

# Testing Ptask Library

Luca Donzella and Michele Scaletti

We have created 4 types of test to use all the features offered by the library Ptask.

## Test Type:

### 1. System Test

It allows to test the system configuration such as:

- scheduler algorithm
- protocol of access to resources
- the ability to run a task on one or different core each time the task becomes active ;

### 2. Activation Task Test

It provide the opportunity to try the three different modes of creation-activation:

- creation with activation
- creation with deferred activation
- creation with deferred activation with time offset

### 3. Advanced Creation Test

It allows you to create a task with all parameters. It also provides the ability to change some parameters during the execution of the task.

### 4. Handling Mode Test

It allow you to test Handling Mode Changes.

In the following pages show the screenshot of output of the various tests.

System Test.....	2
Scheduler Test.....	2
Protocol Test.....	3
Partitioning Test.....	3
Activation Test.....	4
Advanced Creation Test.....	4
Handling Mode Test.....	6

# System Test

To compile the program test, enter into directory SystemTest/ and type: make.

Before running the tests, remember to become super-user, otherwise Linux will not allow you to create real-time tasks!

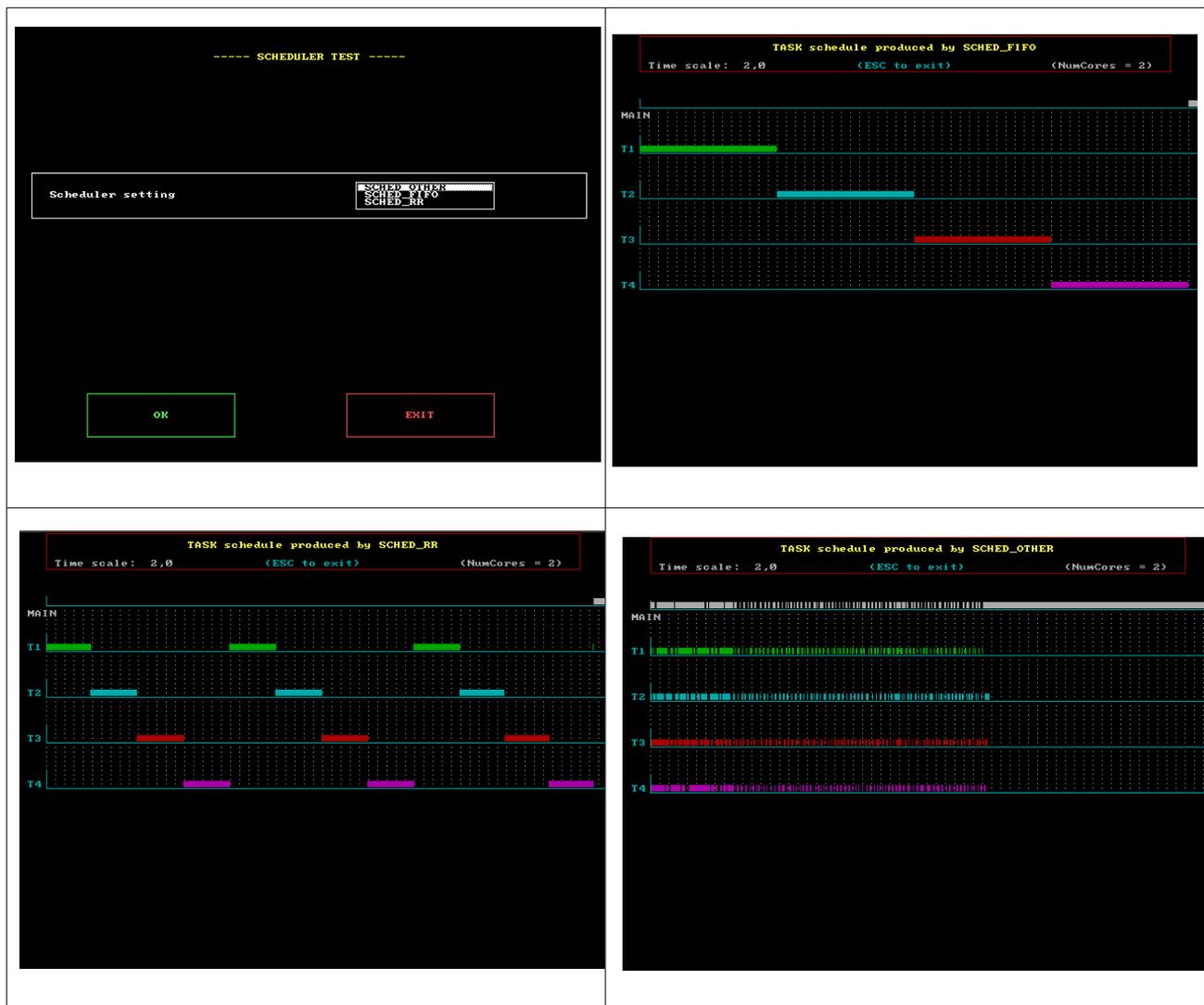
## Scheduler Test

To run Scheduler Test use `taskset 0x00000001 ( mask of processor #0 )`.

Command example : `sudo taskset 0x00000001 ./schedulerTest`

Taskset is used to set or retrieve the CPU affinity of a running process given its PID or to launch a new COMMAND with a given CPU affinity.

The parameters of the task that you can insert or modify as priority , period, deadline and computation time are in the file `scheduler.dat` into directory SystemTest/ .



# Protocol Test

To run Protocol Test use ./protocolTest.

The parameters of the task that you can insert or modify are in the file protocol.dat into directory SystemTest/ .

The Protocol Test interface consists of four main panels. The top-left panel shows the 'Protocol setting' menu with options: `PRIO_INHERITANCE`, `PRIO_CEILING`, and `NO_PROTOCOL`. The top-right panel shows the execution results for `PRIO_INHERITANCE` with a time scale of 2.0 and 2 cores. The bottom-left panel shows the execution results for `PRIO_CEILING` with the same parameters. The bottom-right panel shows the execution results for `NO_PROTOCOL` with the same parameters. Each execution panel displays a Gantt chart for three tasks (T1, T2, T3) and includes the text '(ESC to exit)'.

# Partitioning Test

To run Partitioning Test use ./partitioningTest.

The Partitioning Test interface has two main panels. The left panel shows the 'Partitioning setting' menu with options: `PARTITIONED` and `GLOBAL`. The right panel displays system configuration: `PROTOCOL = NO_PROTOCOL`, `NUMCORES = 2`, `PARTITIONING = GLOBAL`, `SCHEDULER = OTHER`, and `PRIORITY = ALL DIFF.`. Below this is a diagram of task activation on two cores (0 and 1) for tasks T1 through T7. A callout box points to the diagram with the text: 'Show System information'. Another callout box points to the activation sequence table with the text: 'In "PARTITIONED" mode the core for a task is static, it's not change'. The activation sequence table is as follows:

Task	Core
T1	run on core - 0 -
T2	run on core - 0 -
T3	run on core - 1 -
T4	run on core - 0 -
T5	run on core - 1 -
T6	run on core - 0 -
T7	run on core - 1 -

At the bottom of the right panel, it says: `NEW [X] to migrate T1`, `REV [1-9] to activate tasks`, and `ESC exit`.

# Activation Test

To compile this program test, enter into directory ActivationTaskTest/ and type: make.

To run Activation Test use ./taskFunTest.

--- TASK FUNCTION TEST ---

ACTIVATION FLAG setting:

OK EXIT

PROTOCOL = NO\_PROTOCOL -- PARTITIONING = GLOBAL -- SCHEDULER = RR  
 NumCores = 2 MODE = DEFERRED (no time offset) PRIORITY = ALL EQUAL  
 Time reference t0 = 0 (time creation/visualization of this screen)

T1  
T2  
T3

ACTIVATION SEQUENCE = T1 T2 T3

	TIME ACTIVATION(ms)	TIME START(ms)	TIME OFFSET measured(ms)
T1:	3129	3129	
T2:	3299	3299	
T3:	3599	3599	

KEY [1-9] to activate tasks (mod != MOD\_DEF\_OFFSET) ESC exit

# Advanced Creation Test

To compile this program test, enter into directory AdvancedTest/ and type: make.

To run Activation Test use ./taskFunTest.

The parameters of the task as : priority , period, deadline and cpu id are in the file param.dat into directory AdvancedTest/ .

PROTOCOL = NO\_PROTOCOL -PARTITIONING = PARTITIONED- SCHEDULER = OTHER  
 NumCores = 2 PRIORITY = ALL DIFF.

T0  
T1  
T3  
T4

id Task	period	deadline	cpu id	priority
0	20	20	0	80
1	30	30	0	80
2	40	40	0	80
3	1000	1000	1	79
4	50	50	1	79

KEY [ M ] to terminate tasks and show measure  
 KEY [ A ] to activate tasks whit act\_flag = 0 ESC exit

PROTOCOL = NO\_PROTOCOL -PARTITIONING = PARTITIONED- SCHEDULER = OTHER  
 NumCores = 2 PRIORITY = ALL DIFF.

T0  
T1  
T2  
T3  
T4

id Task	period	deadline	cpu id	priority
0	20	1	0	80
1	30	30	0	80
2	40	40	0	80
3	500	1000	1	79
4	50	50	1	79

index task:  select param:  insert value:  OK

KEY [ M ] to terminate tasks and show measure  
 KEY [ A ] to activate tasks whit act\_flag = 0 ESC exit

# Handling Mode Test

To compile this program test, enter into directory HandlingModeTest/ and type: make.

To run Handling Mode Change Test use ./modeTest.

The image displays two screenshots of a terminal-based program titled "HANDLING MODE CHANGE TEST".

**Left Screenshot:** Shows a menu for five tasks. Each task has two mode selection options (Mode 1 and Mode 2). The options are represented by small boxes containing "DEFAULT", "MODE\_A", and "MODE\_B". A callout box points to these boxes with the text: "For every task you can select one or two mode". At the bottom, there are two buttons: "OK" (green) and "EXIT" (red).

**Right Screenshot:** Shows the program's configuration and active mode. At the top, it lists: "PROTOCOL = NO\_PROTOCOL", "-- PARTITIONING = GLOBAL --", "SCHEDULER = FIFO", and "TASK CREATED : 5". Below this is a list of tasks (T1 to T5) with their active modes. A callout box points to the "MODE ACTIVE = MODE\_A" line with the text: "Actual active mode". Below that is a key legend: "KEY [ Z ] to active DEFAULT mode", "KEY [ X ] to active MODE\_A", and "KEY [ C ] to active MODE\_B". A callout box points to this legend with the text: "Key for change active mode". At the bottom right, it says "ESC exit".