

Graduation project

EDF Scheduler Implementation

The Content :

- EDF Scheduler Implementation.
- Example And Test The EDF Scheduler:
 - Analytic Designing.
 - Offline Designing.
 - Monitoring the System And Run Time Analysis.

EDF Scheduler Implementation

- (1) At the first create the EDF ready list.
- (2) Use the EDF ready list instead of the normal list.
- (3) Going to “`tskTaskControlBlock`” and add the period as a parameter.
- (4) Going to “`prvInitialiseTaskLists`” and initialize the EDF ready list.
- (5) Create a new API “`xTaskPeriodicCreate`” and update the list item value by deadline
- (6) Going to “`vTaskSwitchContext`” to select a new task to run according to its deadline.
- (7) Going to “`vTaskStartScheduler`” and create the idle task by “`xTaskPeriodicCreate`” with high deadline.
- (8) Going the idle task “`portTASK_FUNCTION`” and update the deadline and set the list item value with the new deadline
And check if there are other tasks in EDF ready list.

```
#if (configUSE_EDF_SCHEDULER == 1 )
listSET_LIST_ITEM_VALUE( &(amp; (pxCurrentTCB)->xStateListItem ), (pxCurrentTCB)->xTaskPeriod + xTaskGetTickCount());
#endif

#if (configUSE_EDF_SCHEDULER == 0 )
{
if( listCURRENT_LIST_LENGTH( &(amp; pxReadyTasksLists[ tsxIDLE_PRIORITY ] ) ) > ( UBaseType_t ) 1 )
{
taskYIELD();
}
else
{
mtCOVERAGE_TEST_MARKER();
}
}
#else
{
if( listCURRENT_LIST_LENGTH( &(xReadyTasksListEDF) ) > ( UBaseType_t ) 1 )
{
/*if there is a task in the ready list other than the IDLE task */
taskYIELD();
}
else
{
mtCOVERAGE_TEST_MARKER();
}
}
#endif
```

- (9) Going to “**xTaskIncrementTick**” and every tick we need to update the deadline of current task before context switching or adding to the ready list

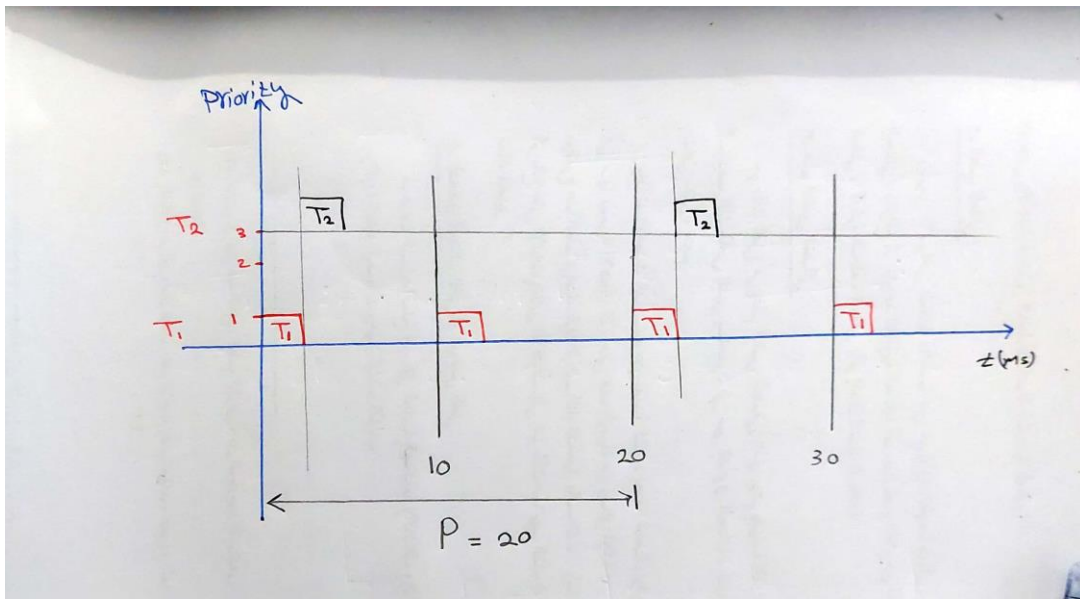
```
/*UPDATING THE DEADLINE OF THE CURRENT TASK*/
#if(configUSE_EDF_SCHEDULER==1)
listSET_LIST_ITEM_VALUE(&( ( pxTCB )->xStateListItem ), ( ( pxTCB )->xTaskPeriod + xTaskGetTickCount() ) );
#endif
/* Place the unblocked task into the appropriate ready list. */
prvAddTaskToReadyList( pxTCB );
/*After adding a task to the EDF Ready list a context switch is required */
/*So xSwitchRequired is a flag that indicate a need for context switch*/
xSwitchRequired = pdTRUE;
```

Example And Test The EDF Scheduler

1- Analytic Designing

- Task1 $T_1\{P:10, E:0.414, D:10\}$ the priority is 1
- Task2 $T_2\{P:10, E:0.414, D:20\}$ the priority is 3

*Timeline:



the second task has a higher priority than the first task

but the first task executes before the second task because the deadline of the first task is close more than the second task .

$$\text{CPU_load} = (((E_1 \cdot n_1) + (E_2 \cdot n_2)) / P) \cdot 100$$

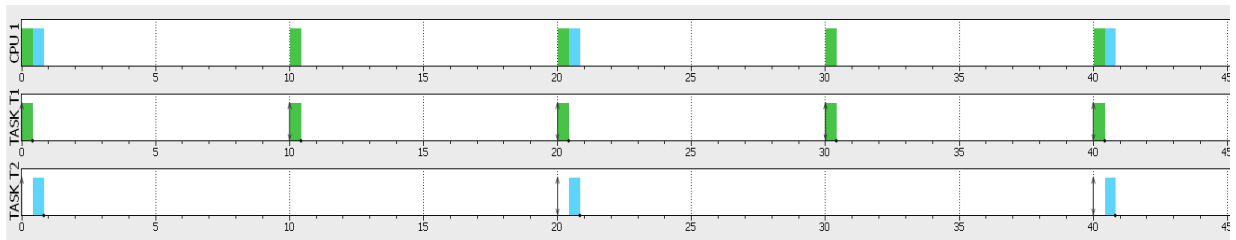
$$= (((0.414 \cdot 2) + (0.414 \cdot 1)) / 20) \cdot 100 = 6.2$$

2- Off line Designing

(1) Add the tasks into “Simo”

Qt Model data									
General Scheduler Processors Tasks									
id	Name	Task type	Abort on miss	Act. Date (ms)	Period (ms)	List of Act. dates (ms)	Deadline (ms)	WCET (ms)	Followed by
1	TASK T1	Periodic	<input type="checkbox"/> No	0	10	-	10	0.414	
2	TASK T2	Periodic	<input type="checkbox"/> No	0	20	-	20	0.414	

(2) Run the system



As expected task1 executes before task2

(3) Read the information

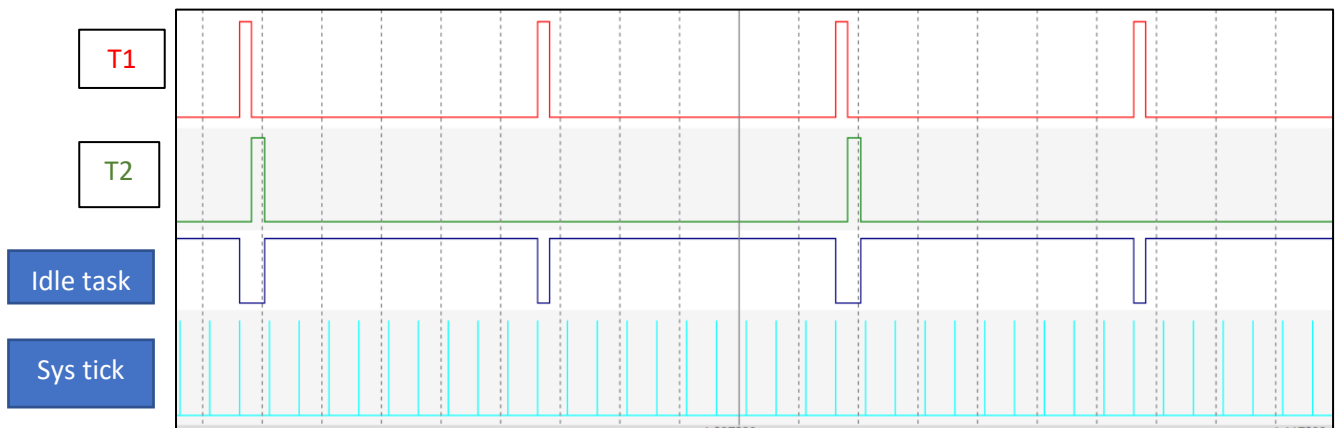
Qt Results			
General Logs Tasks Scheduler Processors			
Observation Window:			
from 0.00 to 100.00 ms		Configure...	
	Total load	Payload	System load
CPU 1	0.0621	0.0621	0.0000
Average	0.0621	0.0621	0.0000

As expected the **CPU_load** is 6.2

3- Monitoring the System And Run Time Analysis

- Use idle and system tick hooks with GPIO and the analyzer to monitor the idle task and system tick
- Use the Trace hooks with GPIO and the analyzer to monitor the tasks and calculate the CPU_load and execution time for each task

After building the system :



As expected Task1 has lower priority but execute before task2

Watch 1		
Name	Value	Type
cpu_load	6.20386028	float
task1_totalTime	163245	int
task2_totalTime	81620	int
<Enter expression>		

As expected and calculated **CPU_load** is 6.2