



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China



Getting Started with the SAM L10/L11 Xplained Pro

Abstract

This application note aims at getting started with the Atmel® SAM L10 and SAM L11 ARM®Cortex®-M23 based microcontrollers using their respective Xplained Pro evaluation kits.

The Atmel SAM L10 and SAM L11 Xplained Pro evaluation kits are hardware platforms used to evaluate the ATSAML10E16A and ATSAML11E16A microcontrollers.

Each kit is supported by the Atmel Studio Integrated development Platform, and provides an easy access to the features of the microcontroller.

Table of Contents

Abstract.....	1
1. Device Documentation.....	3
2. Obtain the SAM L10/SAM L11 Xplained Pro Evaluation Kit.....	4
3. Obtain the Tools.....	6
4. Getting Started With SAM L10/SAM L11 Using Atmel Studio 7 and START.....	7
4.1. Instructional Guide	7
5. Getting Started With SAM L11 Secure Solution Using Atmel Studio 7 and Start....	13
5.1. SAM L11 Security Concept Overview.....	13
5.2. Instructional Guide.....	15
The Microchip Web Site.....	38
Customer Change Notification Service.....	38
Customer Support.....	38
Microchip Devices Code Protection Feature.....	38
Legal Notice.....	39
Trademarks.....	39
Quality Management System Certified by DNV.....	40
Worldwide Sales and Service.....	41

1. Device Documentation

Data Sheet

Web page: <http://www.microchip.com/>.

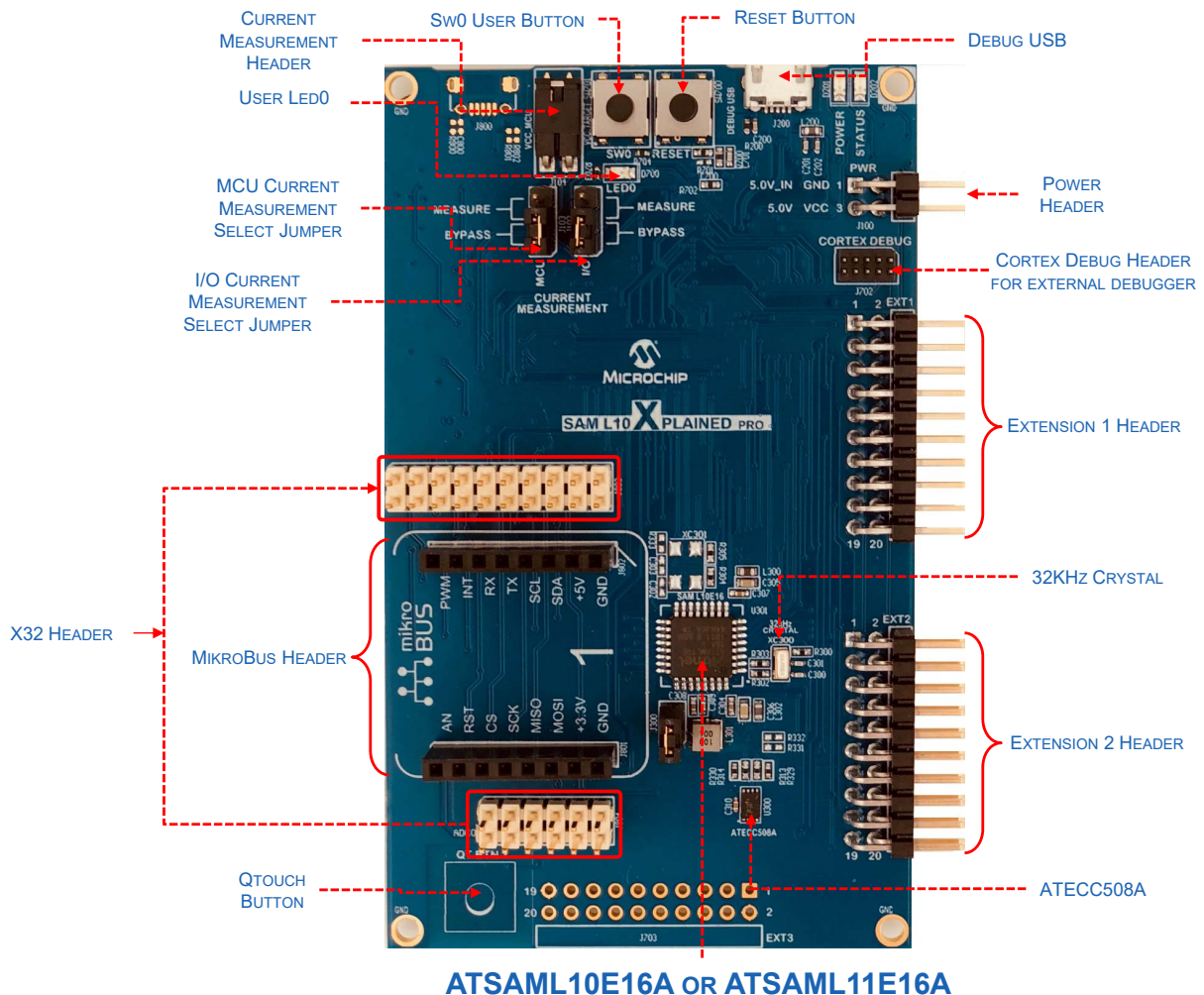
Document: SAM L10/L11 Family Data Sheet, which provides peripheral descriptions and electrical characteristics.

Silicon Errata

Web page: <http://www.microchip.com/>.

Document: SAM L10/L11 Family Silicon Errata and Data Sheet Clarification.

2. Obtain the SAM L10/SAM L11 Xplained Pro Evaluation Kit



Web page: <http://www.microchip.com/>.

To order the kit, access [Microchip Direct](#)

Documents:

- SAM L10/L11 Xplained Pro User Guide (.pdf)
- SAML10-Xplained-Pro_Design-Documentation (.zip)
- SAML11-Xplained-Pro_Design-Documentation (.zip)

Key Features:

- ATSAML10E16A-AU or ATSAML11E16A-AU microcontrollers
- One mechanical reset button
- One mechanical programmable button
- One QTouch® button
- One user LED (yellow)
- 32.768 kHz crystal

- ATECC508A Crypto Authentication IC
- Two Xplained Pro extension headers
- One X32 header
- One mikroBUS header
- Embedded Debugger
 - Auto-ID for board identification in Atmel Studio
 - One status LED (yellow)
 - One board power LED (green)
 - Symbolic debug of complex data types including scope information
 - Programming and debugging, including power measurements
 - Data Gateway Interface: SPI, I²C, four GPIOs
 - Virtual COM-port (CDC)
- Embedded current measurement circuitry with Atmel Data Visualizer support for data visualization
- USB powered



Tip: The SAM L10/SAM L11 Xplained Pro User's Guide describes how to power the kit, and it also describes the detailed information on board components, extension interface and the hardware guide.

3. Obtain the Tools

Atmel Studio 7, which uses a GCC compiler, is the preferred IDE to get started with SAM L10/SAM L11 MCUs. Atmel START is the preferred user interface to generate the initial project. Any Atmel START project can be used in Atmel Studio 7, IAR or KEIL IDEs.

Atmel Studio 7

- **Web page:** [Atmel Studio](#)
Document: Atmel Studio 7.0 web installer(.exe)

Atmel Studio 7 is the preferred IDE for developing and debugging firmware for SAM L10/SAM L11.

Atmel Start

- **Web page:** [Atmel Start](#)
Document: Atmel Start User's guide: Also, use the help menu from the interface.



Atmel Start enables users to select and configure software components and tailor embedded applications in a usable and optimized manner.

4. Getting Started With SAM L10/SAM L11 Using Atmel Studio 7 and START

Note: The following steps are described using the SAM L10 Xplained Pro as reference; however, the same sequence can be performed using the SAM L11 Xplained Pro.

4.1 Instructional Guide

Follow these steps to start exploring the Atmel Xplained Pro platform:

1. Download [Atmel Studio](#).
2. Install Atmel Studio.
3. Launch Atmel Studio.



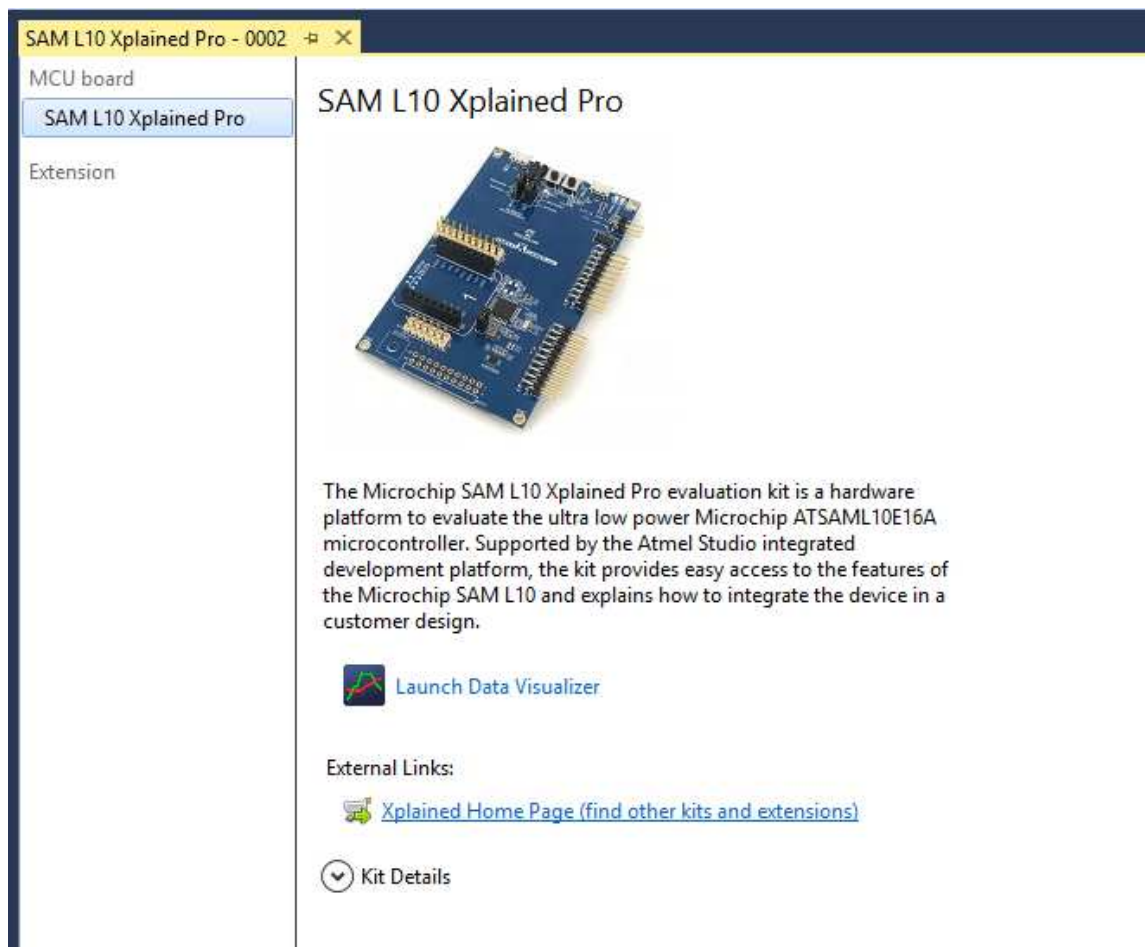
- Connect the kit to the PC using a micro-USB cable (Standard-A to Micro-AB).
When the Xplained Pro MCU kit is connected to the computer for the first time, the operating system will install the software driver. The driver file supports both 32-bit and 64-bit versions of Microsoft® Windows®XP, Windows Vista®, Windows 7, Windows 8 and Windows 10.



When the Xplained Pro MCU board is powered, the power LED (green) will glow and Atmel Studio will auto-detect the specific Xplained Pro MCU and extension boards that might be connected. Atmel Studio will provide links to relevant information, such as data sheets and kit documentation.

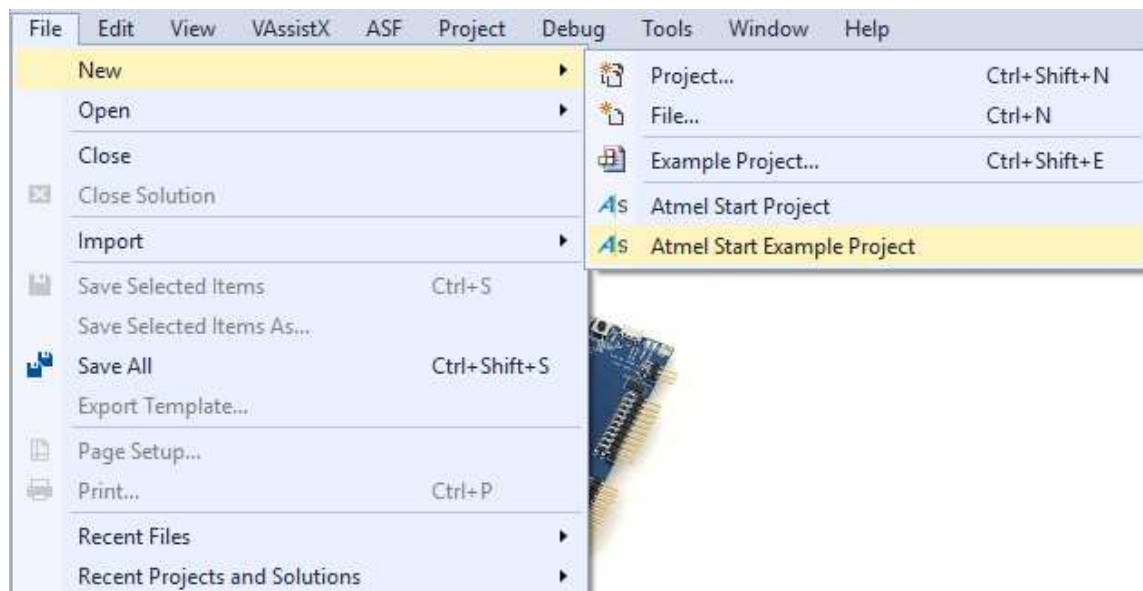
Note: The ATSAML10E16A and ATSAML11E16A devices are programmed and debugged by the on-board embedded debugger, hence no external programmer or debugger tool is required.

Figure 4-1. Atmel Studio SAM L10/SAM L11 Xplained Pro Introduction



5. Launch Atmel Start from Atmel Studio: *File > New > Atmel Start Example Project*, and then open Atmel Start directly into Atmel studio.

Figure 4-2. Opening Atmel Start Example Project in Atmel Studio



Atmel Start will load the existing example list.

6. Choose the example "LED Flasher" in the list, and then click **Open Selected Example**.

Figure 4-3. Atmel start SAM L1x Available Example List

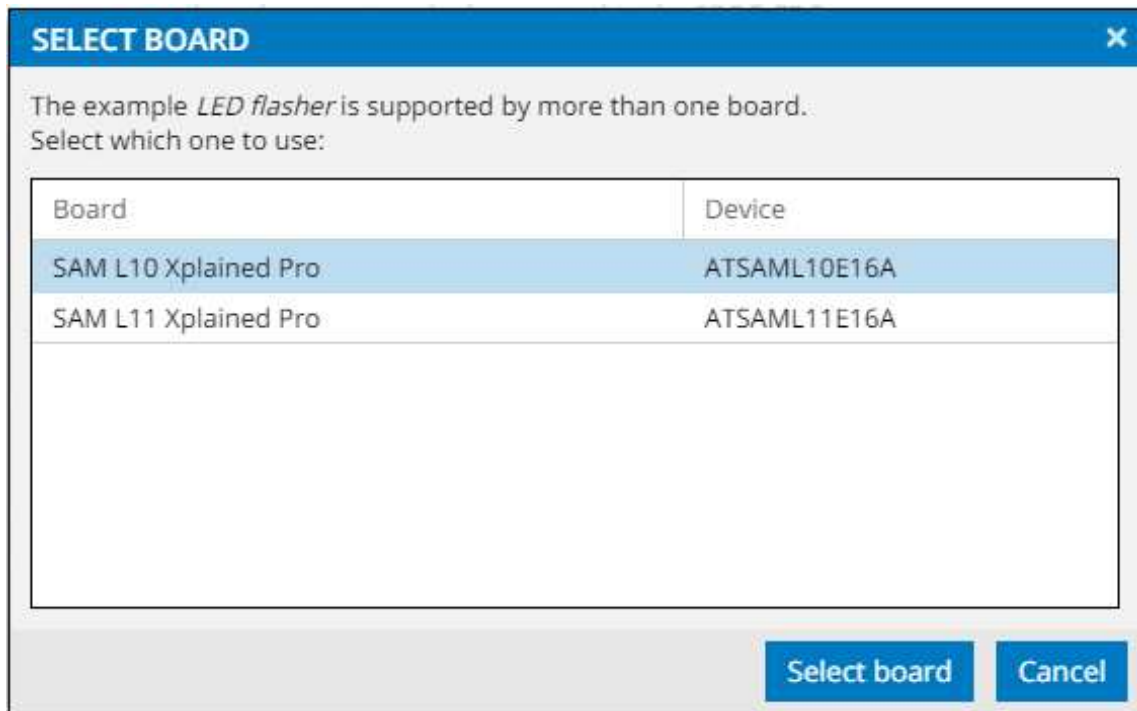
Name	Categories	Description	Board(s) supported	User guide
Calendar demo	I/O	This example shows the use of the Calendar driver and alarms to blink LED every 10 seconds.	SAM L10 Xplained Pro SAM L11 Xplained Pro	User guide
EDBG UART		This demo uses the UART usage example function to write data to the EDBG Virtual COM Port. Echo back input characters and toggle LED.	SAM L10 Xplained Pro SAM L11 Xplained Pro	User guide
LED flasher	I/O	This example periodically toggles an on-board LED.	SAM L10 Xplained Pro SAM L11 Xplained Pro	User guide
LED switcher	I/O	This example toggles an on-board LED every time when an on-board switch is pressed.	SAM L10 Xplained Pro SAM L11 Xplained Pro	User guide
Low power for SAML1X		This example demonstrates the different low power modes of the SAM L10/L11 which are DLE, STANDBY and OFF modes.	SAM L10 Xplained Pro SAM L11 Xplained Pro	User guide
PWM Example		This demo read light sensor on IO Xplained over ADC, and then output PWM on LED to reflect the current measured light level.	SAM L10 Xplained Pro SAM L11 Xplained Pro	User guide
Smart Card Example		This example does a read and write data example for smart	SAM L10 Xplained Pro	User guide

Showing 7 of 14 examples.

OPEN SELECTED EXAMPLE >

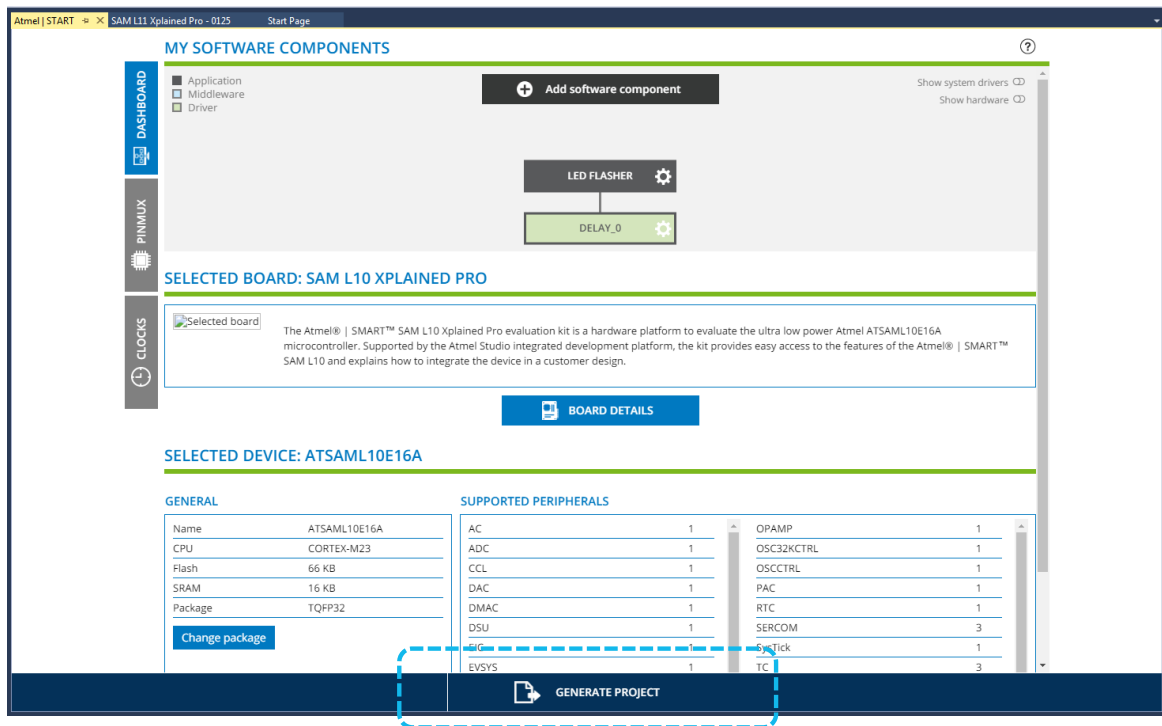
7. Choose either the SAM L10 or SAM L11 Xplained Pro Board, and then click **Select board**.

Figure 4-4. Board Selection



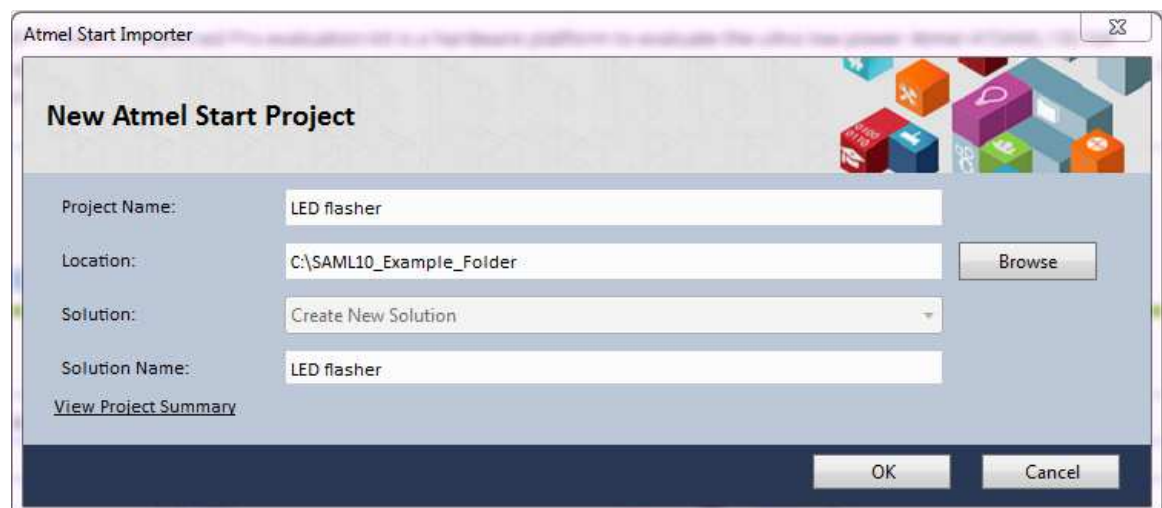
- Click **GENERATE PROJECT** to generate the project from Atmel Start. My "Software Components Window" will be displayed.

Figure 4-5. Atmel Start Project Generation



- The following window will be displayed: Enter Project Name, Solution, Solution Name, and then browse and choose a location for the project. Click **OK** to open the project in Atmel Studio.

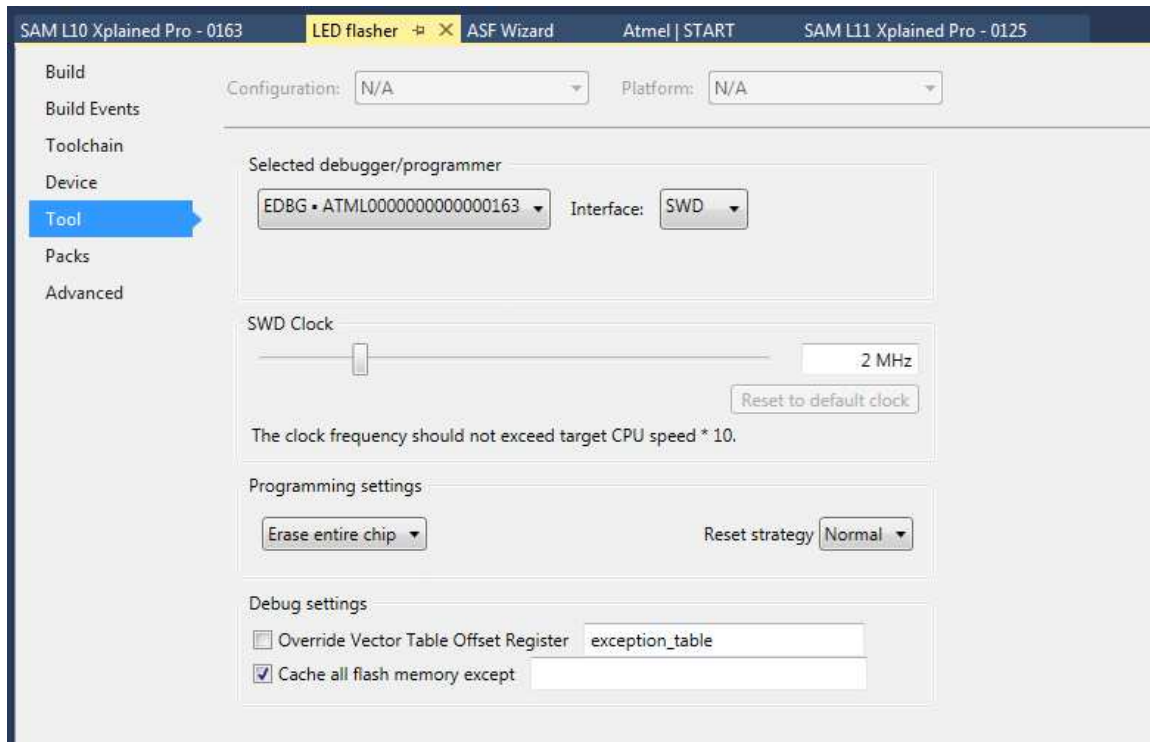
Figure 4-6. Atmel Studio New Atmel Start Project Importation



The Atmel Studio will create the project.

10. Configure the Debugger/Programmer Interface by following these steps:
 - Open the project properties: *Project > Properties* or <ALT+F7>.
 - Click **Tool**.
 - For the Selected debugger/programmer, choose "EDBG ATMLxxx".
 - For the Interface, choose "SWD".

Figure 4-7. Programming Tool Selection



11. Compile and run the LED Flasher application.
 - Build the project: *Build > Build Solution* or <F7>.
 - Load the code into the SAM L10 Xplained Pro and start debugging: *Debug > Start debugging and break* or <ALT+F5>.
 - The application is programmed and the debugger breaks in main.
 - Run the code: *Debug > Continue* or <F5>.
 - The example runs out of the Xplained Pro target.

5. Getting Started With SAM L11 Secure Solution Using Atmel Studio 7 and Start

5.1 SAM L11 Security Concept Overview

Using the ATSAM L11 requires to be familiar with different security features and concepts that involve a TrustZone® for ARMv8-M devices.

The TrustZone technology is a System-on-Chip (SoC) and MCU system-wide approach to security that enables Secure and Non-Secure code to run on a MCU. It enables creating multiple software security domains that restrict access to selected memory, peripherals, and I/O to trusted software without compromising the system performances. The user can consider the following deployment approaches:

- Single-developer approach (Customer A)
- Dual-developer approach (Customer A+ Customer B)

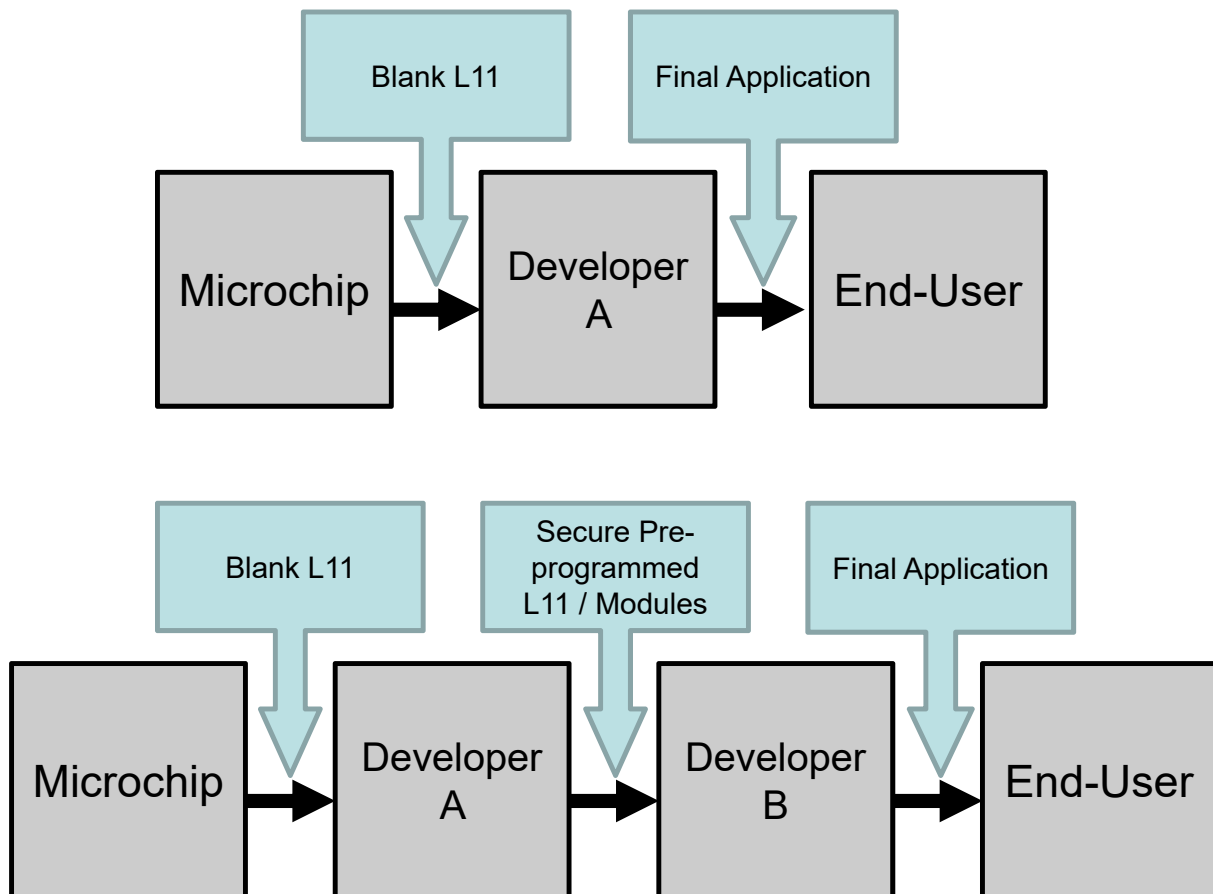
The Single developer approach involves a unique developer (Customer A), which is in charge of the following:

- Developing, deploying, and protecting the Secure code
- Developing and deploying the Non-Secure code

In the Dual-Developer approach, the first developer (Customer A) is in charge of developing the Secure application and its associated Non-Secure callable library. The Secure Application must be loaded in the SAM L11 NVM and protected.

A different developer (Customer B) will then start Non-Secured application development on a preprogrammed SAM L11 with limited access to Secure resources (call to Non-Secure API only).

Figure 5-1. Single-Developer and Dual Developer Approaches



This document describes how to debug a solution composed of the following two projects:

- Secure Project
- Non-Secure Project

Note: Refer to the "SAM L11 Security Reference Guide" application note, which describes the security features available in the Microchip SAM L11 microcontroller that fulfill the security requirement of most embedded systems.

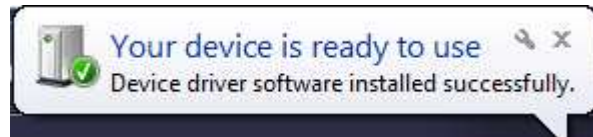
5.2 Instructional Guide

Follow these steps to explore the Atmel Xplained Pro platform:

1. Download [Atmel Studio](#).
2. Install Atmel Studio.
3. Launch Atmel Studio.



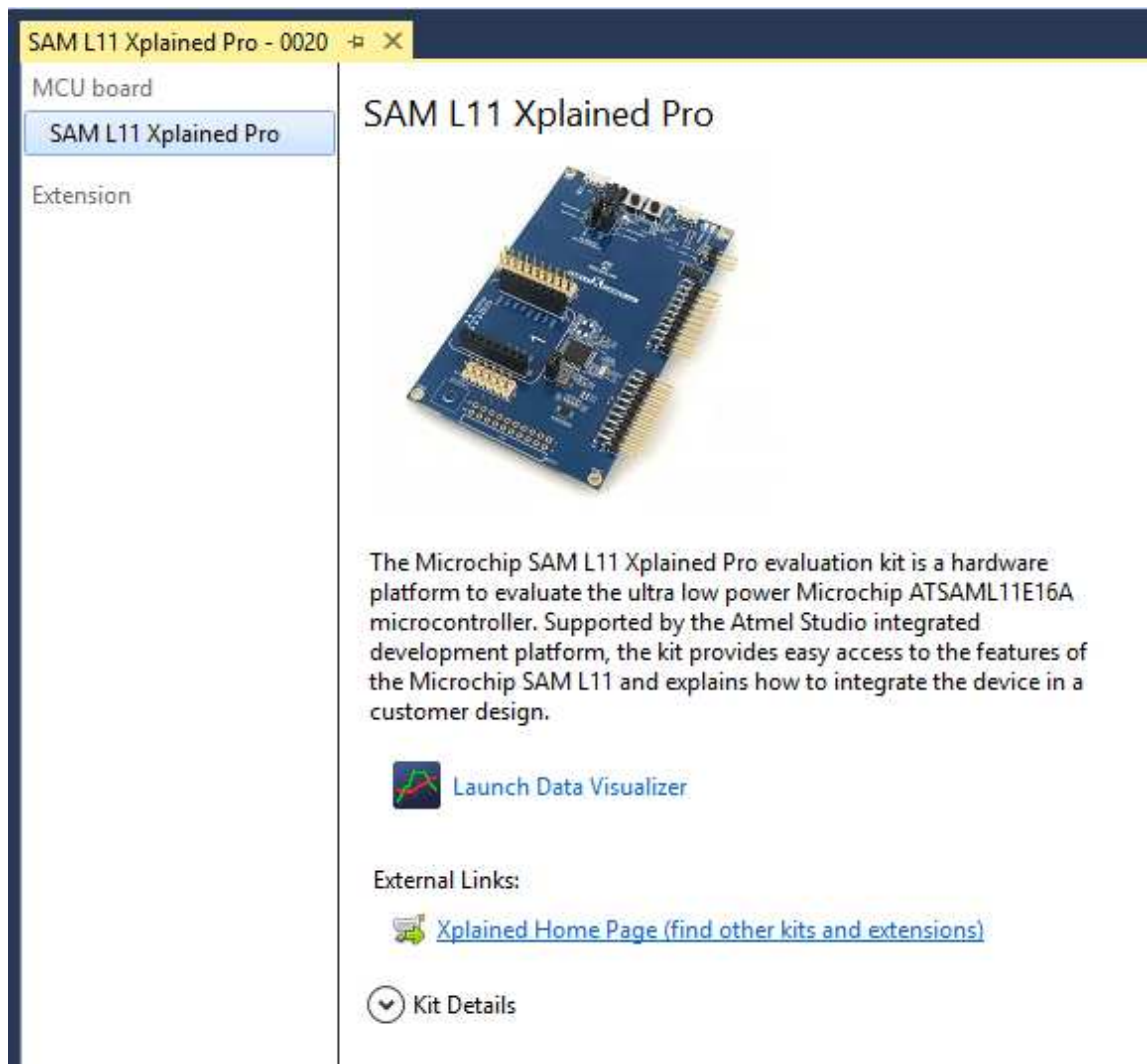
4. Connect the DEBUG USB port on the kit to the PC using a micro-USB cable (Standard-A to Micro-AB). When the Xplained Pro MCU kit is connected to the computer for the first time, the operating system will install the software driver. The driver file supports 32-bit and 64-bit versions of Microsoft® Windows®XP, Windows Vista®, Windows 7, Windows 8 and Windows10.



When the Xplained Pro MCU board is powered, the power LED (green) will glow. Atmel Studio will auto-detect the specific Xplained Pro MCU and extension boards that are connected. Atmel Studio will present relevant information, such as data sheets and kit documentation.

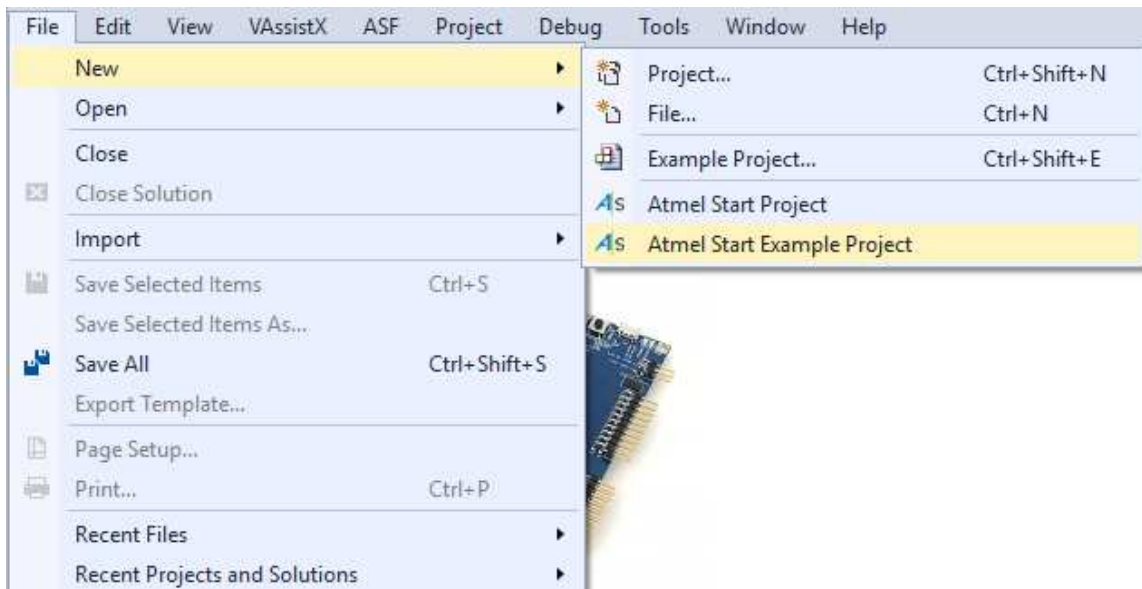
The ATSAML11E16A device is programmed and debugged by the on-board embedded debugger, hence no external programmer or debugger tool is required.

Figure 5-2. Atmel Studio SAM L11 Xplained Pro Introduction



5. Launch Atmel Start from Atmel Studio to Open the Secure Application Project: *File > New > Atmel Start Example Project* and then open Atmel Start into Atmel studio 7.

Figure 5-3. Open a New Atmel Start Example Project in Atmel Studio



Note: After few seconds the example list will be displayed.

6. Select the "TrustZone Getting Started Example" for the Secure Project (TZ-GetStart-S) from the examples list, and then click **OPEN SELECTED EXAMPLE**.

Figure 5-4. TrustZone Getting started SAM L11 Secure Project Example Selection

Search: Category: Board:

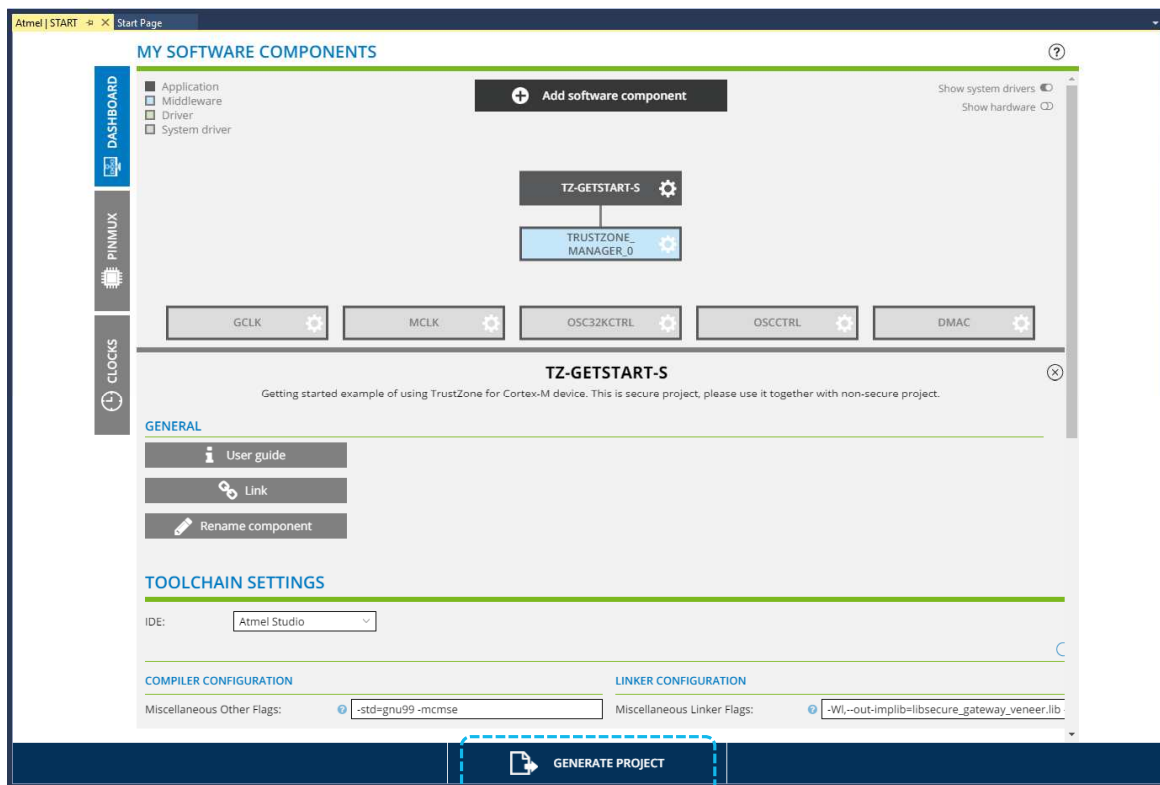
Name	Categories	Description	Board(s) supported	User guide
		ATECC508A CryptoAuthentication™ device to demonstrate secure node provisioning and node authentication using PKI methodology. Specific details include internal key generation, creation of signer and device certificates, and certificate chain verification from device to the certificate root.	SAM L10 Xplained Pro	User guide
PWM Example		This demo read light sensor on IO Xplained over ADC, and then output PWM on LED to reflect the current measured light level.	SAM L10 Xplained Pro SAM L11 Xplained Pro	User guide
Smart Card Example		This example does a read and write data example for smart card.	SAM L10 Xplained Pro SAM L11 Xplained Pro	User guide
TZ-GetStart-NS		Getting started example of using TrustZone for Cortex-M device. This is non-secure project, please use it together with secure project.	SAM L11 Xplained Pro	User guide
TZ-GetStart-S		Getting started example of using TrustZone for Cortex-M device. This is secure project, please use it together with non-secure project.	SAM L11 Xplained Pro	User guide
TZ-MixSecureEIC-NS		This example illustrates the use of TrustZone for Cortex-M device of mix secure peripheral EIC. This is non-secure project, please use it together with secure project.	SAM L11 Xplained Pro	User guide
TZ-MixSecureEIC-S		This example illustrates the use of TrustZone for Cortex-M device of mix secure peripheral EIC. This is secure project, please use it together with non-secure project.	SAM L11 Xplained Pro	User guide
TZ-SecureDriver-NS		This example illustrates the use of TrustZone for Cortex-M device by isolating temperature sensor driver source code	SAM L11 Xplained Pro	User guide

Showing 19 of 19 examples.

[OPEN SELECTED EXAMPLE](#)

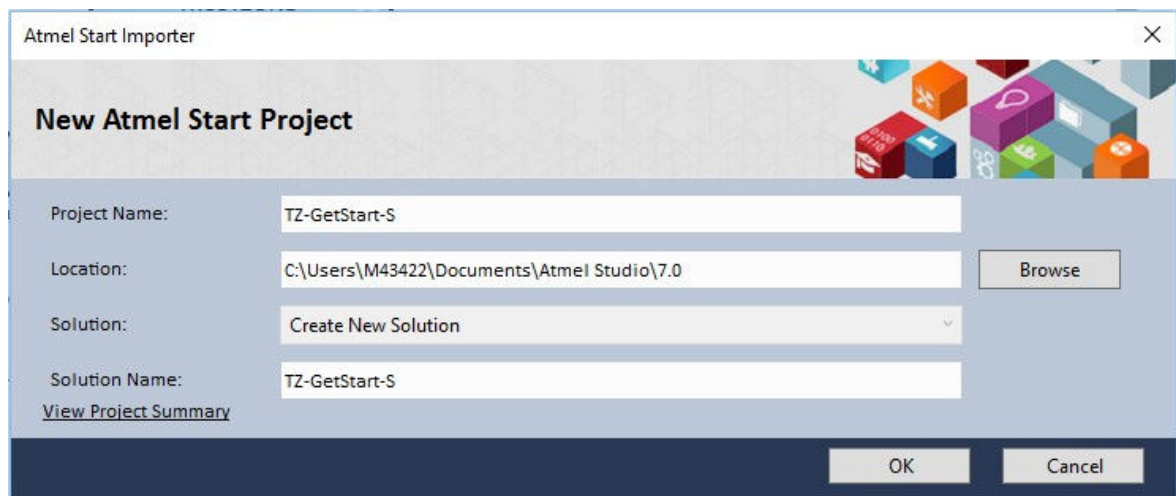
7. Generate the project from Atmel Start to open it in Atmel Studio. Once the project is open, click **GENERATE PROJECT**.

Figure 5-5. Atmel Start Secure Project Overview and Generation



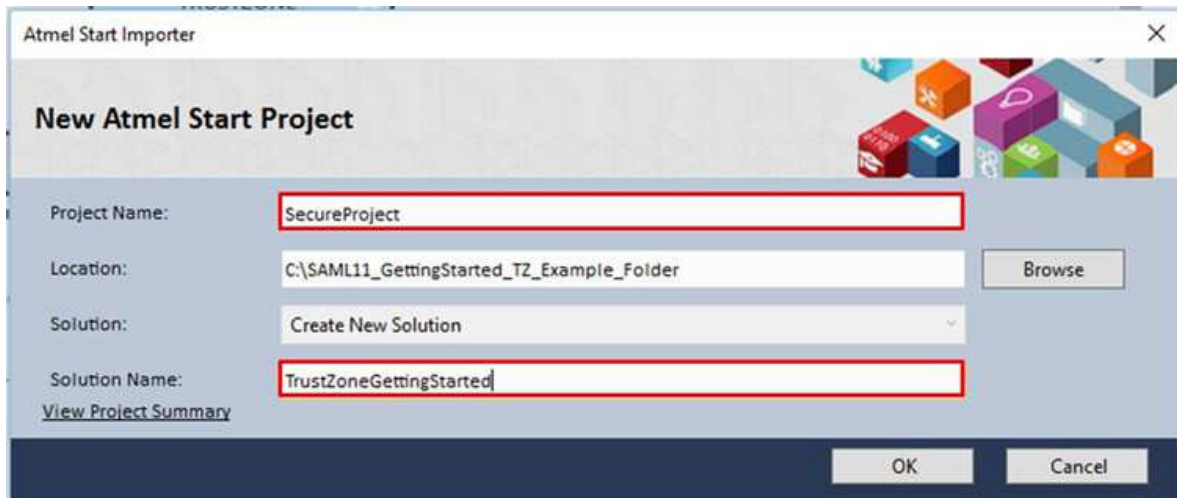
8. The following "New Atmel Start Project" window will be displayed.

Figure 5-6. Secure Project Importation View



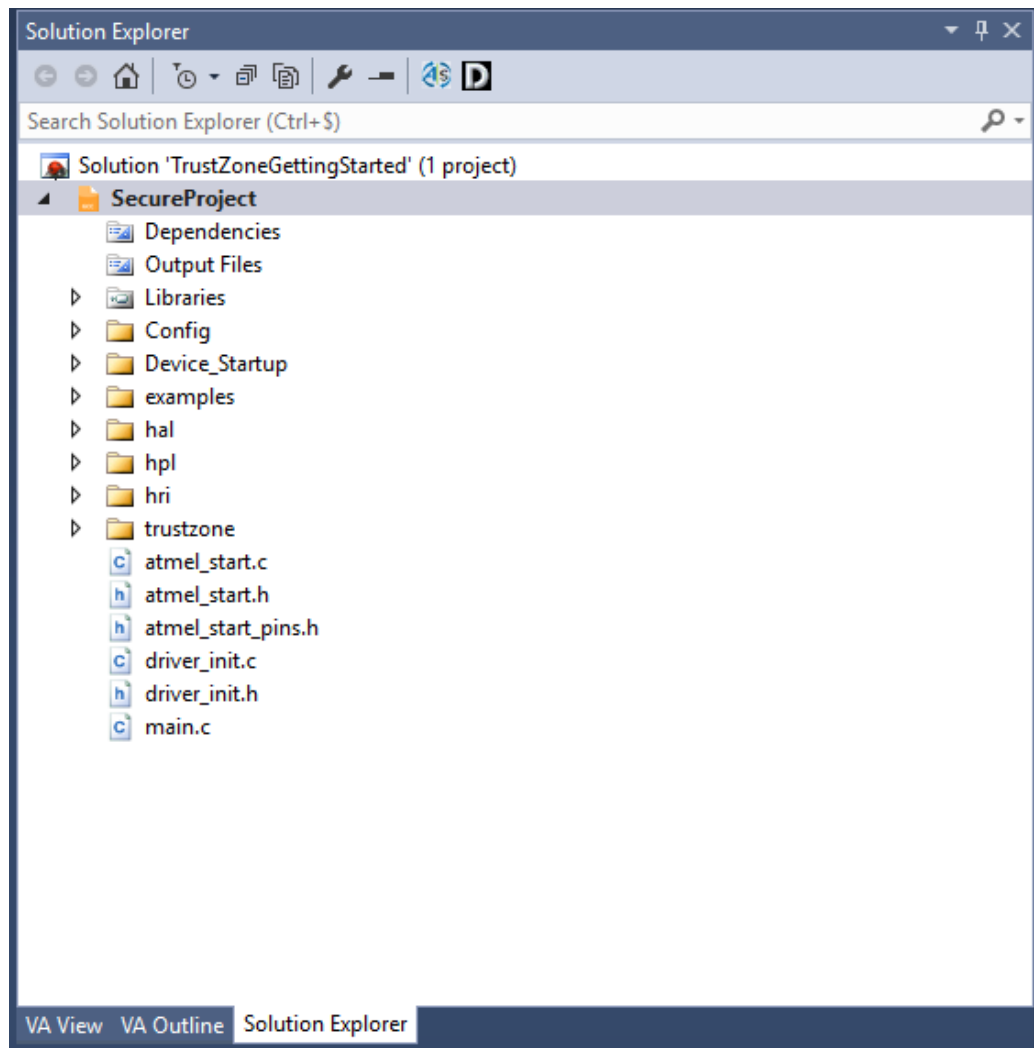
9. Modify the following project information:
 - Enter a new Project Name.
 - Click **Browse** to choose a location.
 - For Solution: Create New Solution.
 - For Solution Name, enter *TrustZoneGettingStarted*.
 - Click **OK** to open the project in Atmel Studio.

Figure 5-7. Modifying the Project Information



10. To see the project in the Atmel Studio Solution Explorer: *View > Solution Explorer* or <CTRL+ALT+L>.

Figure 5-8. Atmel Studio Solution Explorer Showing Secure Project



Currently, the TrustZoneGettingStarted solution is only composed of the *SecureProject*.

- To compile the "TrustZoneGettingStarted " Secure application, build the project: *Build > Build Solution* or *<F7 >..*

The current building of the project is important as it enables the generation of the Secure library gateway used in the future Non-Secure application.

Figure 5-9. Secure Project Building Output Window

```

Output
Show output from: Build
C:\Program Files (x86)\Atmel\Studio\7.0\Vs\Avr.common.targets (line 106, column 1) : output byte
make: Nothing to be done for 'all'.
Done executing task "RunCompilerTask".
Task "RunOutputFileVerifyTask"
    Program Memory Usage : 1220 bytes 1.8 % Full
    Data Memory Usage : 1568 bytes 9.6 % Full
Done executing task "RunOutputFileVerifyTask".
Done building target "CoreBuild" in project "SecureProject.cproj".
Target "PostBuildEvent" skipped, due to false condition; ('$(PostBuildEvent)' != '') was evaluated as ('' != '').
Target "Build" in file "C:\Program Files (x86)\Atmel\Studio\7.0\Vs\Avr.common.targets" from project "C:\SAML11_GettingStarted_TZ_Examp
Done building target "Build" in project "SecureProject.cproj".
Done building project "SecureProject.cproj".

Build succeeded.
===== Build: 1 succeeded or up-to-date, 0 failed, 0 skipped =====
|
Output
Build succeeded

```

The Secure Application is built, and the Secure library gateway has been generated in the *SecureProject /Debug* folder:

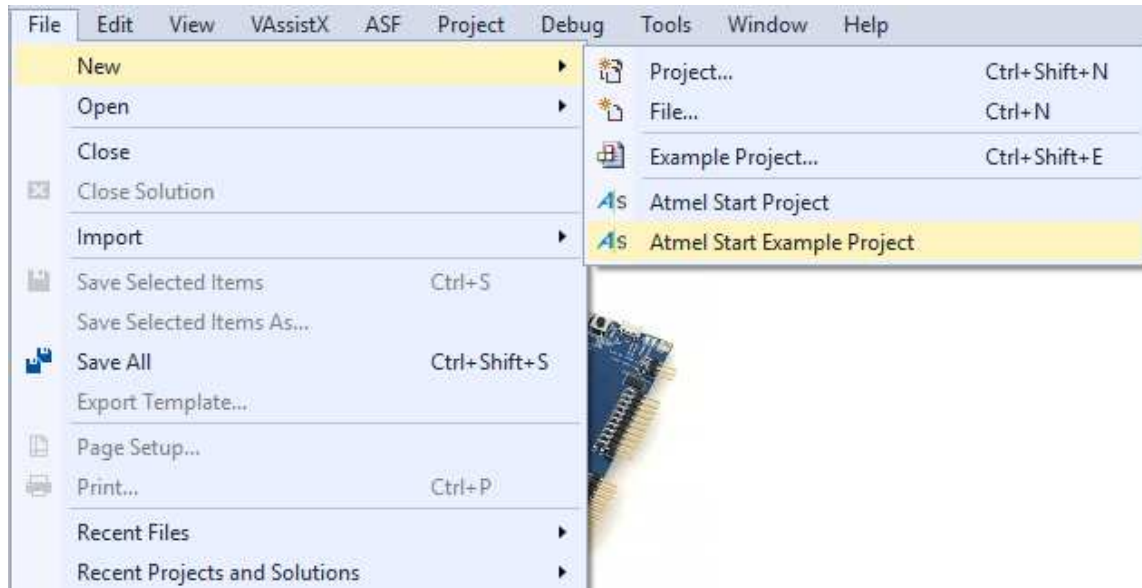
Figure 5-10. Compilation Resulting Secure Library File

The screenshot shows a file explorer window with the following path: `ed_TZ_Example_Folder > TrustZoneGettingStarted > SecureProject > Debug`. The window title is "Search Debug". The file list is as follows:

Name	Date modified	Type	Size
Config	5/25/2018 10:32 AM	File folder	
Device_Startup	5/25/2018 10:56 AM	File folder	
examples	5/25/2018 10:56 AM	File folder	
hal	5/25/2018 10:32 AM	File folder	
hpl	5/25/2018 10:32 AM	File folder	
hri	5/25/2018 10:32 AM	File folder	
trustzone	5/25/2018 10:56 AM	File folder	
atmel_start.d	5/25/2018 10:56 AM	D File	19 KB
atmel_start.o	5/25/2018 10:56 AM	O File	728 KB
driver_init.d	5/25/2018 10:56 AM	D File	18 KB
driver_init.o	5/25/2018 10:56 AM	O File	712 KB
libsecure_gateway_veneer.lib	5/25/2018 10:56 AM	Altium Library	1 KB
main.d	5/25/2018 10:56 AM	D File	19 KB
main.o	5/25/2018 10:56 AM	O File	725 KB
makedep.mk	5/25/2018 10:32 AM	MK File	1 KB
Makefile	5/25/2018 10:56 AM	File	31 KB
SecureProject.bin	5/25/2018 10:56 AM	BIN File	8,209 KB
SecureProject.eep	5/25/2018 10:56 AM	EEP File	0 KB
SecureProject.elf	5/25/2018 10:56 AM	ELF File	833 KB
SecureProject.hex	5/25/2018 10:56 AM	HEX File	4 KB
SecureProject.lss	5/25/2018 10:56 AM	LSS File	24 KB
SecureProject.map	5/25/2018 10:56 AM	MAP File	432 KB
SecureProject.srec	5/25/2018 10:56 AM	SREC File	4 KB

12. Close the Atmel Start window from Atmel Studio.
13. Reopen a new Atmel Start Example Project to create the Non-Secure application:
 - *File > New > Atmel Start Example Project.*
 - Open Atmel START directly into Atmel Studio 7.

Figure 5-11. Open a New Atmel Start Example Project in Atmel Studio



14. Choose the "TrustZone Getting Started Example" for the Non-Secure Project (TZ-GetStart-NS) from the existing examples list, and then click **OPEN SELECTED EXAMPLE**.

Figure 5-12. TrustZone Getting started SAM L11 Non-Secure Project Example Selection

Search: Category: Board:

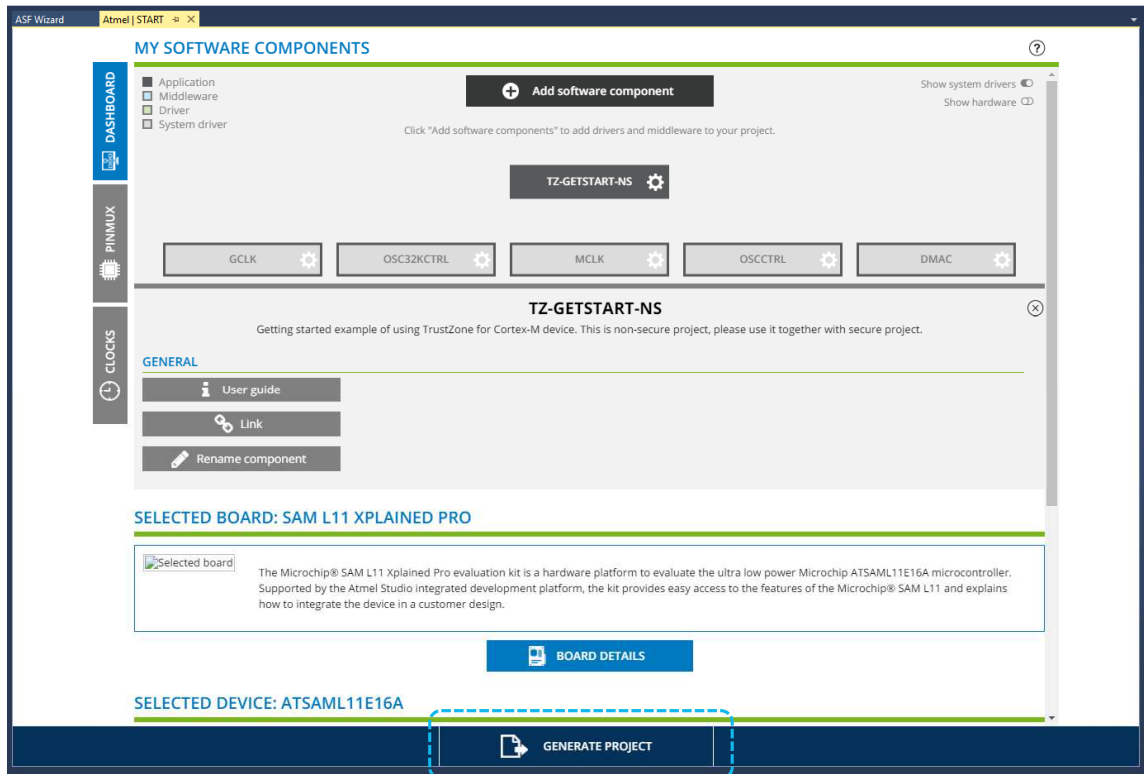
Name	Categories	Description	Board(s) supported	User guide
PWM Example		This demo read light sensor on IO Xplained over ADC, and then output PWM on LED to reflect the current measured light level.	SAM L10 Xplained Pro SAM L11 Xplained Pro	User guide
Smart Card Example		This example does a read and write data example for smart card.	SAM L10 Xplained Pro SAM L11 Xplained Pro	User guide
TZ-GetStart-NS		Getting started example of using TrustZone for Cortex-M device. This is non-secure project, please use it together with secure project.	SAM L11 Xplained Pro	User guide
TZ-GetStart-S		Getting started example of using TrustZone for Cortex-M device. This is secure project, please use it together with non-secure project.	SAM L11 Xplained Pro	User guide
TZ-MixSecureEIC-NS		This example illustrates the use of TrustZone for Cortex-M device of mix secure peripheral EIC. This is non-secure project, please use it together with secure project.	SAM L11 Xplained Pro	User guide
TZ-MixSecureEIC-S		This example illustrates the use of TrustZone for Cortex-M device of mix secure peripheral EIC. This is secure project, please use it together with non-secure project.	SAM L11 Xplained Pro	User guide
TZ-SecureDriver-NS		This example illustrates the use of TrustZone for Cortex-M device by isolating temperature sensor driver source code from non-secure world. This is non-secure project, please use it together with secure project.	SAM L11 Xplained Pro	User guide
TZ-SecureDriver-S		This example illustrates the use of TrustZone for Cortex-M device by isolating temperature sensor driver source code from non-secure world. This is secure project, please use it together with non-secure project.	SAM L11 Xplained Pro	User guide

Showing 19 of 19 examples.

OPEN SELECTED EXAMPLE

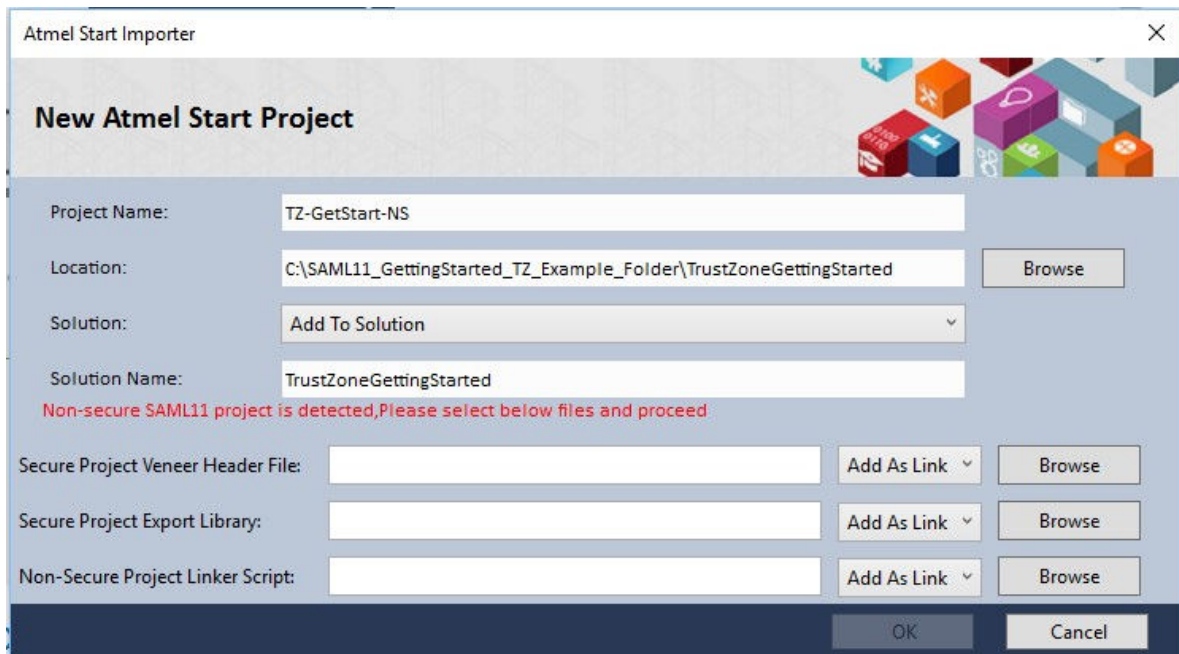
15. Generate the project from Atmel Start to open it in Atmel Studio, and once it is loaded, click **GENERATE PROJECT**.

Figure 5-13. Atmel Start Non-Secure Project Overview and Generation



16. The following "New Atmel Start Project " window will be displayed.

Figure 5-14. Non-Secure Project Importation Window



Atmel Start Importer

New Atmel Start Project

Project Name: TZ-GetStart-NS

Location: C:\SAML11_GettingStarted_TZ_Example_Folder\TrustZoneGettingStarted

Solution: Add To Solution

Solution Name: TrustZoneGettingStarted

Non-secure SAML11 project is detected, Please select below files and proceed

Secure Project Veneer Header File:

Secure Project Export Library:

Non-Secure Project Linker Script: