IP Protection and Sandboxing Using the PIC32CM LSx MCUs

AN5501

Introduction

The PIC32CM Lx family of microcontroller units (MCUs) provide a robust security, ultra-low power, enhanced touch and smart analog while running at 48 MHz with memory configurations of up to 512 KB Flash and 64 KB SRAM. The PIC32CM LS00 provides security features, such as Secure Boot and Arm® TrustZone® technology. The PIC32CM LS60 integrates Arm TrustZone technology with the Trust Platform ECC608 Secure element in one package.

The TrustZone technology on the PIC32CM LSx MCUs provides an effective way to safeguard sensitive source code and data. Leveraging the hardware isolation creates two distinct worlds: Secure and Non-Secure. The user's intellectual property (IP) resides in the Secure world, separate from the main application running in the Non-Secure world. This isolation prevents unauthorized access or tampering with the Secure code. In addition, TrustZone acts like a sandbox for the Secure world. Within this controlled environment, limited resources and restricted interactions with the Non-Secure world ensure the code's integrity and functionality. This combined approach of isolation and sandboxing empowers TrustZone to deliver robust confidentiality, integrity, and availability for critical code.

This document describes how the TrustZone feature on Microchip PIC32CM LSx MCUs helps achieve IP protection and sandboxing by implementing the dual-developer application use case.

Table of Contents

1. Use Case: Smartphone Payment Application

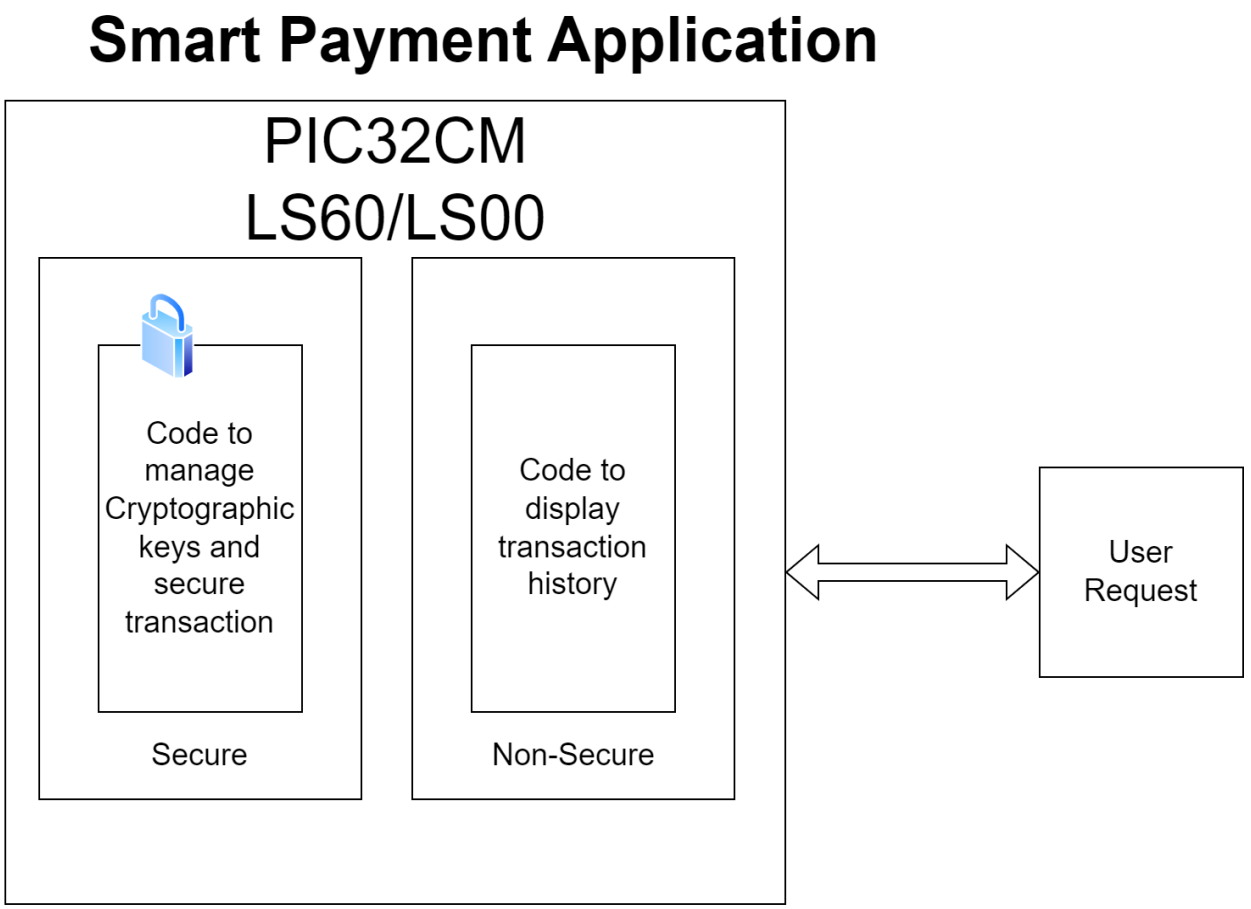
A smartphone payment application incorporates both Secure and Non-Secure entities. The organization developing this application collaborates with a third-party developer to integrate a new feature into the application while ensuring that the core payment functionality and sensitive data remain Secure and protected. The Secure application typically handles sensitive user data, such as credit card authentication tokens. Protecting this data from unauthorized access is crucial for maintaining customer trust and complying with regulations.

Secure World Development (Company/Developer A): The company's internal team works within the Secure world to develop and maintain the application's core payment functionality. This includes handling sensitive user data, managing cryptographic keys, and executing Secure transactions. The Secure world protects this critical code from unauthorized access or tampering.

Non-Secure World Development (Company/Developer B): In this world, the developer works on the non-sensitive parts of the application, such as the user interface and the logic for displaying the transaction history. Since this part of the application does not handle sensitive data directly, it does not require the same level of security as the sensitive components. Simultaneously, the third-party developer works within the Non-Secure world to develop the new feature requested by the company. This could involve implementing a new user interface, integrating additional services, or adding non-sensitive functionalities. While this work is essential for enhancing the application's capabilities, it does not require access to the Secure data or processes handled in the Secure world.

The TrustZone-based dual-developer application model of the PIC32CM LSx MCUs enables and facilitates the implementation of this use case.

Figure 1-1. Smartphone Payment Application



Company -A Firmware IP

- Fully Protected
- No readout capability
- Only necessary APIs are provided to Company B for App development

Company -B Application Code

- Full access to program and debug non-trusted zone only
- Can't access Company A Firmware IP other than exposed APIs

2. Dual-Developer Application

The dual-developer application model involves two developers. Initially, Developer A is responsible for developing the Secure application, then Developer B is responsible for developing the Non-Secure application.

A Secure application implements software security domains that restrict access to selected memory, peripherals, and I/O to trusted software without compromising the system performances. It achieves this by configuring the microcontroller's memory regions, peripherals, and I/O in a Secure mode. This contrasts with the Non-Secure application, where the memory regions, peripherals, and I/O functions do not implement any access restrictions.

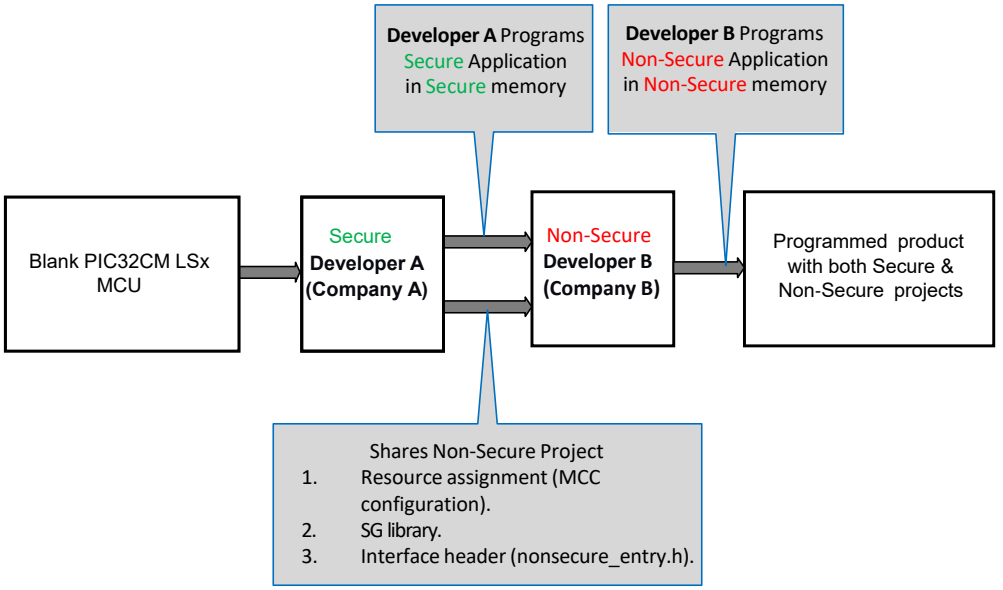
Developer A develops the Secure application and configures the Chip Erase keys in the fuse settings to prevent the Secure memory regions from being erased (or copied or modified) by the Non-Secure developer or an outsider. They also configure the Secure peripherals and Secure memory regions' footprint. Using a compiling application, the Secure Gateway (SG) library and the header file (generally `nonSecure_entry.h`) containing the Secure Gateway APIs declarations are generated. Developer A programs the Secure application onto the PIC32CM LSx Flash and sets the microcontroller to Debug Access Level 1 (DAL1) to prevent further access to the device's Secure memory region. Finally, Developer A shares the generated files, resource assignments, Secure peripherals, pins, memory attribution, and the pre-programmed PIC32CM LSx with Developer B.

Note: The SG Library is also called the Veneer library, which is generated by the Secure project and placed during the Secure project link into the Non-Secure Callable (NSC) memory region. The SG Library acts as a bridge between the Secure world and the Non-Secure worlds through the specific Secure Gateway.

Developer B starts developing a Non-Secure application on a pre-programmed PIC32CM LSx microcontroller with limited access to Secure resources. The Developer B can access the shared Secure APIs through calls to the Non-Secure callable APIs only. To do so, Developer B will use a predefined linker file (SG library) and the Non-Secure entry header file `nonSecure_entry.h` provided by Developer A. After the development is completed, the Non-Secure application is programmed into the Non-Secure memory region of the device. Therefore, the device is programmed with a TrustZone-based application.

The following diagram illustrates the dual-developer application development.

Figure 2-1. Dual-Developer Application Development



2.1 PIC32CM Lx Family of Microcontrollers

The PIC32CM Lx fmily of microcontrollers provide a robust security, ultra-low power, enhanced touch, and smart analog while running at 48 MHz with memory configurations of up to 512 KB Flash and 64 KB SRAM. These MCUs come in three variants: PIC32CM LS60, PIC32CM LS00 and PIC32CM LE00 with 48-pin, 64-pin and 100-pin variations.

- The PIC32CM LE00 is a general-purpose variant.
- The PIC32CM LS00 provides security features, such as Secure Boot and TrustZone technology.
- The PIC32CM LS60 integrates TrustZone technology with the Trust Platform ECC608 Secure element in one package and supported by Trust Platform key provisioning services and Trust Platform Design Suite v2.

2.2 Implementation Using MPLAB Harmony v3 and MCC

Note: The implementation steps shown are for the PIC32CM LS60 MCU, similar steps can be followed for the PIC32CM LS00 MCU.

The dual-developer approach involves two developers to develop the end application. The Secure application developer and the Non-Secure application developer discuss and agree on the peripherals and memory regions that will be used to create Secure and Non-Secure applications. The developers work separately and create two separate projects using the MPLAB® Harmony v3 software framework. These projects are generated using the MPLAB Code Configurator (MCC) to assign Secure and Non-Secure memory regions and peripherals. The project details are as follows:

Secure Project:

The Secure project is developed by the Developer A:

1. Configure the Secure resources and generate the Secure project using MCC.
2. Develop the Secure application and SG API declarations in the generated `nonSecure_entry.h` file.
3. Configure the Chip Erase keys for the Secure region and all memory regions in the Fuse settings.
4. Generate the SG library by building the Secure project and programming the Secure memory region.

5. Share the pre-programmed device along with the SG library and the `nonSecure_entry.h` file with a Non-Secure application developer.

Non-Secure Project:

The Non-Secure project is developed by the Developer B:

1. Add required peripherals and software components to the MCC project graph, and then generate the code.
2. Develop the Non-Secure application using the SG library and the `nonSecure_entry.h` file.
3. Build and program only the Non-Secure memory region.

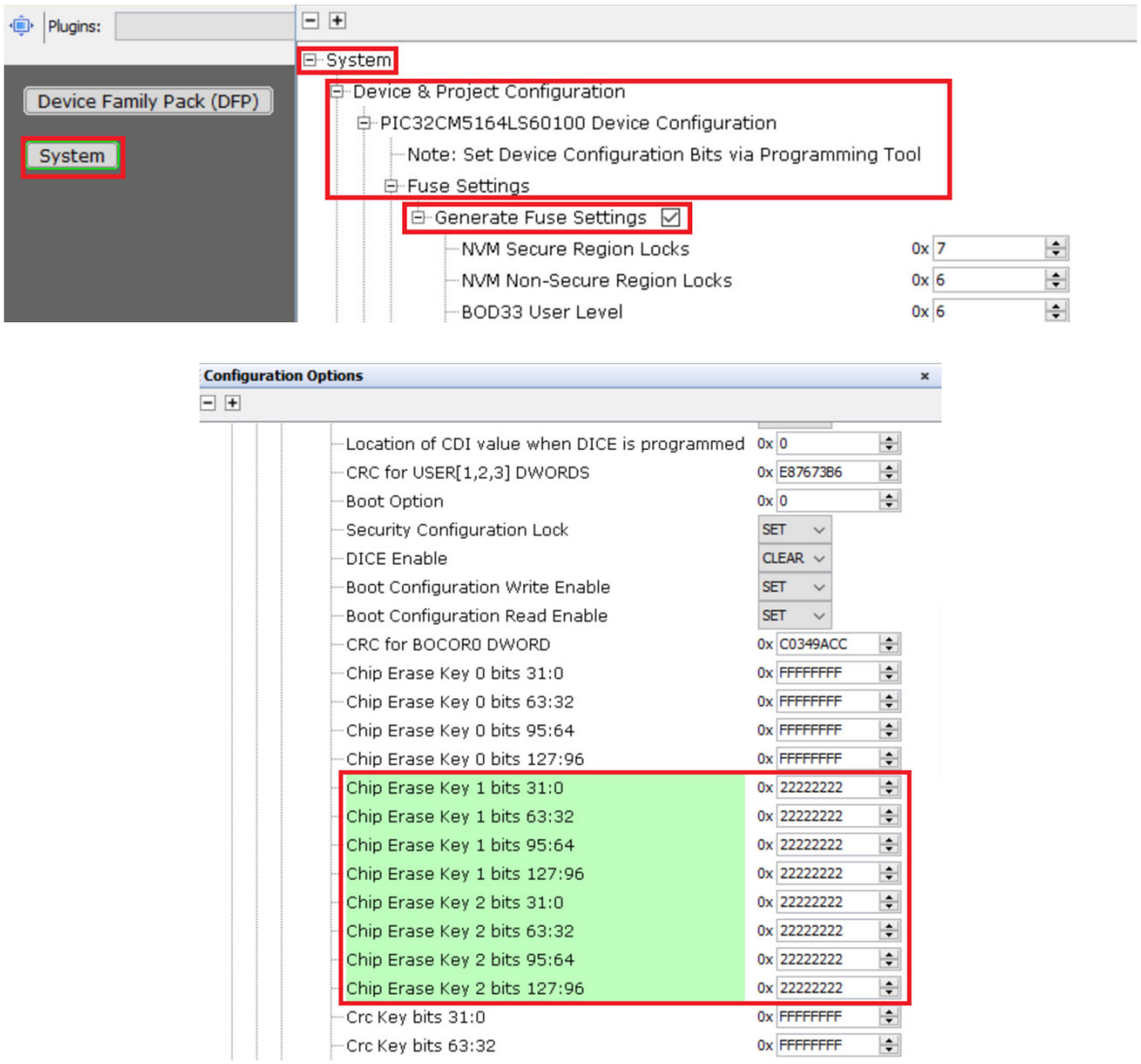
2.2.1 Secure Project Configuration

1. The Secure project development requires creating an Arm TrustZone-enabled MPLAB Harmony v3 project by configuring the peripherals and memory regions in a Secure mode. For the Secure project development, any one of the following methods can be used:
   a. To create an MPLAB Harmony v3 Arm TrustZone-based project using MPLAB X IDE from scratch, refer to the steps provided in the document: Creating the First Application on PIC32CM LSx Microcontrollers Using MPLAB Harmony v3 with MPLAB Code Configurator (MCC).

      Note: When a PIC32CM LS60 TrustZone device is selected for creating the project, MCC creates and enables configuring both the Secure and Non-Secure applications in the same project. The Secure application developer should only follow instructions relevant to the Secure application.

   b. Use the PIC32CM LS60 TrustZone Getting Started Application demonstration, developerhelp.microchip.com/xwiki/bin/view/software-tools/harmony/pic32cm-trustzone-getting-started-training-module/.
2. Go to the PIC32CM LS60 TrustZone Getting Started Secure application project path: *<DemoDeveloped or Extracted Path>**firmware**tz_pic32cm_ls60_cprogroup*.
   Note: In the current implementation, when an MPLAB Harmony v3 TrustZone project is created, MPLAB X IDE creates two projects: one for the Secure application (MPLAB X IDE project name ending with `_secure`) and the other for the Non-Secure application.
3. Open the pre-developed Secure application in MPLAB X IDE.
4. Under **Project**, right-click on the project name `tz_pic32cm_ls60_crpo_secure`, and then click **Set as Main Project.**
5. In MPLAB X IDE, go to *Tools> Embedded* and then launch MPLAB Code Configurator (MCC).
6. Under **Project Graph**, select **System** and in the right pane, click and expand *System > Device & Project Configuration > PIC32CM5164LS60100 Device Configuration > Fuse Settings*.
7. Select **Generate Fuse Settings** and configure the Chip Erase Key 1 bits 31:0 through Chip Erase Key 2 bits 127:96 with unique keys as shown in the following figure.

Figure 2-2. Configuring the Chip Erase Key in MCC

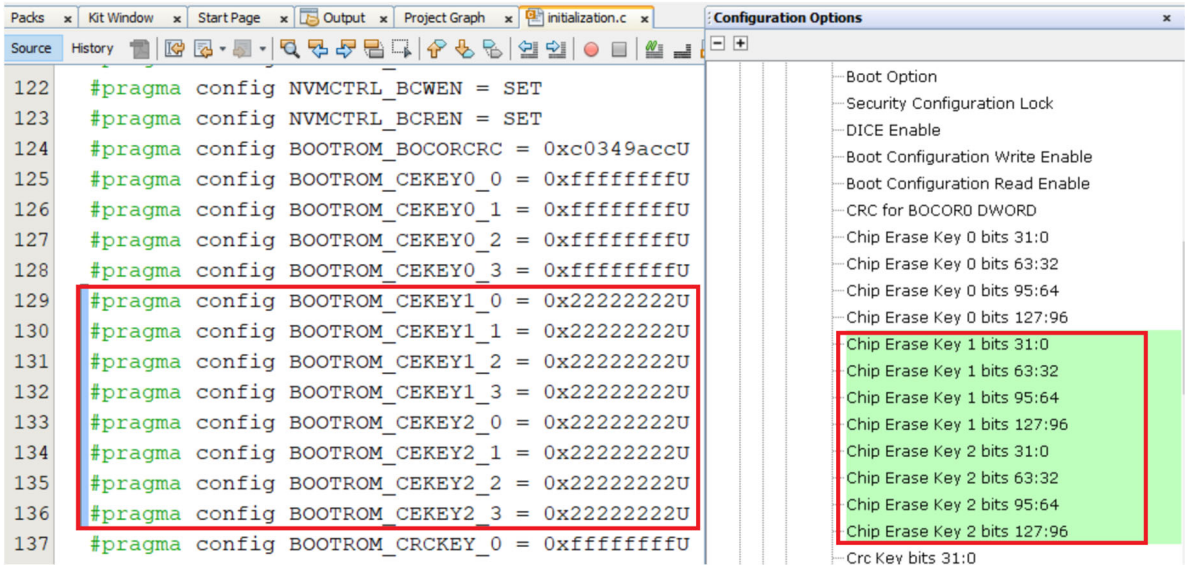


Note: Developer A must keep these keys confidential, which can be reused whenever an update is required for the Secure firmware or application. In this way, Developer B or an outsider cannot modify or erase the Secure firmware intentionally or accidentally. For example, the Chip Erase Secure and Chip Erase All memory keys used are as follows: `0x22222222,0x22222222,0x22222222,0x22222222`. The modified key will be stored in the Secure project configuration file.

This step helps in achieving IP protection, none other than Developer A can modify the code because only Developer A will have the confidential keys to program or change the Secure application.

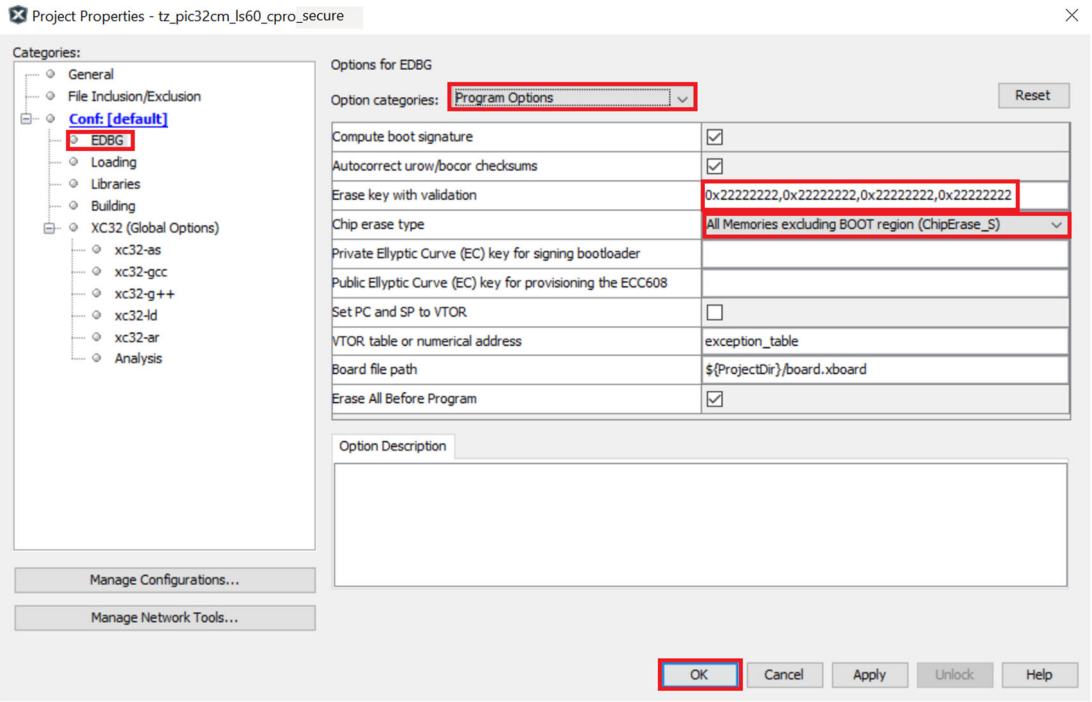
8. In the **Resource Management [MCC]** window, click **Generate code** to generate the updated Chip Erase keys. The following figure shows the updated pragmas for the Boot ROM Chip Erase Secure and Chip Erase All memory regions in the `initialization.c` file.

Figure 2-3. Updated Boot ROM Chip Erase Keys for Secure and All Memory Regions



9. Connect the PIC32CM LS60 Curiosity Pro device and program the Secure application to the device by clicking **Make and Program**. Now the device is programmed with the Chip Erase keys which are Secure.
   Note: When the Secure application developer wants to update the Secure firmware, the Chip Erase key must be entered under the device programming configuration options in MPLAB X IDE before programming the Secure application.
   a. In the MPLAB X IDE **Project Properties** window, under Categories, select **EDBG**.
   b. In the **Options for EDBG** section:
      i. For Option categories choose **Program Options.**
      ii. For Erase key with validation enter **0x22222222, 0x22222222, 0x22222222, 0x22222222.**
      iii. For the Chip erase type, choose **All Memories excluding BOOT region (ChipErase_S)**.
      Note: The programming might fail due to a mismatch in the Erase key. An Erase key is 4 x 32 bit numbers separated by a comma. This should match with the key entered in the MCC Project Graph Device Fuse Settings.

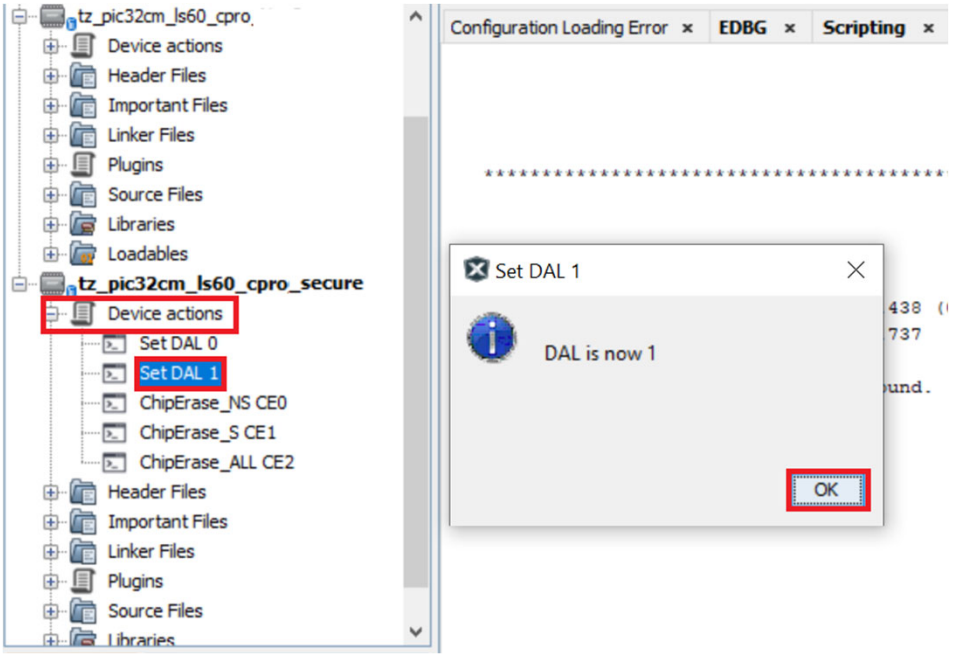
Figure 2-4. Secure Project EDBG Program Options Properties



c. Click **OK**.

10. Set the PIC32CM LS60 device Debug Access Level (DAL) to DAL1. DAL1 limits the device's access to the Non-Secure memory regions and the Secure memory regions accesses are forbidden.
11. To set the Debug Access Level to DAL1, follow these steps:
    a. In the **Projects** window, click and expand *tz_pic32cm_ls60_cpro_secure > Device Actions*, and then double-click **Set DAL1.**
    b. After the Debug Access Level is set, a popup message will be displayed indicating *DAL is now 1*.
    c. Click **OK**.

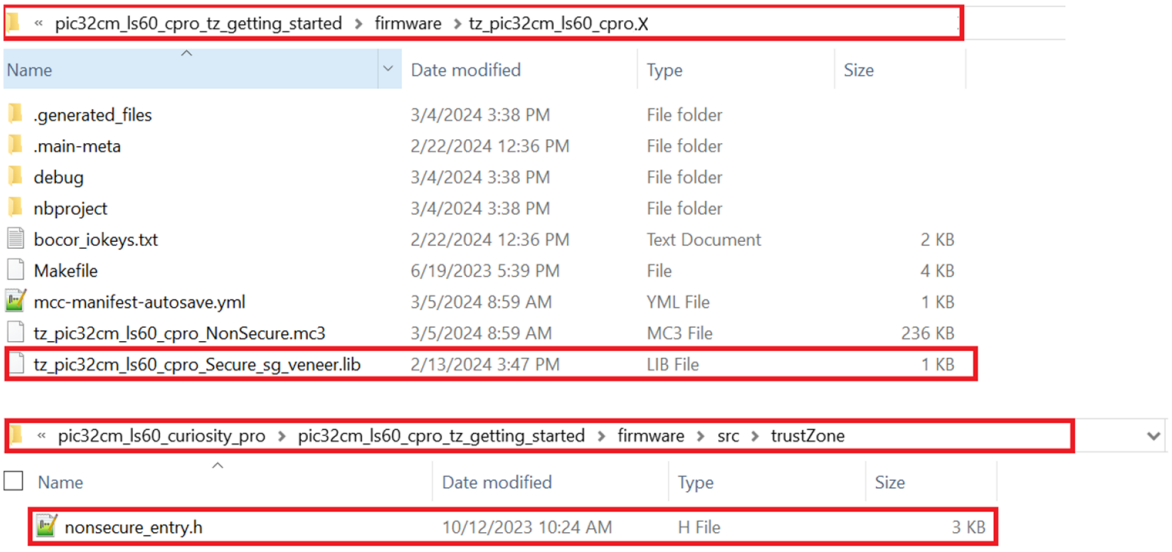
Figure 2-5. Set the Device to DAL1 to Lock the Access to the Secure Memory Region



Note: Setting the Chip Erase keys prevents Developer B from erasing the Secure memory content, but still they can read Secure memories and then read the Chip Erase keys. To avoid this, Developer A must set the device to DAL1 to lock Secure memories at a debug-level point. Refer to the product data sheet and the *PIC32CM LS60 Security Reference Guide* for additional information.

12. Developer A shares the programmed (Secure firmware) device with Developer B. The developer also shares the SG library and the `nonsecure_entry.h` file. The following figure shows the SG library and `nonsecure_entry.h` file locations.

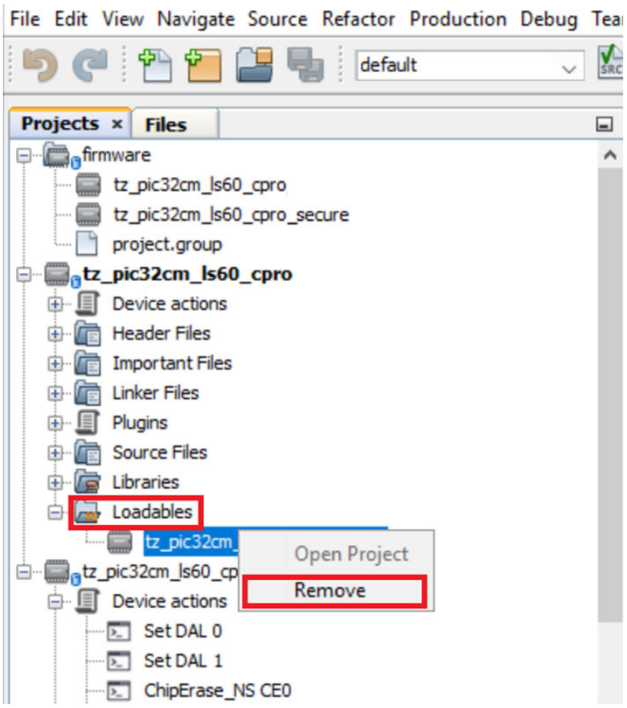
Figure 2-6. SG Library and the nonsecure_entry.h Header File



2.2.2 Non-Secure Project Configuration

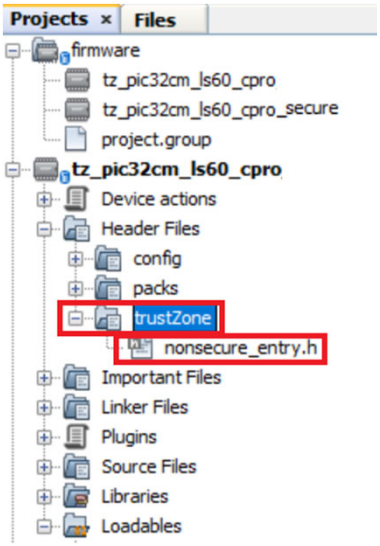
1. Developer B receives the Secure application pre-programmed on the PIC32CM LS60 device from Developer A.
2. The Non-Secure project development requires creating a TrustZone-enabled MPLAB Harmony v3 project by configuring the peripherals and memory regions in the Non-Secure mode. For the Non-Secure project development, any one of the following methods can be used:
   a. To create an MPLAB Harmony v3 TrustZone-based project using MPLAB X IDE, refer to the steps provided in the document Creating the First Application on PIC32CM LSx Microcontrollers Using MPLAB Harmony v3 with MPLAB Code Configurator (MCC).
   Note: When a PIC32CM LS60 TrustZone device is selected for creating the project, the MCC creates and enables configuring both the Secure and Non-Secure applications in the same project. The Non-Secure application developer must follow instructions relevant to the Non-Secure application.
   b. Use the PIC32CM LS60 TrustZone Getting Started Application demonstration, developerhelp.microchip.com/xwiki/bin/view/software-tools/harmony/pic32cm-trustzone-getting-started-training-module/.
3. Go to the PIC32CM LS60 TrustZone Getting Started Non-Secure application project path: *<DemoDeveloped or Extracted Path>\firmware\tz_pic32cm_ls60_cprogroup.*
   Note: The Secure project will have the name ending *_secure*.
   a. Open the pre-developed Non-Secure application in MPLAB X IDE.
   b. Under Projects, right-click on the project name `tz_pic32cm_ls60_cpro`, and then click **Set as Main Project.**
4. Follow these steps to remove the Secure project:
   a. Under Projects, click and expand *tz_pic32cm_ls60_cpro> Loadables*, and then select ***pic32cm_ls60_cpro_secure***.
   b. Click **Remove.**

Figure 2-7. Remove the Secure Project Under the Loadables



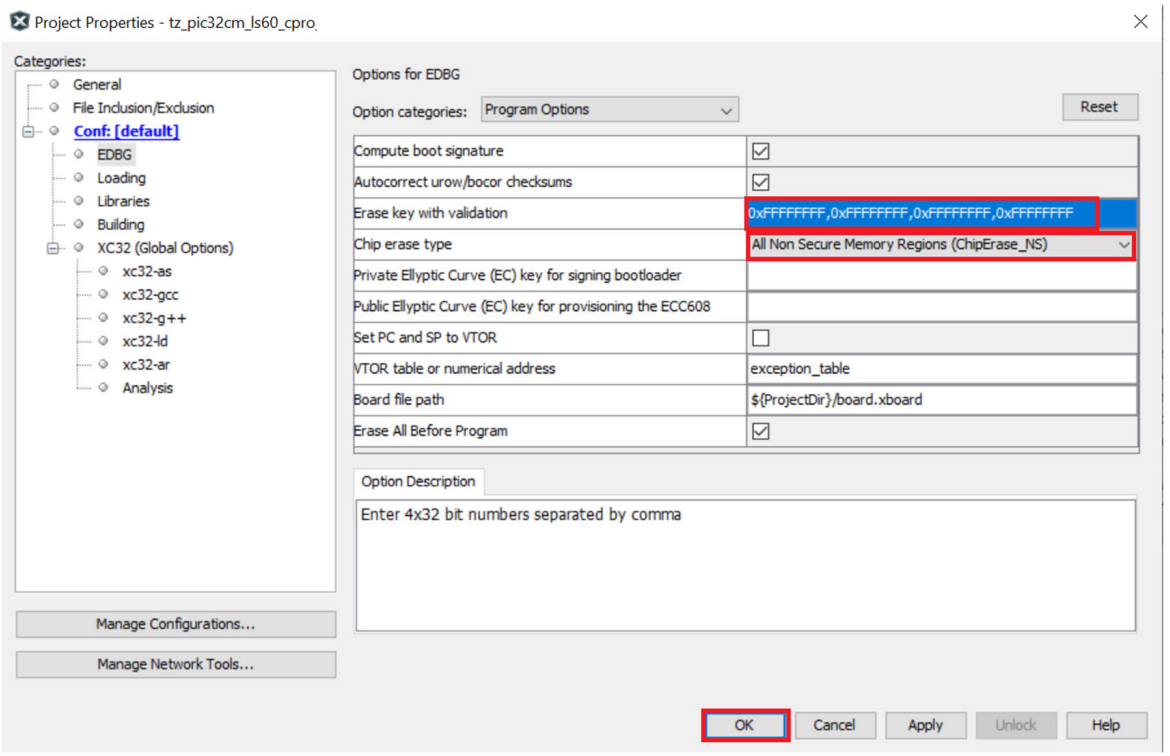
5. Add the SG library received from Developer A in the following path: `<Demo Developed or Demo ExtractedPath>\<firmware>\pic32cm_ls60_cpro.X`.
   Note: The library file ends with an extension `.lib`, for example, `pic32cm_ls60_cpro_secure_sg_veneer.lib`.
6. Add the Non-Secure entry header file (`nonsecure_entry.h`) received from Developer A in the following path: `<DemoDeveloped or Demo Extracted Path>\firmware\src\trustZone`.
7. Ensure that the Non-Secure header file is added to the project. Under Projects, click and expand `tz_pic32cm_ls60_cpro > Header Files > trustZone > nonsecure_entry.h` as shown in the following figure.

Figure 2-8. Non-Secure Entry Header File



8. Perform the task *Step 6: Add Application Code to the Non-Secure Project* mentioned in the document Creating the First Application on PIC32CM LSx Microcontrollers Using MPLAB Harmony v3 with MPLAB Code Configurator (MCC). Use the API declarations from the Non-Secure entry `nonsecure_entry.h` file to perform Secure applications requests.
9. Change the EDBG settings.
   a. In the MPLAB X IDE **Project Properties** window, under Categories, select **EDBG**.
   b. In the Options for the EDBG section:
      i. For Option categories choose **Program Options.**
      ii. For Erase key with validation enter **0xFFFFFFFF, 0xFFFFFFFF, 0xFFFFFFFF, 0xFFFFFFFF**.
      iii. For Chip erase type, choose **All Non Secure Memory Regions (ChipErase_NS)**.
   c. Click **OK**.

Figure 2-9. Non-Secure Project EDBG Program Options Properties



10. Connect the PIC32CM LS60 device and click **Make and Program** in the MPLAB X IDE to program the Non-Secure memory region with the Non-Secure application.

3. Conclusions

This document describes how a TrustZone-based application can be developed involving two developers working on the Secure and Non-Secure portions of the end application.

1. The two developers discuss and agree on the peripherals and memory regions configured in a Secure or Non-Secure mode. The developers develop Secure and Non-Secure applications separately.
2. The Secure application developer shares the Secure application programmed device, Secure gateway library file, and Non-Secure entry header file with the Non-Secure application developer.
3. The Non-Secure application developer uses the SG library and Non-Secure entry header file provided by the Secure application developer and implements the Non-Secure part of the end application by calling the functionality implemented in the pre-programmed device.
4. In this use case, the Secure World acts as the sandbox, providing a controlled and Secure environment for executing critical operations, while the Normal World handles non-sensitive tasks and interactions with the user. This dual developer approach enabled by TrustZone technology ensures that security and usability are effectively balanced in the smart payment application.
5. Since the Secure World is inaccessible from the Normal World without proper authorization, the IP developed within it remains Secure. This prevents unauthorized access or tampering with critical components, thereby safeguarding the intellectual property of the development team working in the Secure World. The ChipErase keys can be configured so that if the Secure region developer wants to update the Secure firmware next time, the developer must enter the key to program the Secure region, and no intruder can access this Secure region.

4. References

The following documents are used as reference. For additional information, visit the Microchip web site or contact a local Microchip sales office.

- PIC32CM Lx Family of Microcontrollers (MCUs)
- PIC32CM LS60 Curiosity Pro Evaluation Kit
- PIC32CM LS00 Curiosity Pro Evaluation Kit
- TrustZone Getting Started on PIC32CM LS60 Curiosity Pro Evaluation Kit
- TrustZone Getting Started on PIC32CM LS00 Curiosity Pro Evaluation Kit
- Getting Started With the TrustZone-based Security on PIC32CM LSx Microcontrollers Video
- PIC32CM LS60 Curiosity Pro Evaluation Kit Video
- PIC32CM LS00 Curiosity Pro Evaluation Kit Video
- MPLAB® Harmony v3

Microchip Information

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 products:

- 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, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip product is strictly prohibited and may violate the Digital Millennium Copyright Act.
- 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. Microchip is committed to continuously improving the code protection features of our products.

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure

that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP "AS IS". MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, Adaptec, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, CryptoMemory, CryptoRF, dsPIC, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Kleer, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

AgileSwitch, ClockWorks, The Embedded Control Solutions Company, EtherSynch, Flashtec, Hyper Speed Control, HyperLight Load, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet-Wire, SmartFusion, SyncWorld, TimeCesium, TimeHub, TimePictra, TimeProvider, and ZL are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, Augmented Switching, BlueSky, BodyCom, Clockstudio, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, Espresso T1S, EtherGREEN, EyeOpen, GridTime, IdealBridge, IGaT, In-Circuit Serial Programming, ICSP, INICnet, Intelligent Paralleling, IntelliMOS, Inter-Chip Connectivity, JitterBlocker, Knob-on-Display, MarginLink, maxCrypto, maxView, memBrain, Mindi, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, mSiC, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, Power MOS IV, Power MOS 7, PowerSmart, PureSilicon, QMatrix, REAL ICE, Ripple Blocker, RTAX, RTG4, SAM-ICE, Serial Quad I/O, simpleMAP, SimpliPHY, SmartBuffer, SmartHLS, SMART-I.S., storClad, SQI, SuperSwitcher, SuperSwitcher II, Switchtec, SynchroPHY, Total Endurance, Trusted Time, TSHARC, Turing, USBCheck, VariSense, VectorBlox, VeriPHY, ViewSpan, WiperLock, XpressConnect, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

The Adaptec logo, Frequency on Demand, Silicon Storage Technology, and Symmcom are registered trademarks of Microchip Technology Inc. in other countries.

GestIC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

ISBN: 978-1-6683-0027-5

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	**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	**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	**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
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	**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	**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	**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
San Jose, CA Tel: 408-735-9110 Tel: 408-436-4270 **Canada - Toronto** Tel: 905-695-1980 Fax: 905-695-2078			**UK - Wokingham** Tel: 44-118-921-5800 Fax: 44-118-921-5820