EEE 4483

Digital Electronics and Pulse Techniques

Lecture 2

Logic Families

Nadim Ahmed
Lecturer, Department of EEE
Islamic University of Technology (IUT)

Referred Textbooks

Digital Electronics: Principles, Devices and Applications
by Anil K. Maini

Digital Integrated Electronics
by Herbert Taub and Donald Schilling

There are a variety of circuit configurations or more appropriately various approaches used to produce different types of digital integrated circuit. Each such fundamental approach is called a **logic family**.

The idea is that different logic functions, when fabricated in the form of an IC with the same approach, or in other words belonging to the same logic family, will have identical electrical characteristics. These characteristics include supply voltage range, speed of response, power dissipation, input and output logic levels, current sourcing and sinking capability, fan-in, fan-out, noise margin, etc. In other words, the set of digital ICs belonging to the same logic family are electrically compatible with each other.

Types of Logic Family

The entire range of digital ICs is fabricated using either bipolar devices or MOS devices or a combination of the two. Different logic families falling in the first category are called bipolar families, and these include –

- Diode Logic (DL)
- Resistor Transistor Logic (RTL)
- Diode Transistor Logic (DTL)
- Transistor Transistor Logic (TTL)
- Emitter Coupled Logic (ECL) – also known as Current Mode Logic (CML)
- Integrated Injection Logic (I^2L)

The logic families that use MOS devices as their basis are known as MOS families, and the prominent members belonging to this category are –

- PMOS family (using P-channel MOSFETs)
- NMOS family (using N-channel MOSFETs)
- CMOS family (using both N- and P-channel devices)

The Bi-MOS logic family uses both bipolar and MOS devices. Of all the logic families listed above, the first three, that is, diode logic (DL), resistor transistor logic (RTL) and diode transistor logic (DTL), are of historical importance only. Diode logic used diodes and resistors and in fact was never implemented in integrated circuits. The RTL family used resistors and bipolar transistors, while the DTL family used resistors, diodes and bipolar transistors. Both RTL and DTL suffered from large propagation delays owing to the need for the transistor base charge to leak out if the transistor were to switch from conducting to nonconducting state.

The DL, RTL and DTL families, however, were rendered obsolete very shortly after their introduction in the early 1960s owing to the arrival on the scene of transistor transistor logic (TTL).

Logic families that are still in widespread use include TTL, CMOS, ECL, NMOS and Bi-CMOS. The PMOS and I^2L logic families, which were mainly intended for use in custom large-scale integrated (LSI) circuit devices, have also been rendered more or less obsolete, with the NMOS logic family replacing them for LSI and VLSI applications.

Each logic family has multiple specialized **subfamilies** that are optimized to meet specific requirements such as lower power consumption, higher speed, or improved input/output compatibility. Within each logic family, different subfamilies are developed to address distinct performance or application needs. These logic subfamilies allow circuit designers to make informed trade-offs between speed, power, noise margins, and interface logic levels based on the demands of their application.

How to Read a Digital Logic IC

Let us now examine how logic ICs from the 74-series are identified using a structured alpha-numeric code, typically printed directly on the IC package.

This identification code is composed of up to five segments –

1. Manufacturer Prefix (optional)
2. Family Code (2 digits)
3. Subfamily Code (0 to 3 letters)
4. Function Code (2 to 5 digits)
5. Package Suffix (optional)

1. Manufacturer Prefix (optional)

The IC code often begins with a two-letter prefix that indicates the name of the manufacturer. This prefix is not part of the logic designation itself but is included to identify the origin of the device. Table 2 lists common manufacturer prefixes.

For example, in the IC code SN74LS32N, the prefix “SN” denotes that the device was manufactured by Texas Instruments.

Basic 74-series code

AA 74XXX000 ZZ

MANUFACTURERS PREFIX
e.g.,
CD = RCA
DM = National Semiconductor
MC = Motorola
MM = National Semiconductor
N = Signetics
PC = Signetics
SN = Texas Instruments

MANUFACTURERS SUFFIX
e.g.,
N = Plastic dual-in-line package
P = Eight-pin plastic D.I.L. package
J = Ceramic package

74 XXX 000

FAMILY (2 digits).
e.g., 74 = Commercial grade
54 = Military grade
75 = Interface device

FUNCTION
(2 to 5 digits)

SUB-FAMILY (0 to 3 letters)
e.g., blank or N = Standard TTL
L = Low-power TTL
H = High-power TTL
S = Schottky TTL
LS = Low-power Schottky
AS = Advanced Schottky
ALS = Advanced Low-power Schottky TTL
F = FAST TTL (Fairchild ALS TTL)
C = CMOS version of TTL device
HC = High-speed CMOS, with CMOS-compatible inputs
HCT = High-speed CMOS, with TTL-compatible inputs
AC = Advanced high-speed CMOS
ACT = Advanced high-speed CMOS with TTL-compatible inputs

Fig 1: Interpretation of 74-series logic IC codes

Table 1: Common manufacturer prefixes for 74-series ICs

Prefix	**Manufacturer**
SN	Texas Instruments
DM, MM	National Semiconductor
MC	Motorola
CD	RCA
N, PC	Signetics

2. Family Code (2 Digits)

The first numerical portion of the IC code typically consists of two digits (74, 54, or 75) and denotes the device's grade and intended application environment.

- **74:** Indicates a commercial-grade device. These ICs are designed for general-purpose applications, are typically housed in plastic dual-in-line packages (DIPs) and operate reliably within a temperature range of 0°C to +70°C. The allowable supply voltage typically ranges from +4.75 V to +5.25 V .
- **54:** Denotes a military-grade device, characterized by robust construction and extended temperature tolerance from −55°C to +125°C. These ICs may be packaged in ceramic or hermetically sealed metal containers and can operate on supply voltages from +4.5 V to +5.5 V .
- **75:** Represents interface logic components intended to operate in conjunction with 74-series ICs, often serving as level converters or buffers.

In the code SN74LS32N, the digits “74” confirm that this IC is part of the commercial-grade TTL family.

3. Subfamily Code (0 to 3 Letters)

Following the family code, the IC may include up to three alphabetical characters that define the **logic subfamily** or **technology variant** used in the device. Each subfamily is optimized for a specific set of parameters, such as speed, power consumption, and compatibility.

Table 2: Common subfamily codes and their characteristics

Code	Subfamily	Logic Type	Description
N	Standard TTL	Bipolar	Basic TTL characteristics
L	Low-power TTL	Bipolar	Reduced power consumption; slower switching
H	High-speed TTL	Bipolar	Higher speed; increased power dissipation
S	Schottky TTL	Bipolar + Schottky	Uses Schottky clamping for faster switching
LS	Low-power Schottky	Bipolar + Schottky	Balanced speed and power characteristics
ALS	Advanced Low-power Schottky	Enhanced Bipolar	Improved LS with faster switching and lower power
HC	High-speed CMOS	CMOS	Fast switching; CMOS voltage levels
HCT	TTL-compatible CMOS	CMOS	CMOS switching with TTL compatible input thresholds
AC	Advanced high-speed CMOS	CMOS	Faster than HC; capable of driving large loads
ACT	TTL-compatible advanced CMOS	CMOS	AC speed with TTL-compatible inputs

Each subfamily brings distinct performance trade-offs. For instance, LS devices offer lower power consumption than S devices while maintaining acceptable speed, whereas HC and HCT devices offer the power efficiency of CMOS technology with different input level compatibilities.

In the IC SN74LS32N, the subfamily code “LS” identifies it as a low-power Schottky TTL device.

4. Function Code (2 to 5 Digits)

The function code portion of the IC name specifies the logical operation performed by the IC. This segment generally consists of two to five digits. Each function code corresponds to a specific logic function such as gates, decoders, counters, or flip-flops.

Table 3: Example function codes

Code	Function
00	Quad 2-input NAND gate
08	Quad 2-input AND gate
32	Quad 2-input OR gate
74	4-bit binary counter
138	3-to-8 decoder/demultiplexer
161	4-bit synchronous binary counter

In the example SN74LS32N, the code “32” signifies that the IC contains four 2-input OR gates.

5. Package Suffix (Optional)

The final letter(s) in the IC code designate the package type of the device. This determines how the IC can be mounted on a printed circuit board (PCB) and may also impact thermal and mechanical properties.

Table 4: Common package suffixes

Suffix	Package Type
N	Plastic dual-in-line package (DIP)
J	Ceramic DIP
P	8-pin plastic DIP

In the IC SN74LS32N, the suffix “N” confirms that the device is encapsulated in a plastic DIP suitable for through-hole mounting.

Logic Levels and Their Importance in Digital Circuits

In digital systems, information is represented using two discrete voltage levels, typically denoted as **logic 0 (LOW)** and **logic 1 (HIGH)**. The most intuitive way to distinguish between these two states might seem to be the use of a single threshold voltage: voltages below the threshold are interpreted as logic 0, while voltages above the threshold are interpreted as logic 1. However, this simplistic approach proves unreliable in real-world scenarios due to several practical limitations, such as –

- **Voltage noise** on signal lines due to electromagnetic interference, crosstalk, or switching transients
- **Voltage drops** in wires or traces due to resistance and current flow
- **Temperature-dependent behaviors** of semiconductor devices
- **Loading effects**, where the presence of other devices connected to the signal line affects voltage levels

Because of these factors, the voltage at a particular node in a circuit may not always remain precisely at the expected HIGH or LOW value. For instance, a logic HIGH signal that should ideally be at 5 V might occasionally dip to 4.2 V due to noise or load, while a logic LOW might not fall all the way to 0 V but instead settle at 0.6 V. If only a single threshold voltage were used (say, 2.5 V), such fluctuations could lead to ambiguous interpretations, causing incorrect logic decisions and unreliable circuit behavior.

To overcome this issue, defined **logic levels** with **noise margins** are introduced. These margins ensure that even if voltage levels shift slightly, the signal can still be interpreted correctly and safely.

Logic levels refer to defined voltage ranges that represent binary states in a digital circuit. Rather than assigning a single, fixed voltage to indicate logic 0 or logic 1, logic levels specify **safe voltage intervals** within which a signal can be confidently interpreted as either LOW or HIGH. These ranges take into account electrical noise, component tolerances, and signal degradation, allowing digital systems to operate reliably even under non-ideal conditions.

For example, a voltage between 0 V and 0.8 V might be considered a valid logic 0, while a voltage between 2 V and 5 V might be accepted as a logic 1. Voltages that fall in between these two regions (e.g., 0.8 V to 2 V) are undefined or indeterminate and may lead to unpredictable behavior. By clearly separating these ranges, logic levels provide a margin of safety and ensure consistent logic interpretation across interconnected digital devices.

To further enhance reliability, these voltage ranges are defined separately for inputs and outputs. That is, the voltages a device produces when it drives a signal (output logic levels) are specified differently from the voltages it expects when receiving a signal (input logic levels).

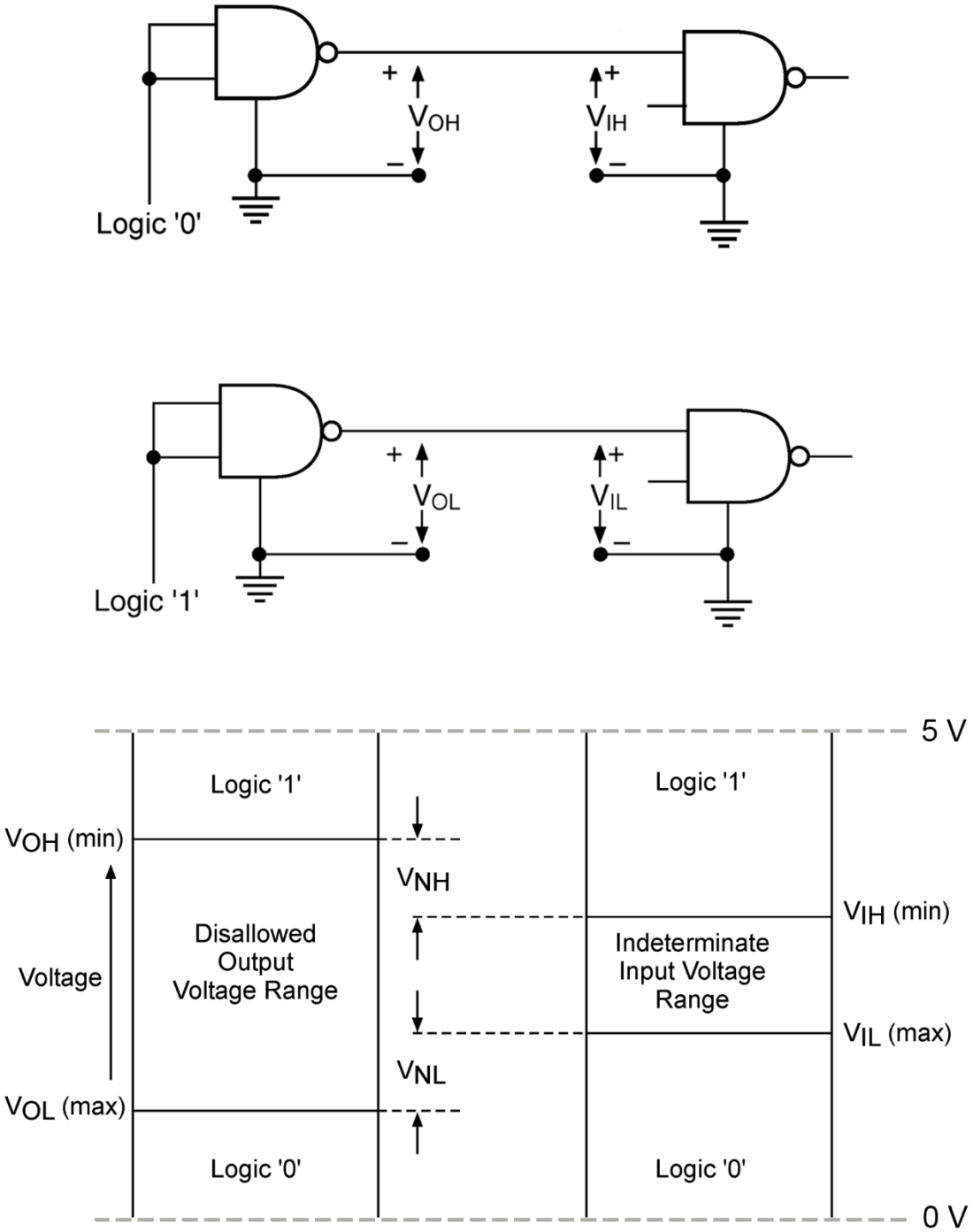


Figure 2: Parameters for defining logic levels

To distinguish between reliable HIGH and LOW voltages, both input and output logic levels are defined using four parameters:

HIGH-level output voltage (V_{OH} or $V_{OH}(min)$)
This is the minimum voltage level that a circuit guarantees when it outputs a logic 1 (HIGH). Any voltage $\geq V_{OH}$ is interpreted by the receiving device as a logic 1.

LOW-level output voltage (V_{OL} or $V_{OL}(max)$)
This is the maximum voltage level that a digital circuit guarantees when it outputs a logic 0 (LOW). Any voltage $\leq V_{OL}$ is interpreted as a logic 0.

HIGH-level input voltage (V_{IH} or $V_{IH}(min)$)
This is the minimum voltage that must be applied to a digital input for it to reliably recognize the signal as a logic 1.

LOW-level input voltage (V_{IL} or $V_{IL}(max)$)
This is the maximum voltage that a digital input can accept and still recognize the signal as a logic 0.

Noise Margin

This is a quantitative measure of noise immunity offered by the logic family. When the output of a logic device feeds the input of another device of the same family, a legal HIGH logic state at the output of the feeding device should be treated as a legal HIGH logic state by the input of the device being fed. Similarly, a legal LOW logic state of the feeding device should be treated as a legal LOW logic state by the device being fed.

We have seen earlier while defining important characteristic parameters that legal HIGH and LOW voltage levels for a given logic family are different for outputs and inputs. Figure 2 shows the generalized case of legal HIGH and LOW voltage levels for output.

As we can see from the diagram, there is a disallowed range of output voltage levels from V_{OL} to V_{OH} and an indeterminate range of input voltage levels from V_{IL} to V_{IH} . Since V_{IL} is greater than V_{OL} , the LOW output state can therefore tolerate a positive voltage spike equal to ($V_{IL} - V_{OL}$) and still be a legal LOW input. Similarly, V_{OH} is greater than V_{IH} , and the HIGH output state can tolerate a negative voltage spike equal to ($V_{OH} - V_{IH}$) and still be a legal HIGH input. Here, ($V_{IL} - V_{OL}$) and ($V_{OH} - V_{IH}$) are respectively known as the LOW-level noise margin (V_{NL}) and HIGH-level noise margin (V_{NH}).

Let us illustrate it further with the help of data for the standard TTL family. The minimum legal HIGH output voltage level (V_{OH}) in the case of the standard TTL is 2.4 V. Also, the minimum legal HIGH input voltage level (V_{IH}) for this family is 2 V. This implies that, when the output of one device feeds the input of another, there is an available margin of 0.4 V. That is, any negative voltage spikes of amplitude less than or equal to 0.4 V on the signal line do not cause any spurious transitions.

Similarly, when the output is in the logic LOW state, the maximum legal LOW output voltage level (V_{OL}) in the case of the standard TTL is 0.4 V . Also, the maximum legal LOW input voltage level (V_{IL}) for this family is 0.8 V . This implies that, when the output of one device feeds the input of another, there is again an available margin of 0.4 V . That is, any positive voltage spikes of amplitude less than or equal to 0.4 V on the signal line do not cause any spurious transitions. This leads to the standard TTL family offering a noise margin of 0.4 V .

To generalize, the noise margin offered by a logic family, as outlined earlier, can be computed from the HIGH-state noise margin, $V_{NH} = V_{OH} - V_{IH}$, and the LOW-state noise margin, $V_{NL} = V_{IL} - V_{OL}$. If the two values are different, the noise margin is taken as the lower of the two.

Characteristic Parameters of Logic Families

In this section, we will briefly describe the parameters used to characterize different logic families. We have already learned about the basic voltage-level parameters such as V_{OH}, V_{OL}, V_{IH}, and V_{IL}. Some of the other characteristic parameters, as we will see in the paragraphs to follow, are also used to compare different logic families.

HIGH-level input current, I_{IH}

This is the the maximum current flowing into (taken as positive) or out of (taken as negative) an input when a HIGH-level input voltage equal to V_{OH} specified for the family is applied.

LOW-level input current, I_{IL}

The LOW-level input current is the maximum current flowing into (taken as positive) or out of (taken as negative) the input of a logic function when the voltage applied at the input equals to V_{OL} specified for the family.

HIGH-level and LOW-level input current or loading are also sometimes defined in terms of unit load (UL). For devices of the TTL family, 1 UL (HIGH) = 40 µA and 1 UL (LOW) = 1.6 mA.

HIGH-level output current, I_{OH}

This is the maximum current flowing out of an output when the input conditions are such that the output is in the logic HIGH state. It tells about the current sourcing capability of the output. The magnitude of I_{OH} determines the number of inputs the logic function can drive when its output is in the logic HIGH state. For example, for the standard TTL family, the minimum guaranteed I_{OH} is -400μA, which can drive 10 standard TTL inputs with each requiring 40μA in the HIGH state, as shown in Fig. 3(a).

LOW-level output current, I_{OL}

This is the maximum current flowing into the output pin of a logic function when the input conditions are such that the output is in the logic LOW state. It tells about the current sinking capability of the output. The magnitude of I_{OL} determines the number of inputs the logic function can drive when its output is in the logic LOW state. For example, for the standard TTL family, the minimum guaranteed I_{OL} is 16 mA , which can drive 10 standard TTL inputs with each requiring 1.6 mA in the LOW state, as shown in Fig. 3(b).

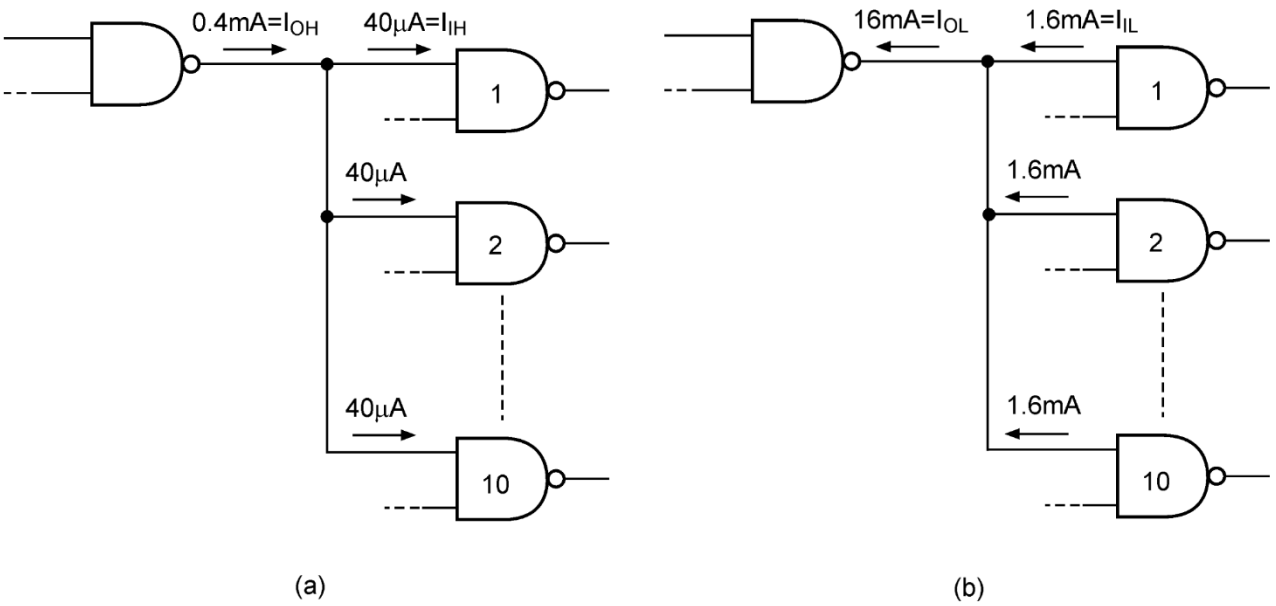


Figure 3: Input and output current specifications

Supply current, I_{CC}

The supply current when the output is HIGH and LOW state is respectively designated as I_{CCH} and I_{CCL}.

Rise time, t_r

This is the time that elapses between 10 and 90% of the final signal level when the signal is making a transition from logic LOW to logic HIGH.

Fall time, t_f

This is the time that elapses between 90 and 10% of the signal level when it is making HIGH to LOW transition.

Propagation delay, t_p

The propagation delay is the time delay between the occurrence of change in the logical level at the input and before it is reflected at the output. It is the time delay between the specified voltage points on the input and output waveforms. Propagation delays are separately defined for LOW-to-HIGH and HIGH-to-LOW transitions at the output.

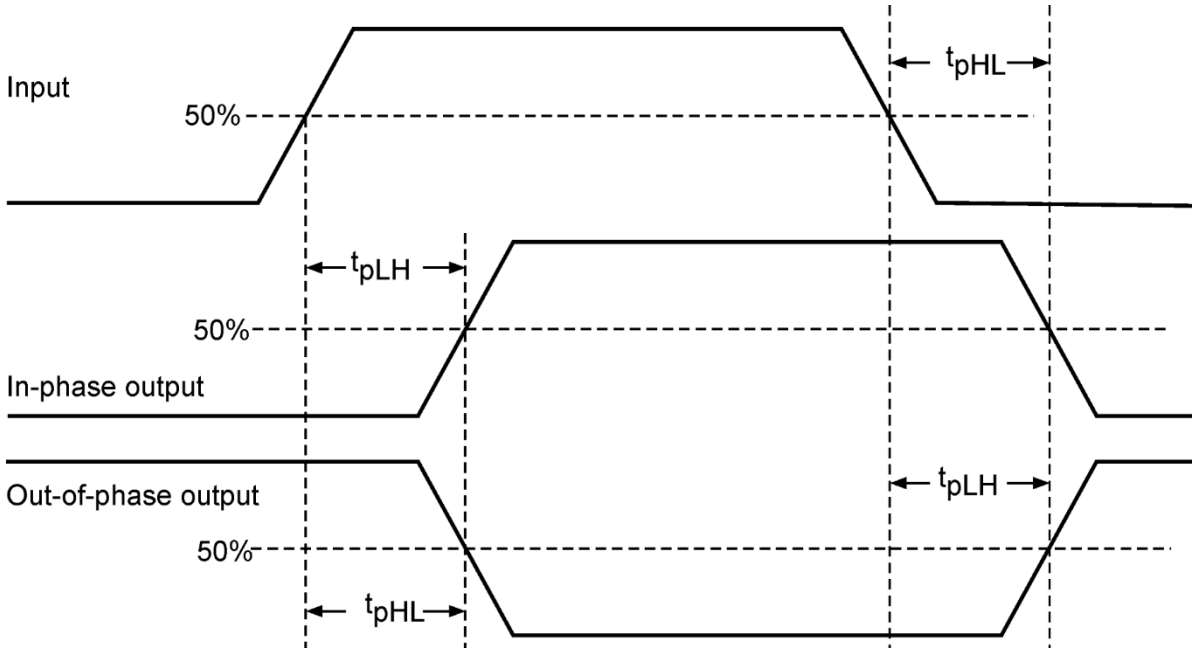


Figure 4: Propagation delay parameters

Propagation delay (LOW-to-HIGH), t_{pLH}

This is the time delay between specified voltage points on the input and output waveforms with the output changing from LOW to HIGH.

Propagation delay (HIGH-to-LOW), t_{pHL}

This is the time delay between specified voltage points on the input and output waveforms with the output changing from HIGH to LOW. Figure 5.4 shows the two types of propagation delay parameter.

Maximum clock frequency, f_{max}

This is the maximum frequency at which the clock input of a flip-flop can be driven through its required sequence while maintaining stable transitions of logic level at the output. It is also referred to as the maximum toggle rate for a flip-flop or counter device.

Power dissipation

The power dissipation parameter for a logic family is specified in terms of power consumption per gate and is the product of supply voltage V_{CC} and supply current I_{CC}. The supply current is taken as the average of the HIGH-level supply current I_{CCH} and the LOW-level supply current I_{CCL} .

Speed-power product

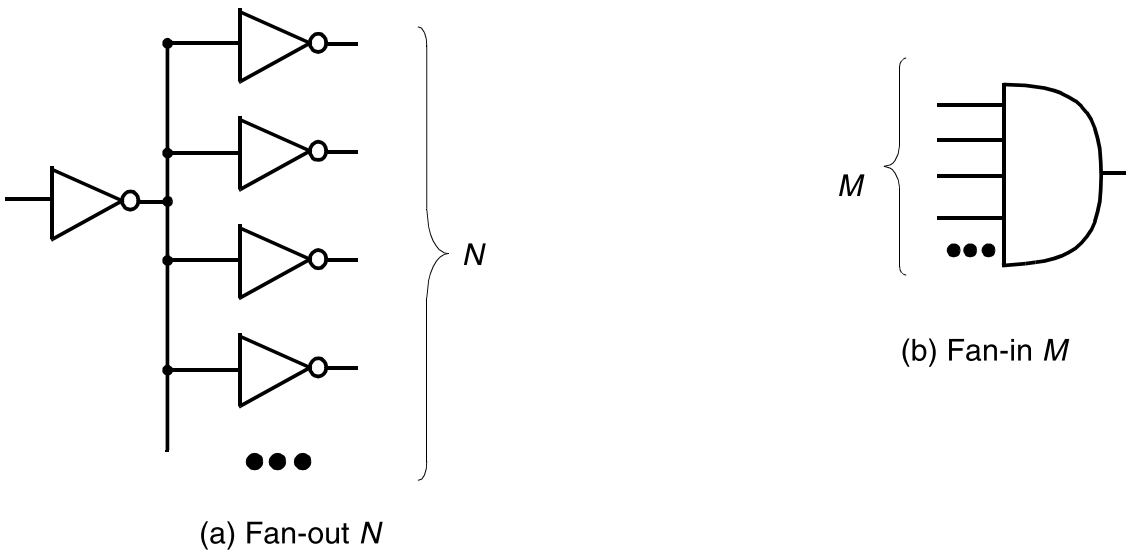
The speed of a logic circuit can be increased, that is, the propagation delay can be reduced, at the expense of power dissipation. It is always desirable to have in a logic family low values for both propagation delay and power dissipation parameters. A useful figure-of-merit used to evaluate different logic families is the speed-power product, expressed in picojoules, which is the product of the propagation delay (measured in nanoseconds) and the power dissipation per gate (measured in milliwatts).

Fan-out

The fan-out is the number of inputs of a logic function that can be driven from a single output without causing any false output. It is a characteristic of the logic family to which the device belongs. It can be computed from I_{OH}/I_{IH} in the logic HIGH state and from I_{OL}/I_{IL} in the logic LOW state. If, in a certain case, the two values I_{OH}/I_{IH} and I_{OL}/I_{IL} are different, the fan-out is taken as the smaller of the two.

Fan-in

The fan-in is the number of inputs a logic gate can accept. For example, A 3 input AND gate has a fan-in of 3.



(a) Fan-out *N*

(b) Fan-in *M*

Figure 5: Definition of fan-out and fan-in