Chipsmall

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

AN2722

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

1.   Device Documentation

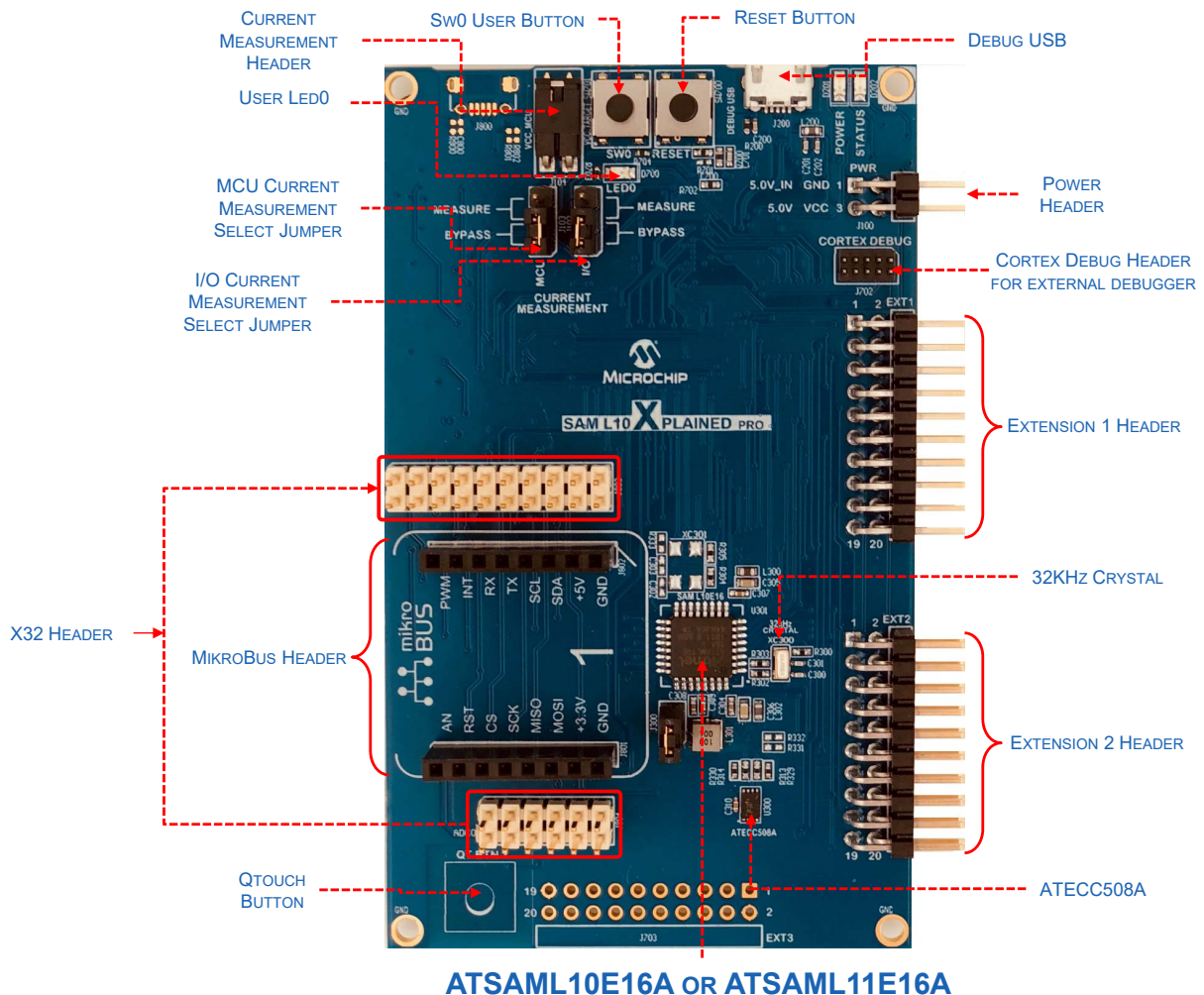**Data Sheet**
Web page: http://www.microchip.com/.

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

CURRENT MEASUREMENT HEADER

SW0 USER BUTTON

RESET BUTTON

DEBUG USB

USER LED0

MCU CURRENT MEASUREMENT SELECT JUMPER

I/O CURRENT MEASUREMENT SELECT JUMPER

X32 HEADER

MIKROBUS HEADER

QTOUCH BUTTON

POWER HEADER

CORTEX DEBUG HEADER FOR EXTERNAL DEBUGGER

EXTENSION 1 HEADER

32KHZ CRYSTAL

EXTENSION 2 HEADER

ATECC508A

ATSAML10E16A OR ATSAML11E16A

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, I^2C, 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.

4. 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
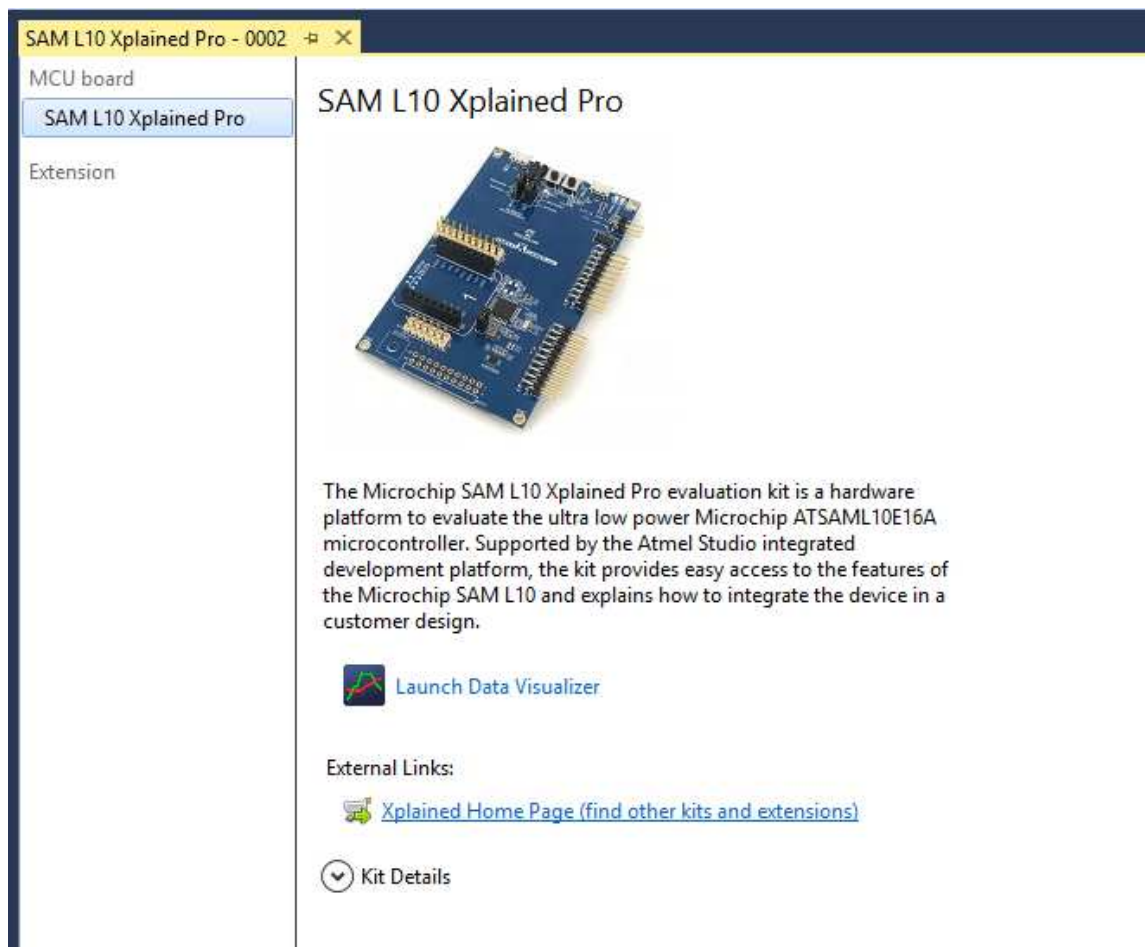
5. Launch Atmel Start from Atmel Studio: *File > New > Atmel Start Example Project*, and then open Atmel Start directly into Atmel studio.

    Figure 4-2. Opening Atmel Start Example Project in Atmel Studio
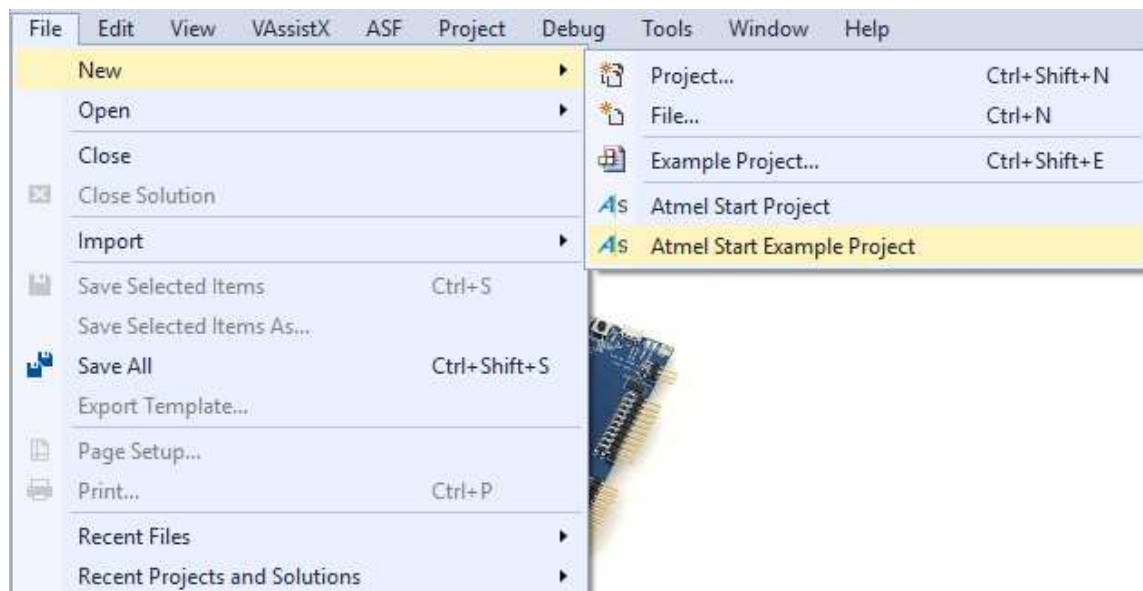
    Atmel Start will load the existing example list.

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

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

Figure 4-4. Board Selection
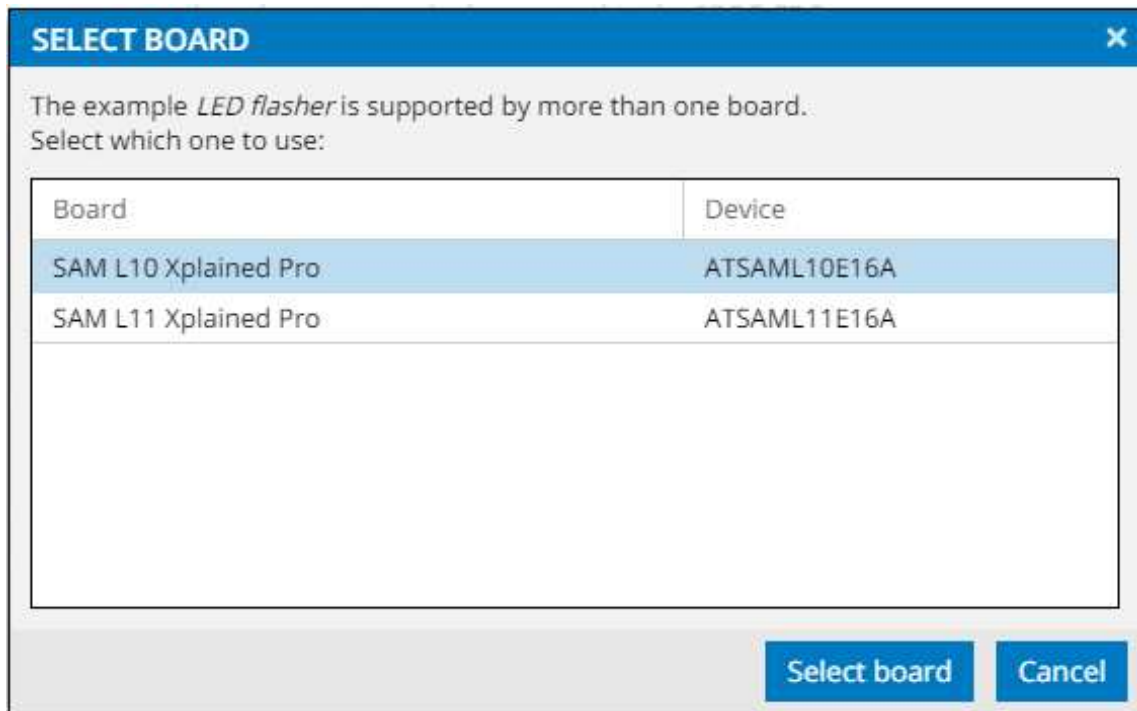
8. Click **GENERATE PROJECT** to generate the project from Atmel Start. My "Software Components Window" will be displayed.

Figure 4-5. Atmel Start Project Generation
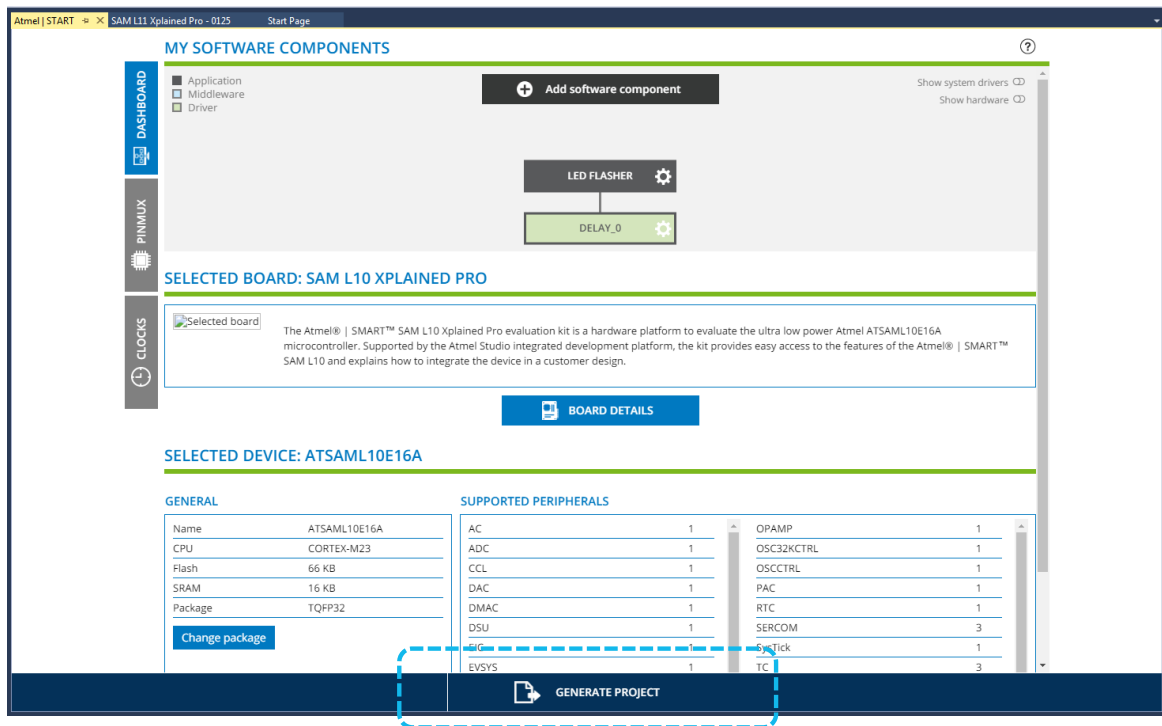
9. The following window will be displayed: Enter Project Name, Solution, Solution Name, and then browse and choose a location for the project. Click **OK** to open the project in Atmel Studio.

Figure 4-6. Atmel Studio New Atmel Start Project Importation
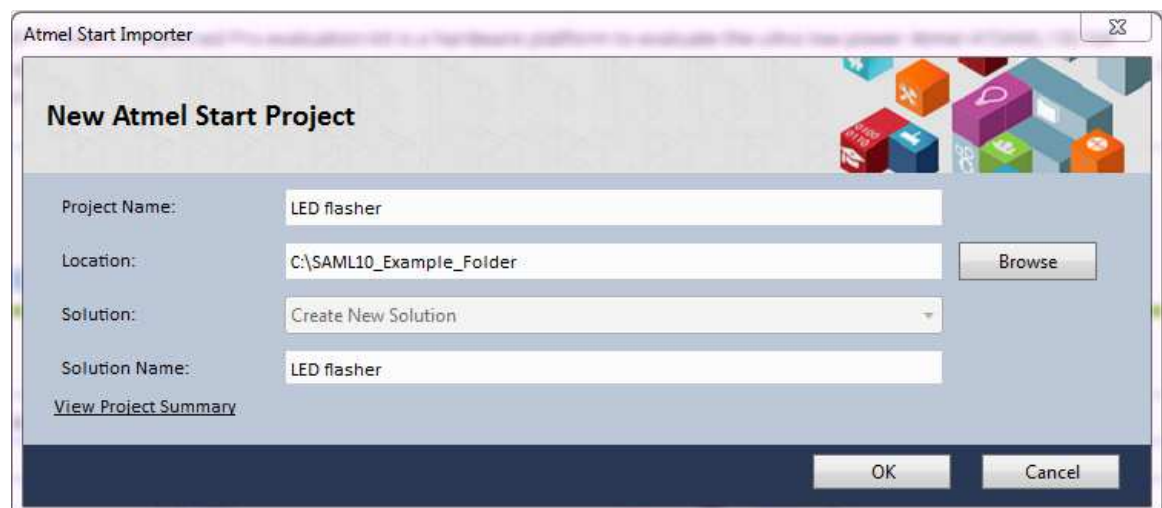
The Atmel Studio will create the project.

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
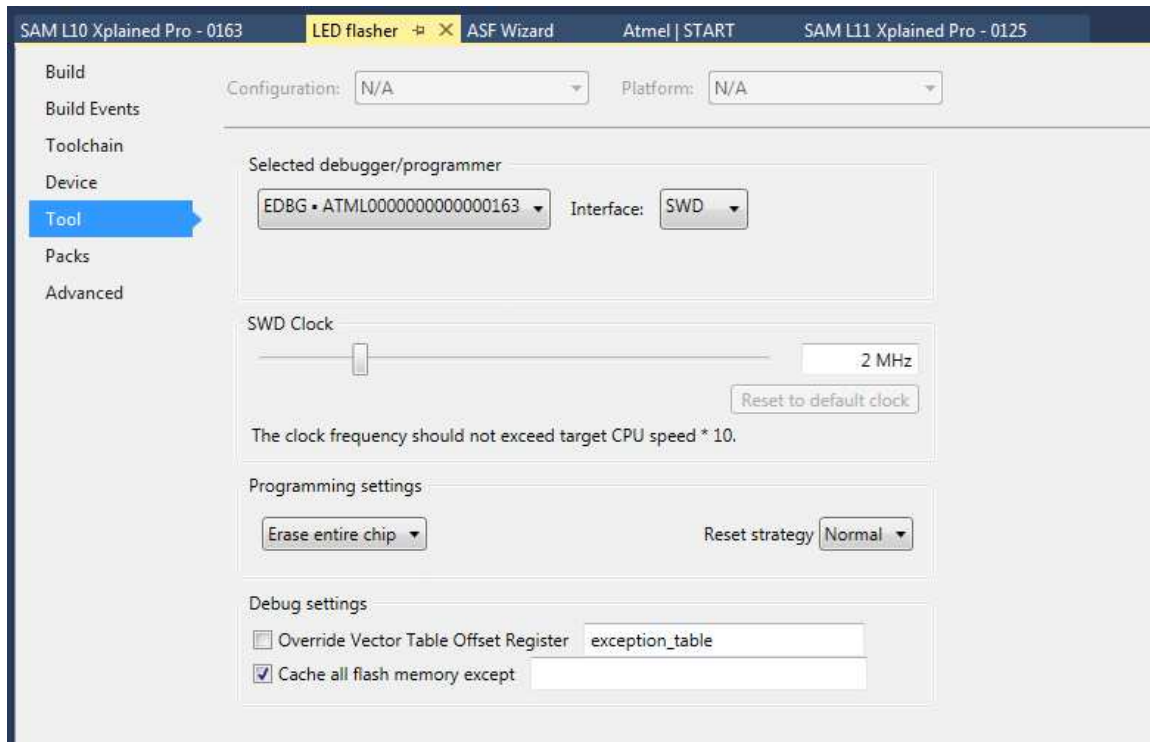
11. Compile and run the LED Flasher application.
    – Build the project: *Build > Build Solution* or <F7>.
    – Load the code into the SAM L10 Xplained Pro and start debugging: *Debug > Start debugging and break* or <ALT+F5>.
    – The application is programmed and the debugger breaks in main.
    – 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
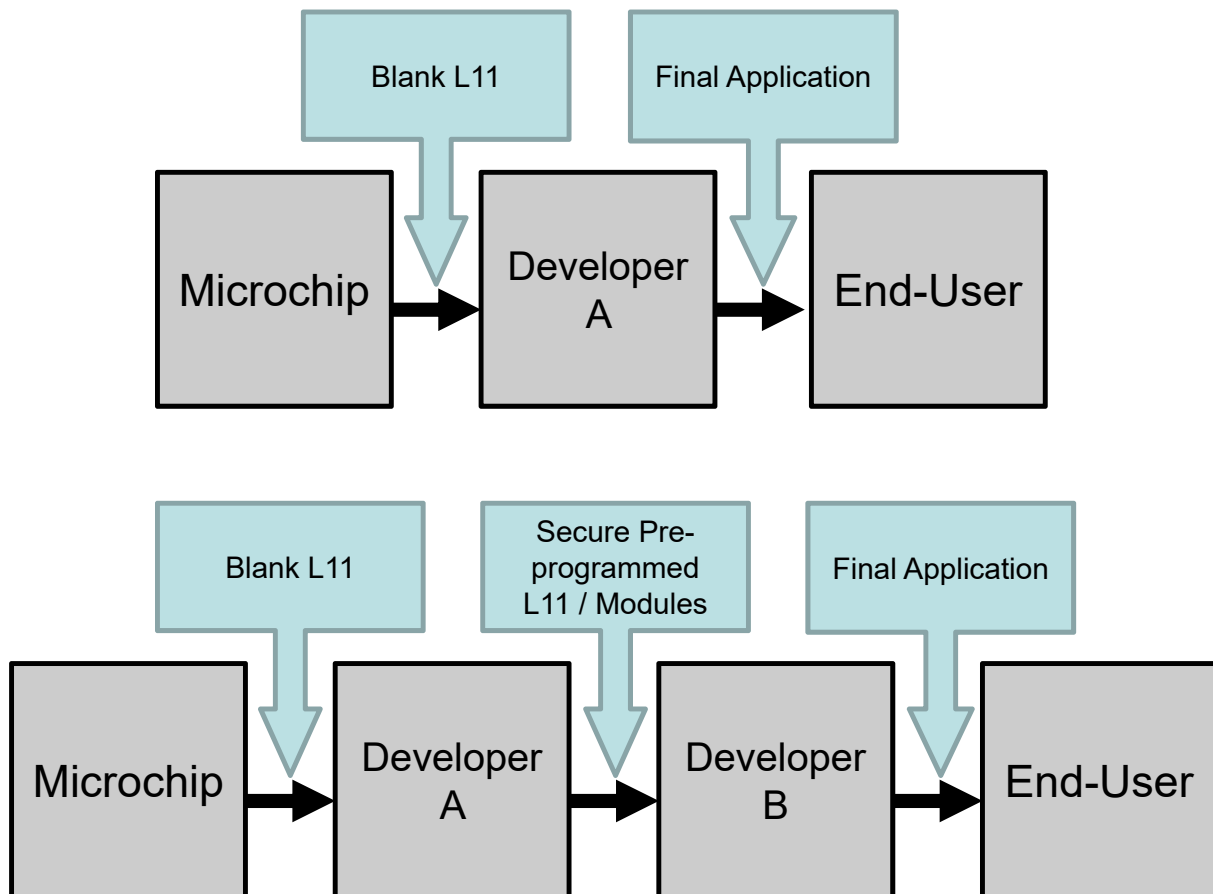
This document describes how to debug a solution composed of the following two projects:

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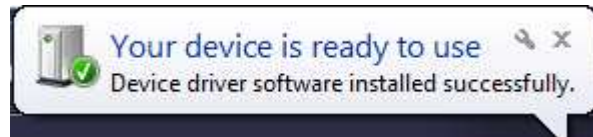
When the Xplained Pro MCU board is powered, the power LED (green) will glow. Atmel Studio will auto-detect the specific Xplained Pro MCU and extension boards that are connected. Atmel Studio will present relevant information, such as data sheets and kit documentation.

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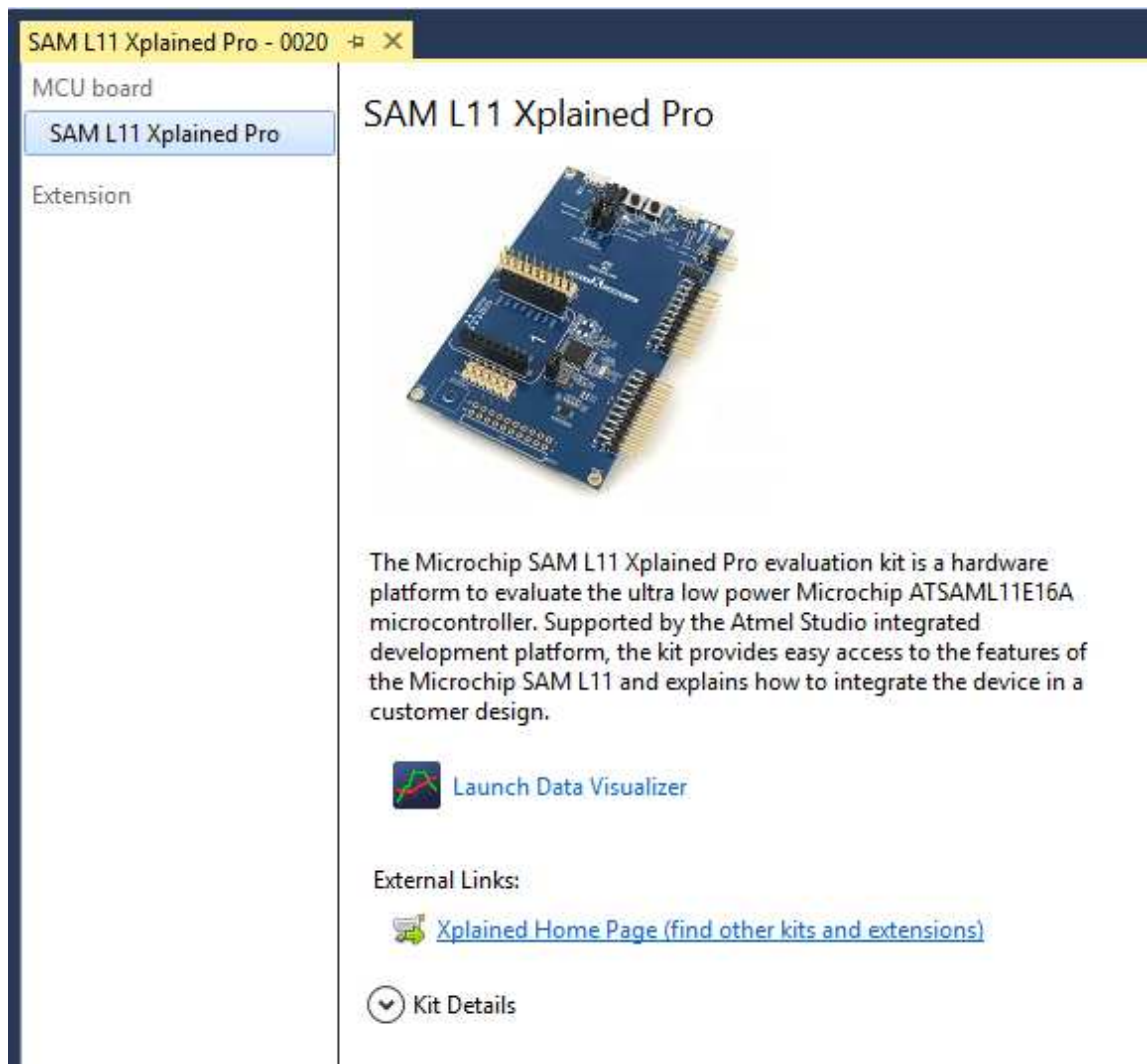
5. Launch Atmel Start from Atmel Studio to Open the Secure Application Project: *File > New > Atmel Start Example Project* and then open Atmel Start into Atmel studio 7.

Figure 5-3. Open a New Atmel Start Example Project in Atmel Studio
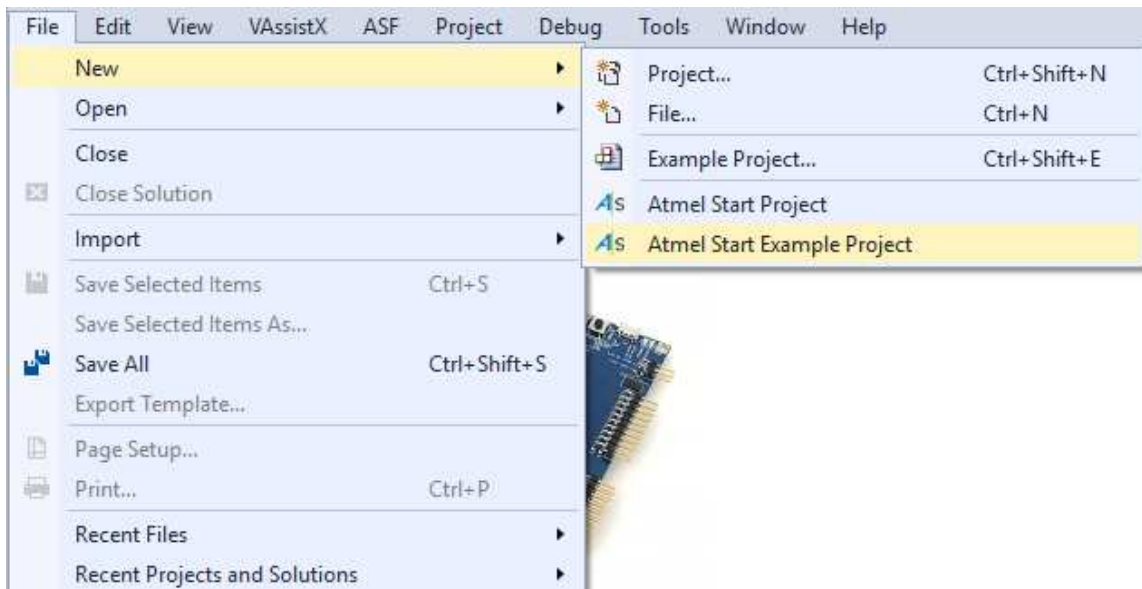
Note: After few seconds the example llist will be displayed.

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

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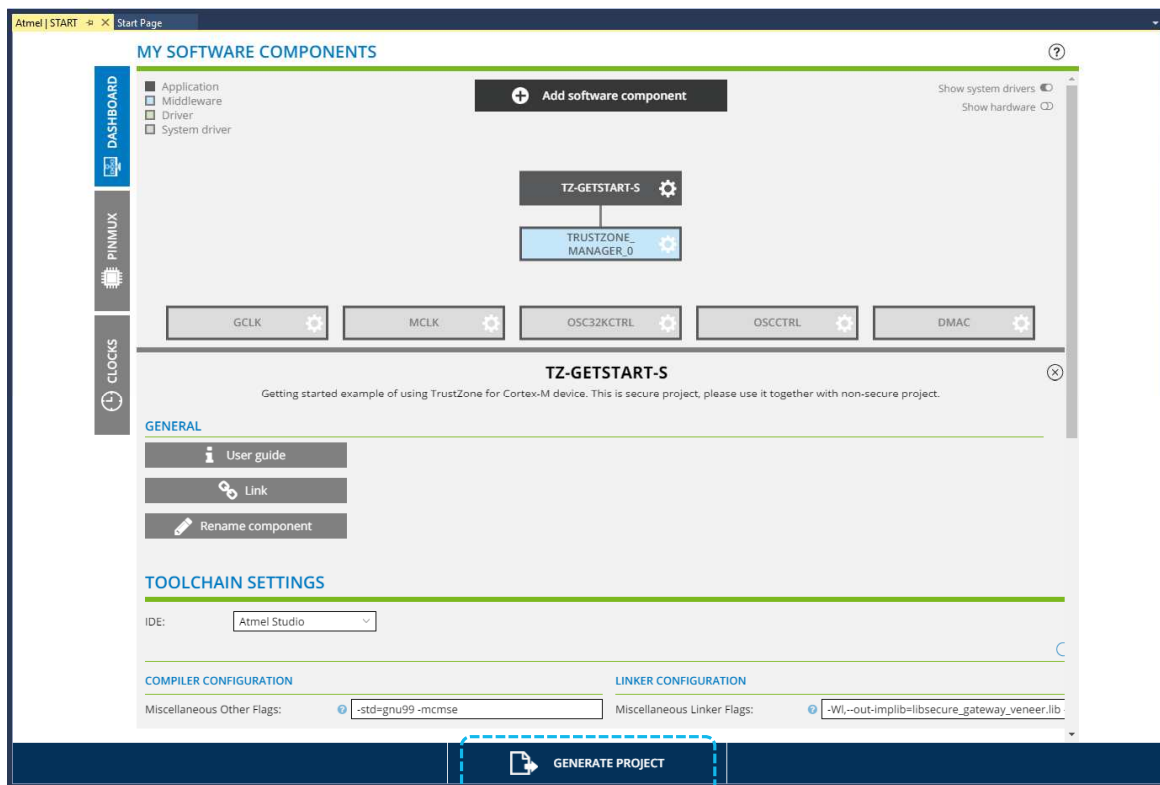
8. The following "New Atmel Start Project" window will be displayed.

Figure 5-6. Secure Project Importation View
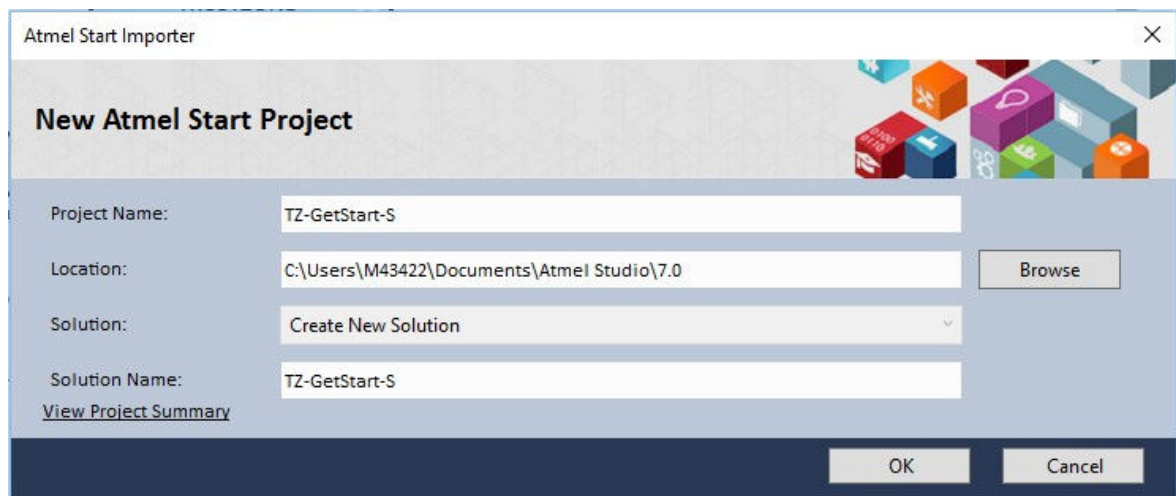
9. Modify the following project information:
   – Enter a new Project Name.
   – Click **Browse** to choose a location.
   – For Solution: Create New Solution.
   – For Solution Name, enter *TrustZoneGettingStarted*.
   – Cick **OK** to open the project in Atmel Studio.

Figure 5-7. Modifying the Project Information
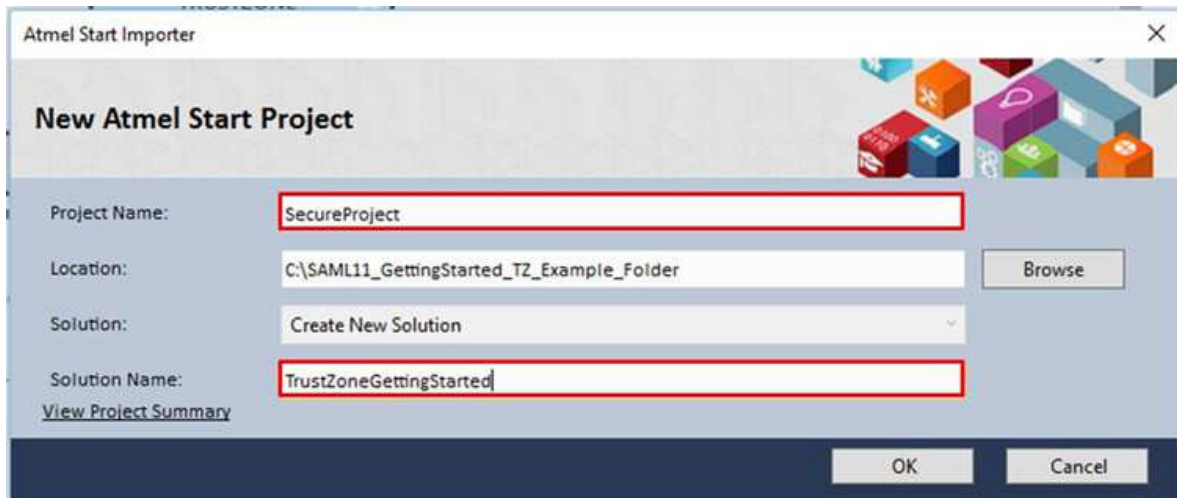
10. To see the project in the Atmel Studio Solution Explorer: *View > Solution Explorer* or <CTRL+ALT +L>.

Figure 5-8. Atmel Studio Solution Explorer Showing Secure Project
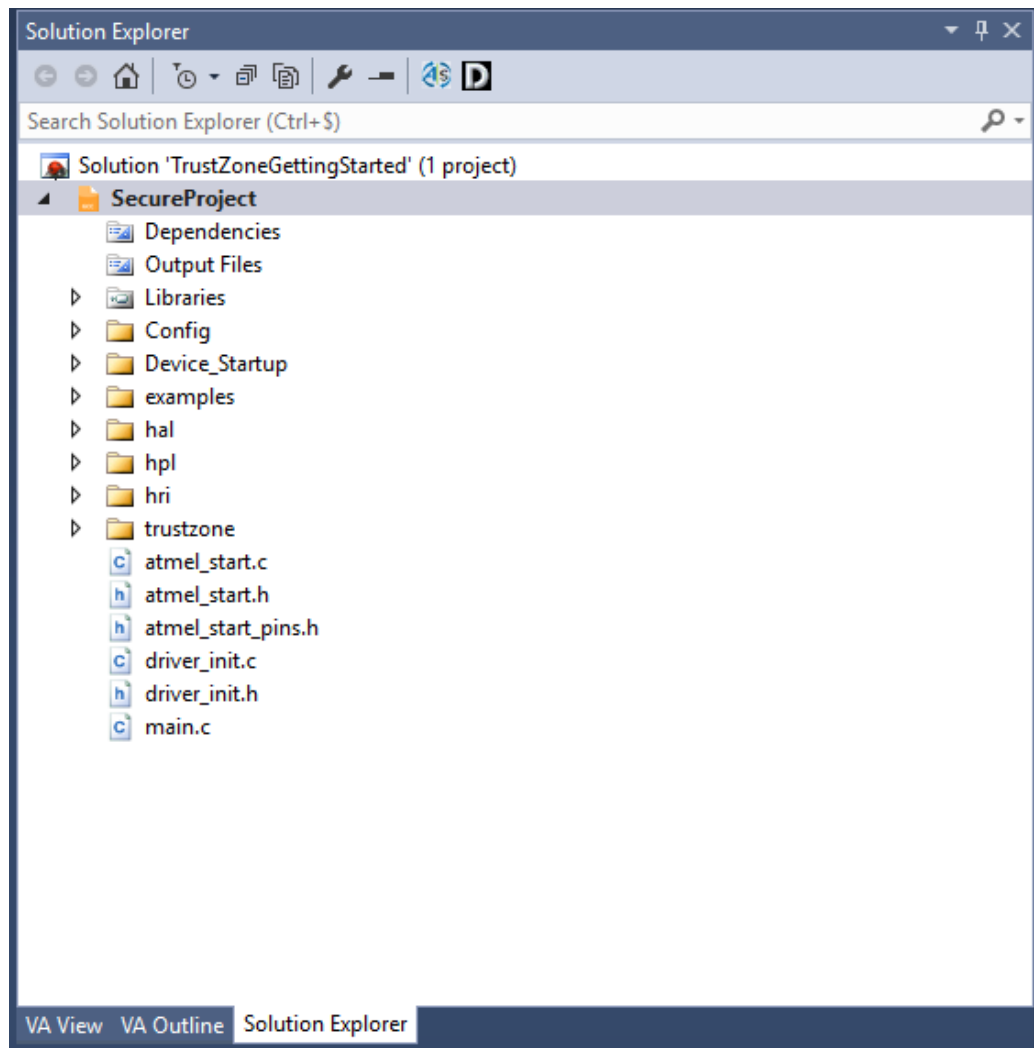
Currently, the TrustZoneGettingStarted solution is only composed of the *SecureProject*.

11. 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

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

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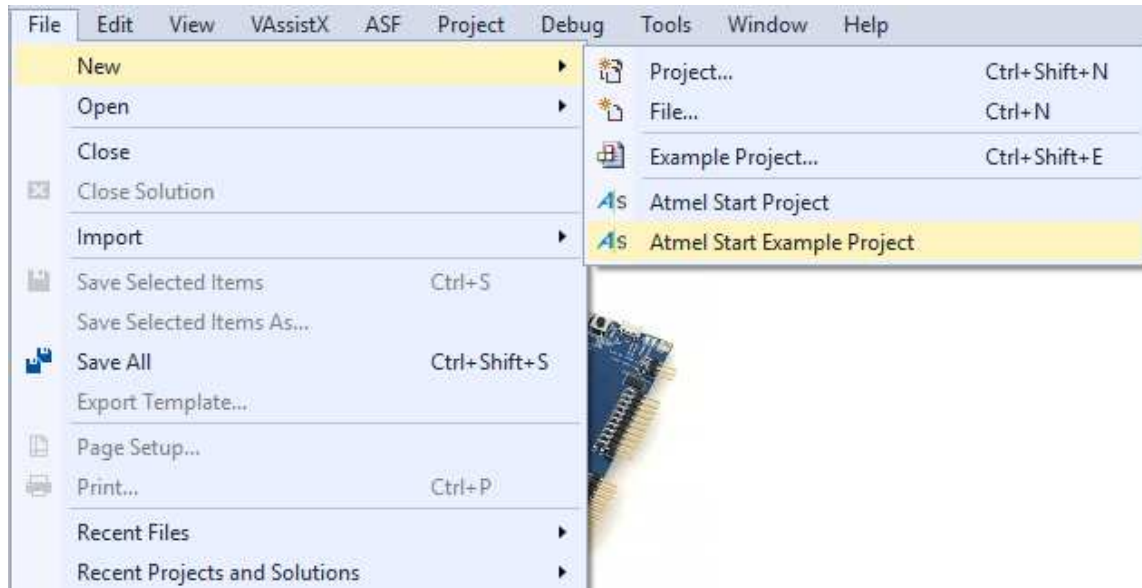
14. Choose the "TrustZone Getting Started Example" for the Non-Secure Project (TZ-GetStart-NS) from the existing examples list, and then click **OPEN SELECTED EXAMPLE.**

Figure 5-12. TrustZone Getting started SAM L11 Non-Secure Project Example Selection
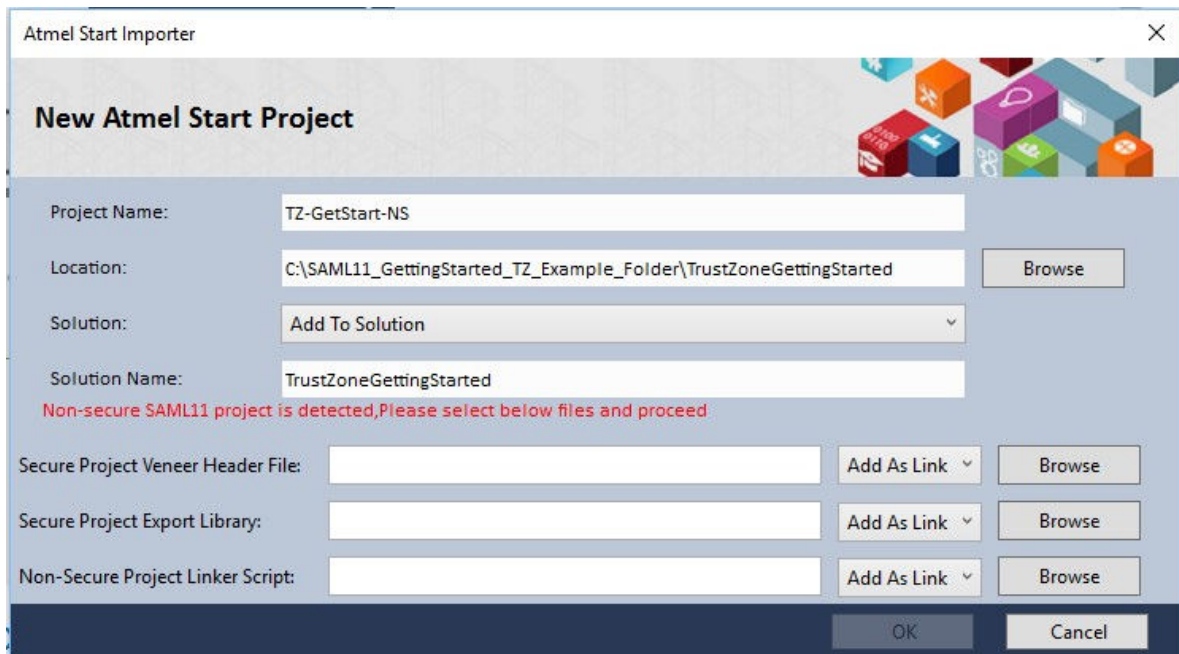
15. Generate the project from Atmel Start to open it in Atmel Studio, and once it is loaded, click **GENERATE PROJECT.**

Figure 5-13. Atmel Start Non-Secure Project Overview and Generation
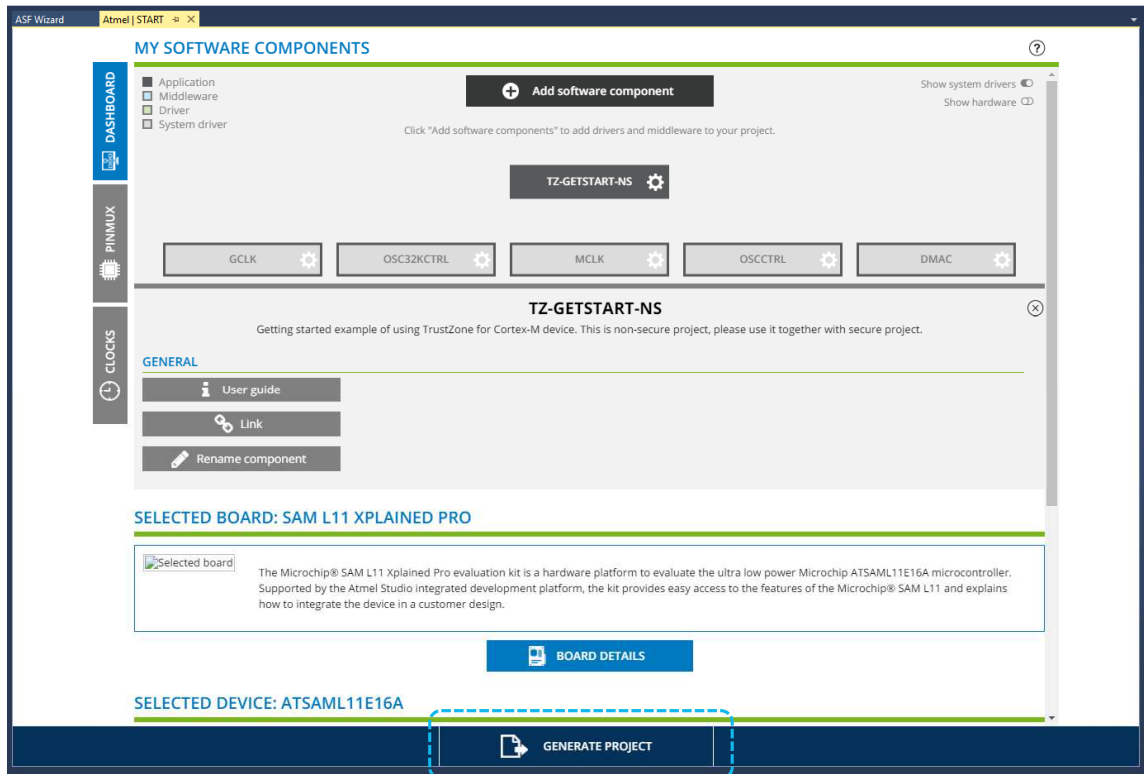
16. The following "New Atmel Start Project " window will be displayed.

Figure 5-14. Non-Secure Project Importation Window