M487 使用 AES 和 SPROM 安全 ISP 升級

NuMicro® 32 位系列微控制器範例代碼介紹

文件資訊

應用簡述	本範例代碼使用 M487 通過 ISP Programming tool 下載加密韌體後，使用 AES crypto 還原韌體內容並進行更新，實現安全下載。
BSP 版本	M480_BSP_CMSIS_V3.06.000
開發平台	NuMaker-PFM-M487 V3.0

The information described in this document is the exclusive intellectual property of Nuvoton Technology Corporation and shall not be reproduced without permission from Nuvoton.

Nuvoton is providing this document only for reference purposes of NuMicro microcontroller and microprocessor based system design. Nuvoton assumes no responsibility for errors or omissions.

All data and specifications are subject to change without notice.

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com

1. 概述

本範例代碼提供使用者使用 UART 介面進行韌體更新時，可先通過新唐 CryptoTool 進行韌體加密，再通過新唐 NuMicro_ISP_Programming_Tool 將加密後的韌體寫入待更新的 NuMaker-PFM-M487 開發板。M487 在確認接收完成後會調用 SPROM 內部的解密 API，搭配硬體加密加速器，以及儲存在 SPROM 的 AES 密鑰將加密韌體進行解密，而後將韌體寫入 APROM 完成更新流程。

1.1 原理

韌體加密採用 Advanced Encryption Standard (AES) 進行，為一種對稱式加解密算法，用作加解密的密鑰為同一把，長度可為 128、192 或者 256 bits。方式為將要加密的資料分為數個固定大小的區塊進行加密，其優點為安全性高、速度快，為美國國家標準與技術研究院(NIST)所發布。

目前 NIST 針對 AES 算法提供了多種模式，新唐 CryptoTool 採用 AES-CFB-256 ，在進行檔案加解密前需設定 256 bits 的密鑰，以及 128 bits 的 Initialization Vector (IV)，並指定需做加密檔案的位置，以及加密完檔案的存放路徑及檔名，可參考如下:

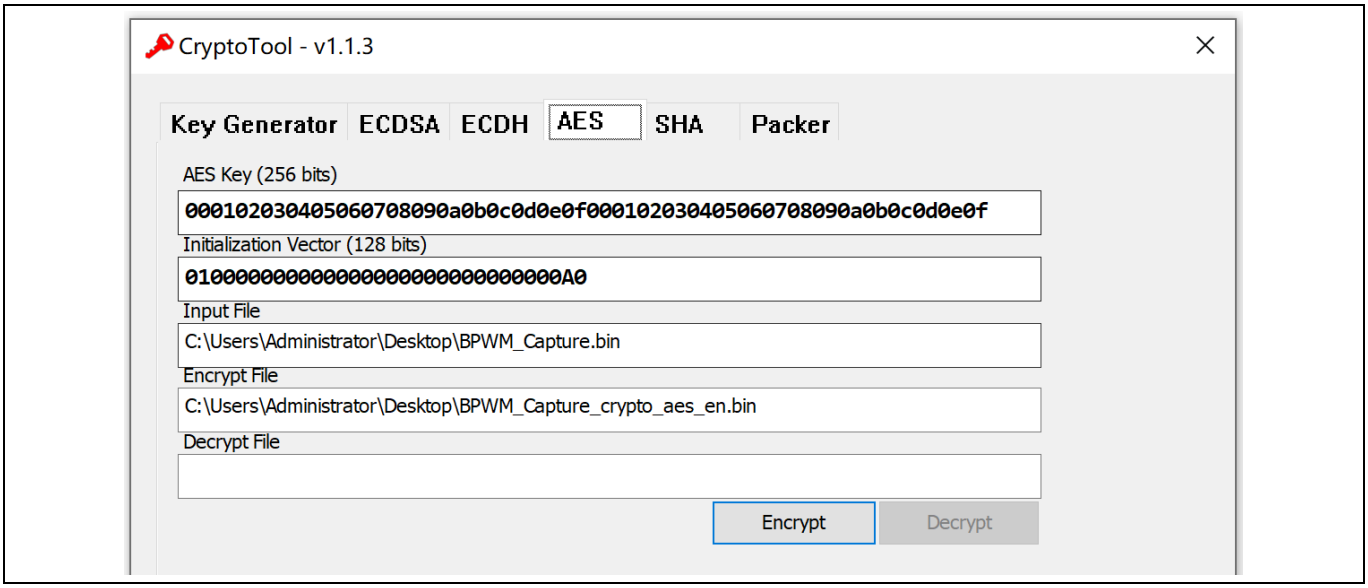


圖 1-1 Nuvoton CryptoTool -v1.1.3

1.2 Security Protection Memory (SPROM)

M487 可將內部 SPROM 配置成獨立的安全區域，該區域儲存的指令(程式碼)無法被讀取只能執行，最多可配置 4K 位元組的安全空間，用戶可將機密程式碼放在該區域，防止外部程式意外讀取資料，以及洩漏客戶軟體資訊，詳細說明可參考 M480 Series Technical Reference Manual 章節 6.5.4.4。範例中我們將解密韌體所用到的 API 以及 AES 加解密所使用的 IV 放在該區域內。

1.3 執行結果

1.3.1 ISP Bootloader 燒錄到 MCU

本範例代碼執行編譯後，會生成兩個 bin 檔如下圖：

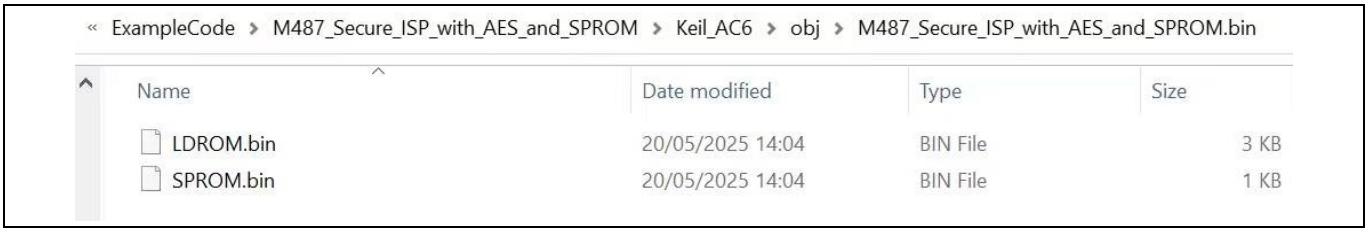


圖 1-2 編譯後生成的 bin 檔

燒錄時使用 NuMicro ICP Programming Tool 軟體進行燒錄，如圖 1-3 ，LDROM 選擇 “LDROM.bin” ，SPROM 選擇 “SPROM.bin” 。

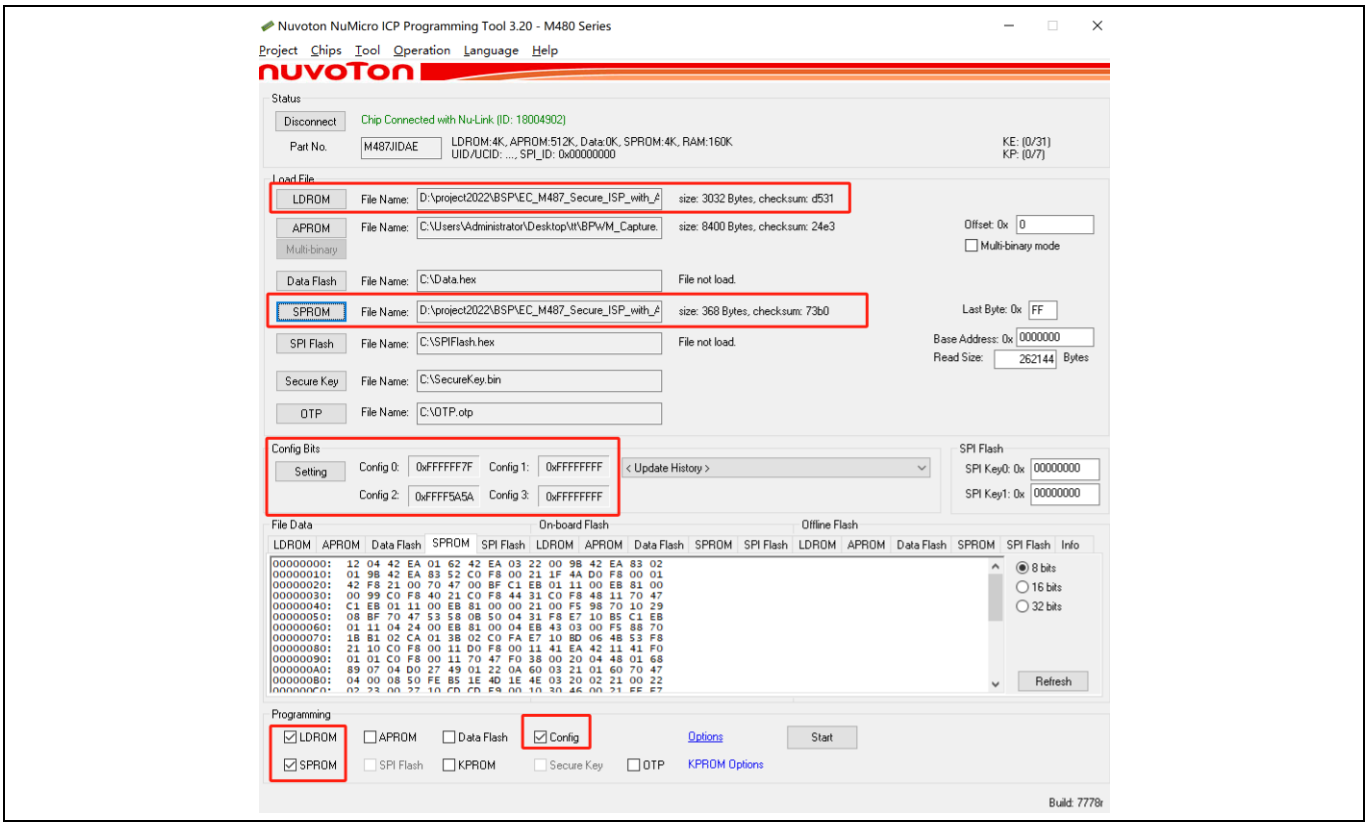


圖 1-3 ICP 配置

Chip Setting 選擇如下圖：

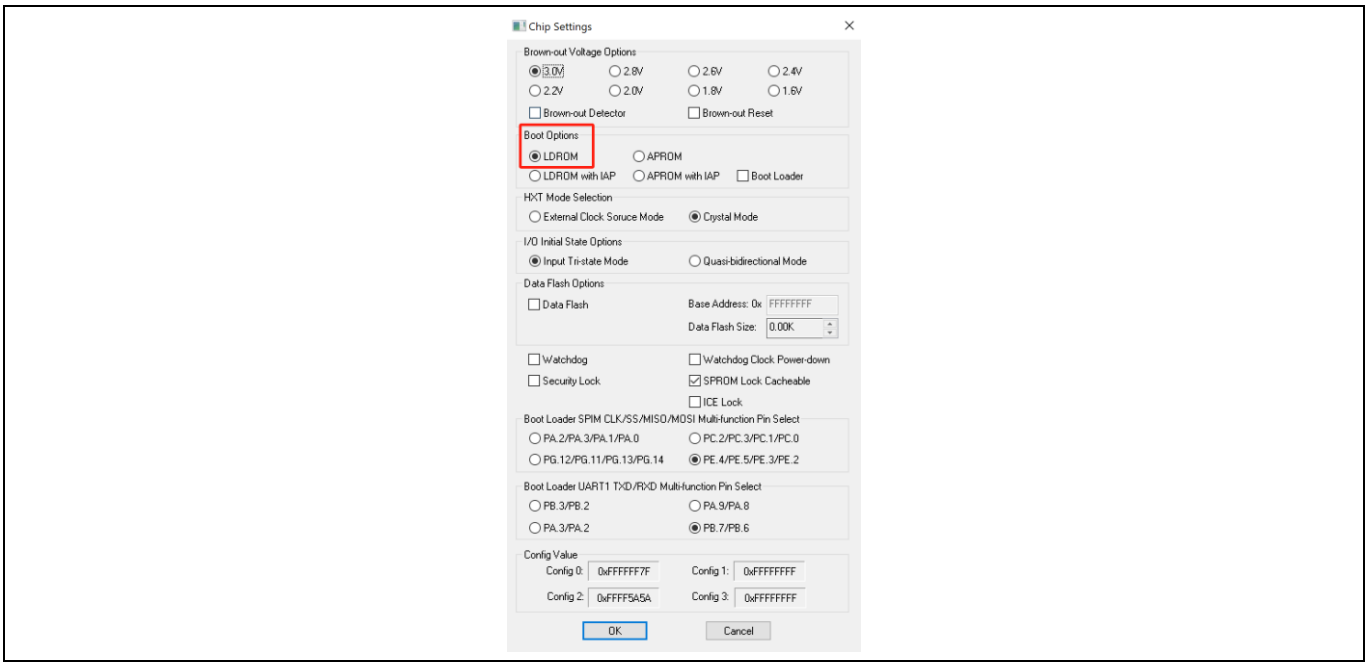


圖 1-4 Chip Setting

1.3.2 生成加密 bin 檔

燒錄後按下圖 1-5 使用新唐 CryptoTool 工具加密程式，這裡選擇 AES 欄位，設定 AES key 以及 Initialization Vector。編譯 M480 BSP 內任意一範例程式（這裡以 Template 程式為例），並將編譯完成的 bin 檔複製到 CryptoTool 底下。設定 Input File 及 Encrypt File 的路徑，按下 **Encrypt** 按鈕，幾秒後會在該路徑下看到生成的加密 bin 檔。

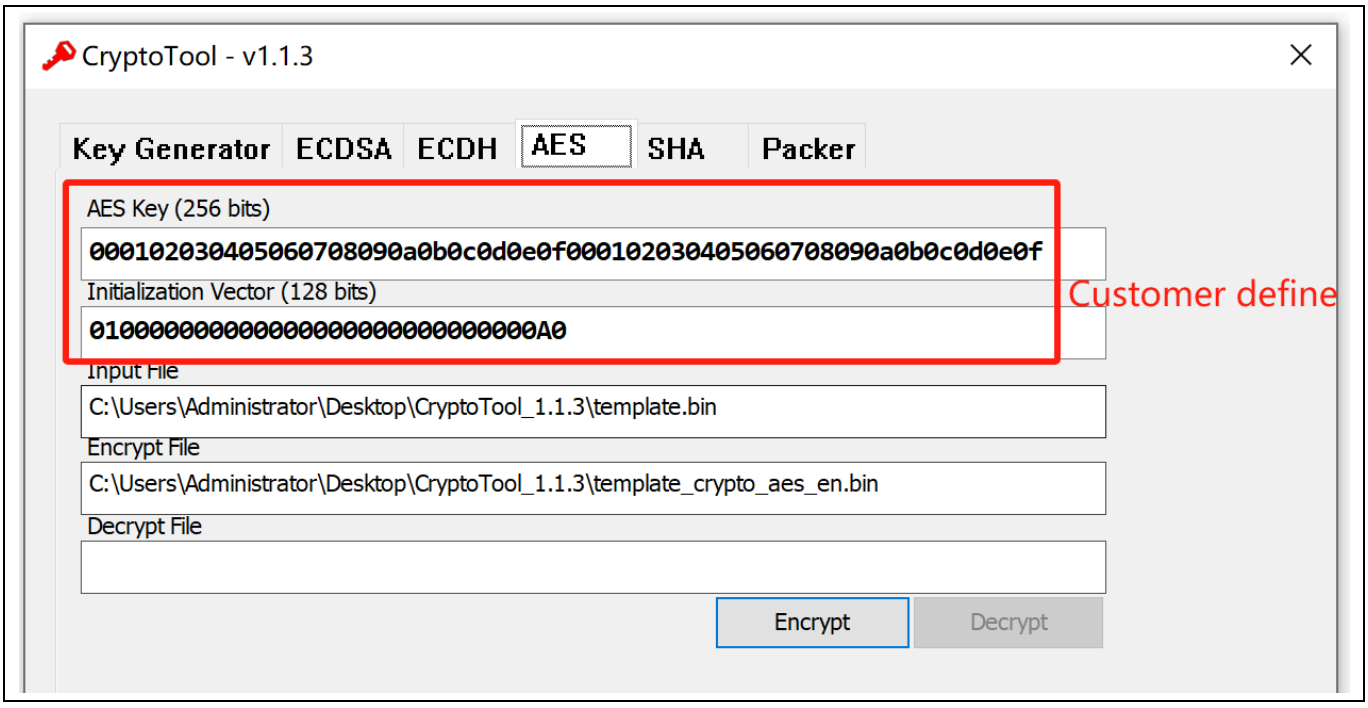


圖 1-5 檔案加密

1.3.3 執行 ISP 更新 APROM

接著打開新唐 ISP 軟體，點擊 UART 選擇對應串口，點擊 “**Connect**” ，開發板上按 reset 按鍵，之後會連上新唐 ISP，如下圖 1-6，連上後會顯示晶片型號，這時選擇 APROM 載入剛剛加密後的 “*template_crypto_aes_en.bin*” 檔，點擊 “**Start**” 進行升級。

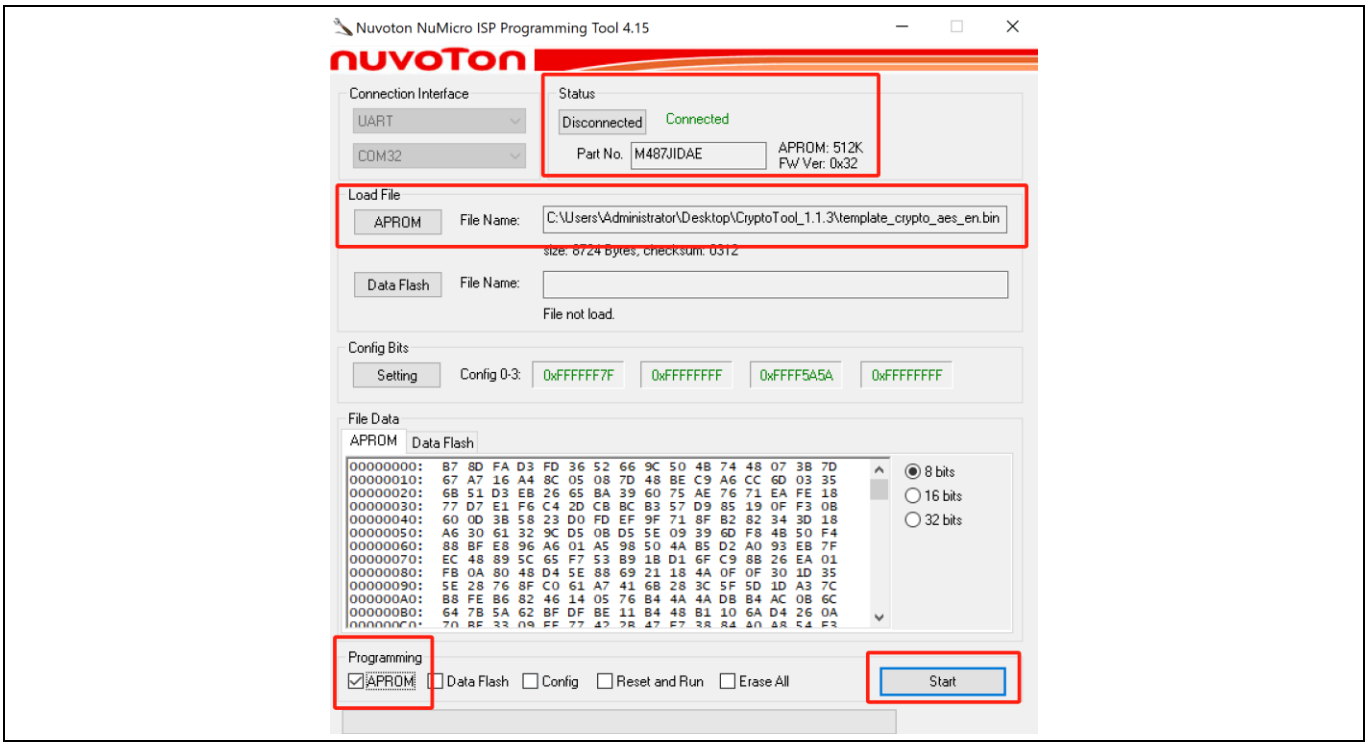


圖 1-6 ISP 升級介面

升級後關閉 ISP 軟體，打開串口工具，選擇相應列印串口，在開發板上按 reset 按鍵，可以看到 Template 程式列印的 “Hello World” 字樣，如下圖 1-7，說明 ISP 燒錄的加密後的 “*template_crypto_aes_en.bin*” 文檔已解密完成並燒錄到 APROM 中並且正確運行。

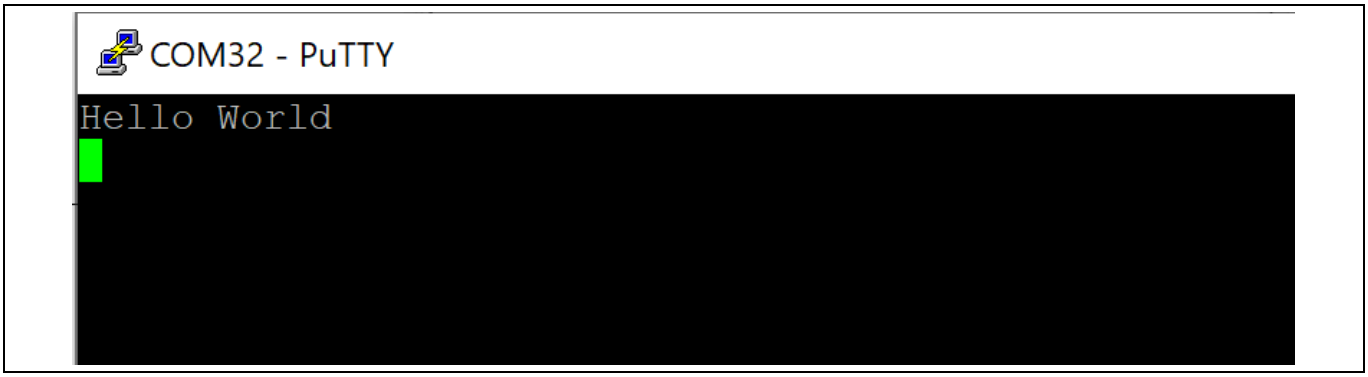


圖 1-7 Template 程式列印結果

2. 代碼介紹

以下 main()程式實現了範例功能的架構，前半段通過 UART 以及 ISP 協議，將加密的韌體寫入到 SRAM 區域，後半部在 void UpdateRoutine(void)實現了安全更新的細節，通過 SPROM 進行保護，讓駭客無法得知詳細流程指令。Secure ISP 程式提供給客戶一個安全更新的基本範例，客戶可根據其具體需求，改寫 UpdateRoutine 的流程，如調整 AES 加解密模式以及 IV 值，藉此來完善屬於產品本身的更新機制。需注意 M487 的 SRAM 只有 160K 位元組，所以該範例能夠更新的 APROM（bin 檔）大小是有限制的。

```
int main(void)
{
    uint8_t u8SysTickCnt = 0;

    /* Unlock protected registers */
    SYS_UnlockReg();
    /* Init System, peripheral clock and multi-function I/O */
    SYS_Init();
    /* Init UART to 115200-8n1 */
    UART_Init();

    CLK->AHBCLK |= CLK_AHBCLK_ISPCKEN_Msk;
    FMC->ISPCTL |= FMC_ISPCTL_ISPEN_Msk;
    g_apromSize = GetApromSize();
    GetDataFlashInfo(&g_dataFlashAddr, &g_dataFlashSize);
     SYS_UnlockReg();                   /* Unlock protected registers */
    NVIC_EnableIRQ(CRPT_IRQn);
    AES_ENABLE_INT(CRPT);

    SysTick->LOAD = 0xFFFFFF;
    SysTick->VAL  = (0x00);
    SysTick->CTRL = SysTick->CTRL | SysTick_CTRL_CLKSOURCE_Msk
                    | SysTick_CTRL_ENABLE_Msk; /*using cpu clock*/

    g_u8ISPDone = FALSE;
    while (1)
    {

        if ((bufhead >= 4) || (bUartDataReady == TRUE))
        {
            uint32_t lcmd;
            lcmd = inpw(uart_rcvbuf);

            if (lcmd == CMD_CONNECT)
            {
                goto _ISP;
            }
            else
            {
                bUartDataReady = FALSE;
                bufhead = 0;
```

```
            }
        }

        if (SysTick->CTRL & SysTick_CTRL_COUNTFLAG_Msk) /*timeout, then goto APROM*/
        {
            SysTick->VAL = 0;
            u8SysTickCnt++;
            if (u8SysTickCnt >= 5)
            {
                goto _APROM;
            }
        }
    }

_ISP:

    while (1)
    {
        if (bUartDataReady == TRUE)
        {
            bUartDataReady = FALSE;
            ParseCmd(uart_rcvbuf, 64);
            PutString();
        }
       if (g_u8ISPDone == TRUE)
        {
            UpdateRoutine();
            g_u8ISPDone = FALSE;
        }

    }

_APROM:
    SYS->RSTSTS = (SYS_RSTSTS_PORF_Msk | SYS_RSTSTS_PINRF_Msk);
    FMC->ISPCTL &= ~(FMC_ISPCTL_ISPEN_Msk | FMC_ISPCTL_BS_Msk);
    SCB->AIRCR = (V6M_AIRCR_VECTKEY_DATA | V6M_AIRCR_SYSRESETREQ);

    /* Trap the CPU */
    while (1);
}
```

以下 UpdateRoutine() 程式實現了 AES 解密檔案，核心是要實現 AES engine 做加解密時密鑰的寫入及功能測試，其中關鍵的 AES key 透過 au32MyAESKey 陣列實現，使用者在熟悉操作後可自由替換成實際產品所使用的密鑰。AES 密鑰屬性，建議設定為加密後的 SPROM 常數，這樣可確保該密鑰只能被 AES engine 所取用，藉此達到完整的保護。

```
void UpdateRoutine(void)
{

    uint32_t StartAddress = 0x0000,  TotalLen;
    TotalLen = g_au32Encryptedimage[0];
```

```
    /*---------------------------------------
    *  step 1, decrypt image
    *  AES-256 CFB mode decrypt fw image
    *---------------------------------------*/
    AES_Open(CRPT, 0, 0, AES_MODE_CFB, AES_KEY_SIZE_256, AES_IN_OUT_SWAP);
    /* Use key in key store */
    AES_SetKey(CRPT, 0, (uint32_t *)au32MyAESKey, AES_KEY_SIZE_256);
    AES_SetInitVect(CRPT, 0, (uint32_t*)au32MyAESIV);
    AES_SetDMATransfer(CRPT, 0, (uint32_t)&g_au32Encryptedimage[1],
                       (uint32_t)au8OutputData, TotalLen);
     g_AES_done = 0;
    AES_Start(CRPT, 0, CRYPTO_DMA_ONE_SHOT);
    while (!g_AES_done);

    /* step 2, update APROM */
    FMC_ENABLE_AP_UPDATE();
     EraseAP(FMC_APROM_BASE,g_apromSize); /* erase APROM */
    WriteData(StartAddress, StartAddress + TotalLen, (uint32_t *)au8OutputData);

    return;
}
```

3. 軟體與硬體需求

3.1 軟體需求

- BSP 版本

  - M480_BSP_CMSIS_V3.06.000

- IDE 版本

  - Keil uVersion 5.41

3.2 硬體需求

- 電路元件

  - NuMaker-PFM-M487 V3.0

- 線路示意圖

  - 將 UART0 的 pin 腳透過 ICE 連接到 PC UART，以連接 ISP 和顯示範例代碼的執行結果。

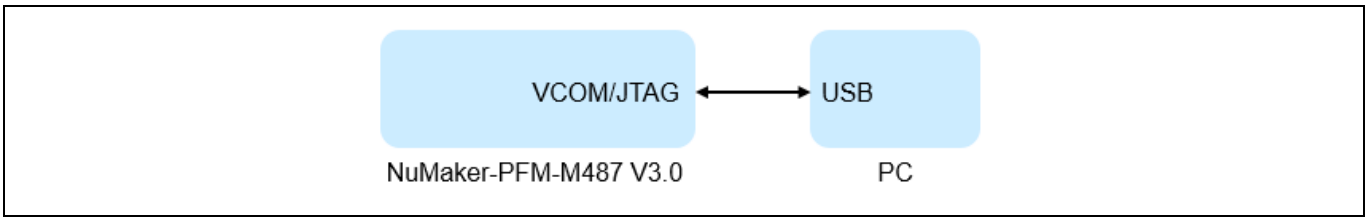


圖 3-1 線路示意圖

4. 目錄資訊

目錄結構如下圖所示。

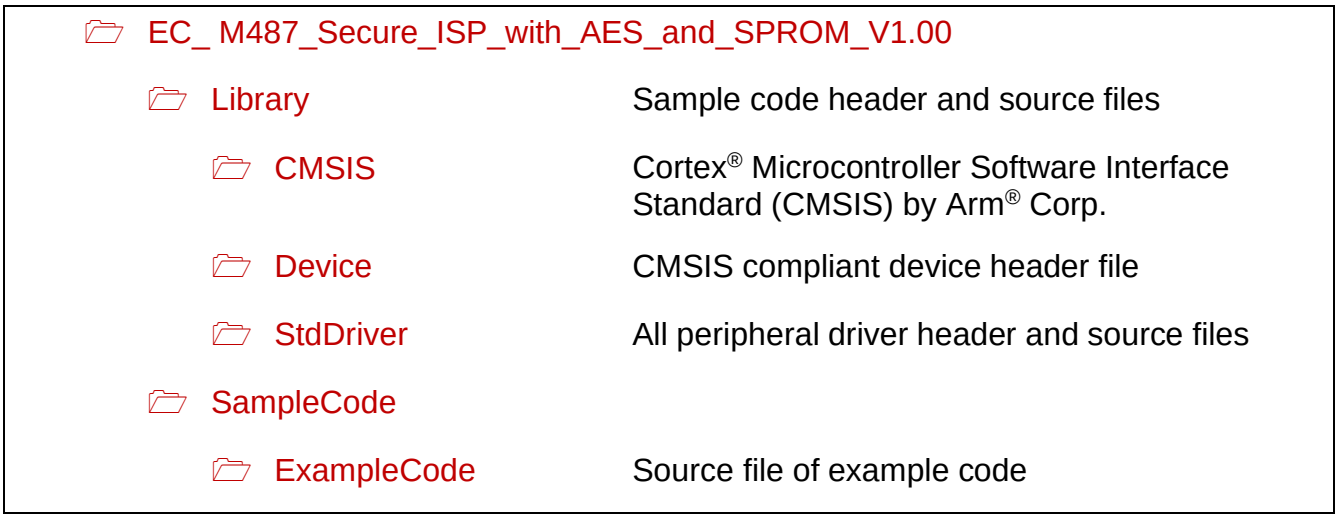


圖 4-1 目錄資訊

5. 範例程式執行

1. 根據目錄資訊章節進入 ExampleCode 路徑中的 KEIL 資料夾，雙擊 *M487_Secure_ISP_with_AES_and_SPROM.uvprojx*。
2. 進入編譯模式介面
   - 編譯
   - 下載代碼至記憶體
   - 進入 / 離開除錯模式
3. 進入除錯模式介面
   - 執行代碼

6. 修訂紀錄

Date	Revision	Description
2025.04.24	1.00	初始發布。

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.

Please note that all data and specifications are subject to change without notice.
All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.