**Digital Waveforms**

Digital waveforms are graphical representations of digital signals, which are discrete-time signals that take on values from a finite set of distinct levels.

Digital waveforms consist of voltage levels that are changing back and forth between the **HIGH** and **LOW** levels or states.

**HIGH ­ 1 and LOW ­ 0**

It is used in digital communication, clock signals, and various digital circuits.

**Logic Levels**

The voltages used to represent a 1 and a 0 are called logic levels. Ideally, one voltage level represents a **HIGH** and another voltage level represents a **LOW**. In a practical digital circuit, however, a **HIGH** can be any voltage between a specified minimum value and a specified maximum value. Likewise, a **LOW** can be any voltage between a specified minimum and a specified maximum. There can be no overlap between the accepted range of **HIGH** levels and the accepted range of **LOW** levels.

For example, the **HIGH** input values for a certain type of digital circuit technology called CMOS may range from 2 V to 3.3 V and the **LOW** input values may range from 0 V to 0.8 V. If a voltage of 2.5 V is applied, the circuit will accept it as a **HIGH** or binary 1. If a voltage of 0.5 V is applied, the circuit will accept it as a **LOW** or binary 0. For this type of circuit, voltages between 0.8 V and 2 V are unacceptable.

**Pulse**

A pulse is a short-duration signal that transitions from one level to another and back. It is characterized by a rapid change in voltage or current and a well-defined width or duration. A pulse typically has a relatively constant amplitude during its active state and then returns to its original state.

Pulse waveforms are commonly used in digital systems for various purposes, including data transmission, clock signals, and modulation.

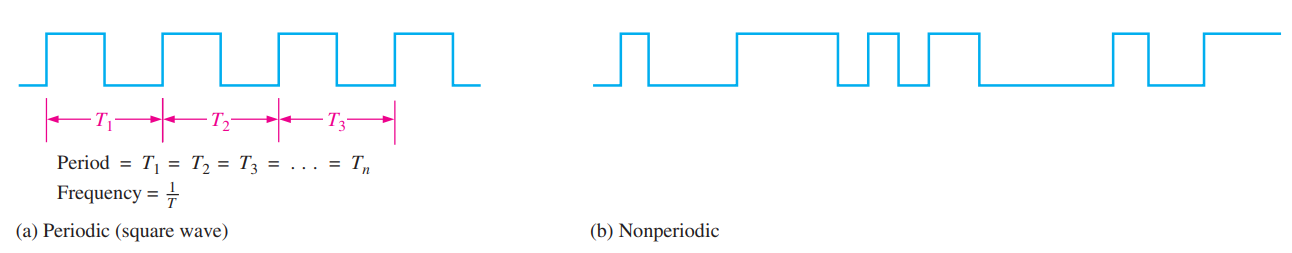
A **pulse train** is a series of pulses that are separated by a fixed interval of time. The pulses can be identical or unique, and the duration and amplitude of each pulse are often constant.

A pulse wave or pulse train or rectangular wave is a non-sinusoidal waveform that is the periodic version of the rectangular function.

**Waveform Characteristics**

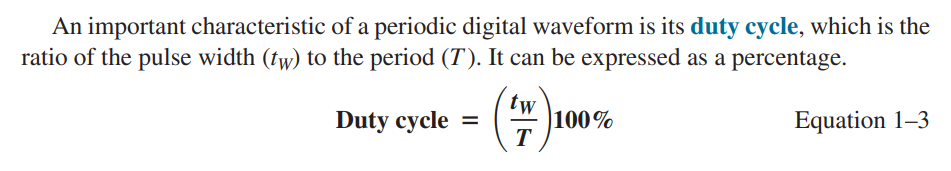
Most waveforms encountered in digital systems are composed of series of pulses, sometimes called pulse trains, and can be classified as either periodic or nonperiodic. A periodic pulse waveform is one that repeats itself at a fixed interval, called a period (T). The frequency (f) is the rate at which it repeats itself and is measured in hertz (Hz).

A non-periodic pulse waveform, of course, does not repeat itself at fixed intervals and may be composed of pulses of randomly differing pulse widths and/or randomly differing time intervals between the pulses.

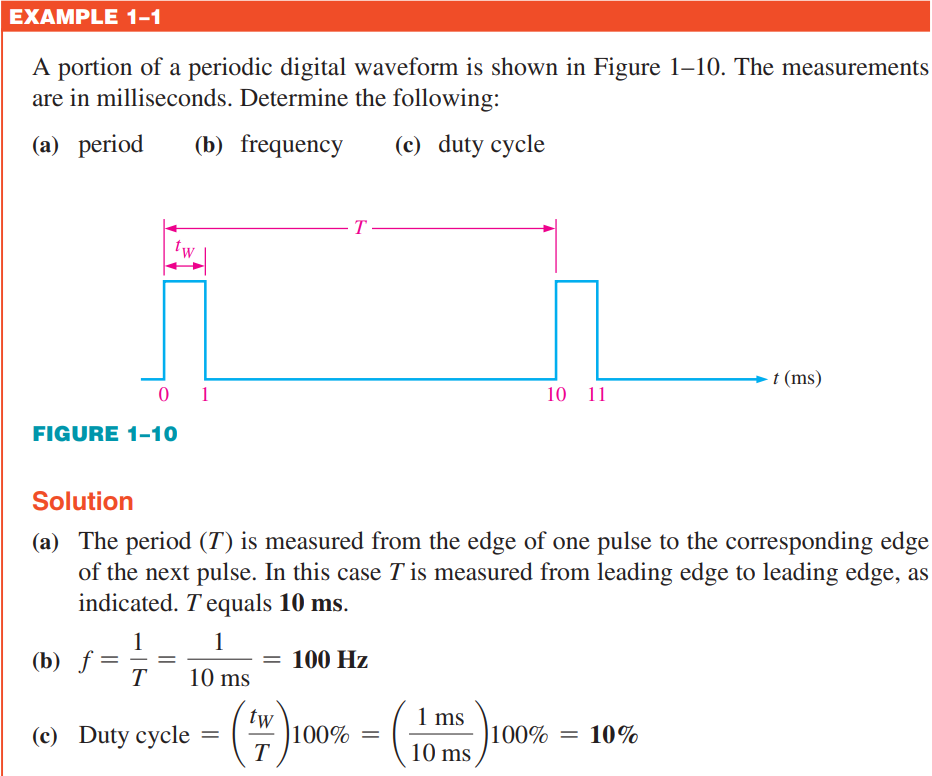


A math equation with black text

Description automatically generated



**Problems:**

b) f = 1/T = 1/10 ms

-- ms = milliseconds = 10-3 seconds

1/(10)( 10-3)

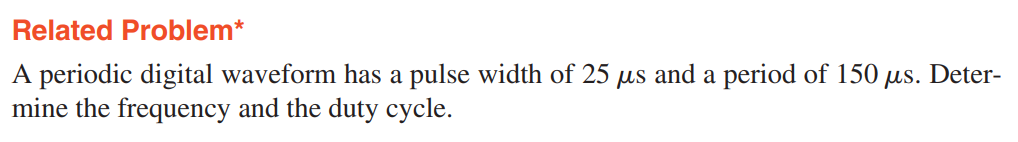
103/10

1000/10

100Hz

-us = microseconds = 10-6 seconds

-ns = nanosecond = 10-9 seconds



F = 1/T = 1/150 us = 1/150 10-6 = 106 / 150 = 1000 000/150 = 1000 00/15 = 6666.66Hz

Duty Cycle = (tw/T)(100%) = (25 us/ 150 us)(100%) = (1/6)(100%) = 16.7%

f = 6.67 kHz; Duty cycle = 16.7%

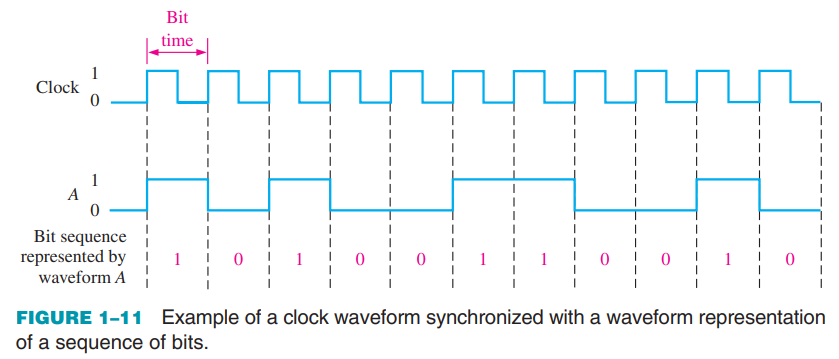
**A Digital Waveform Carries Binary Information**

Binary information that is handled by digital systems appears as waveforms that represent sequences of bits. When the waveform is **HIGH**, a binary 1 is present; when the waveform is **LOW**, a binary 0 is present. Each bit in a sequence occupies a defined time interval called a **bit time**.

**The Clock**

In digital systems, all waveforms are synchronized with a basic timing waveform called the clock. The clock is a periodic waveform in which each interval between pulses (the period) equals the time for one bit.

Notice that, in this case, each change in level of waveform A occurs at the leading edge of the clock waveform. In other cases, level changes occur at the trailing edge of the clock. During each bit time of the clock, waveform A is either **HIGH** or **LOW**. These **HIGH**s and **LOW**s represent a sequence of bits as indicated. A group of several bits can contain binary information, such as a number or a letter. The clock waveform itself does not carry information.



**Timing Diagrams**

A timing diagram is a graph of digital waveforms showing the actual time relationship of two or more waveforms and how each waveform changes in relation to the others. By looking at a timing diagram, you can determine the states (HIGH or LOW) of all the waveforms at any specified point in time and the exact time that a waveform changes state relative to the other waveforms.

Figure 1–12 is an example of a timing diagram made up of four waveforms. From this timing diagram you can see, for example, that the three waveforms A, B, and C are HIGH only during bit time 7 (shaded area) and they all change back LOW at the end of bit time 7.

A diagram of a timing diagram

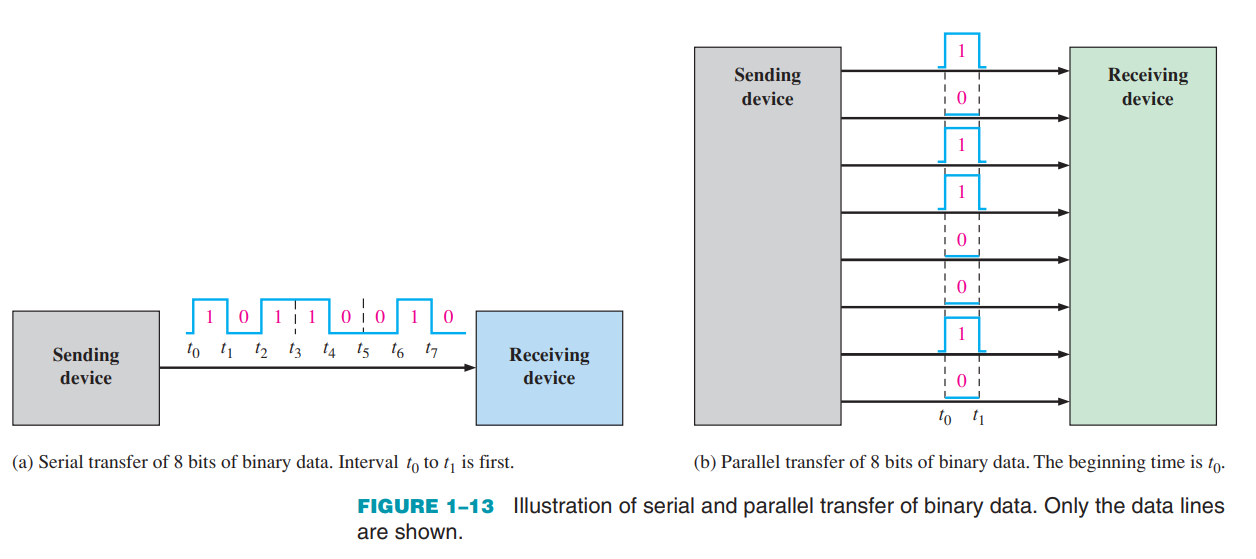
Description automatically generated

**Data Transfer**

Data refers to groups of bits that convey some type of information. Binary data, which are represented by digital waveforms, must be transferred from one device to another within a digital system or from one system to another in order to accomplish a given purpose. For example, numbers stored in binary form in the memory of a computer must be transferred to the computer’s central processing unit in order to be added. The sum of the addition must then be transferred to a monitor for display and/or transferred back to the memory. As illustrated in Figure 1–13, binary data are transferred in two ways: serial and parallel.

**Parallel Transfer**

When bits are transferred in serial form from one point to another, they are sent one bit at a time along a single line, as illustrated in Figure 1–13(a). During the time interval from t0 to t1, the first bit is transferred. During the time interval from t1 to t2, the second bit is transferred, and so on. To transfer eight bits in series, it takes eight time intervals



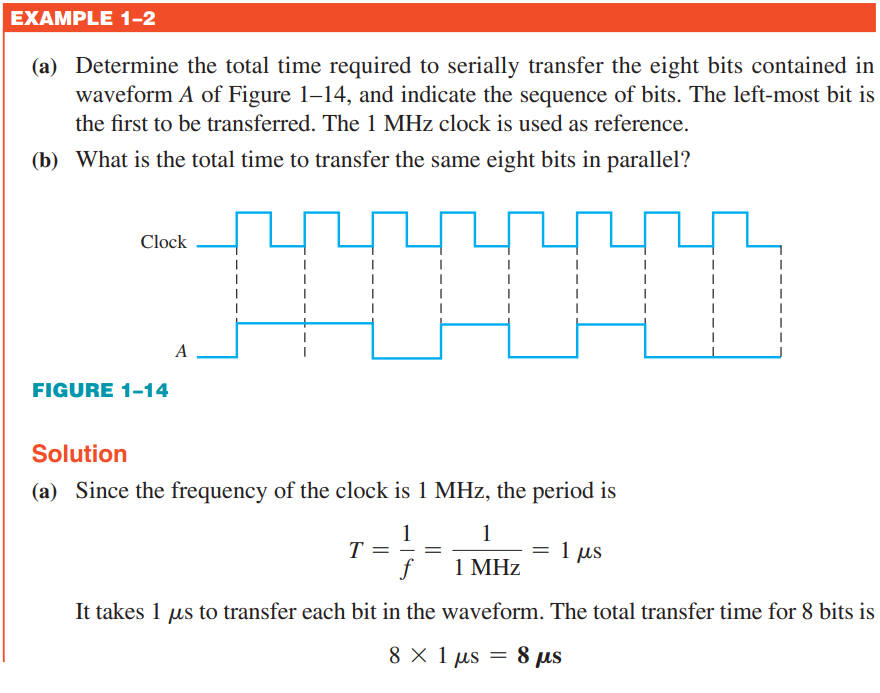
**Parallel Transfer**

When bits are transferred in parallel form, all the bits in a group are sent out on separate lines at the same time. There is one line for each bit, as shown in Figure 1–13(b) for the example of eight bits being transferred. To transfer eight bits in parallel, it takes one time interval compared to eight time intervals for the serial transfer. To summarize, an advantage of serial transfer of binary data is that a minimum of only one line is required. In parallel transfer, a number of lines equal to the number of bits to be transferred at one time is required. A disadvantage of serial transfer is that it takes longer to transfer a given number of bits than with parallel transfer at the same clock frequency. For example, if one bit can be transferred in 1us, then it takes 8 us to serially transfer eight bits but only 1us to parallel transfer eight bits. A disadvantage of parallel transfer is that it takes more lines than serial transfer.

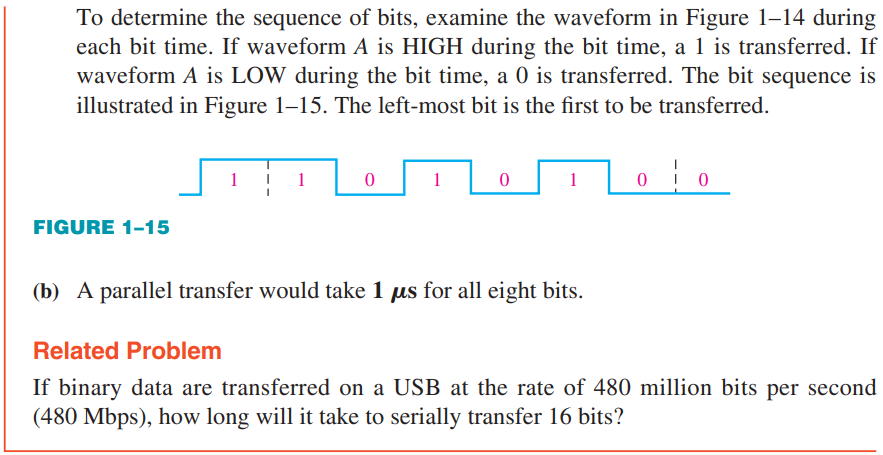
**Discuss the significance of timing diagrams in digital systems.**

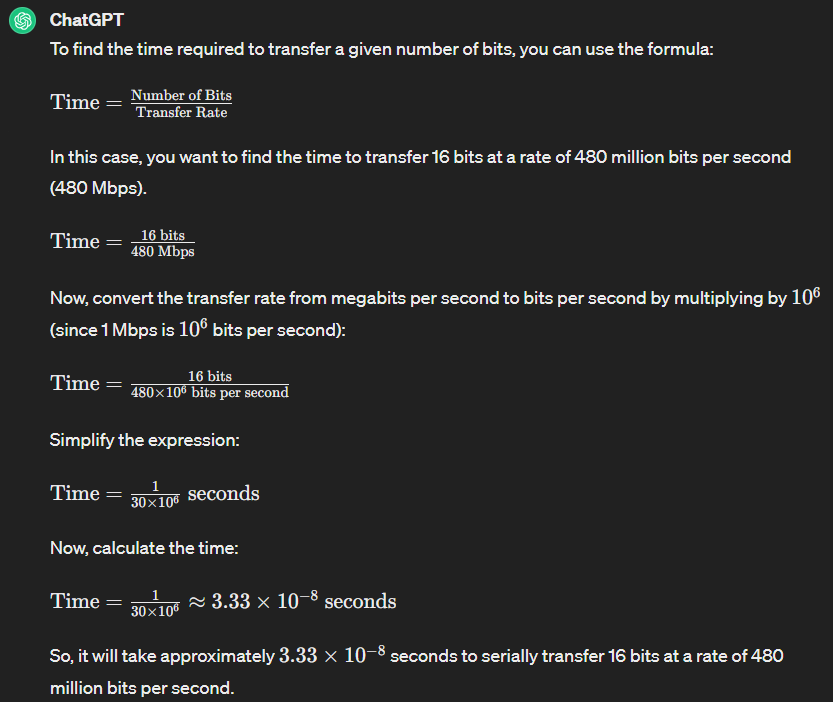
Timing diagrams play a crucial role in digital systems for visualizing and understanding the timing relationships between different signals and events. Here are some key aspects of the significance of timing diagrams in digital systems:

1. **Visualization of Signal Relationships:**
   * Timing diagrams provide a graphical representation of how signals change over time. This visualization is essential for understanding the relationships between different signals and their transitions.
2. **Timing Analysis:**
   * Engineers use timing diagrams to perform timing analysis, which involves evaluating the timing characteristics of digital circuits. This includes understanding signal propagation delays, setup and hold times, and overall system timing constraints.
3. **Clock Synchronization:**
   * Timing diagrams help in analyzing the synchronization of clock signals within a digital system. They illustrate how different components synchronize with the clock, aiding in the detection of potential timing issues.
4. **Verification of Timing Requirements:**
   * By examining timing diagrams, engineers can verify whether the system meets specified timing requirements and constraints. This is crucial in ensuring the proper operation and performance of digital systems.
5. **Debugging and Troubleshooting:**
   * When debugging digital systems, timing diagrams are invaluable for identifying and resolving timing-related issues. They provide insights into unexpected behavior, glitches, or signal integrity problems.

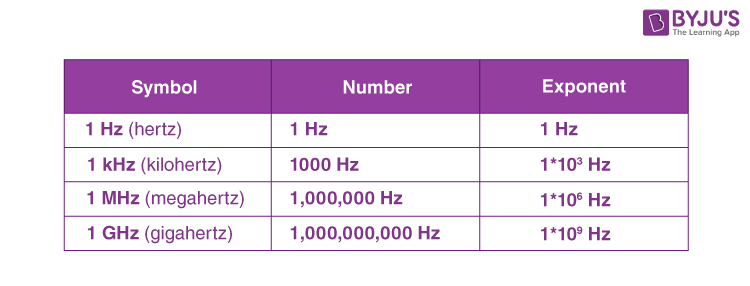


MHz = 106





**Time** = 16/(48 x 107 ) = (1/3) x 10-7 = 0.033 x 10-7 = 3.3 x 10-9 = 3.3 ns



A graph with numbers and lines

Description automatically generated

Time of one cycle = Tc = 4 frequency = 1/T = 0.25Hz

Pulse Width = tw = 2 Duty Cycle = (tw/T)(100%) = (2/4)(100%) = 50%

