Sequential Logic Design Principles.Latches and Flip-Flops

Doru Todinca

Department of Computers Politehnica University of Timisoara

◆□▶ ◆□▶ ◆三▶ ◆三▶ - 三 - のへぐ

Introduction

Bistable Elements

Latches and Flip-Flops S-R Latch **Š**-**R** Latch S-R Latch with Enable D I atch Edge-Triggered D Flip-Flop Edge-Triggered D Flip-Flop with Enable Scan Flip-Flop Master/Slave S-R Flip-Flop Master/Slave J-K Flip-Flop Edge-Triggered J-K Flip-Flop T Flip-Flop

Introduction

Bistable Elements

Master/Slave S-R Flip-Flop Master/Slave J-K Flip-Flop Edge-Triggered J-K Flip-Flop

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三回 ● のへで

Introduction. Definitions

- The outputs of a sequential circuit depend not only on its current inputs, but also on the past sequence of inputs, possibly arbitrarily back in time.
- It means that it is unpractical, or even impossible, to describe the behavior of a sequential circuit by a table containing the outputs as functions of a limited sequence of inputs
- We can describe the behavior of the circuit if we know its current *state*.
- The best definition of state ([Wakerly]) is from the Herbert Hellerman's book *Digital Computer System Principles* (McGraw-Hill, 1967):

Definition

The *state* of a sequential circuit is a collection of *state variables* whose values at any one time contain all the information about the past necessary to account for the circuit's future behavior.

Introduction

- In digital circuits, state variables are binary values
- ▶ A circuit with *n* state variables can have 2^{*n*} states
- Since 2ⁿ is a finite value, sequential circuits are called also finite state machines.
- We will discuss two types of sequential circuits:
 - 1. *feedback sequential circuits:* use ordinary gates and feedback loops to obtain memory in a logic circuit (are used to create the building blocks of other sequential circuits, i.e., latches and flip-flops)
 - 2. *clocked synchronous state machines:* use latches and flip-flops to build machines that examine their inputs and change their outputs according to a controlling clock signal
- The state changes of most sequential circuits occur at times specified by a free running *clock* signal.
- Figure 1 gives the timing diagrams and nomenclature for typical clock signals.

Nomenclature for Typical Clock Signals

- A clock is said to be *active high* if state changes occur at the clock's rising edge or when the clock is HIGH
- A clock is said to be *active low* if state changes occur at the clock's falling edge or when the clock is LOW
- Clock period is the time between successive transitions in the same directions
- clock frequency is the reciprocal of the period
- clock tick is the first edge or pulse in a clock period (sometimes the period itself)
- duty cycle: the percentage of time that the clock signal is at its asserted level
- Typical clock frequencies: from 5 MHZ to 500 MHz; maximum of 4 GHz
- Clock signals are obtained from a quartz-crystal oscillator

Nomenclature for Typical Clock Signals



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 1 : Clock signals: (a) active high; (b) active low

・ロト ・ 一下・ ・ ヨト ・ 日 ・

3

Introduction

Bistable Elements

Master/Slave S-R Flip-Flop Master/Slave J-K Flip-Flop Edge-Triggered J-K Flip-Flop

Bistable Element. Digital Analysis





Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 2 : A pair of inverters forming a bistable element

- One stable point for Q=LOW and Q_L=HIGH:
 - ► If $Q_{-L} = V_{IN1} = HIGH \Rightarrow$ $V_{OUT1} = LOW \Rightarrow V_{IN2} =$ LOW (since $V_{IN2} = V_{OUT1}$) ► $V_{IN2} = LOW \Rightarrow V_{OUT2} =$
 - $HIGH = V_{IN1}$
- Another stable point for Q=HIGH and Q_L=LOW:
 - ► If $Q_{-L} = V_{IN1} = LOW \Rightarrow$ $V_{OUT1} = HIGH \Rightarrow$ $V_{IN2} = HIGH$ (since
 - $V_{IN2} = V_{OUT1}$
 - ► $V_{IN2} = HIGH \Rightarrow$ $V_{OUT2} = LOW = V_{IN1}$

イロト 不得 トイヨト イヨト 二日

Bistable Element. Analog Analysis



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 3 : A pair of inverters forming a bistable element

- If we look at the analog values from the input-output transfer characteristic of a CMOS inverter (see next slide), there is a third stable point for Q=Q_L=2.5 V:
 - ▶ If $Q_{-L} = V_{IN1} = 2.5V \Rightarrow V_{OUT1} = 2.5V$ (from the transfer characteristic) ⇒ $V_{IN2} = 2.5V$ (since $V_{IN2} = V_{OUT1}$) ▶ $V_{IN2} = 2.5V \Rightarrow V_{OUT2} = 2.5V = V_{IN1}$ (from the transfer characteristic)

The 2.5 V value of the outputs is not ok, because it is not in the range of valid digital values !! (the circuit will have an unpredictable behavior)

Input-Output Transfer Characteristic of a CMOS Inverter



Figure 4 : Input-output transfer characteristic of a CMOS inverter

< ロ > < 同 > < 回 > < 回 >

Bistable Element. Analog Analysis



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 5 : A pair of inverters forming a bistable element

 However, the third stable point is not so stable, we call it metastable.

- If a small noise (say 0.01 V) adds to the 2.5 V of V_{IN1} we have:
 - If $V_{IN1} = 2.51V$
 - From the transfer characteristic of a CMOS gate it results that
 - $V_{OUT1} = 2.0 V \Rightarrow V_{IN2} = 2.0 V$ (since

$$V_{IN2} = V_{OUT1}$$
)

- ► V_{IN2} = 2.0V, from the transfer characteristic ⇒ V_{OUT2} = 4.8V = V_{IN1}
- ► $V_{IN1} = 4.8V$, from the transfer characteristics $\Rightarrow V_{OUT2} = 0.0V =$ $V_{IN1} \Rightarrow V_{OUT1} = 5.0V$

Bistable Element. Analog Analysis

- Hence, a small increase (0.01 V) of the V_{IN1} value brings the circuit from the metastable point to the stable point where Q=HIGH and Q_L=LOW
- In a similar way we can show that a small decrease of the V_{IN1} value brings the circuit from the metastable point to the stable point where Q=LOW and Q_L=HIGH
- Next slide presents the same analysis, considering the transfer functions of the two inverter represented on the same figure.
- The three intersection points of the two curves are the three stable points (actually two stable points and one metastable point)
- We can make an analogy with a ball and a hill: the metastable position is when the ball is on top of the hill because the ball can fall on either the right or the left part of the hill.

Analog Analysis of the Bistable Element



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 6 : Transfer functions for inverters in a bistable feedback loop

▲ロ ▶ ▲周 ▶ ▲ 国 ▶ ▲ 国 ▶ ● の Q @

Metastable Behavior



Figure 7 : Ball and hill analogy for metastable behavior

Metastable Behavior

- If we want to move the ball from the left-side of the hill to the right-side, we have to apply a force
 - 1. If the force is too small the ball will return to the initial position
 - 2. If the force is medium, the ball will go on the top of the hill, in the metastable position
 - 3. If the force is big enough, the ball will go to the other side of the hill, in a stable position
- We can make an analogy for circuits, when we want to change the state of the circuit:
 - 1. If the input signal is too short (*pulse width* too short), the circuit will ignore it (this is not dangerous)
 - 2. If the pulse width is longer, but not long enough (shorter than the minimum specified pulse width), the circuit will go to the metastable position (this is dangerous !)
 - 3. If the input signal is long enough (i.e. \geq minimum pulse width), the circuit will change state (will go to the other stable point) (this is what we want)

Metastability

- All sequential circuits are subject to metastability (because they incorporate the bistable element).
- > We want to avoid the metastable state of a sequential circuit
- In order to avoid the metastable state, it is important that "analog" (i.e., not the clock) input signals of the latches and flip-flops meet the setup and hold times (will be discussed)
- The problem is more severe as the system's speed increase
- Many designs, designers and companies failed because of the metastability problem !

The bistable element cannot be controlled (it has no inputs). In order to control it, the inverters are replaced with NOR or NAND gates, obtaining latches (and flip-flops).

Introduction

Bistable Elements

Latches and Flip-Flops

Master/Slave S-R Flip-Flop Master/Slave J-K Flip-Flop Edge-Triggered J-K Flip-Flop

Latches and Flip-Flops. Introduction

- Latches and flip-flops are the basic building blocks of most sequential circuits.
- Definition: A flip-flop is a sequential circuit that normally samples its inputs and changes its outputs only when a clocking signal is changing (from LOW to HIGH for positive edge triggered flip-flops, or from HIGH to LOW for negative-edge triggered flip-flops).
- A latch is a sequential device that watches its inputs continuously and can change its outputs at any time (in some cases requiring the enable input to be asserted).
- We can consider that a device that watches its inputs and changes its outputs during the HIGH (or LOW) value of the clocking signal to be a latch.
- In the following subsections we discuss the most used types of latches and flip-flops.

Introduction

Bistable Elements

Latches and Flip-Flops

S-R Latch Master/Slave S-R Flip-Flop Master/Slave J-K Flip-Flop Edge-Triggered J-K Flip-Flop

- It has two inputs, S (set) and R (reset), and two outputs Q and QN (normally QN=Q') (see fig 8 (a))
- ► If S=R=0, the circuit behaves like the bistable element: it retains either the state LOW (Q=0), or the state HIGH (Q=1) (see fig 8 (b))
- Inputs S and R may force the feedback loop to a desired state (see fig 9 (a)):
 - ▶ S=1 sets (or presets) the Q output to 1;
 - R=1 resets (or clears) the output Q to 0;
 - After the input S or R is negated, the latch remains in the state into which it was forced
 - Figure 9 (a) shows the functioning of S-R latch for a typical set of inputs.

 Colored arrows indicate causality (which input transitions cause which output transitions

- ► If S=R=1, both Q=0 and QN=0 (see figure 9 (b));
- After that, if one input (either S or R) is negated, the outputs will be again complementary
- But, if both S and R are negated simultaneously, then the latch outputs will oscillate (metastability)
- Metastability may also occur if a 1 pulse applied to S or R is too short.

Figure 10 shows different symbols for the S-R latch. Last symbol is wrong.



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 8 : S-R latch: (a) circuit design using NOR gates; (b) function table

・ロト ・ 理 ト ・ ヨ ト ・ ヨ ト

æ



Figure 9 : Typical operation of an S-R latch: (a) "normal" inputs; (b) S and R asserted simultaneously

◆□▶ ◆□▶ ◆三▶ ◆三▶ - 三 - のへぐ



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 10 : Symbols for an S-R latch: (a) without bubble; (b) preferred for bubble to bubble design; (c) incorrect because of double negation

< 日 > < 同 > < 回 > < 回 > < 回 > <

3

S-R Latch: Timing Parameters

- Propagation delay is the time it takes for a transition on an input signal to produce a transition on an output signal.
- A latch or flip-flop can have several different propagation delay specifications, one for each pair of input and output signals
- The propagation delays can be different if the output makes a LOW-to-HIGH transition or a HIGH-to-LOW transition. In figure 11 we see for the S-R latch:
 - ► the propagation delay t_{pLH(SQ)}: a LOW to HIGH transition of Q caused by a LOW to HIGH transition on S
 - ▶ the propagation delay t_{pHL(RQ)}: a HIGH-to-LOW propagation of Q due to a LOW-to-HIGH transition on R
- ► The transitions on QN t_{pHL(SQN)} and t_{pLH(RQN)} are not shown in the figure.
- Minimum pulse width specifications are given for S and R:
- If a pulse shorter than the minimum pulse width t_{pw(min)} is applied to S or R, the latch may go into the metastable state and remain there a random period of time
- ► An S of R pulse that is $\geq t_{pw(min)}$ will bring the latch out of the metastable state.



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 11 : Timing parameters for an S-R latch

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 三臣 - のへで

Introduction

Bistable Elements

Latches and Flip-Flops

Š-**R** Latch Master/Slave S-R Flip-Flop Master/Slave J-K Flip-Flop Edge-Triggered J-K Flip-Flop

◆□ > ◆□ > ◆臣 > ◆臣 > ─ 臣 ─ のへで

Š-R Latch

- Using NAND gates instead NOR gates, a S-R
 Iatch can be build (fig 12) (read S bar R bar)
- In TTL and CMOS technologies, the S-R
 Iatches are more used than S-R latches because NAND gates are preferred over NOR gates
- ▶ Operation of S̄-R̄ latch is quite similar to that of the S-R latch, but the inputs are active low, and, when both inputs are asserted, Q=QN=1 (not 0, like for S-R latch)
- In rest, the operation of the two latches is the same (e.g., for timing and metastability).

Š-**R** Latch

Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e



Figure 12 : \bar{S} - \bar{R} latch: a) circuit design using NAND gates; (b) function table; (c) logic symbol

・ロト ・ 一下・ ・ ヨト ・ 日 ・

= √Q (~

Introduction

Bistable Elements

Latches and Flip-Flops

S-R Latch S-R Latch

S-R Latch with Enable

D Latch Edge-Triggered D Flip-Flop Edge-Triggered D Flip-Flop with Enable Scan Flip-Flop Master/Slave S-R Flip-Flop Master/Slave J-K Flip-Flop Edge-Triggered J-K Flip-Flop T Flip-Flop

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三回 ● のへで

S-R Latch with Enable

- An enable input C is added to the S-R latch, so that the latch will be sensitive to S and R only when C is asserted.
- ▶ When C=1 the circuit behaves like a S-R latch
- ▶ When C=0, the latch retains its previous state
- The latch can become unpredictable (metastable) if S=R=1 when C changes from 1 to 0 (it is like both S and R were negated simultaneously)

A D M 4 目 M 4 日 M 4 1 H 4

S-R Latch with Enable

Copyright © 2000 by Prentice Hall, Inc.



Figure 13 : S-R latch with enable: a) circuit design using NAND gates; (b) function table; (c) logic symbol

▲ロ ▶ ▲周 ▶ ▲ 国 ▶ ▲ 国 ▶ ● の Q @

S-R Latch with Enable



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 14 : Typical operation of an S-R latch with enable

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 三臣 - のへで

Introduction

Bistable Elements

Latches and Flip-Flops

D L atch Master/Slave S-R Flip-Flop Master/Slave J-K Flip-Flop Edge-Triggered J-K Flip-Flop

D Latch

- S-R latches are more useful in control applications, when we want to set and reset a flag independently
- If we simply want to store bits of information (each bit being presented on a single signal line), a D latch will be more appropriate
- Figure 15 shows a D latch (implementation, function table, logic symbol)
- The inverter that generates the S and R inputs from the single data input D eliminates the problem of S-R latches caused by asserting S and R are simultaneously
- ► The control input (labeled C in fig 15) is sometimes named ENABLE, *CLK*, or G.
- The control input can be active high (like in fig 15), or active low.
- The control input always has a minimum pulse-width requirement
D Latch



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 15 : D latch: a) circuit design using NAND gates; (b) function table; (c) logic symbol

・ロト ・ 理 ト ・ ヨ ト ・ ヨ ト

= nac

D Latch. Functional Behavior

- ► The functional behavior of the D latch is given in figure 16
- When C is asserted (C=1), the Q output follows the D input
- The latch is said to be "open" and the path from D to Q is "transparent"
- For this reason, this type of latch is called also a *transparent latch*)
- ▶ When C=0, the latch "closes": Q retains its previous value and do not change in response to D (until C will be again 1)

D Latch. Functional Behavior



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 16 : Functional behavior of a D latch for various inputs

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 三臣 - のへで

D Latch. Timing Parameters

- The timing behavior of the D latch is shown in figure 17
- For the Q output we can define four parameters, depending on the transition (LOW-to-HIGH or HIGH-TO-LOW) and on the input signal that causes the transition:
 - ▶ When C=0 (the latch is closed) and D input is opposite of Q, when C changes to 1 the latch will be open and Q will change after the delay t_{pLH(CQ)} or t_{pHL(CQ)} (transitions 1 and 4 in fig 17)
 - ▶ When C=1 (the latch is open), Q follows the transitions on D with delays t_{pHL(DQ} and t_{pLH(DQ} (transitions 2 and 3)

D Latch. Setup and Hold Times

- The D latch eliminates the S=R=1 problem, but not the metastability problem.
- There is a time window D around the falling (latching) edge of C when the input must not change (shaded in fig 17)
- The window consists of
 - the setup time (t_{setup}): the D input must be stable (i.e., must not change) at least t_{setup} time units before the latching edge of C
 - the hold time t_{hold}: the input D must not change at least t_{hold} time units after the latching edge of C

If the setup or hold time are not met, the latch may become metastable.

D Latch. Timing Parameters



Figure 17 : Timing parameters for a D latch

◆□ > ◆□ > ◆豆 > ◆豆 > ̄豆 = つへで

Outline

Introduction

Bistable Elements

Latches and Flip-Flops

S-R Latch S-R Latch S-R Latch with Enable D Latch

Edge-Triggered D Flip-Flop

Edge-Triggered D Flip-Flop with Enable Scan Flip-Flop Master/Slave S-R Flip-Flop Master/Slave J-K Flip-Flop Edge-Triggered J-K Flip-Flop T Flip-Flop

Positive-Edge-Triggered D Flip-Flop

- ► A positive-edge-triggered D flip-flop is shown in figure 18
- It samples the D input only and changes its output only at the rising edge of a controlling CLK signal
- It consists of two D latches, the first being called the *master*, and the second the *slave*
- When CLK=0, the master latch is open and follows the input
- When CLK goes to 1, the master latch closes and its output is transferred to the slave latch
- The slave latch is open all the time when CLK=1, but it changes only at the beginning of this interval (that means, only at the rising edge of CLK), because the master is closed and unchanging for the rest of the interval
- The triangle on the CLK input on the symbol of the D flip-flop is called *dynamic-input indicator* and it indicates edge-triggered behavior

Positive-Edge-Triggered D Flip-Flop



Figure 18 : Positive-edge-triggered D flip-flop: a) circuit design using D latches; (b) function table; (c) logic symbol

・ロト ・ 理 ト ・ ヨ ト ・ ヨ ト

3

Positive-Edge-Triggered D Flip-Flop. Functional Behavior

- Figure 19 presents the functional behavior of the D positive-edge-triggered flip-flop
- The signal QM is the output of the master latch
- QM changes only when CLK=0
- ▶ When CLK goes to 1, the current value of QM is transferred to Q and QM will not change again as long as CLK=1

A D M 4 目 M 4 日 M 4 1 H 4

Positive-Edge-Triggered D Flip-Flop. Functional Behavior



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

▲□▶ ▲□▶ ▲□▶ ▲□▶ □ のQ@

Figure 19 : Functional behavior of a positive-edge-triggered D flip-flop

Positive-Edge-Triggered D Flip-Flop. Timing Parameters

- Timing parameters for the D flip-flop are shown in figure 20
- All propagation delays are measured from the rising edge of the CLK, since this is the only event that causes the change of Q.
- ► There will be only two delays for Q: LOW-to-HIGH (t_{pLH(CQ}) and HIGH-to-LOW (t_{pHL(CQ}))
- The D flip-flop has also setup and hold time-windows, during which the D must not change
- The setup and hold time window occurs around the rising edge of the CLK (shadowed in fig 20)
- If the setup (t_{setup}) or hold time (t_{hold}) are not met, the flip-flop can go either to a stable, but unpredictable state, or it can go to a metastable state
- From the metastable state the flip-flop will return either on its own, after a probabilistic delay, or at a new clock edge, if the setup and hold times are met.

Positive-Edge-Triggered D Flip-Flop. Timing Parameters



Figure 20 : Timing behavior of a positive-edge-triggered D flip-flop

▲ロ ▶ ▲周 ▶ ▲ 国 ▶ ▲ 国 ▶ ● の Q @

Negative-Edge-Triggered D Flip-Flop



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 21 : Negative-edge-triggered D flip-flop: a) circuit design using D latches; (b) function table; (c) logic symbol

The negative-edge-triggered D flip-flop simply inverts the clock input, such that the active edge is the falling edge of the clock. By convention, the falling edge is considered to be active low, hence the inverting bubble on the CLK input of the circuit's symbol.

Positive-Edge-Triggered D Flip-Flop with Preset and Clear

- It is useful for the flip-flops to have asynchronous inputs, that can be used to force the flip-flop to a particular state, independent of the CLK and D inputs
- The PR (*preset*) input can force the flip-flop to 1, while the CLR (*clear*) can force the flip-flop to 0.
- PR and CLR inputs behave like the set and reset inputs of an S-R latch
- They are called asynchronous because do not depend on CLK
- Asynchronous inputs can be used to set or reset the output of a flip-flop (not very recommended), but their main purpose is to force a sequential circuit to a known initial (starting) state
- Such a circuit is shown in figure 22, while figure 23 shows the implementation of the commercial TTL circuit 74LS74 (positive-edge-triggered D flip-flop).
- The circuit from fig 23 has fewer gates than the circuit from fig 22 (hence it is cheaper and faster).

Positive-Edge-Triggered D Flip-Flop with Preset and Clear



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 22 : Positive-edge-triggered D flip-flop with preset and clear: a) circuit design; (b) function table; (c) logic symbol

Commercial TTL Circuit for a Positive-Edge-Triggered D Flip-Flop



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

▲ロ ▶ ▲周 ▶ ▲ 国 ▶ ▲ 国 ▶ ● の Q @

Figure 23 : Commercial TTL circuit for a positive-edge-triggered D flip-flop such as 74LS74

Outline

Introduction

Bistable Elements

Latches and Flip-Flops

Edge-Triggered D Flip-Flop with Enable Master/Slave S-R Flip-Flop Master/Slave J-K Flip-Flop Edge-Triggered J-K Flip-Flop

Positive-Edge-Triggered D Flip-Flop with Enable

- Sometimes we want that the D flip-flop to hold the stored value, instead of loading a new one, at the (rising) edge of the clock
- This can be achieved by adding an *enable input* EN, or *clock enable* CE.
- The name clock enable is misleading, because the implementation will not control the clock (by gating the clock)
- In general it is strongly recommended to avoid controlling (gating) the clock in a sequential circuit, because this can lead to the lose of synchronization within the circuit.
- Instead, a two-input multiplexer controls the value applied to the internal D flip-flop (see fig 24)
- ► If EN=1, the external D input is selected, if EN=0, the flip-flop's current output is selected.

Positive-Edge-Triggered D Flip-Flop with Enable



Figure 24 : Positive-edge-triggered D flip-flop with enable: a) circuit design; (b) function table; (c) logic symbol

< ロ > < 同 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ >

э

Outline

Introduction

Bistable Elements

Latches and Flip-Flops

S-R Latch S-R Latch S-R Latch with Enable D Latch Edge-Triggered D Flip-Flop Edge-Triggered D Flip-Flop with Enabl Scan Flip-Flop

◆□ > ◆□ > ◆臣 > ◆臣 > ─ 臣 ─ のへで

Master/Slave S-R Flip-Flop Master/Slave J-K Flip-Flop Edge-Triggered J-K Flip-Flop T Flip-Flop

Scan Flip-Flop. Motivation

- A sequential circuit has a combinational part and a pure sequential part (i.e., flip-flops).
- ► In order to completely test a combinational circuit with n inputs, we must apply all possible input combinations, i.e., 2ⁿ input combination
- ► If the circuit has also m flip-flops (m state variables) it means that is has also 2^m states.
- ▶ Which means that, if we want to completely test the circuit, we must apply the 2ⁿ inputs for *each* of the 2^m states, resulting a far too long test sequence
- A solution is provided within the frame of *design for testability* (DFT): when we design a circuit we must consider the fact that the circuit should be tested, and to design the circuit such that to ease the testing
- In the case of sequential circuits, a DFT method is to separate the combinational and the sequential parts, and to test them separately

Positive-Edge-Triggered D Flip-Flop with Scan

- The scan flip-flops can work in either test mode, or normal mode
- The selection between the two modes is performed by the TE (*test enable*) input at the input multiplexer of the scan flip-flop
- ▶ When TE=0, the flip-flop behaves like an ordinary D flip-flop
- ▶ When TE=1, the flip-flop takes data from TI (*test input*)
- The extra inputs are used to connect all flip-flops from an ASIC in a *scan chain* (for testing)
- Figure 26 shows such a scan chain consisting of four flip-flops: the TE inputs of all flip-flops are connected together, while the Q output of a flip-flop is connected to the TI of the next flip-flop from the scan chain, in a serial (*daisy chain*) way
- The TE, TI and TO connections are used only for test purposes
- The combinational logic around the scan flip-flops is not shown here

Scan Flip-flops. Test Procedure

- The TE is set to 1 and data (i.e., a *n*-bits test vector) is scanned in (serially) into the scan chain in *n* clock ticks (in fig 26 n = 4)
- Then TE is negated and the circuit is allowed to run one or more additional clock ticks
- ► The new state of the circuit (i.e., the new values of the n flip-flops) can be scanned out and observed at TO (making TE=1 again) in another n clock ticks
- During the *n* clock ticks used for scanning out the internal state of the circuit, a new *n*-bits test vector can be scanned in into the scan chain
- The scan capability can be added to D flip-flops with enable (*how* ?), or applied to other types of flip-flops (R-S, J-K, T – see next subsections)

Positive-Edge-Triggered D Flip-Flop with Scan



< ロ > < 同 > < 回 > < 回 > < □ > <

э

Figure 25 : Positive-edge-triggered D flip-flop with scan: a) circuit design; (b) function table; (c) logic symbol

Scan Chain



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

◆□▶ ◆□▶ ◆三▶ ◆三▶ - 三 - のへぐ

Figure 26 : A scan chain with four flip-flops

Outline

Introduction

Bistable Elements

Latches and Flip-Flops

Master/Slave S-R Flip-Flop Master/Slave J-K Flip-Flop Edge-Triggered J-K Flip-Flop

- If we replace in figure 21 (negative edge-triggered D flip-flop) the D latches with S-R latches, we obtain a master/slave S-R flip-flop, shown in figure 27
- Similar with the D flip-flop from fig 21, the S-R master/slave flip-flop changes its output only at the falling edge of the clock signal C
- But the new output depends not only of the input values at the falling edge (like for the D flip-flop), but during the entire interval when C=1 before the falling edge
- As shown on figure 28, a pulse on S any time in the interval when C=1 sets the master latch, and a pulse on R resets the master latch
- The value stored in the slave latch at the falling edge depends if the master latch was set or reset last time
- The circuit behaves more like a latch that follows its inputs for the entire interval when C=1, but stores them only when C becomes 1

- The symbol of this flip-flop (see fig 27 (c)) is not using the dynamic indicator, because the flip-flop is not truly edge triggered
- Instead, a postponed output indicator is used to indicate that the output value does not change until the input C is negated
- The circuit is sometimes called pulse-triggered flip-flop
- ► If both S and R are asserted at the falling edge of C: just before the falling edge S=R=1 will make QM = QM_L = 1,

When C becomes 0 the master latch's output becomes unpredictable and then propagates to the slave's output

Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e



・ロト ・ 一下・ ・ ヨト ・ 日 ・

3

Figure 27 : Master/Slave S-R flip-flop: a) circuit design using S-R latches; (b) function table; (c) logic symbol



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

◆□▶ ◆□▶ ◆三▶ ◆三▶ - 三 - のへぐ

Figure 28 : Internal and functional behavior of a master/slave S-R flip-flop

Outline

Introduction

Bistable Elements

Latches and Flip-Flops

Master/Slave S-R Flip-Flop Master/Slave J-K Flip-Flop Edge-Triggered J-K Flip-Flop

- The problem of S and R asserted simultaneously is solved in a master/slave J-K flip-flop (fig 29):
- Asserting J asserts the master's S only if QN=1 (i.e., Q=0), and asserting K asserts master's R only if Q=1
- Thus, if J=K=1, the flip-flop will go to the opposite of its current state
- Figure 30 shows the functional behavior of the J-K master/slave flip-flop
- If we look at the last two events we see that the current value of Q (and QN) can modify the desired behavior of the master/slave flip-flop:
- ► The output Q changes to 1 when K=1 (not when J=1) at the one before last event; this behavior is called 1 catching
- The output Q changes to 0 when J=1 (not when K=1) at the last event; this is called 0 catching
- ► The J and K inputs of this flip-flop must be held valid during the entire interval when C=1

```
Master/Slave J-K Flip-Flop
```



Figure 29 : Master/Slave J-K flip-flop: a) circuit design using S-R latches; (b) function table; (c) logic symbol

< ロ > < 同 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ >

э



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

▲□▶ ▲□▶ ▲□▶ ▲□▶ □ のQ@

Figure 30 : Internal and functional behavior of a master/slave J-K flip-flop

Outline

Introduction

Bistable Elements

Latches and Flip-Flops

Master/Slave S-R Flip-Flop Master/Slave J-K Flip-Flop Edge-Triggered J-K Flip-Flop
Edge-Triggered J-K Flip-Flop

- The problem of 0s and 1s catching is solved in th edge-triggered J-K flip-flop, shown in figure 31
- The inputs are sampled at the rising edge of the clock
- ► Next output (denoted Q*) has the characteristic equation Q* = J · Q' + K' · Q
- Typical functional behavior of the J-K edge-triggered flip-flop is shown in figure 32
- The J and K inputs must meet the published setup- and hold-time specifications with respect to the triggering clock edge
- The J-K edge-triggered flip-flops replaced the R-S and J-K master/slave flip-flops and were very popular in the past
- ► Figure 33 shows the TTL circuit 74LS109, J-K flip-flop
- Today the J-K flip-flops are not so much used, being replaced in integrated circuits by D edge-triggered flip-flops

Edge-Triggered J-K Flip-Flop

Copyright © 2000 by Prentice Hall, Inc.



Figure 31 : Edge-triggered J-K flip-flop: a) equivalent function using an edge-triggered D flip-flop; (b) function table; (c) logic symbol

イロト 不同 トイヨト イヨト

-

Edge-Triggered J-K Flip-Flop



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 32 : Functional behavior of a positive-edge-triggered J-K flip-flop

Commercial Circuit for a Positive-Edge-Triggered J- \overline{K} Flip-Flop



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 33 : Internal logic diagram for the 74LS109 positive-edge-triggered J- \overline{K} flip-flop

Outline

Introduction

Bistable Elements

Latches and Flip-Flops

Master/Slave S-R Flip-Flop Master/Slave J-K Flip-Flop Edge-Triggered J-K Flip-Flop T Flip-Flop

Edge-Triggered T Flip-Flop

- ► A T (toggle) flip-flop changes state on every clock tick
- Figure 34 shows the symbol and the behavior of a positive-edge-triggered flip-flop
- The output Q of a T flip-flop has the half of the frequency of the T input
- Figure 35 shows how to obtain a T flip-flop from a D or J-K flip-flop
- The T flip-flops are used most often for counters and frequency dividers
- If the T flip-flop need not be toggled on every clock tick, it can be modified to obtain a *T flip-flop with enable* (figures 36 and 37), which toggles only if the enable input EN=1
- EN input must meet setup and hold times specified for the circuit



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

・ロト ・ 一下・ ・ ヨト

-

Figure 34 : Positive-edge-triggered T flip-flop: a) logic symbol; (b) functional behavior



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 35 : Possible circuit designs for a T flip-flop: a) using a D flip-flop; (b) using a J-K flip-flop

Edge-Triggered T Flip-Flop with Enable



Digital Design Principles and Practices, 3/e

Figure 36 : Positive-edge-triggered T flip-flop with enable: a) logic symbol; (b) functional behavior



Copyright © 2000 by Prentice Hall, Inc. Digital Design Principles and Practices, 3/e

Figure 37 : Possible circuit designs for a T flip-flop with enable: a) using a D flip-flop; (b) using a J-K flip-flop