



LONMARK[®] Functional Profile: Real Time Keeper

3300-10 © 1997, LONMARK Interoperability Association

Echelon, LON, LONWORKS, LONMARK, and the LONMARK logo are trademarks of Echelon Corporation registered in the United States and other countries.

Overview

The basic function of this object is to provide the real time on the network. The real time information on the network can be used to “time-stamp” events (e.g. alarms) and to synchronize schedulers. A device using this object will have built in Time Keeping hardware (e.g. a battery backup time keeper, or an AM receiver). Optionally a time and date display could be connected to the device for display purposes, and also hardware inputs for manual adjustment of the time. There is no embedded time and/or date encryption method provided in this object. This may be a problem for security systems, but for lighting it is not deemed necessary.

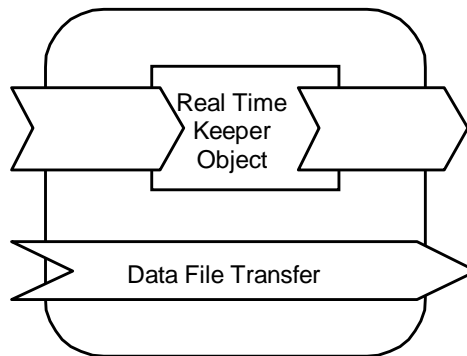


Figure 1.1 Real Time Keeper Functional Profile

Example Usage

Each LONMARK device has embedded within it a Node Object with the network input variable `nviTimeSet` (SNVT_time_stamp). The real time keeper object will provide at regular intervals real time updates to those devices which require the actual time of day for e.g. time stamping alarm messages through the node object NV.

It should be possible to have more than one time keeper in the network, if this is the case only one should be defined as master, for synchronization. If the master timing is not received the slave node shall provide the nodes internal real time (timer redundancy).

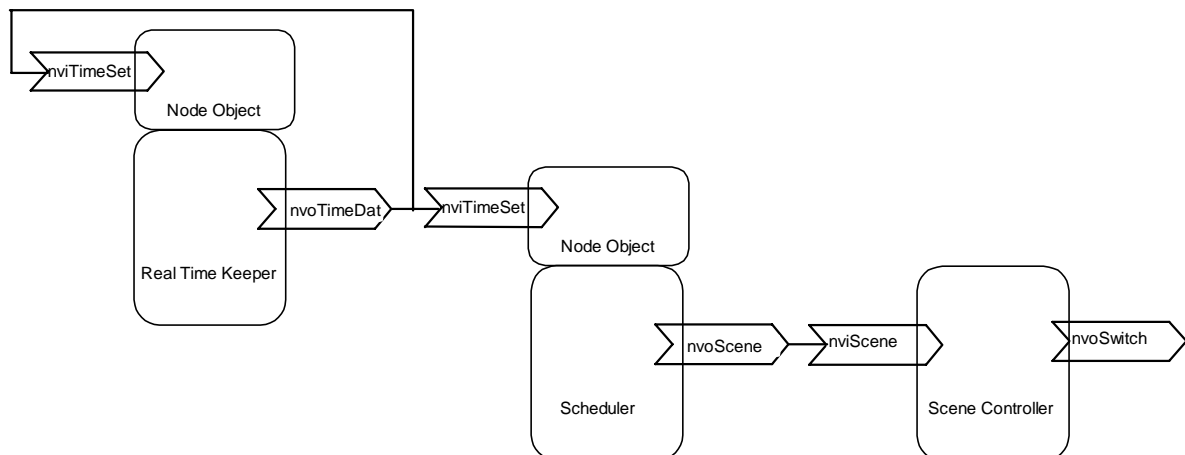


Figure 1.2 Example Usage of Real Time Keeper Object

Object Details

The real time keeper object provides the actual real time on the network.

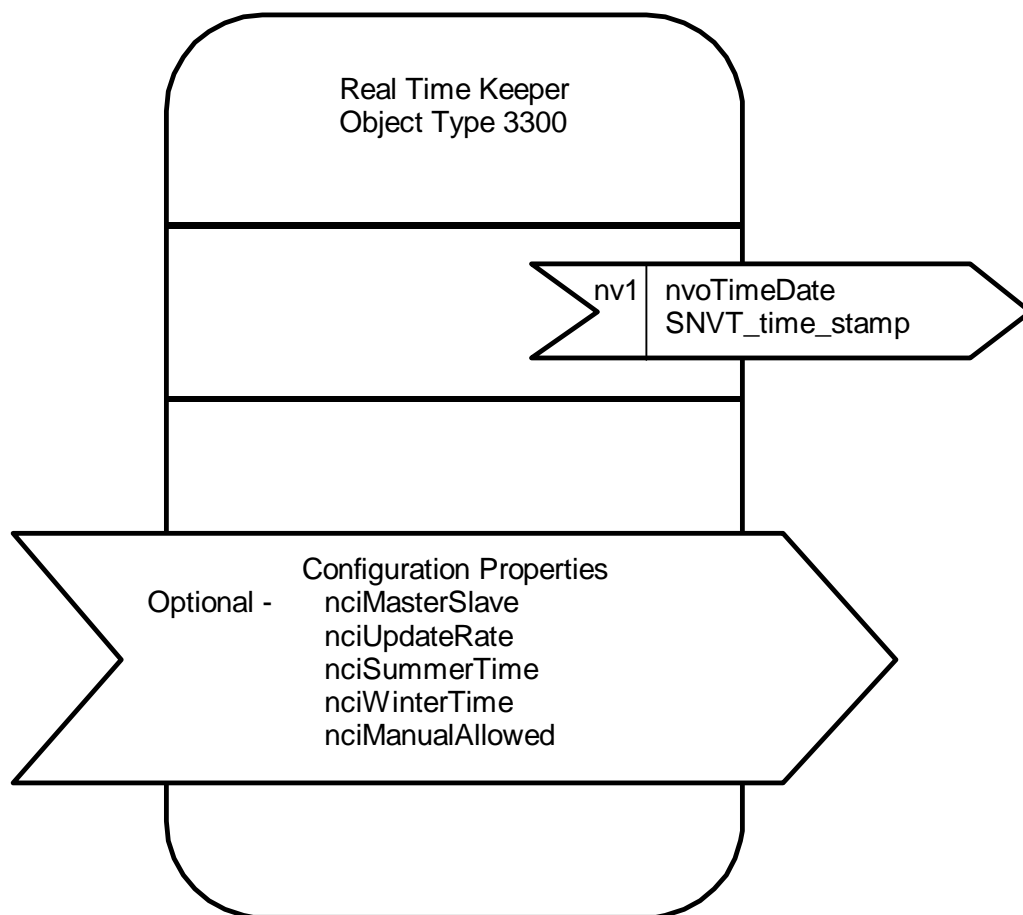


Figure 1.3 Real Time Keeper Object Details

Table 1.1 SNVT Details

NV # (M/O)*	Name	In/Out	SNVT Type (SNVT Index)	Class	Description
1 (M)	nvoTimeDate	Out	SNVT_time_stamp (#84)	nv	Real Time output to the network

* M = mandatory, O = optional

Table 1.2 SCPT Details

SCPT index (M/O)*	Name	Description
97 (O)	SCPTMasterSlave	Master Slave operation, if more than 1 time keeper in the network
98 (O)	SCPTUpdateRate	Time stamp Update Rate
99 (O)	SCPTSummerTime	Start Summer date and time
100 (O)	SCPTWinterTime	Start Winter date and time
101 (O)	SCPTManualAllowed	Manual update allowed

* M = mandatory, O = optional

Mandatory Network Variables

The only mandatory network variable in the object is `nvoTimeDate` variable. For those hardware implementations without an external time synchronisation (AM driven timers) the node object shall contain the `nviTimeSet` to enable time and date settings for synchronisation.

Variable Name Output

```
network output SNVT_time_stamp nvoTimeDate;
```

This output network variable provides the actual real time and date on the network for all linked objects.

Valid Range

The complete range of the `SNVT_time_stamp`.

When Transmitted

If master slave indication is set to “master”, every Time Update interval. If the object is working as a slave object, then only when a manual time setting is performed on the host module.

Update Rate

Configurable, default 1 minute.

Default Service Type

Unacknowledged repeated.

Optional Network Variables

There are no optional network variables.

Mandatory Configuration Properties

There are no mandatory configuration properties.

Optional Configuration Properties

The object has following configuration properties:

Master Operation:	definition of the master timer in a multi timer solution
Time Update:	heart beat of time updates
Summer date & time:	change over to summer time definition
Winter date & time:	change over to winter time definition
Manual allowed:	allow manual updating on product

Master Operation

```
network input config SNVT_Boolean nciMasterSlave;
```

This input configuration (network) variable provides the possibility to set this object as a master Clock or as a slave Clock, in case of multiple clock use.

Valid Range

TRUE or FALSE

Default Value

TRUE

Only one time keeper in the network

SCPT Reference

SCPTMasterSlave #97

Time Broadcast

```
network input config SNVT_time_sec nciUpdateRate;
```

This input configuration (network) variable provides for a Clock in master configuration the update rate of the output nvoTimeDate network variable.

Valid Range

1 minute, 1 hour, 1 day

Default Value

1 minute

SCPT Reference

SCPTUpdateRate #98

Summer date and time

```
network input config SNVT_time_stamp nciSummerTime;
```

This input configuration (network) variable provides the definition of the start of the summer time. At the defined Summer date and time the Clock will increase its time automatically with one hour.

Valid Range

1 January 0 hours to 31 December 23 hours.
Year, Minutes and Seconds are ignored.

When not used entry shall be all "0".

Default Value

Not used.

SCPT Reference

SCPTSummerTime #99

Winter date and time

```
network input config SNVT_time_stamp nciWinterTime;
```

This input configuration (network) variable provides the definition of the start of the winter time. At the defined Winter date and time the Clock will decrease its time automatically with one hour.

Valid Range

1 January 0 hours to 31 December 23 hours.
Year, Minutes and Seconds are ignored.

When not used entry shall be all "0".

Default Value

Not used.

SCPT Reference

SCPTWinterTime #100

Manual allowed

```
network input config SNVT_Boolean nciManualAllowed;
```

This input configuration network variable provides for a Clock with a manual time update input the possibility to discard manual time updating for this clock.

Valid Range

TRUE or FALSE

Default Value

TRUE

SCPT Reference

SCPTManualAllows #101

Data Transfer

No File transfer for data in this object.

Power-up State

No action at Power Up. The object (if designated Master) will transmit on the first occurrence of the update definition the time over the network.

Boundary and Error Conditions

The summer and winter time should not be at the same moment. If this becomes the case than no change to winter or summer time is executed.

When the internal time keeper is not functional, then an alarm of the type (1) is generated via the node object.

If the timer operates in a slave mode, and when no updates from the Master timer are received, the timer will provide the other objects in the node with real time information.

Additional Considerations

The object can have following states:

in_alarm	internal timer malfunction
self_test_in_progress	when factory test (e.g. verification of the working of the hardware time keeper is in progress

The object can be requested to do a manufacturing test.