Mälardalen University
School of Innovation Design and Engineering
Västerås, Sweden

Project in Embedded Systems - 7.5 ECTS

API DESIGN FINAL CLOCK

Afram Afrem

Lukas Dust

Aqeel Iqbal

Johannes Deivard

2020-11-24

Contents

1. Clock

1.1. ROSA_sysTickWait

Prototype:	void ROSA_sysTickWait(uint_32 ticksToWait);
Description:	Suspends the using Task for the given amount of system scheduler ticks. If the delay is 0, it forces a rescheduling.
Parameters:	uint_32 ticksToWait - Amount of Ticks which the task should be suspended
Return:	void
Motivation:	Required by the customer, name is mentioned in the ROSA_timerInterrupt.pdf document.

1.2. ROSA_sysTickWaitUntil

Prototype:	void ROSA_sysTickWaitUntil(uint_32 *previousWakeTime, uint_32 timeIncrement);
Description:	Suspends the using Task until a certain point in time is reached (previousWakeTime+timeIncrement).
Parameters:	uint_32 *previousWakeTime - Pointer to the variable, which holds the time when the task last woke up uint_32 timeIncrement - Absolute time in ticks till when the task should be awoken
Return:	void
Motivation:	Required by the customer

1.3. ROSA_getTickCount

Prototype:	uint_32 ROSA_getTickCount();
Description:	Returns the value of the tick count which keeps track of the system scheduler tick count.
Parameters:	NONE
Return:	uint_32 - Actual value of the system scheduler tick count
Motivation:	This function is not directly mentioned by the customer, but it can help the user to have the opportunity to get the actual system time. E.g. in the situation for using it to define the end of the absolute delay.