	0	1	1	0
q[0] u				
00	1		x	
01			x	1
11		1	x	
10			x	1

ns[2]

q[2]	0	0	1	1
q[1]	0	1	1	0
q[0] u				
00			x	1
01		1	x	
11	1		x	
10		1	x	

ns[1]

q[2]	0	0	1	1
q[1]	0	1	1	0
q[0] u				
00	1	1	x	1
01	1	1	x	1
11			x	
10			x	

ns[0]

$$ns[2] = \overline{q[2]} \overline{q[1]} \overline{q[0]} \overline{u} + q[1] q[0] u + q[2] \overline{q[0]} u + q[2] q[0] \overline{u}$$

$$ns[1] = \overline{q[2]} \overline{q[1]} q[0] u + q[1] \overline{q[0]} u + q[1] q[0] \overline{u} + q[2] \overline{q[0]} \overline{u}$$

DFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 次の状態:

q[2]	0	0	1	1
q[1]	0	1	1	0
q[0] u				
00	1		x	
01			x	1
11		1	x	
10			x	1

ns[2]

q[2]	0	0	1	1
q[1]	0	1	1	0
q[0] u				
00			x	1
01		1	x	
11	1		x	
10		1	x	

ns[1]

q[2]	0	0	1	1
q[1]	0	1	1	0
q[0] u				
00	1	1	x	1
01	1	1	x	1
11			x	
10			x	

ns[0]

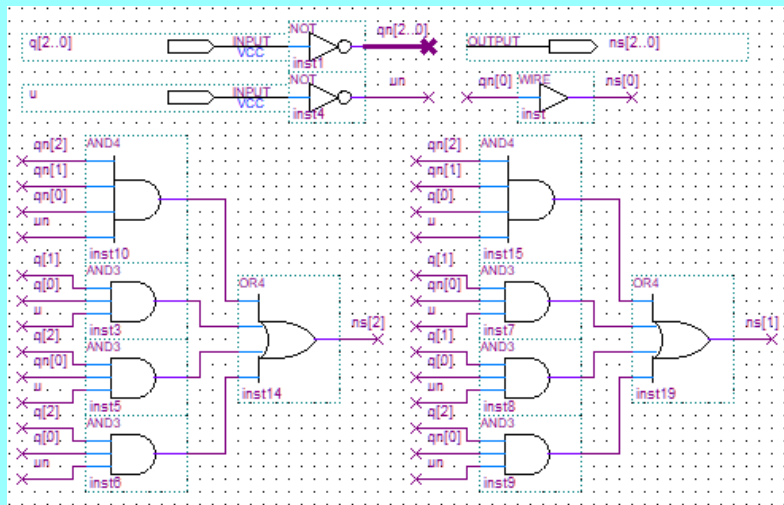
$$ns[2] = \overline{q[2]} \overline{q[1]} \overline{q[0]} \overline{u} + q[1] q[0] u + q[2] \overline{q[0]} u + q[2] q[0] \overline{u}$$

$$ns[1] = \overline{q[2]} \overline{q[1]} q[0] u + q[1] \overline{q[0]} u + q[1] q[0] \overline{u} + q[2] \overline{q[0]} \overline{u}$$

$$ns[0] = \overline{q[0]}$$

DFF で同期式N進カウンターの設計

0 ~ 5 カウンター: 次の状態回路:

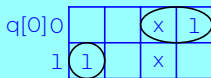


DFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 出力 (0: 点灯; 1: 消灯):

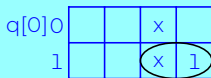
q[2:0]	g	f	e	d	c	b	a
0 0 0	1	0	0	0	0	0	0
0 0 1	1	1	1	1	0	0	1
0 1 0	0	1	0	0	1	0	0
0 1 1	0	1	1	0	0	0	0
1 0 0	0	0	1	1	0	0	1
1 0 1	0	0	1	0	0	1	0

q[2] 0 0 1 1
q[1] 0 1 1 0



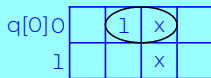
a

q[2] 0 0 1 1
q[1] 0 1 1 0



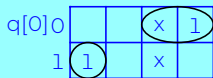
b

q[2] 0 0 1 1
q[1] 0 1 1 0



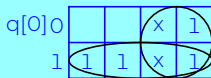
c

q[2] 0 0 1 1
q[1] 0 1 1 0



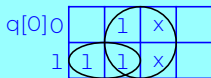
d

q[2] 0 0 1 1
q[1] 0 1 1 0



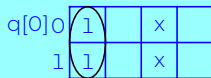
e

q[2] 0 0 1 1
q[1] 0 1 1 0



f

q[2] 0 0 1 1
q[1] 0 1 1 0



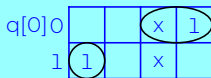
g

DFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 出力 (0: 点灯; 1: 消灯):

q[2:0]	g	f	e	d	c	b	a
0 0 0	1	0	0	0	0	0	0
0 0 1	1	1	1	1	0	0	1
0 1 0	0	1	0	0	1	0	0
0 1 1	0	1	1	0	0	0	0
1 0 0	0	0	1	1	0	0	1
1 0 1	0	0	1	0	0	1	0

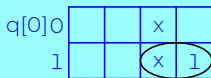
q[2]	0	0	1	1
q[1]	0	1	1	0



a

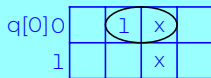
$$a = \overline{q[2]} \overline{q[1]} q[0] + q[2] \overline{q[0]}$$

q[2]	0	0	1	1
q[1]	0	1	1	0



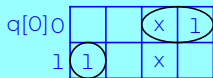
b

q[2]	0	0	1	1
q[1]	0	1	1	0



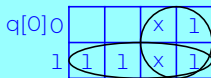
c

q[2]	0	0	1	1
q[1]	0	1	1	0



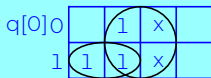
d

q[2]	0	0	1	1
q[1]	0	1	1	0



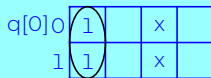
e

q[2]	0	0	1	1
q[1]	0	1	1	0



f

q[2]	0	0	1	1
q[1]	0	1	1	0



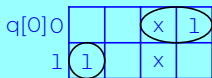
g

DFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 出力 (0: 点灯; 1: 消灯):

q[2:0]	g	f	e	d	c	b	a
0 0 0	1	0	0	0	0	0	0
0 0 1	1	1	1	1	0	0	1
0 1 0	0	1	0	0	1	0	0
0 1 1	0	1	1	0	0	0	0
1 0 0	0	0	1	1	0	0	1
1 0 1	0	0	1	0	0	1	0

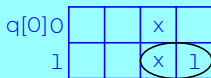
q[2]	0	0	1	1
q[1]	0	1	1	0



a

$$a = \overline{q[2]} \overline{q[1]} q[0] + q[2] \overline{q[0]}$$

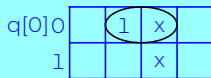
q[2]	0	0	1	1
q[1]	0	1	1	0



b

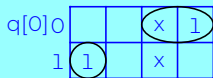
$$b = q[2] q[0]$$

q[2]	0	0	1	1
q[1]	0	1	1	0



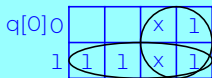
c

q[2]	0	0	1	1
q[1]	0	1	1	0



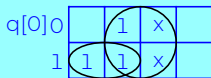
d

q[2]	0	0	1	1
q[1]	0	1	1	0



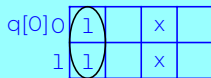
e

q[2]	0	0	1	1
q[1]	0	1	1	0



f

q[2]	0	0	1	1
q[1]	0	1	1	0



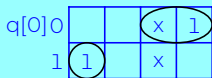
g

DFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 出力 (0: 点灯; 1: 消灯):

q[2:0]	g	f	e	d	c	b	a
0 0 0	1	0	0	0	0	0	0
0 0 1	1	1	1	1	0	0	1
0 1 0	0	1	0	0	1	0	0
0 1 1	0	1	1	0	0	0	0
1 0 0	0	0	1	1	0	0	1
1 0 1	0	0	1	0	0	1	0

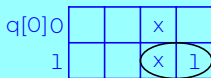
q[2]	0	0	1	1
q[1]	0	1	1	0



a

$$a = \overline{q[2]} \overline{q[1]} q[0] + q[2] \overline{q[0]}$$

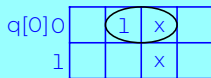
q[2]	0	0	1	1
q[1]	0	1	1	0



b

$$b = q[2] q[0]$$

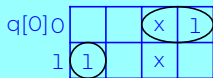
q[2]	0	0	1	1
q[1]	0	1	1	0



c

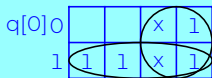
$$c = q[1] \overline{q[0]}$$

q[2]	0	0	1	1
q[1]	0	1	1	0



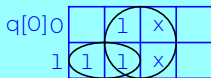
d

q[2]	0	0	1	1
q[1]	0	1	1	0



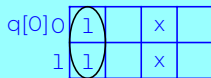
e

q[2]	0	0	1	1
q[1]	0	1	1	0



f

q[2]	0	0	1	1
q[1]	0	1	1	0



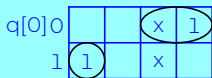
g

DFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 出力 (0: 点灯; 1: 消灯):

q[2:0]	g	f	e	d	c	b	a
0 0 0	1	0	0	0	0	0	0
0 0 1	1	1	1	1	0	0	1
0 1 0	0	1	0	0	1	0	0
0 1 1	0	1	1	0	0	0	0
1 0 0	0	0	1	1	0	0	1
1 0 1	0	0	1	0	0	1	0

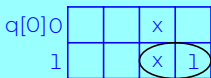
q[2] 0 0 1 1
q[1] 0 1 1 0



a

$$a = \overline{q[2]} \overline{q[1]} q[0] + q[2] \overline{q[0]}$$

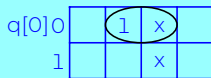
q[2] 0 0 1 1
q[1] 0 1 1 0



b

$$b = q[2] q[0]$$

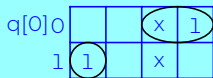
q[2] 0 0 1 1
q[1] 0 1 1 0



c

$$c = q[1] \overline{q[0]}$$

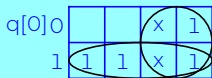
q[2] 0 0 1 1
q[1] 0 1 1 0



d

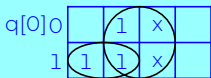
$$d = a$$

q[2] 0 0 1 1
q[1] 0 1 1 0



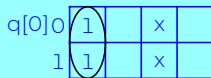
e

q[2] 0 0 1 1
q[1] 0 1 1 0



f

q[2] 0 0 1 1
q[1] 0 1 1 0



g

DFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 出力 (0: 点灯; 1: 消灯):

q[2:0]	g	f	e	d	c	b	a
0 0 0	1	0	0	0	0	0	0
0 0 1	1	1	1	1	0	0	1
0 1 0	0	1	0	0	1	0	0
0 1 1	0	1	1	0	0	0	0
1 0 0	0	0	1	1	0	0	1
1 0 1	0	0	1	0	0	1	0

q[2]	0	0	1	1
q[1]	0	1	1	0

q[0]	0	1	0	1
0			x	1
1	1		x	

a

$$a = \overline{q[2]} \overline{q[1]} q[0] + q[2] \overline{q[0]}$$

q[2]	0	0	1	1
q[1]	0	1	1	0

q[0]	0	1	0	1
0			x	
1			x	1

b

$$b = q[2] q[0]$$

q[2]	0	0	1	1
q[1]	0	1	1	0

q[0]	0	1	0	1
0		1	x	
1			x	

c

$$c = q[1] \overline{q[0]}$$

q[2]	0	0	1	1
q[1]	0	1	1	0

q[0]	0	1	0	1
0			x	1
1	1		x	

d

$$d = a$$

q[2]	0	0	1	1
q[1]	0	1	1	0

q[0]	0	1	0	1
0			x	1
1	1	1	x	1

e

$$e = q[2] + q[0]$$

q[2]	0	0	1	1
q[1]	0	1	1	0

q[0]	0	1	0	1
0		1	x	
1	1	1	x	

f

q[2]	0	0	1	1
q[1]	0	1	1	0

q[0]	0	1	0	1
0	1		x	
1	1		x	

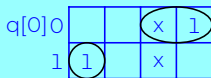
g

DFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 出力 (0: 点灯; 1: 消灯):

q[2:0]	g	f	e	d	c	b	a
0 0 0	1	0	0	0	0	0	0
0 0 1	1	1	1	1	0	0	1
0 1 0	0	1	0	0	1	0	0
0 1 1	0	1	1	0	0	0	0
1 0 0	0	0	1	1	0	0	1
1 0 1	0	0	1	0	0	1	0

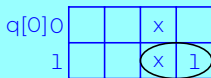
q[2] 0 0 1 1
q[1] 0 1 1 0



a

$$a = \overline{q[2]} \overline{q[1]} q[0] + q[2] \overline{q[0]}$$

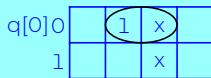
q[2] 0 0 1 1
q[1] 0 1 1 0



b

$$b = q[2] q[0]$$

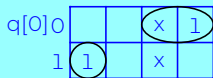
q[2] 0 0 1 1
q[1] 0 1 1 0



c

$$c = q[1] \overline{q[0]}$$

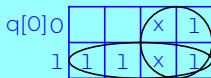
q[2] 0 0 1 1
q[1] 0 1 1 0



d

$$d = a$$

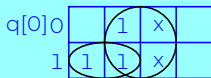
q[2] 0 0 1 1
q[1] 0 1 1 0



e

$$e = q[2] + q[0]$$

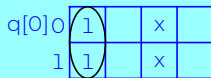
q[2] 0 0 1 1
q[1] 0 1 1 0



f

$$f = \overline{q[2]} q[0] + q[1]$$

q[2] 0 0 1 1
q[1] 0 1 1 0



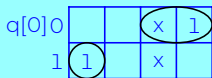
g

DFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 出力 (0: 点灯; 1: 消灯):

q[2:0]	g	f	e	d	c	b	a
0 0 0	1	0	0	0	0	0	0
0 0 1	1	1	1	1	0	0	1
0 1 0	0	1	0	0	1	0	0
0 1 1	0	1	1	0	0	0	0
1 0 0	0	0	1	1	0	0	1
1 0 1	0	0	1	0	0	1	0

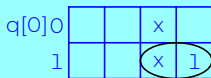
q[2] 0 0 1 1
q[1] 0 1 1 0



a

$$a = \overline{q[2]} \overline{q[1]} q[0] + q[2] \overline{q[0]}$$

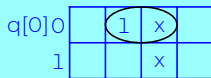
q[2] 0 0 1 1
q[1] 0 1 1 0



b

$$b = q[2] q[0]$$

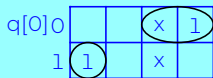
q[2] 0 0 1 1
q[1] 0 1 1 0



c

$$c = q[1] \overline{q[0]}$$

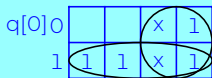
q[2] 0 0 1 1
q[1] 0 1 1 0



d

$$d = a$$

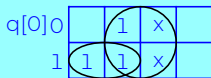
q[2] 0 0 1 1
q[1] 0 1 1 0



e

$$e = q[2] + q[0]$$

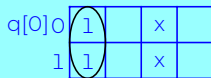
q[2] 0 0 1 1
q[1] 0 1 1 0



f

$$f = \overline{q[2]} q[0] + q[1]$$

q[2] 0 0 1 1
q[1] 0 1 1 0

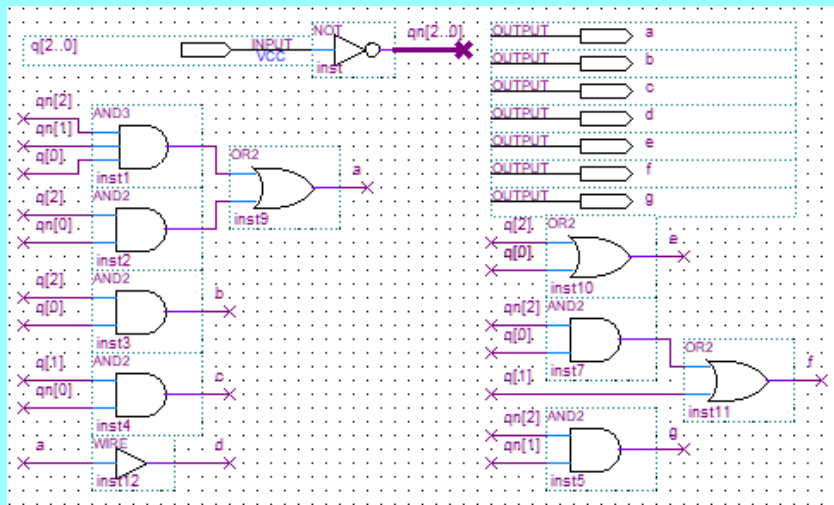


g

$$g = \overline{q[2]} \overline{q[1]}$$

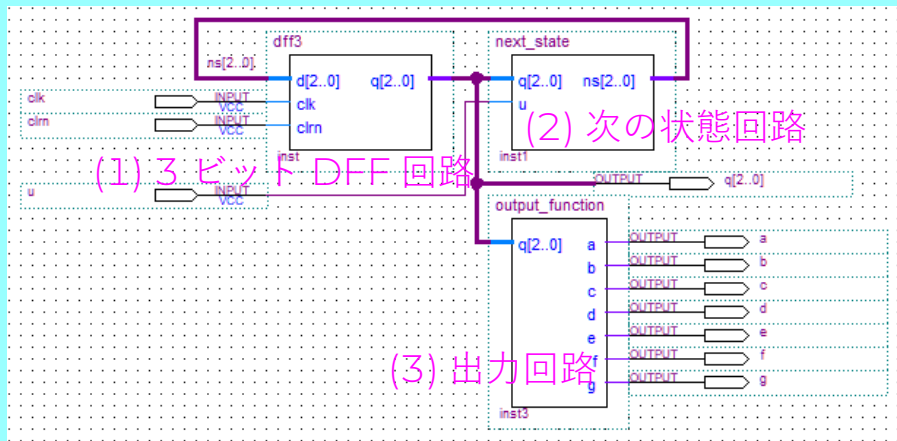
DFF で同期式N進カウンターの設計

0～5 カウンター: 出力回路:



DFF で同期式N進カウンターの設計

0 ~ 5 カウンター: 全体の回路:

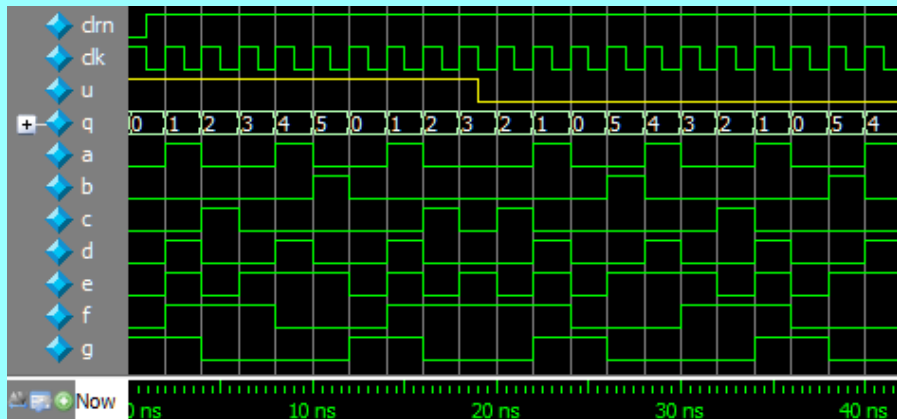


DFF で同期式 N 進カウンターの設計

```
'timescale 1ns/1ps // unit = 1 ns; accuracy = 1 ps
module counter6_tb;
  reg      u,clk,clrn;
  wire     a,b,c,d,e,f,g;
  wire [2:0] q;
  counter6 i1 (
    .u(u),
    .clk(clk),
    .clrn(clrn),
    .a(a),.b(b),.c(c),.d(d),.e(e),.f(f),.g(g),
    .q(q)
  );
  initial begin
    #0  clk = 1; clrn = 0; u = 1;
    #1  clrn = 1;
    #18 u = 0;
    #25 $stop;
  end
  always #1 clk = ~clk;
endmodule
```

DFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: シミュレーション波形:



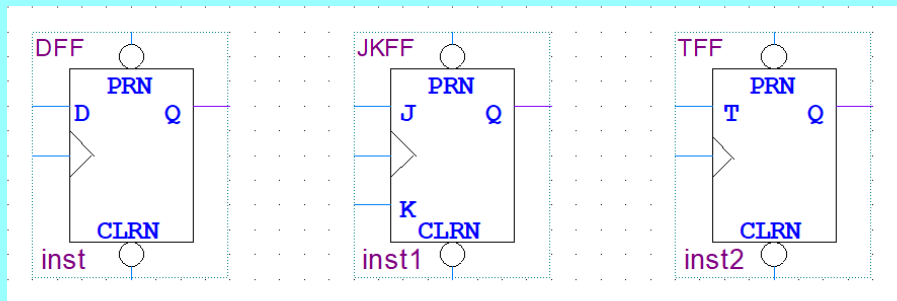
[counter6_tb.v](#)

Count Up

Count Down

J K F F で同期式 N 進 カウンターの設計

フリップフロップ



DFF : CLK の立ち上がりで、 $Q_{\text{next}} = D$

JKFF : CLK の立ち上がりで、 $Q_{\text{next}} = J\bar{Q} + \bar{K}Q$

TFF : CLK の立ち上がりで、 $Q_{\text{next}} = T\bar{Q} + \bar{T}Q$

JKFF で同期式 N 進カウンターの設計

- JKFF: CLK の立ち上がりで、 $Q_{\text{next}} = J \bar{Q} + \bar{K} Q \dots \dots \dots (= ns)$

例 : $ns = A \bar{q} + B q + C$

$$= A \bar{q} + B q + C (\bar{q} + q)$$

$$= A \bar{q} + B q + C \bar{q} + C q$$

$$= (A + C) \bar{q} + (B + C) q$$

$$= j \bar{q} + \bar{k} q$$

▶ $j = A + C$

▶ $k = \overline{B + C}$

JKFF で同期式 N 進カウンターの設計

- JKFF: CLK の立ち上がりで、 $Q_{\text{next}} = J\bar{Q} + \bar{K}Q \dots \dots \dots (= ns)$

JKFF で同期式 N 進カウンターの設計

- JKFF: CLK の立ち上がりで、 $Q_{\text{next}} = J \bar{Q} + \bar{K} Q \dots \dots \dots (= ns)$
- $ns[0] = \bar{q}[0] = 1 \cdot \bar{q}[0] + 0 \cdot q[0] = j[0] \cdot \bar{q}[0] + \bar{k}[0] \cdot q[0]$
 - ▶ $j[0] = 1 \dots \dots \dots j[0] \bar{q}[0]$
 - ▶ $k[0] = \bar{0} = 1 \dots \dots \dots \bar{k}[0] q[0]$

JKFF で同期式 N 進カウンターの設計

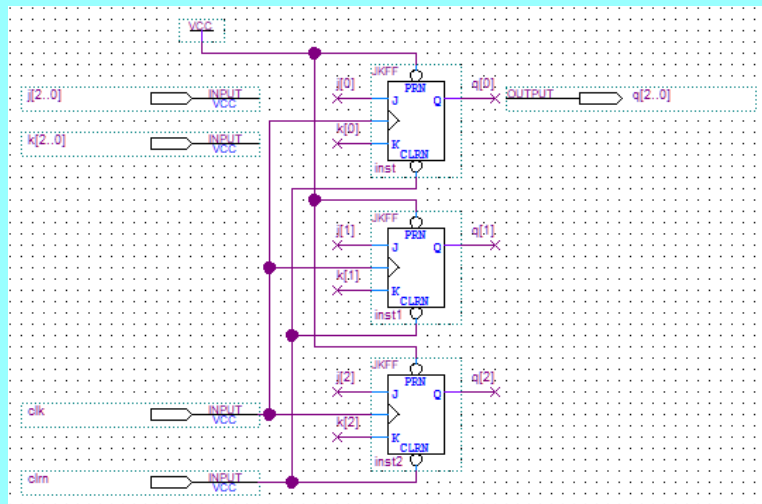
- JKFF: CLK の立ち上がりで、 $Q_{\text{next}} = J \bar{Q} + \bar{K} Q \dots \dots \dots (= ns)$
- $ns[0] = \bar{q}[0] = 1 \cdot \bar{q}[0] + 0 \cdot q[0] = j[0] \cdot \bar{q}[0] + \bar{k}[0] \cdot q[0]$
 - ▶ $j[0] = 1 \dots \dots \dots j[0] \bar{q}[0]$
 - ▶ $k[0] = \bar{0} = 1 \dots \dots \dots \bar{k}[0] q[0]$
- $ns[1] = \bar{q}[2] \bar{q}[1] q[0] u + q[1] \bar{q}[0] u + q[1] q[0] \bar{u} + q[2] \bar{q}[0] \bar{u}$
 - ▶ $j[1] = \bar{q}[2] q[0] u + q[2] \bar{q}[0] \bar{u} \dots \dots \dots j[1] \bar{q}[1]$
 - ▶ $k[1] = \overline{\bar{q}[0] u + q[0] \bar{u} + q[2] \bar{q}[0] \bar{u}} \dots \dots \dots \bar{k}[1] q[1]$

JKFF で同期式 N 進カウンターの設計

- JKFF: CLK の立ち上がりで、 $Q_{\text{next}} = J \bar{Q} + \bar{K} Q \dots \dots \dots (= ns)$
- $ns[0] = \bar{q}[0] = 1 \cdot \bar{q}[0] + 0 \cdot q[0] = j[0] \cdot \bar{q}[0] + \bar{k}[0] \cdot q[0]$
 - ▶ $j[0] = 1 \dots \dots \dots j[0] \bar{q}[0]$
 - ▶ $k[0] = \bar{0} = 1 \dots \dots \dots \bar{k}[0] q[0]$
- $ns[1] = \bar{q}[2] \bar{q}[1] q[0] u + q[1] \bar{q}[0] u + q[1] q[0] \bar{u} + q[2] \bar{q}[0] \bar{u}$
 - ▶ $j[1] = \bar{q}[2] q[0] u + q[2] \bar{q}[0] \bar{u} \dots \dots \dots j[1] \bar{q}[1]$
 - ▶ $k[1] = \overline{\bar{q}[0] u + q[0] \bar{u} + q[2] \bar{q}[0] \bar{u}} \dots \dots \dots \bar{k}[1] q[1]$
- $ns[2] = \bar{q}[2] \bar{q}[1] \bar{q}[0] \bar{u} + q[1] q[0] u + q[2] \bar{q}[0] u + q[2] q[0] \bar{u}$
 - ▶ $j[2] = \bar{q}[1] \bar{q}[0] \bar{u} + q[1] q[0] u \dots \dots \dots j[2] \bar{q}[2]$
 - ▶ $k[2] = \overline{q[1] q[0] u + \bar{q}[0] u + q[0] \bar{u}} \dots \dots \dots \bar{k}[2] q[2]$

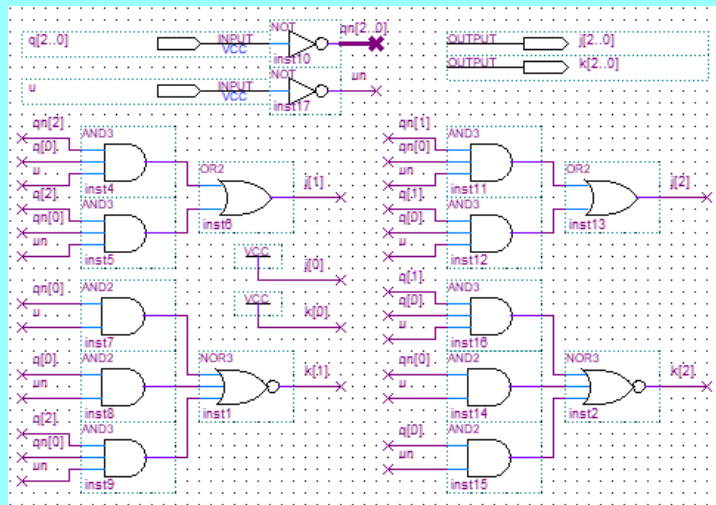
JKFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 3 ビット JKFF 回路:



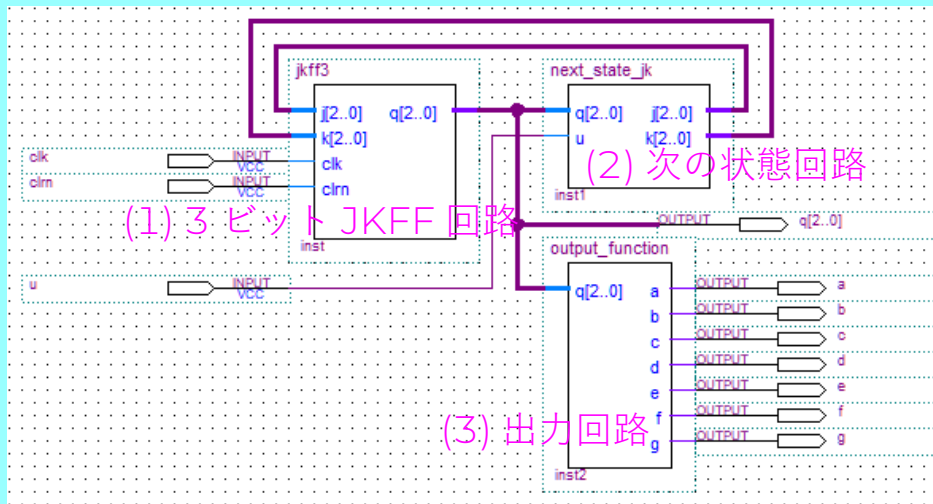
JKFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 次の状態回路:



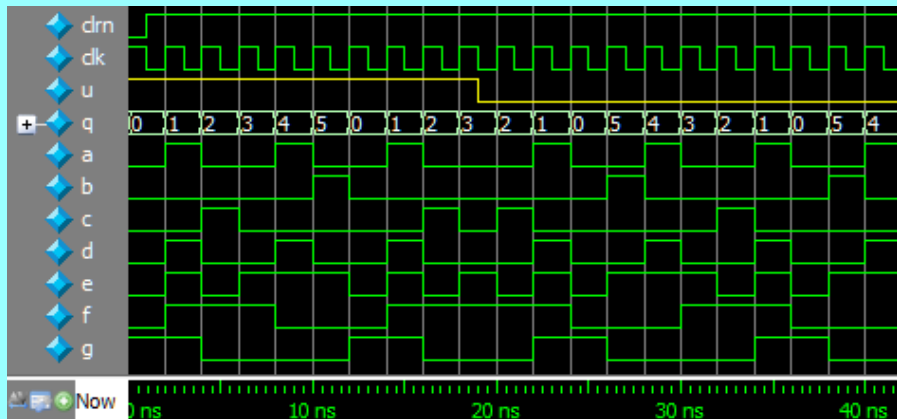
JKFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 全体の回路:



JKFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: シミュレーション波形:



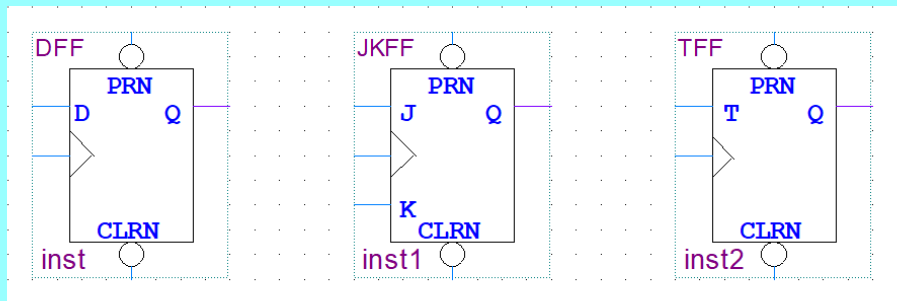
[counter6_jk_tb.v](#)

Count Up

Count Down

TFF で同期式 N 進 カウンターの設計

フリップフロップ



DFF : CLK の立ち上がりで、 $Q_{\text{next}} = D$

JKFF : CLK の立ち上がりで、 $Q_{\text{next}} = J \bar{Q} + \bar{K} Q$

TFF : CLK の立ち上がりで、 $Q_{\text{next}} = T \bar{Q} + \bar{T} Q$

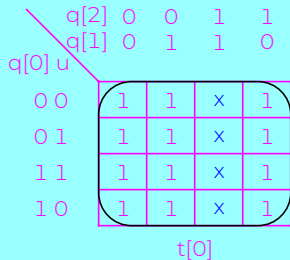
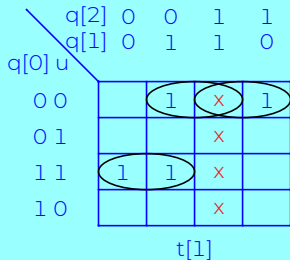
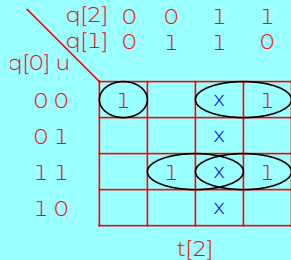
TFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 状態遷移表 (t = 1: Toggle)

現在の状態			入力	次の状態						
q[2]	q[1]	q[0]	u	ns[2]	ns[1]	ns[0]	t[2]	t[1]	t[0]	
S0	0	0	1	S1	0	0	1	0	0	1
			0	S5	1	0	1	1	0	1
S1	0	0	1	S2	0	1	0	0	1	1
			0	S0	0	0	0	0	0	1
S2	0	1	0	S3	0	1	1	0	0	1
			1	S1	0	0	1	0	1	1
S3	0	1	1	S4	1	0	0	1	1	1
			0	S2	0	1	0	0	0	1
S4	1	0	0	S5	1	0	1	0	0	1
			1	S3	0	1	1	1	1	1
S5	1	0	1	S0	0	0	0	1	0	1
			0	S4	1	0	0	0	0	1
入力として	1	1	0	X	Don't care	X	X	X	X	X
存在しない	1	1	1	X	ドントケア	X	X	X	X	X

TFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 次の状態:



TFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 次の状態:

	q[2]	0	0	1	1
q[1]	0	1	1	0	
q[0] u					
00	1		x	1	
01			x		
11	1	x	1		
10			x		

t[2]

	q[2]	0	0	1	1
q[1]	0	1	1	0	
q[0] u					
00		1	x	1	
01			x		
11	1	1	x		
10			x		

t[1]

	q[2]	0	0	1	1
q[1]	0	1	1	0	
q[0] u					
00	1	1	x	1	
01	1	1	x	1	
11	1	1	x	1	
10	1	1	x	1	

t[0]

$$t[2] = q[1] q[0] u + q[2] \overline{q[0]} \overline{u} + q[2] q[0] u + \overline{q[2]} \overline{q[1]} \overline{q[0]} \overline{u}$$

TFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 次の状態:

q[2]	0	0	1	1
q[1]	0	1	1	0
q[0] u				
00	1		x	1
01			x	
11		1	x	1
10			x	

t[2]

q[2]	0	0	1	1
q[1]	0	1	1	0
q[0] u				
00		1	x	1
01			x	
11	1	1	x	
10			x	

t[1]

q[2]	0	0	1	1
q[1]	0	1	1	0
q[0] u				
00	1	1	x	1
01	1	1	x	1
11	1	1	x	1
10	1	1	x	1

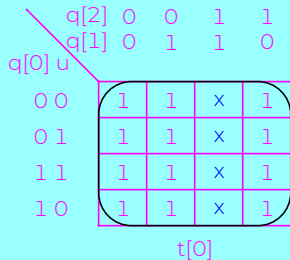
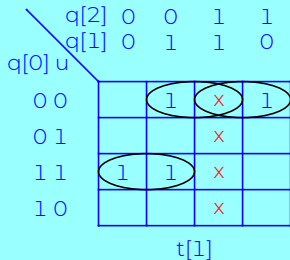
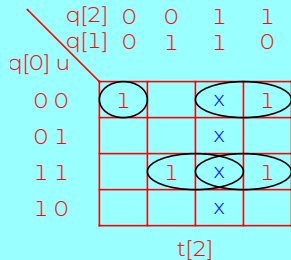
t[0]

$$t[2] = q[1] q[0] u + q[2] \overline{q[0]} \overline{u} + q[2] q[0] u + \overline{q[2]} \overline{q[1]} \overline{q[0]} \overline{u}$$

$$t[1] = \overline{q[2]} q[0] u + q[1] \overline{q[0]} \overline{u} + q[2] \overline{q[0]} \overline{u}$$

TFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 次の状態:



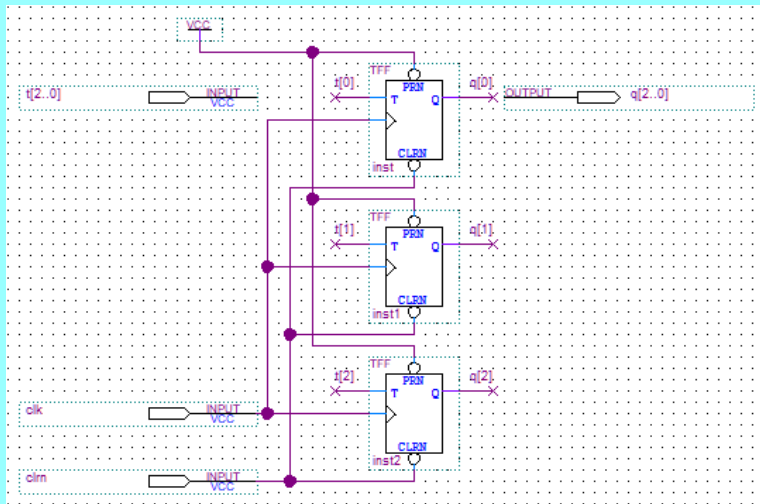
$$t[2] = q[1] q[0] u + q[2] \overline{q[0]} \overline{u} + q[2] q[0] u + \overline{q[2]} \overline{q[1]} \overline{q[0]} \overline{u}$$

$$t[1] = \overline{q[2]} q[0] u + q[1] \overline{q[0]} \overline{u} + q[2] \overline{q[0]} \overline{u}$$

$$t[0] = 1$$

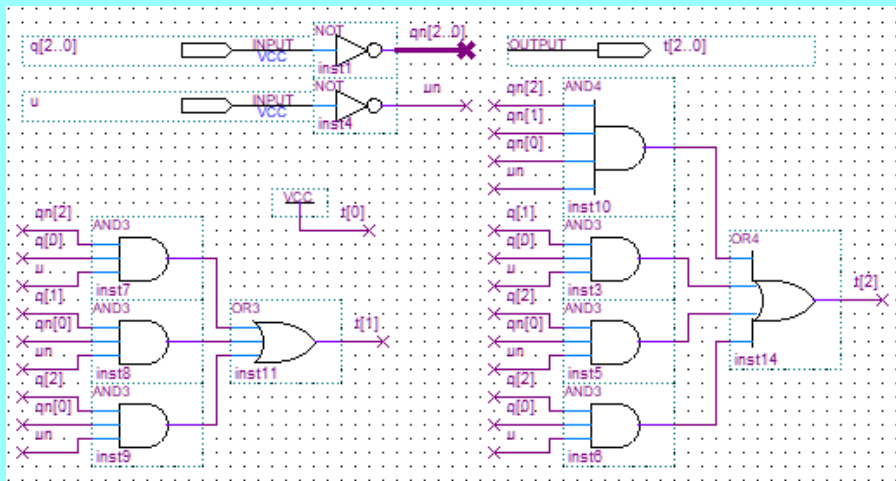
TFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 3 ビット TFF 回路:



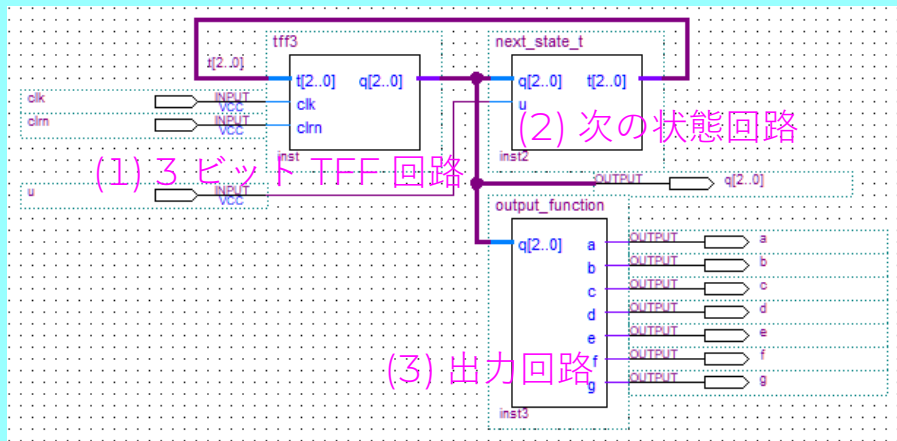
TFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 次の状態回路:



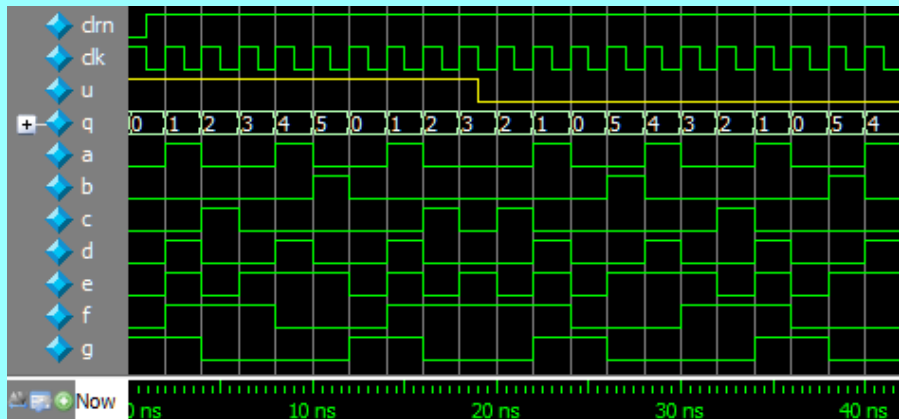
TFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: 全体の回路:



TFF で同期式 N 進カウンターの設計

0 ~ 5 カウンター: シミュレーション波形:



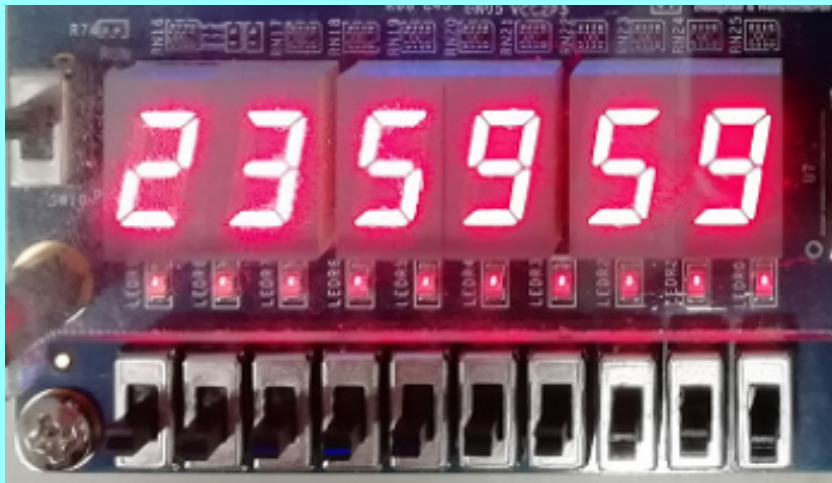
[counter6_t_tb.v](#)

Count Up

Count Down

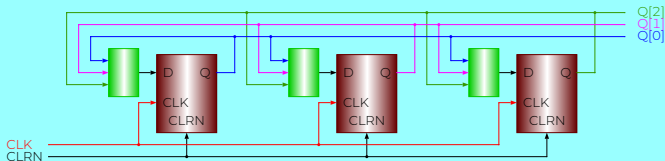
デジタル時計の実装

FPGAで実装したデジタル時計



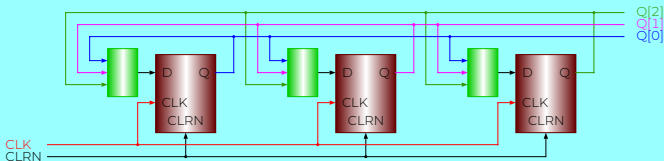
同期式と非同期式カウンタ

- 同期式 (Synchronous) とは、構成するフリップフロップの動作が同一のクロックで行われる方式のことである。

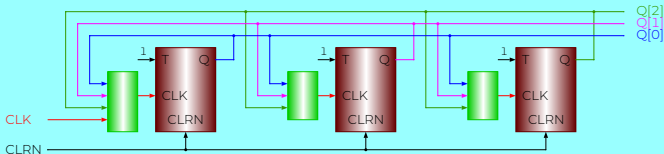


同期式と非同期式カウンタ

- 同期式 (Synchronous) とは、構成するフリップフロップの動作が同一のクロックで行われる方式のことである。

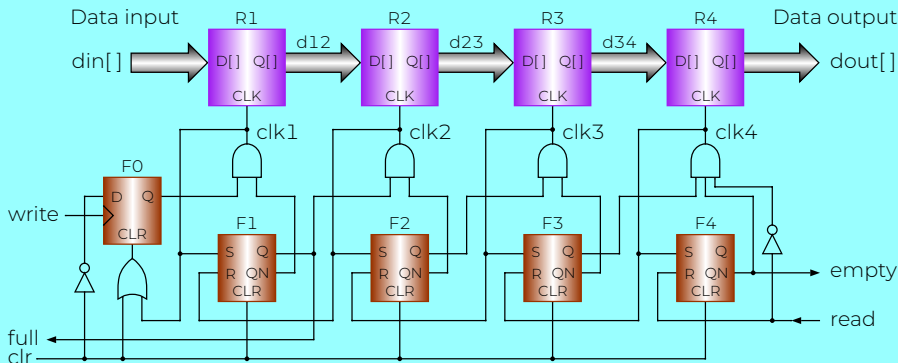


- 非同期式 (Asynchronous) とは、カウンタへの入力と各フリップフロップの出力の変化により、自身のフリップフロップの入力が決定され、その変化で動作する方式のことである。



非同期式 FIFO の設計

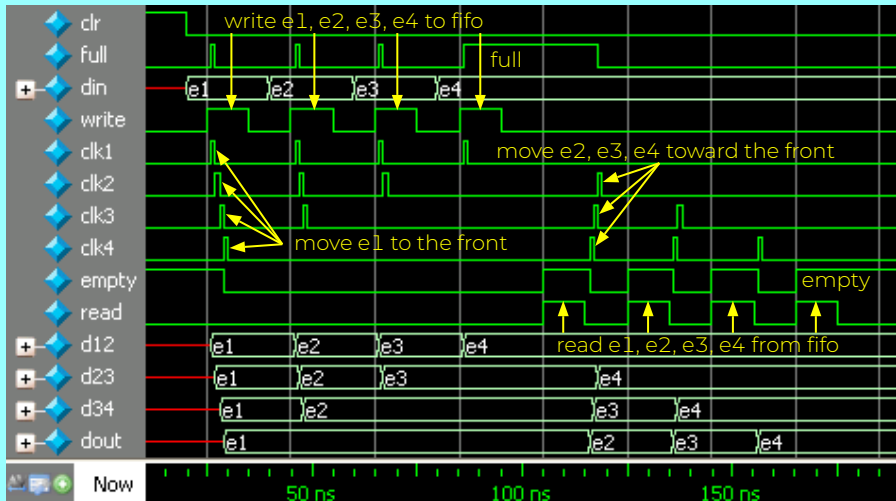
FIFO: First-In-First-Out (Queue — キュー)



R1、R2、R3、R4 : FIFO レジスタ (深さ : 4)

clk1、clk2、clk3、clk4 : 非同期 clock

非同期式 FIFO の設計



write: write data to FIFO; **read**: read data from FIFO.

同期式N進カウンターの設計

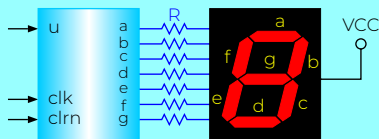
まとめ

- 同期式N進カウンター
 - ▶ DFF で同期式N進カウンター
 - ▶ JKFF で同期式N進カウンター
 - ▶ TFF で同期式N進カウンター
- 非同期式カウンター
- 非同期式 FIFO

課題 XIII (100 点 + 100 点)

7セグメント LED を使い同期式 10 進カウントアップ/カウントダウンする回路を、**DFF** を用いて設計し、動作検証シミュレーションして下さい (Don't Care 項を利用)。

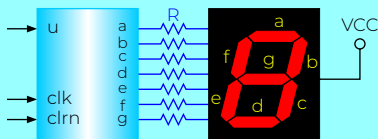
注意：AND5、OR5 を使用せず、
AND6、OR6 を使用してください。
第 6 回資料 P20 を参照してください。



- カウント範囲: 0 ~ 9
- $u = 1$ (カウントアップ):
0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, 1, 2, 3, ...
- $u = 0$ (カウントダウン):
0, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0, 9, 8, 7, ...
- 7セグメント LED で表示する (0: 点灯; 1: 消灯)

オプション (+50 点)

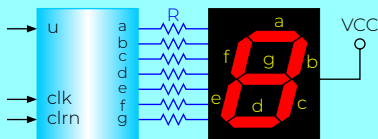
7セグメント LED を使い同期式 10 進カウントアップ/カウントダウンする回路を、JKFF を用いて設計し、動作検証シミュレーションして下さい (Don't Care 項を利用)。



- カウント範囲: 0 ~ 9
- $u = 1$ (カウントアップ):
0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, 1, 2, 3, ...
- $u = 0$ (カウントダウン):
0, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0, 9, 8, 7, ...
- 7セグメント LED で表示する (0: 点灯; 1: 消灯)

オプション (+50 点)

7セグメント LED を使い同期式 10 進カウントアップ/カウントダウンする回路を、TFF を用いて設計し、動作検証シミュレーションして下さい (Don't Care 項を利用)。



- カウント範囲: 0 ~ 9
- $u = 1$ (カウントアップ):
0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, 1, 2, 3, ...
- $u = 0$ (カウントダウン):
0, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0, 9, 8, 7, ...
- 7セグメント LED で表示する (0: 点灯; 1: 消灯)

発展：自由練習

Verilog HDL による課題の実装

[1]. 4ビット DFF の設計

```
module cnt10dff4 (clk, clrn, d, q);
  input  [3:0] d; // next state
  input      clk, clrn;
  output [3:0] q; // current state
  reg      [3:0] q; // 4-bit DFFs
  always @(posedge clk or negedge clrn) begin
    if (!clrn) begin
      q <= 0; // clear DFFs
    end else begin
      q <= d; // save D to DFFs
    end
  end
end
endmodule
```

発展：自由練習

Verilog HDL による課題の実装

[2]. 次の状態回路の設計

```
module cnt10nextstate (u, q, d);
    input  [3:0] q; // current state
    input      u; // input: count up/down
    output [3:0] d; // next state
    assign d[3] = (q[3] & u) | (q[3] & !u & q[2]); // next state d[3]
    assign d[2] = (q[2] & u) | (q[2] & !u & q[1]); // next state d[2]
    assign d[1] = (q[1] & u) | (q[1] & !u & q[0]); // next state d[1]
    assign d[0] = (q[0] & u) | (q[0] & !u & !q[3]); // next state d[0]
endmodule
```

上記コードを完成しシミュレーションして下さい。

発展：自由練習

Verilog HDL による課題の実装

[3]. 出力関数の回路の設計

```
module cnt10output (q, seven_seg);
    input  [3:0] q; // current state
    output [6:0] seven_seg; // output: 7-seg LED control
    assign seven_seg[0] =          ; // a
    assign seven_seg[1] =          ; // b
    assign seven_seg[2] =          ; // c
    assign seven_seg[3] =          ; // d
    assign seven_seg[4] =          ; // e
    assign seven_seg[5] =          ; // f
    assign seven_seg[6] =          ; // g
endmodule
```

上記コードを完成しシミュレーションして下さい。

発展：自由練習

Verilog HDL による課題の実装

[全体]. 10進カウントアップ/カウントダウンする回路の設計

```
module cnt10top (clk, clrn, u, a, b, c, d, e, f, g);
    input  clk, clrn, u;                // inputs
    output a, b, c, d, e, f, g;        // 7-seg outputs
    wire [6:0] seven_seg;              // internal wire
    wire [3:0] p;                       // next state
    wire [3:0] q;                       // current state
    cnt10dff4    i1 (                    ); // [1]. 4-bit D flip flops
    cnt10nextstate i2 (                  ); // [2]. next state
    cnt10output  i3 (                    ); // [3]. output function
    assign {g, f, e, d, c, b, a} = seven_seg; // 7-seg outputs
endmodule
```

上記コードを完成しシミュレーションして下さい。