M480 CRYPTO AES in SPROM

Example Code Introduction for 32-bit NuMicro® Family

Information

Application	The example code is to perform AES encryption and decryption, storing the private key of AES in SPROM.
BSP Version	M480 Series BSP CMSIS V3.03.001
Hardware	NuMaker-PFM-M487 V3.0

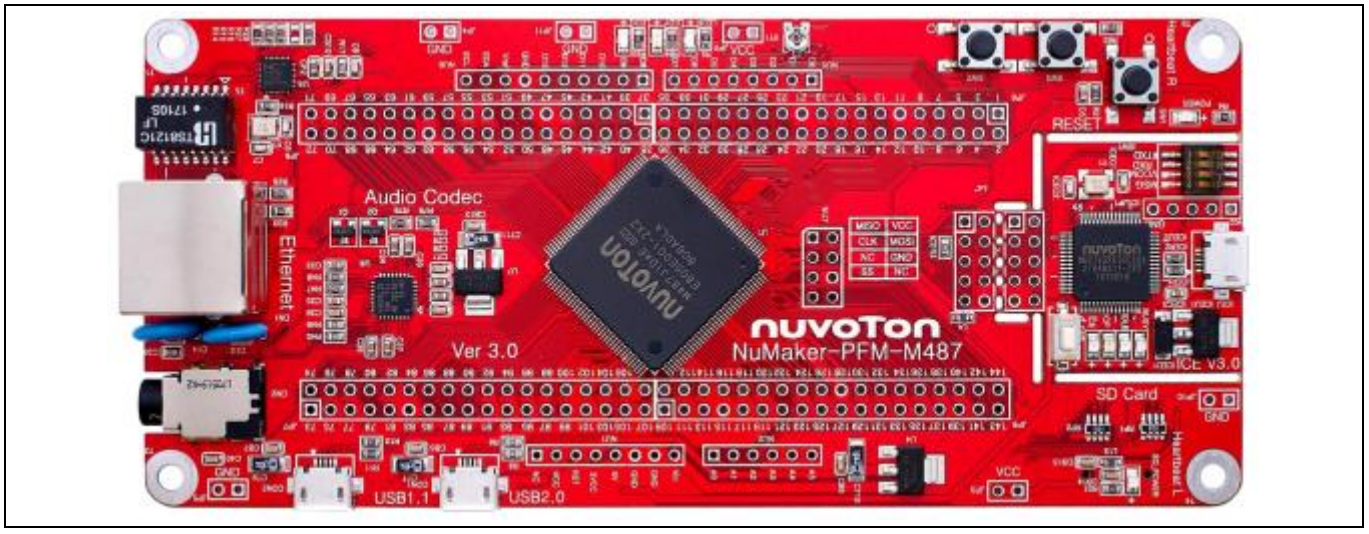
1 Function Description

1.1 Introduction

M480 series have designed hardware CRYPTO and SPROM for intellectual property protection. The Crypto (Cryptographic Accelerator) includes a secure pseudo random number generator (PRNG) core and supports AES, DES/TDES, SHA and HMAC algorithms. The security protection memory (SPROM) is used to store instructions for security application. The purpose of this example code is to perform AES encryption and decryption, storing the private key of AES in SPROM on the NuMaker-PFM-M487 V3.0 board.

For more information about the NuMaker-PFM-M487 board, please visit Nuvoton website to download the UM_NuMaker-PFM_M487_User_Manual:

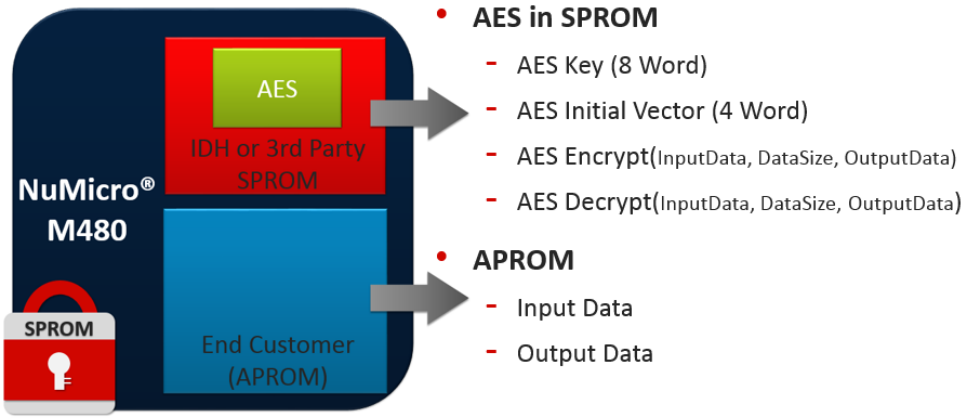
https://www.nuvoton.com/hq/products/microcontrollers/arm-cortex-m4-mcus/User-Manual/?__locale=en&resourcePage=Y&category=&pageIndex=3



NuMaker-PFM-M487 Board

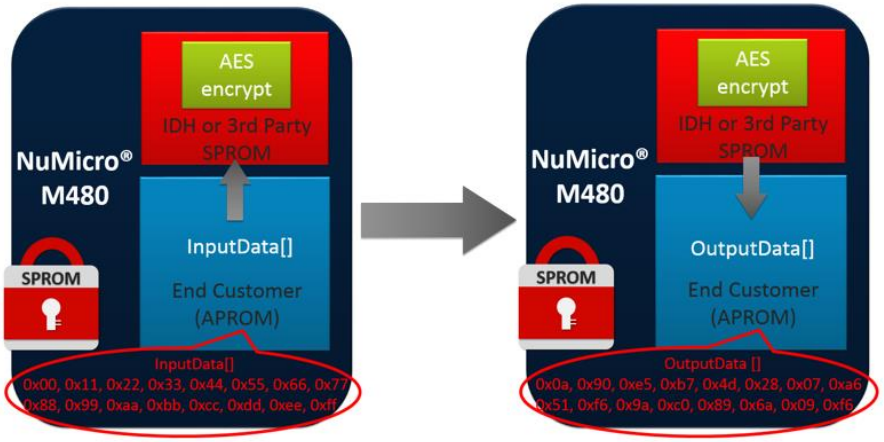
1.2 Principle

The mainly principle for user need to know that the private key of AES is stored in SPROM. AES encryption and decryption is also executed in SPROM.



1.3 Demo Result

AES Encrypt mode:

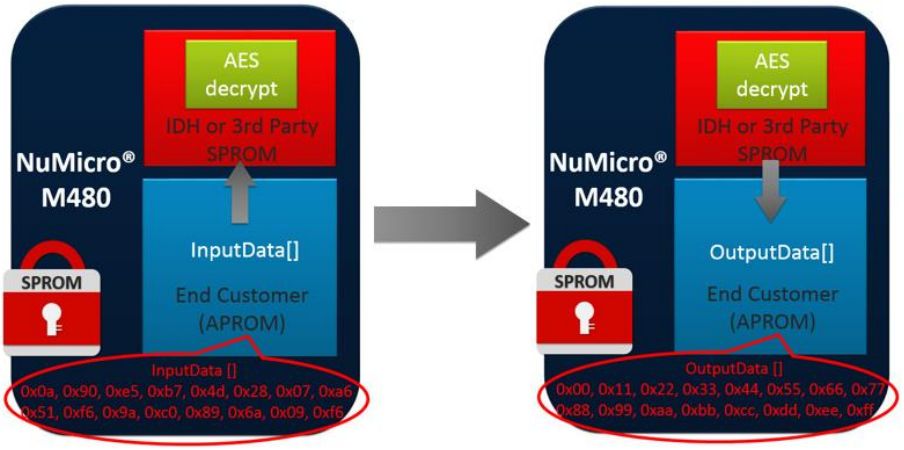



```
+----------------------------------------+
| M480 AES in SPROM Sample Demo          |
| [1] AES encrypt                        |
| [2] AES decrypt                        |
+----------------------------------------+
ISPSTS=0x8000001e, SPROM is secured.
AES encrypt done.
0x0000  0a 90 e5 b7 4d 28 07 a6 51 f6 9a c0 89 6a 09 f6    ....M(..Q....j..
```

發送區：已發送6位元組

1

AES Decrypt mode:




```
+--------------------------------------+
| M480 AES in SPROM Sample Demo        |
| [1] AES encrypt                      |
| [2] AES decrypt                      |
+--------------------------------------+
ISPSTS=0x8000001e, SPROM is secured.
AES decrypt done.
0x0000  00 11 22 33 44 55 66 77 88 99 aa bb cc dd ee ff   ..“3DUfw........
```

發送區：已發送6位元組

2

2 Code Description

This main() function is used to initialize platform for SPROM and CRYPTO.

```
void main(void)
{
    char chr;                           /* user input character */
    SYS_Init();                         /* Init System, IP clock and multi-function I/O */
    UART0_Init();                       /* Initialize UART0 */
    SYS_UnlockReg();                    /* Unlock protected registers */
    FMC_Open();                         /* Enable FMC ISP function */
    SYS_UnlockReg();                    /* Unlock protected registers */
    NVIC_EnableIRQ(CRPT_IRQn);
    AES_ENABLE_INT(CRPT);
        while(1){
            printf("\n\n\n");
            printf("+--------------------------------------+\n");
            printf("| M480 AES in SPROM Sample Demo        |\n");
            printf("| [1] AES encrypt                      |\n");
            printf("| [2] AES decrypt                      |\n");
            printf("+--------------------------------------+\n");
            if (FMC->ISPSTS & FMC_ISPSTS_SCODE_Msk)
                printf("ISPSTS=0x%x, SPROM is secured.\n", FMC->ISPSTS);
            else
                printf("ISPSTS=0x%x, SPROM is not secured.\n", FMC->ISPSTS);
            chr = getchar();
            switch (chr)
            {
            case '1':
                 /*AES encryption in SPROM*/
                sprom_aes_encrypt(au8InputData, sizeof(au8InputData), au8OutputData);
                dump_buff_hex(au8OutputData, sizeof(au8InputData));
                break;
            case '2':
                 /*AES decryption in SPROM*/
                sprom_aes_decrypt(au8OutputData, sizeof(au8InputData), au8InputData);
                dump_buff_hex(au8InputData, sizeof(au8InputData));
                break;
            }
        }
}
```

The sprom_aes_encrypt() and sprom_aes_decrypt() function are executed in SPROM.

```
static const uint32_t au32MyAESKey[8] =
{
    0x00010203, 0x04050607, 0x08090a0b, 0x0c0d0e0f,
    0x00010203, 0x04050607, 0x08090a0b, 0x0c0d0e0f
};
uint32_t au32MyAESIV[4] =
{
    0x00000000, 0x00000000, 0x00000000, 0x00000000
};

void sprom_aes_encrypt(uint8_t *inp, int length, uint8_t *outp)
{
    /*---------------------------------------
     *  AES-128 ECB mode encrypt
     *---------------------------------------*/
    AES_Open(CRPT, 0, 1, AES_MODE_ECB, AES_KEY_SIZE_128, AES_IN_OUT_SWAP);
    AES_SetKey(CRPT, 0, (uint32_t*)au32MyAESKey, AES_KEY_SIZE_128);
    AES_SetInitVect(CRPT, 0, au32MyAESIV);
    AES_SetDMATransfer(CRPT, 0, (uint32_t)inp, (uint32_t)outp, length);
    g_AES_done = 0;
    AES_Start(CRPT, 0, CRYPTO_DMA_ONE_SHOT);
    while (!g_AES_done);
    sprom_put_string("AES encrypt done.\n");
}

void sprom_aes_decrypt(uint8_t *inp, int length, uint8_t *outp)
{
    /*---------------------------------------
     *  AES-128 ECB mode decrypt
     *---------------------------------------*/
    AES_Open(CRPT, 0, 0, AES_MODE_ECB, AES_KEY_SIZE_128, AES_IN_OUT_SWAP);
    AES_SetKey(CRPT, 0, (uint32_t *)au32MyAESKey, AES_KEY_SIZE_128);
    AES_SetInitVect(CRPT, 0, au32MyAESIV);
    AES_SetDMATransfer(CRPT, 0, (uint32_t)inp, (uint32_t)outp, length);
    g_AES_done = 0;
    AES_Start(CRPT, 0, CRYPTO_DMA_ONE_SHOT);
    while (!g_AES_done);

    sprom_put_string("AES decrypt done.\n");
}
```

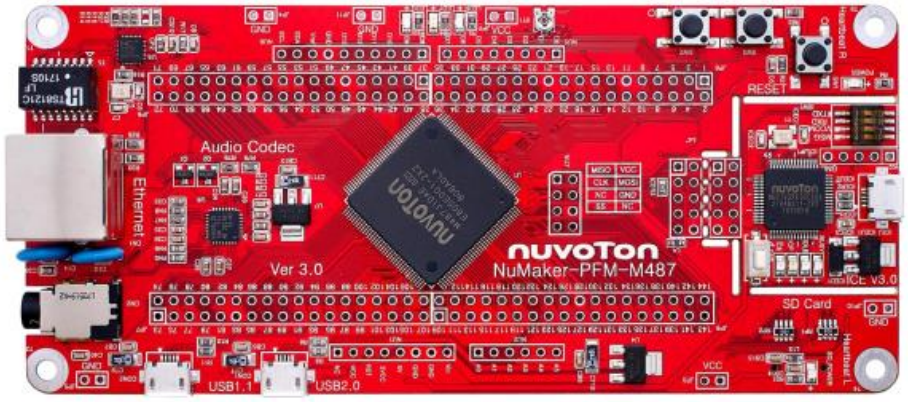
3 Software and Hardware Environment

Software Environment

- BSP version
  - M480 Series BSP CMSIS V3.03.001
- IDE version
  - Keil uVersion 5.26

Hardware Environment

- Circuit components
  - NuMaker-PFM-M487 V3.0 board
- Diagram



4 Directory Information

- EC_M480_CRYPTO_AES_In_SPROM_V1.00
  - Library — Sample code header and source files
    - CMSIS — Cortex® Microcontroller Software Interface Standard (CMSIS) by Arm® Corp.
    - Device — CMSIS compliant device header file
    - StdDriver — All peripheral driver header and source files
  - SampleCode
    - ExampleCode — Source file of example code

5 How to Execute Example Code?

1. Browsing into sample code folder by Directory Information (section 4) and double click CRYPTO_AES_In_SPROM.uvproj.
2. Enter Keil compile mode
   a. Build
   b. Download
   c. Start/Stop debug session
3. Enter debug mode
   a. Run

6 Revision History

Date	Revision	Description
Jul. 18, 2019	1.00	1. Initially issued.

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.