AN1312

Deviations Sorting Algorithm for CSM Applications

Author: Aurelian Floricica
Microchip Technology Inc.

INTRODUCTION

The purpose of this algorithm is to create the means of developing capacitive sensing applications in systems affected by conducted noise. This document describes the ideas and concepts behind this algorithm, as well as the firmware implementation and tuning for PIC® mid-range microcontrollers with the Capacitive Sensing Module (CSM). Please refer also to AN1171, "*How to use the Capacitive Sensing Module (CSM)*" (DS01171).

DESCRIPTION

This algorithm is based on three distinct concepts.

The First Concept – Frequency Deviation

The first concept is that the system's frequency of oscillation can be disturbed by two factors, noise and finger presses, in three different scenarios:

A) Only a finger press affects the system (lack of noise)

B) Only noise affects the system

C) A composed action of both noise and finger press affect the system

First, let's define the term "deviation" as a modification of the oscillation frequency away from its natural value.

In scenario A, due to the lack of noise, the frequency deviation will tend to be constant and easy to detect.

In cases B and C, the system's frequency of oscillation can be overpowered by the frequency of the disturbance source. This leads to the situation where the system's new oscillation frequency has a value close or equal to the frequency of the disturbance source. This new frequency presents a deviation in regard with the natural frequency of the system, which can be anywhere from multiple orders of magnitude to zero.

The Second Concept – Most Pressed Button

The second concept states that conducted noise will affect the whole sensor pool (case B), causing deviations similar in magnitude to be present on all the sensor inputs. A pressed sensor will show a deviation of a much higher magnitude than the other sensors, with or without the noise (case A and C). Thus, in case A, the pressed sensor will likely be the only one showing deviation. In case C, all the sensors are already affected by noise and show deviation. Here, the pressed sensor will show an additional deviation. This is the largest deviation with respect to the deviations of the other sensors which are affected only by noise, and not also by a finger press. This second concept can be simplified best by the expression "searching for the most pressed button."

The Third Concept – Multiple Scan Modes

The third concept states that more than one current range needs to be employed at a certain time in order to catch deviations occurring in scenarios B and C. Naturally, a capacitive sensing system can run at different frequencies of oscillations, according to the current source that is employed.

The relationship between the current through a capacitor and the rate of change of the voltage across the capacitor is the following:

EQUATION 1:

$$I = Cdv / dt$$

In this equation, the current is inversely proportional with the time. If we modify the current, we modify the time, hence the frequency.

While deviations produced in scenario A can be detected at all times, it is not the case with those deviations that appear in the other two scenarios. Here, if the noise frequency has a value almost equal to one of the two oscillation frequencies employed by the system, the result is a minor or zero deviation in that current range. However, the deviation will show up when a different current range is used because the system runs now with a frequency which is distinct than that of the noise.

To summarize, let's say that two current ranges used alternately means that there are two sensor readings at two different natural oscillation frequencies. If the deviations can't be detected in one range, they will be detected in the other one.

Consequently, all deviations caused by disturbance sources in scenarios B and C can be detected by performing a logic OR between the results of the scans at different frequencies of oscillation.

FIRMWARE IMPLEMENTATION AND TUNING

Sampling

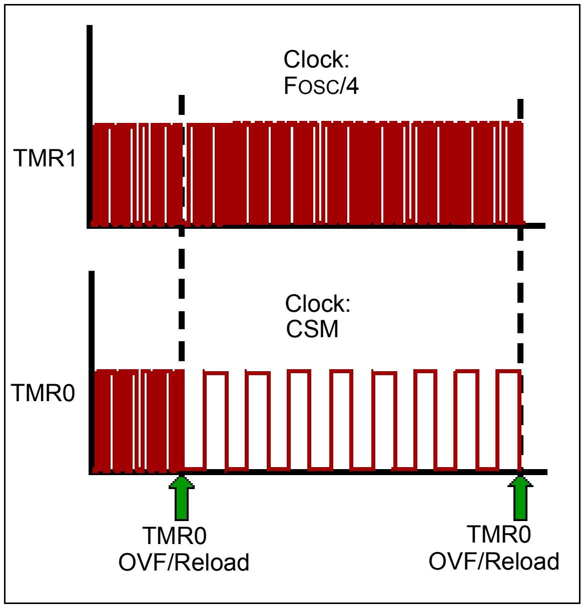
The firmware uses a dual scan mechanism for acquiring pulses from the CSM. The capacitive sensing oscillator is set to run in two different ranges: the high range and the medium range. Thus, we obtain two separate oscillation frequencies and the system is able to scan alternately for deviations using two independent firmware branches, one for each setting of the oscillator. The results of the two branches are OR'ed together to obtain the final response of the system. This is an effort to detect all possible deviation in frequency. Sampling at the same time, in parallel, for two separate values of the oscillation frequency increases the chances of detecting a deviation and thus, a possible press.

There are two methods of sampling the channels of the CSM:

- A predefined number of pulses from the CSM are acquired by one timer, while another timer is counting the time taken by this operation. This method is called "period measurement." Please refer to AN1268, "*mTouch™ Capacitive Sensing Using Period Method*" (DS01268).
- A timer is used to create a time window during which pulses from the CSM are counted by another timer. This method is called "frequency measurement."

The attached firmware is using the "period measurement" method to keep track of the oscillation frequency of the CSM channels. Figure 1 illustrates how TMR0 acts as a counter, acquiring pulses coming from CSM, while TMR1 measures the period.

FIGURE 1: PERIOD MEASUREMENT



One advantage of this method is that one can create a faster sampling rate. The timer acquiring the pulses from the CSM can be set to overflow after a specific number of pulses have been counted.

There are two labels which can be modified by the user in order to set the reload values for counter Timer0:

- `TMR0_HIGH` will be the value loaded into Timer0 each time a sampling step is performed for the high range of the capacitive sensing oscillator.
- `TMR0_MED` will be the value loaded into Timer0 each time a sampling step is performed for the medium range of the capacitive sensing oscillator.

The two reload values of Timer0 do not have to be equal, as in one range the frequency of oscillation is greater than in the other range. By defining these labels, the user sets the sampling rates in each of the two oscillator ranges: high and medium.

Storing the Deviations

For each setting/range of the oscillator, the channels are scanned consecutively. As soon as a key shows a shift in frequency (a threshold is crossed), the system runs a procedure that gathers data about the deviations of all keys (each key will have its deviation stored in a value, no matter if it is zero). The data gathering can be performed in one or more steps, each step consisting of the storage of the deviations of the keys. This procedure can be executed in a predefined number of steps. A higher number of steps leads to a higher resolution of the deviations and makes it easier to differentiate between them. However, the user should keep in mind that too many steps will slow down the system's response significantly. The label `SAMPLING_MULTIPLIER` defines the number of steps executed in this procedure.

This deviation-storing procedure is performed with the purpose of having a clear perspective about the situation of the keys deviations when one or more of them starts to show modifications in the oscillation frequency. The labels that define the threshold for the deviation in frequency are `FIX_TRIP_MED` for the medium range of the capacitive sensing oscillator, and `FIX_TRIP_HIGH` for the high range.

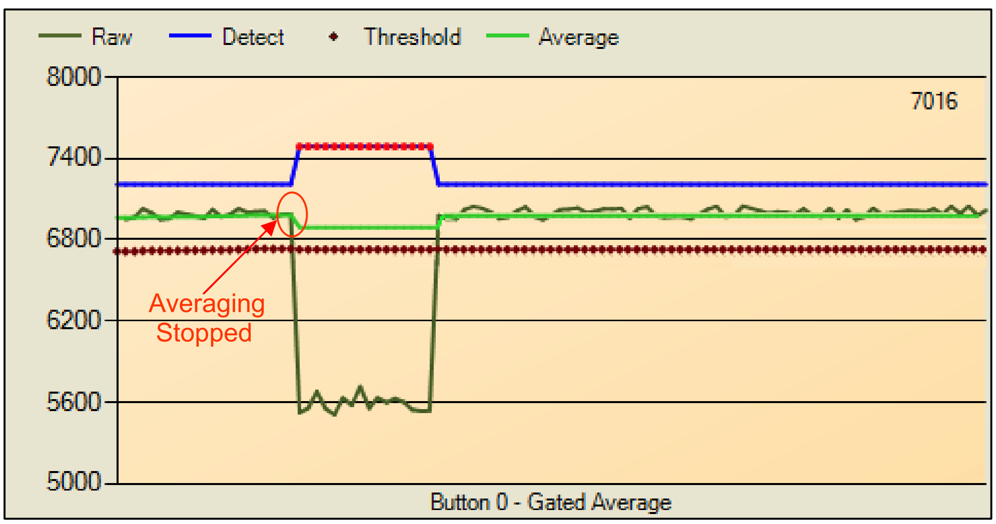
Averaging

Averaging is performed in a slow, gated manner.

- Slow: this is an effort to avoid deviations with a slow slope that might affect the keys values. For example, if a finger is slowly moving towards a key, it will likely create a small deviation. In a faster averaging system, this will typically be included in the new average values and cannot be seen. When the finger suddenly leaves the board, it creates a great amount of deviation that could be seen by the system as a false press. A slower averaging system will likely avoid this situation. The small deviation created by the slowly approaching finger will not be averaged right away and will build up into a greater deviation that will be detected as a potential press by the system. The label defining how slow the average is performed is: `AVG_DELAY`.
- Gated: due to the unpredictable nature of the noise, it makes sense not to include its potential effect in the average values. Thus, if a deviation is present when running on either of the two settings of the oscillator, then no averaging is performed.

Figure 2 shows the gated average principle.

FIGURE 2: GATED AVERAGE PRINCIPLE



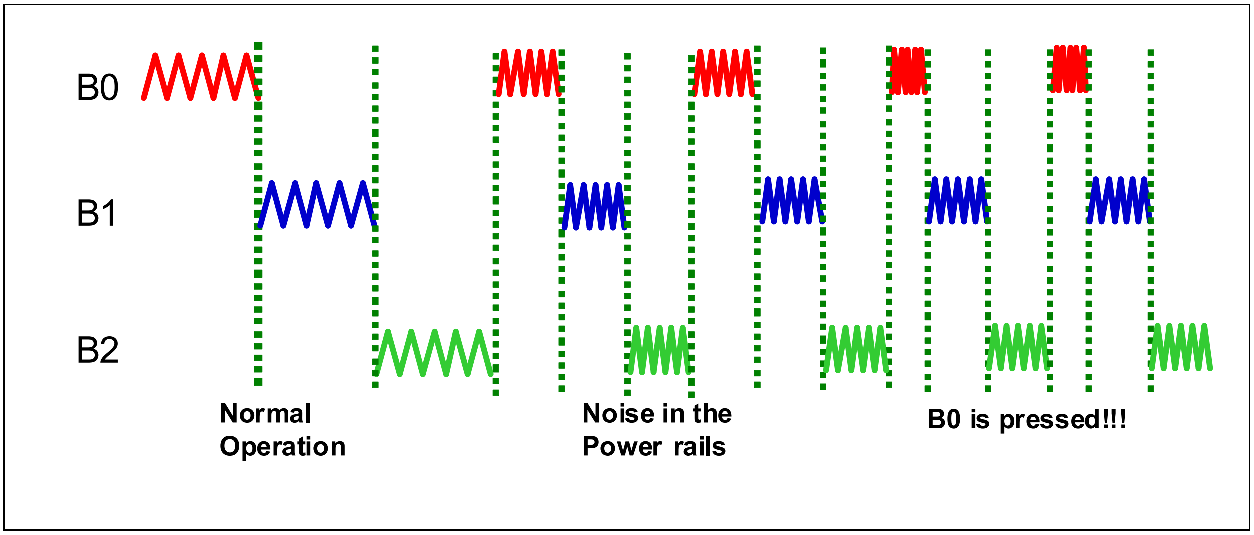
Finding the Most Pressed Button

The sorting procedure decides which one of the keys has the largest deviation in frequency, and thus, which one is going to be signaled as pressed. This deviation has to be larger than the others by a factor of 3 by default. This can be configured by the user through the `DEVIATION_MULTIPLIER` label. This label defines how many times the highest deviation must be greater than any of the other deviations and can be chosen according to the target application.

Thus, if a key exhibits a deviation that is larger than the other keys deviations by at least delta, then that specific key might be pressed.

Figure 3 illustrates how a key might be detected as the most pressed. When noise is injected in the power rails, all buttons exhibit about the same deviation in frequency (compared to the normal operation). Next, Button 0 shows additional deviation compared to the other keys, which qualifies it as a potentially pressed button.

FIGURE 3: DETECTING THE MOST PRESSED KEY



Debouncing

The debouncing routine is executed next, and there, the target is the key which exhibits the largest deviation. A debounce cycle consist of two parts: a sampling part and a sorting part. During each debounce step, all the keys are scanned and the sorting procedure searches for the key with the largest deviation. The user can define the `WINNER_DEBOUNCE_MAX` label for setting up the maximum number of debouncing cycles. Depending on the application, it might take fewer or several cycles of debouncing before a key can be signaled as pressed.

The sorting procedure is supposed to highlight the most pressed key. However, it might happen that, although some or all keys show deviations, there isn't a clear winner which meets the `DEVIATION_MULTIPLIER` criterion. Some examples illustrating this situation are: an imprecise finger touch that targets the border between two keys, more than one key pushed or noise.

If this scenario (where although keys are showing deviation, there isn't a clear winner) repeats, the firmware will clear all the keys flags. This means that the capacitive sensing system finds itself in an undefined state and doesn't have a clear response to be reported to the main application.

Signaling

To signal a key ON, there is a basic operation of setting the flag corresponding to the most pressed key. At the same time, the flags of the other keys are cleared. In this way, the system follows the rule stating that only one key can be the most pressed at a time.

The procedure for declaring a key in an OFF state is much simpler. If the deviation of a key drops below the threshold, then that specific key is run through a debouncing routine. The number of times this routine is executed before a key can be signaled as OFF is set using the `DEBOUNCE_OFF` label. The complete list and a more detailed description of the variables and labels that are used by the firmware as tuning parameters, can be found in the header file *TuningParameters.h*.

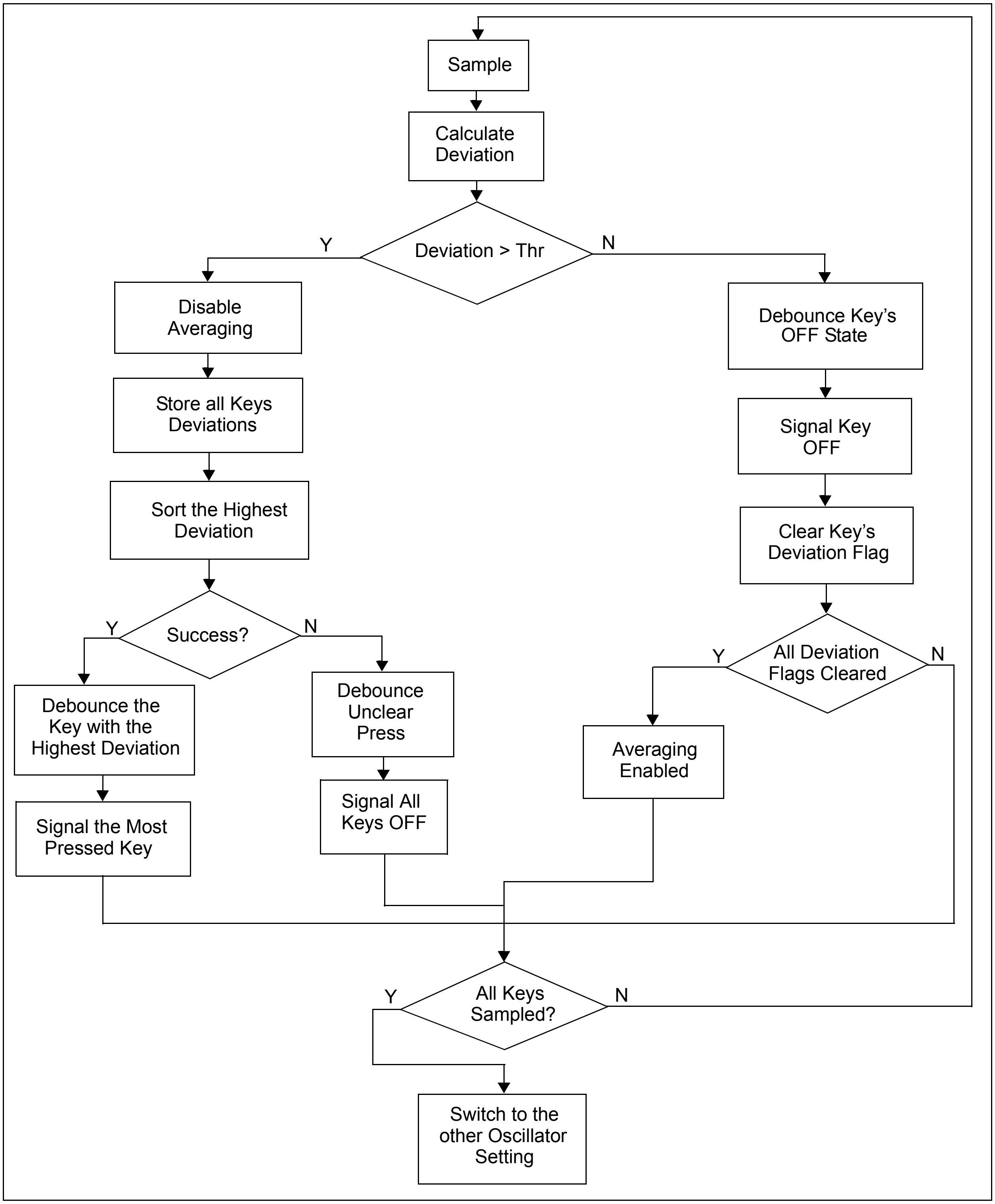
The firmware occupies about 1.6K of program memory and about 98 bytes of data memory for a 2-button version. Each increment of the number of buttons will cause the firmware to use 18 additional bytes of data memory (see Table 1 below).

TABLE 1: FLASH/RAM USAGE

	2 Buttons	3 Buttons	4 Buttons
Flash	1.6K	1.6K	1.6K
RAM	98 Bytes	116 Bytes	134 Bytes

Figure 4 is a simplified flowchart of the firmware. It contains the basic description of one of the two branches that is designated for an oscillator setting. Each branch scans all the keys before passing the control to the other one. A press/release is signaled by either one of the two branches.

FIGURE 4: FIRMWARE FLOWCHART



CONCLUSION

This algorithm is designed as a solution for touch applications operating in a noisy environment. The firmware implementation presents a high degree of customization, which allows the user to tune the firmware for a specific target application.

For more information on Microchip's mTouch™ Sensing solutions, please check our web site at www.microchip.com/mtouch.

REFERENCES

AN1101, *"Introduction to Capacitive Sensing"* (DS01101)

AN1103, *"Software Handling for Capacitive Sensing"* (DS01103)

AN1171, *"How to use the Capacitive Sensing Module (CSM)"* (DS01171)

AN1268, "*mTouch™ Capacitive Sensing Using Period Method"* (DS01268)

WEBINARS

Introduction to mTouch™ Capacitive Touch Sensing.

Capacitive mTouch™ Sensing Solutions: Design Guidelines.

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.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Trademarks

Printed on recycled paper.

ISBN: 978-1-60932-059-1

Microchip received ISO/TS-16949:2002 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.

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://support.microchip.com
Web Address:
www.microchip.com

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

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

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

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

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

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Kokomo
Kokomo, IN
Tel: 765-864-8360
Fax: 765-864-8387

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

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

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

China - Beijing
Tel: 86-10-8528-2100
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 - Hong Kong SAR
Tel: 852-2401-1200
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: 86-21-5407-5533
Fax: 86-21-5407-5066

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

China - Shenzhen
Tel: 86-755-8203-2660
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

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

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

ASIA/PACIFIC

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: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Yokohama
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

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: 886-3-6578-300
Fax: 886-3-6578-370

Taiwan - Kaohsiung
Tel: 886-7-536-4818
Fax: 886-7-536-4803

Taiwan - Taipei
Tel: 886-2-2500-6610
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

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

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

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

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

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

UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

01/05/10