TB3267

Channel Sequencing and Context Saving Using the ADC with Computation and Context Switching Module

Introduction

The Analog-to-Digital Converter with Computation and Context Switching module allows the conversion of an analog input signal into either a 10-bit or 12-bit binary representation of that signal. The ADC uses analog inputs that are multiplexed into a single Sample-and-Hold (S/H) circuit. The output of the Sample-and-Hold circuit is connected to the input of the converter. The converter generates either a 10-bit or 12-bit binary result via successive approximation, and stores the conversion result into the ADC result registers. The ADC's computation block provides hardware-driven computation features, such as averaging or low-pass filtering. The context switching block allows for up to four analog inputs to be sampled, converted, and have computation features performed automatically, without the need to reconfigure the ADC registers for each input.

This technical brief will discuss the ADC's context switching feature.

Table of Contents

1. ADC Module Overview

The ADC module consists of three main blocks:

- 10/12-bit Analog-to-Digital Converter (ADC) block
- Computation block
- Context switching block

The ADC block samples and converts an analog input into a 10/12-bit binary representation of the signal. The ADC block contains the following features:

- Acquisition timer, which allows hardware to track the acquisition time rather than creating a software delay.
- Hardware Capacitive Voltage Divider (CVD) support, which provides a precharge timer, an adjustable Sample-and-Hold capacitor array and guard rings outputs, which simplifies capacitive touch applications.
- Automatic repeat and sequencing, which provides a double conversion feature for use with CVD.

The computation block performs arithmetic operations on the ADC's conversion result. The computation block provides the following features:

- Averaging and low-pass filtering, which is performed in hardware, saving software overhead.
- Reference comparison, which allows the conversion result to be compared to a configurable reference value.
- Two-level threshold comparison, which allows the conversion result to be compared to an upper and/or lower user-configurable threshold.
- Selectable interrupts, which allows an interrupt to be triggered, depending on the results of the conversion and computation.

The context switching block allows hardware to perform conversion and computation functions on up to four analog input channels automatically. The context switching block provides the following features:

- Supports up to four unique channel configurations, with each configuration saved as a context.
- Stop on threshold interrupt during scan.
- Continuous scanning capabilities, which allows hardware to continuously scan through all enabled channels until software intervention halts scanning.

2. Context Saving and Channel Sequencing

The ADC's context switching block automates context saving and channel sequencing. The block features a channel sequencer that reduces software overhead when the ADC is using multiple input channels by storing each channel's context in memory. When the sequencer is enabled, module hardware automatically transfers the active channel's context from memory into the associated ADC registers and performs the desired conversion.

2.1 Channel Context

Channel context refers to the unique set of control, status and data register configurations that define the operation of the ADC for a specific type of conversion. For example, if the ADC is configured to read input channel ANA2 and perform for Burst-Average computations, that configuration can be saved as a channel context. Up to four channel contexts can be configured for sequencing. Context information is stored in duplicated registers located in device memory and can only be accessed through the A/D Context Selection (ADCTX) register or via Direct Memory Access (DMA).

Important: The ADC Clock Divider (ADCLK) and ADC Auto-Conversion Trigger Source Selection (ADACT) registers are not included as part of a channel context. Additionally, the ADC Control Register 0 (ADCON0) is not included as a part of a channel context, with the exception of the ADC Continuous Operation Enable (CONT) bit.

The conversion clock rate selected by ADCLK and the auto-conversion trigger source selected by ADACT are used for all contexts. For example, if Context 1 enables the Timer1 overflow as the auto-conversion trigger source, the Timer1 overflow trigger will be used for all other contexts as well. If user software configures the auto-conversion trigger to use the Timer0 overflow as the trigger source for Context 2, Context 1 will be reconfigured in hardware to also use the Timer0 overflow as the trigger source.

ADC Context Registers highlights the registers that are included as part of a channel context.

Table 2-1. ADC Context Registers

Register	Bit Pos.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCON0[(1)]	7:0	ON	CONT	CSEN	CS		FM		GO
ADCON1	7:0	PPOL	IPEN	GPOL					DSEN
ADCON2	7:0	PSIS	CRS[2:0]			ACLR	MD[2:0]		
ADCON3	7:0		CALC[2:0]			SOI	TMD[2:0]		
ADSTAT	7:0	AOV	UTHR	LTHR	MATH		STAT[2:0]		
ADREF	7:0				NREF			PREF[1:0]	
ADPCH	7:0			PCH[5:0]					
ADPRE	7:0	PRE[7:0]							
	15:8					PRE[12:8]			
ADACQ	7:0	ADACQ[7:0]							
	15:8					ADACQ[12:8]			
ADCAP	7:0				CAP[4:0]				
ADRPT	7:0	RPT[7:0]							
ADCNT	7:0	CNT[7:0]							
ADFLTR	7:0	FLTR[7:0]							
	15:8	FLTR[15:8]							
ADRES	7:0	RES[7:0]							
	15:8	RES[15:8]							
ADPREV	7:0	PREV[7:0]							
	15:8	PREV[15:8]							
ADACC	7:0	ACC[7:0]							
	15:8	ACC[15:8]							
	23:16							ACC[17:16]	
ADSTPT	7:0	STPT[7:0]							
	15:8	STPT[15:8]							
ADERR	7:0	ERR[7:0]							
	15:8	ERR[15:8]							
ADLTH	7:0	LTH[7:0]							
	15:8	LTH[15:8]							
ADUTH	7:0	UTH[7:0]							
	15:8	UTH[15:8]							

Note:

1. The ADCON0 register is not included as part of a channel context with the exception of bit 6 (CONT).

The A/D Context Selection (ADCTX) register selects the context number that will be given read/write access. The A/D Context Display Select bit (CTXSW) is used to determine the read/write status of the A/D Channel Context Selection bits (CTX), which are used to determine the channel context number.

When CTXSW is set (CTXSW = 1), the CTX bits display the context number the sequencer is currently scanning, or the context number that was active when the sequencer stopped scanning due to a context threshold interrupt.

Important: When CTXSW is set, the CTX bits are read-only and should only be read while context sequencing is enabled (CSEN = 1).

When CTXSW is clear (CTXSW = 0), the CTX bits display the context number as selected by user software. Any context can be selected by writing the CTX bits with the desired context number.

2.1.1 Context Configuration

The following steps are used to configure channel context via the ADCTX register:

1. Write the desired context number into the CTX bits; hardware automatically clears CTXSW.
2. Configure all desired ADC registers (e.g., ADCON1, ADCON2, etc.).
3. Repeat steps 1 and 2 until all channel contexts have been configured.

Remember: The context number is always one number greater than the value written into the CTX bits. For example, when the CTX bits are written with zero (CTX = 0), Context 1 is in view.

Once the context number has been written into the CTX bits, the selected channel context is available for software access. After each ADC register is configured, the register value is copied into the associated context register.

Data can also be read from the ADC's data registers (ADFLTR, ADRES, etc.) using the same steps above. Rather than configuring the desired ADC registers in step 2, user software can instead read data from the desired data registers.

It is highly recommended to verify that the STAT bits of the ADSTAT register are clear (STAT = 0) before changing any context registers. Modifying context registers during an active conversion may lead to data corruption.

2.1.1.1 Context Configuration Example

The code example below illustrates the configuration of four channel contexts.

Example 2-1. Configuring Channel Context

```
void configADCContext(void)
{
    // Configure ADC registers for Context 1
    ADCTX = 0x00;                           // Select Context 1
    ADLTHL = 0x10;                          // Lower thresh = 1000
    ADLTHH = 0x27;
    ADUTHL = 0xE8;                          // Upper thresh = 10,000
    ADUTHH = 0x03;
    ADSTPTL = 0x00;                         // Setpoint = 0
    ADSTPTH = 0x00;
    ADACCU = 0x00;
    ADRPT = 0x10;                           // Accum 16 samples
    ADPCH = 0x00;                           // PCH ANA0
    ADCON1 = 0x00;
    ADCON2 = 0x01;                          // Accumulate_mode
    ADCON3 = 0x04;
    ADSTAT = 0x00;
    ADREF = 0x00;                           // NREF VSS; PREF VDD
    ADCON0 = 0x04;                          // CONT = 0

    // Configure ADC registers for Context 2
    ADCTX = 0x01;                           // Select Context 2
    ADLTHL = 0xF4;                          // Lower thresh = 500
    ADLTHH = 0x01;
    ADUTHL = 0x88;                          // Upper thresh = 5000
    ADUTHH = 0x13;
    ADSTPTL = 0x00;                         // Setpoint = 0
    ADSTPTH = 0x00;
    ADACCU = 0x00;
    ADRPT = 0x20;                           // Accumulate 32 samples
    ADPCH = 0x01;                           // PCH ANA
    ADCON1 = 0x00;
    ADCON2 = 0x52;                          // AVG_mode; CRS = /32
    ADCON3 = 0x04;
    ADSTAT = 0x00;
    ADREF = 0x00;                           // NREF VSS; PREF VDD
    ADCON0 = 0x04;                          // CONT = 0


    // Configure ADC registers for Context 3
    ADCTX = 0x02;                           // Select Context 3
    ADLTHL = 0xF4;                          // Lower thresh = 500
    ADLTHH = 0x01;
    ADUTHL = 0x88;                          // Upper thresh = 5000
    ADUTHH = 0x13;
    ADSTPTL = 0x00;                         // Setpoint = 0
    ADSTPTH = 0x00;
    ADACCU = 0x00;
    ADRPT = 0x20;                           // Accum 32 samples
    ADPCH = 0x02;                           // PCH ANA2
    ADCON1 = 0x00;
    ADCON2 = 0x53;                          // Burst_AVG; CRS = /32
    ADCON3 = 0x04;
    ADSTAT = 0x00;
    ADREF = 0x00;                           // NREF VSS; PREF VDD
    ADCON0 = 0x04;                          // CONT = 0

    // Configure ADC registers for Context 4
    ADCTX = 0x03;                           // Select Context 4
    ADLTHL = 0x00;
    ADLTHH = 0x00;
    ADUTHL = 0x00;
    ADUTHH = 0x00;
    ADSTPTL = 0x00;
    ADSTPTH = 0x00;
    ADACCU = 0x00;
    ADRPT = 0x04;
    ADPCH = 0x03;                          // PCH ANA3
    ADCON1 = 0x00;
```

```
    ADCON2 = 0x24;                          // LPF mode; CRS = 2
    ADCON3 = 0x57;                          // Context 4 int always
    ADSTAT = 0x00;
    ADREF = 0x00;                           // NREF VSS; PREF VDD
    ADCON0 = 0x04;                          // CONT = 0

    // Apply to all context
    ADACT = 0x00;                           // No ACT
    ADCLK = 0x00;                           // ADCS FOSC/2
    ADCON0 = 0x84;                          // Enable ADC
    PIR2bits.ADCH1IF = 0;                   // Clear interrupt flags
    PIR2bits.ADCH2IF = 0;
    PIR2bits.ADCH3IF = 0;
    PIR2bits.ADCH4IF = 0;
    PIE2bits.ADCH1IE = 1;                  // Enable Thresh ints
    PIE2bits.ADCH2IE = 1;
    PIE2bits.ADCH3IE = 1;
    PIE2bits.ADCH4IE = 1;
}
```

2.1.1.2 DMA Access

The Direct Memory Access (DMA) module can be used to access the context registers as an alternative to using the ADCTX register. Each set of context registers are mirrored into an area of memory that can only be accessed through the DMA Source and Destination Address registers. These registers are flat-mapped, meaning that all of the register addresses are sequential. The sequential addressing allows the DMA to read/write a single register within a context, all the registers within a single context or all contexts. For example, user software may load the DMA Source Address register with context 2's ADRESL register address, configure the source pointer to increment after each transfer, and load the DMA Destination register with the UART's Transmit Buffer's address. In this example, the DMA will read context 2's ADRES register pair and transmit the value over the UART.

For information on configuring and using the DMA module, please refer to the applicable device data sheet.

2.2 Channel Sequencer

The channel sequencer automatically performs the operations defined by each channel context. The channel sequencer is enabled when the Context Scan Enable bit (CSEN) is set (CSEN = 1), and the scan sequence is executed when the GO bit is set by software or by the reception of an auto-conversion trigger.

The sequence includes all channel contexts that are enabled via the context's Channel Scan Enable bit (CHEN). A context channel is included in the scan sequence when the CHEN bit is set (CHEN = 1). If all context channels' CHEN bits are clear (CHEN = 0), the GO bit remains set, but no conversions or interrupts will occur.

A sequence always begins with Context 1, as long as its CHEN bit is enabled. If Context 1's CONT bit is set (CONT = 1), the sequencer will scan Context 1 and continue scanning until the threshold test invokes the ADC Channel Threshold Interrupt flag (ADCHxIF). If the ADC Stop On Interrupt bit (SOI) is set (SOI = 1) and the channel threshold interrupt occurs, the scanner will proceed to check Context 1's ADC Scan Stop On Interrupt bit (SSI) to determine whether to proceed to the next channel context. If SSI is set (SSI = 1), the scanner will clear the GO bit and the scanner will stop. If the SSI bit is clear (SSI = 0), the scanner proceeds to the next channel context that has the CHEN bit set.

If Context 1's CONT bit is set, the SOI bit is clear (SOI = 0) and the scanner will repeatedly scan Context 1 until software clears the GO bit.

Important: If a context's CONT bit is set and both the SOI and SSI bits are clear, the scanner will repeatedly scan that context indefinitely without scanning any further channels. Software must clear GO to stop the scanner.

If Context 1's CONT bit is clear (CONT = 0), the sequencer will scan Context 1. When the conversion completes, the channel threshold test is performed. If ADCH1IF is set, the sequencer checks Context 1's SSI bit to determine whether to proceed to the next channel context. If SSI is set, the sequencer clears GO and the scanner stops. If the SSI bit is clear, the sequencer will proceed to the next channel that has the CHEN bit set.

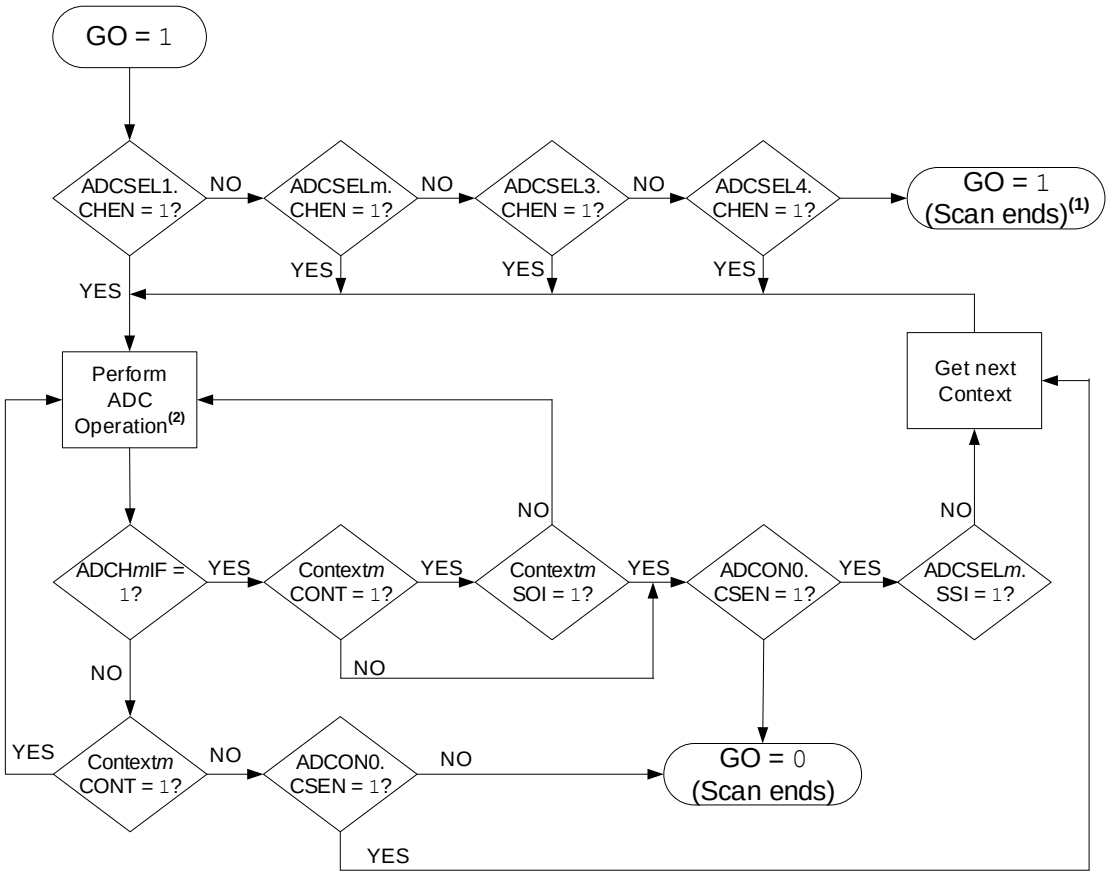
Important: When the CONT bit is clear, the SOI bit is ignored.

The scan sequence ends when any context's channel threshold interrupt occurs and that context's SSI bit is set. If the sequencer has scanned all enabled channels and no interrupt occurs, or no context's SSI bit is set, the sequencer will return to Context 1 (or the first enabled channel context) and repeat the scanning process until software clears the GO bit.

Important: The final channel in a sequence should set the Threshold Interrupt Mode Select (TMD) bits to "Interrupt regardless of the threshold test results" (TMD = 111) and set the SSI bit. These settings ensure that the scanner stops after the scan sequence has completed.

Figure 2-1 illustrates the operation of the channel sequencer.

Figure 2-1. Channel Sequencer Operation



Notes:

1. If all channels are disabled (ADCSEL[0..3].CHEN = 0), the scanner stops, but the GO bit remains set. No interrupts will occur. If software relies on a hardware clear of GO to continue code execution, the program will stall indefinitely.
2. ADC computation modes (Average, Burst-Average, etc.) are included in the 'Perform ADC Operation' block.

2.2.1 Channel Sequencer Configuration Examples

The following code examples illustrate the configuration of the channel sequencer. The first example uses software to set the GO bit and perform one complete sequence, while the second uses an auto-conversion trigger to perform sequences at a timed interval. Both examples scan Contexts 1, 2 and 4, while skipping Context 3.

Example 2-2. Software Sets GO

```
void scanOnce(void)
{
    configScan124();            // Configure sequencer
    ADCON0bits.ON = 1;          // Enable the ADC
    ADCON0bits.GO = 1;          // Software starts the sequencer
}
```

```
void configScan124(void)
{
    configADCContext();         // Configure channel contexts
    ADCSEL1.CHEN = 1;           // Enable Context 1
    ADCSEL1.SSI = 0;            // Don't stop on interrupt
    ADCSEL2.CHEN = 1;           // Enable Context 2
    ADCSEL2.SSI = 0;            // Don't stop on interrupt
    ADCSEL3.CHEN = 0;           // Context 3 not included in scan
    ADCSEL3.SSI = 0;            // Don't stop on interrupt
    ADCSEL4.CHEN = 0;           // Enable Context 4
    ADCSEL4.SSI = 1;            // Stop scan on last context

    ADCTX = 0x03;               // Select Context 4
    ADCON3bits.TMD = 0b111;     // Interrupt regardless of test results
    ADCON0bits.CSEN = 1;        // Enable the sequencer
}
```

Example 2-3. Auto-Conversion Trigger Sets GO

```
void main(void)
{
    TMR1_Initialize();          // Configure TMR1
    configScan124();            // Configure sequencer
    ADACT = TMR1_overflow;      // TMR1 is Auto-Conversion trigger
    ADCON0bits.ON = 1;          // Enable the ADC
    T1CONbits.ON = 1;           // Start TMR1
    while(1)
    {
        // wait for ADCH4IF = 1 and service ISR
    }
}
```

2.3 Tips

When using the context switching feature, there are a few things to consider:

- If a channel is configured for use in Low-Pass Filter mode, it is important to understand the effects that the RPT and CRS bits have on the filtered output. If the RPT value is high, it will take more time before the threshold interrupt occurs. If other channels are enabled for sequencing, and require conversion results to be obtained quickly, it may make sense to remove the LPF channel from the scan sequence.
- If the sequencer is configured such that it repeatedly scans all enabled channels without clearing the GO bit, only the most recent data is kept in the result registers. If data is required each time the scanner sequences through the enabled channels, the threshold interrupt should be configured to interrupt on the last enabled channel, regardless of the threshold test results. The SSI bit for that channel must also be set.

3. Conclusion

The ADC’s context feature automates context saving and channel sequencing. Up to four channels can be configured independently, with each channel’s register configuration saved as a context in program memory. The context feature saves software overhead by eliminating the need to reconfigure the ADC registers each time a new channel is selected. Once the sequencer has completed scanning the enabled channels, the results can be read either through the ADC Context Selection (ADCTX) register or the DMA.

The Microchip Website

Microchip provides online support via our website at www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to www.microchip.com/pcn and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: www.microchip.com/support

Microchip Devices Code Protection Feature

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

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods being used in attempts to breach the code protection features of the Microchip devices. We believe that these methods require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Attempts to breach these code protection features, most likely, cannot be accomplished without violating Microchip's intellectual property rights.
- Microchip is willing to work with any customer who is concerned about the integrity of its code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is "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.

Legal Notice

Trademarks

ISBN: 978-1-5224-6220-0

Quality Management System

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
Corporate Office 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 Technical Support: 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	**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	**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	**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: 47-72884388 **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