TB3169

Enabling Various RTOS Support with MPLAB® Harmony

INTRODUCTION

In embedded applications, there are situations where the application needs to stop its current activity and start another task or respond to an external event. In an environment without an operating system, the only way to achieve this is by using interrupts. Interrupts allow application code preemption.

In applications that are developed using the MPLAB® Harmony software framework, the various layers like application, middleware, and driver run in a cooperative manner by breaking down each task into small units of execution. However, the events are processed only when the program gets the chance to run a task. As a result, the cycle time to process the events may be non-deterministic and may vary as functions are added, removed, or changed.

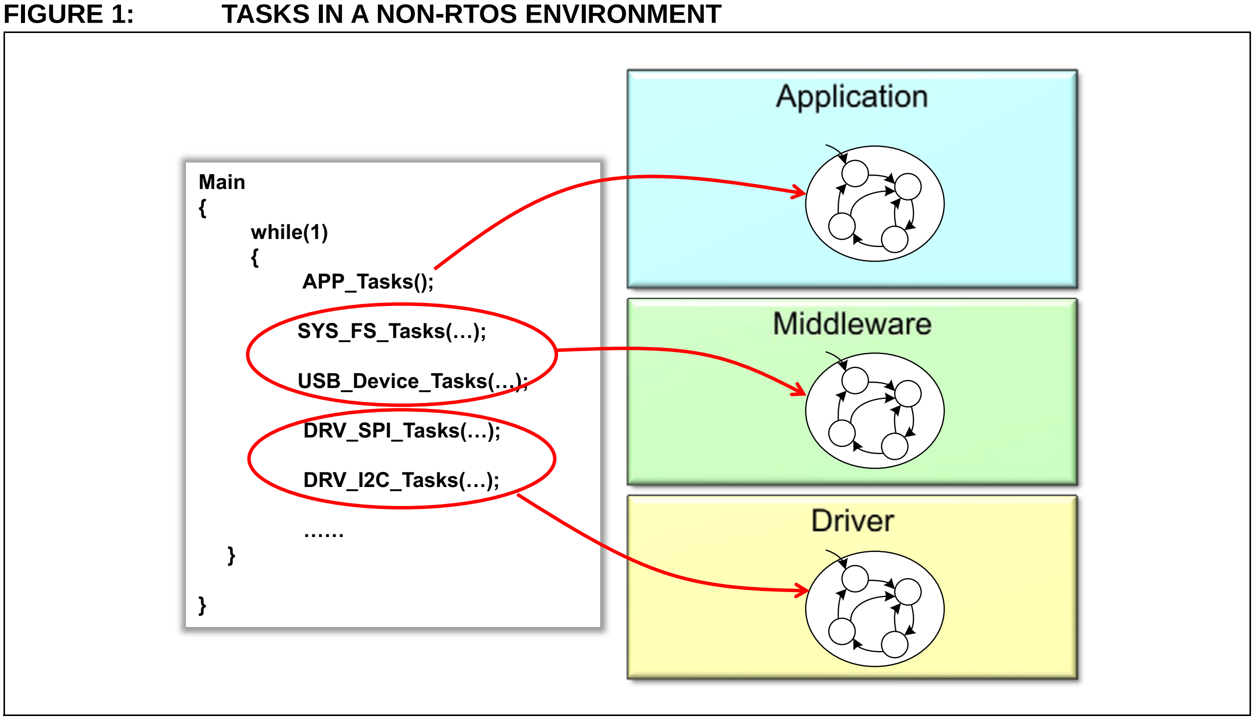
Real-Time Operating Systems (RTOS) can preempt a task and allow other high-priority tasks to execute. In addition, the RTOS scheduler ensures that the tasks waiting for the resources or events do not waste the CPU time. Such tasks move to a blocked state, and the tasks that are ready to run get the CPU time. This results in a more effective utilization of the CPU bandwidth.

CONCEPT

In an RTOS environment, instead of running all the tasks from a big super loop, each individual task can be run in a dedicated loop. This allows the individual task to be assigned a priority, and therefore, provides for a better responsiveness by assigning a high priority to tasks that are time-critical (hard real-time) in nature. Figure 1 shows how tasks in a MPLAB Harmony-based application are run from a super loop in a non-RTOS environment.

Figure 2 shows how each task can be run in an individual thread when the MPLAB Harmony application is configured to run in a RTOS environment. Usually, the driver task routines may still be run from an interrupt context. Interrupts usually support the best real-time response latency and hardware usually provides mechanisms for setting interrupt priorities.

FIGURE 1: TASKS IN A NON-RTOS ENVIRONMENT



RTOS SUPPORT IN MPLAB HARMONY

Microchip offers solutions from industry-leading RTOS specialist vendors. These solutions are compatible with the MPLAB Harmony framework and integrate with MPLAB Harmony Configurator (MHC).

MPLAB Harmony supports the following RTOS options. In addition, support for other RTOS (even custom developed ones) can be easily added.

- FreeRTOS™
- OPENRTOS®
- SEGGER embOS®
- Micriµm® µC/OS-II™
- Micriµm® µC/OS-III™
- Express Logic ThreadX

MHC allows for enabling and configuration of each individual RTOS. Figure 3 shows the selection of RTOS in MHC and the configuration of various parameters of the selected RTOS. Once a RTOS is selected and configured, the application task, the driver tasks running in polled mode, middleware libraries, and system tasks are automatically configured by MHC to run with the selected RTOS. These configuration parameters include the stack size for each thread, priority of the thread, and the task delay as shown in Figure 4. Setting the task delay allows a thread to be put into a blocked state, thereby allowing other threads in the system to run. Users can modify these parameters using MHC.

Note:	For FreeRTOS, details of the RTOS configuration options can be located at: http://www.freertos.org/a00110.html Similarly, the details of the configuration options for other RTOS can be found in the respective RTOS documentation.

FIGURE 2: TASKS IN A RTOS ENVIRONMENT

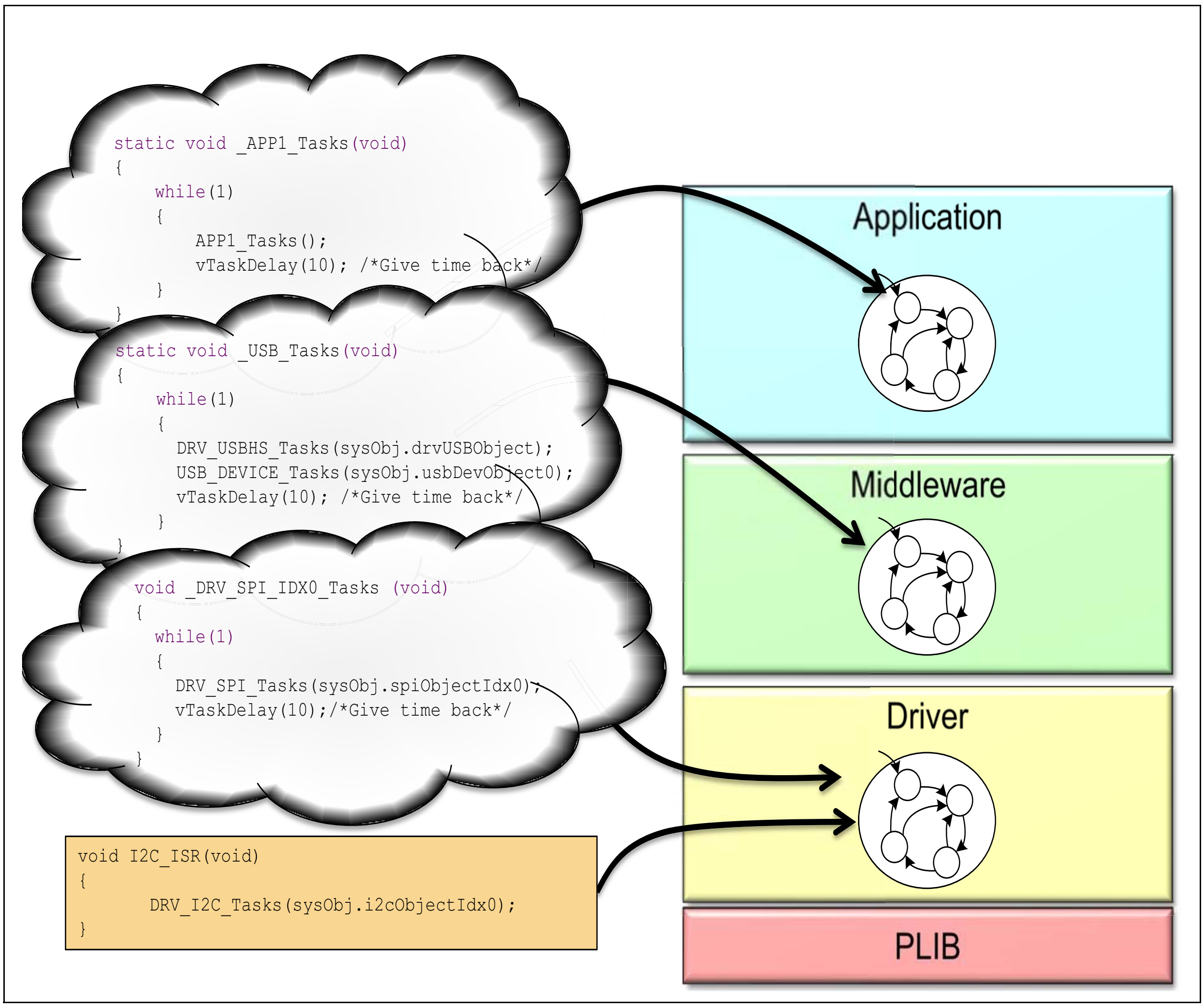


FIGURE 3: ENABLE AND CONFIGURE RTOS IN MHC

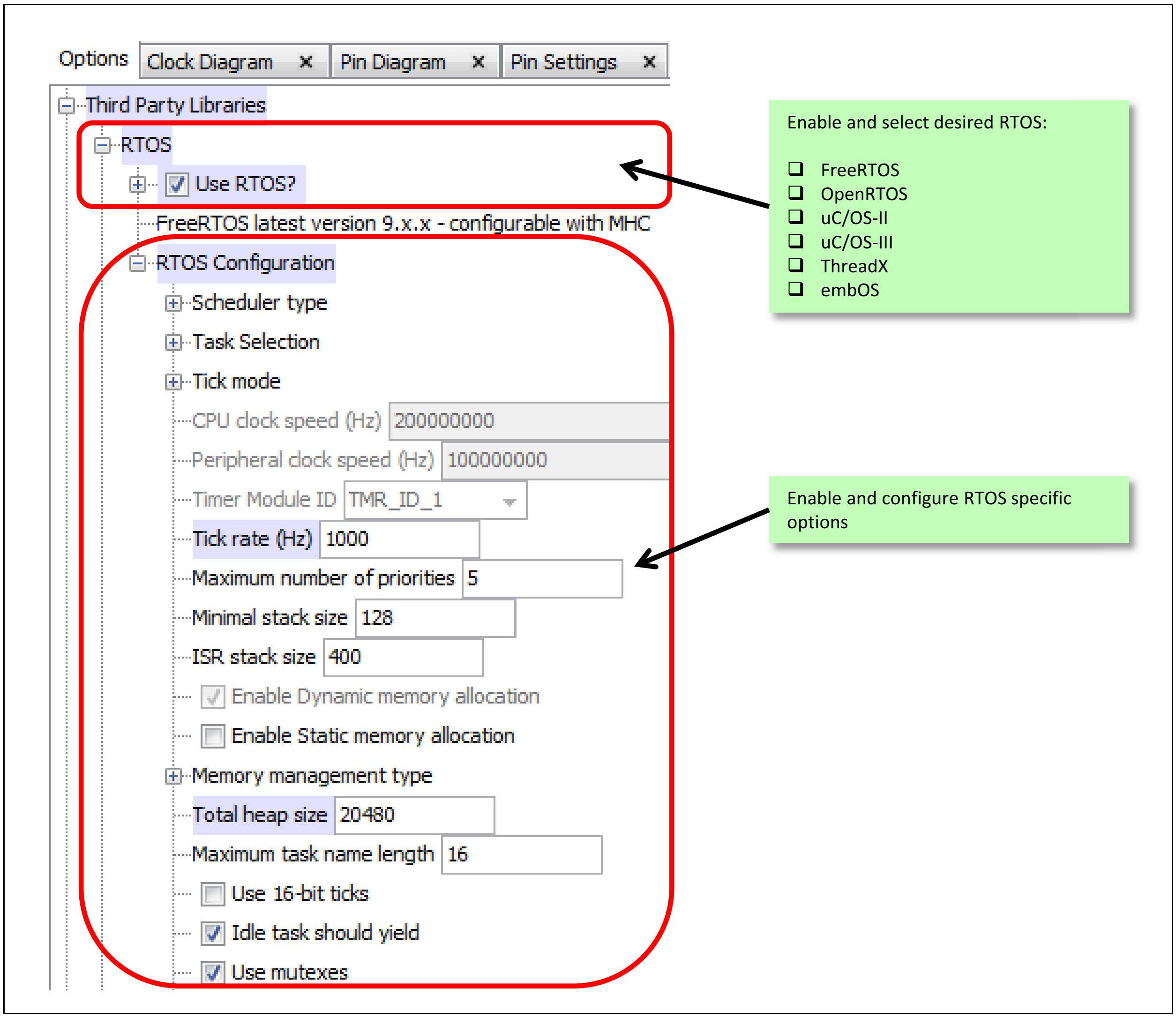
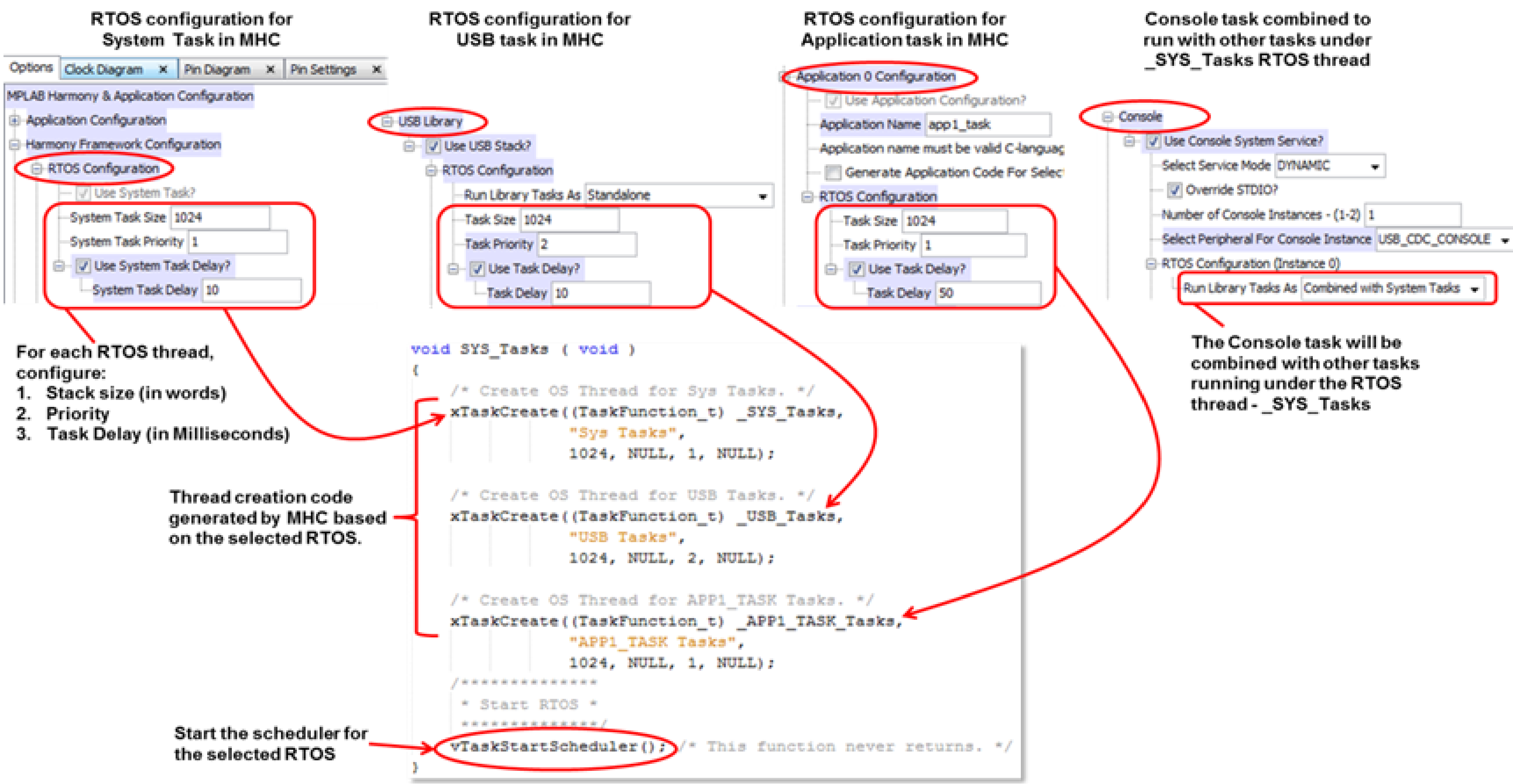


FIGURE 4: FreeRTOS CONFIGURATION EXAMPLE FOR THE MPLAB HARMONY LIBRARIES AND APPLICATION TASKS



MHC allows combining MPLAB Harmony library tasks under the common RTOS thread (_SYS_Tasks) that runs the MPLAB Harmony System modules. Figure 4 shows the MPLAB Harmony Console System Service task combined to run under the common RTOS thread. When MHC generates the code, the MPLAB Harmony System task (SYS_Tasks) routine contains the RTOS specific thread creation code. The System task routine then calls the RTOS scheduler, thereafter all RTOS threads will be managed by the scheduler and the SYS_Tasks function will not return. MHC also generates the individual daemon threads that will run the MPLAB Harmony libraries and the application tasks as shown in Figure 5.

The application remains unchanged, regardless of whether the MPLAB Harmony libraries are configured to work with or without RTOS. The modular design of the MPLAB Harmony libraries ensures that the interface provided by these libraries remains same, regardless of whether the library module is working in an RTOS or non-RTOS environment. Also, the MPLAB Harmony libraries are thread safe as the shared resources are protected against asynchronous access by multiple contesting threads or interrupts.

FIGURE 5: INDIVIDUAL RTOS THREADS CREATED BY MHC THAT RUN THE MPLAB HARMONY LIBRARY AND APPLICATION TASKS

Thread 1 – System Task

```
static void _SYS_Tasks ( void)
{
    while(1)
    {
        /* Maintain system services */
        SYS_CONSOLE_Tasks(sysObj.sysConsole0);

        /* Task Delay */
        vTaskDelay(10 / portTICK_PERIOD_MS);
    }
}
```

Thread 2 – USB Task

```
void _USB_Tasks(void)
{
    while(1)
    {
        /* USBHS Driver Task Routine */
        DRV_USBHS_Tasks(sysObj.drvUSBObject);

        /* USB Device layer tasks routine */
        USB_DEVICE_Tasks(sysObj.usbDevObject0);

        vTaskDelay(10 / portTICK_PERIOD_MS);
    }
}
```

Thread 3 – App1 Task

```
static void _APP1_TASK_Tasks(void)
{
    while(1)
    {
        APP1_TASK_Tasks();
        vTaskDelay(50 / portTICK_PERIOD_MS);
    }
}
```

Tip1: FreeRTOS

For FreeRTOS the Timer1 peripheral is used for generating the kernel ticks. If the Timer1 is already in use by another MPLAB Harmony module, MHC will indicate an error when the RTOS is enabled. To solve this error, the user must configure the MPLAB Harmony module to use a timer peripheral other than Timer1.

Tip2: Combining Library Tasks Under a Common RTOS Thread

Running each library from an individual thread allows the user to better prioritize the tasks of different libraries. However, each thread takes resources and has the task-switching overhead, using CPU bandwidth. Therefore, it often makes sense to combine several simple libraries into a common thread while leaving major libraries, such as USB, TCP/IP, etc., in their own thread.

Tip3: Task Delay

It is not mandatory to configure the Task Delay parameter for the RTOS thread. The Task Delay is only needed to yield the CPU to another thread. If the RTOS thread is designed so that it does not wait on the resources or events, and always uses the resource sharing and inter-task communication methods provided by the RTOS, such as semaphores, mutex, queues etc., the thread will be put in a blocked state until such resources or events are available. This then enables the scheduler to run other threads that are ready to run, and hence does not require using Task Delay to explicitly yield the CPU to other threads in the system.

Tip4: Stack Size

Determining the stack size required by a RTOS thread is important to ensure system stability. However, it is not easy to determine the stack requirements of a RTOS thread. This depends on several factors, such as the number of local variables used, number of nested function calls, space required to save the thread context, and stack required by the nested interrupts if a dedicated interrupt stack is not used.

Some RTOS-specific tools provide several ways to detect stack overflow at run-time or determine the worst case stack used by the individual threads. Therefore, it is generally a good practice to allocate a fairly large stack for each thread in the beginning and then adjust the stack size based on the stack usage analysis.

OTHER RELEVANT RESOURCES

For additional information on RTOS and the MPLAB Harmony Integrated Software Framework, download the MPLAB Harmony framework from the Microchip Website:

http://www.microchip.com/mplab/mplab-harmony

Examples

Refer to the following location in your installation of MPLAB Harmony for examples:

`<install-dir>/apps/rtos`

RTOS Support

MPLAB HARMONY INSTALLATION:

`<install-dir>/doc/help_harmony`

MICROCHIP WIKI:

http://microchipdeveloper.com/harmony:overview-rtos

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

QUALITY MANAGEMENT SYSTEM CERTIFIED BY DNV

== ISO/TS 16949 ==

Trademarks

ISBN: 978-1-5224-2211-2

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon

Hong Kong
Tel: 852-2943-5100
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115
Fax: 86-571-8792-8116

China - Hong Kong SAR
Tel: 852-2943-5100
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-3326-8000
Fax: 86-21-3326-8021

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

ASIA/PACIFIC

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-3019-1500

Japan - Osaka
Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

Japan - Tokyo
Tel: 81-3-6880- 3770
Fax: 81-3-6880-3771

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-5778-366
Fax: 886-3-5770-955

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

France - Saint Cloud
Tel: 33-1-30-60-70-00

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-67-3636

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-7289-7561

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820