Please note that Cypress is an Infineon Technologies Company.
The document following this cover page is marked as “Cypress” document as this is the company that originally developed the product. Please note that Infineon will continue to offer the product to new and existing customers as part of the Infineon product portfolio.

Continuity of document content
The fact that Infineon offers the following product as part of the Infineon product portfolio does not lead to any changes to this document. Future revisions will occur when appropriate, and any changes will be set out on the document history page.

Continuity of ordering part numbers
Infineon continues to support existing part numbers. Please continue to use the ordering part numbers listed in the datasheet for ordering.

CE220465 – PSoC 6 MCU Cryptography: AES Demonstration

Objective

This code example demonstrates encryption and decryption of data using the Advanced Encryption Scheme (AES) algorithm in PSoC® 6 MCU.

Requirements

Tool: PSoC Creator™ 4.2; Peripheral Driver Library (PDL) 3.1

Programming Language: C (Arm® GCC 5.4.1 and Arm MDK 5.22)

Associated Parts: All PSoC 6 MCU parts

Related Hardware: CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit

Overview

This code example encrypts and decrypts the user input data using the AES algorithm with a 128-bit long key. The encrypted and decrypted data are displayed on a UART terminal emulator.

Hardware Setup

This example uses the kit's default configuration. Refer to the kit guide to ensure the kit is configured correctly.

Software Setup

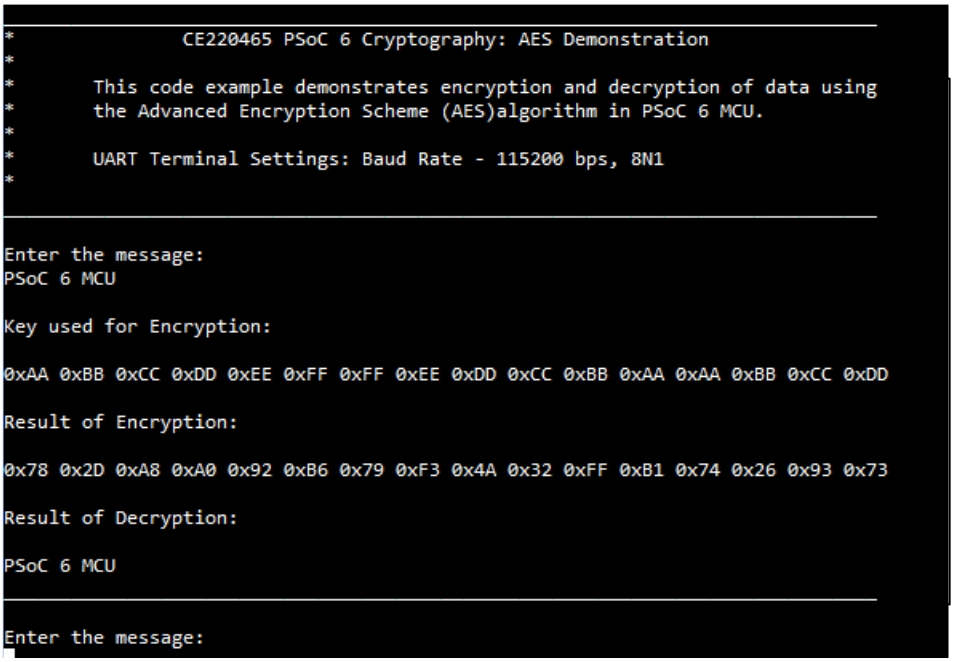
This example uses a terminal emulator to display the encryption and decryption results. The example uses Tera Term.

Operation

1. Plug the CY8CKIT-062 board into your computer's USB port.
2. Build the project and program it into the PSoC 6 MCU device. Choose **Debug** > **Program**. For more information on device programming, see PSoC Creator Help. Flash for both CPUs is programmed in a single program operation.
3. Open your terminal software and select the KitProg COM port. Set the other serial port parameters as follows:
   - Baud rate: 115200 bps
   - Data: 8 bit
   - Parity: None
   - Stop: 1 bit
   - Flow control: None
4. Press the reset button on the kit and enter the message to be encrypted. Note that in this example, the maximum message size is restricted to 100 characters. If you need to increase the message size change the macro `MAX_MESSAGE_SIZE` to the message size that you require.
5. Observe the results in the terminal window.

Figure 1 shows a sample output as displayed on Tera Term UART terminal.

Figure 1. Sample Output showing AES Encryption



The sections that follow discuss the Components, parameter settings, and resources used to make the example.

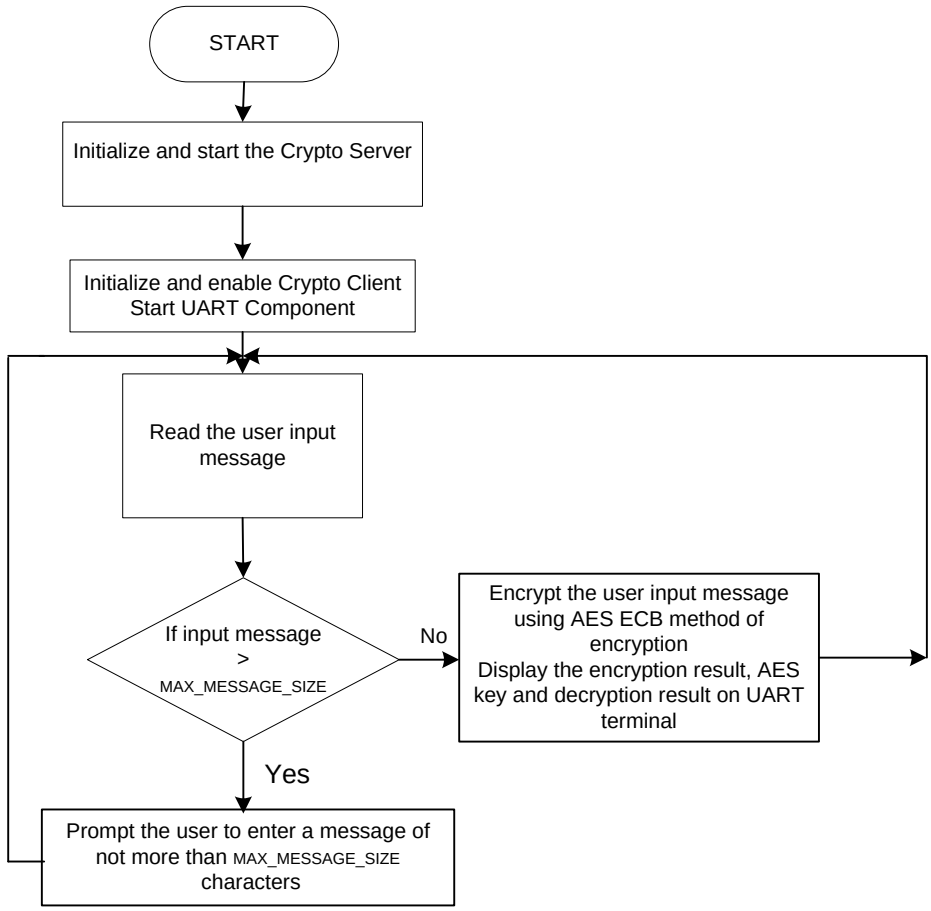
Design and Implementation

AES is a symmetric block cipher data encryption algorithm which means that it uses the same key for encryption and decryption of data. The AES operation works on 128-bit block size and uses keys of 128 bits, 192 bits, or 256 bits of length.

Cryptographic operation implemented in this example is based on a Client-Server model. The firmware initializes and starts the Crypto server. Access to the server is through the Inter-Processor Communication (IPC) driver. The firmware initializes and starts the client. The firmware then provides the configuration data required for AES encryption technique and requests the crypto server to run the cryptographic operation.

In this example, the user input message is read from the UART terminal and encrypted using the AES algorithm with a key length of 128 bits. The 128-bit encrypted data is displayed on the UART terminal. Then, you can view the decrypted message on the UART terminal and verify that the decryption operation produces the same original encrypted message. Figure 2 shows the firmware flowchart.

Figure 2. Firmware Flow



Components and Settings

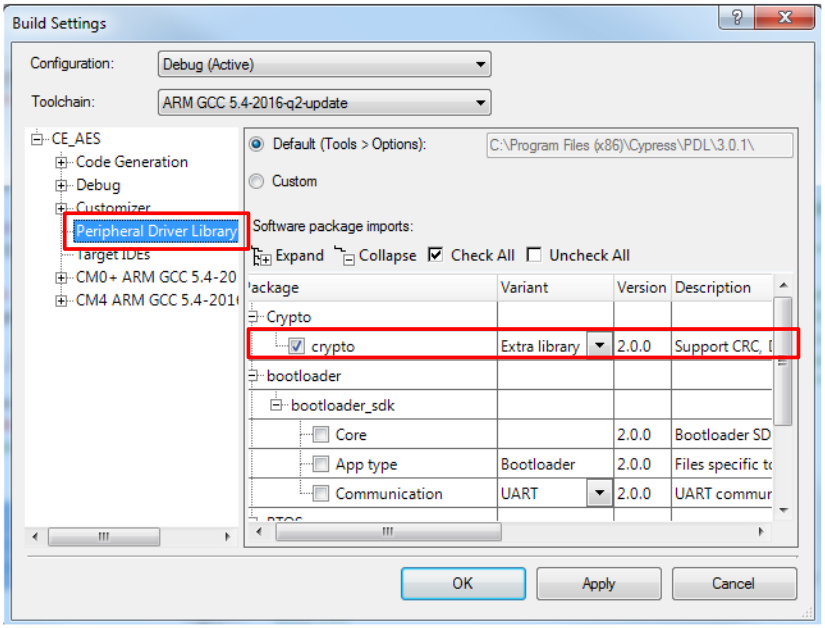
Table 1 lists the PSoC Creator Components used in this example, how they are used in the design, and the non-default settings required so they function as intended.

Table 1. PSoC Creator Components

Component	Instance Name	Purpose	Non-default Settings
UART	KIT_UART	Send to and receive data from the UART Terminal	Baud Rate set to 115200 bps

To use the Crypto block of PSoC 6 MCU in your design, the Crypto driver must be enabled. To enable the drivers, check the crypto option under **Project** > **Build Settings** > **Peripheral Driver Library** as shown in Figure 3.

Figure 3. Enabling Crypto PDL Drivers



Reusing This Example

This example is designed for the CY8CKIT-062-BLE Pioneer Kit. To port the design to a different PSoC 6 MCU device and/or kit, change the target device using Device Selector and update the pin assignments in the Design-Wide Resources Pins settings as needed.

Table 2. Device and Pin Mapping Table Across PSoC 6 MCU Kits

Kit Name	Device Used	KIT_UART_RX	KIT_UART_TX
CY8CKIT-062-BLE	CY8C6347BZI-BLD53	P5[0]	P5[1]
CY8CKIT-062-WiFi-BT	CY8C6247BZI-D54	P5[0]	P5[1]

In some cases, a resource used by a code example (for example, an IP block) is not supported on another device. In that case, the example will not work. If you build the code targeted at such a device, you will get errors. See the device datasheet for information on what a particular device supports.

Related Documents

For a comprehensive list of PSoC 6 MCU resources, see KBA223067 in the Cypress community.

Application Notes	
AN210781 – Getting Started with PSoC 6 MCU with Bluetooth Low Energy (BLE) Connectivity	Describes PSoC 6 MCU with BLE Connectivity devices and how to build your first PSoC Creator project
AN215656 – PSoC 6 MCU: Dual-CPU System Design	Describes the dual-CPU architecture in PSoC 6 MCU, and shows how to build a simple dual-CPU design
AN219434 – Importing PSoC Creator Code into an IDE for a PSoC 6 MCU Project	Describes how to import the code generated by PSoC Creator into your preferred IDE
PSoC Creator Component Datasheets	
UART	Supports standard UART interface
Device Documentation	
PSoC 6 MCU Datasheets	PSoC 6 Technical Reference Manuals
Development Kit Documentation	
CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit	
CY8CKIT-062-WiFi-BT PSoC 6 WiFi-BT Pioneer Kit	
CY8CPROTO-062-4343W PSoC 6 Wi-Fi BT Prototyping Kit	
CY8CPROTO-063 BLE PSoC 6 BLE Prototyping Kit	
Tool Documentation	
PSoC Creator	Look in the downloads tab for Quick Start and User Guides
Peripheral Driver Library (PDL)	Get the latest version for use with PSoC Creator. Look in the *<PDL install folder>/doc* for the User Guide and the API Reference

Document History

Document Title: CE220465 - PSoC 6 MCU Cryptography: AES Demonstration

Document Number: 002-20465

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	5846203	VKVK	08/02/2017	New code example
*A	5887785	VKVK	9/18/2017	Initial public release version
*B	6002370	VKVK	12/22/2017	Updated for PSoC Creator 4.2
*C	6513780	VKVK	03/19/2019	Updated for PDL 3.1.0

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

Arm® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6 MCU

Cypress Developer Community

Community | Code Examples | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709