The circuits we use to assemble electronic devices in general and computers in particular can be roughly divided into two groups: synchronous circuits and asynchronous circuits.

Asynchronous circuits are circuits with components that operate independently without a clock component scheduling their operations. Usually, these components work slower than synchronous circuits: each component starts operating only when the previous component has finished its task, even if theoretically, they can work in parallel. For this reason, the subsequent component depends on the leading component (No pipelining).

Synchronous circuits, on the other hand, are circuits that synchronize operations within the circuit which allows for parallel work (pipelining).

This kind of clock will emit a clock signal that reaches each component and begin its operation when the clock rises or falls. The clock signal will be graphically plotted as a Duty Cycle which when rising (Rising edge) will give a logical value of '1' and when falling (Falling edge) will give a logical '0' value.

The clock cycle is measured in units of time (usually ns) and the frequency of the clock is measured in units of hertz (Hz) usually kHz, MHz and GHz.

The clock component is basically a component capable of emitting an AC electric voltage that is graphically expressed as a Sin wave converted to a square wave. This component is usually composed of a crystal (Generally quartz) which is under electrical voltage. Crystals have a unique feature of being able to generate electrical voltage when they are under electrical voltage themselves, this is the basis for all crystal-based computer clocks. In addition to these clocks there are other clocks based on RC or LC circuits (circuits consisting of opposite and capacitor or coil and capacitor).

Since the clocks are components that ultimately rely on the properties of an organic substance, the attempt to use them in order to produce an accurate product is **almost** absolutely successful. Nonetheless the properties of the material cause slight deviations that produce synchronization problems between the components of the circuit which manifest in difficulties reading and writing the components that are synchronized by the clock. Small problems create big problems.

These problems are divided into two: internal problems within the clock circuit and problems that arise in synchronization between clocks. The internal problems of the clock circuit are due to two main elements: clock skew and clock jitter. Clock skew is a deviation from the clock signal propagation rate to all the components that are dependent on the clock signal to commence their operation, whereas clock jitter is an internal deviation which produces a slight deviation in ascending or descending (or both).

The problems that result from synchronizing between two clock circuits are even more complex. When two components communicate with each other by virtue of their function (e.g., CPU communicates with DDR) and each of them runs at a different clock speed, it is necessary to synchronize the interactions between them or otherwise Overflow or Underflow problems can occur. I.e., problems in reading and writing data from one component to another, basically by overriding new data or using old data that disrupts the rate of operation of the whole device.

Because the clock signal rate determines the start of operation of each of the components in the circuit, end users have realized that the device can be forced to operate at a different clock frequency from the factory setting frequency and thus achieve faster and more effective results than the device states. This process is called Overclocking and if used correctly it can optimize the operation of the device. However if used incorrectly (under extremely high frequencies) the device can overheat. Its packaging cannot withstand the high heat emitted and it will probably catch fire.