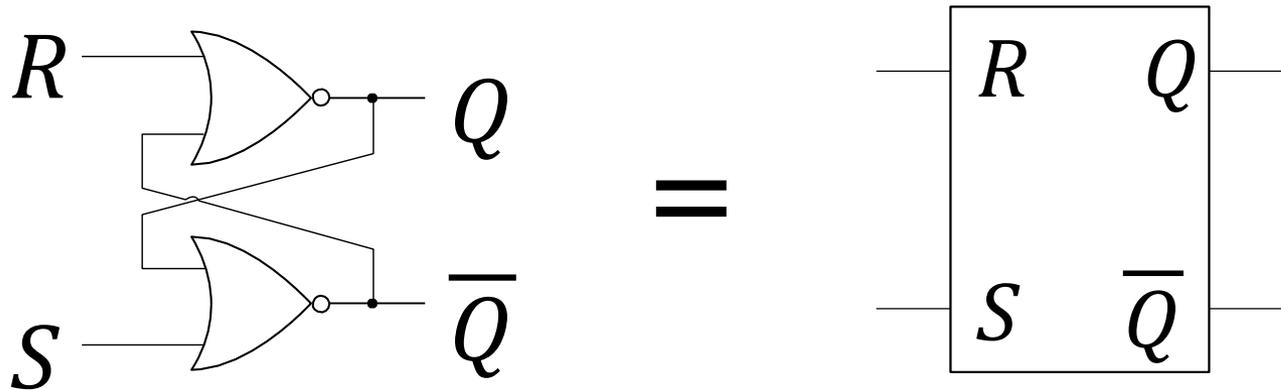


CIRCUITOS LÓGICOS FLIP-FLOPS

Marco A. Zanata Alves

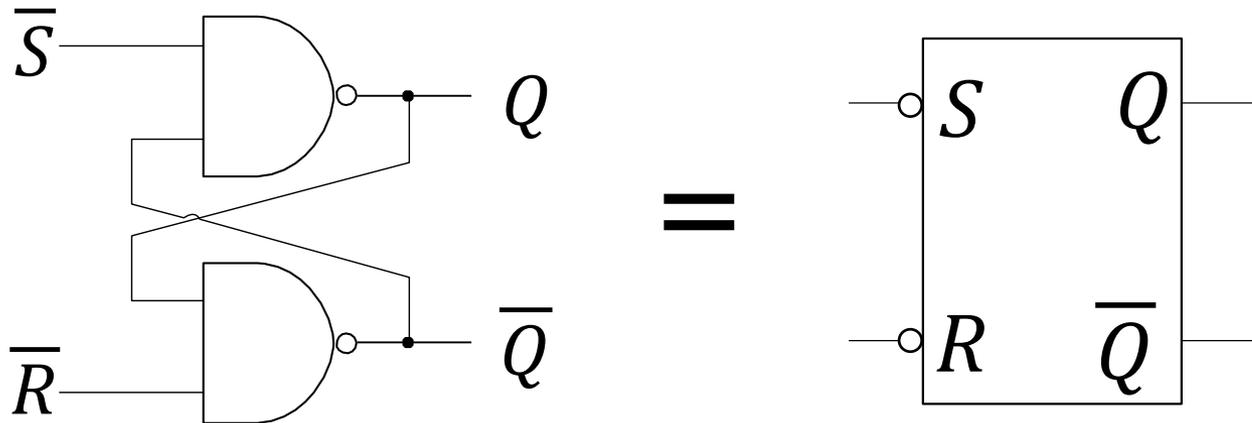
LATCH DO TIPO R-S (RESET-SET)



| R | S | Q_i | \overline{Q}_i | |
|---|---|-----------|----------------------|-----------------|
| 1 | 0 | 0 | 1 | (reset Q) |
| 0 | 1 | 1 | 0 | (set Q) |
| 0 | 0 | Q_{i-1} | \overline{Q}_{i-1} | (mantém Q) |
| 1 | 1 | X | X | Estado Proibido |

LATCH DO TIPO $\bar{S} - \bar{R}$ COM PORTAS NAND

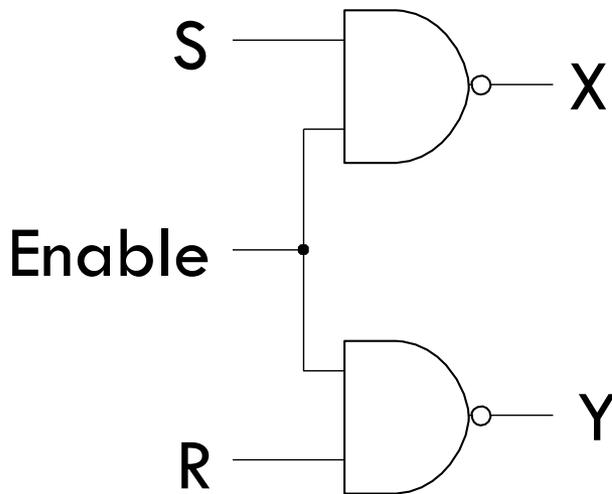
É possível construir um latch similar com portas NAND, mas as entradas se tornam ativas em nível baixo.



| \bar{S} | \bar{R} | Q_i | \bar{Q}_i | |
|-----------|-----------|-----------|----------------------|-----------------|
| 0 | 1 | 0 | 1 | (reset Q) |
| 1 | 0 | 1 | 0 | (set Q) |
| 0 | 0 | Q_{i-1} | $\overline{Q_{i-1}}$ | (mantém Q) |
| 1 | 1 | X | X | Estado Proibido |

CIRCUITO DE HABILITAÇÃO (ENABLE)

Problema 1: considere o circuito abaixo. Qual é o estado de cada saída X e Y quando $En = 0$ e quando $En = 1$?



| Enable | X | Y |
|--------|-----------|-----------|
| 0 | 1 | 1 |
| 1 | \bar{S} | \bar{R} |

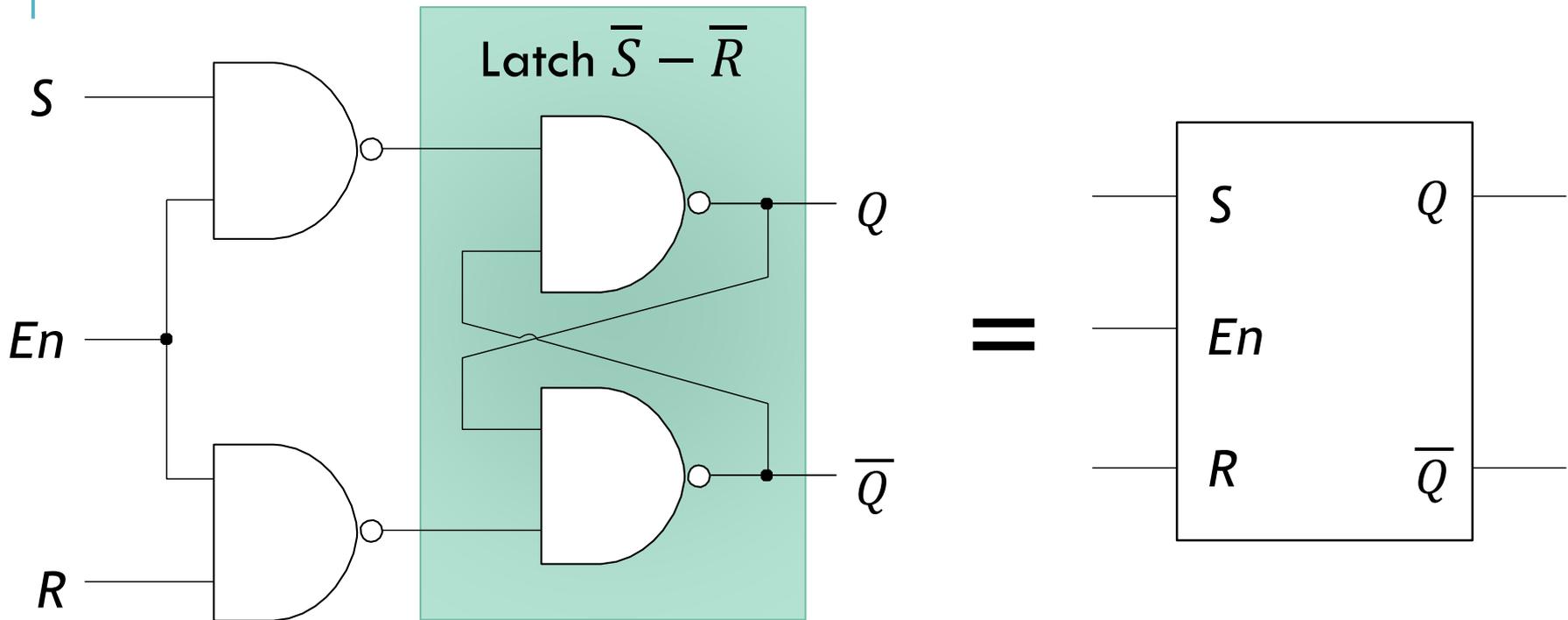
Circuito de habilitação com portas NAND torna as entradas S e R:

Se $En = 1$: ativas em nível baixo, (\bar{S} e \bar{R})

Se $En = 0$, desabilitadas

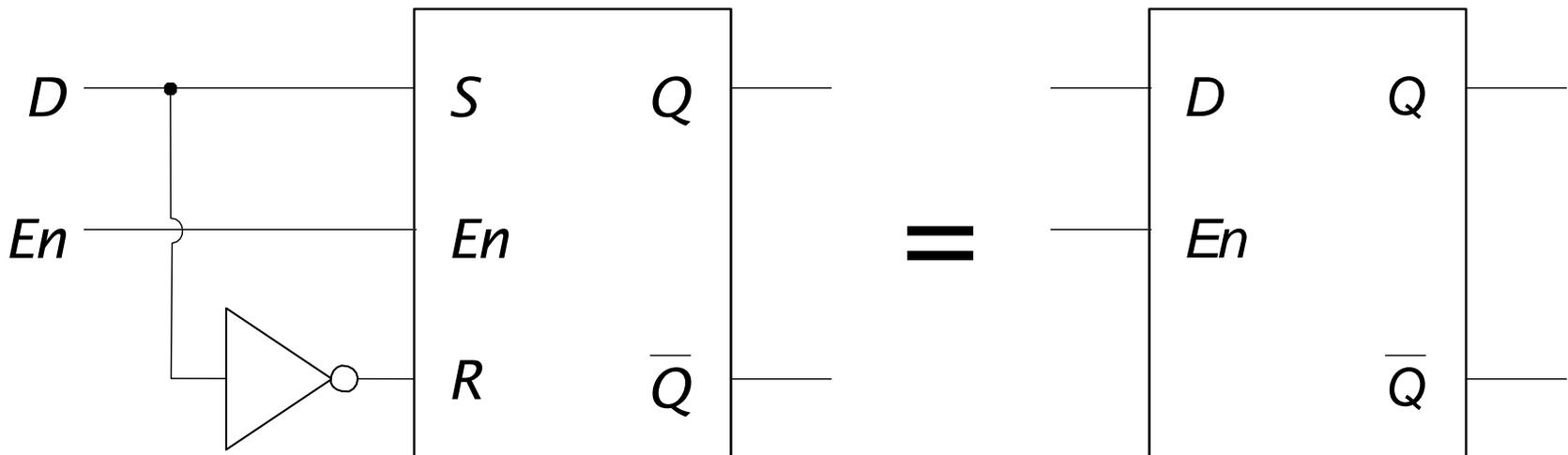
A entrada **Enable** é chamada entrada de habilitação (enable input)

LATCH DO TIPO S-R COM ENABLE

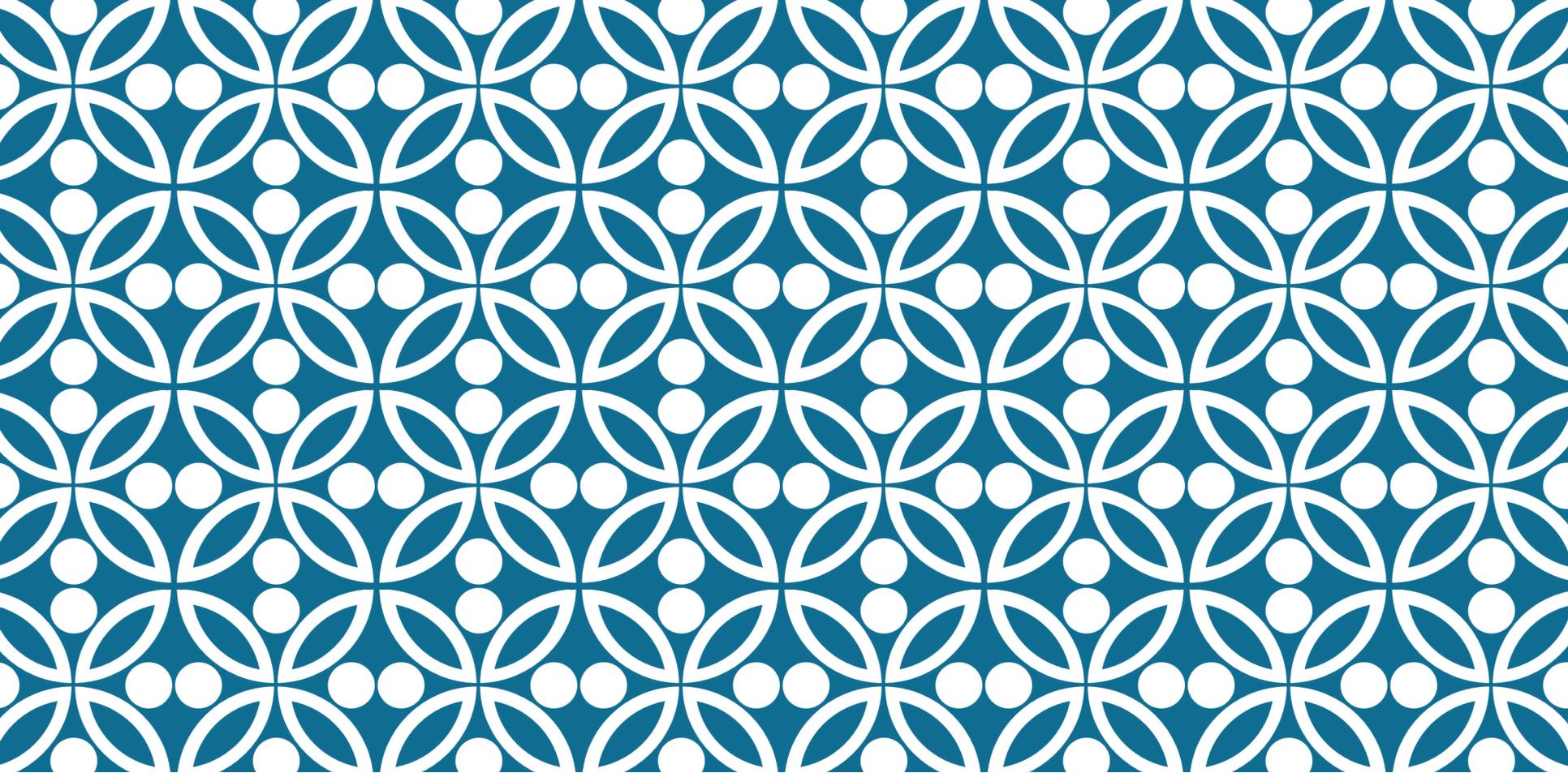


| En | S | R | Q_i | |
|----|---|---|-----------|---------------------------------|
| 1 | 0 | 1 | 0 | (reseta Q) |
| 1 | 1 | 0 | 1 | (seta Q) |
| 1 | 0 | 0 | Q_{i-1} | (mantém Q) |
| 0 | ? | ? | Q_{i-1} | (mantém Q, não importa R nem S) |

LATCH DO TIPO D (DATA)



| D | En | Q_i | |
|-----|------|-----------|------------------------------|
| 0 | 1 | 0 | (reset) |
| 1 | 1 | 1 | (set) |
| ? | 0 | Q_{i-1} | (mantém, sem importar com D) |

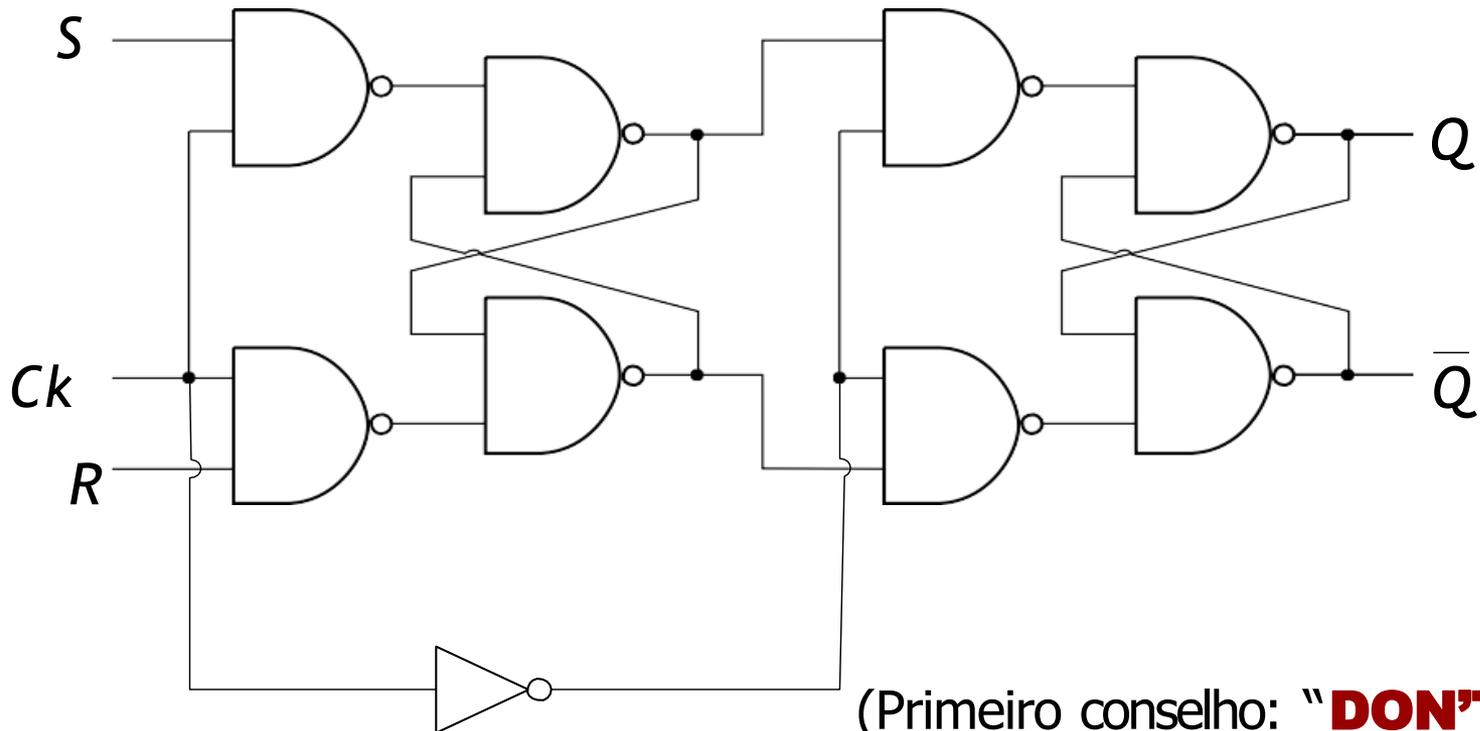


FLIP FLOPS



FLIP-FLOPS

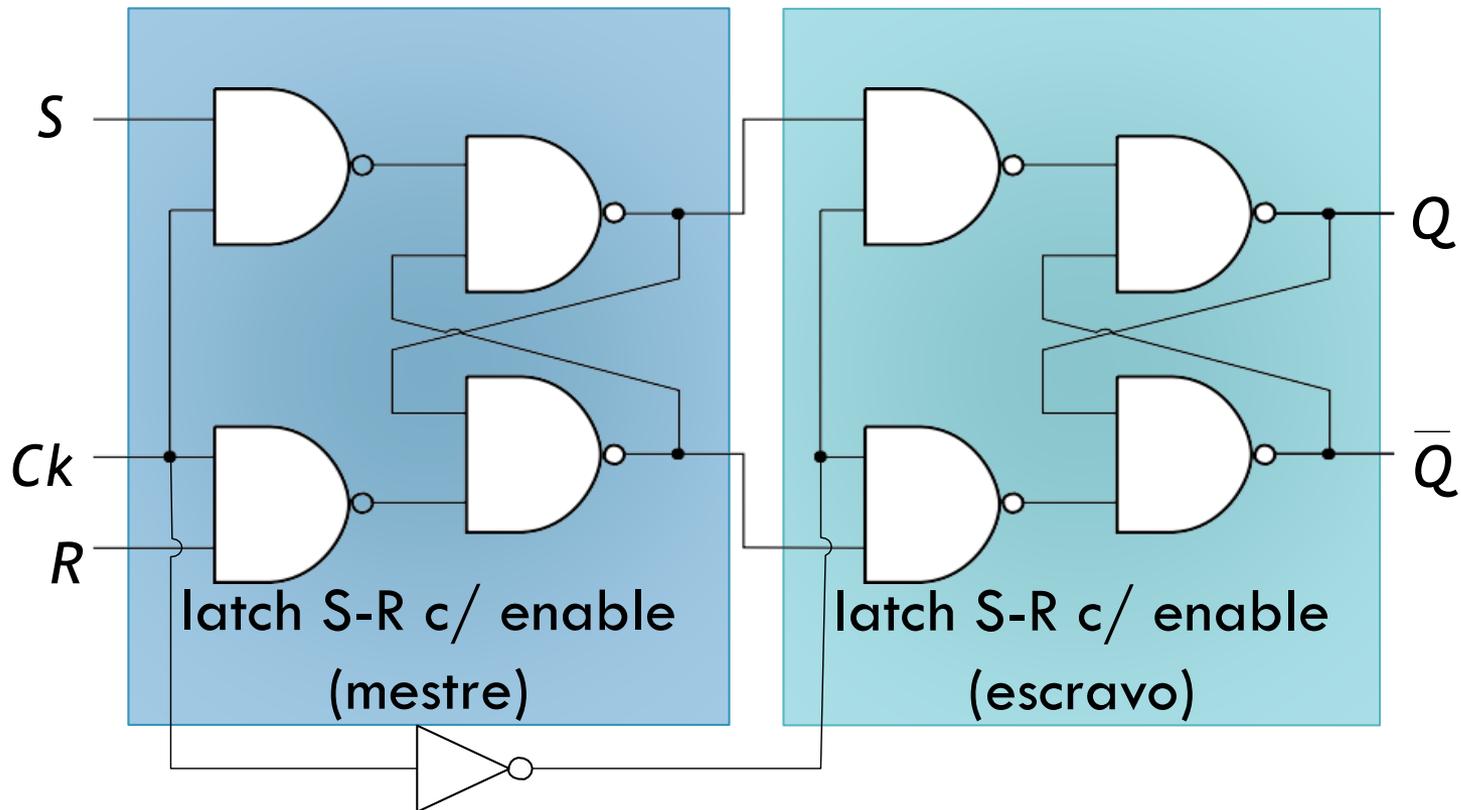
Analise o comportamento do circuito abaixo.



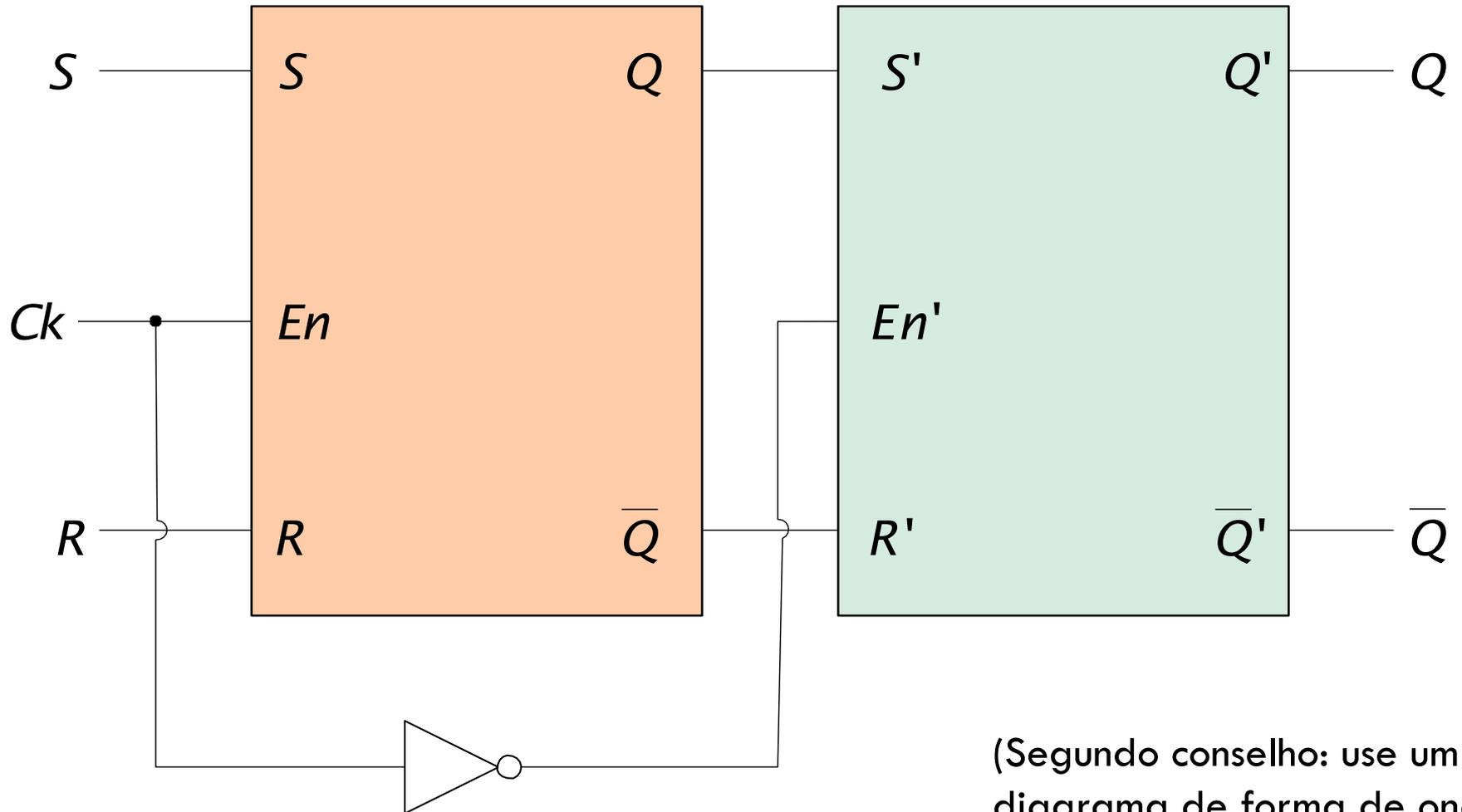
(Primeiro conselho: **"DON'T PANIC"**)

FLIP-FLOPS

Analise o comportamento do circuito abaixo.

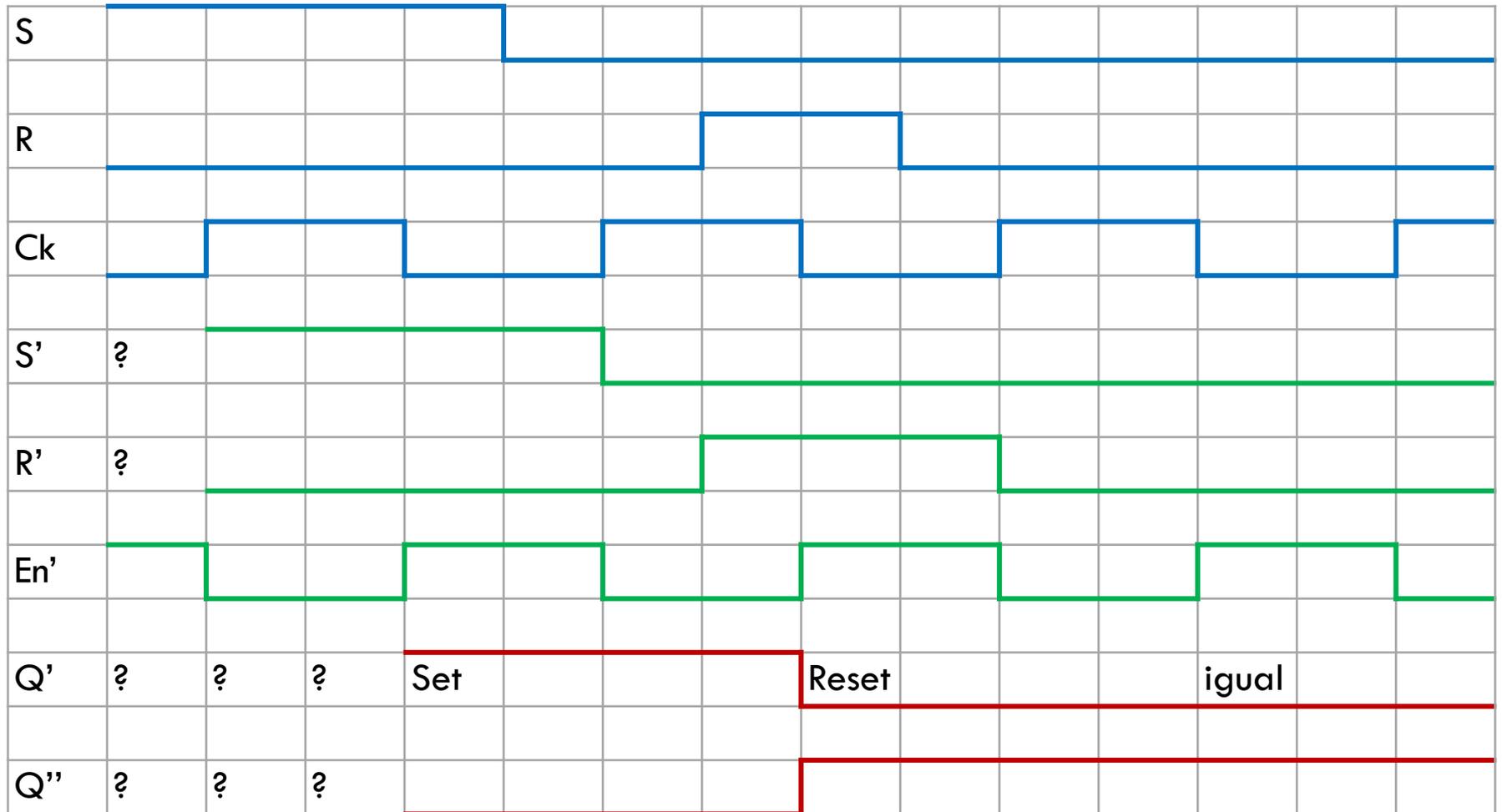


FLIP-FLOPS



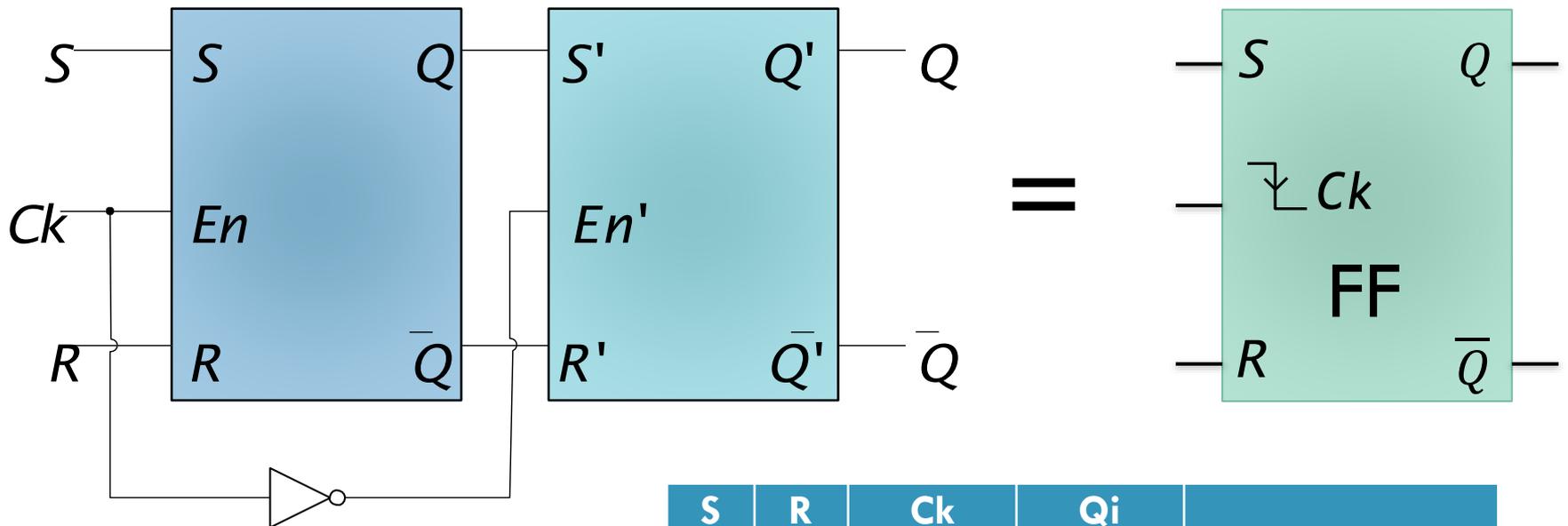
(Segundo conselho: use um diagrama de forma de onda)

FLIP-FLOP S-R: DIAGRAMA DE FORMA DE ONDA



FLIP-FLOP S-R

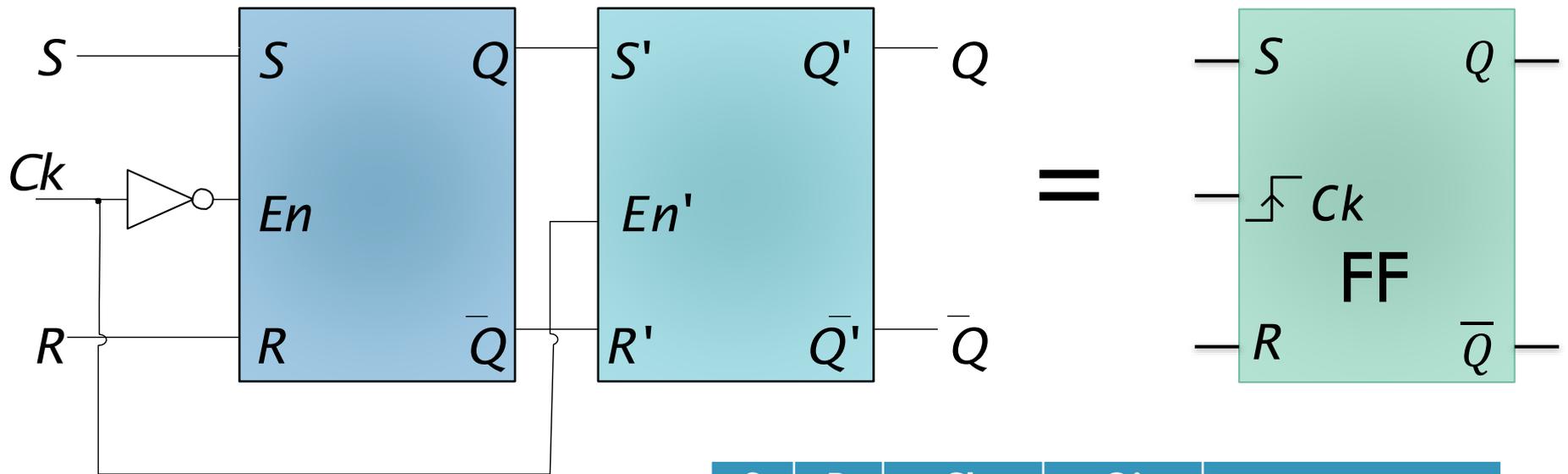
Flip-flop S-R sensível à borda de descida do clock (borda negativa)



| S | R | Ck | Qi | |
|---|---|-----|------|------------|
| 0 | 0 | ? | Qi-1 | (mantem Q) |
| 0 | 1 | 1→0 | 0 | (reset Q) |
| 1 | 0 | 1→0 | 1 | (set Q) |
| 1 | 1 | 1→0 | X | (proibido) |

FLIP-FLOP S-R

Flip-flop S-R sensível à borda de subida do clock (borda positiva)



| S | R | Ck | Qi | |
|---|---|-----|------|------------|
| 0 | 0 | ? | Qi-1 | (mantem Q) |
| 0 | 1 | 0→1 | 0 | (reset Q) |
| 1 | 0 | 0→1 | 1 | (set Q) |
| 1 | 1 | 0→1 | X | (proibido) |

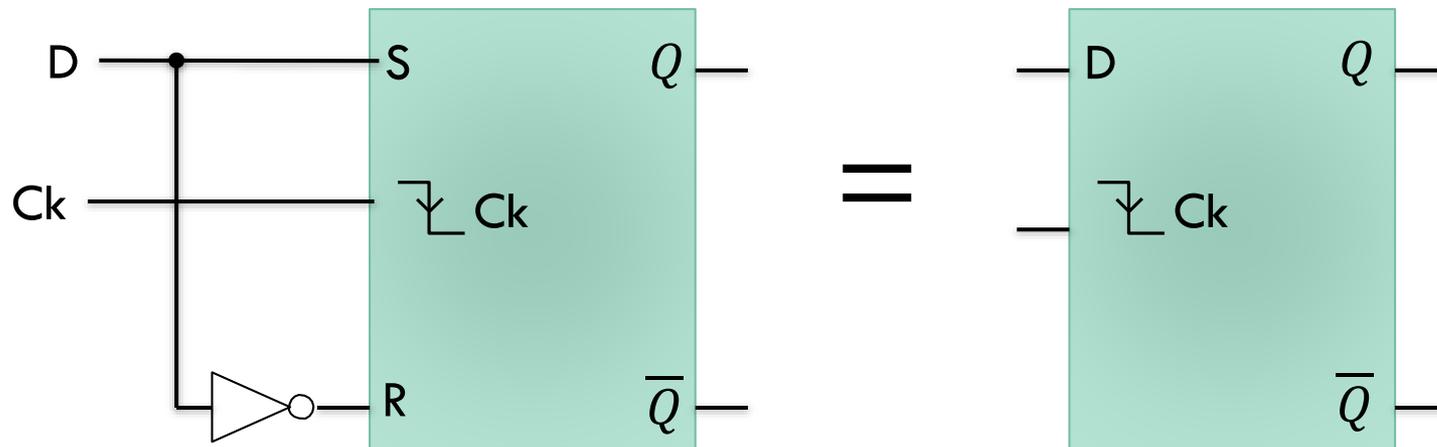
FLIP-FLOP S-R: ENTRADAS PROIBIDAS

Assim como o latch S-R e o latch S-R com enable, o flip-flop S-R não admite que ambas as entradas S e R estejam ativas quando a borda de descida/subida do clock é detectada.

Para um flip-flop S-R sensível à borda de subida, se $S = 1$, $R = 1$ e Ck fizer a transição $0 \rightarrow 1$, o circuito entra em oscilação descontrolada

FLIP-FLOP D: MEMÓRIA SÍNCRONA DE 1 BIT

Solução 1: evitar que ambas as entradas fiquem em 1, fazendo um flip-flop D



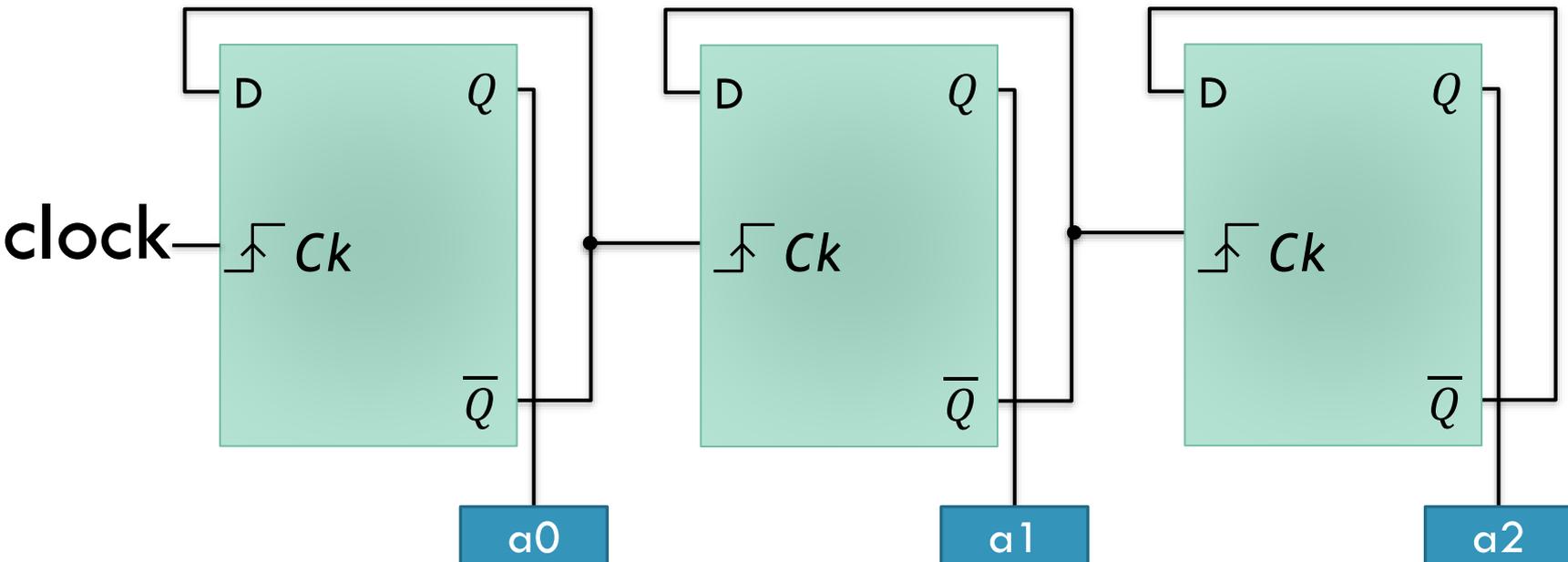
| D | Ck | Q_i | |
|---|-----|-------|----------------------|
| 0 | 1→0 | 0 | (reset = armazena 0) |
| 1 | 1→0 | 1 | (set = armazena 1) |

FLIP-FLOP D: APLICAÇÃO

O que faz o circuito abaixo?

- entrada: Ck
- saídas: a_2, a_1, a_0

Suponha que o estado inicial de cada saída é 0.



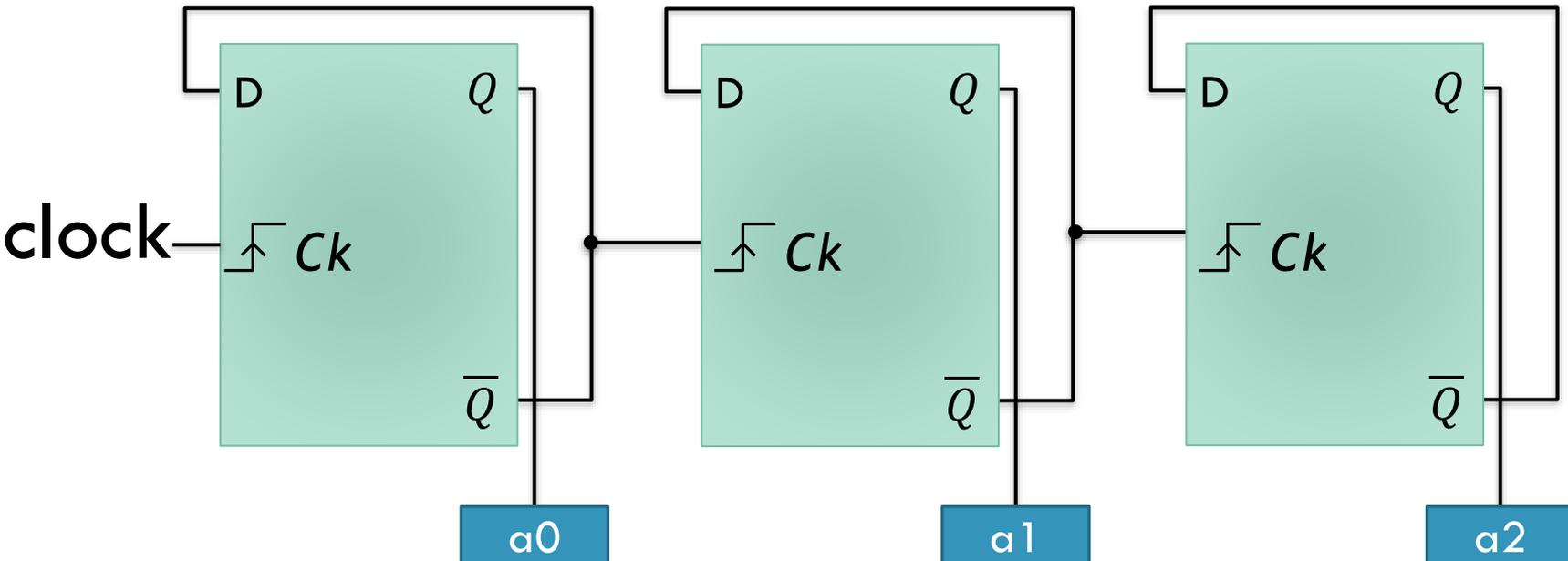
FLIP-FLOP D: APLICAÇÃO

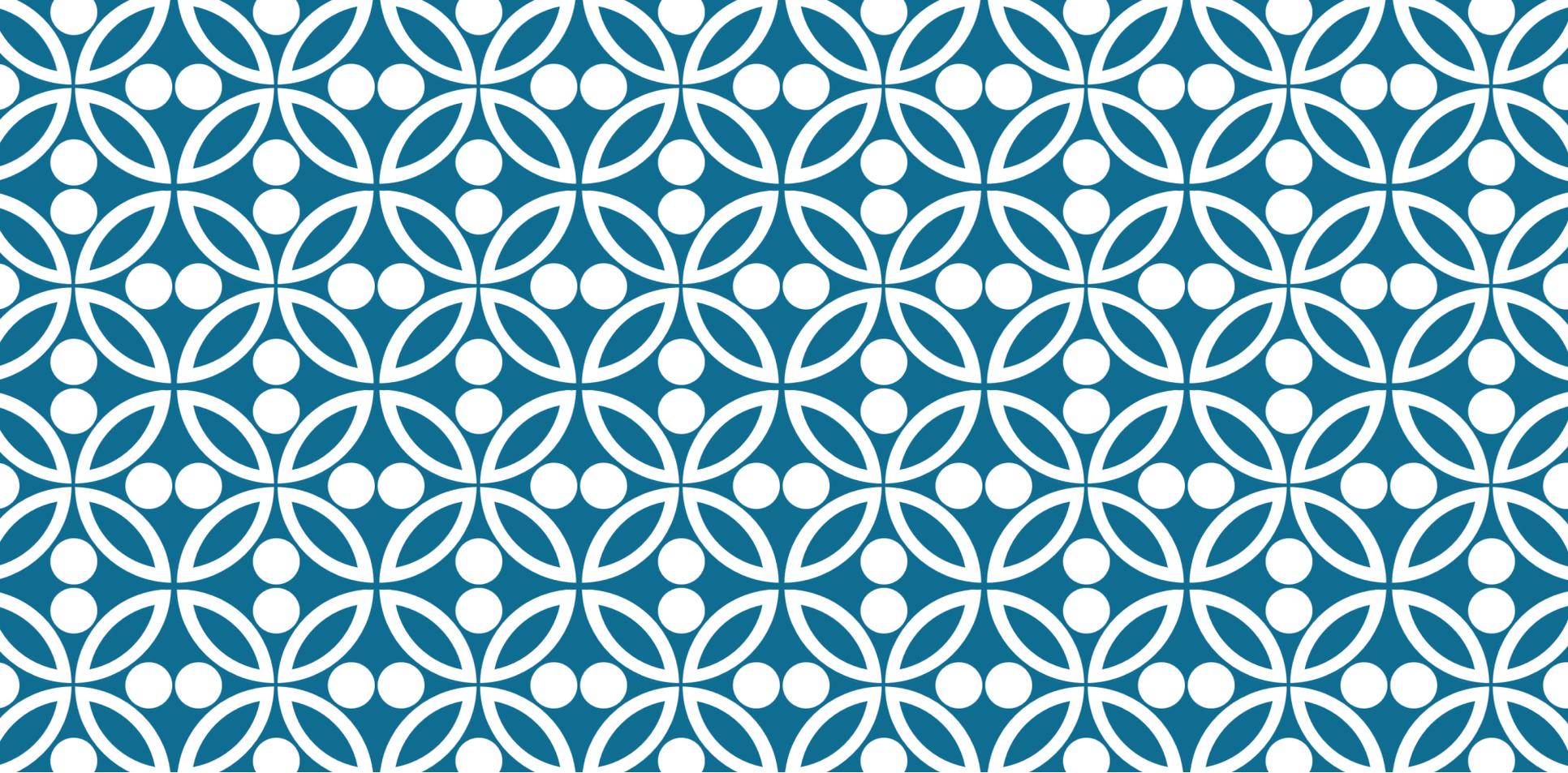
Contador de 3 bits!

O que faz o circuito abaixo?

- entrada: Ck
- saídas: a2, a1, a0

Suponha que o estado inicial de cada saída é 0.





CIRCUITOS PROBLEMÁTICOS

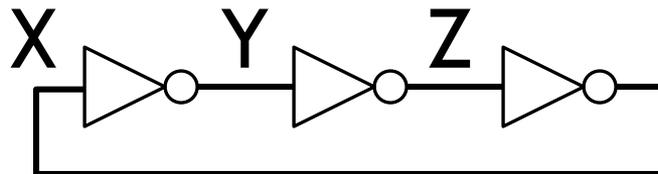


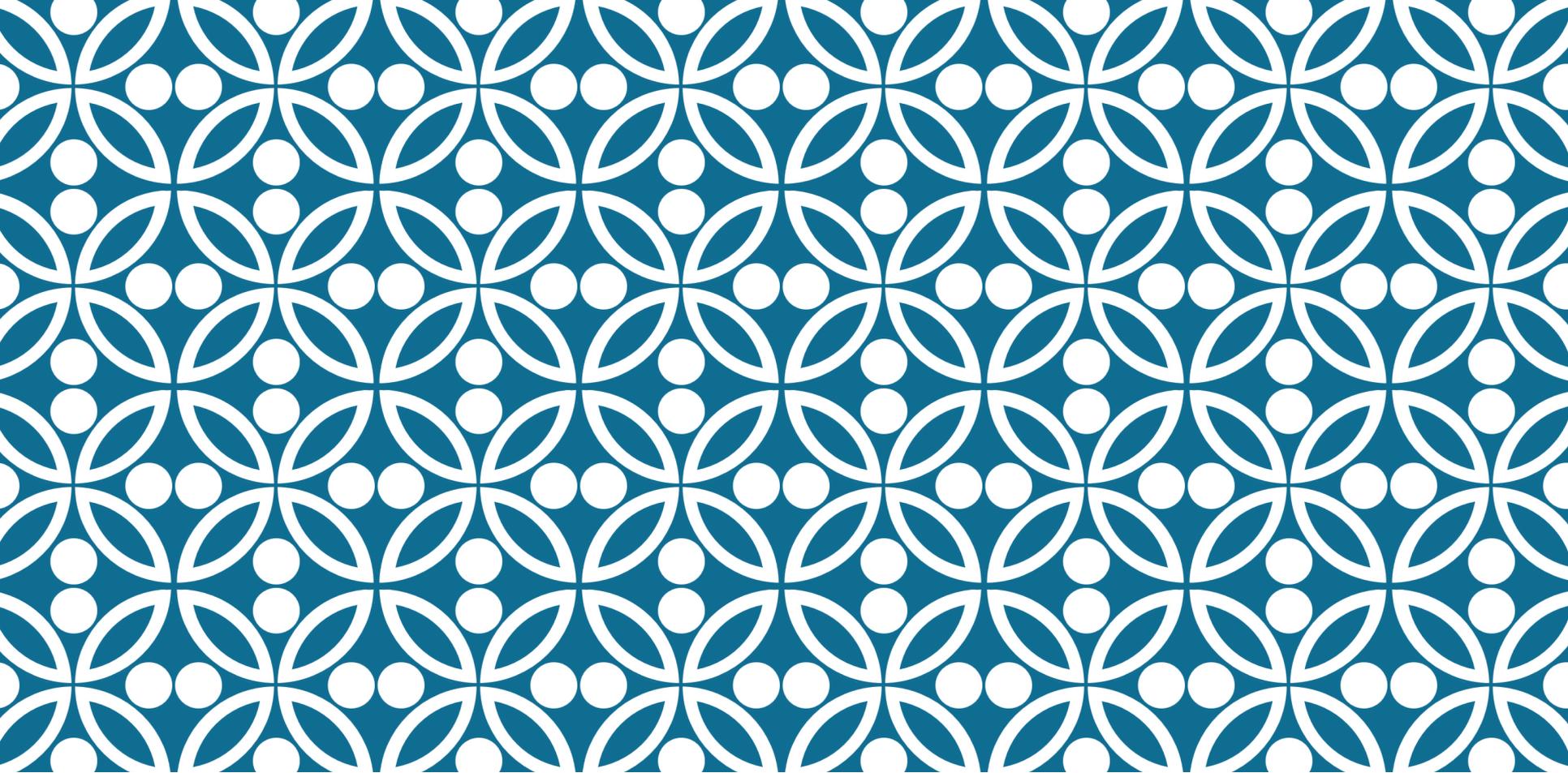
CIRCUITOS INSTÁVEIS

Analisando esse circuito observamos:

Supondo que $X=0$, então $Y=1$, $Z=0$ e $X=1$, o que é inconsistente com nossa suposição inicial.

Esse circuito não possui um estado estável, por isso é chamado de instável





CIRCUITOS SÍNCRONOS



CIRCUITOS SÍNCRONOS

Circuitos combinacionais não possuem nenhum caminho cíclico, onde a saída realimenta a entrada.

- Uma vez definida a entrada do circuito combinacional, o valor correto sempre será visto na saída.

Os circuitos sequenciais possuem caminhos cíclicos, ou seja, realimentação.

- Tais circuitos podem possuir problemas de instabilidade
- A análise desses circuitos demandam tempo e muitas pessoas brilhantes comentem enganos.

CIRCUITOS SÍNCRONOS

Para evitar esses problemas de circuitos sequenciais, os designers quebram as realimentações inserindo registradores (latches ou flip-flops) no caminho.

Assim, o circuito será transformado em uma coleção de lógica combinacional e registradores.

Se tais registradores forem governados pelo mesmo clock, estamos lidando com um circuito síncrono.

REGRAS PARA CIRCUITOS SEQUENCIAIS SÍNCRONOS

Todo elemento do circuito deve ser ou um registrador ou um circuito combinacional

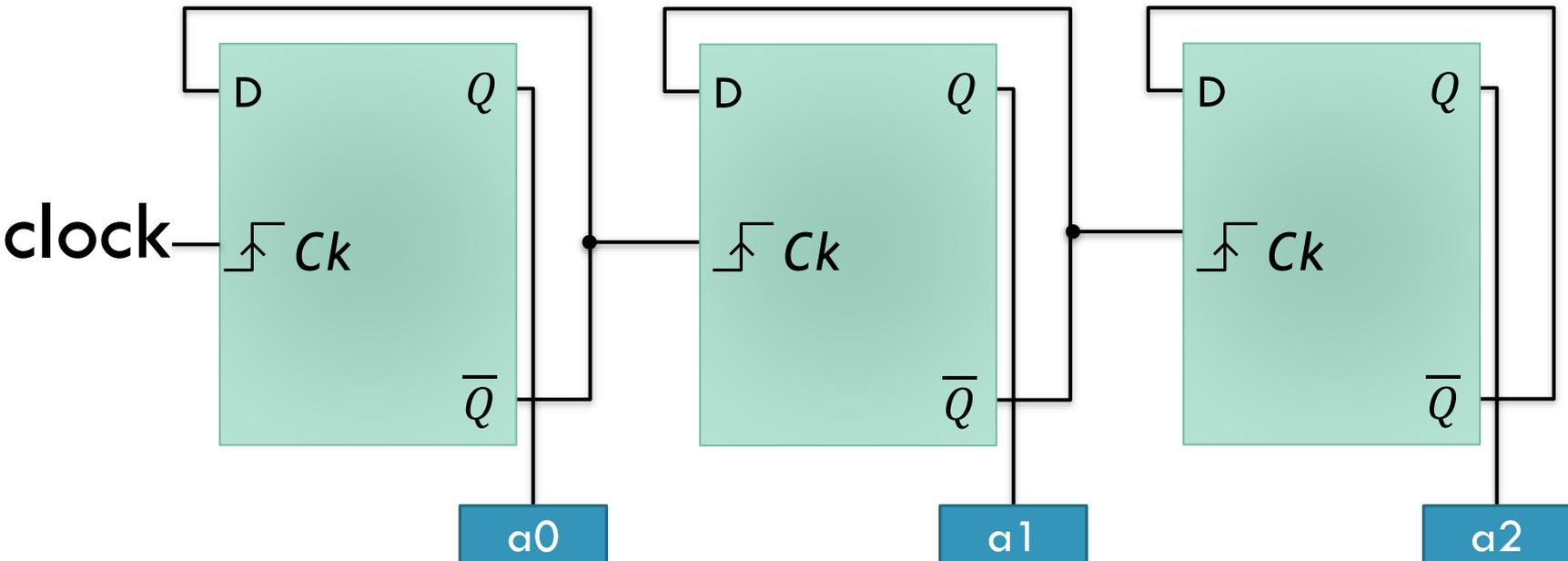
No mínimo um elemento do circuito deve ser um registrador

Todos registradores devem receber o mesmo sinal de clock

Todo caminho ciclico possui pelo menos um registrador

EXEMPLO

O circuito abaixo é síncrono ou assíncrono?



SÍNCRONO VS. ASSÍNCRONO

Em teoria circuitos assíncronos são mais genéricos que os síncronos, uma vez que eles não estão limitados por registradores ligados ao clock.

Da mesma maneira que circuitos analógicos (podem usar qualquer tensão) são mais genéricos que os circuitos lógicos.

Entretanto, circuitos lógicos síncronos se mostraram ao longo da história, muito mais fáceis de projetar e usar que os circuitos assíncronos.

Embora houveram décadas de estudo sobre circuitos assíncronos, praticamente todos os sistemas digitais são síncronos.

SÍNCRONO VS. ASSÍNCRONO

Claro que circuitos assíncronos são ocasionalmente necessários, quando estamos comunicando dois sistemas com clocks diferentes ou recebendo entradas em tempos arbitrários.

Da mesma maneira que sistemas analógicos ainda são necessários para conversar com o mundo real de tensões contínuas

CONTADOR SÍNCRONO COM FLIP-FLOPS DO TIPO D

