Computer Architecture

Lecture + Tutorial 05

Extension of Latches to Flip-Flops
Asynchronous and Synchronous Logic Circuits
Sequential Logic Circuits

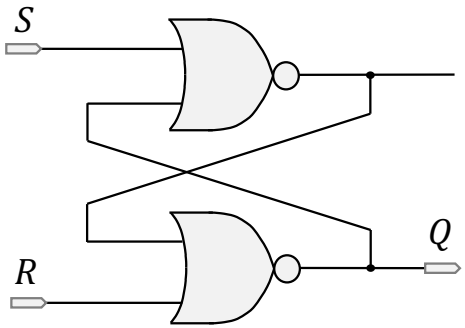
Artem Burmyakov, Alexander Tormasov

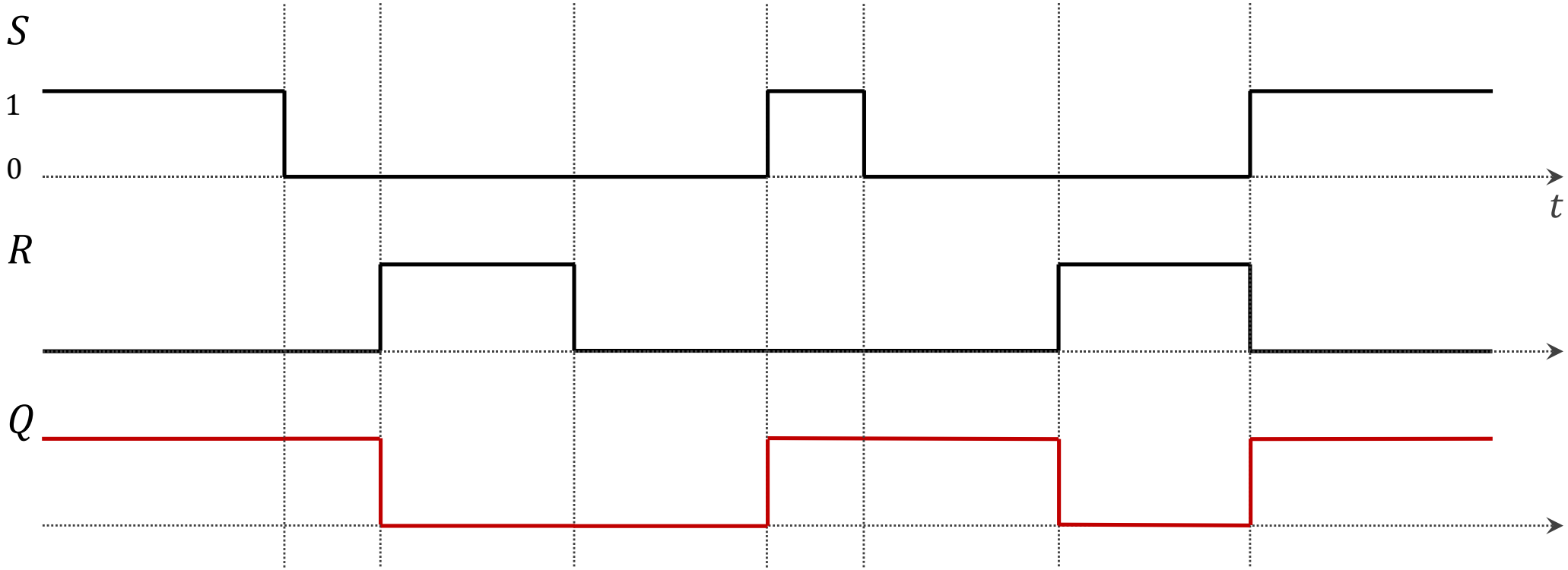
September 23, 2021

INNOPOLIS UNIVERSITY

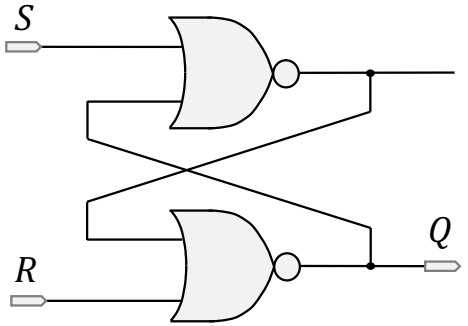
The Recap of a Transparent S/R Latch

(Transparent = the change of inputs triggers immediately the update of the output)

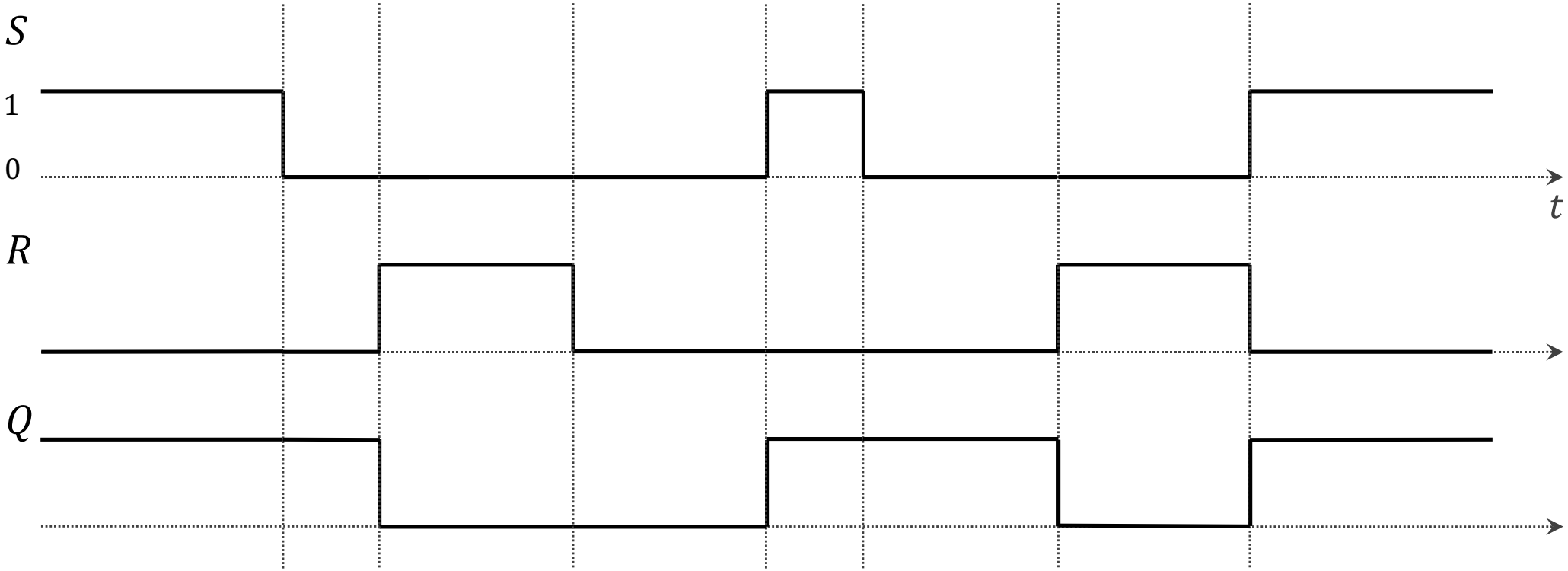




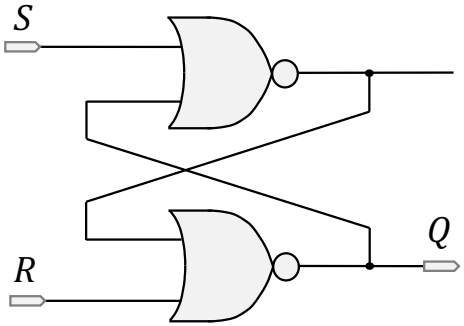
The Recap of a Transparent S/R Latch



An idealistic case (no propagation overhead considered):

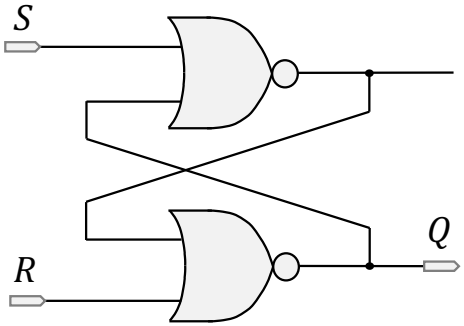


The Recap of a Transparent S/R Latch



Values for inputs S and R are produced at these times
(somewhere in neighbouring subcircuits)

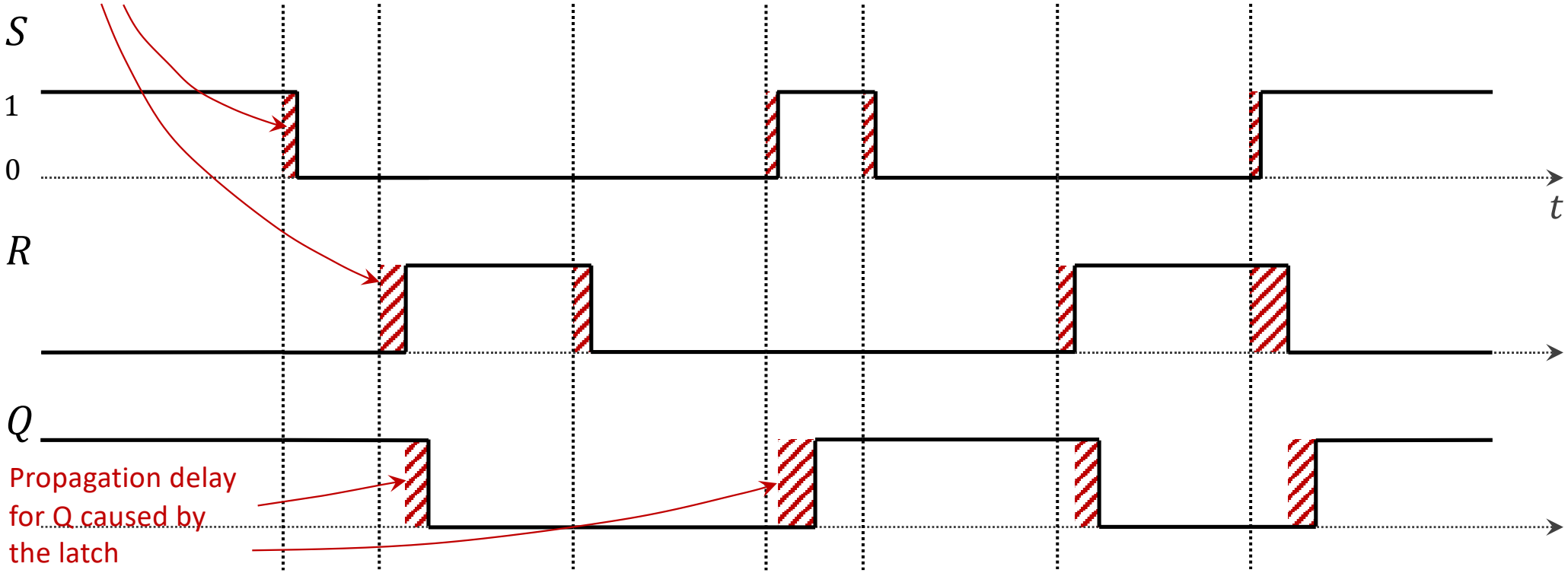
The Recap of a Transparent S/R Latch



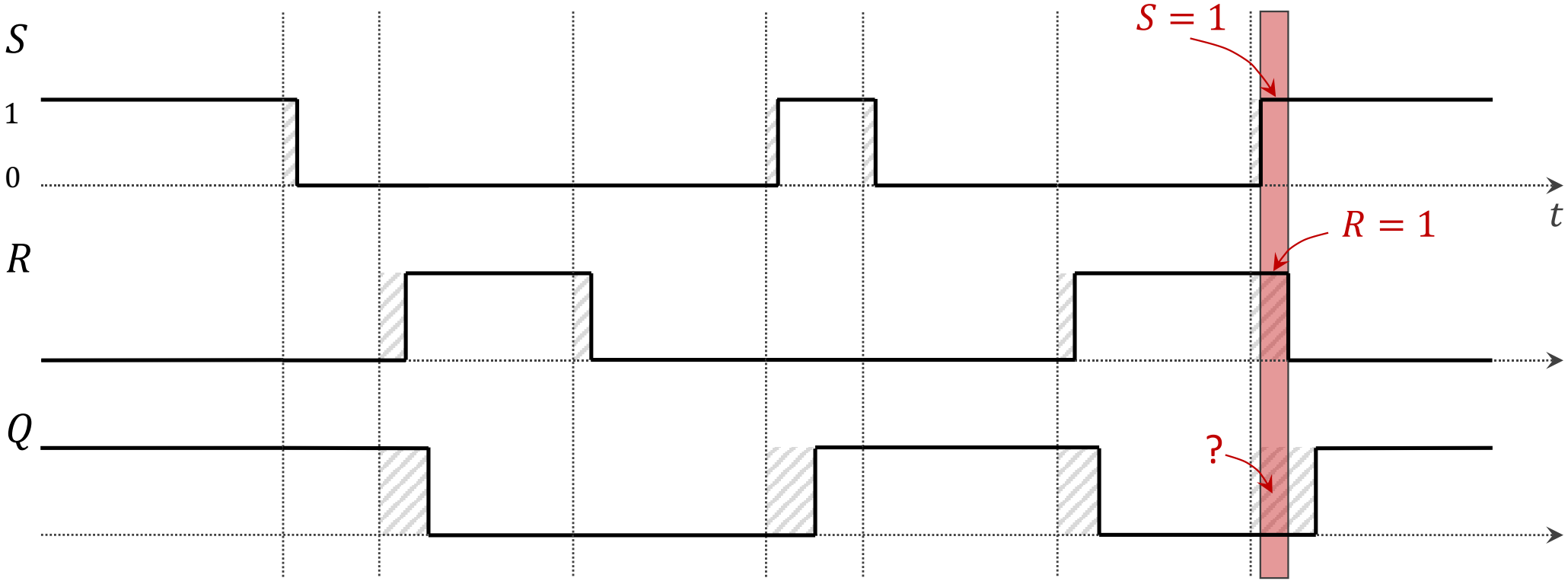
Values for inputs S and R are produced at these times (somewhere in neighbouring subcircuits)

...

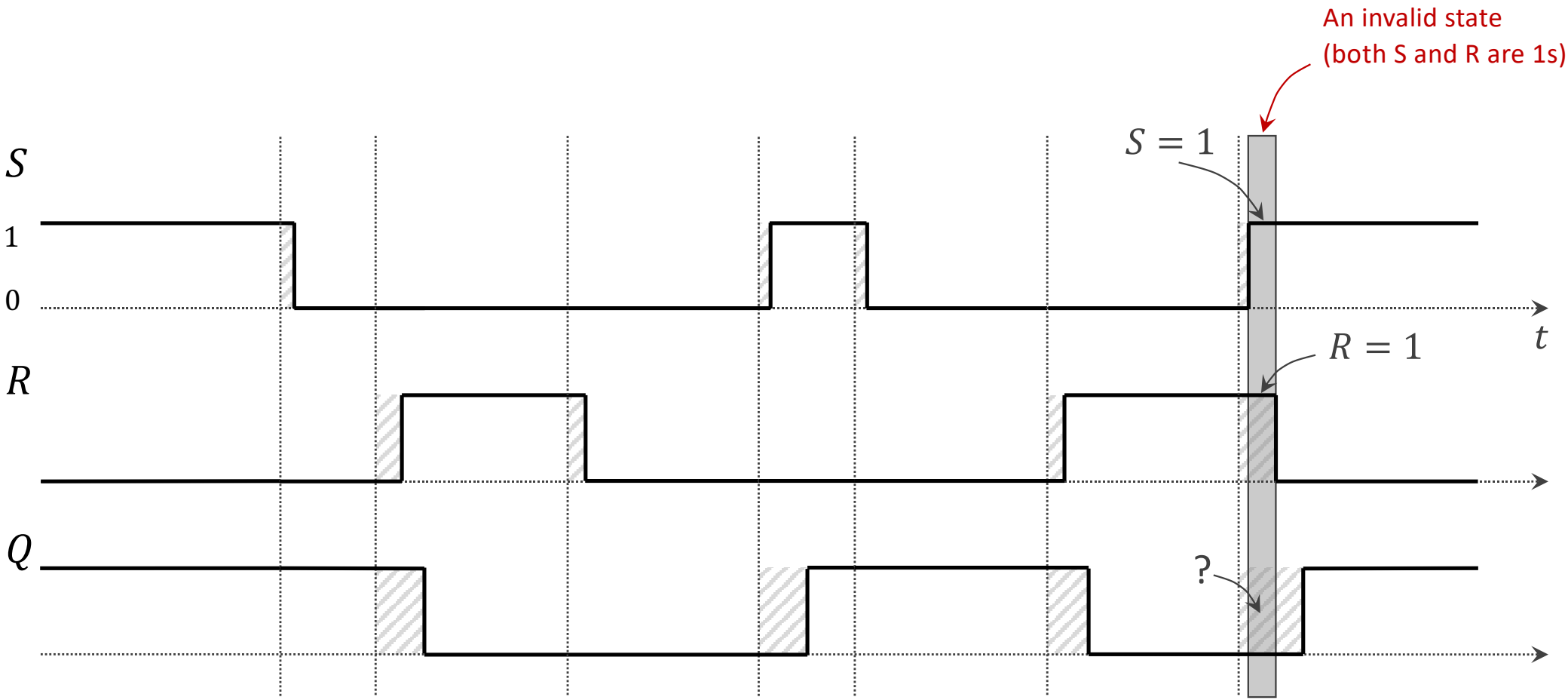
Propagation delays for S and R, to reach the latch (always varying)



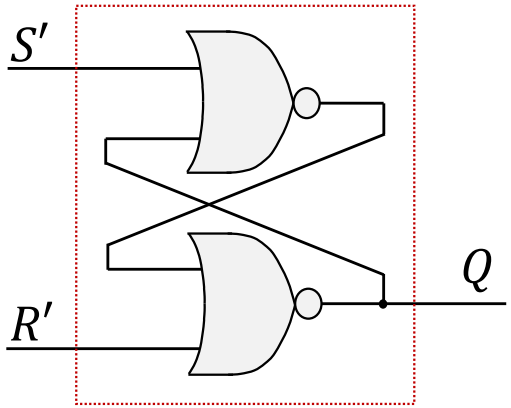
Propagation delay for Q caused by the latch


S
1
0
S = 1
t
R
R = 1
Q
?

Various problems might be caused by unpredicted propagation delays, such as an invalid state for a transparent S/R latch

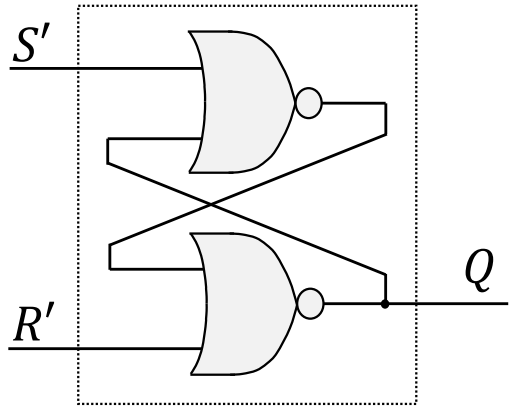


Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



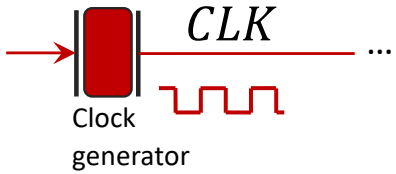
S/R Latch

Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



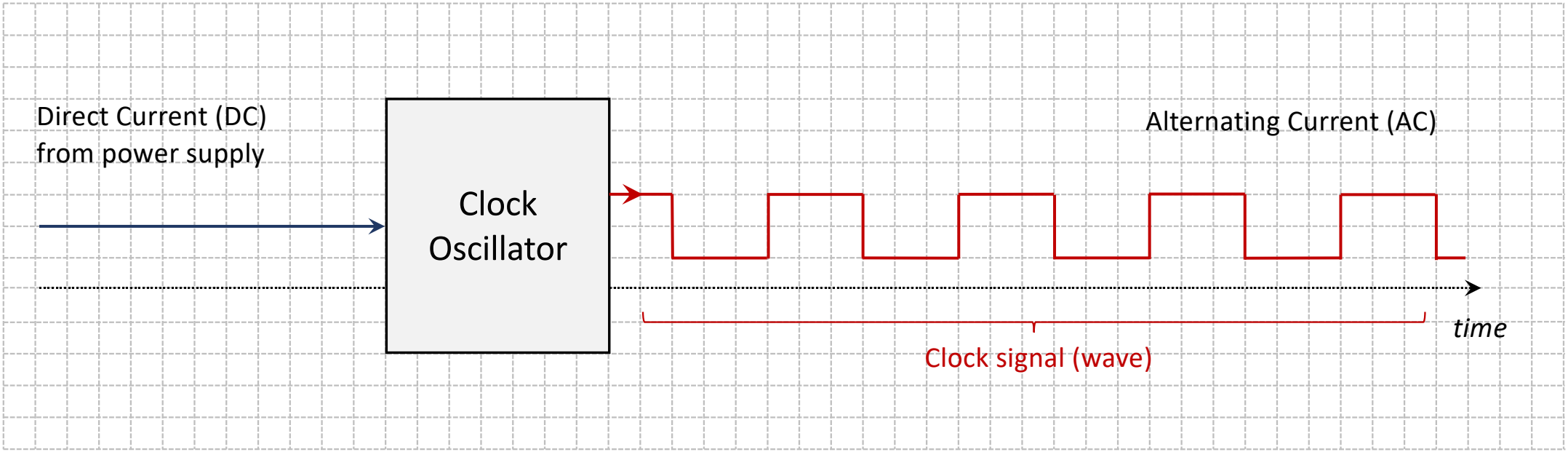
S/R Latch

Step 1:
The inclusion of a clock generator



Clock Generator (Oscillator)

An electronic device, producing a periodic clock signal

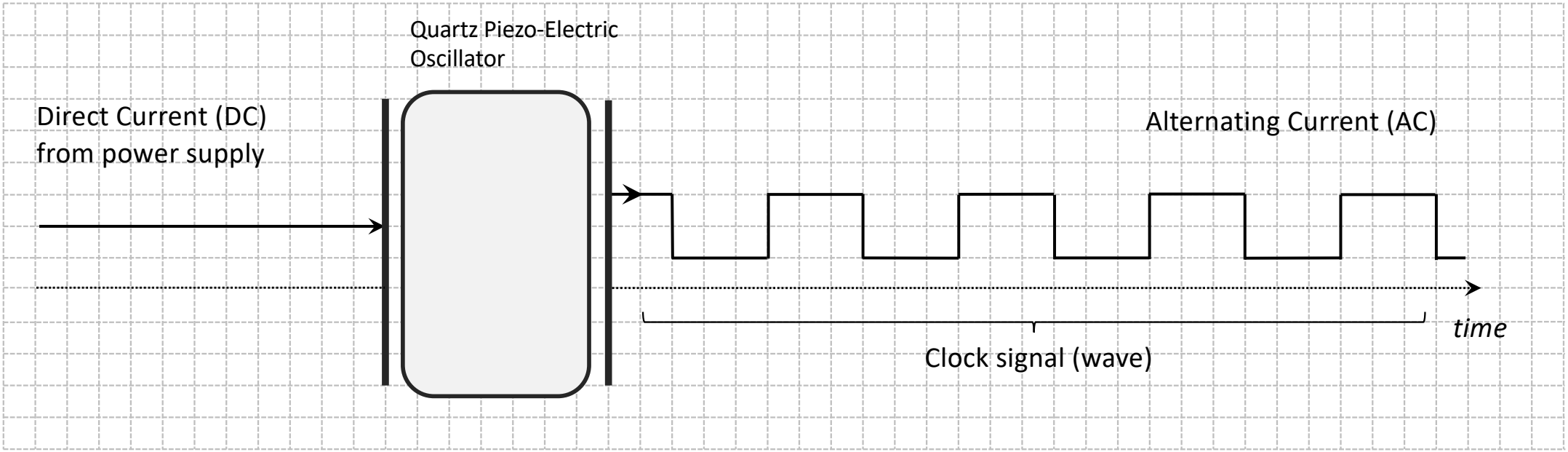


Clock oscillator – a physical device, that converts direct current (DC) from a power supply into an alternating periodic signal

Oscillation – a repetitive variation, typically in time, of some measure between two or several values

Clock Generator (Oscillator)

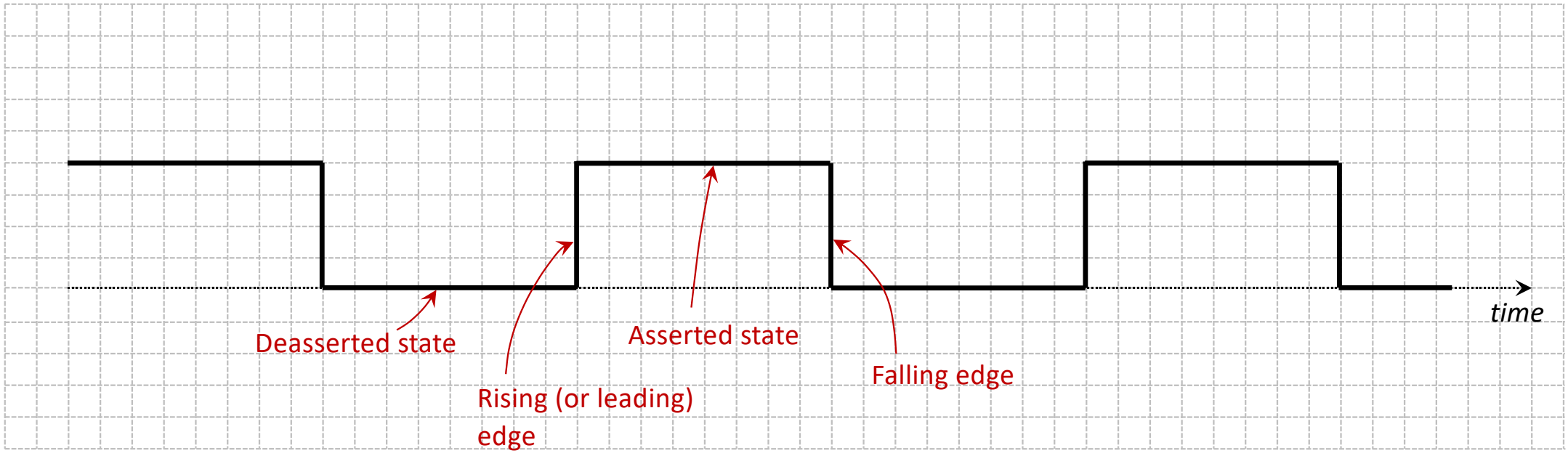
An electronic device, producing a periodic clock signal



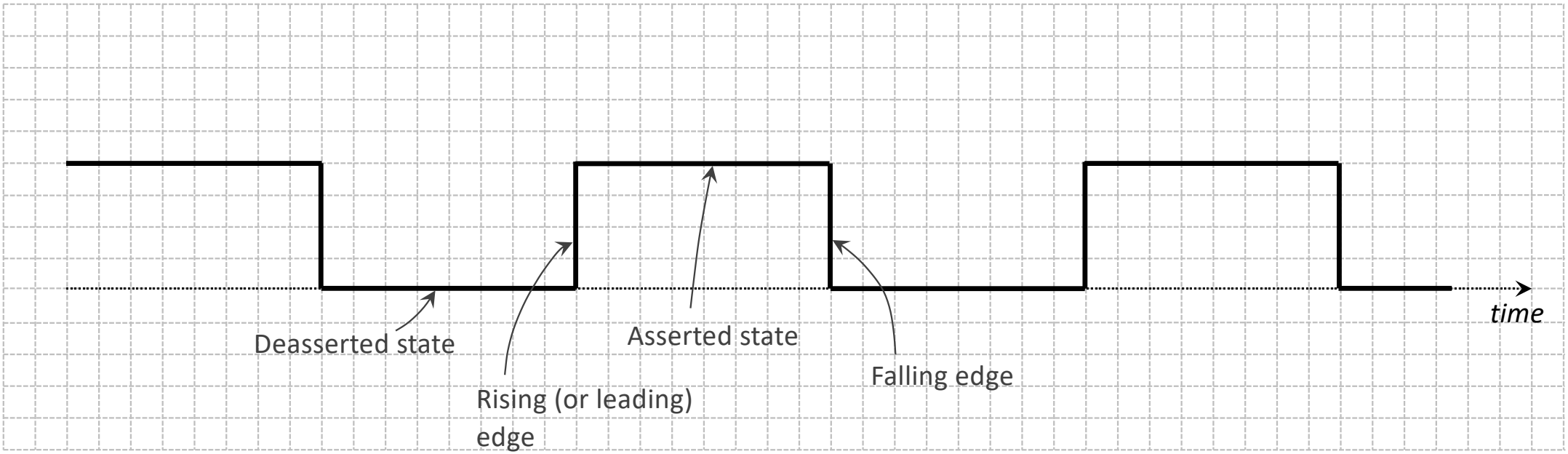
Clock oscillator – a physical device, that converts direct current (DC) from a power supply into an alternating periodic signal

Oscillation – a repetitive variation, typically in time, of some measure between two or several values

Clock Signal Terminology



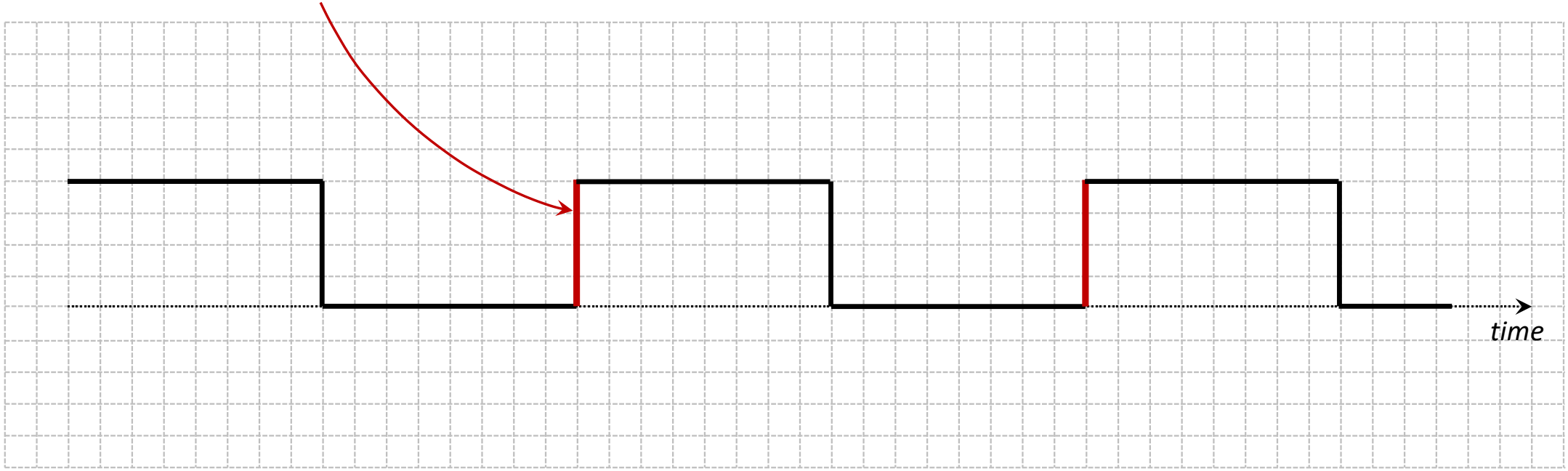
Edge-Triggered and Level-Triggered Timing Methodologies



Edge-Triggered and Level-Triggered Timing Methodologies

Edge-Triggered Circuit:

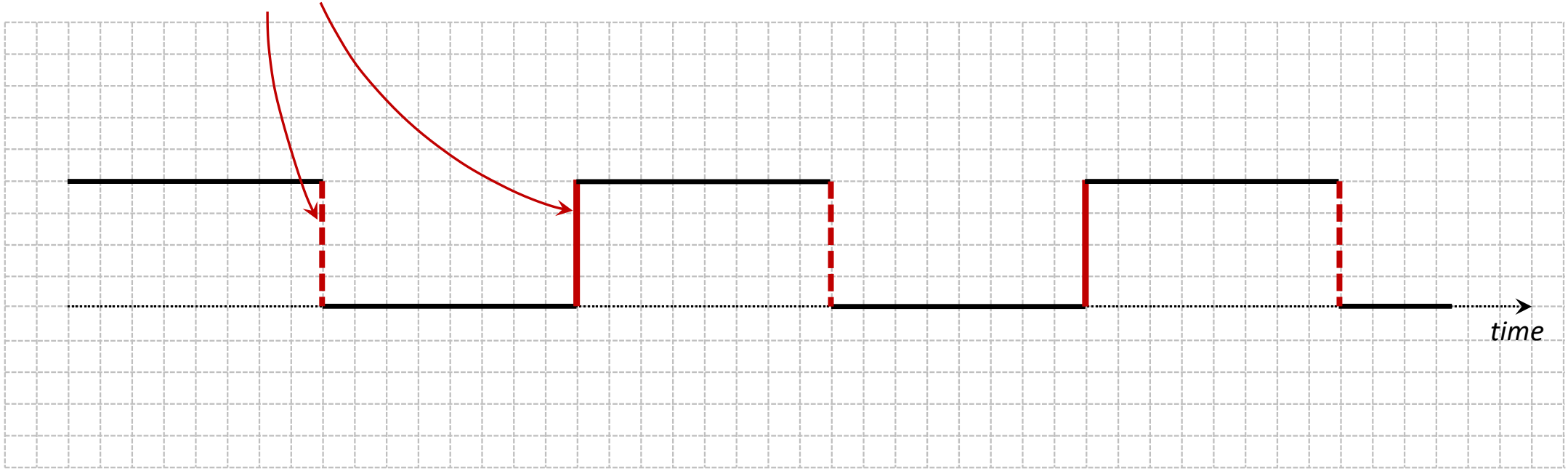
a circuit is activated either on a raising or a falling edge



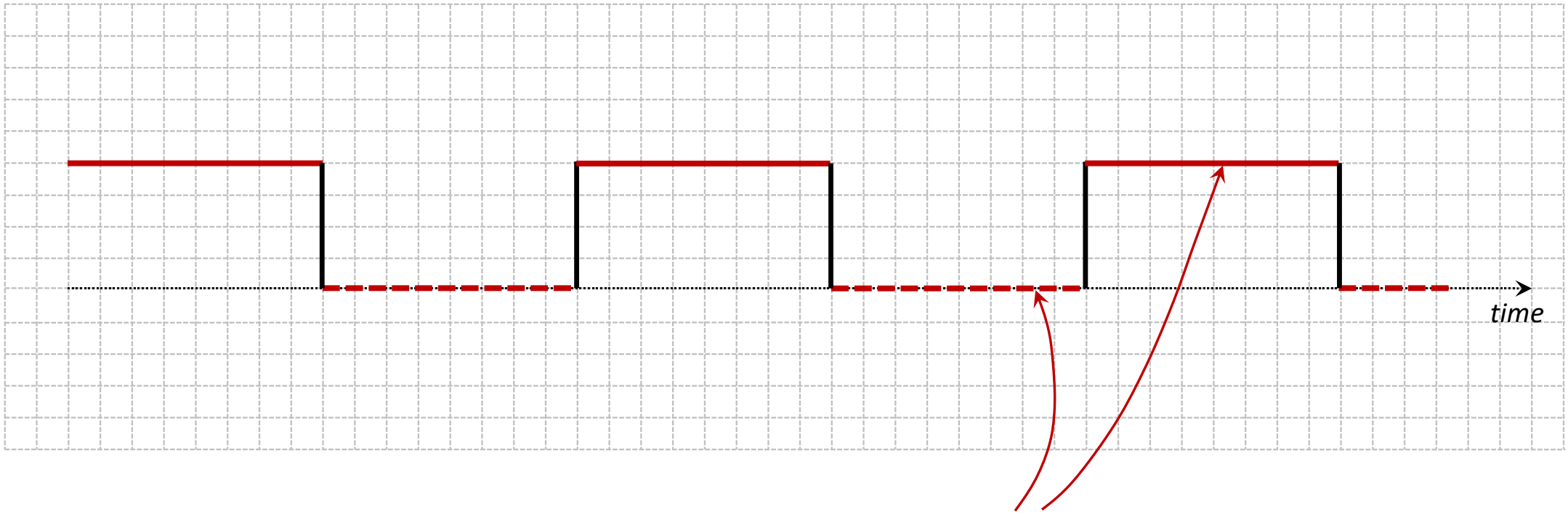
Edge-Triggered and Level-Triggered Timing Methodologies

Edge-Triggered Circuit:

a circuit is activated either on a raising or a falling edge



Edge-Triggered and Level-Triggered Timing Methodologies



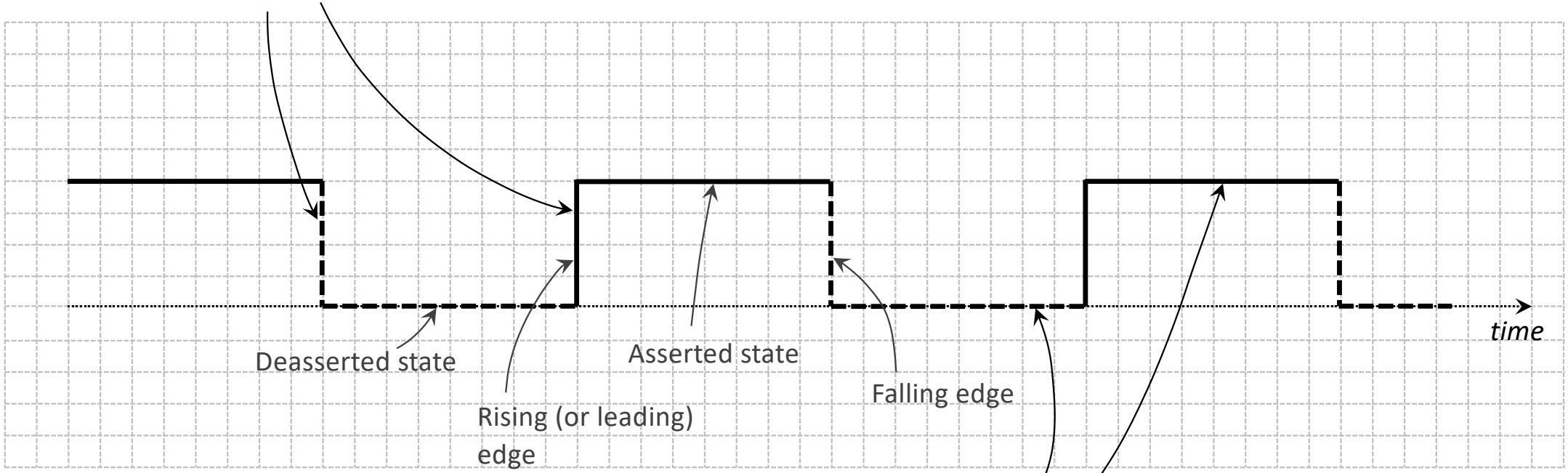
Level-Triggered Circuit:

a circuit is activated either when clock signal is asserted, or deasserted

Edge-Triggered and Level-Triggered Timing Methodologies

Edge-Triggered:

a circuit is activated either on a raising or a falling edge

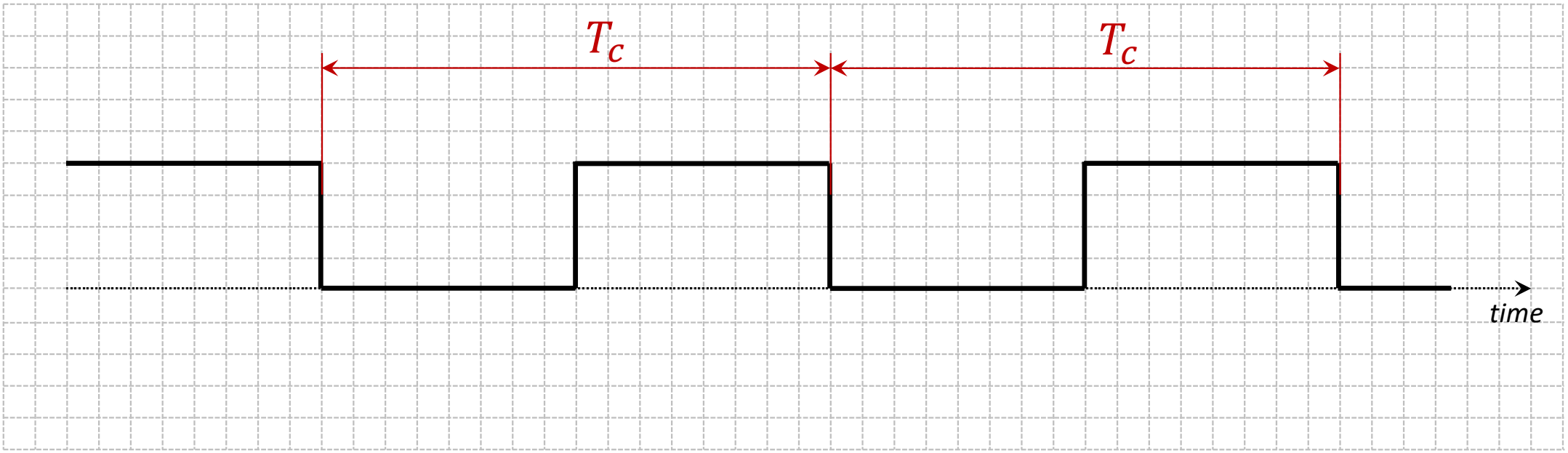


Level-Triggered:

a circuit is activated either when clock signal is asserted, or deasserted

Characteristics of a Rectangular Clock Signal

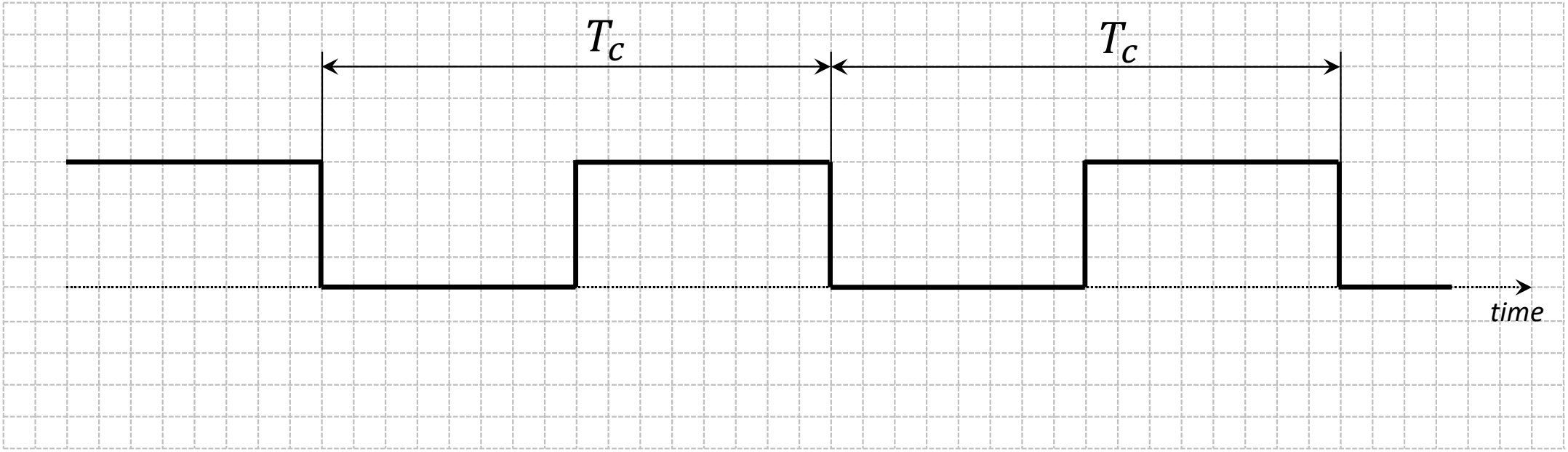
T_c - clock cycle time, or period, or cycle length



Characteristics of a Rectangular Clock Signal

T_c - clock cycle time, or period, or cycle length

$F_c = {}^{1}/_{T_c}$ - clock frequency

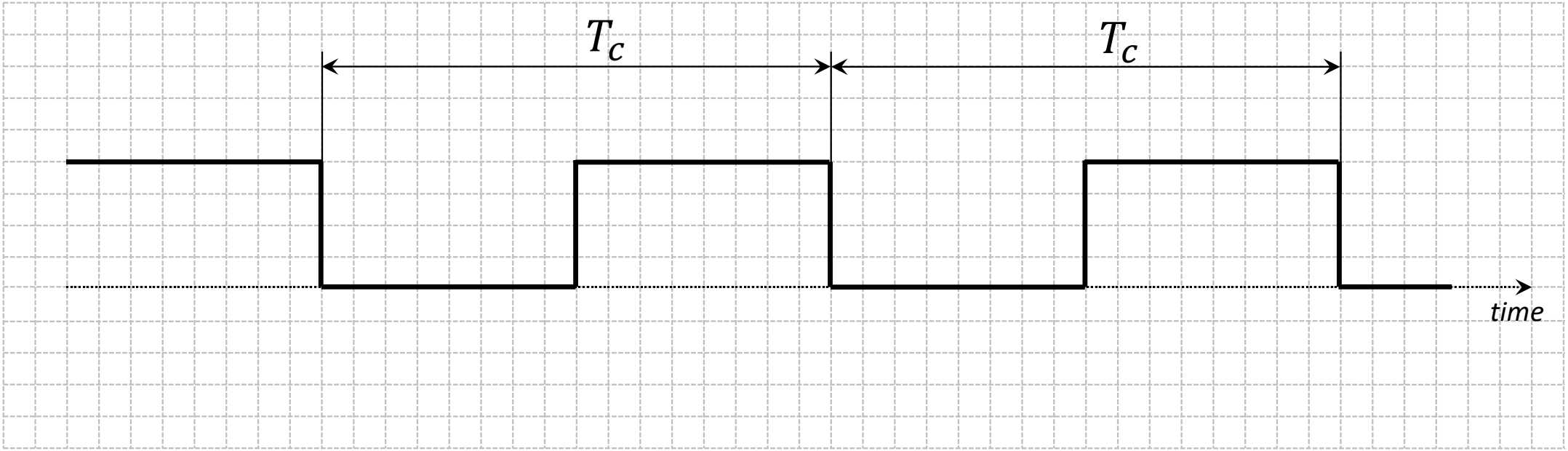


Clock frequency sets the tempo for the processor to execute instructions

Characteristics of a Rectangular Clock Signal

T_c - clock cycle time, or period, or cycle length

$F_c = {}^{1}/_{T_c}$ - clock frequency



Clock frequency sets the tempo for the processor to execute instructions

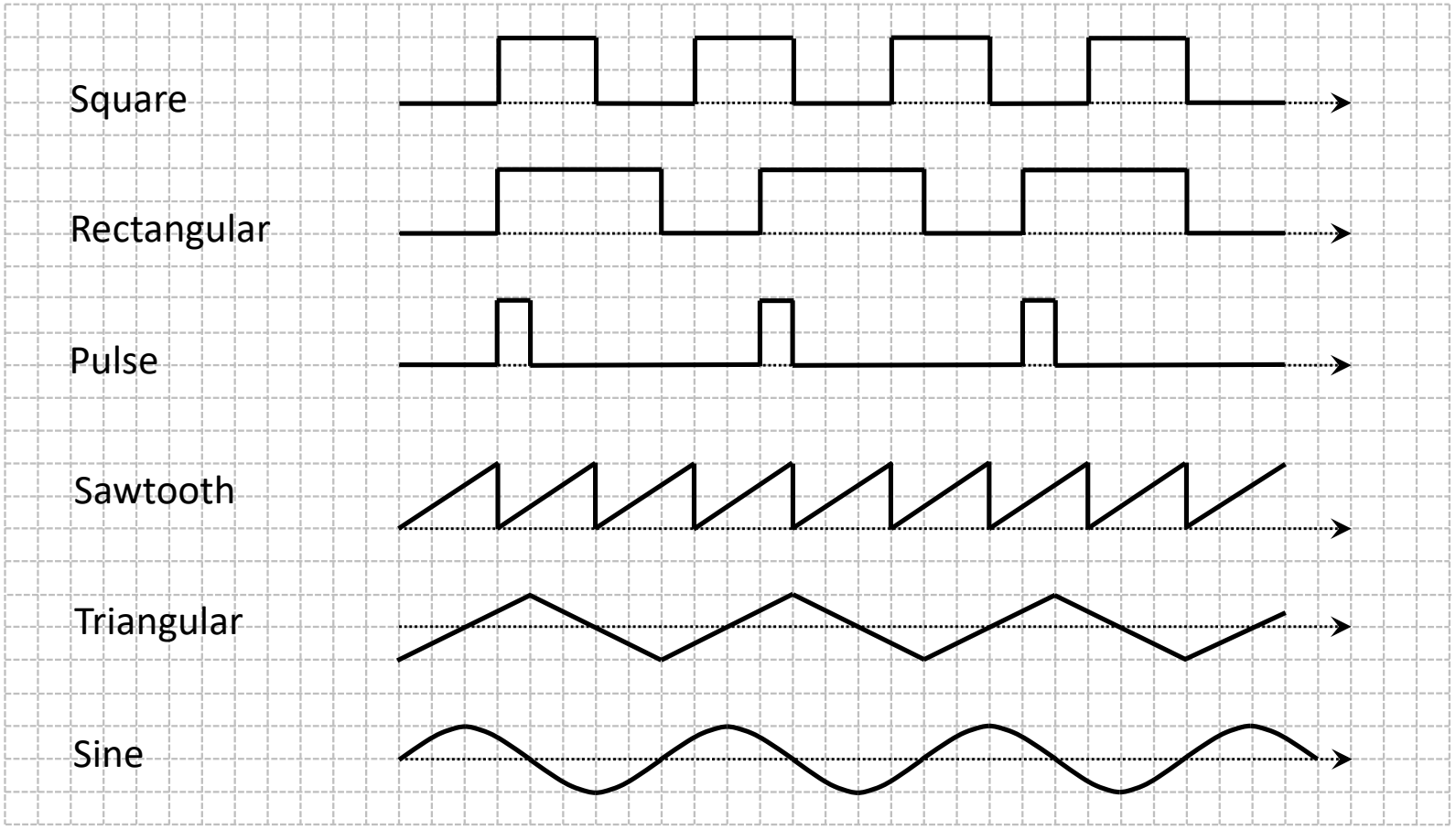
Common characteristicts for modern general-purpose CPUs:

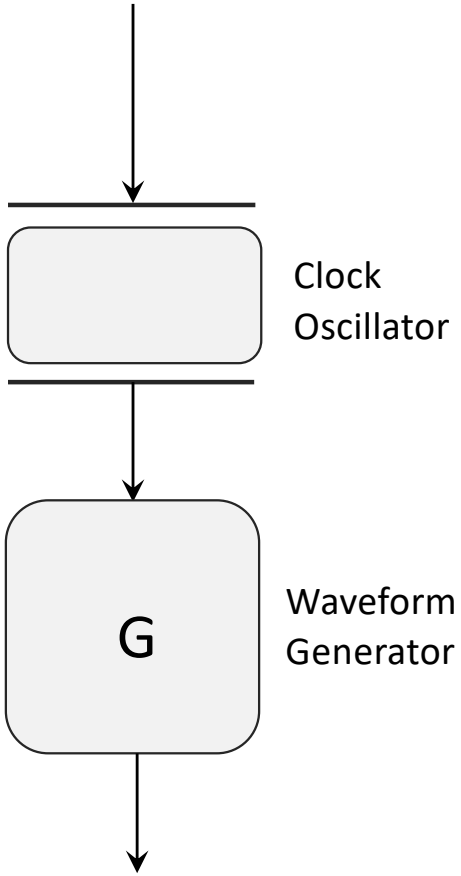
T_c - a few nanoseconds, $\sim 10^{-9}$ sec

F_c - a few gigahertz, 2-3 GHz

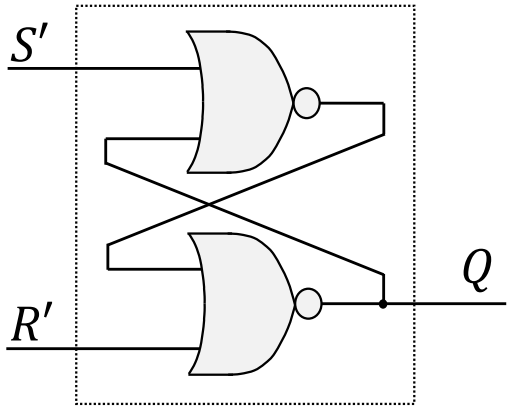
Clock Signal Variations

(A waveform generator is used to control the form of a clock wave signal)



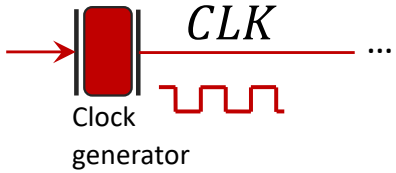


Transformation of an Asynchronous Latch into a Synchronous Flip-Flop

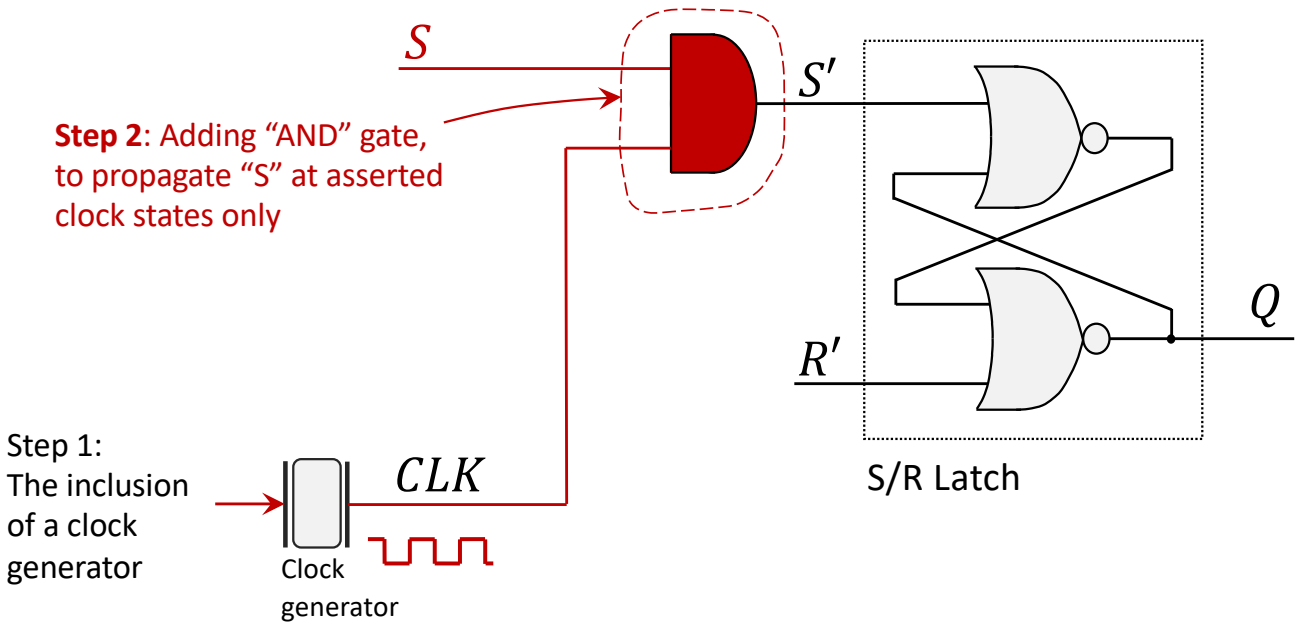


S/R Latch

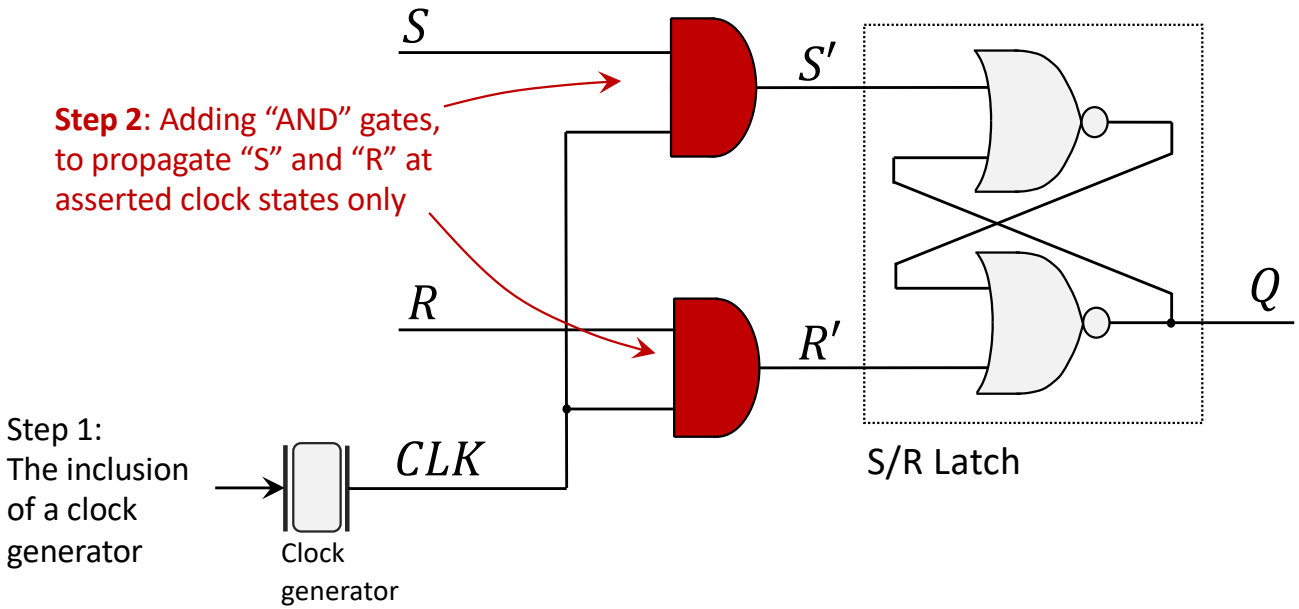
Step 1:
The inclusion of a clock generator



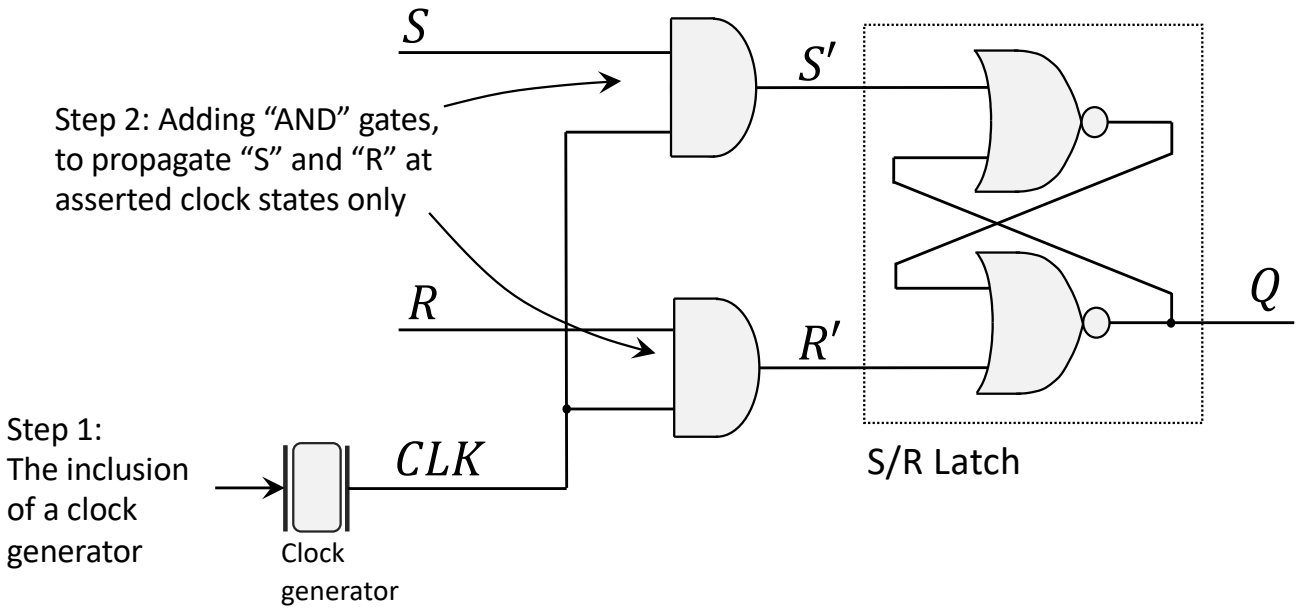
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



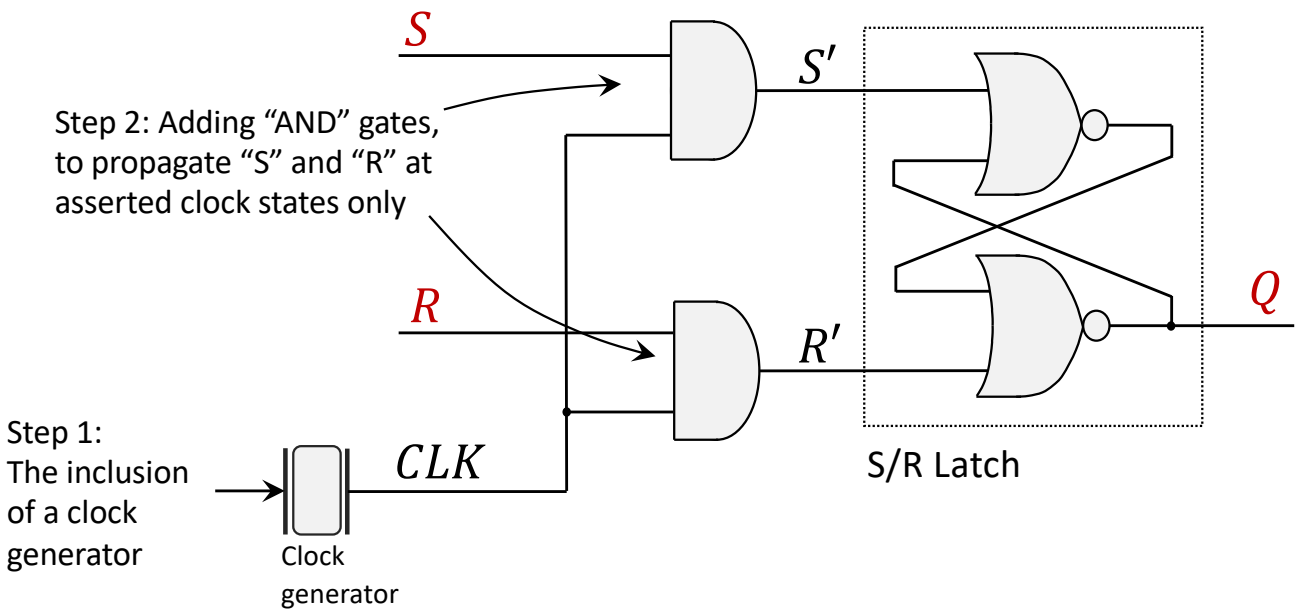
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



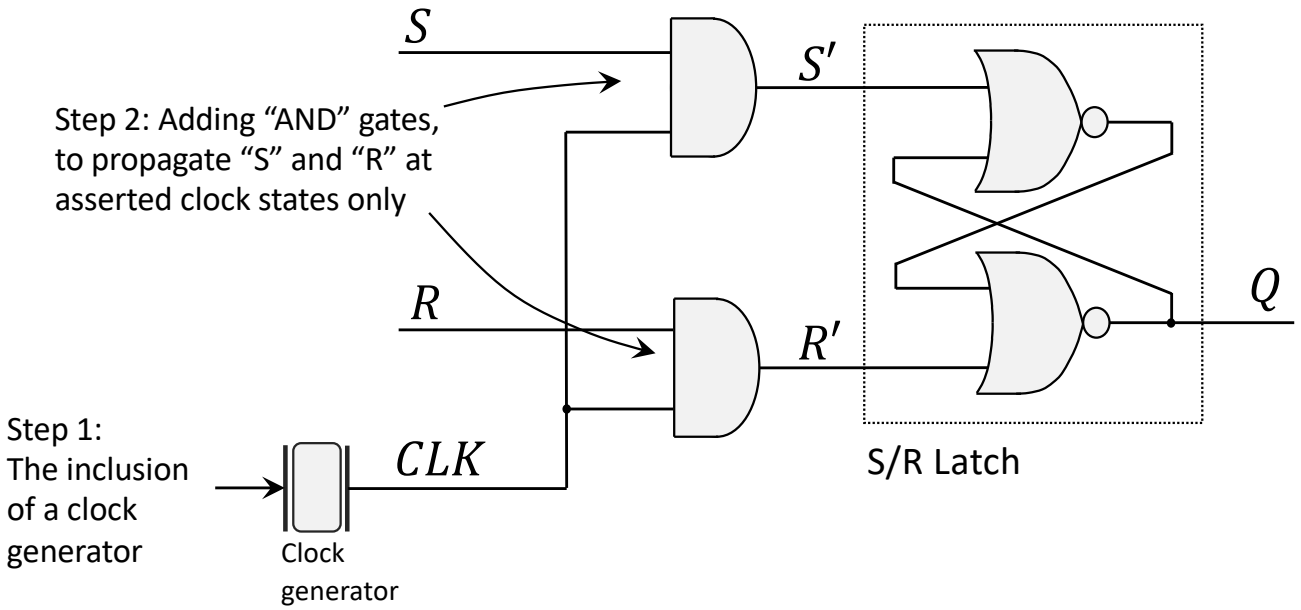
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



Observation:

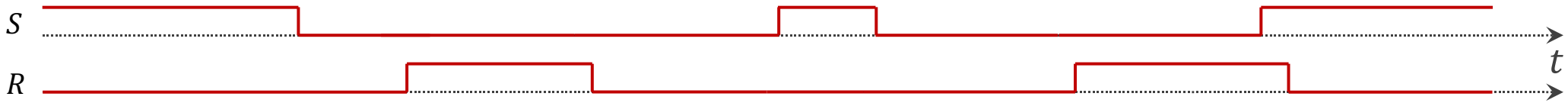
Inputs “S” and “R” make effect on “Q” only when clock is asserted

Transformation of an Asynchronous Latch into a Synchronous Flip-Flop

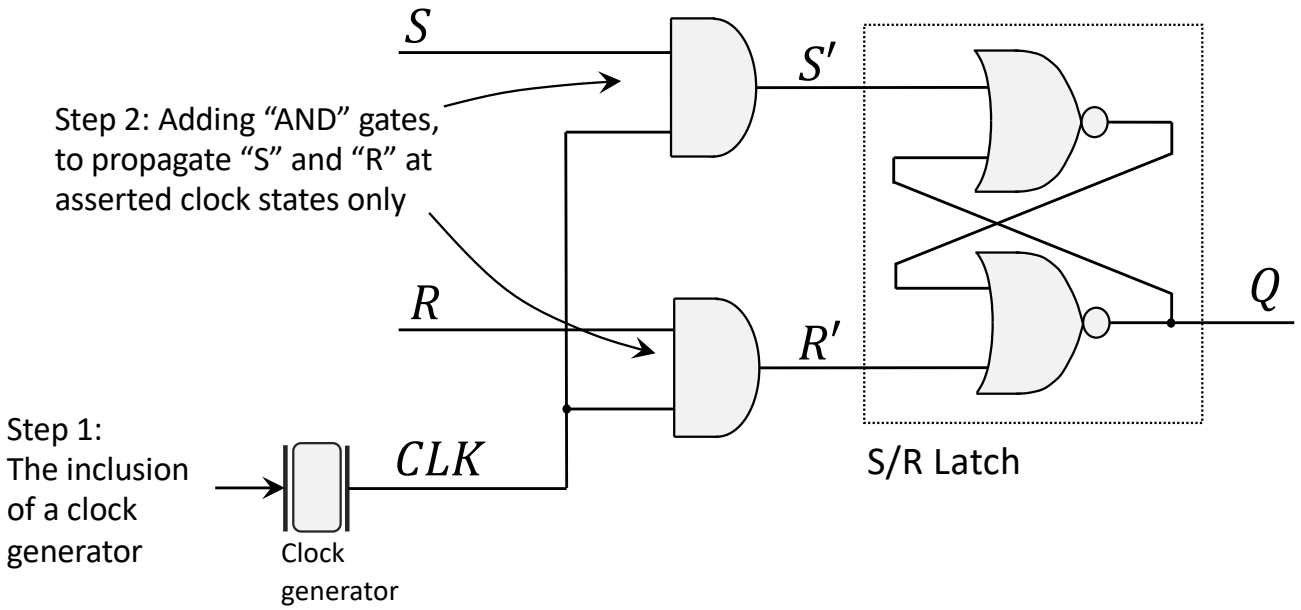


Observation:

Inputs “S” and “R” make effect on “Q” only when clock is asserted

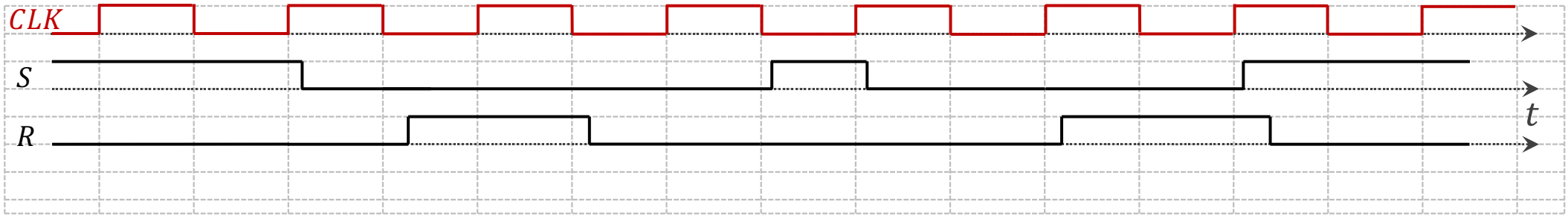


Transformation of an Asynchronous Latch into a Synchronous Flip-Flop

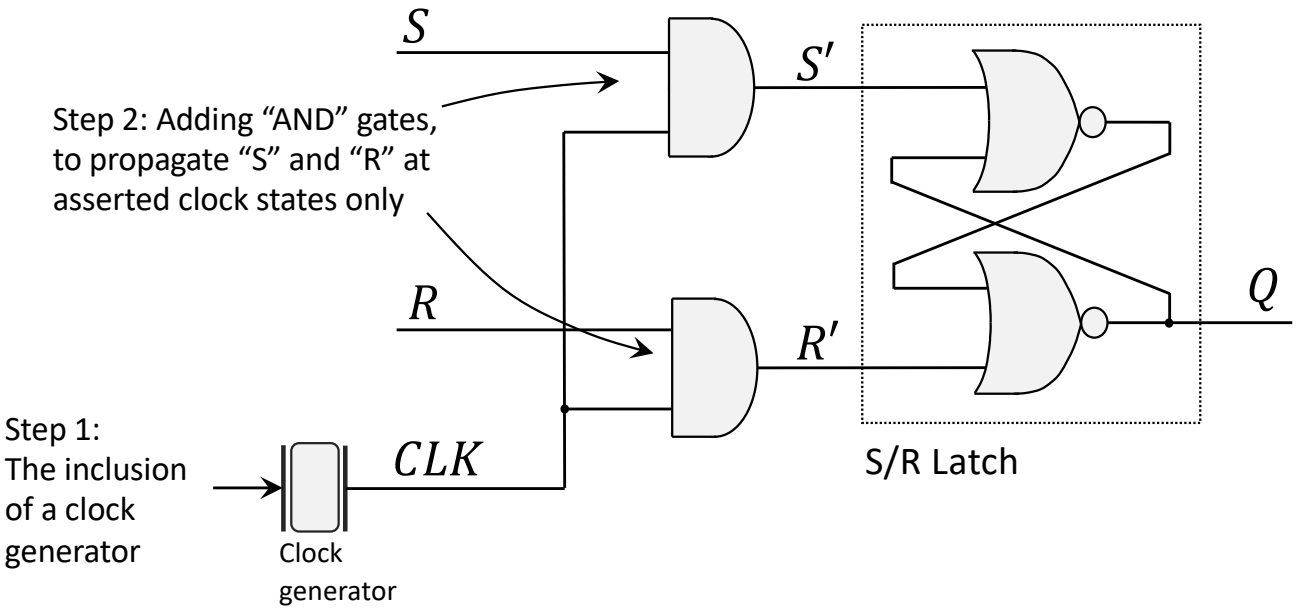


Observation:

Inputs “S” and “R” make effect on “Q” only when clock is asserted

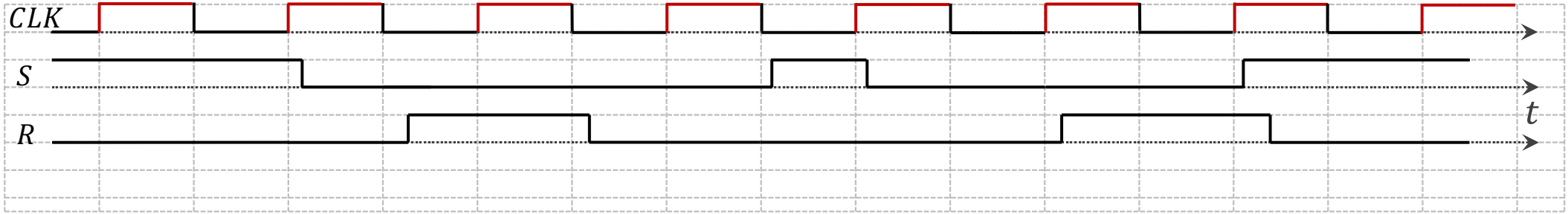


Transformation of an Asynchronous Latch into a Synchronous Flip-Flop

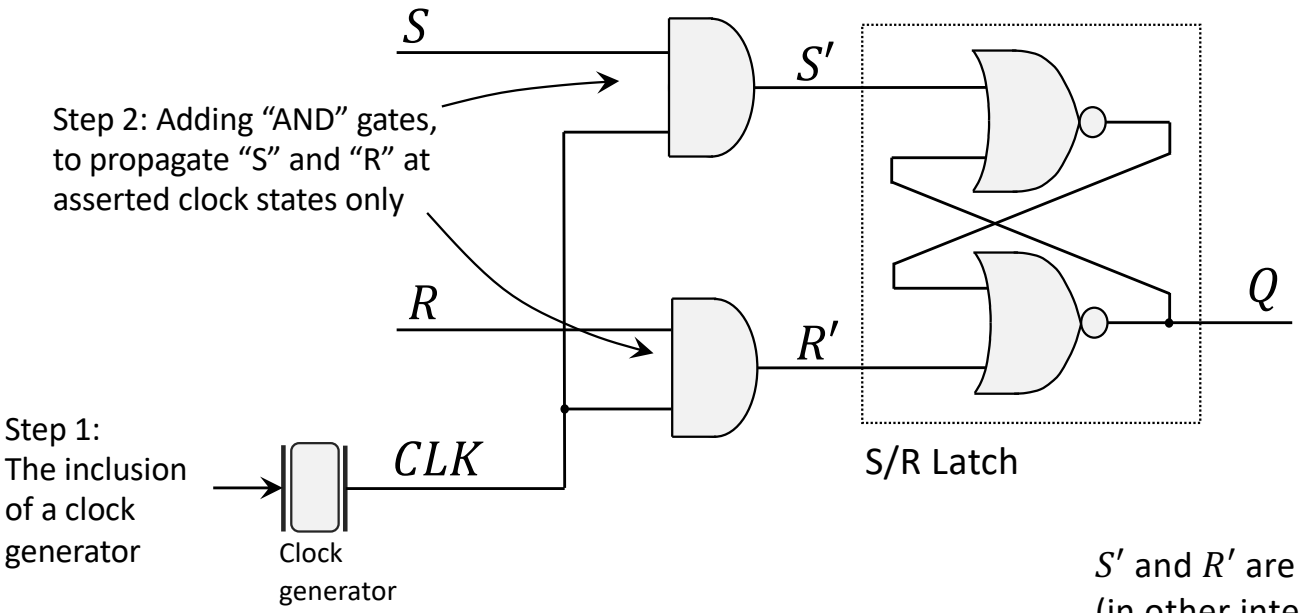


Observation:

Inputs “S” and “R” make effect on “Q” only when clock is asserted

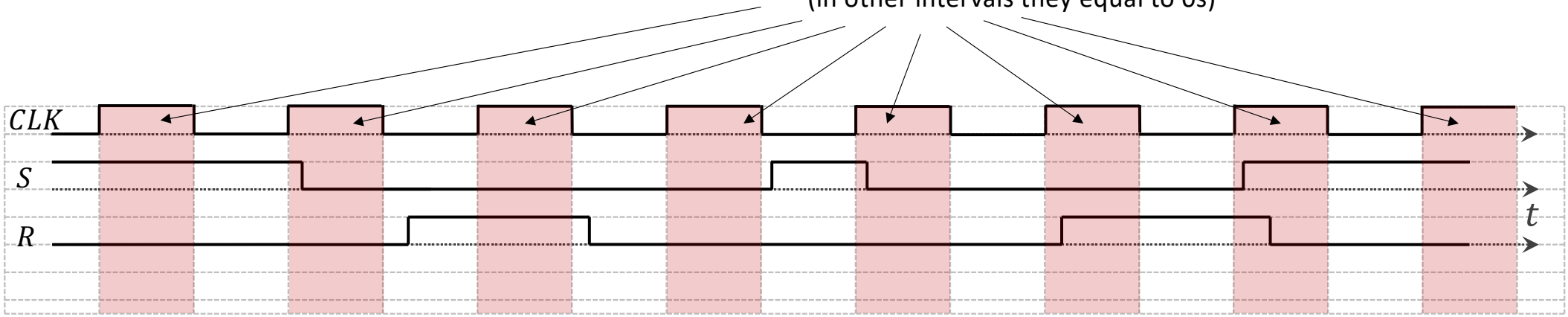


Transformation of an Asynchronous Latch into a Synchronous Flip-Flop

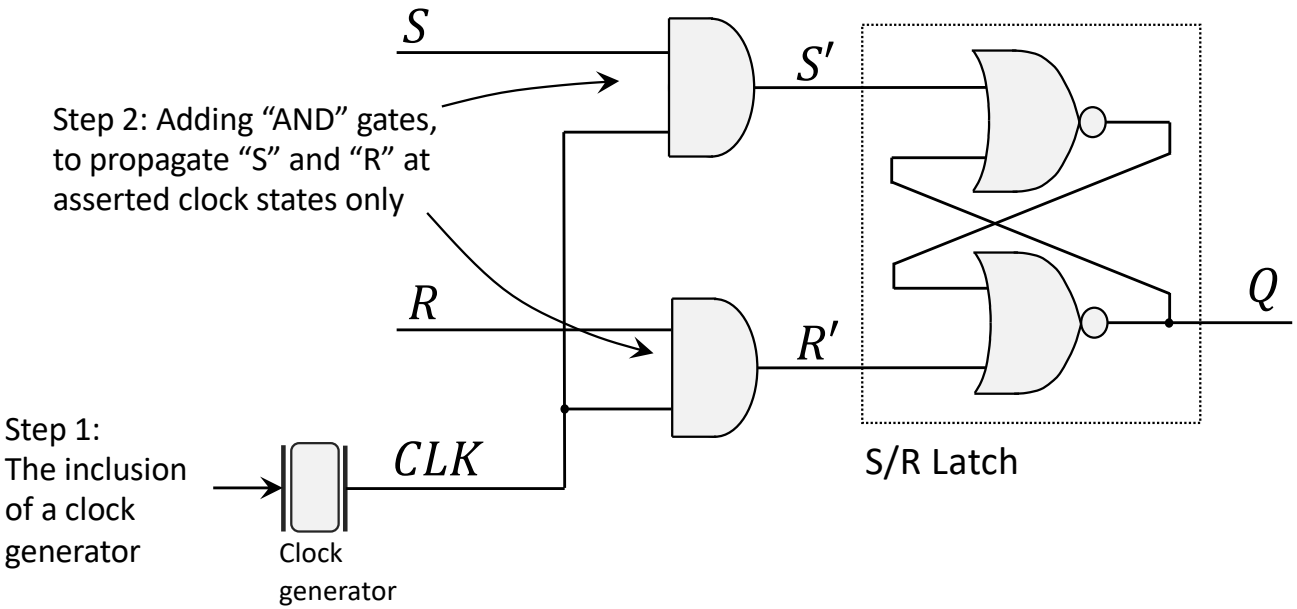


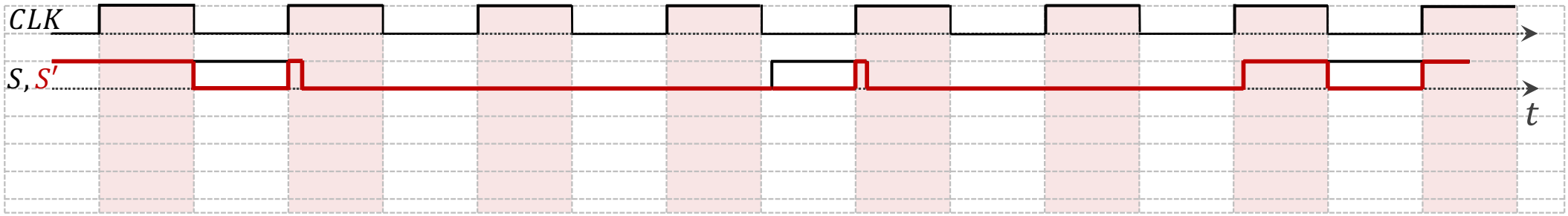
Observation:

Inputs "S" and "R" make effect on "Q" only when clock is asserted

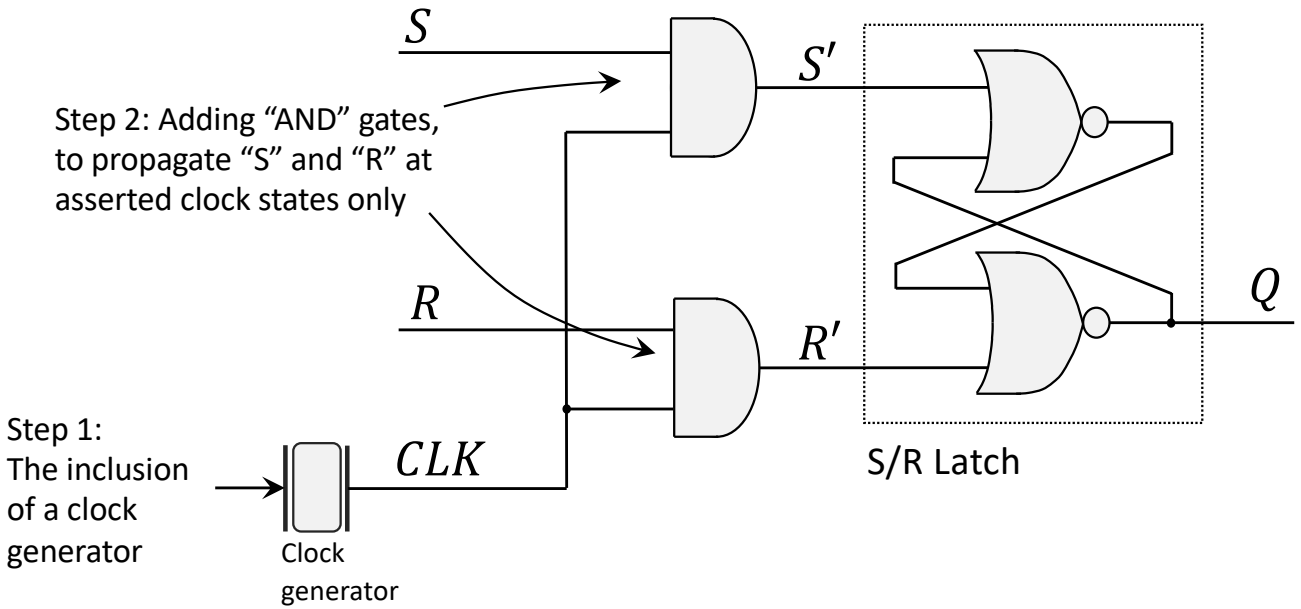


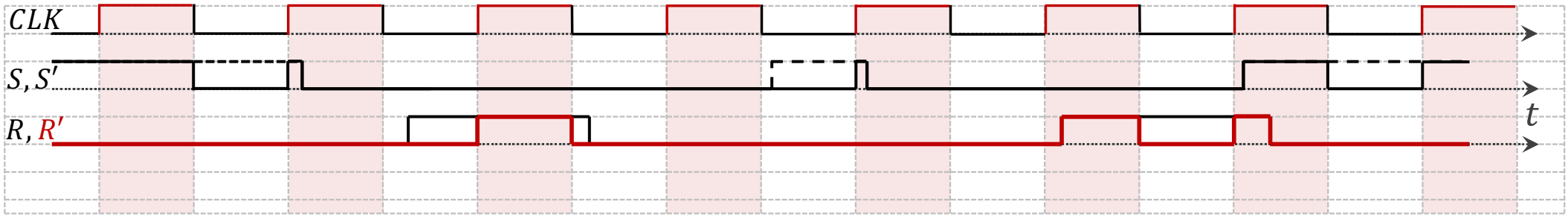
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



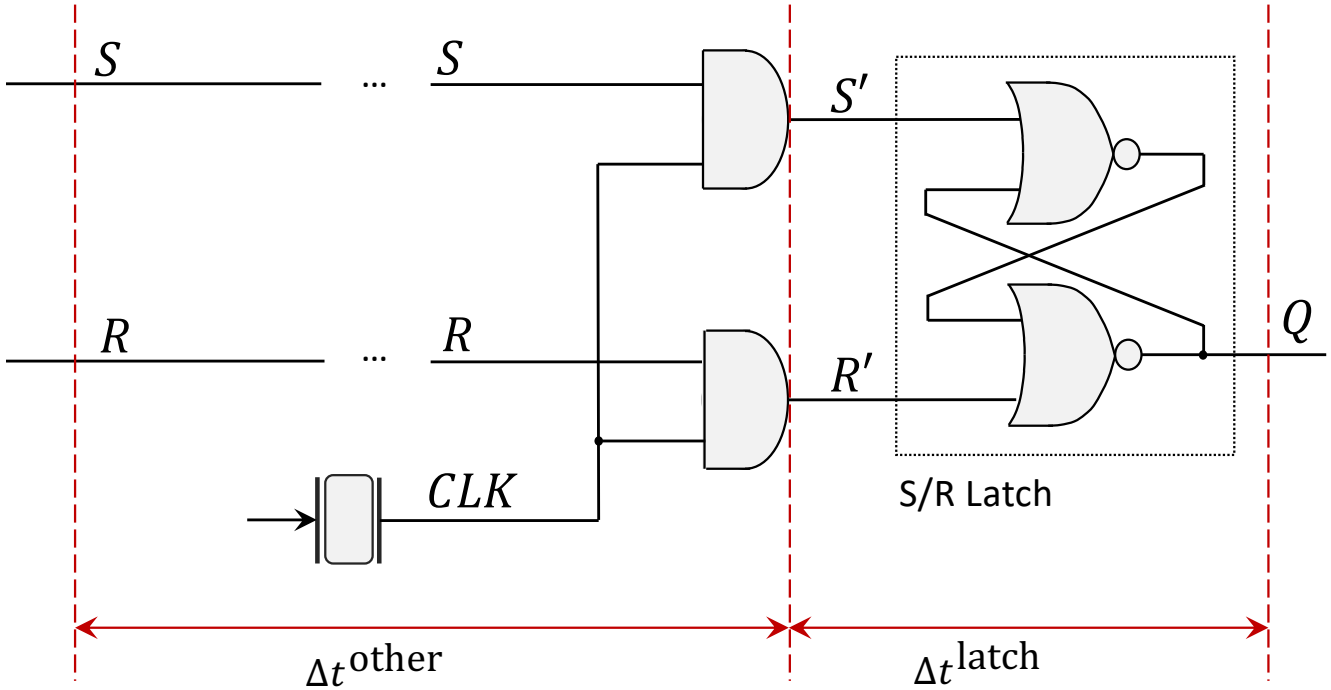


Transformation of an Asynchronous Latch into a Synchronous Flip-Flop





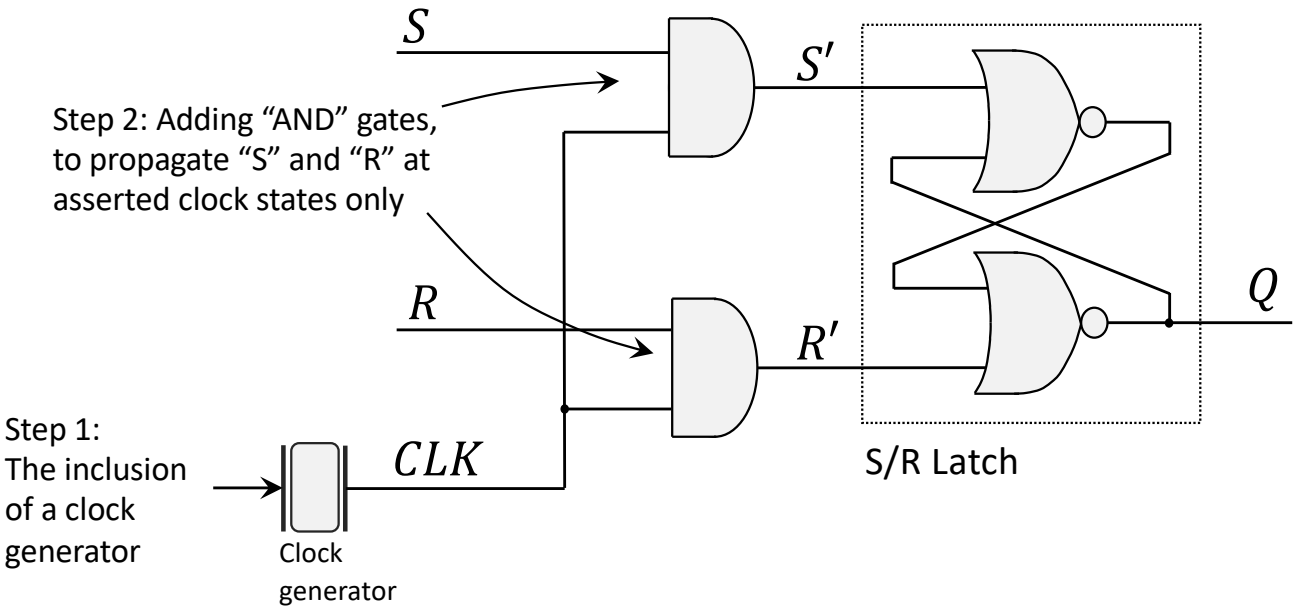
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop

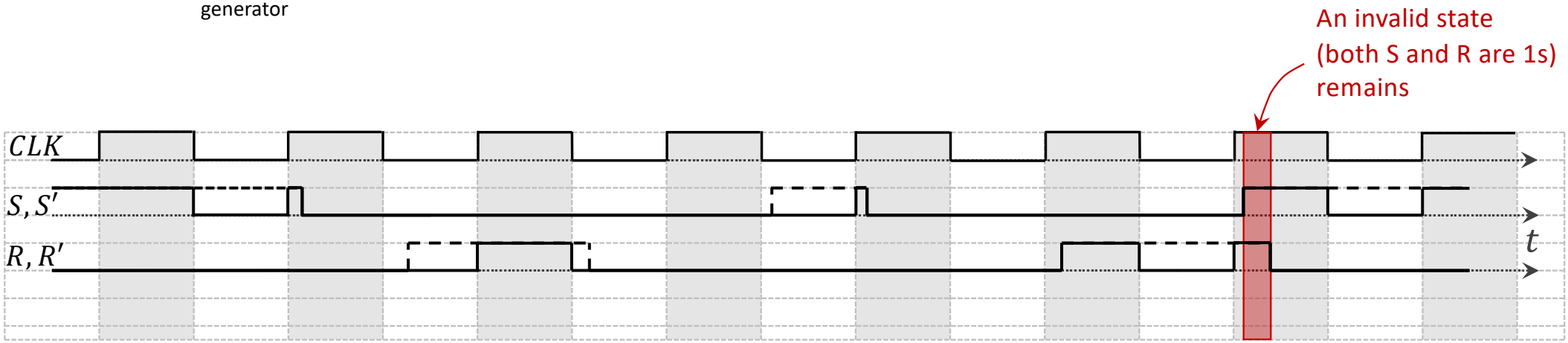


Assumption:

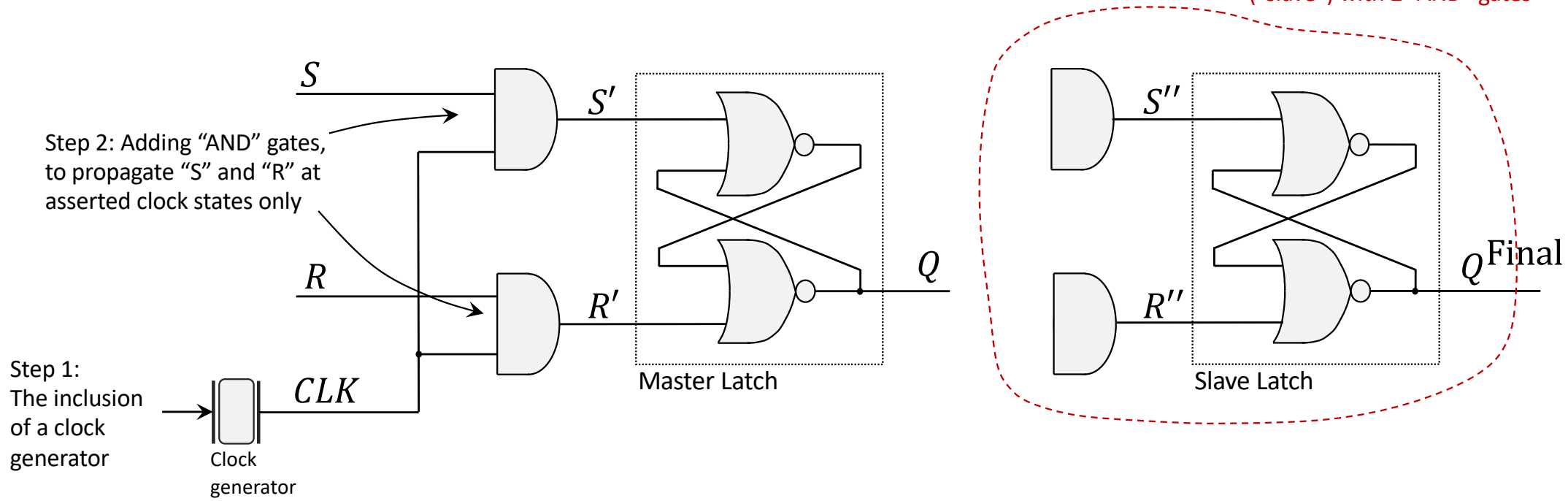
Latch propagation delay Δt^{latch} is insignificant, as compared to Δt^{other}

Transformation of an Asynchronous Latch into a Synchronous Flip-Flop

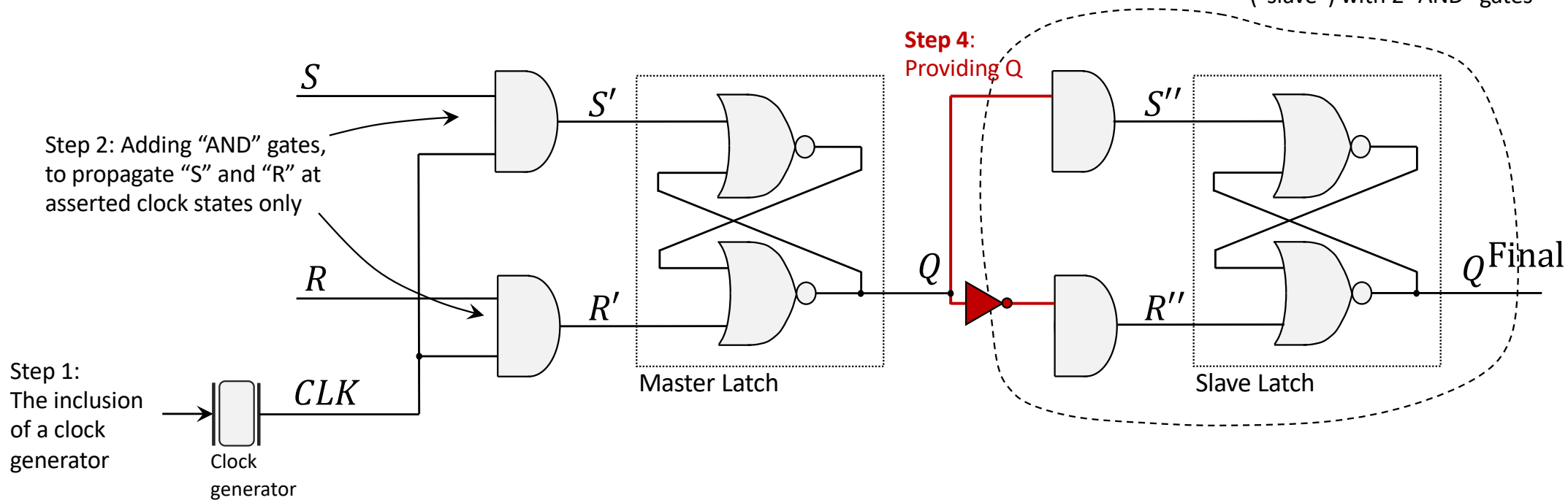




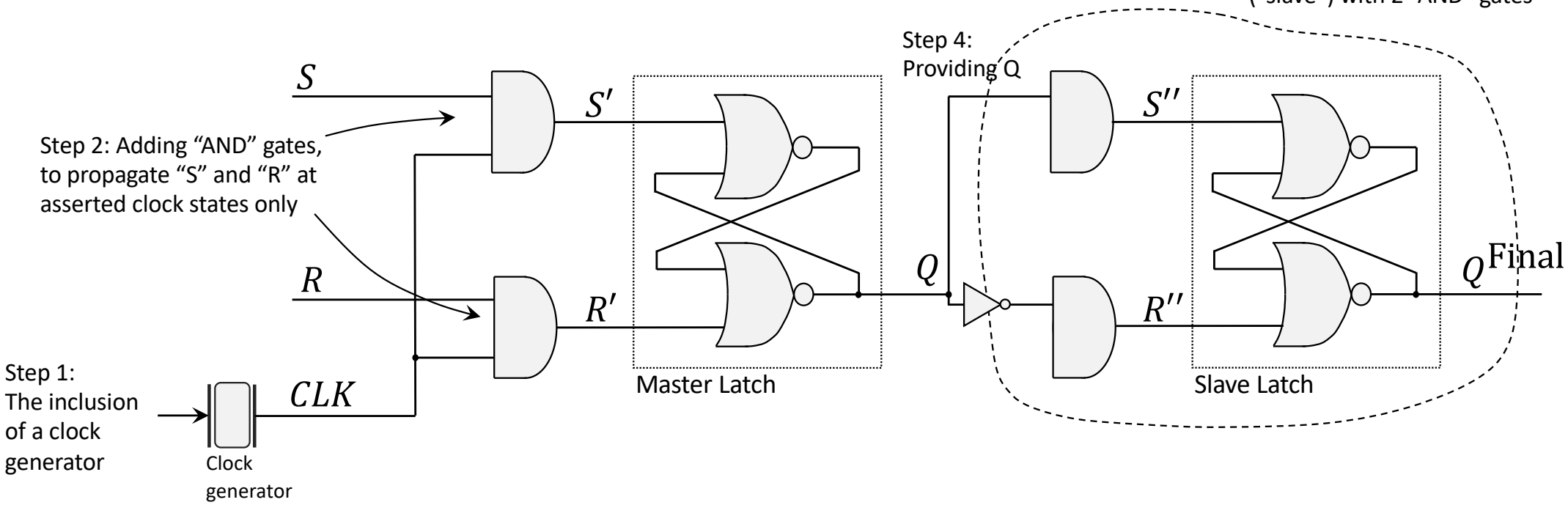
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



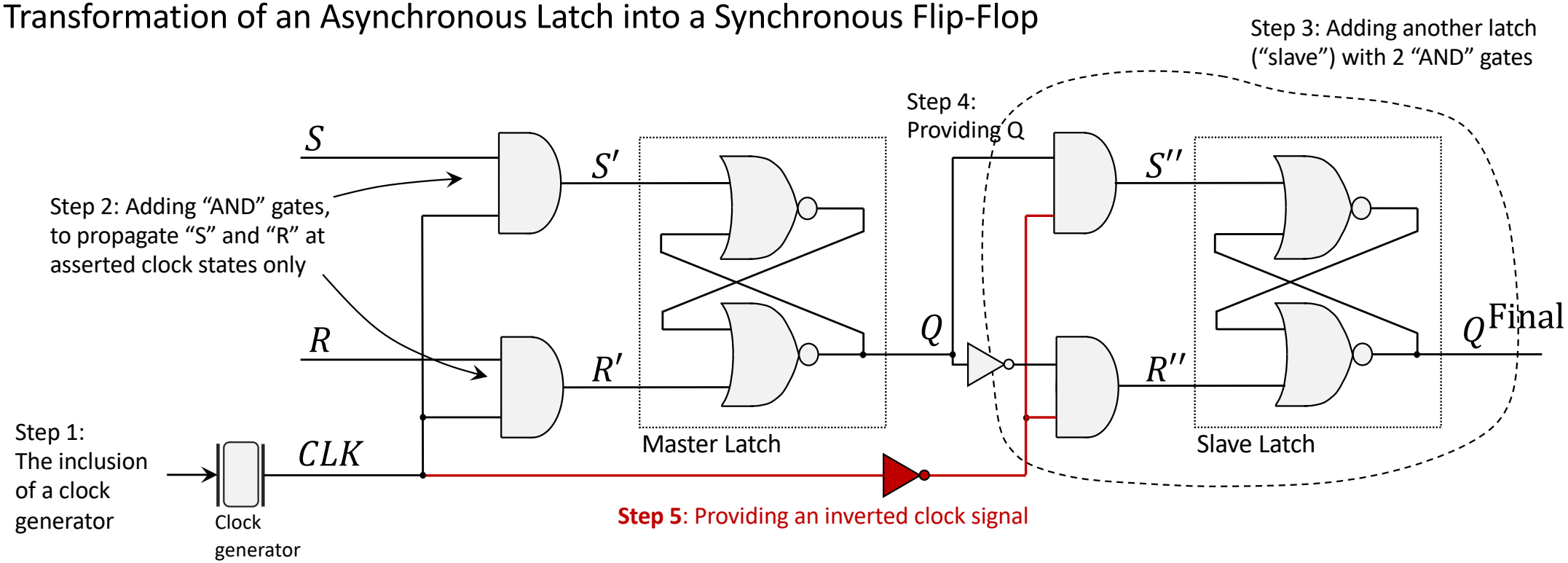
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



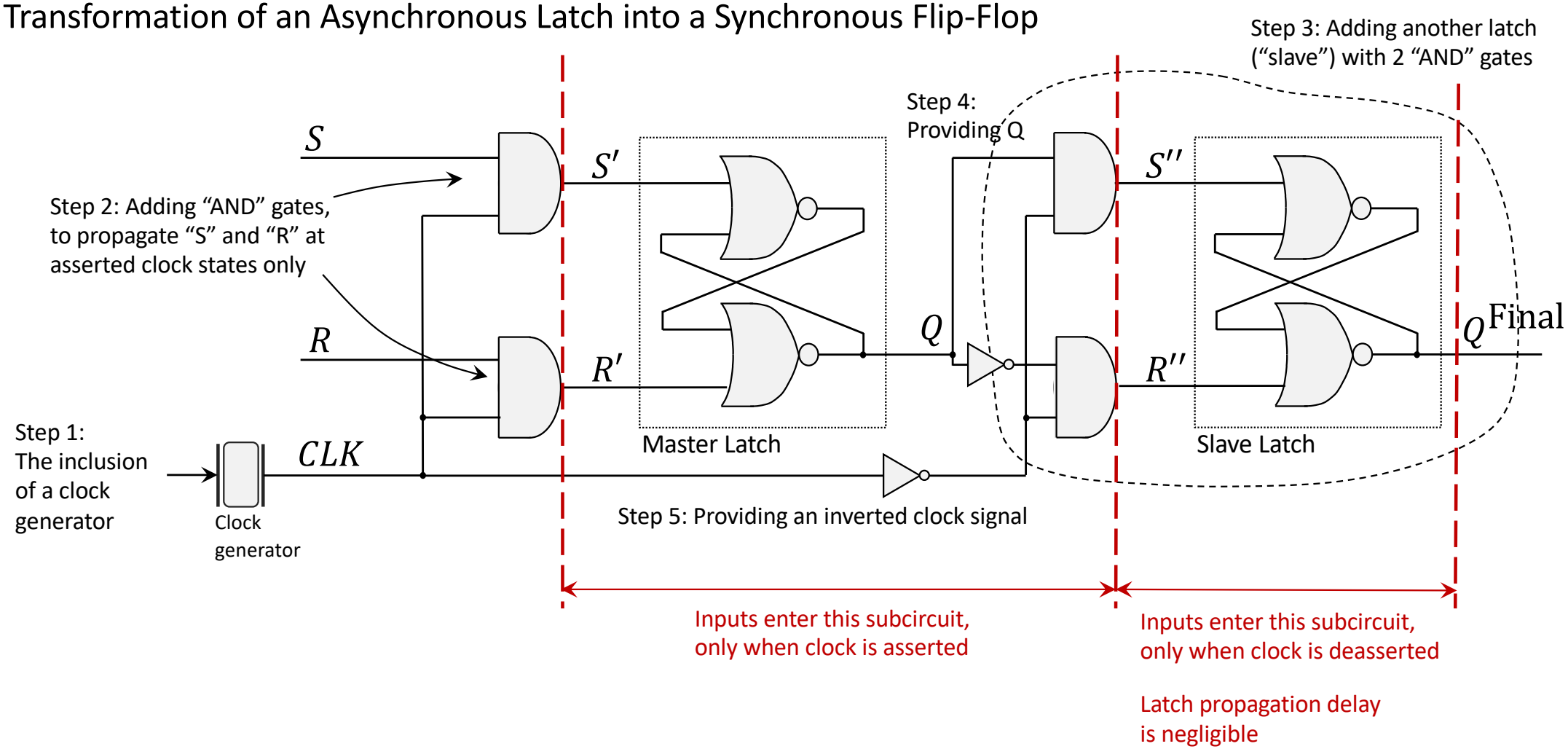
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



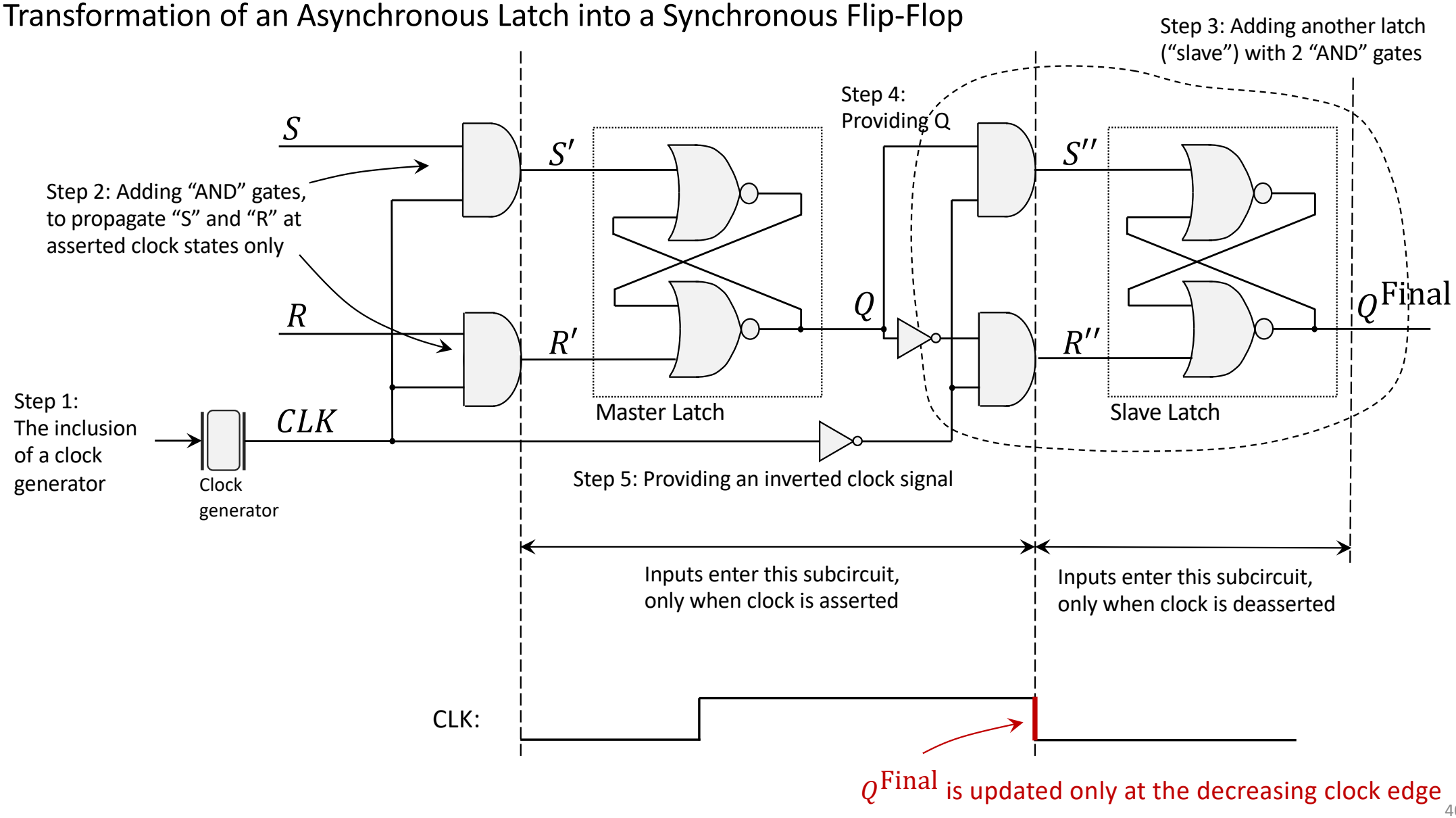
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



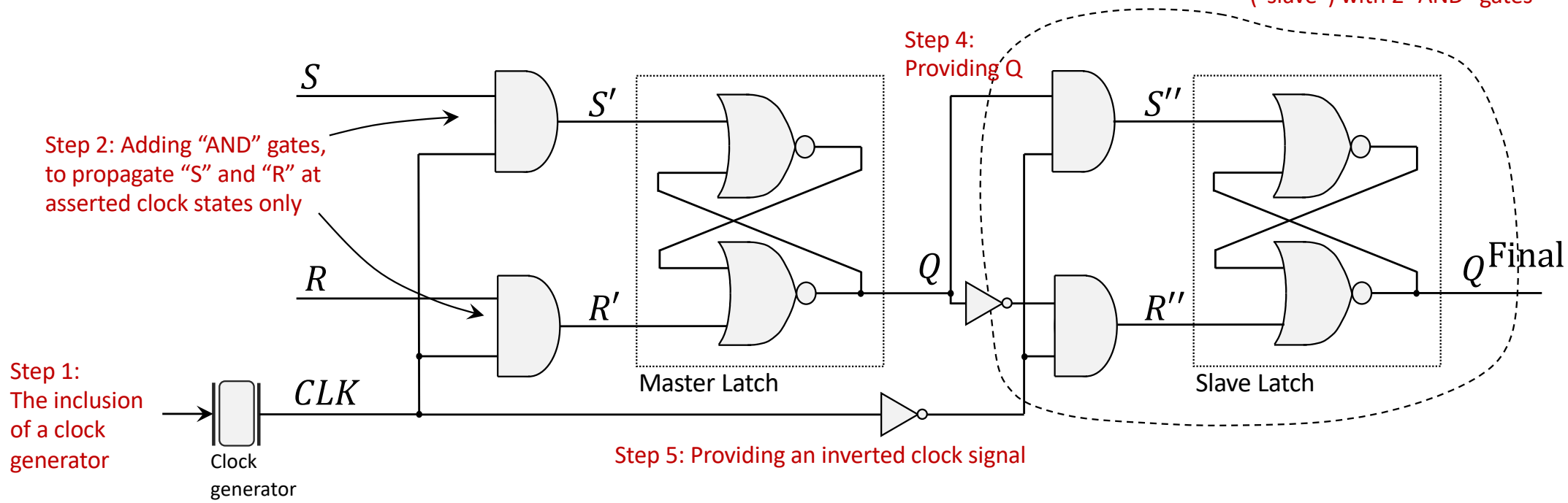
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



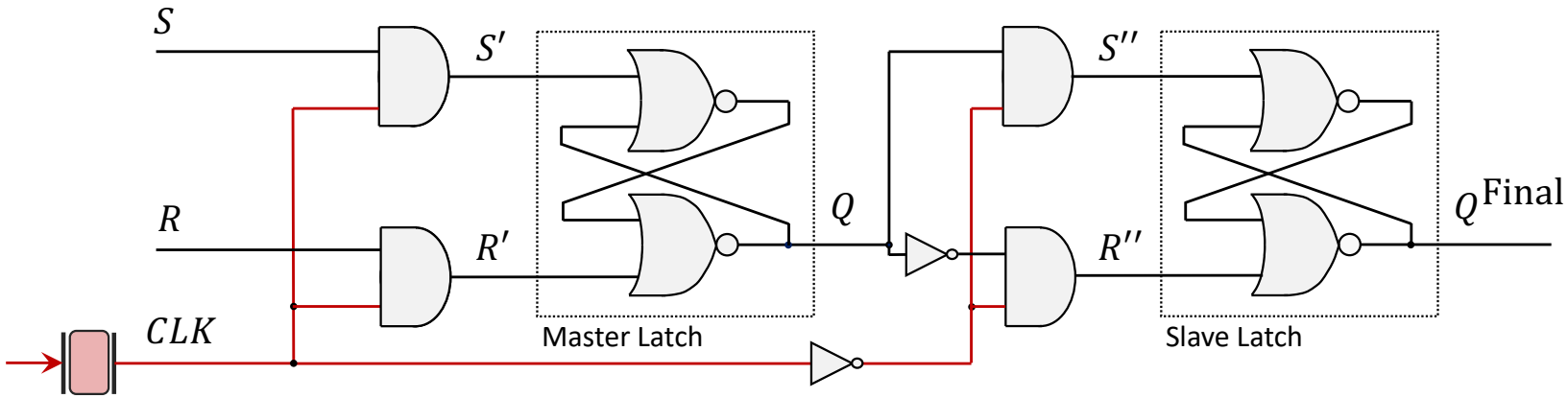
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



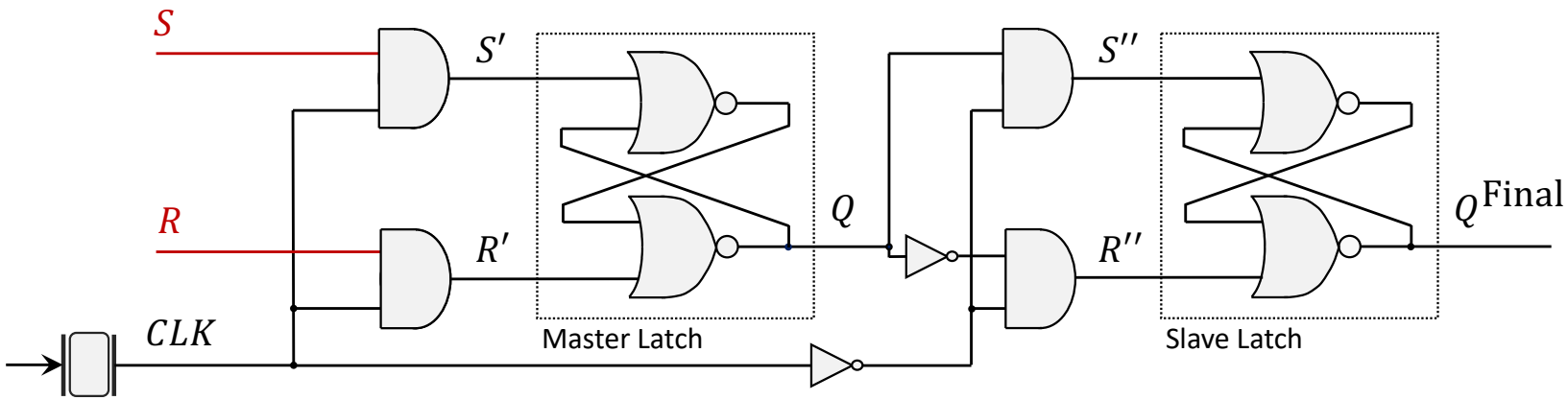
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop

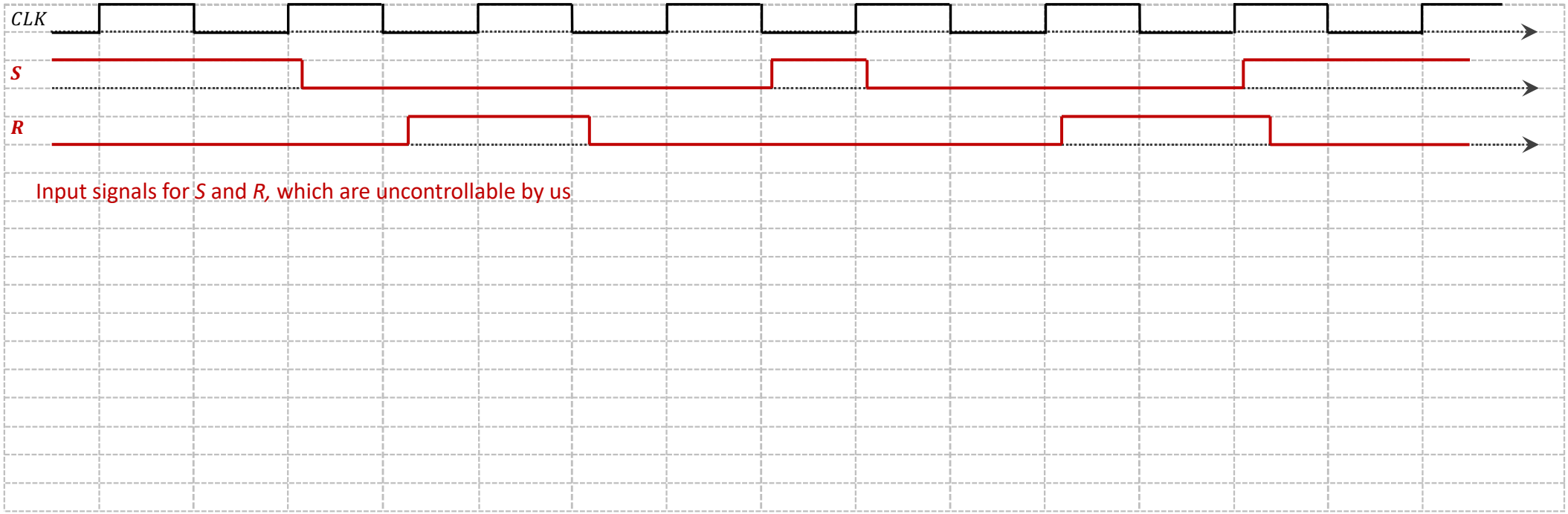


A clock signal, which governs the entire activity (that is the propagation of electrical signals) in our circuit;

A clock periodicity is decided by a computer architect, accounting the expected propagation delays

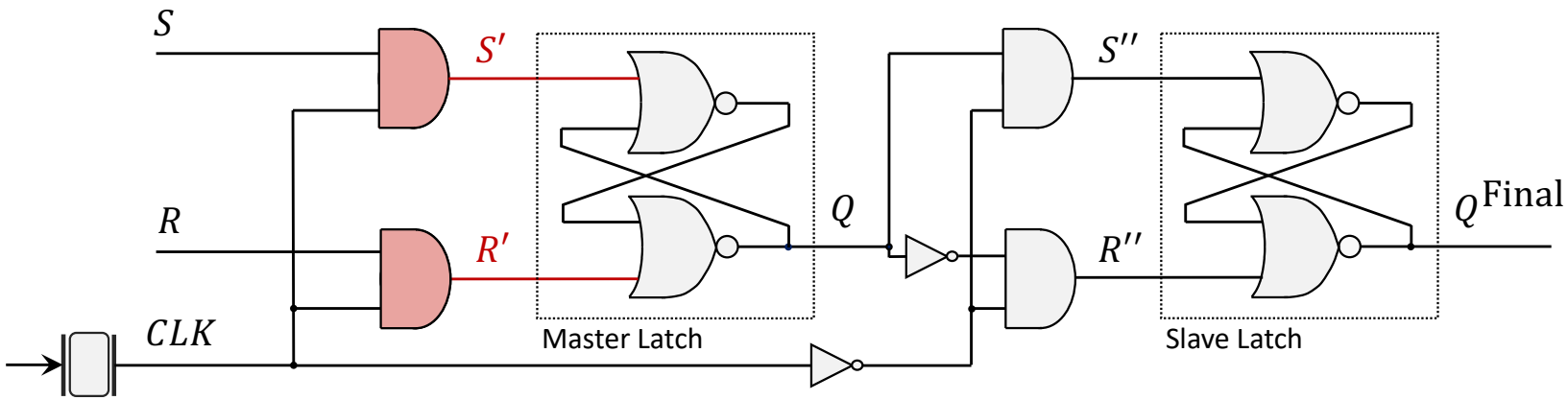
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop

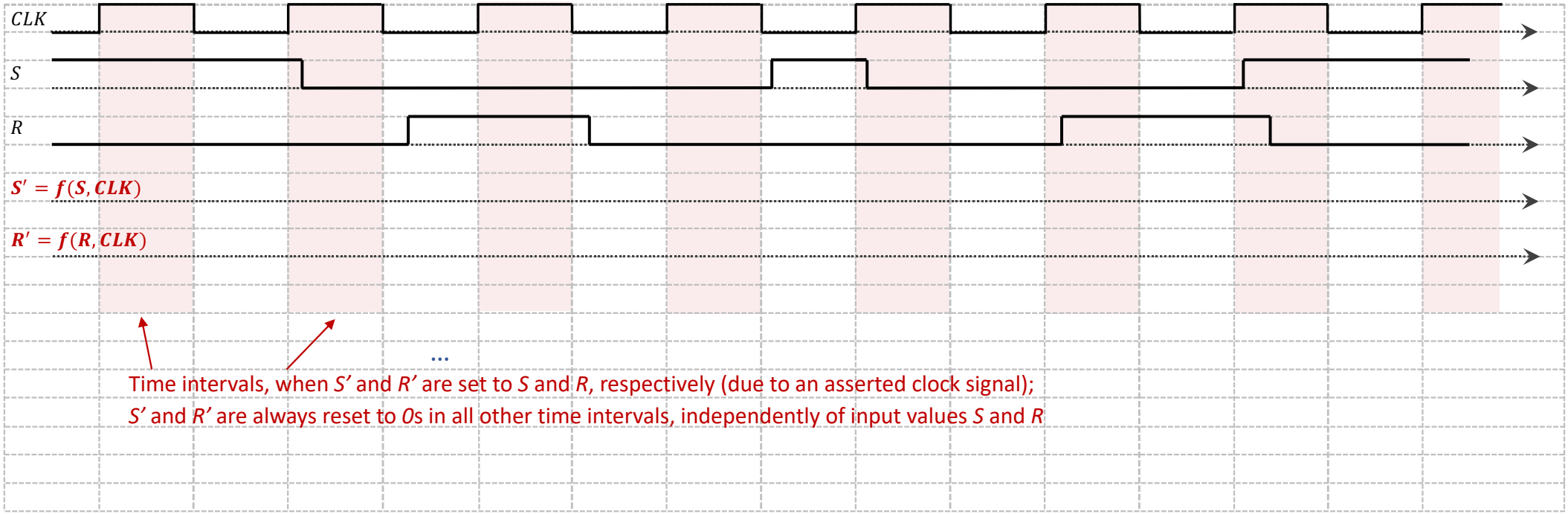




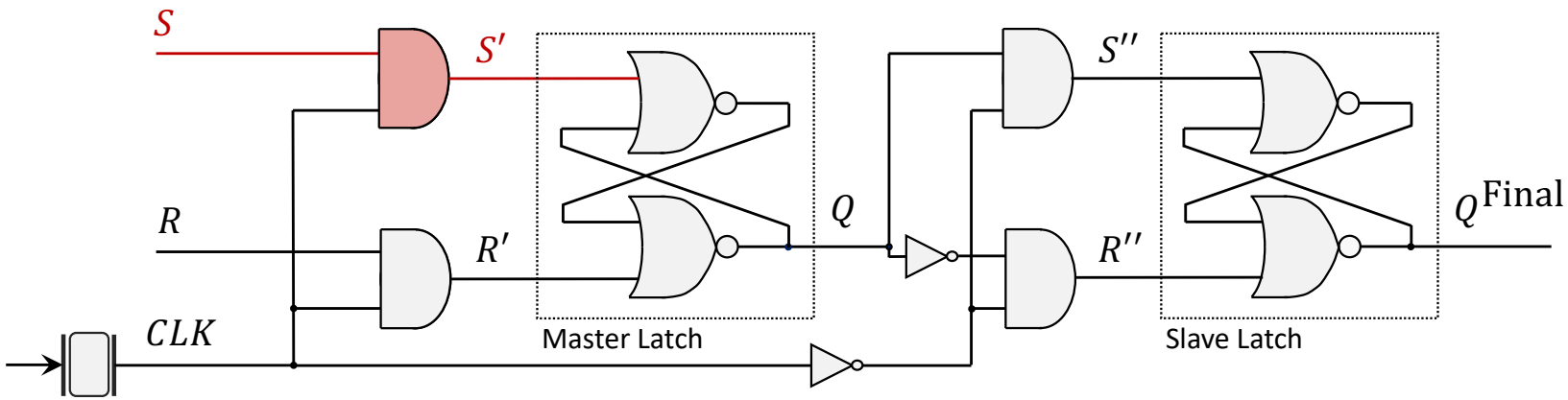
Input signals for *S* and *R*, which are uncontrollable by us

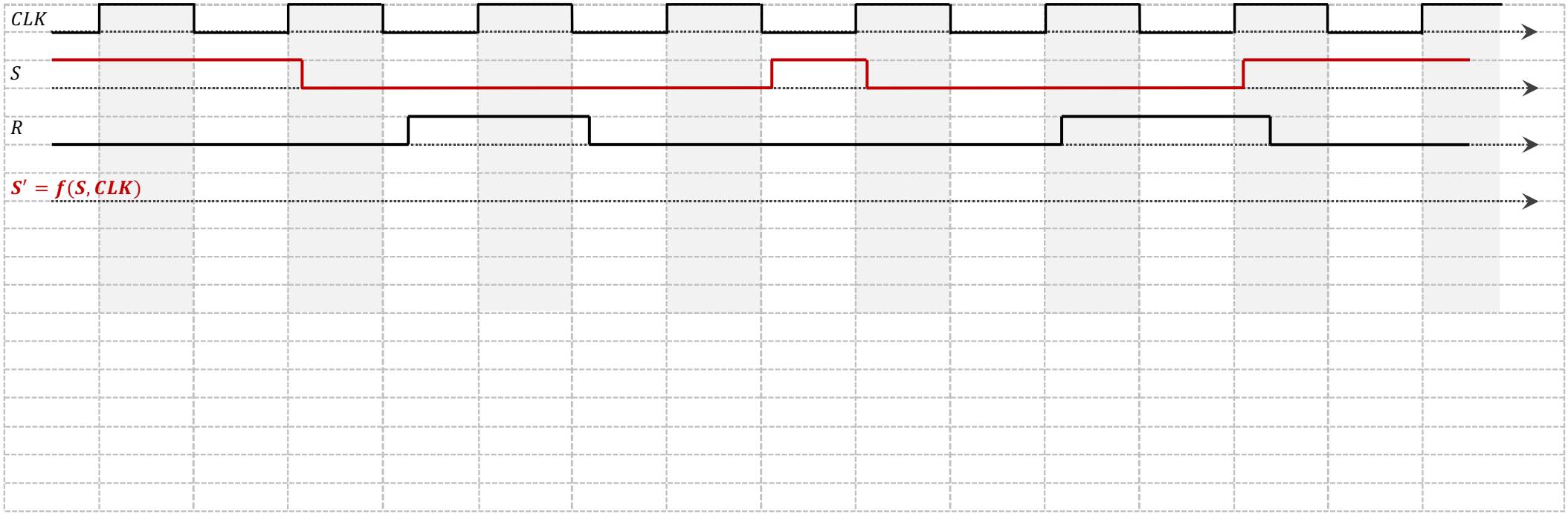
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



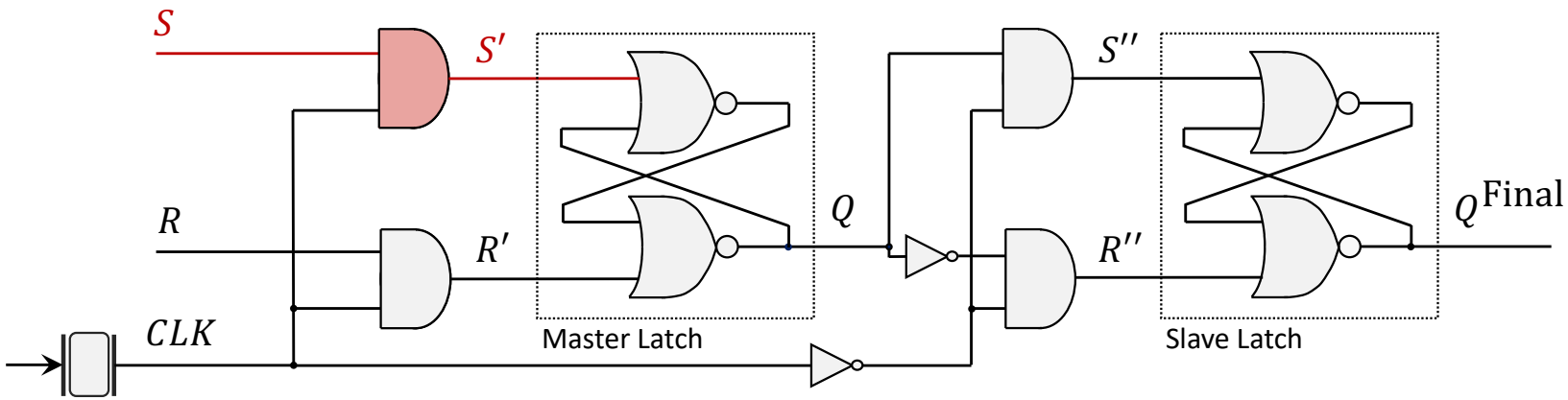


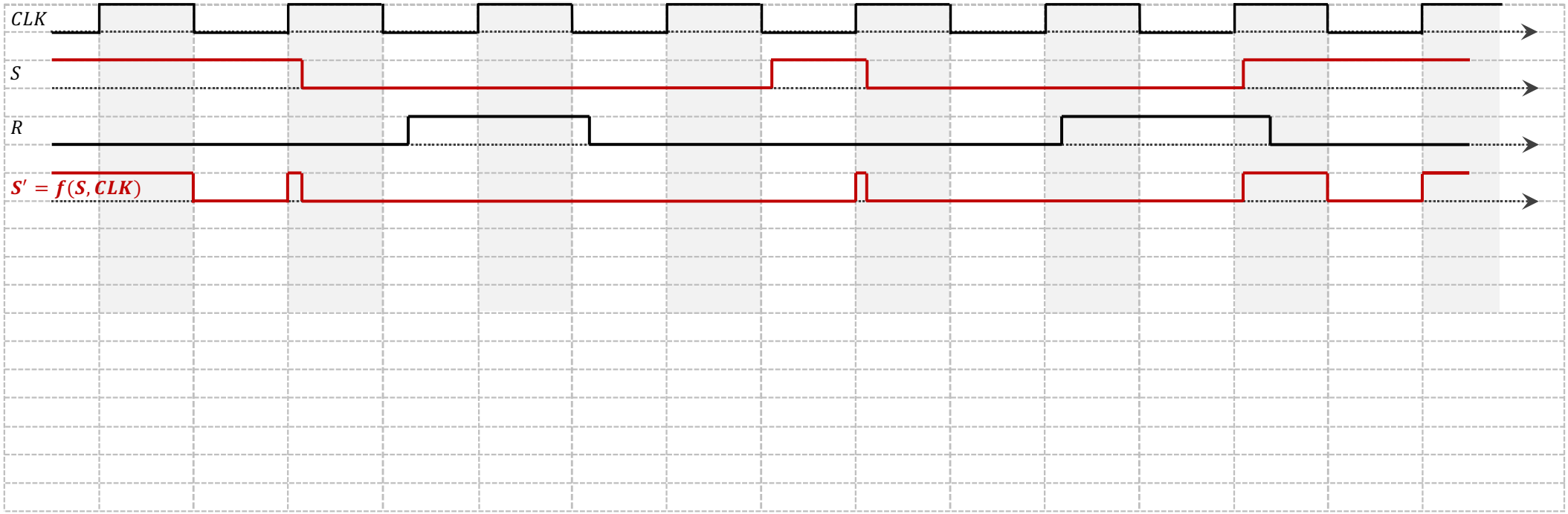
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



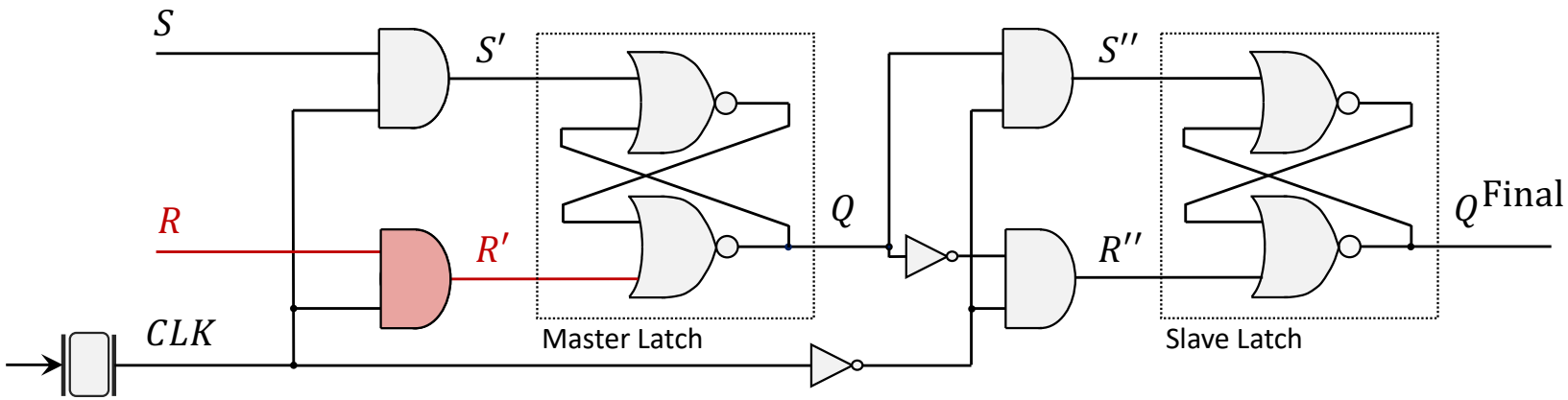


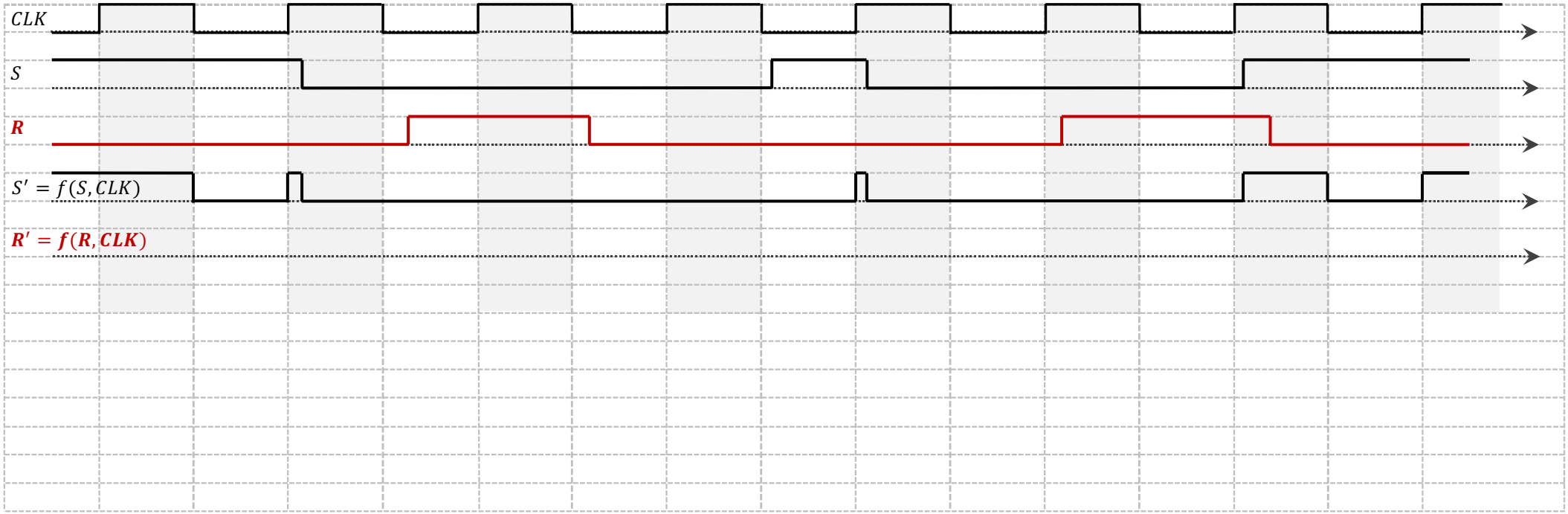
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



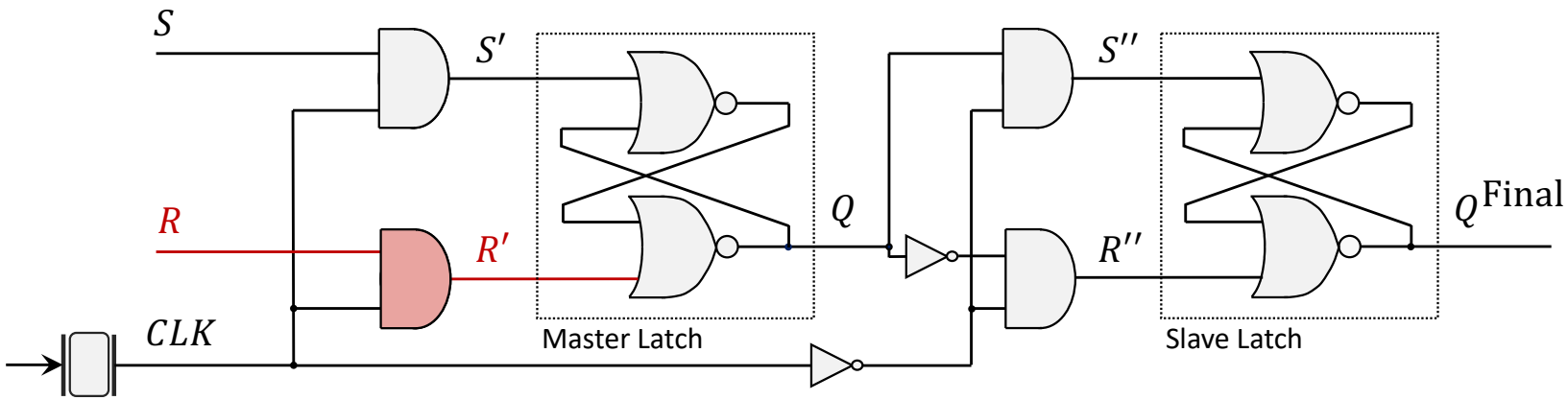


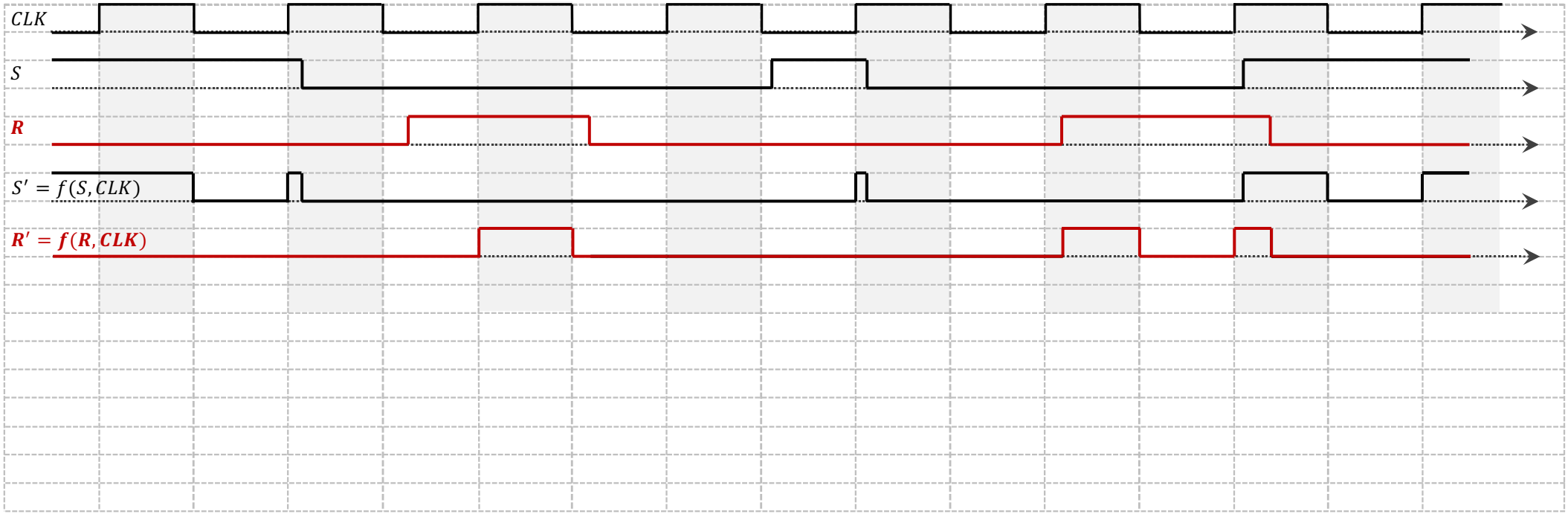
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



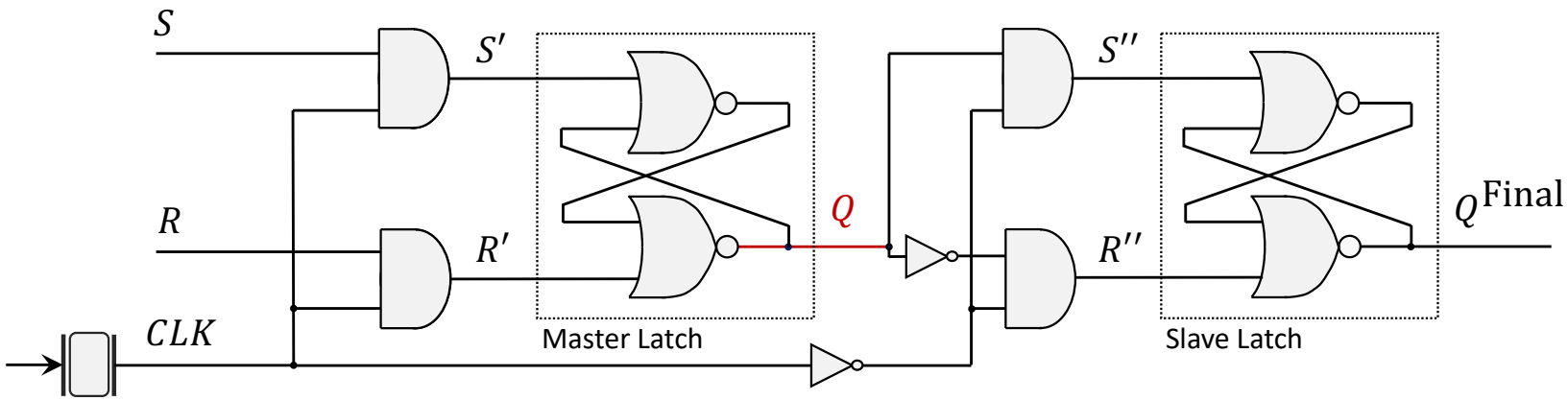


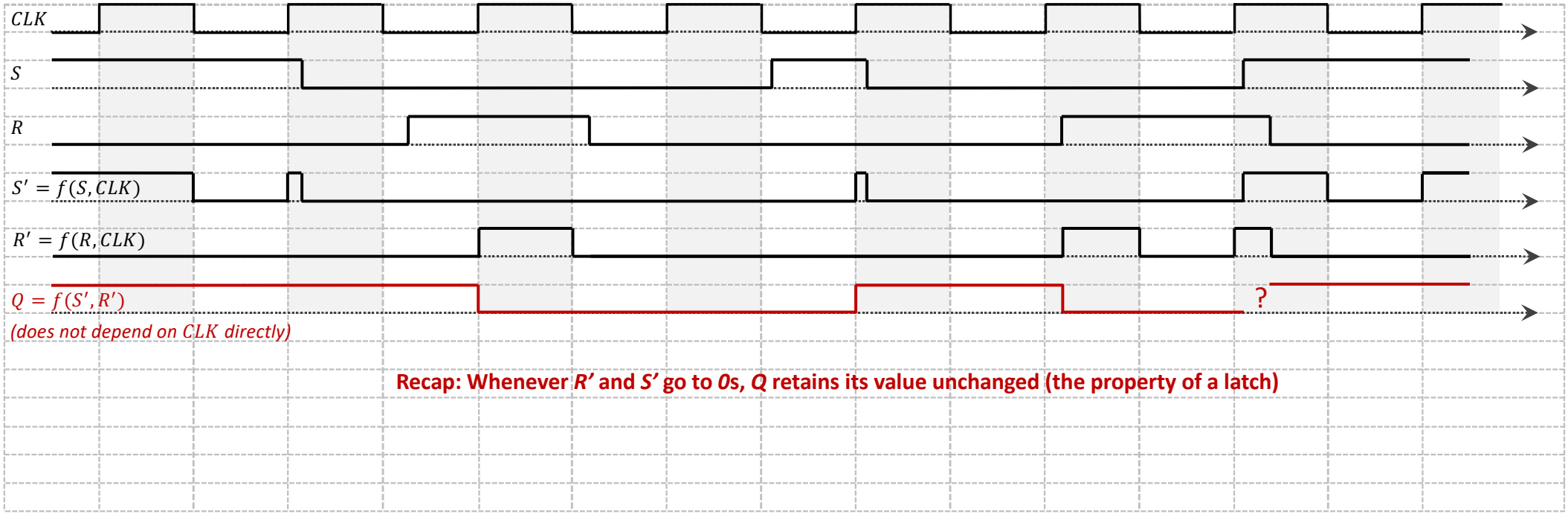
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop





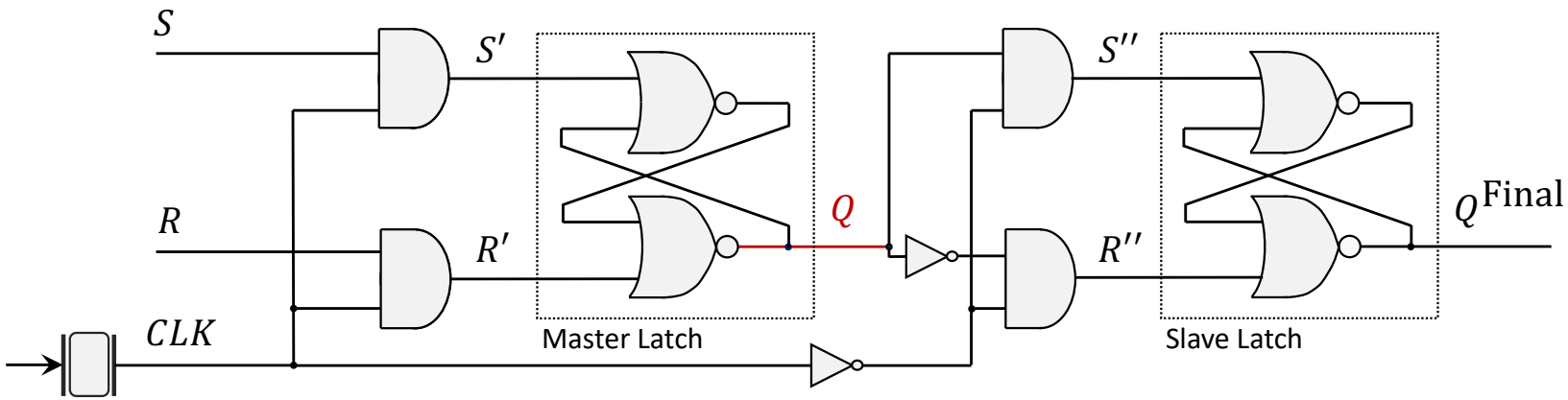
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop

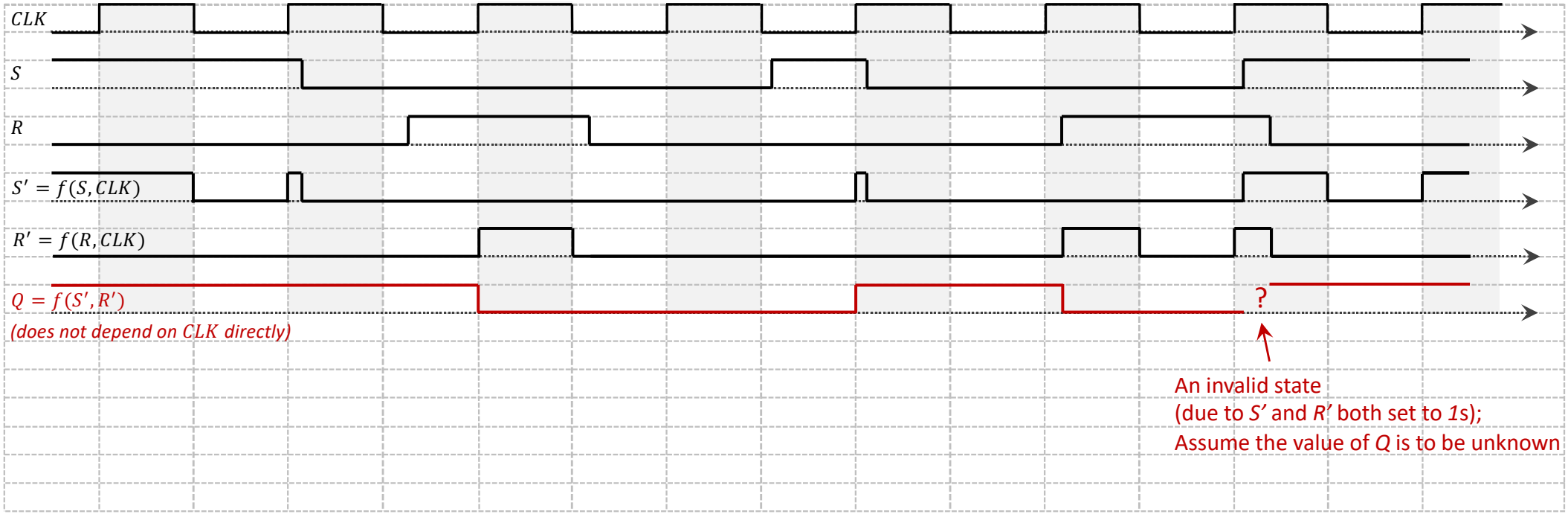




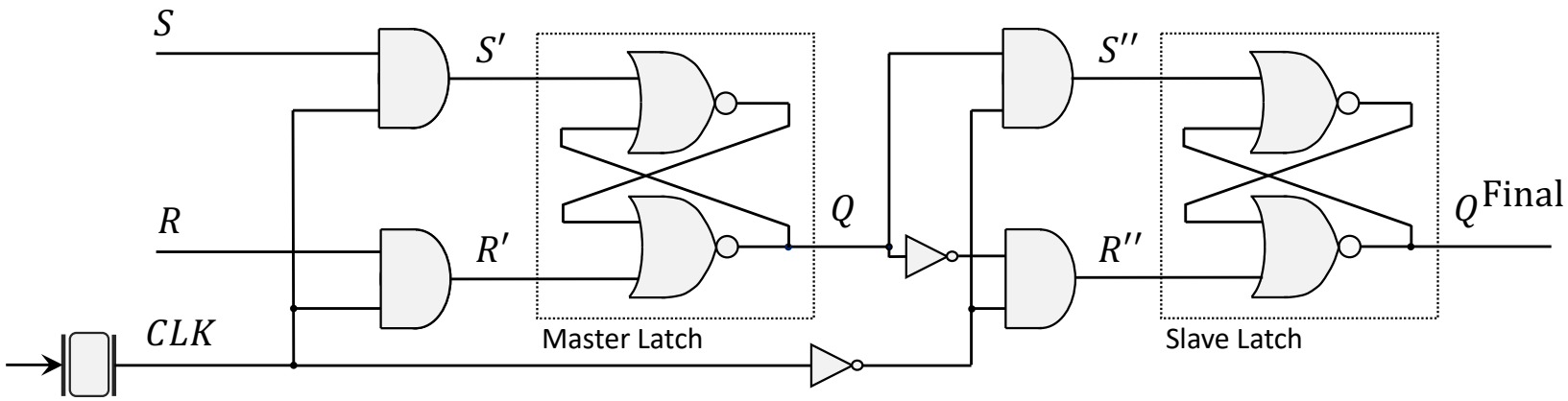
Recap: Whenever *R'* and *S'* go to *0*s, *Q* retains its value unchanged (the property of a latch)

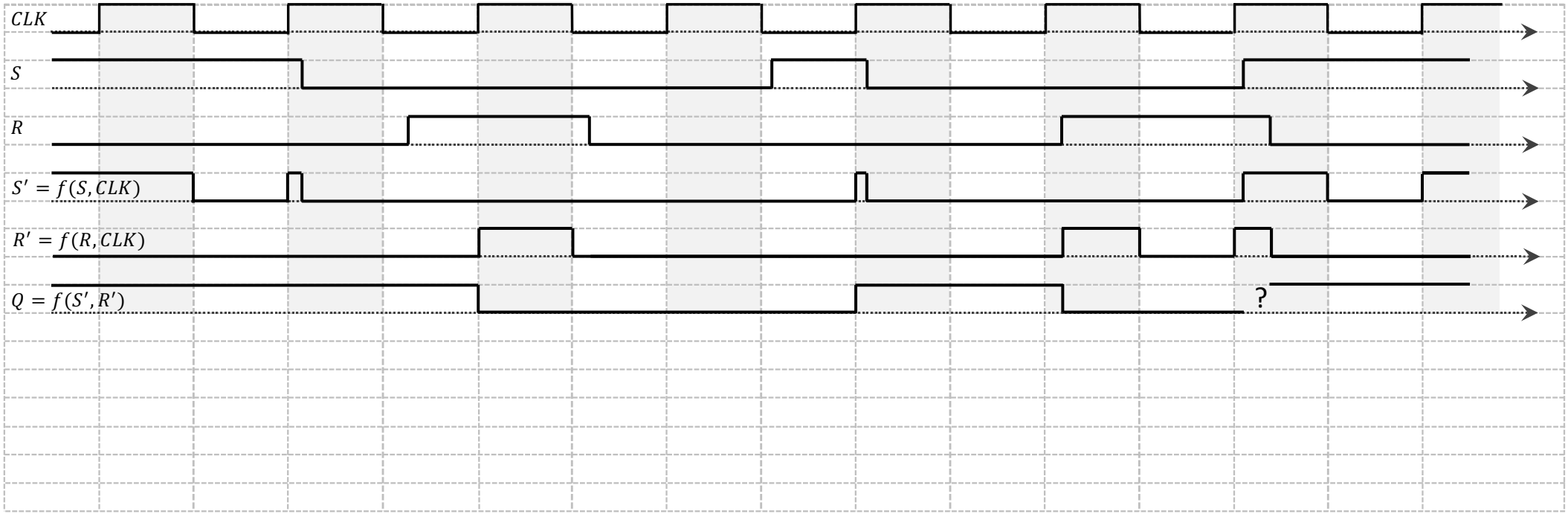
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



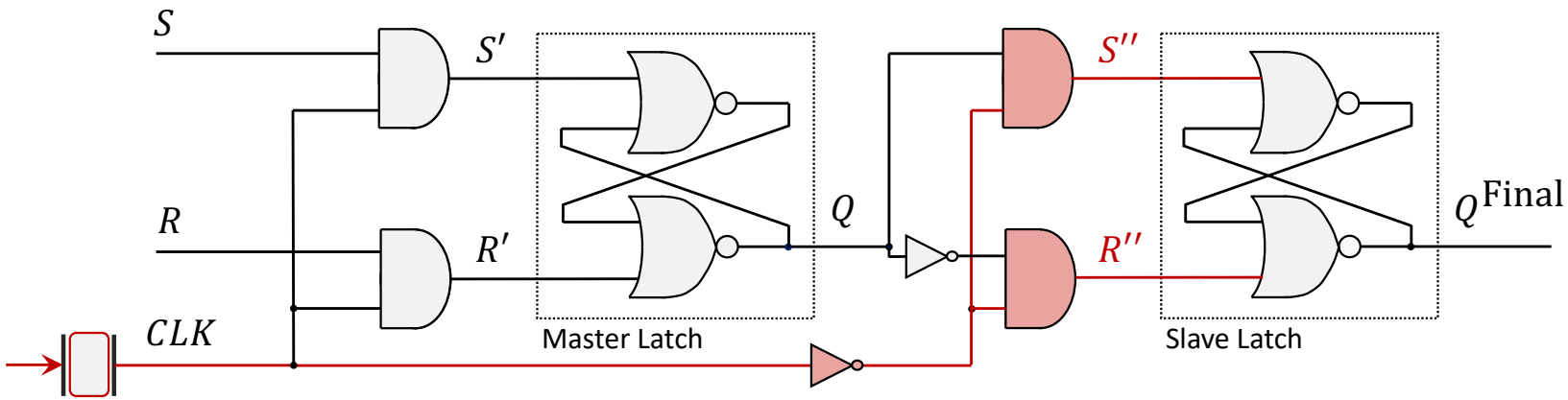


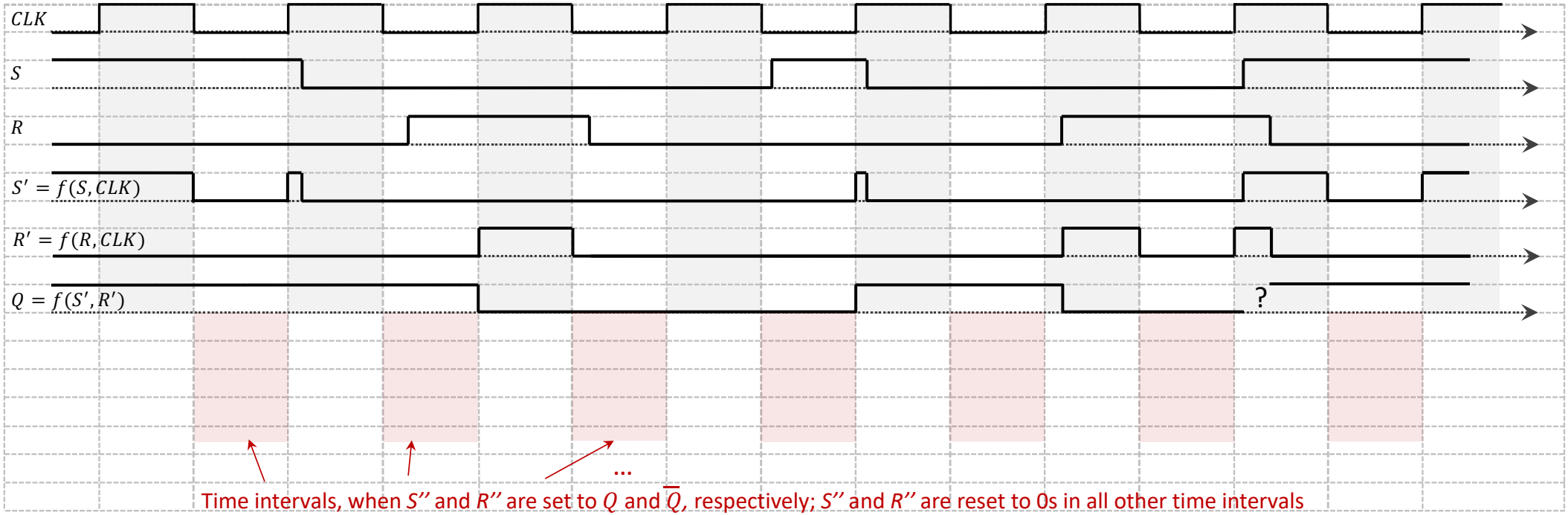
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



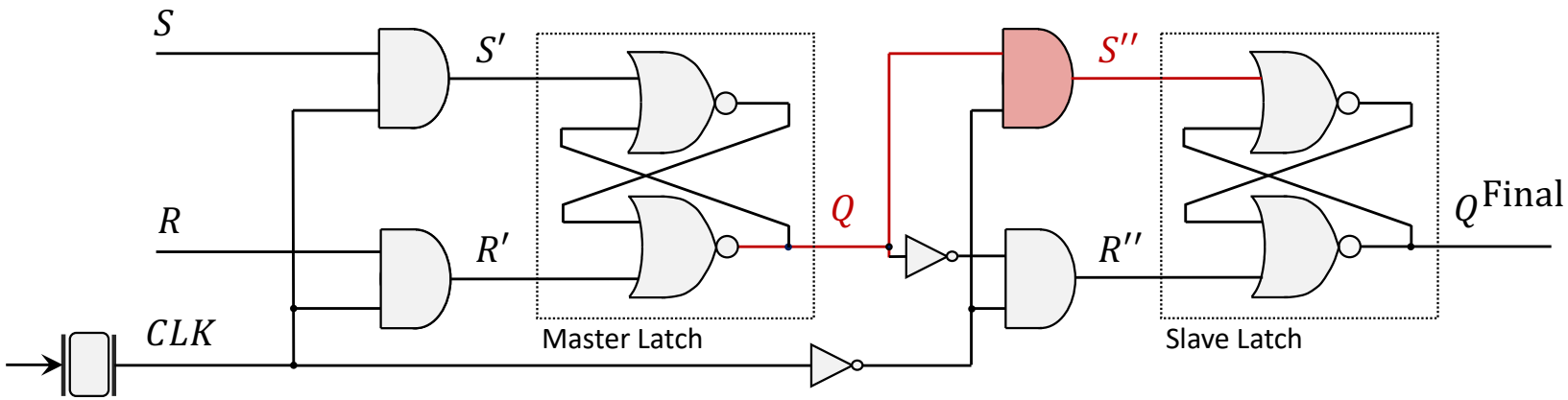


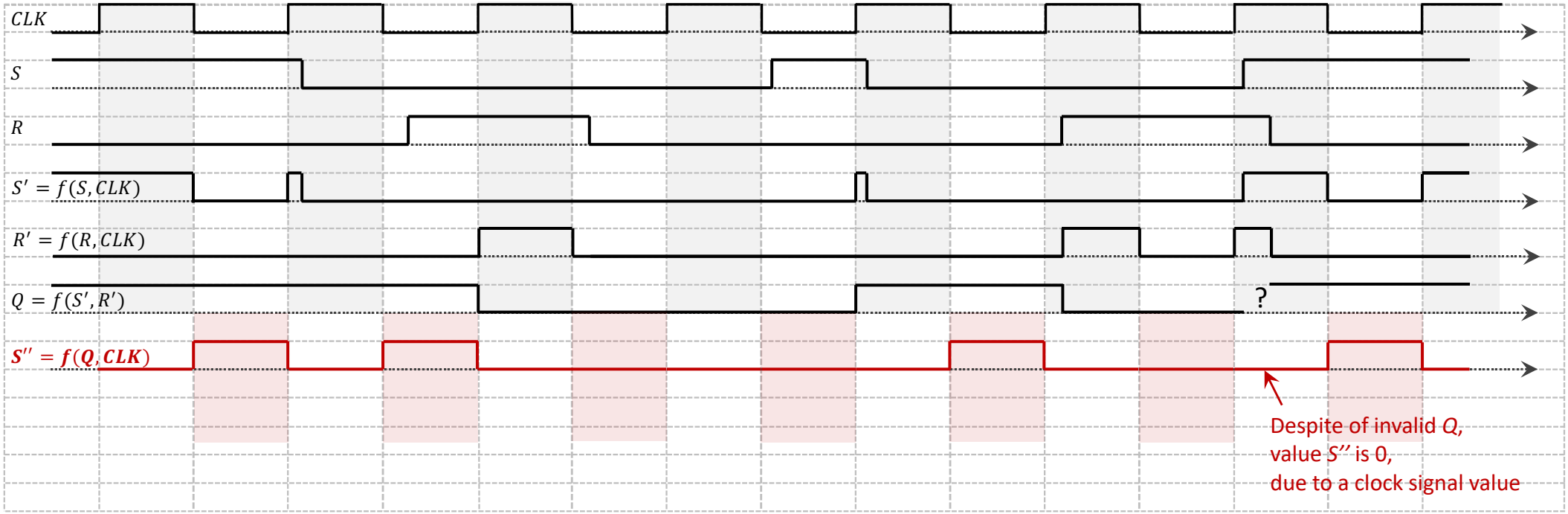
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



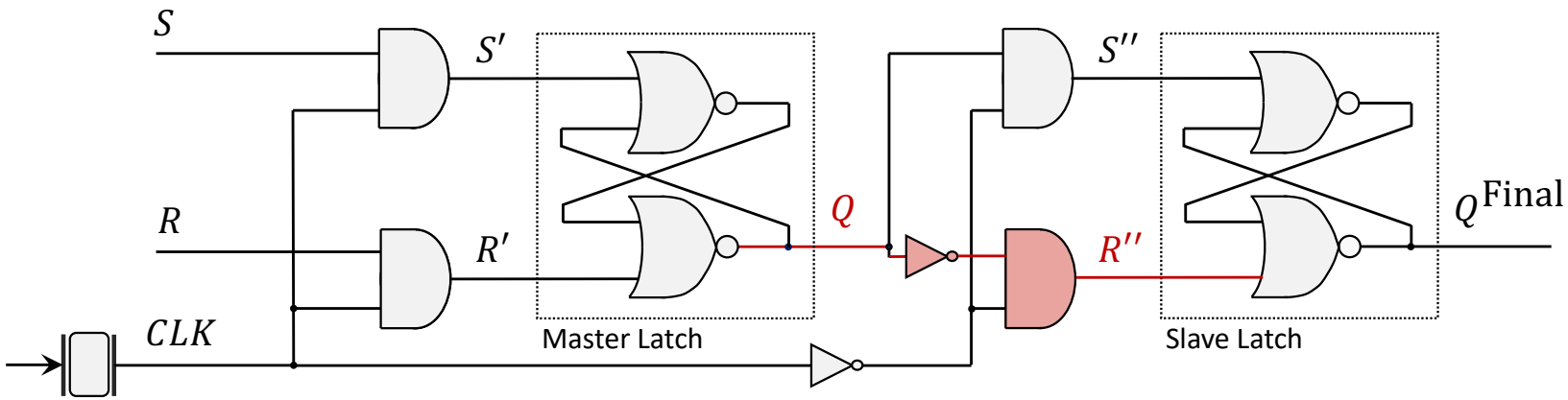


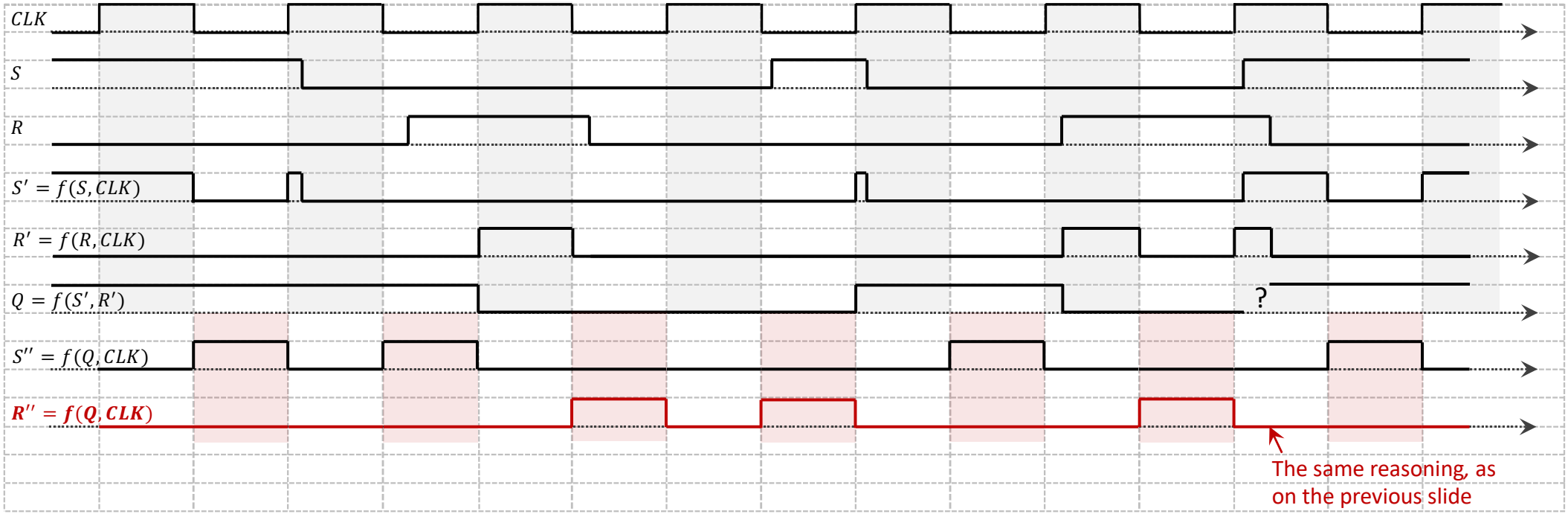
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop



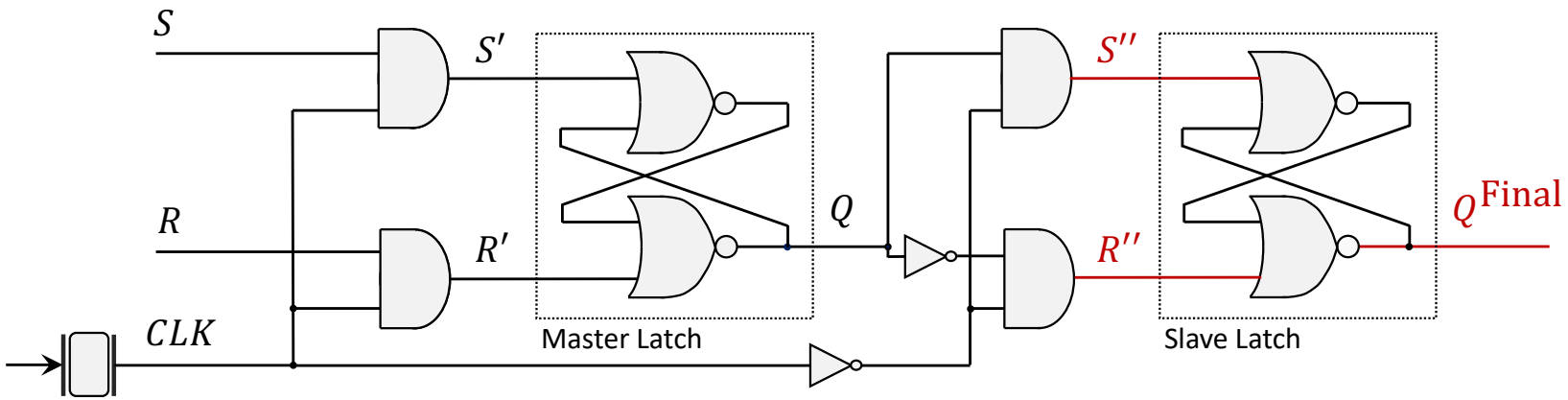


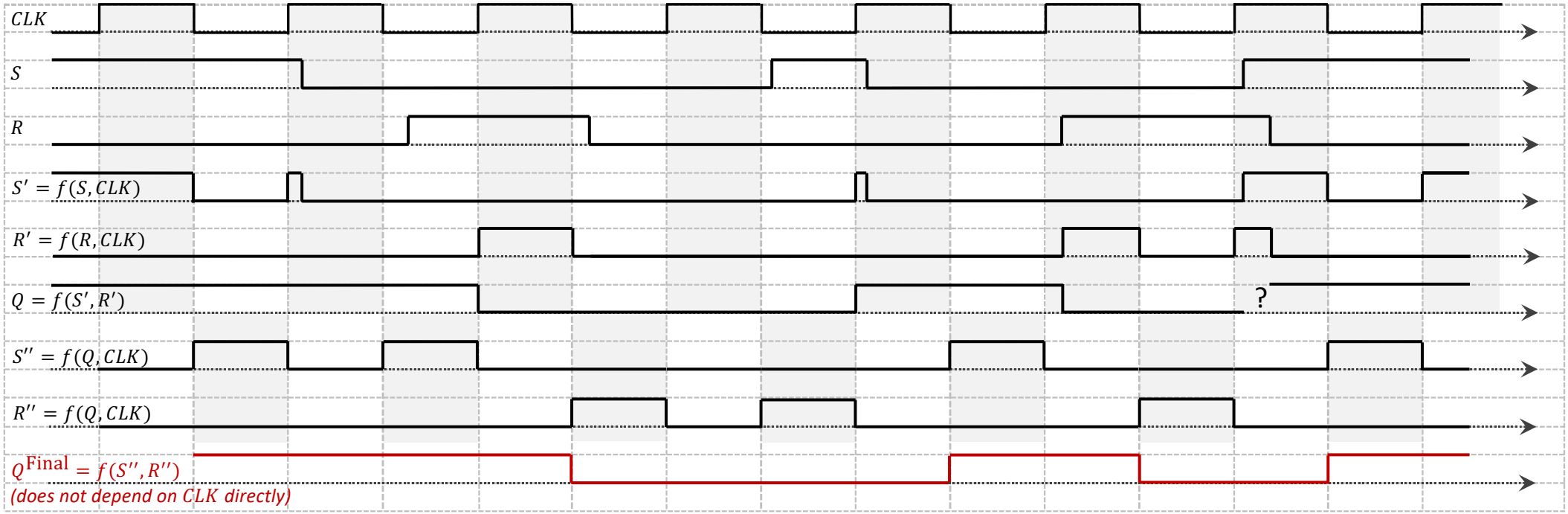
Transformation of an Asynchronous Latch into a Synchronous Flip-Flop

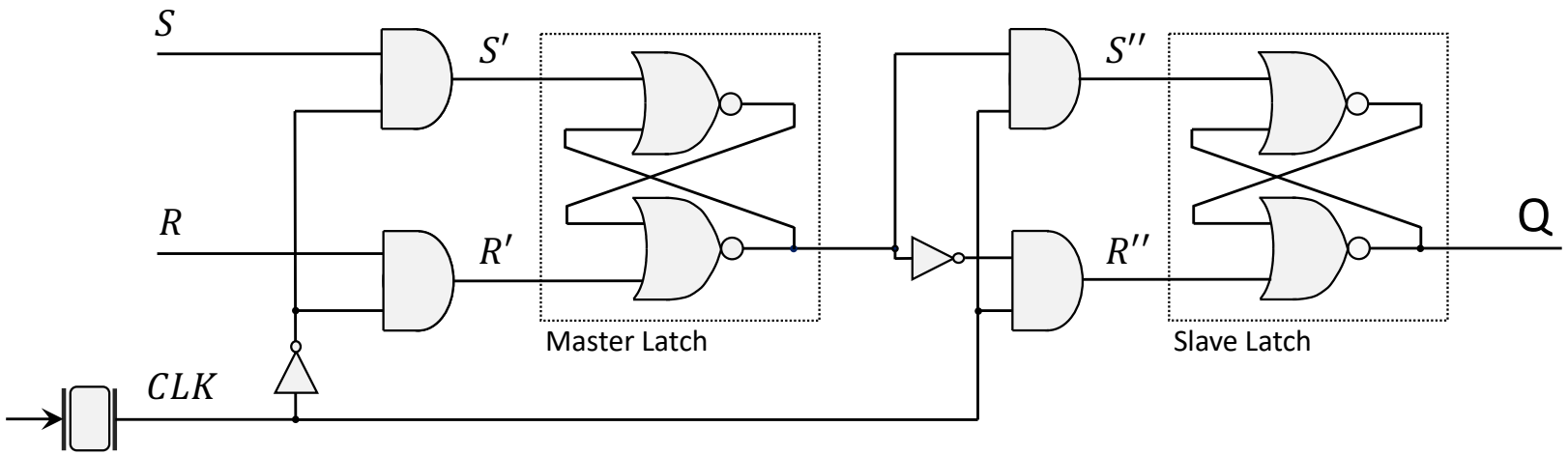




Transformation of an Asynchronous Latch into a Synchronous Flip-Flop






S
S′
R
R′
CLK
Master Latch
S″
R″
Slave Latch
Q

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger		
Clock Presence		
Reliability (for result correctness)		
Memory Element Presence		
Operation Speed		
Power Consumption		
Logical Complexity, Size (the number of elements)		
Sample Use Cases		

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger	Any change of one of the inputs	
Clock Presence		
Reliability (for result correctness)		
Memory Element Presence		
Operation Speed		
Power Consumption		
Logical Complexity, Size (the number of elements)		
Sample Use Cases		

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger	Any change of one of the inputs	A clock signal, together with other input signals
Clock Presence		
Reliability (for result correctness)		
Memory Element Presence		
Operation Speed		
Power Consumption		
Logical Complexity, Size (the number of elements)		
Sample Use Cases		

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger	Any change of one of the inputs	A clock signal, together with other input signals
Clock Presence		Clock governs the entire circuit activity
Reliability (for result correctness)		
Memory Element Presence		
Operation Speed		
Power Consumption		
Logical Complexity, Size (the number of elements)		
Sample Use Cases		

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger	Any change of one of the inputs	A clock signal, together with other input signals
Clock Presence	No clock present	Clock governs the entire circuit activity
Reliability (for result correctness)		
Memory Element Presence		
Operation Speed		
Power Consumption		
Logical Complexity, Size (the number of elements)		
Sample Use Cases		

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger	Any change of one of the inputs	A clock signal, together with other input signals
Clock Presence	No clock present	Clock governs the entire circuit activity
Reliability (for result correctness)		Reliable (guarantees a correct result, independently of propagation delays)
Memory Element Presence		
Operation Speed		
Power Consumption		
Logical Complexity, Size (the number of elements)		
Sample Use Cases		

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger	Any change of one of the inputs	A clock signal, together with other input signals
Clock Presence	No clock present	Clock governs the entire circuit activity
Reliability (for result correctness)	Less reliable (prone to incorrect result, subject to propagation delays)	Reliable (guarantees a correct result, independently of propagation delays)
Memory Element Presence		
Operation Speed		
Power Consumption		
Logical Complexity, Size (the number of elements)		
Sample Use Cases		

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger	Any change of one of the inputs	A clock signal, together with other input signals
Clock Presence	No clock present	Clock governs the entire circuit activity
Reliability (for result correctness)	Less reliable (prone to incorrect result, subject to propagation delays)	Reliable (guarantees a correct result, independently of propagation delays)
Memory Element Presence		Used
Operation Speed		
Power Consumption		
Logical Complexity, Size (the number of elements)		
Sample Use Cases		

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger	Any change of one of the inputs	A clock signal, together with other input signals
Clock Presence	No clock present	Clock governs the entire circuit activity
Reliability (for result correctness)	Less reliable (prone to incorrect result, subject to propagation delays)	Reliable (guarantees a correct result, independently of propagation delays)
Memory Element Presence	Not used	Used
Operation Speed		
Power Consumption		
Logical Complexity, Size (the number of elements)		
Sample Use Cases		

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger	Any change of one of the inputs	A clock signal, together with other input signals
Clock Presence	No clock present	Clock governs the entire circuit activity
Reliability (for result correctness)	Less reliable (prone to incorrect result, subject to propagation delays)	Reliable (guarantees a correct result, independently of propagation delays)
Memory Element Presence	Not used	Used
Operation Speed	Faster (no clock signal for synchronisation delay is used	
Power Consumption		
Logical Complexity, Size (the number of elements)		
Sample Use Cases		

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger	Any change of one of the inputs	A clock signal, together with other input signals
Clock Presence	No clock present	Clock governs the entire circuit activity
Reliability (for result correctness)	Less reliable (prone to incorrect result, subject to propagation delays)	Reliable (guarantees a correct result, independently of propagation delays)
Memory Element Presence	Not used	Used
Operation Speed	Faster (no clock signal for synchronisation delay is used	Slower (due to overpessimistic synchronisation delays)
Power Consumption		
Logical Complexity, Size (the number of elements)		
Sample Use Cases		

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger	Any change of one of the inputs	A clock signal, together with other input signals
Clock Presence	No clock present	Clock governs the entire circuit activity
Reliability (for result correctness)	Less reliable (prone to incorrect result, subject to propagation delays)	Reliable (guarantees a correct result, independently of propagation delays)
Memory Element Presence	Not used	Used
Operation Speed	Faster (no clock signal for synchronisation delay is used	Slower (due to overpessimistic synchronisation delays)
Power Consumption	Lower	
Logical Complexity, Size (the number of elements)		
Sample Use Cases		

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger	Any change of one of the inputs	A clock signal, together with other input signals
Clock Presence	No clock present	Clock governs the entire circuit activity
Reliability (for result correctness)	Less reliable (prone to incorrect result, subject to propagation delays)	Reliable (guarantees a correct result, independently of propagation delays)
Memory Element Presence	Not used	Used
Operation Speed	Faster (no clock signal for synchronisation delay is used	Slower (due to overpessimistic synchronisation delays)
Power Consumption	Lower	Higher (e.g. due to the presence of flip-flops, consuming power for data storage)
Logical Complexity, Size (the number of elements)		
Sample Use Cases		

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger	Any change of one of the inputs	A clock signal, together with other input signals
Clock Presence	No clock present	Clock governs the entire circuit activity
Reliability (for result correctness)	Less reliable (prone to incorrect result, subject to propagation delays)	Reliable (guarantees a correct result, independently of propagation delays)
Memory Element Presence	Not used	Used
Operation Speed	Faster (no clock signal for synchronisation delay is used	Slower (due to overpessimistic synchronisation delays)
Power Consumption	Lower	Higher (e.g. due to the presence of flip-flops, consuming power for data storage)
Logical Complexity, Size (the number of elements)	Simpler, smaller (*Note:* an asynchronous circuit might behave as a synchronous, but at the price of a higher hardware implementation complexity)	
Sample Use Cases		

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger	Any change of one of the inputs	A clock signal, together with other input signals
Clock Presence	No clock present	Clock governs the entire circuit activity
Reliability (for result correctness)	Less reliable (prone to incorrect result, subject to propagation delays)	Reliable (guarantees a correct result, independently of propagation delays)
Memory Element Presence	Not used	Used
Operation Speed	Faster (no clock signal for synchronisation delay is used	Slower (due to overpessimistic synchronisation delays)
Power Consumption	Lower	Higher (e.g. due to the presence of flip-flops, consuming power for data storage)
Logical Complexity, Size (the number of elements)	Simpler, smaller (*Note:* an asynchronous circuit might behave as a synchronous, but at the price of a higher hardware implementation complexity)	More complex, larger
Sample Use Cases		

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger	Any change of one of the inputs	A clock signal, together with other input signals
Clock Presence	No clock present	Clock governs the entire circuit activity
Reliability (for result correctness)	Less reliable (prone to incorrect result, subject to propagation delays)	Reliable (guarantees a correct result, independently of propagation delays)
Memory Element Presence	Not used	Used
Operation Speed	Faster (no clock signal for synchronisation delay is used	Slower (due to overpessimistic synchronisation delays)
Power Consumption	Lower	Higher (e.g. due to the presence of flip-flops, consuming power for data storage)
Logical Complexity, Size (the number of elements)	Simpler, smaller (*Note:* an asynchronous circuit might behave as a synchronous, but at the price of a higher hardware implementation complexity)	More complex, larger
Sample Use Cases		Register files; Most of the circuits containing memory elements

Synchronous vs. Asynchronous Circuits

Characteristic	**Asynchronous** (like S/R latch)	**Synchronous** (like Flip-Flop)
Output Update Trigger	Any change of one of the inputs	A clock signal, together with other input signals
Clock Presence	No clock present	Clock governs the entire circuit activity
Reliability (for result correctness)	Less reliable (prone to incorrect result, subject to propagation delays)	Reliable (guarantees a correct result, independently of propagation delays)
Memory Element Presence	Not used	Used
Operation Speed	Faster (no clock signal for synchronisation delay is used)	Slower (due to overpessimistic synchronisation delays)
Power Consumption	Lower	Higher (e.g. due to the presence of flip-flops, consuming power for data storage)
Logical Complexity, Size (the number of elements)	Simpler, smaller (*Note:* an asynchronous circuit might behave as a synchronous, but at the price of a higher hardware implementation complexity)	More complex, larger
Sample Use Cases	Arithmetic-Logic Unit (ALU); Small fast peripheral circuits supporing CPU operation	Register files; Most of the circuits containing memory elements

Digital Circuits Classification