



Parker Motion Controller

PAC340

Quickstart Guide MultiCore



ENGINEERING **YOUR** SUCCESS.

Multicore Task Configuration

With the PAC340 CPU four different cores can be used.

CODESYS version 3.5.16 or higher must be used for this.

Under „Task Configuration-> Task Groups“ additional task groups can be added and the scheduling algorithm for the task group can be selected. Then tasks can be added to a task group.

Motion applications using the EtherCAT fieldbus require a strong real time behavior. It is mandatory to assign core #3 to the EtherCAT task. The task itself must have priority 0.

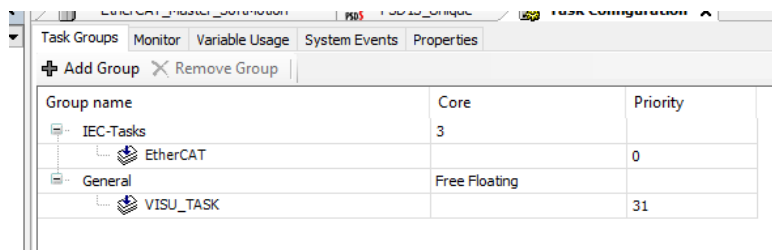
The usage of the other cores depends on the structure of the project.

But for an easy to use start we recommend having one more task group besides the EtherCAT task group and set it to “Free Floating”.

It is also possible to use only core #3 for all tasks. In this case it is very important to consider the real time behavior of all assigned tasks:

A task with real time behavior (like EtherCAT, Profinet...) can get a priority in the range of 1 .. 15.

Other tasks without real time behavior (like Visualization, blocking program parts) must get a priority in the range of 16 ..31.



The screenshot shows the 'Task Configuration' window in CODESYS. It has tabs for 'Task Groups', 'Monitor', 'Variable Usage', 'System Events', and 'Properties'. The 'Task Groups' tab is active, showing a tree structure with 'IEC-Tasks' and 'General' groups. Below the tree is a table with columns 'Group name', 'Core', and 'Priority'.

Group name	Core	Priority
IEC-Tasks	3	
EtherCAT		0
General	Free Floating	
VISU_TASK		31

In “Free Floating” mode the system is responsible for scheduling the tasks over the four cores.

More information about the multicore features and the scheduling modes can be found in the CODESYS help:

https://help.codesys.com/webapp/cds_obj_task_config_task_groups;product=codesys;version=3.5.16.20;language=en

Be aware that MultiCore systems perform real multitasking on separate cores, and thus race conditions can occur on unoptimized applications that were previously running on SingleCore systems.

Please make sure that different task groups are not writing the same variables or lock parallel writing with the help of state machines or semaphores.