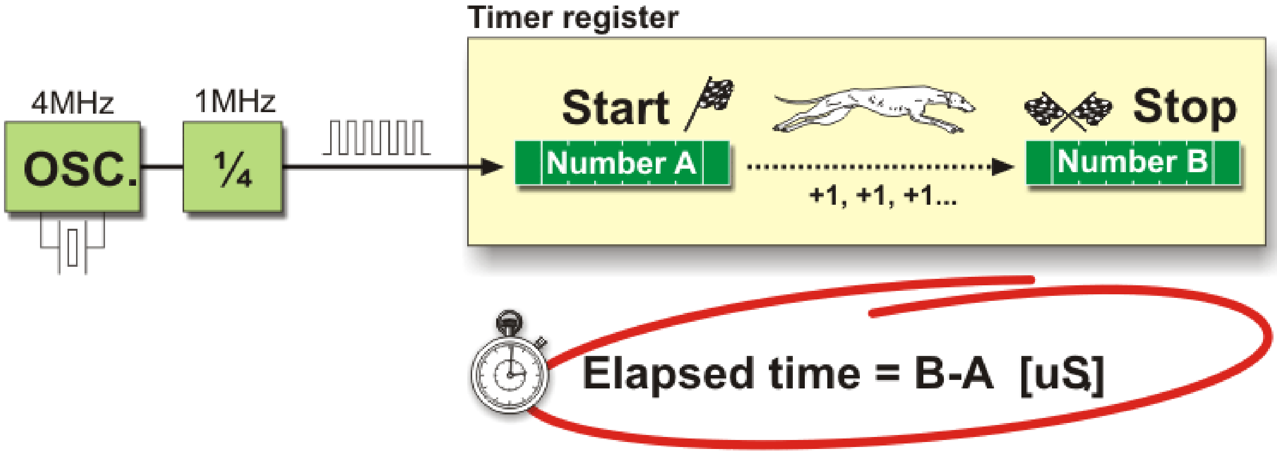
loaded - an interrupt is generated!

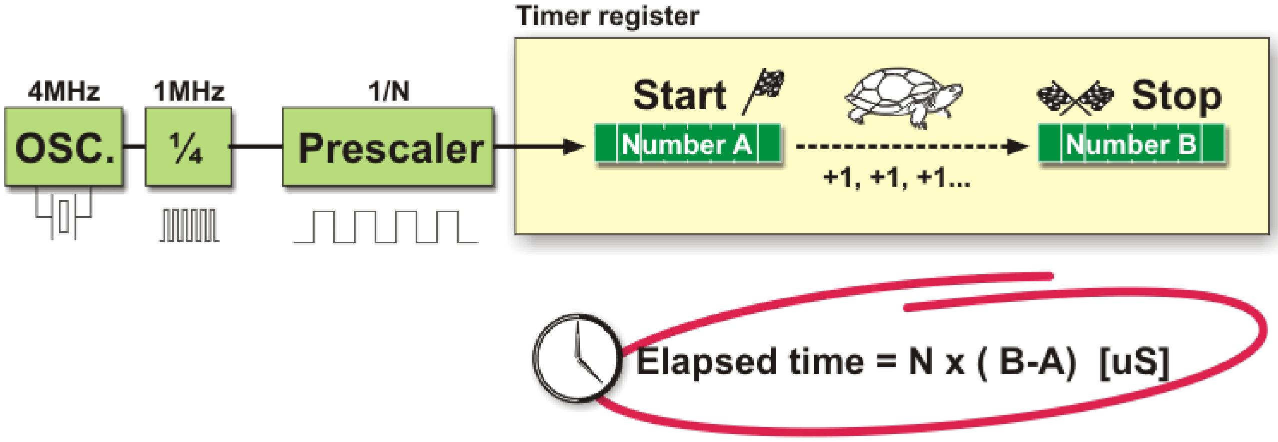
If the timer registers use internal quartz oscillator for their operation then it is possible to measure time between two events (if the register value is T1 at the moment measurement has started, and T2 at the moment it has finished, then the elapsed time is equal to the result of subtraction T2-T1). If the registers use pulses coming from external source then such a timer is turned into a counter.



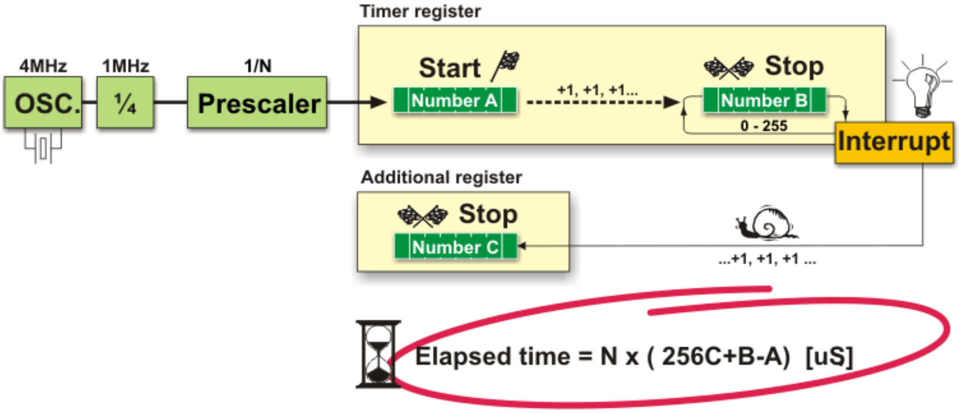
This is only a simple explanation of the operation itself.

How does a timer operate?

In practice, everything works as follows: pulses coming from quartz oscillator are once per each machine cycle directly or via pre-scaler brought to the circuit which increments number in the timer register. If one instruction (one machine cycle) lasts for four quartz oscillator periods then, by embedding quartz with the frequency of 4MHz, this number will be changed a million times per second (each microsecond).



It is easy to measure short time intervals (up to 256 microseconds) in a way described above because it is the largest number that one register can contain. This obvious disadvantage may be easily overcome in several ways by using slower oscillator, registers with more bits, prescaler or interrupts. The first two solutions have some weaknesses so it is more recommended to use prescaler and/or interrupt.



Using prescaler in timer operating

A prescaler is an electronic device used to reduce a frequency by a pre-determined factor.

Meaning that in order to generate one pulse on its output, it is necessary to bring 1, 2 , 4 or more pulses to its input. One such circuit is built in the microcontroller and its division rate can be changed from within the program. It is used when it is necessary to measure longer periods of time.

One prescaler is usually shared by timer and watch-dog timer, which means that it cannot be used by both of them simultaneously.

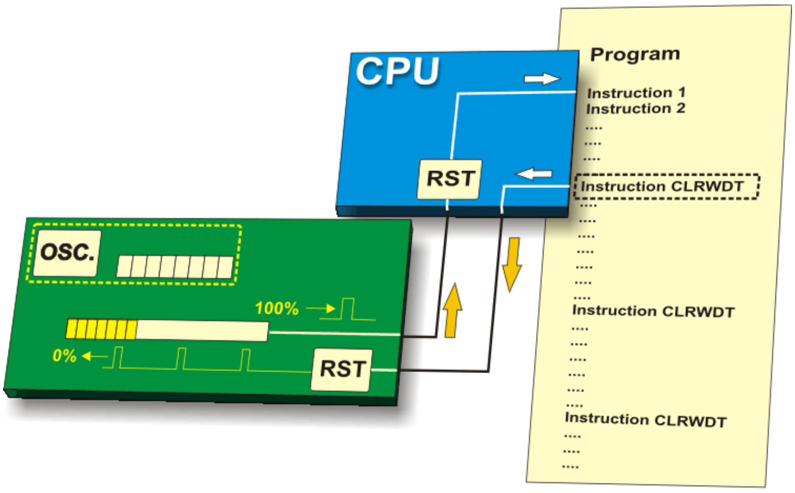
Using interrupt in timer operating

If the timer register consists of 8 bits, the largest number that can be written to it is 255 (for 16-bit registers it is the number 65535). If this number is exceeded, the timer will be automatically reset and counting will start from zero. This condition is called overflow. If enabled from within the program, such overflow can cause interrupt, which gives completely new possibilities. For example, the state of registers used for counting seconds, minutes or days can be changed in an interrupt routine. The whole this process (except interrupt routine) is automatically performed “in the background”, which enables main circuits of the microcontroller to perform other operations.

Counters

If a timer is supplied with pulses over the microcontroller input pin then it turns into a counter. Clearly, It is about the same electronic circuit. The only difference is that in this case pulses to be counted come through the ports and their duration (width) is mostly not defined. That is why they cannot be used for time measurement, but can be used to measure anything else: products on an assembly line, number of axis rotation, passengers etc. (depending on sensor in use).

Watchdog Timer

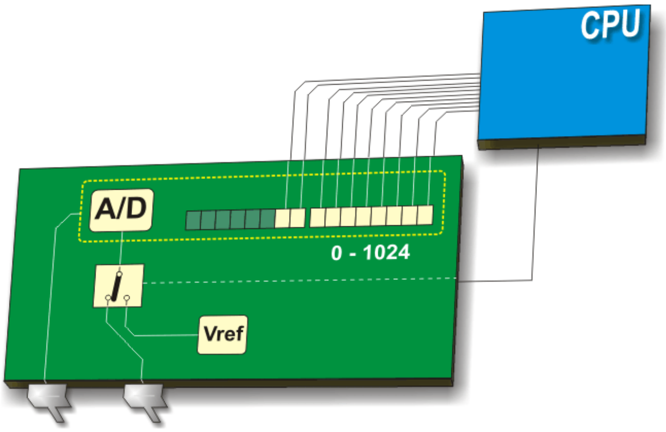


As name itself indicates a lot about its purpose. Watchdog Timer is a timer connected to a completely separate RC oscillator within the microcontroller.

If the watchdog timer is enabled, every time it counts up to end, the microcontroller reset occurs and program execution starts from the first instruction. The point is to prevent this from happening by using a specific command. The whole idea is based on the fact that every program is executed in several longer or shorter loops.

If instructions which reset the watchdog timer are set on the appropriate program locations, besides commands being regularly executed, then the operation of watchdog timer will not affect program execution. If for any reason (usually electrical noises in industry), the program counter “gets stuck” on some memory location from which there is no return, the watchdog will not be cleared and the register’s value being constantly incremented will reach the maximum et voila! Reset occurs!

A/D Converter



External signals are usually fundamentally different from those the microcontroller understands (zero and one), so that they have to be converted in order the microcontroller can understand them. An analog-to digital converter is an electronic circuit which converts continuous signals to discrete digital numbers. This module is therefore used to convert some analog value into binary number and forwards it to the CPU for further processing. In other words, this module is used for input pin voltage measurement (analog