AN3451

Enhanced Write Protection for SPI EEPROM Devices

Author: *Steven Gann*
Microchip Technology Inc.

INTRODUCTION

Many EEPROM, Flash, and other memory technologies feature different forms of software write protection. Compared to typical, legacy write protection modes, Enhanced Write Protection allows users to better partition their write protection requirements across the memory.

Enhanced Write Protection mode is different and separate from the Legacy Write Protection mode, and devices featuring Enhanced Write Protection mode can be configured for Legacy Write Protect mode also. The key difference is that Microchip implements Enhanced Write Protection through Memory Partition Registers (MPRs) and associated commands for extensive customization of write protection behavior.

This application note provides a top-down overview of how MPRs work, how they are accessed and configured, how they behave in practice, and some practical examples of how MPRs can be used.

MEMORY PARTITION REGISTER ORGANIZATION AND LOGIC

Every device with Enhanced Write Protection has two or more Memory Partition Registers (MPRs), which enable the user to split the array into multiple partitions, each with separate user-defined size and write protection behavior. Each MPR contains the endpoint of the partition as well as the protection behavior, as shown in Table 1.

The two Most Significant bits of the MPR select one of the four protection behaviors as shown in Table 2. The Least Significant six bits represent the endpoint of the partition. The precision varies by device, but they always represent the Most Significant six bits of an array address. This 6-bit resolution allows a partition's endpoint to be anywhere on the device's memory array, but restricts the exact location to aligning with certain multiples of pages, usually ranging from one page to 32 pages, depending on the size of the device.

For example, the 25CS640 is a 64-Kbit (8-Kbyte) SPI EEPROM with a 32-byte page size, featuring Enhanced Write Protection with four MPRs. Being 8-Kbyte (8,192 bytes), array addresses are 13 bits long. The six endpoint bits in each MPR truncate the Least Significant seven bits, dividing the 8,192 possible addresses between 64 possible endpoints. This means partition endpoints can be set every 128 bytes, or four pages, as shown in Figure 1.

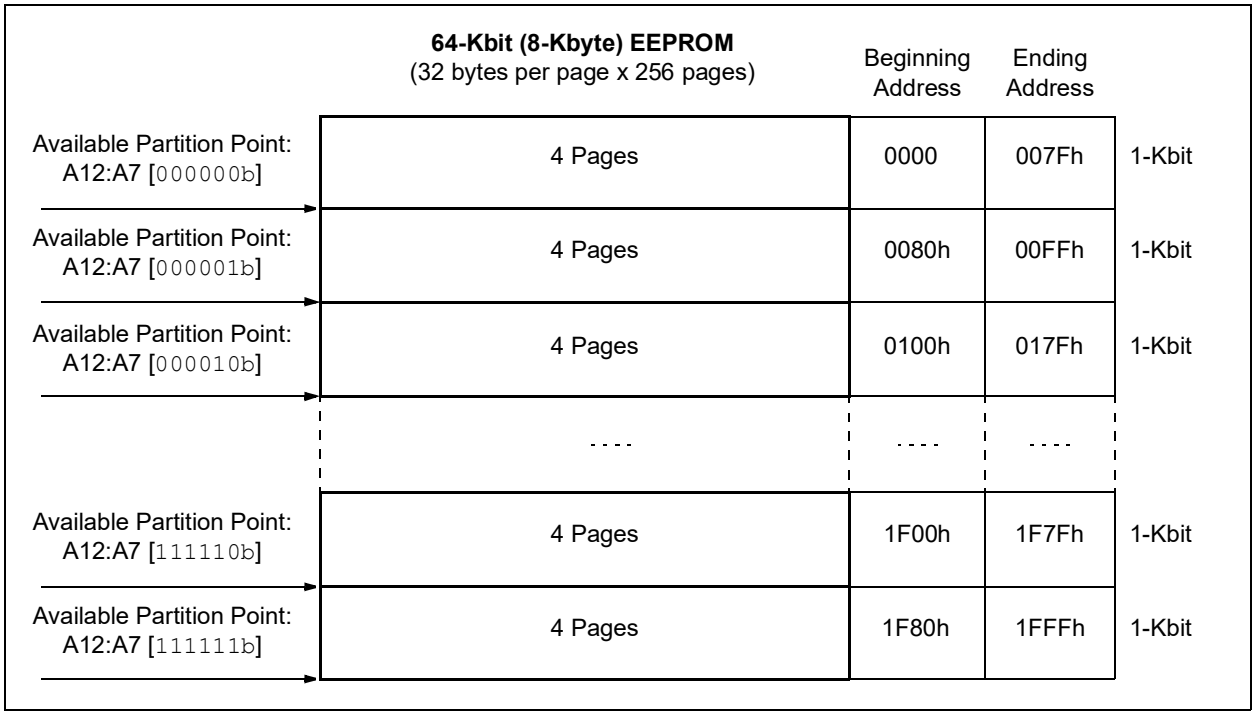
TABLE 1: 25CS640 MEMORY PARTITION REGISTER

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PB1	PB0	A12	A11	A10	A9	A8	A7
Partition Behavior		Partition Endpoint Address					

TABLE 2: PARTITION PROTECTION BITS

PB<1:0>		Behavior
0	0	Partition is open and writing is permitted (factory default).
0	1	Partition is always write-protected but can be opened at a later time (software write-protected).
1	0	Partition is write-protected only when $\overline{WP}$ pin is asserted (hardware write-protected).
1	1	Partition is software write-protected and Memory Partition register is permanently locked.

FIGURE 1: 25CS640 MEMORY PARTITION MAP



Write Protection Factory Defaults

For select SPI EEPROM devices, there are two software write protection methods: the Legacy Write Protection mode and Enhanced Write Protection mode. The default from factory is Legacy Write Protection mode.

The Enhanced Write Protection mode is not enabled by default. Specifically, the Write Protection Mode bit of the STATUS register (bit 7, byte 1) is set to logic '0' from the factory and must be changed to '1' to enable Enhanced Write Protection mode. Every MPR is also initialized to 00h, specifying that writes are allowed and the endpoint is set to the lowest available partition point, which varies based on the size of the EEPROM memory array.

MPR Overlap Behavior

Because only the endpoint of the memory partition can be specified in the MPR, the beginning of the partition is defined implicitly by the endpoint of the previous MPR, or as address 00h for MPR0. As a result, the only valid values for an MPR endpoint are the ones higher than the previous MPR. When an MPR's endpoint is set to the same or lower value than a preceding MPR, the MPR is disabled. For example, if MPR0's endpoint is set to 01h and MPR1's endpoint is set to 00h, then MPR1 is ignored and its partition behavior bits will be ignored. At the same time, if MPR2's endpoint is set to 000100b, then MPR0 will determine the write protection behavior until the end of address block 000001b, then the endpoint block value you program into MPR2 will determine the write protection behavior for the next contiguous blocks until the end of address block 000100b. The MPR3's programmed value will then be examined to determine the WP settings for the contiguous block above that.

Effectively, the MPRs follow a sequential priority scheme to handle overlaps, with MPR0 being highest priority, followed by MPR1, then MPR2, and so on. An important consequence of this behavior is that MPR0 is able to supersede all other MPRs. Regardless of how any other MPRs are configured, if MPR0 is configured to an endpoint of 3Fh it will cover the entire array and its configured Partition Behavior bits will override all other partitions.

The supremacy of MPR0 can be leveraged in practical applications for strategic protection of data. For example, in some applications it is desirable to write-protect the entire memory device and then allow writes to certain regions for purposes such as firmware updates. To accomplish this, MPR1 and above can be configured to protect or allow writes to specific regions of memory, and then MPR0 set to 7Fh. This will configure MPR0's endpoint to the last available partition point at the end of the memory array, while setting the partition behavior bits to '01' for software write protection. When protection needs to be rolled back, MPR0 can be set to 00h to restore the more complex protection scheme configured in the other MPRs.

Legacy Write Protection Mode

In Legacy Write Protection mode, the memory device can write-protect the upper quarter, upper half or all of the memory array by the STATUS register Block Protection bits (bits 3-2, byte 0). The impact of Legacy Write Protection mode on Enhanced Write Protection mode is that all Enhanced Write Protection features are completely disabled and the configuration of every MPR is ignored. See the device data sheet for a detailed description of the Legacy Write Protection mode.

When a device is in Enhanced Write Protection mode, it can also be set to Legacy Write Protection mode by changing the Write Protection Mode (WPM) bit of the STATUS register (bit 7, byte 1) to logic '0'. This makes it very convenient to alternate between two very different write protection configurations.

Note:	To prevent the Write Protection mode from being changed, the WPM bit can be permanently locked by setting the Freeze Memory Protection Configuration (FMPC) bit to '1'.

COMMON OPERATIONS

Like any feature of a memory device, interfacing with Enhance Write Protection and configuring the MPRs requires specific sequences of opcodes.

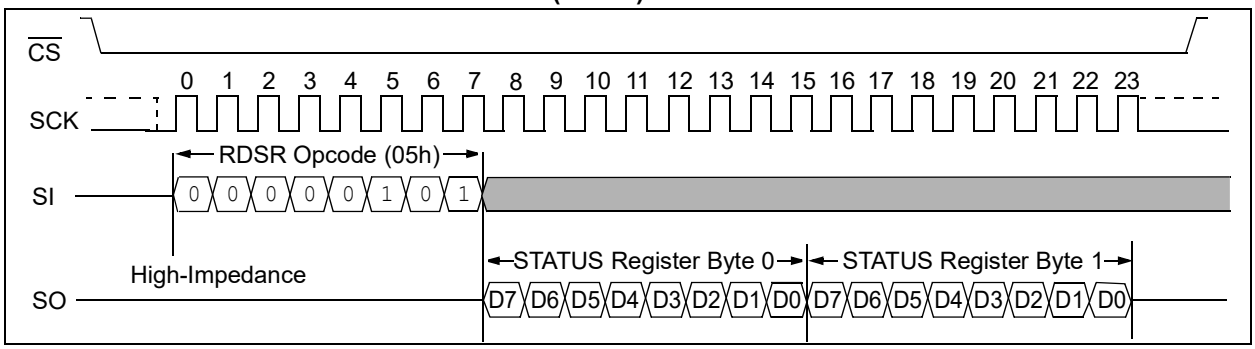
Checking If the Enhanced Write Protection Mode Is Enabled

The Enhanced Write Protection mode is configured by the WPM bit of the STATUS register, bit 7 of byte 1. When the WPM bit is set to a logic '1', the Enhanced Write Protection mode is activated, and when it is set to a logic '0', the Legacy Write Protection mode is activated.

To read the STATUS register and check the WPM bit, first pull the $\overline{CS}$ line low and send the Read STATUS Register (RDSR) opcode. Read the two bytes before driving the $\overline{CS}$ line high again to end the transaction. The two bytes read will be the contents of the STATUS register. Of these two bytes, evaluate bit 7 of byte 1 (the WPM bit) to confirm that it is set to '1' (See Figure 2).

Note:	The Legacy Write Protection mode is enabled by factory default. However, always check the current state before proceeding with the operation.

FIGURE 2: READ STATUS REGISTER (RDSR) WAVEFORM



Enabling or Disabling the Enhanced Write Protection Mode

Enabling the Enhanced Write Protection mode or switching to Legacy Write Protection mode requires setting bit 7, byte 1 of the STATUS register to either logic '1' or '0', respectively. For this, the Write STATUS Register (WRSR) command is used to write two bytes of the STATUS register, setting new values to the writable bits.

First, read the STATUS register bytes and modify bit 7 of byte 1 to the desired value. Then pull the $\overline{CS}$ line low, send the Write Enable (WREN) command (see Figure 3) to set the Write Enable Latch, pull the $\overline{CS}$ line high and then low again. Afterwards, send the WRSR opcode followed by the modified bytes before bringing $\overline{CS}$ high once more (see Figure 4). Note that some STATUS register bits are read-only and those bits are ignored in the WRSR command.

FIGURE 3: SET WRITE ENABLE LATCH (WREN) WAVEFORM

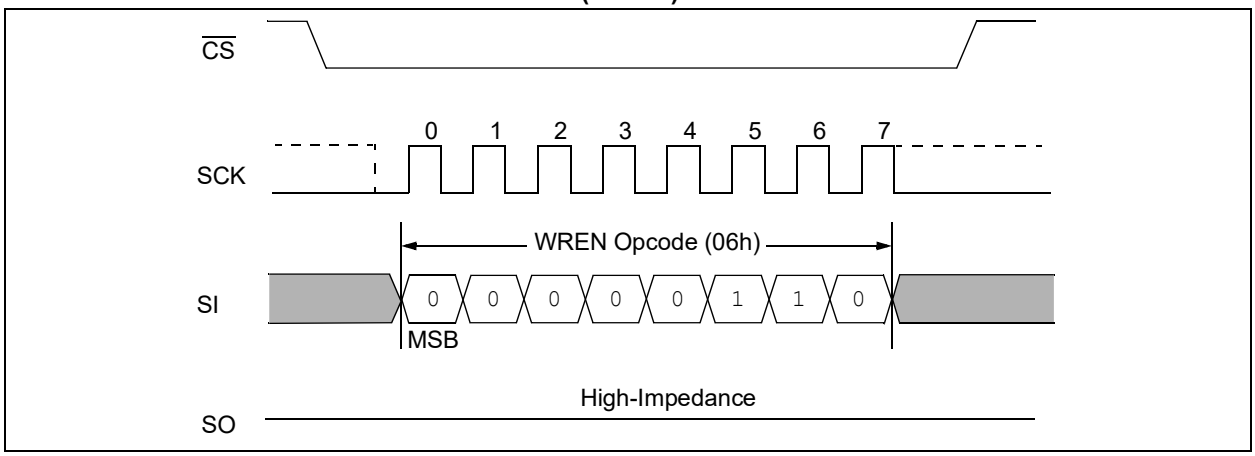
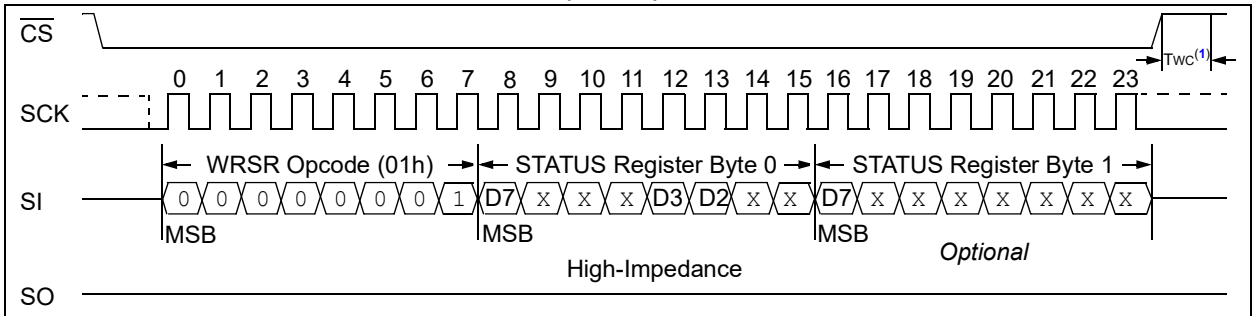


FIGURE 4: WRITE STATUS REGISTER (WRSR) WAVEFORM



Note 1: This command initiates a self-timed internal write cycle on the rising edge of $\overline{CS}$ after a valid command sequence.

Writing to an MPR

Writing to an MPR requires a specific sequence of three different opcodes. First, pull the $\overline{CS}$ line low and send the WREN opcode to set the Write Enable Latch, drive the $\overline{CS}$ line high and then pull $\overline{CS}$ low again, followed by the Partition Register Write Enable (PRWE) opcode to set the Partition Register Write Enable Latch (see Figure 5). Drive the $\overline{CS}$ line high and then low again. Then send the Write Memory Partition Registers (WMPR) opcode, followed by the number of addressing bytes your device requires for specifying the number of the MPR to write to (Address bits) and then one byte containing the desired Partition Behavior and Partition Endpoint (data bits). Finally, drive the $\overline{CS}$ line high once more to end the transaction and begin the write cycle.

Note: From all the address bits required by the `WMPR` command, only one, two or three bits are evaluated, depending on the device (see Figure 6). Refer to the device data sheet for the specific address bits used to select the MPRs.

The data bits are structured in the same way as the Memory Partition Registers, where the Most Significant two bits specify the Protection Behavior of the partition, and the remaining six Least Significant bits specify the endpoint address.

If the Freeze Memory Protection Configuration (FMPC) bit of the STATUS register (byte 1, bit 5) has been enabled (set to logic '`1`'), the `WMPR` command is ignored as the MPRs have been frozen permanently. The MPRs cannot be reconfigured after this.

FIGURE 5: PARTITION REGISTER WRITE ENABLE (PRWE) WAVEFORM

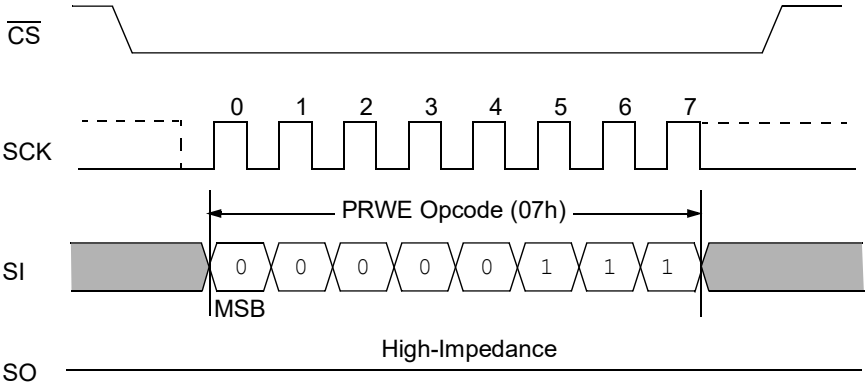
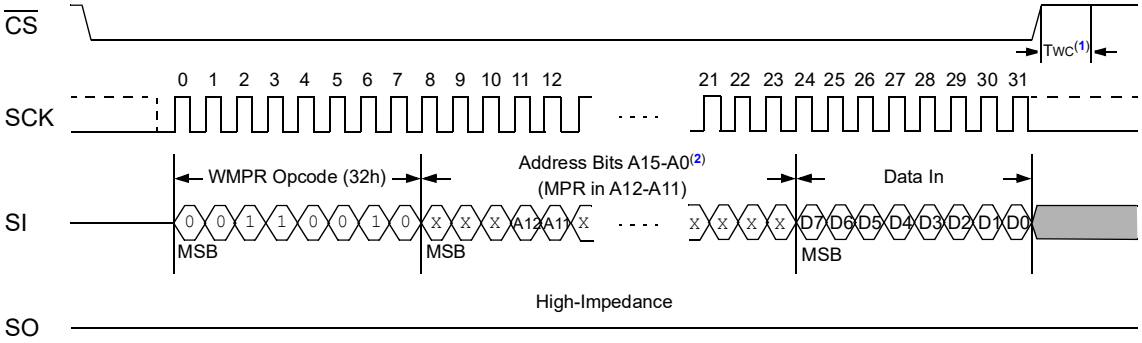


FIGURE 6: WRITE MEMORY PARTITION REGISTER (WMPR) WAVEFORM



Note 1: This command initiates a self-timed internal write cycle on the rising edge of $\overline{CS}$ after a valid command sequence.

2: This waveform is device-specific. Refer to the data sheet of each device for the specific bit locations.

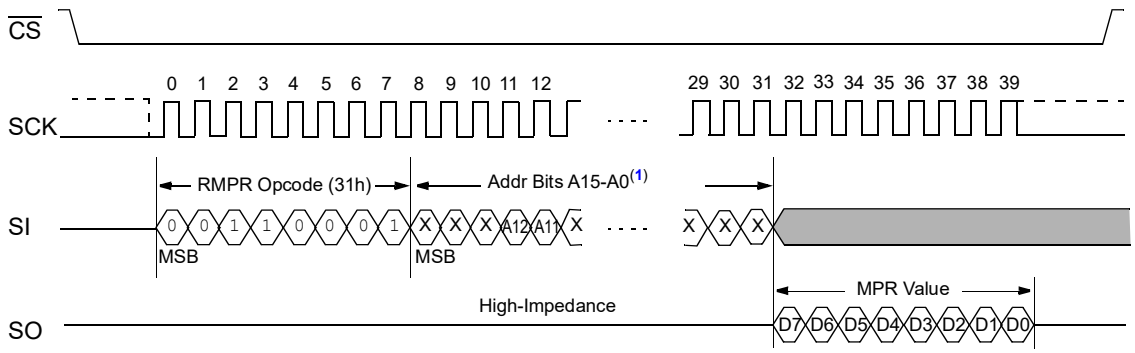
Reading an MPR

Reading an MPR is performed through the Read Memory Partition Register (`RMPR`) command. The RMPR command does not require setting any latches, and remains operational even if the FMPC bit of the STATUS register has been set.

To read an MPR, pull the $\overline{CS}$ line low and send the RMPR opcode, followed by one or more bytes specifying which MPR to read (Address bits). Read one byte, which will be the contents of the specified MPR, before driving the $\overline{CS}$ line high to end the transaction (see Figure 7).

Note: From all the address bits required by the RMPR command, only one, two or three bits are evaluated, depending on the device. Refer to the device data sheet for the specific address bits used to select the MPRs.

FIGURE 7: READ MEMORY PARTITION REGISTER (RMPR) WAVEFORM



Note 1: This waveform is device-specific. Refer to the data sheet of each device for the specific bit locations.

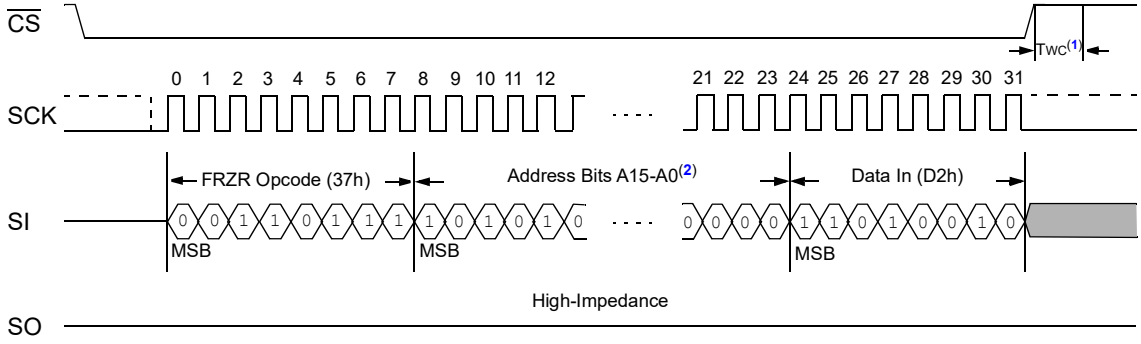
Freezing the MPRs

Once the MPRs are configured as needed, it may be useful to freeze them permanently. This is accomplished by using the Freeze Memory Protection Configuration (FRZR) command.

Note:	Once the FRZR command is issued correctly, all Memory Partition Registers on the device, as well as the WPM bit of the STATUS register, become read-only.

To freeze the state of the Memory Protection Configuration with the FRZR command, a WREN command must be issued first. Next, the CS line must be driven high and afterwards low. This is followed by issuing the PRWE command and then driving the CS line low again. Lastly, the FRZR opcode must be sent, followed by the address bits, which vary by device, and then a data byte containing D2h (see Figure 8), followed by bringing CS high once more.

FIGURE 8: FREEZE MEMORY PROTECTION CONFIGURATION (FRZR) WAVEFORM

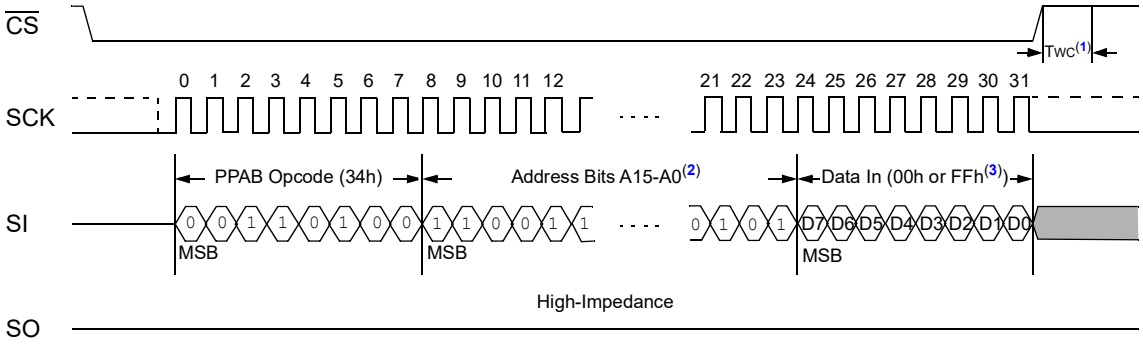


Note 1: This command initiates a self-timed internal write cycle on the rising edge of CS after a valid command sequence.
2: This waveform is device-specific. Refer to the data sheet of each device for the specific bit locations.

Locking the MPR Endpoint Addresses

For some applications, it may be useful to lock the endpoints of the MPRs, while leaving the Partition Behavior bits of each MPR writable for later configuration. This is accomplished by executing the Protect Partition Address Boundary (PPAB) command, which makes the Partition Endpoint Address bits of each MPR read-only. However, this can be reversed by using the same command. The procedure for using the PPAB command requires issuing the WREN command first, followed by the PRWE command. Lastly, the PPAB opcode must be sent, followed by the address bits which vary by device, and then a byte containing either FFh to lock or 00h to unlock the Partition Endpoint Address bits (see Figure 9).

FIGURE 9: PROTECT PARTITION ADDRESS BOUNDARIES (PPAB) WAVEFORM



Note 1: This command initiates a self-timed internal write cycle on the rising edge of $\overline{CS}$ after a valid command.

2: This waveform is device-specific. Refer to the data sheet of each device for the specific bit locations.

3: A data byte of FFh will set the PPAB bit, while a data byte of 00h will clear the PPAB bit.

APPLICATION 1: PERMANENTLY PROTECTING A SINGLE REGION AT THE START OF THE ARRAY

The simplest application of Enhanced Write Protection is to permanently protect a single region of memory, starting at 0000h and ending at some arbitrary point. Such an application is useful for purposes such as protecting manufacturer data, serial numbers, or even firmware.

For this example, a 25CS640 is used and a 3 Kb region has been chosen to be protected, from 0000h to 017Fh.

Because the region begins at 0000h, MPR0 must be used. If any other MPR is configured to cover this region, MPR0 will block that MPR's configured protection behavior and prevent it from having any effect.

To have the partition permanently protected, the two Most Significant bits are set to `11b` and the Partition Endpoint Address bits are set to `000010b`. Combining the two components results in a value of `11000010b` (C2h) for MPR0. It is assumed that the other MPRs are set to the factory default of 00h, and Enhanced Write Protection mode has been enabled. It is recommended practice to confirm this in advance, either by reading the other MPRs to confirm their state or setting them to 00h directly.

The procedure is presented in Figure 10 and the protected region is illustrated by the dashed box in Figure 11.

FIGURE 10: APPLICATION 1 FLOWCHART

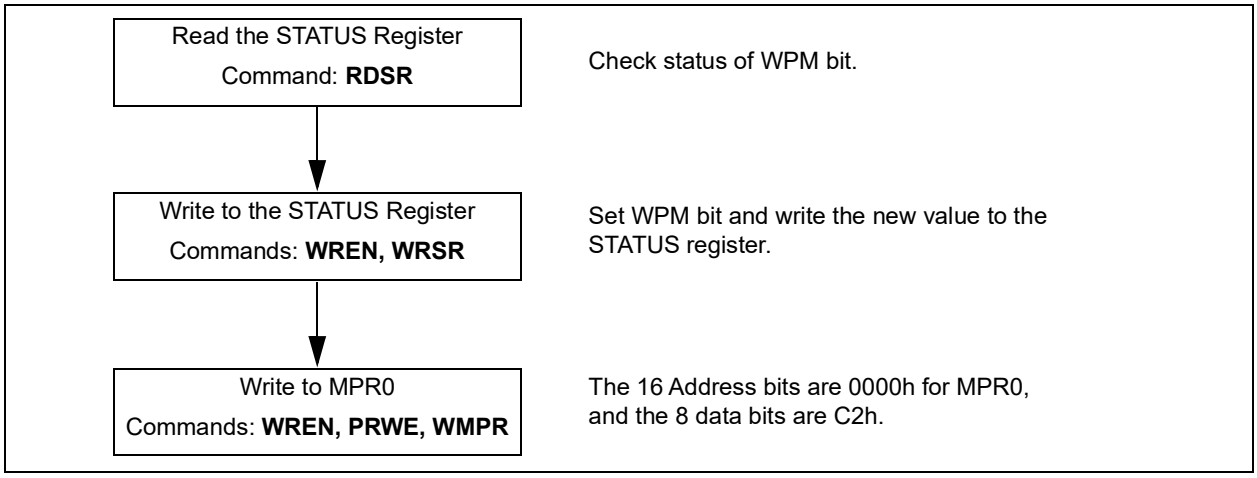
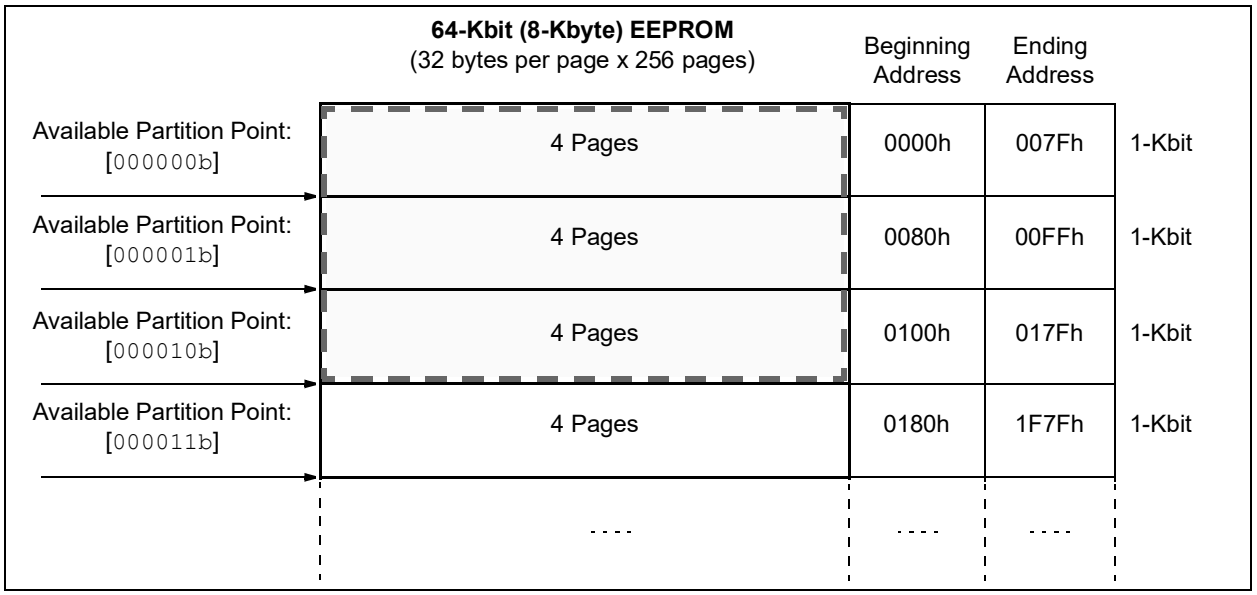


FIGURE 11: 25CS640 APPLICATION MEMORY PARTITION MAP



APPLICATION 2: PERMANENTLY PROTECTING A SINGLE REGION IN THE ARRAY

A more complex application of Enhanced Write Protection is to permanently protect a single region of memory, starting and ending at some arbitrary points. Such an application is useful for purposes such as protecting a special entry in an event log, blocking off worn-out regions of memory, etc.

For this example, a 25CS640 is used and a 1 Kb region has been chosen to be protected, from 1E80h to 1EFFh.

MPR0 always begins at 0000h. Because the chosen region does not begin at 0000h, MPR0 must be used to partition the lower portion of the array, from 0000h to 1E7Fh, while MPR1 is used to protect the region from 1E80h to 1EFFh.

For MPR0, which must be unprotected, the Partition Behavior bits are `00b` and the Partition Endpoint Address is `111100b` (F0h).

For MPR1, which must be permanently protected, the Partition Behavior bits are `11b` and the Partition Endpoint Address bits are `111101b` (F4h). Because the partition defined by MPR1 begins at MPR0's endpoint, this defines a protected partition from F0h to F4h.

Combining the components results in a value of `00111100b` for MPR0 and `11111101b` for MPR1, respectively. It is assumed that all MPR registers are initially set to the factory default 00h, and Enhanced Write Protection mode has been enabled. It is recommended practice to confirm this in advance.

The procedure is presented in Figure 12 and the protected region is illustrated by the dashed box in Figure 13.

FIGURE 12: APPLICATION 2 FLOWCHART

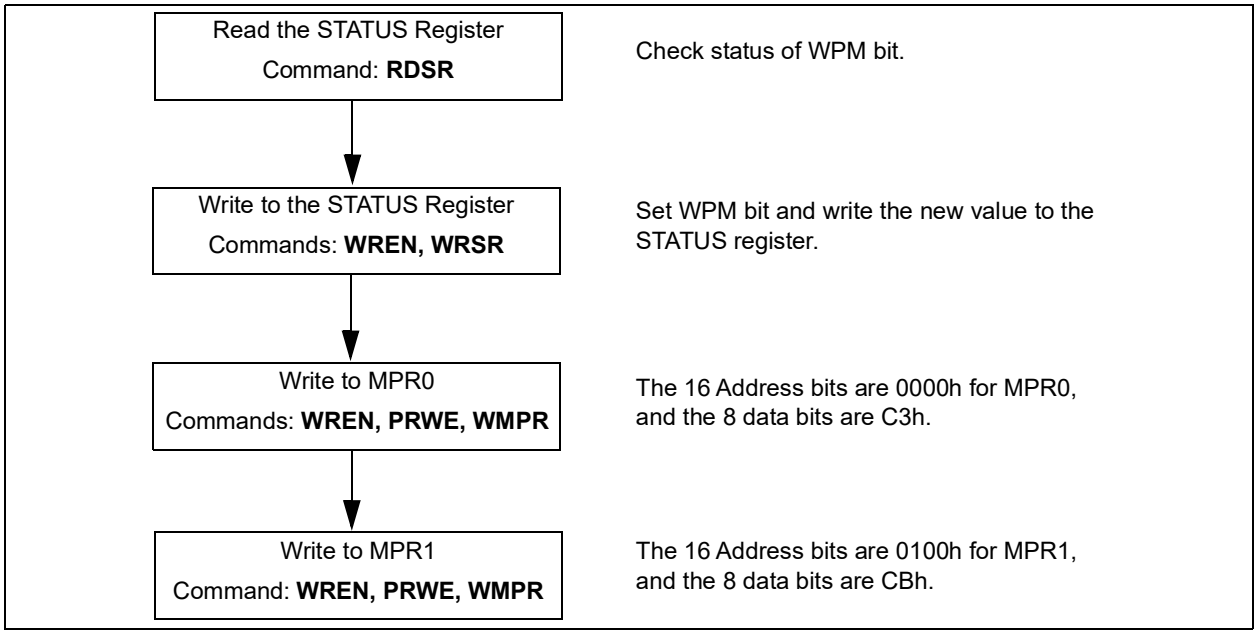
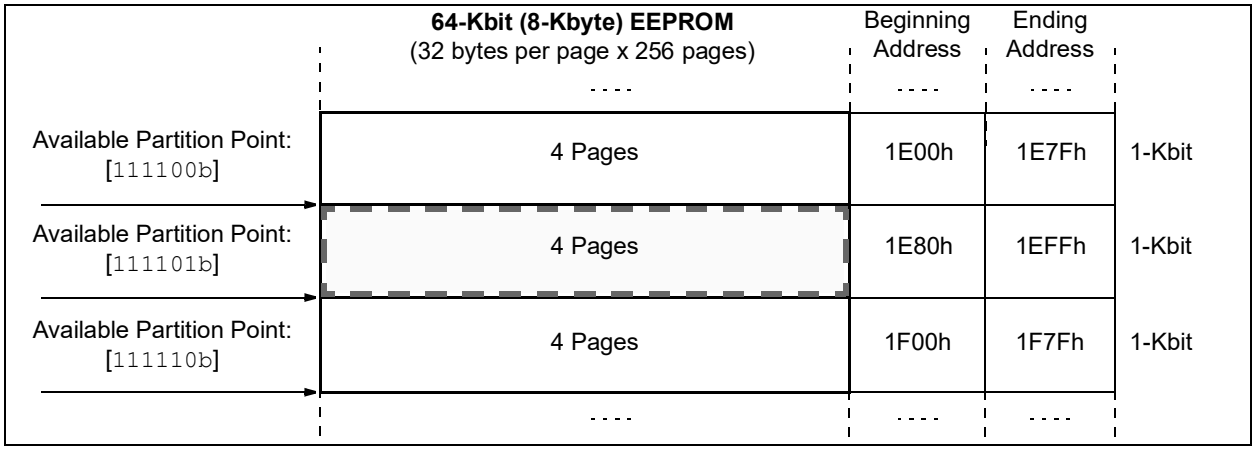


FIGURE 13: 25CS640 APPLICATION MEMORY PARTITION MAP



APPLICATION 3: PERMANENT AND SOFTWARE PROTECTED REGION

In many cases, it may be useful to permanently protect one memory region immediately (for manufacture data such as serial numbers and hardware information), permanently protect another region later (for calibration data or other data programmed after manufacturing), software protect a region for sensitive data (such as event logs) and still leave an unprotected area for storing general purpose data. This complex use case is where the Enhanced Write Protection mode demonstrates its value.

For this example, a 25CS640 is used. The memory array is divided into four regions, as described in Table 3.

The first region, controlled by MPR0, covers 0000h to 01FFh, which requires 000011 for the Partition Endpoint Address bits, and is immediately set to Permanently Protected, requiring 11b for the Partition Behavior bits. Combining the two results in the 11000011b (C3h) value for MPR0.

The second region, controlled by MPR1, covers 0200h to 05FFh, which requires 001011 for the Partition Endpoint Address bits. Initially the Partition Behavior bits are set to 00b (unprotected) prior to writing and later they change to 11b (permanently protected). Combining the two results in the needed values for MPR1: 00001011b (0Bh) initially, and then 11001011b (CBh).

The third region, controlled by MPR2, covers 0600h to 1DFFh, which requires 111011 for the Partition Endpoint Address bits and 01b (software write protection) for the Partition Behavior bits. Combining the two results in the needed value for MPR2 of 01111011b (7Bh).

The fourth region, controlled by MPR3, covers 1E00h to 1FFFh, which requires 111111 for the Endpoint Address pits, and 00b (unprotected) for the Protection Behavior bits. Combining the two results in the needed value for MPR3 of 00111111b (3Fh).

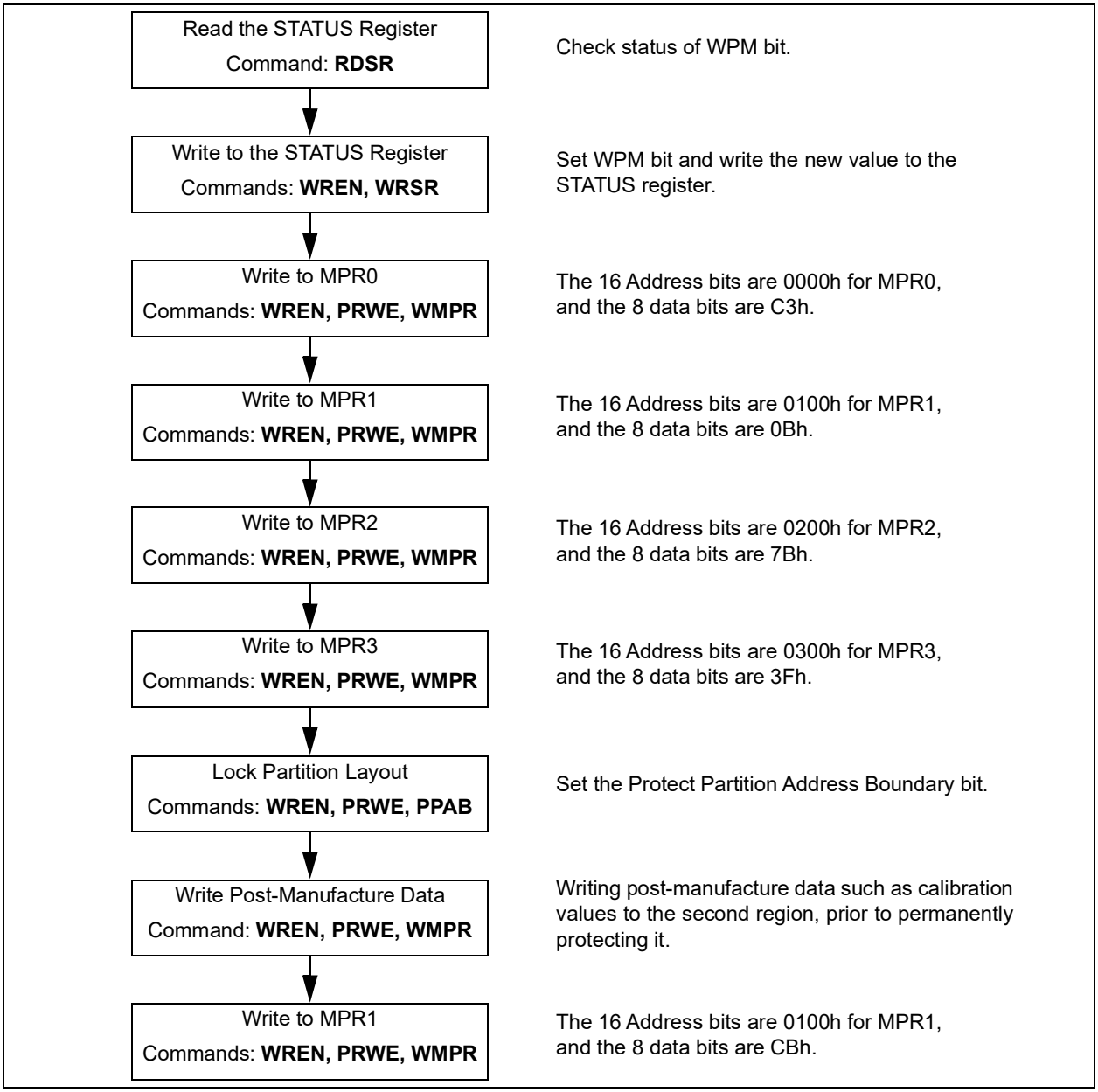
Lastly, to prevent the tampering with the partition end-points, they must be locked using the PPAB command. The PPAB command can be used to set or unset the Partition Address Bit Protection (PABP) bit of the STATUS register. When the PABP bit is set, the partition end points cannot be changed but the Protection Behavior bits of each MPR can still be changed, unless they were set to Permanently Protected.

TABLE 3: MEMORY PARTITION MAP

MPR	Start Address	Endpoint Address	Partition Protection	Partition Behavior Bits	Example Usage
MPR0	0000h (Always)	01FFh	Permanently Protected	11b	Serial number, hardware manufacture data
MPR1	0200h	05FFh	Unprotected	00b	Calibration data, post-manufacture data
MPR2	0600h	1DFFh	Software Write Protection	01b	Event logs
MPR3	1E00h	1FFFh	Unprotected	00b	General purpose

The procedure is presented in Figure 14.

FIGURE 14: APPLICATION 3 FLOWCHART



CONCLUSION

Enhanced Write Protection is a powerful feature that provides fine control over protection and writability of devices that support it. This application note presents how MPRs are structured and the logic behind them, some caveats and consequences to that logic, the low-level details of how to control and configure MPRs, and some practical examples of how MPRs might fit in an application and how to use them for those purposes.

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: 978-1-5224-5995-8

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

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

China - Beijing
Tel: 86-10-8569-7000

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

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

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

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

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

China - Hong Kong SAR
Tel: 852-2943-5100

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

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

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

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

China - Shenzhen
Tel: 86-755-8864-2200

China - Suzhou
Tel: 86-186-6233-1526

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

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

China - Xiamen
Tel: 86-592-2388138

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

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

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

India - Pune
Tel: 91-20-4121-0141

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

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

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

Korea - Seoul
Tel: 82-2-554-7200

Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

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

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

Singapore
Tel: 65-6334-8870

Taiwan - Hsin Chu
Tel: 886-3-577-8366

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

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

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

Vietnam - Ho Chi Minh
Tel: 84-28-5448-2100

EUROPE

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

Denmark - Copenhagen
Tel: 45-4485-5910
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

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-72400

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: 47-7288-4388

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