

Counter (3A)

Copyright (c) 2011-2013 Young W. Lim.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Please send corrections (or suggestions) to youngwlim@hotmail.com.

This document was produced by using OpenOffice and Octave.

Toggling Input

Shift Register

Feedback Flip Flop

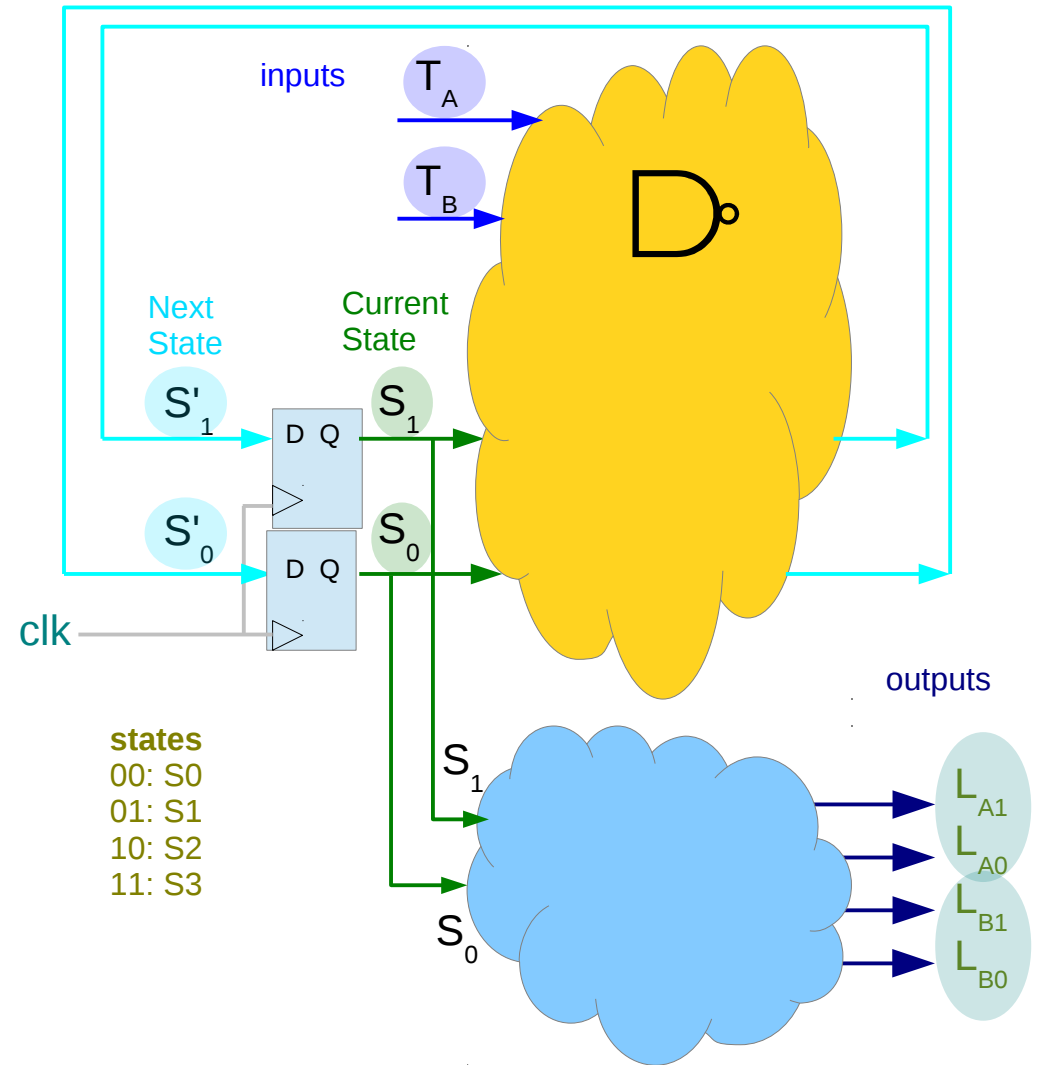
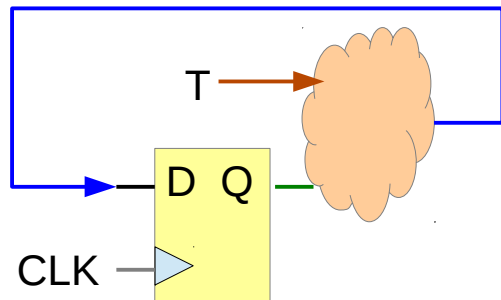
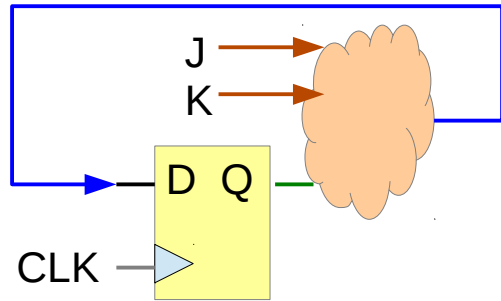
- JK Flip Flop
- T Flip Flop
- Toggling D Flip Flop

Pipeline Stage Register

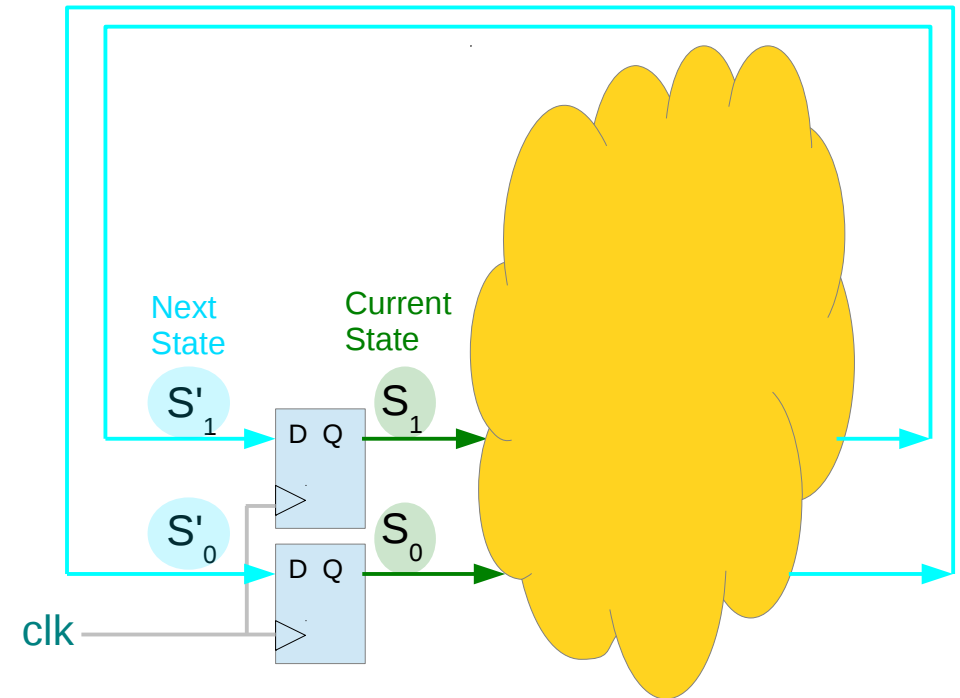
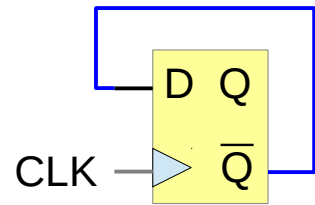
Feedback Register

- FSM State Register
- Counter Register
-

JKFF & TFF v.s. FSM

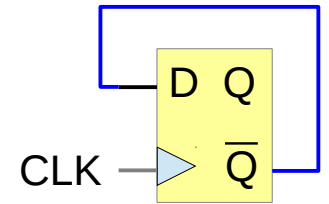
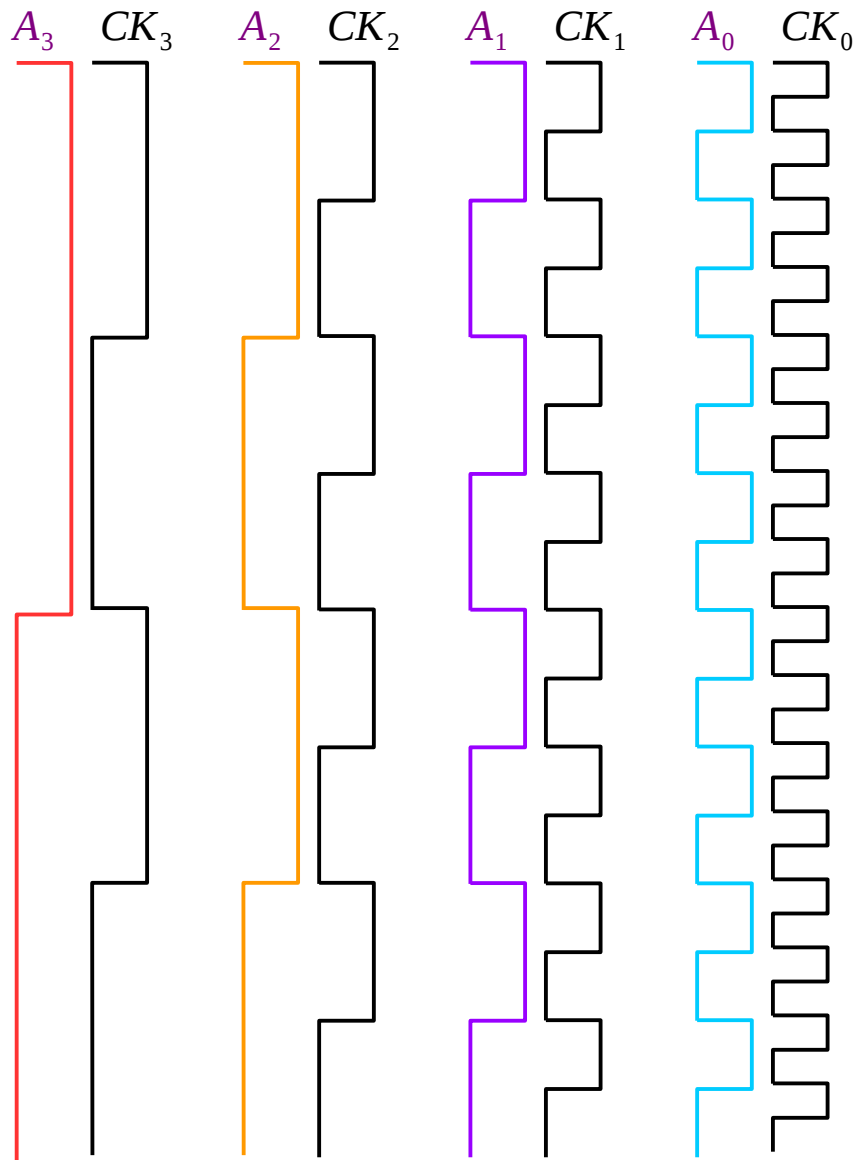


Toggleing DFF v.s. Counter



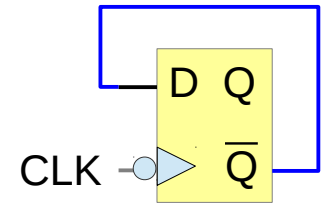
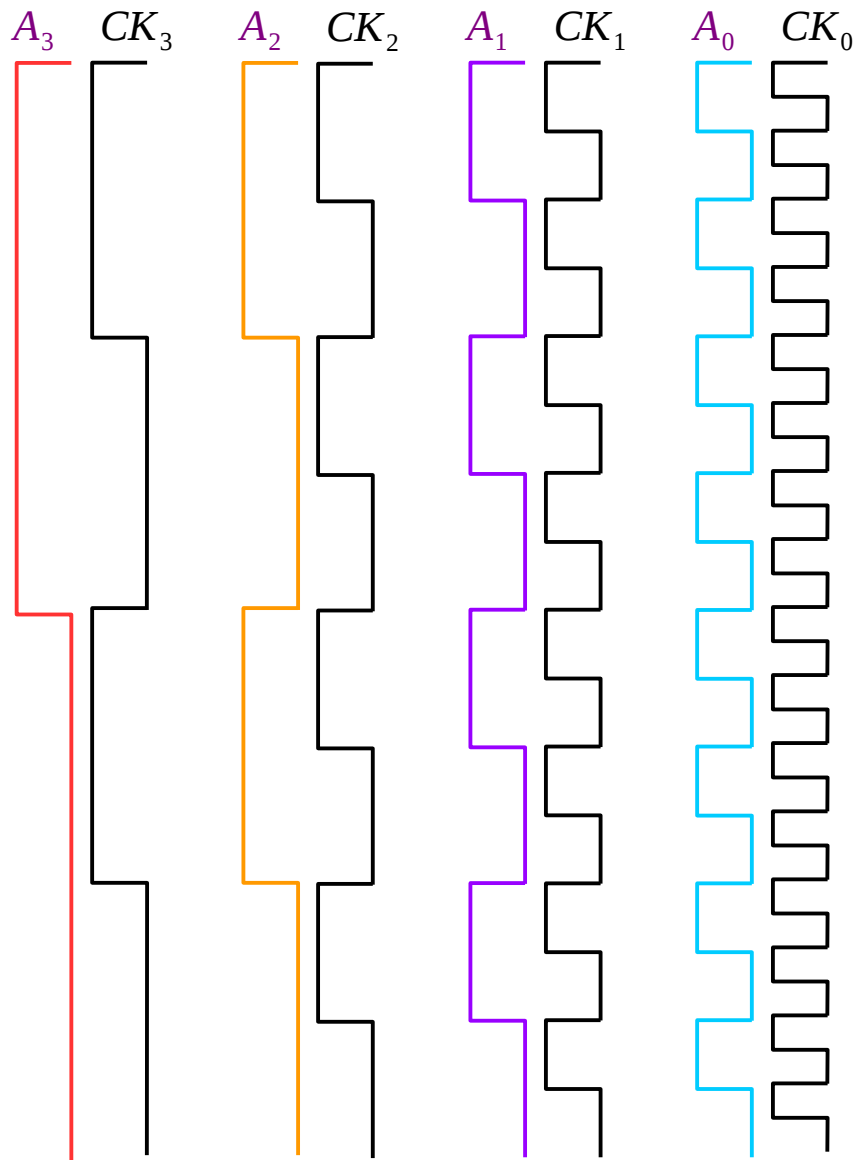
Toggle Count Down

2^3	2^2	2^1	2^0
1	1	1	1
1	1	1	0
1	1	0	1
1	1	0	0
1	0	1	1
1	0	1	0
1	0	0	1
1	0	0	0
0	1	1	1
0	1	1	0
0	1	0	1
0	1	0	0
0	0	1	1
0	0	1	0
0	0	0	1
0	0	0	0

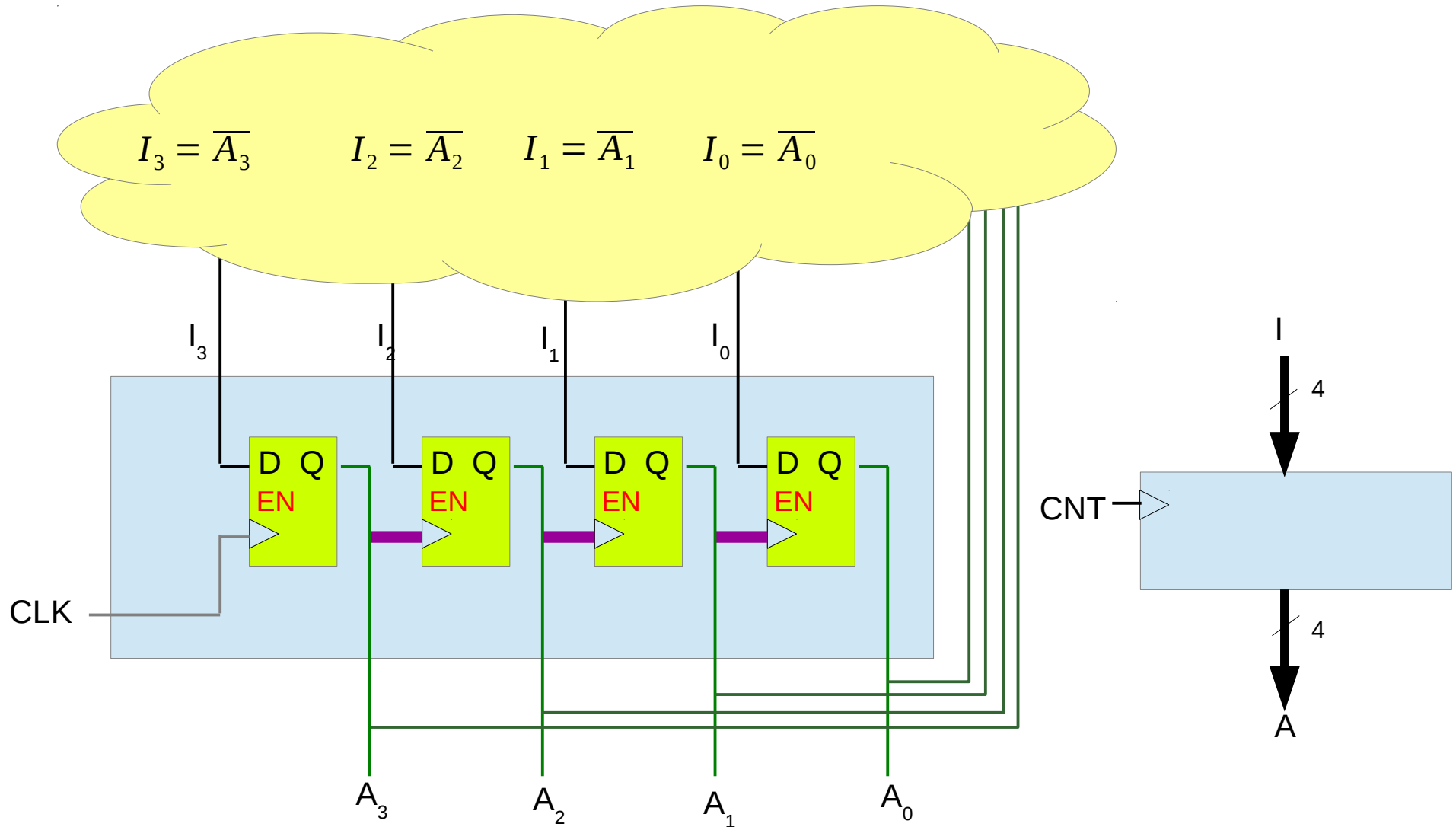


Toggle Count Up

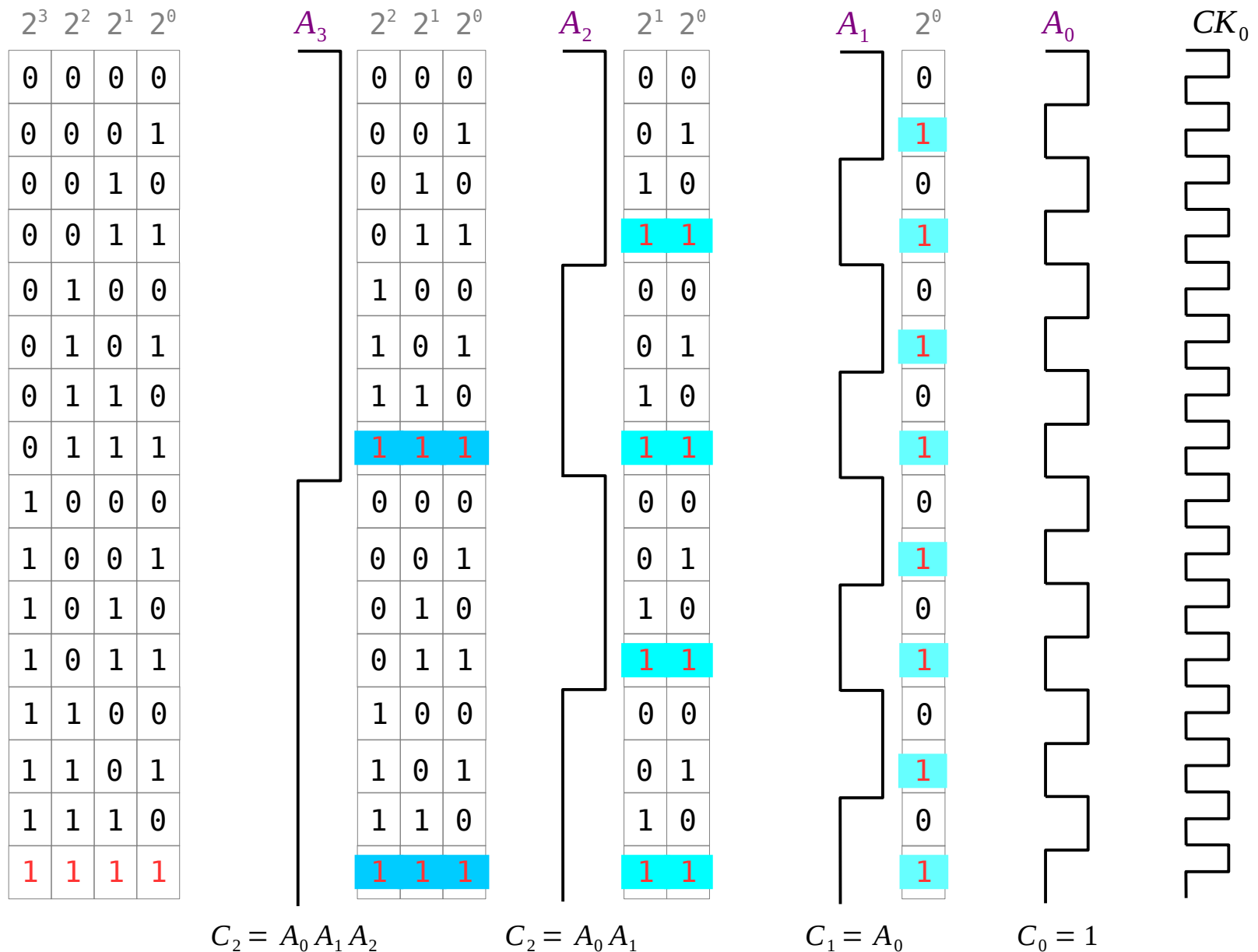
2^3	2^2	2^1	2^0
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1



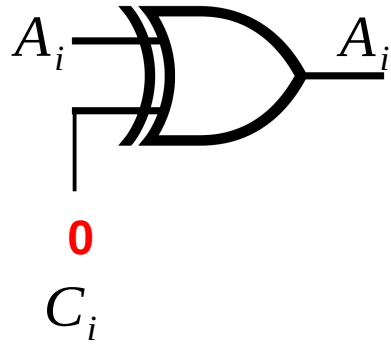
Ripple Counter - multiple clocks



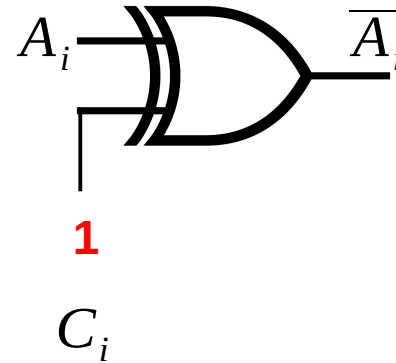
Toggle Conditions



Toggle Conditions



$$x \oplus 0 = x$$



$$x \oplus 1 = \bar{x}$$

$$I_3 = C_3 \mathbf{xor} A_3$$

$$C_3 = A_2 A_1 A_0 \cdot EN$$

$$I_2 = C_2 \mathbf{xor} A_2$$

$$C_2 = A_1 A_0 \cdot EN$$

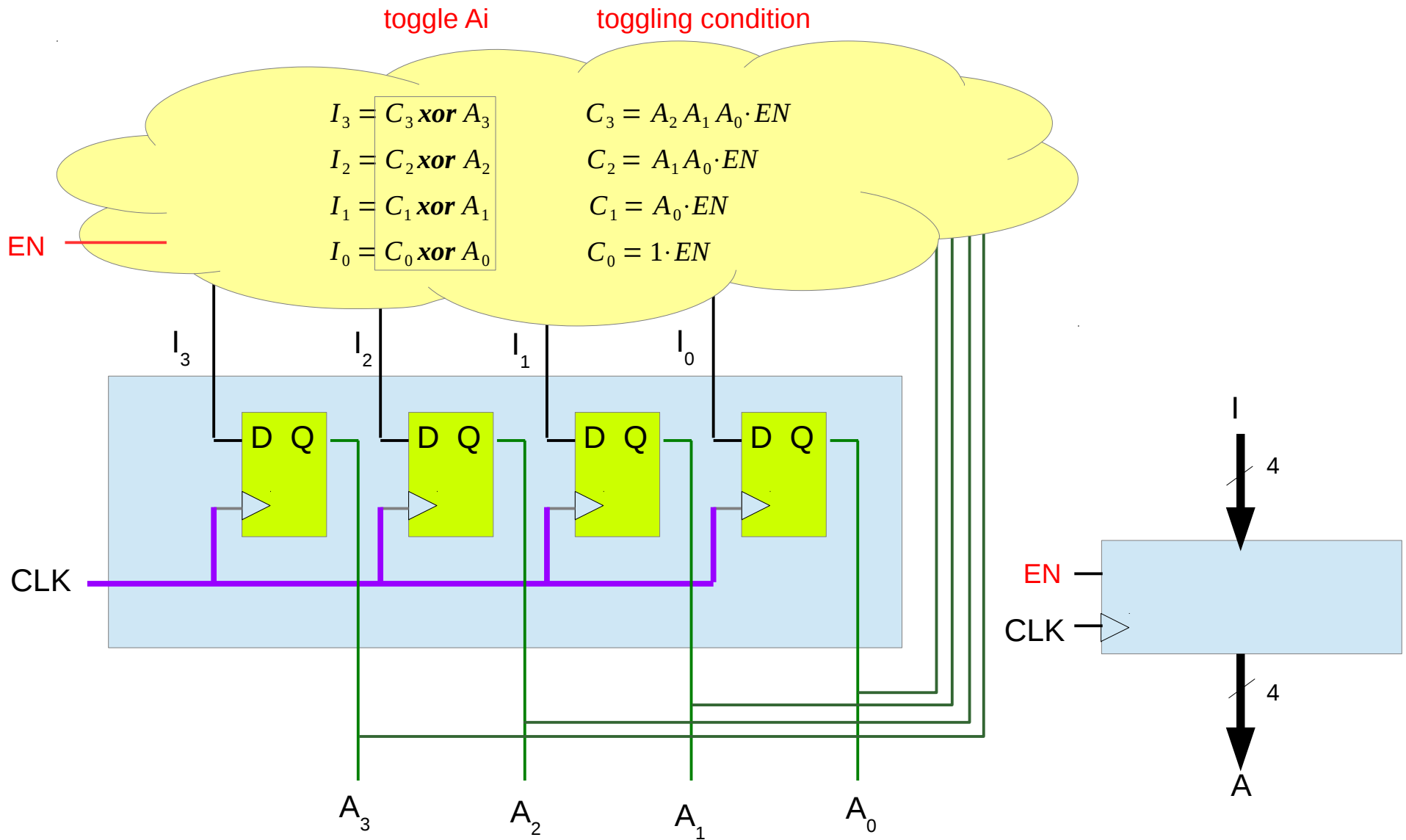
$$I_1 = C_1 \mathbf{xor} A_1$$

$$C_1 = A_0 \cdot EN$$

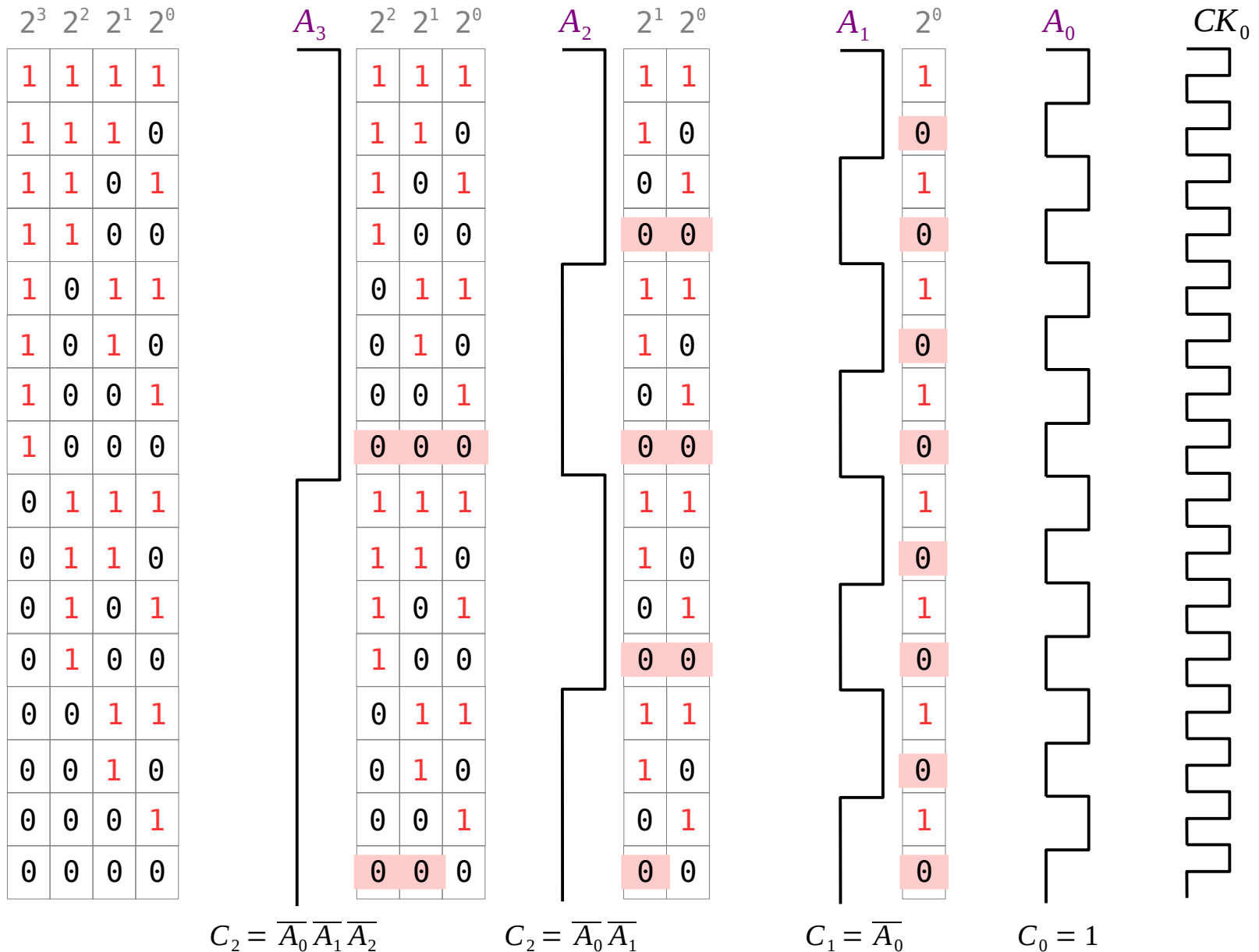
$$I_0 = C_0 \mathbf{xor} A_0$$

$$C_0 = 1 \cdot EN$$

Synchronous Binary Counter – a single clock



Toggle Conditions



Synchronous UpDown Counter – a single clock

toggle A_i

up counting condition

down counting condition

$$I_3 = C_3 \text{ xor } A_3$$

$$I_2 = C_2 \text{ xor } A_2$$

$$I_1 = C_1 \text{ xor } A_1$$

$$I_0 = C_0 \text{ xor } A_0$$

$$C_3 = A_2 A_1 A_0 \cdot EN$$

$$C_2 = A_1 A_0 \cdot EN$$

$$C_1 = A_0 \cdot EN$$

$$C_0 = 1 \cdot EN$$

$$C_3 = \overline{A_2} \overline{A_1} \overline{A_0} \cdot EN$$

$$C_2 = \overline{A_1} \overline{A_0} \cdot EN$$

$$C_1 = \overline{A_0} \cdot EN$$

$$C_0 = 1 \cdot EN$$

toggle A_i

toggle condition

$$I_3 = G_3 \text{ xor } A_3$$

$$I_2 = G_2 \text{ xor } A_2$$

$$I_1 = G_1 \text{ xor } A_1$$

$$I_0 = G_0 \text{ xor } A_0$$

$$G_3 = \overline{S} A_2 A_1 A_0 + S \overline{A_2} \overline{A_1} \overline{A_0} \cdot EN$$

$$G_2 = \overline{S} A_1 A_0 + S \overline{A_1} \overline{A_0} \cdot EN$$

$$G_1 = \overline{S} A_0 + S \overline{A_0} \cdot EN$$

$$G_0 = 1 \cdot EN$$

S=0 Up Counting

S=1 Down Counting

References

- [1] <http://en.wikipedia.org/>
- [2] M. M. Mano, C. R. Kime, "Logic and Computer Design Fundamentals", 4th ed.
- [3] D.M. Harris, S. L. Harris, "Digital Design and Computer Architecture"