Chapter 1

The digital abstraction

The term a *digital circuit* refers to a device that works in a binary world. In the binary world, the only values are zeros and ones. Hence, the inputs of a digital circuit are zeros and ones, and the outputs of a digital circuit are zeros and ones. Digital circuits are usually implemented by *electronic devices* and operate in the *real* world. In the real world, there are no zeros and ones; instead, what matters is the voltages of inputs and outputs. Since voltages refer to energy, they are continuous[1]. So we have a gap between the continuous real world and the two-valued binary world. One should not regard this gap as an absurd. Digital circuits are only an *abstraction* of electronic devices. In this chapter we explain this abstraction, called the *digital abstraction*.

In the digital abstraction one interprets voltage values as binary values. The advantages of the digital model cannot be overstated; this model enables one to focus on the digital behavior of a circuit, to ignore analog and transient phenomena, and to easily build larger more complex circuits out of small circuits. The digital model together with a simple set of rules, called *design rules*, enable logic designers to design complex digital circuits consisting of millions of gates.

1.1 Transistors

Electronic circuits that are used to build computers are mostly build of *transistors*. Small circuits, called *gates* are built from transistors. The most common technology used in VLSI chips today is called CMOS, and in this technology there are only two types of transistors: N-type and P-type. Each transistor has three connections to the outer world, called the *gate*, *source*, and *drain*. Figure 1.1 depicts diagrams describing these transistors.

Figure 1.1: Schematic symbols of an N-transistor and P-transistor

Although inaccurate, we will refer, for the sake of simplicity, to the gate and source as inputs and to the drain as an output. An overly simple explanation of an N-type transistor in CMOS technology is as follows: If the voltage of the gate is high (i.e. above some threshold

[1] unless Quantum Physics is used.

v_1), then there is little resistance between the source and the drain. Such a small resistance causes the voltage of the drain to equal the voltage of the source. If the voltage of the gate is low (i.e. below some threshold $v_0 < v_1$), then there is a very high resistance between the source and the drain. Such a high resistance means that the voltage of the drain is unchanged by the transistor (it could be changed by another transistor if the drains of the two transistors are connected). A P-type transistor is behaves in a dual manner: the resistance between drain and the source is low if the gate voltage is below v_0. If the voltage of the gate is above v_1, then the source-to-drain resistance is very high.

Note that this description of transistor behavior implies immediately that transistors are highly non-linear. (Recall that a linear function $f(x)$ satisfies $f(a \cdot x) = a \cdot f(x)$.) In transistors, changes of 10% in input values above the threshold v_1 have a small effect on the output while changes of 10% in input values between v_0 and v_1 have a large effect on the output. In particular, this means that transistors do not follow Ohm's Law (i.e. $V = I \cdot R$).

Example 1 (A CMOS inverter) *Figure 1.2 depicts a CMOS inverter. If the input voltage is above v_1, then the source-to-drain resistance in the P-transistor is very high and the source-to-drain resistance in the N-transistor is very low. Since the source of the N-transistor is connected to low voltage (i.e. ground), the output of the inverter is low.*

If the input voltage is below v_0, then the source-to-drain resistance in the N-transistor is very high and the source-to-drain resistance in the P-transistor is very low. Since the source of the P-transistor is connected to high voltage, the output of the inverter is high.

We conclude that the voltage of the output is low when the input is high, and vice-versa, and the device is indeed an inverter.

Figure 1.2: A CMOS inverter

The qualitative description in Example 1 hopefully conveys some intuition about how gates are built from transistors. A quantitative analysis of such an inverter requires precise modeling of the functionality of the transistors in order to derive the input-output voltage relation. One usually performs such an analysis by computer programs (e.g. SPICE). Quantitative analysis is relatively complex and inadequate for designing large systems like computers. (This would be like having to deal with the chemistry of ink when using a pen.)

1.2 From analog signals to digital signals

An *analog signal* is a real function $f : \mathbb{R} \rightarrow \mathbb{R}$ that describes the voltage of a given point in a circuit as a function of the time. We ignore the resistance and capacities of wires. Moreover, we assume that signals propagate through wires immediately[2]. Under these assumptions, it follows that the voltage along a wire is identical at all times. Since a signal describes the voltage (i.e. derivative of energy as a function of electric charge), we also assume that a signal is a continuous function.

[2] This is a reasonable assumption if wires are short.

A *digital signal* is a function $g : \mathbb{R} \rightarrow \{0, 1, \text{non-logical}\}$. The value of a digital signal describes the *logical value* carried along a wire as a function of time. To be precise there are two logical values: zero and one. The non-logical value simply means that that the signal is neither zero or one.

How does one interpret an analog signal as a digital signal? The simplest interpretation is to set a threshold V'. Given an analog signal $f(t)$, the digital signal $dig(f(t))$ can be defined as follows.

$$dig(f(t)) \triangleq \begin{cases} 0 & \text{if } f(t) < V' \\ 1 & \text{if } f(t) > V' \end{cases} \tag{1.1}$$

According to this definition, a digital interpretation of an analog signal is always 0 or 1, and the digital interpretation is never non-logical.

There are several problems with the definition in Equation 1.1. One problem with this definition is that all the components should comply with *exactly* the same threshold V'. In reality, devices are not completely identical; the actual thresholds of different devices vary according to a tolerance specified by the manufacturer. This means that instead of a fixed threshold, we should consider a range of thresholds.

Another problem with the definition in Equation 1.1 is caused by perturbations of $f(t)$ around the threshold t. Such perturbations can be caused by *noise* or oscillations of $f(t)$ before it stabilizes. We will elaborate more on noise later, and now explain why oscillations can occur. Consider a spring connected to the ceiling with a weight w hanging from it. We expect the spring to reach a length ℓ that is proportional the the weight w. Assume that all we wish to know is whether the length ℓ is greater than a threshold ℓ_t. Sounds simple! But what if ℓ is rather close to ℓ_t? In practice, the length only tends to the length ℓ as time progresses; the actual length of the spring oscillates around ℓ with a diminishing amplitude. Hence, the length of the spring fluctuates below and above ℓ_t many times before we can decide. This effect may force us to wait for a long time before we can decide if $\ell < \ell_t$. If we return to the definition of $dig(f(t))$, it may well happen that $f(t)$ oscillates around the threshold V'. This renders the digital interpretation used in Eq. 1.1 useless.

Returning to the example of weighing weights, assume that we have two types of objects: light and heavy. The weight of a light (resp., heavy) object is at most (resp., at least) w_0 (resp., w_1). The bigger the gap $w_1 - w_0$, the easier it becomes to determine if an object is light or heavy (especially in the presence of noise or oscillations).

Now we have two reasons to introduce two threshold values instead of one, namely, different threshold values for different devices and the desire to have a gap between values interpreted as logical zero and logical one. We denote these thresholds by V_{low} and V_{high}, and require that $V_{low} < V_{high}$. An interpretation of an analog signal is depicted in Figure 1.3. Consider an analog signal $f(t)$. The digital signal $dig(f(t))$ is defined as follows.

$$dig(f(t)) \triangleq \begin{cases} 0 & \text{if } f(t) < V_{low} \\ 1 & \text{if } f(t) > V_{high} \\ \text{non-logical} & \text{otherwise.} \end{cases} \tag{1.2}$$

We often refer to the logical value of an analog signal f. This is simply a shorthand way of referring to the value of the digital signal $dig(f)$.

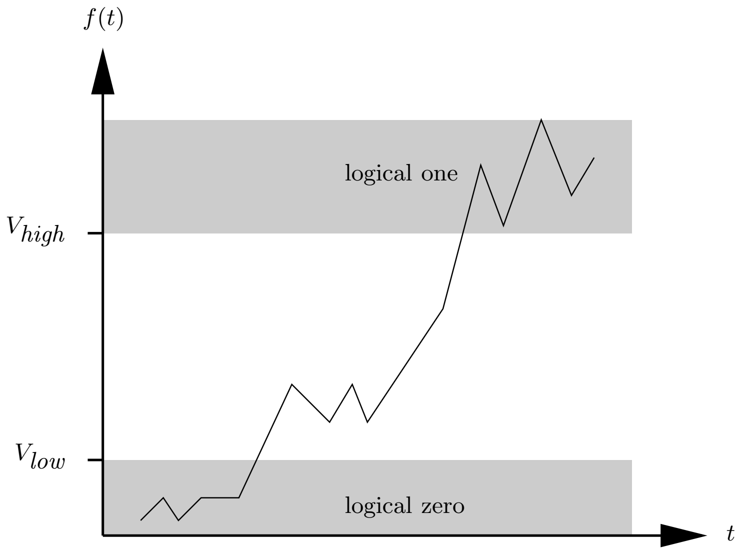

Figure 1.3: A digital interpretation of an analog signal in the zero-noise model.

It is important to note that fluctuations of $f(t)$ are still possible around the threshold values. However, if the two thresholds are sufficiently far away from each other, fluctuations of f do not cause fluctuations of $dig(f(t))$ between 0 and 1. Instead, we will have at worst fluctuations of $dig(f(t))$ between a non-logical value and a logical value (i.e. 0 or 1). A fluctuation between a logical value and a non-logical value is much more favorable than a fluctuation between 0 and 1. The reason is that a non-logical value is an indication that the circuit is still in a transient state and a “decision” has not been reached yet.

Assume that we design an inverter so that its output tends to a voltage that is bounded away from the thresholds V_{low} and V_{high}. Let us return to the example of the spring with weight w hanging from it. Additional fluctuations in the length of the spring might be caused by wind. This means that we need to consider additional effects so that our model will be useful. In the case of the digital abstraction, we need to take *noise* into account. Before we consider the effect of noise, we formulate the static functionality of a gate, namely, the values of its output as a function of its stable inputs.

Question 1 *Try to define an inverter in terms of the voltage of the output as a function of the voltage of the input.*

1.3 Transfer functions of gates

The voltage at an output of a gate depends on the voltages of the inputs of the gate. This dependence is called the *transfer function* (or the *voltage-transfer characteristic* - VTC). Consider, for example an inverter with an input x and an output y. To make things complicated, the value of the signal $y(t)$ at time t is not only a function of the signal x at time t since $y(t)$ depends on the history. Namely, $y(t_0)$ is a function of $x(t)$ over the interval $(-\infty, t_0]$.

Transfer functions are solved by modeling gates with partial differential equations, a rather complicated task. A good approximation of transfer functions is obtain by solving differential equations, still a complicated task that can be computed quickly only for a few transistors. So how are chips that contain millions of chips designed?

The way this very intricate problem is handled is by restricting designs. In particular, only a small set of building blocks is used. The building blocks are analyzed intensively, their properties are summarized, and designers rely on these properties for their designs.

One of the most important steps in characterizing the behavior of a gate is computing its *static transfer function*. Returning to the example of the inverter, a "proper" inverter has a unique output value point for each input value. Namely, if the input $x(t)$ is stable for a sufficiently long period of time and equals x_0, then the output $y(t)$ stabilizes on a value y_0 that is a function of x_0.[3] We formalize the definition of a static transfer function of a gate G with one input x and one output y in the following definition.

Definition 1 *Consider a device G with one input x and one output y. The device G is a* gate *if its functionality is specified by a a function $f : \mathbb{R} \rightarrow \mathbb{R}$ as follows: there exists a $\Delta > 0$, such that, for every x_0 and every t_0, if $x(t) = x_0$ for every $t \in [t_0 - \Delta, t_0]$, then $y(t_0) = f(x_0)$.*
Such a function $f(x)$ is called the static transfer function *of G.*

At this point we should point the following remarks:

1. Since circuits operate over a bounded range of voltages, static transfer functions are usually only defined over bounded domains and ranges (say $[0, 5]$ volts).

2. To make the definition useful, one should allow perturbations of $x(t)$ during the interval $[t_0 - \Delta, t_0]$. Static transfer functions model physical devices, and hence, are continuous. This implies the following definition: For every $\epsilon > 0$, there exist a $\delta > 0$ and a $\Delta > 0$, such that

$$\forall t \in [t_1, t_2] \ : \ |x(t) - x_0| \leq \delta \quad \Rightarrow \quad \forall t \in [t_1 + \Delta, t_2] \ : \ |y(t) - f(x_0)| \leq \epsilon.$$

3. Note that in the above definition Δ does not depend on x_0 (although it may depend on ϵ). Typically, we are interested on the values of Δ only for logical values of $x(t)$ (i.e. $x(t) \leq V_{low}$ and $x(t) \geq V_{high}$). Once the value of ϵ is fixed, this constant Δ is called the *propagation delay* of the gate G and is one of the most important characteristic of a gate.

[3] If this were not the case, then we need to distinguish between two cases: (a) Stability is not reached: this case occurs, for example, with devices called oscillators. Note that such devices must consume energy even when the input is stable. We point out that in CMOS technology it is easy to design circuits that do not consume energy if the input is logical, so such oscillations are avoided. (b) Stability is reached: in this case, if there is more than one stable output value, it means that the device has more than one equilibrium point. Such a device can be used to store information about the "history". It is important to note that devices with multiple equilibriums are very useful as storage devices (i.e. they can "remember" a small amount of information). Nevertheless, devices with multiple equilibriums are not "good" candidates for gates, and it is easy to avoid such devices in CMOS technology..

Question 2 *Extend Definition 1 to gates with n inputs and m outputs.*

Finally, we can now define an inverter in the zero-noise model. Observe that according to this definition a device is an inverter if its static transfer function satisfies a certain property.

Definition 2 (inverter in zero-noise model) *A gate G with a single input x and a single output y is an inverter if its static transfer function $f(z)$ satisfies the following the following two conditions:*

1. *If $z < V_{\text{low}}$, then $f(z) > V_{\text{high}}$.*
2. *If $z > V_{\text{high}}$, then $f(z) < V_{\text{low}}$.*

The implication of this definition is that if the logical value of the input x is zero (resp., one) during an interval $[t_1, t_2]$ of length at least Δ, then the logical value of the output y is one (resp., zero) during the interval $[t_1 + \Delta, t_2]$.

How should we define other gates such a NAND-gates, XOR-gates, etc.? As in the definition of an inverter, the definition of a NAND-gate is simply a property of its static transfer function.

Question 3 *Define a* NAND*-gate.*

We are now ready to strengthen the digital abstraction so that it will be useful also in the presence of bounded noise.

1.4 The bounded-noise model

Consider a wire from point A to point B. Let $A(t)$ (resp., $B(t)$) denote the analog signal measured at point A (resp., B). We would like to assume that wires have zero resistance, zero capacitance, and that signals propagate through a wire with zero delay. This assumption means that the signals $A(t)$ and $B(t)$ should be equal at all times. Unfortunately, this is not the case; the main reason for this discrepancy is *noise*.

There are many sources of noise. The main source is heat that causes electrons to move randomly. These random movements do not cancel out perfectly, and random currents are created. These random currents create perturbations in the voltage of a wire. The difference between the signals $B(t)$ and $A(t)$ is a *noise signal*.

Consider, for example, the setting of *additive noise*: A is an output of an inverter and B is an input of another inverter. We consider the signal $A(t)$ to be a reference signal. The signal $B(t)$ is the sum $A(t) + n_B(t)$, where $n_B(t)$ is the noise signal.

The *bounded-noise model* assumes that the noise signal along every wire has a bounded absolute value. We will use a slightly simplified model in which there is a constant $\epsilon > 0$ such that the absolute value of all noise signals is bounded by ϵ. We refer to this model as the *uniform bounded noise model*. The justification for assuming that noise is bounded is probabilistic. Noise is a random variable whose distribution has a rapidly diminishing tail. This means that if the bound is sufficiently large, then the probability of the noise exceeding this bound during the lifetime of a circuit is negligibly small.

1.5 The digital abstraction in presence of noise

Consider two inverters, where the output of one gate feeds the input of the second gate. Figure 1.4 depicts such a circuit that consists of two inverters.

Assume that the input x has a value that satisfies: (a) $x > V_{high}$, so the logical value of x is one, and (b) $y = V_{low} - \epsilon'$, for a very small $\epsilon' > 0$. This might not be possible with every inverter, but Definition 2 does not rule out such an inverter. (Consider a transfer function with $f(V_{high}) = V_{low}$, and x slightly higher than V_{high}.) Since the logical value of y is zero, it follows that the second inverter, if not faulty, should output a value z that is greater than V_{high}. In other words, we expect the logical value of z to be 1. At this point we consider the effect of adding noise.

Let us denote the noise added to the wire y by n_y. This means that the input of the second inverter equals $y(t) + n_y(t)$. Now, if $n_y(t) > \epsilon'$, then the second inverter is fed a non-logical value! This means that we can no longer deduce that the logical value of z is one. We conclude that we must use a more resilient model; in particular, the functionality of circuits should not be affected by noise. Of course, we can only hope to be able to cope with bounded noise, namely noise whose absolute value does not exceed a certain value ϵ.

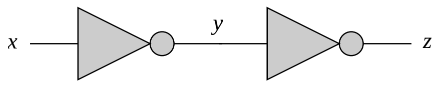

Figure 1.4: Two inverters connected in series.

1.5.1 Redefining the digital interpretation of analog signals

The way we deal with noise is that we interpret input signals and output signals differently. An input signal is a signal measured at an input of a gate. Similarly, an output signal is a signal measured at an output of a gate. Instead of two thresholds, V_{low} and V_{high}, we define the following four thresholds:

- $V_{low,in}$ - an upper bound on a voltage of an input signal interpreted as a logical zero.
- $V_{low,out}$ - an upper bound on a voltage of an output signal interpreted as a logical zero.
- $V_{high,in}$ - a lower bound on a voltage of an input signal interpreted as a logical one.
- $V_{high,out}$ - a lower bound on a voltage of an output signal interpreted as a logical one.

These four thresholds satisfy the following equation:

$$V_{low,out} < V_{low,in} < V_{high,in} < V_{high,out}. \tag{1.3}$$

Figure 1.5 depicts these four thresholds. Note that the interpretation of input signals is less strict than the interpretation of output signals. The actual values of these four thresholds depend on the transfer functions of the devices we wish to use.

The differences $V_{low,in} - V_{low,out}$ and $V_{high,out} - V_{high,in}$ are called *noise margins*. Our goal is to show that noise whose absolute value is less than the noise margin will not change the logical value of an output signal. Indeed, if the absolute value of the noise $n(t)$ is bounded by the noise margins, then an output signal $f_{out}(t)$ that is below $V_{low,in}$ will result with an input signal $f_{in}(t) = f_{out}(t) + n(t)$ that does not exceed $V_{low,out}$.

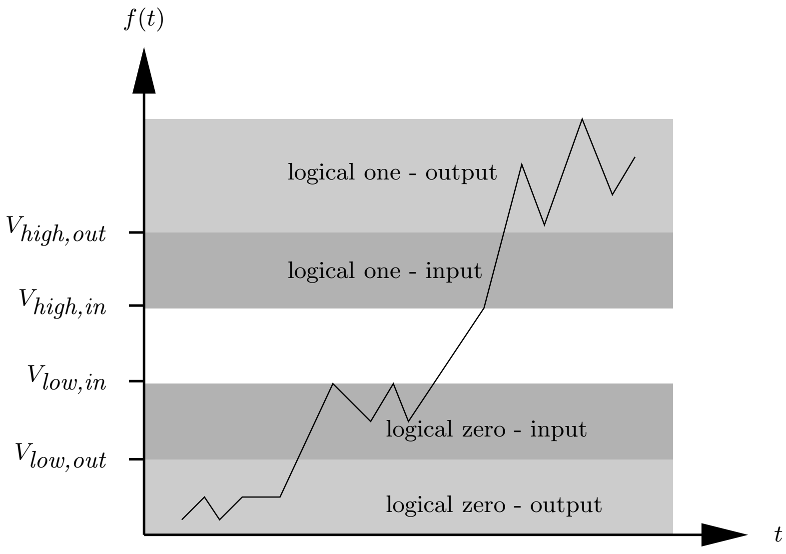

Figure 1.5: A digital interpretation of an input and output signals.

Consider an input signal $f_{in}(t)$. The digital signal $dig(f_{in}(t))$ is defined as follows.

$$dig(f_{in}(t)) \triangleq \begin{cases} 0 & \text{if } f_{in}(t) < V_{low,in} \\ 1 & \text{if } f_{in}(t) > V_{high,in} \\ \text{non-logical} & \text{otherwise.} \end{cases} \tag{1.4}$$

Consider an output signal $f_{out}(t)$. The digital signal $dig(f_{out}(t))$ is defined analogously.

$$dig(f_{out}(t)) \triangleq \begin{cases} 0 & \text{if } f_{out}(t) < V_{low,out} \\ 1 & \text{if } f_{out}(t) > V_{high,out} \\ \text{non-logical} & \text{otherwise.} \end{cases} \tag{1.5}$$

Observe that sufficiently large noise margins imply that noise will not change the logical values of signals.

We can now fix the definition of an inverter so that bounded noise added to outputs, does not affect logical interpretation of signals.

Definition 3 (inverter in the bounded-noise model) *A gate G with a single input x and a single output y is an inverter if its static transfer function $f(z)$ satisfies the following the following two conditions:*

1. *If $z < V_{\text{low,in}}$, then $f(z) > V_{\text{high,out}}$.*
2. *If $z > V_{\text{high,in}}$, then $f(z) < V_{\text{low,out}}$.*

Question 4 *Define a* NAND*-gate in the bounded-noise model.*

Question 5 *Consider the function $f(x) = 1-x$ over the interval $[0,1]$. Suppose that $f(x)$ is a the transfer function of a device C. Can you define threshold values $V_{\text{low,out}} < V_{\text{low,in}} < V_{\text{high,in}} < V_{\text{high,out}}$ so that C is an inverter according to Definition 3?*

Question 6 *Consider a function $f : [0,1] \rightarrow [0,1]$. Suppose that $f(x)$ is monotone decreasing and that the derivative $f'(x)$ of $f(x)$ satisfies the following condition: $f'(x)$ is continuous, $f'(x)$ is strictly decreasing in the interval $[0,\alpha]$ and strictly increasing in the interval $(\alpha,1]$. Moreover, $f'(\alpha) < -1$. Can you define threshold values $V_{\text{low,out}} < V_{\text{low,in}} < V_{\text{high,in}} < V_{\text{high,out}}$ so that C is an inverter according to Definition 3?*

Question 7 *Try to characterize transfer functions $g(x)$ that correspond to inverters. Namely, if C_g is a device, the transfer function of which equals $g(x)$, then one can define threshold values that satisfy Definition 3.*

1.6 Stable signals

In this section we define terminology that will be used later. To simplify notation we define these terms in the zero-noise model. We leave it to the curious reader to extend the definitions and notation below to the bounded-noise model.

An analog signal $f(t)$ is said to be *logical at time t* if $dig(f(t)) \in \{0,1\}$. An analog signal $f(t)$ is said to be *stable* during the interval $[t_1,t_2]$ if $f(t)$ is logical for every $t \in [t_1,t_2]$. Continuity of $f(t)$ and the fact that $V_{low} < V_{high}$ imply the following claim.

Claim 1 *If an analog signal $f(t)$ is stable during the interval $[t_1,t_2]$ then one of the following holds:*

1. $\text{dig}(f(t)) = 0$*, for every $t \in [t_1,t_2]$, or*
2. $\text{dig}(f(t)) = 1$*, for every $t \in [t_1,t_2]$.*

From this point we will deal with digital signals and use the same terminology. Namely, a digital signal $x(t)$ is *logical* at time t if $x(t) \in \{0,1\}$. A digital signal is *stable* during an interval $[t_1,t_2]$ if $x(t)$ is logical for every $t \in [t_1,t_2]$.

1.7 Summary

In this chapter we presented the digital abstraction of analog devices. For this purpose we defined analog signals and their digital counterpart, called digital signals. In the digital abstraction, analog signals are interpreted either as zero, one, or non-logical.

We discussed noise and showed that to make the model useful, one should set stricter requirements from output signals than from input signals. Our discussion is based on the bounded-noise model in which there is an upper bound on the absolute value of noise.

We defined gates using transfer functions and static transfer functions. This functions describe the analog behavior of devices. We also defined the propagation delay of a device as the amount of time that input signals must be stable to guarantee stability of the output of a gate.