

Exception: "ProcessorLoadWatchdog"

This watchdog must be activated in the [CODESYSControl.cfg](#) configuration file.

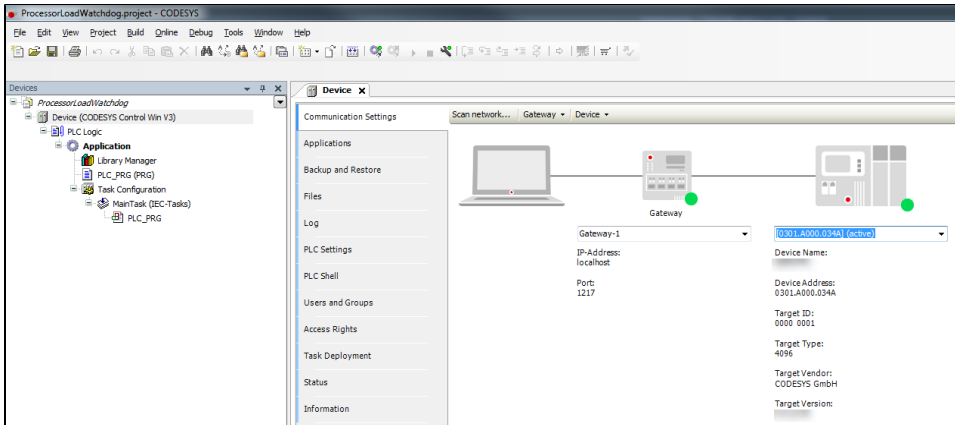


Depending on the runtime, the configuration file is located in the following place: [Location of the configuration file](#)

The limit is defined in the CmpSchedule section.

Experience has shown that communication problems occur more often in the case of controllers under high loads.

- Create a ["Standard project"](#)
- Select CODESYS Control Win V3 as the PLC device.
- Define the target system by means of the [Network scan](#).



- Edit the "PLC_PRG" POU as follows:

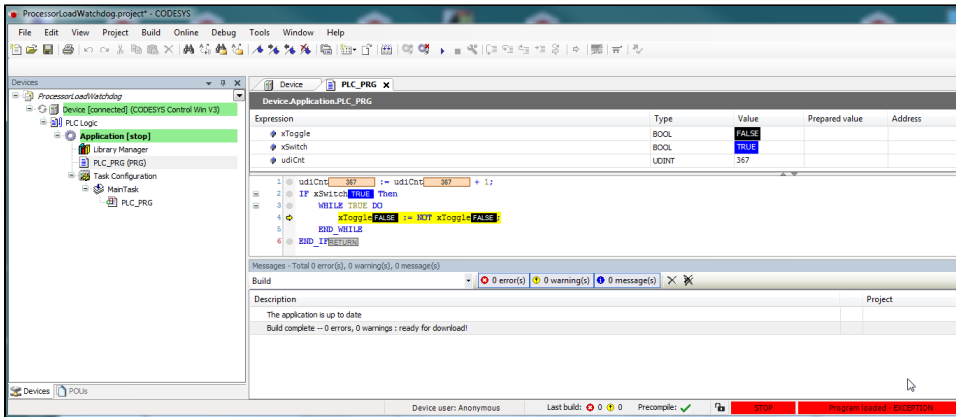
Declaration

```
VAR
    xToggle    : BOOL;
    xSwitch    : BOOL;
    udiCnt     : UDINT;
END_VAR
```

Implementation

```
udiCnt := udiCnt + 1;
IF xSwitch Then
    WHILE TRUE DO
        xToggle := NOT xToggle;
    END_WHILE
END_IF
```

- Start the project and set the "xSwitch" variable to TRUE.



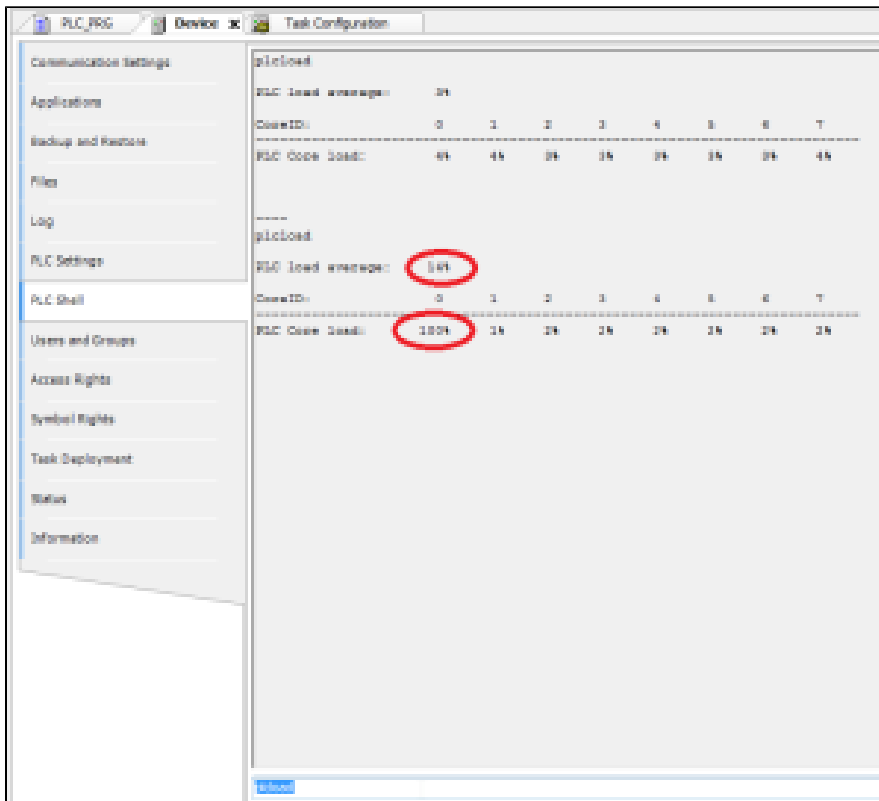
The following entries are located in the PLC log:

Severity	Time Stamp	Description	Component
Information	11.10.2018 15:17:55.616	*SOURCEPOSITION* App=[Application] area=3, offset=18097	CmpIecTask
Information	11.10.2018 15:17:55.616	*EXCEPTION* [Watchdog] occurred: App=[Application], Task=[MainTask]	CmpIecTask
Information	11.10.2018 15:17:55.616	*EXCEPTION* [ProcessorLoadWatchdog] occurred in: App=[all], Task=[all]	CmpIecTask
Information	11.10.2018 15:17:55.616	*DETAILS* Task=[MainTask] does not react within timeout switching to stop! An application reset is necessary!	CmpIecTask

Processor load and Multicore

With the introduction of multicore support (since V3.5.SP13), the processor load is calculated as the average of all CPUs. This also applies for the runtime license without multicore support.

Therefore, the following situation results in the case of an active infinite loop for an 8-core processor:



To trigger the watchdog, the limit has to be set to approximately 10% in the ini file.

```
[CmpSchedule]  
ProcessorLoad.Maximum=10
```