Specification

# Objectives

An objective of this test suite is to confirm task related functionality of MQX.

## Reference documentation

| Document Name | Version |
| --- | --- |
| 1. MQX MFS User Guide | 2.2 |

Table 1. Reference documentation

## Test environment

|  |  |
| --- | --- |
| **Software Item Name** | **Version** |
| IAR ARM | 6.10.1 |

Table 2. Required software resources

|  |  |
| --- | --- |
| **Hardware Item Name** | **Version** |
| TWR – K60N512 | N/A |

Table 3. Required hardware resources

# API

## General API

|  |  |
| --- | --- |
| *API name* | *Test app containing API* |
| \_task\_create | restart, abort |
| \_task\_destroy | restart, abort |
| \_task\_restart | restart |
| \_task\_abort | abort |
| \_task\_block | restart, abort |

# specifications

This test suite is designed to confirm task related functionality of MQX. Tests are resigned so they also utilize and test user mode functionality.

## Test app – ABORT

### Test case #1 – Testing \_task\_abort

At the beginning application tries to abort not existing task, system task and already destroyed task. Then test task is created and aborted in cycle for TEST\_COUNT times. After that another test cycle begins, but with this difference: test task is aborted from ISR raised by timer interrupt. Abortion is confirmed with free memory comparison prior task create and after task abort.

Expected result: Attempts to abort system task, destroyed task and not existed task must return error. Aborting task should release resources owned by task. Task should by abort able also from ISR

API used: \_task\_abort, \_task\_create, \_task\_destroy, \_task\_block

UserMode note: if UserMode is enabled all three testing task (main, test, test2) are running in user mode. Super task is always running in privilege mode, because it’s installing ISR and this operation is allowed only in privilege mode.

## Test app -- RESTART

### Test case # 1 – Testing \_task\_restart

At the beginning application tries to restart not existing task, system task and already destroyed task. Then test task is created and restarted in cycle for TEST\_COUNT times. Test task is then restarted from ISR.

Expected results: restarting nonexistent task, system task and destroyed task should end with error. After restart task descriptor should be same as prior restarting.

API used: \_task\_restart, \_task\_create, \_task\_destroy, \_task\_block

Usermode note:

# known issues

# possible improvements

**Revision SHEET**

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision Date** | **Author** | **Description of Revision & Writer** | **Spec Coord.** |
| 16.11.2011 | Michal Starecek | Initial version |  |