Freescale MQX RTOS Example Guide - Test Example

Introduction

MQX provides core run-time testing that tests the integrity of most MQX components. A test determines, whether the data that is associated with the component is valid and not corrupted. MQX considers the data in a structure valid, if the structure's VALID field is a known value. MQX considers data in a structure corrupted, if its CHECKSUM field is incorrect or pointers are incorrect. An application can use run-time testing during its normal operation.

Explaining the example

An application can use run-time testing during its normal operation. This capability is illustrated on Test example code. It calls test function for each MQX component. Used functions are listed in Table 1.

Application contains only one task called background_test_task() and all tests are called from it.

No.	Test function name	Service
1	_event_test	Events
2	_log_test	Logs
3	_lwevent_test	Lightweight events
4	_lwlog_test	Lightweight logs
5	_lwsem_test	Lightweight memory with variable-size blocks
6	_lwmem_test	Lightweight semaphores
7	_lwtimer_test	Lightweight timers
8	_mem_test_all	Memory with variable-size blocks (need disabled MQX_USE_LWMEM_ALLOCATOR)
9	_msg_create_component	Message creation
10	_msgpool_test	Message pools
11	_msgq_test	Message queues
12	_mutex_test	Mutexes
13	_name_test	Name component
14	_partition_test	Memory with fixed-size blocks (partitions)
15	_sem_test	Semaphores
16	_taskq_test	Task queues
17	_timer_test	Timers
18	_watchdog_test	Watchdogs

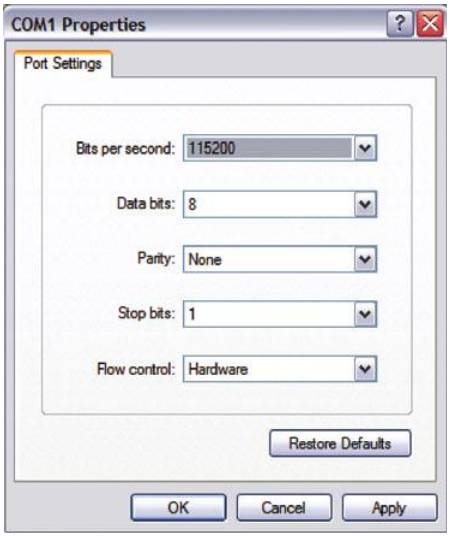
Table 1: Table of services under test.

If it finds an error, it stops the application (_task_block()) and report message will be printed to terminal via serial bus. Each report message contains error code. Use Reference manual (MQXRM.pdf from doc folder in MQX root directory) for error codes decoding. Look for test function name. Numeric values of error codes can be obtained from MQX_root\mqx\source\include\mqx.h.

Running the example

Start HyperTerminal on the PC (Start menu->Programs->Accessories->Communications). Make a connection to the serial port that is connected to the board (usually will be COM1).

Set it for 115200 baud, no parity, 8 bits and click OK.



Reset board and read result.