SIMATIC
S7-200 Tips

| Group	Topic
1	**Handling of the S7-200 Timers**

Overview

Timers are used to regulate the timing of given functions within a program. The SIMATIC S7-200 series of programmable controllers features two types of timers: the On-Delay Timer (TON) and the Retentive On-Delay Timer (TONR). Both types of timers are available in three time base resolutions: 1 msec, 10 msec, and 100 msec.

This tip explains the operation and use of each timer type, with particular emphasis on the differences in timer operation between the various time base resolutions.

TON

TONR

1 msec

10 msec

100 msec

Figure 33.1
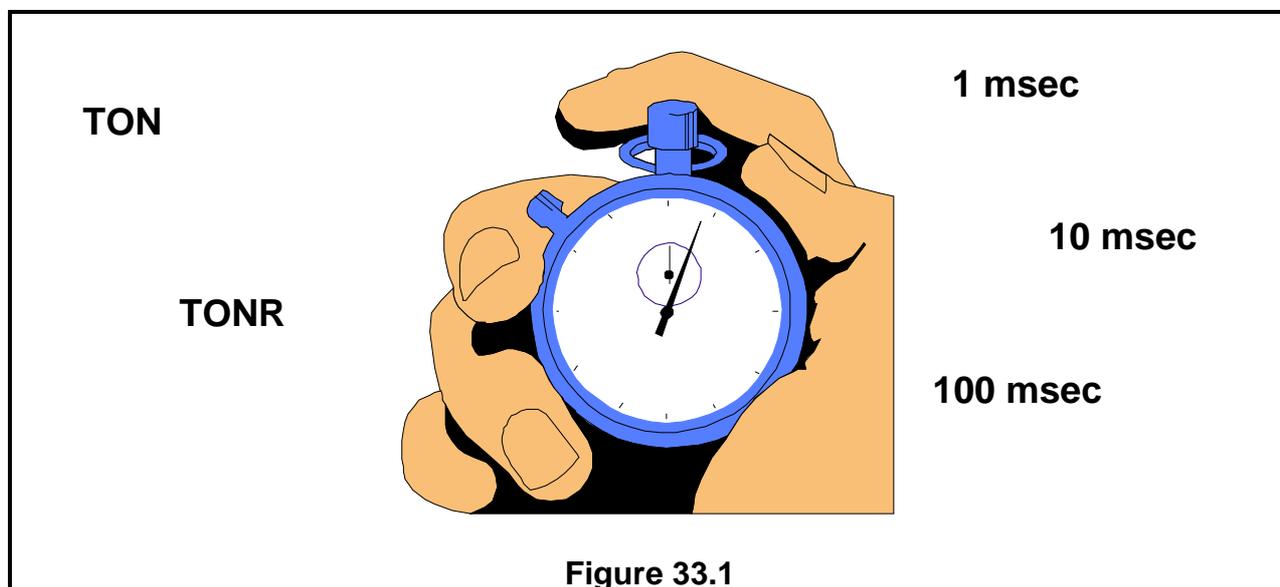
Industrial automation
Elincom Group
European Union: www.elinco.eu
Russia: www.elinc.ru

Description

General Information:

S7-200 timers are controlled with a single enabling input and have a current value that maintains the elapsed time since the timer was enabled. The timers also have a preset time value (PT) that is compared to the current value each time the current value is updated. A timer bit is set/reset based upon the result of this comparison. When the current value is greater than or equal to the preset time value, the timer bit (T-bit) is turned on. Otherwise, the T-bit is turned off. Timing stops when the current value reaches a maximum value.

When a timer is reset, its current value is set to zero, and its T-bit is turned off. Timers can be reset with the Reset instruction (this is the only way a TONR timer can be reset). Writing a zero to a timer's current value will not reset its timer bit. Likewise, writing a zero to a timer's T-bit will not reset its current value.

The two types of timers (TON and TONR) differ in the ways in which they react to the state of the enabling input. Both TON and TONR timers time up while the enabling input is on and do not time up while the enabling input is off. But when the enabling input is off, a TON timer will automatically reset, and a TONR timer will not reset. Therefore, the TON timer is best utilized when timing a single interval, while the TONR timer is more appropriate when it is necessary to accumulate a number of timed intervals.

TON and TONR timers are available in resolutions of 1 msec, 10 msec, and 100 msec, and have a maximum value of 32.767 seconds, 327.67 seconds, and 3276.7 seconds, respectively. The resolution is determined by the timer number, as shown in the chart below. Each count of the current value is a multiple of the time base. For example, a count of 50 on a 10 msec timer represents 500 msec.

	1 msec	**10 msec**	**100 msec**
TON	CPU 212: T32 CPU 214: T32, T96	CPU 212: T33-T36 CPU 214: T33-T36, T97-T100	CPU 212: T37-T63 CPU 214: T37-T63, T101-T127
TONR	CPU 212: T0 CPU 214: T0, T64	CPU 212: T1-T4 CPU 214: T1-T4, T65-T68	CPU 212: T5-T31 CPU 214: T5-T31, T69-T95

Figure 33.2

TON/TONR timers with 1 msec resolution:

In order to maintain the precision of the 1 msec timers, they are updated once per millisecond in the system interrupt routine that maintains the system time base. The current value is updated by incrementing it by 1, unless it is already at the maximum value.

The interrupt routine maintaining the 1 msec system time base is independent of the enabling and disabling of timers; this means a given 1 msec timer will be enabled at a point somewhere within the current 1 msec interval. Therefore, the timed interval for a given 1 msec timer can be up to 1 msec short. The user should program the preset time value to a value 1 greater than the minimum desired timed interval. For example, to guarantee a timed interval of at least 56 msec using a 1 msec timer, the preset time value should be set to 57.

Since the current value of an active 1 msec timer is updated in a system routine, the update is automatic. Once enabled, execution of a 1 msec timer's controlling TON/TONR instruction is required only to control the off/on state of the timer.

To keep the execution time of this system interrupt routine at a manageable level, only a small number of 1 msec timers can be supported. Since the 1 msec timers are updated from within an interrupt routine, the current value and the T-bits of these timers can be updated anywhere in the scan and will be updated more than once per scan if the scan time exceeds one millisecond. Therefore, these values are not guaranteed to remain constant throughout a given execution of the main user program.

A reset of an enabled 1 msec timer will turn it off, in addition to zeroing its current value and clearing its T-bit.

TON/TONR timers with 10 msec resolution:

In order to support a larger number of 10 ms timers with precision, all enabled 10 msec timers are updated at the beginning of each scan by adding a 10 msec accumulation value to the current value. This accumulation value is determined at the beginning of the scan and consists of the number of 10 msec intervals that have transpired since the last update (the beginning of the previous scan). This single accumulation value is applied to all enabled 10 msec timers.

The process of accumulating 10 msec intervals is performed independently of the enabling and disabling of timers, so a given 10 msec timer will be enabled at a point somewhere within the current 10 msec interval. Therefore, the timed interval for a given 10 msec timer can be up to 10 msec short. The user should program the preset time value to a value 1 greater than the minimum desired timed interval. For example, to guarantee a timed interval of at least 140 msec using a 10 msec timer, the preset time value should be set to 15.

Since the current value of an active 10 msec timer is updated at the beginning of the scan, the update is automatic. Once enabled, execution of a 10 msec timer's controlling TON/TONR instruction is required only to control the on/off state of the timer. Unlike the 1 msec timers, a 10 msec timer's current value is updated only once per scan and will remain constant throughout a given execution of the main user program.

A reset of an enabled 10 msec timer will turn it off, in addition to zeroing its current value and clearing its T-bit.

TON/TONR timers with 100 msec resolution:

The largest number of timers are 100 msec timers, and correspondingly, they have the lowest precision. All enabled 100 msec timers are updated when the enabling timer instruction is executed by adding a 100 msec accumulation value to the current value. This accumulation value, similar to the 10 msec accumulation value, is determined at the beginning of the scan and consists of the number of 100 msec intervals that have transpired since the last update (the beginning of the previous scan). This single accumulation value is applied to all enabled 100 msec timers as they are updated.

The process of accumulating 100 msec intervals is performed independently of the enabling and disabling of timers, so a given 100 msec timer will be enabled at a point somewhere within the current 100 msec interval. Therefore, the timed interval for a given 100 msec timer can be up to 100 msec short. The user should program the preset time value to a value 1 greater than the minimum desired timed interval. For example, to guarantee a timed interval of at least 2100 msec using a 100 msec timer, the preset time value should be set to 22.

The update of 100 msec timers is not automatic, since the current value of a 100 msec timer is updated only if the timer instruction is executed. Consequently, if a 100 msec timer is enabled, but the timer instruction is not executed each scan, the current value for that timer will not be updated and it will lose time. Likewise, if the same 100 msec timer instruction is executed multiple times in a single scan, the 100 msec accumulation value will be added to the timer's current value multiple times, and it will gain time. Therefore, 100 ms timers should only be used where the timer instruction will be executed exactly once per scan.

A reset of a 100 msec timer will zero its current value and clear its T-bit.

Examples:

The effect of updating the current value of 1 msec, 10 msec, and 100 msec timers at different times depends on how the timers are used. For example, consider the timer operation in the following program.

Automatically Re-triggered Oneshot Timer

Using a 1 msec timer   Using a 10 msec timer   Using a 100 msec timer
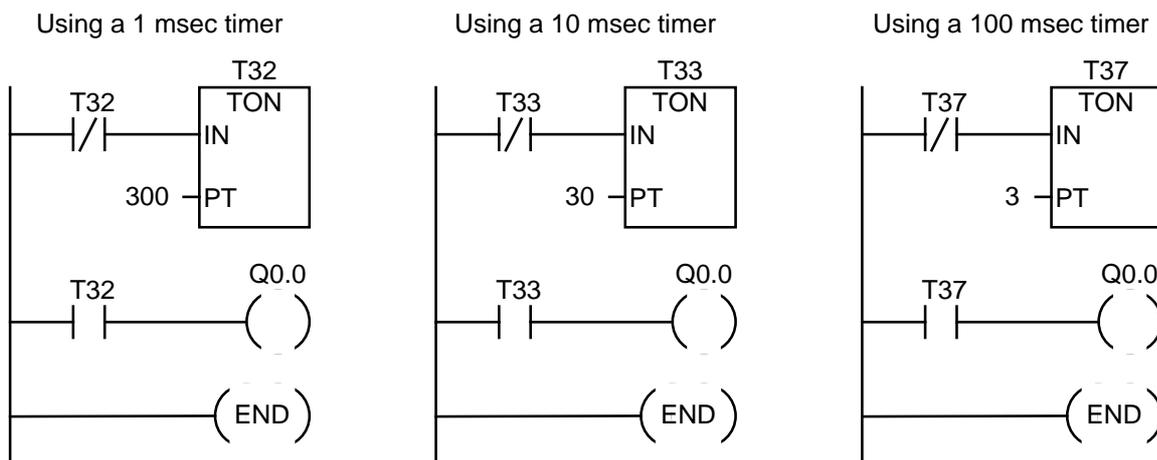
Figure 33.3

In the case where the 1 msec timer is used, Q0.0 will be turned on for one scan whenever the timer's current value is updated, after the normally closed contact T32 is executed, and before the normally open contact T32 is executed.

In the case where the 10 msec timer is used, Q0.0 will never be turned on, because the timer bit T33 will be turned on from the top of the scan to the point where the timer box is executed. Once the timer box has been executed, the timer's current value and its T bit will be set to zero. When the normally open contact T33 is executed, T33 will be off and Q0.0 will be turned off.

In the case where the 100 msec timer is used, Q0.0 will always be turned on for one scan whenever the timer's current value reaches the preset value.

A very simple modification can be made to the ladder programs above that will produce the same result for all three timers. This modification is shown below.
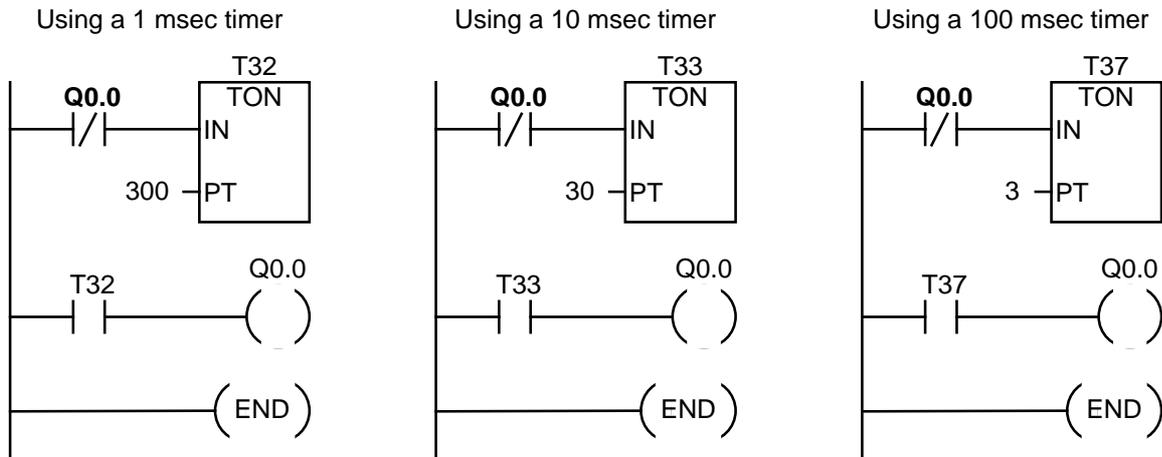
Automatically Re-triggered Oneshot Timer (Fixed)

Using a 1 msec timer	Using a 10 msec timer	Using a 100 msec timer

Figure 33.4

By using the normally closed contact Q0.0 instead of the timer bit as the enabling input to the timer box, the output Q0.0 is guaranteed to be turned on for one scan each time the timer reaches the preset value.

General Notes

The SIMATIC S7-200 Application Tips are provided to give users of the S7-200 some indication as to how, from the view of programming technique, certain tasks can be solved with this controller. These instructions do not purport to cover all details or variations in equipment, nor do they provide for every possible contingency. Use of the S7-200 Application Tips is free.

Siemens reserves the right to make changes in specifications shown herein or make improvements at any time without notice or obligation. It does not relieve the user of responsibility to use sound practices in application, installation, operation, and maintenance of the equipment purchased. Should a conflict arise between the general information contained in this publication, the contents of drawings or supplementary material, or both, the latter shall take precedence.

Siemens is not liable, for whatever legal reason, for damages or personal injury resulting from the use of the application tips.

All rights reserved. Any form of duplication or distribution, including excerpts, is only permitted with express authorization by SIEMENS.