NUC1262 CRC 驗證代碼完整性

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 NUC1262 CRC 來做代碼完整性的自我驗證
BSP 版本	NUC1262_Series_BSP_CMSIS_V3.00.001
開發平台	NuMaker-NUC1262SE V1.0

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

本範例代碼使用 NUC1262 CRC 來做代碼完整性的自我驗證。

確保執行代碼的完整性是一個終端產品應具備的安全條件之一。這可避免產品在執行過程中因為代碼的不正確而產生異常的結果和造成死機，或者是因代碼被竄改而造成資料被竊取。

本示例代碼不需另接外部硬體模組。

需下載 SRecord 工具包 (http://srecord.sourceforge.net/)，使用 srec_cat.exe 來計算原代碼的 CRC32 校驗值，並另外產生一個包含原代碼和校驗值的 binary 格式文件，再將此 binary 格式文件轉換成可直接由 Keil IDE 做燒錄的 HEX 格式文件。

1.1 原理

因本示例代碼不需另接外部硬體模組，此章節將著重於介紹驗證原代碼 CRC32 校驗值的步驟，和在軟體專案中如何使用 srec_cat.exe 來產生包含 CRC32 校驗值文件的說明。

應用程式原代碼和其 CRC32 校驗值位於晶片內存的配置關係如圖 1-1所示。

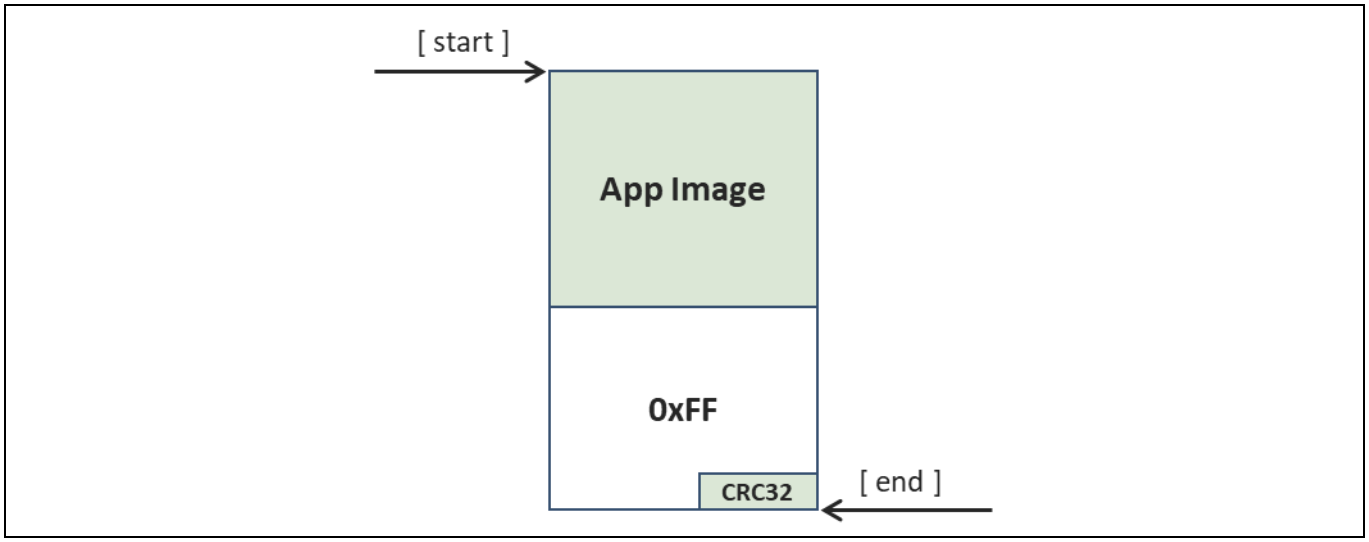

圖 1-1 應用程式原代碼和 CRC32 校驗值

驗證原代碼 CRC32 校驗值的步驟如圖1-2所示。

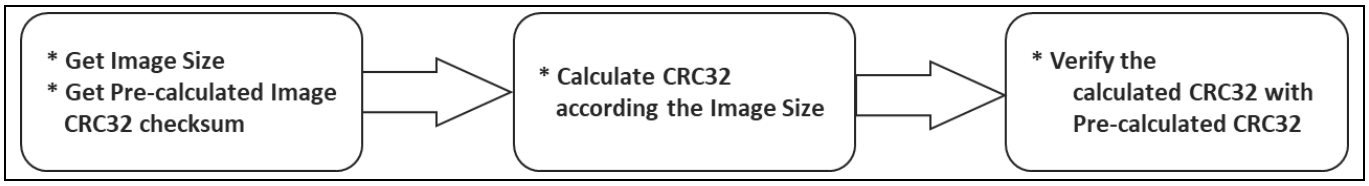

圖1-2 驗證原代碼 CRC32 校驗值

[Get Image Size] :

可由專案內的 ER_RO 和 ER_RW 長度計算出原代碼的位元組數。

(可參考 NUC1262_Verify_Image_CRC32.map)

[Get Pre-calculated Image CRC32 checksum] :

可經由定義在 main.c 內 IMAGE_CRC32_ADDRESS 的地址讀出原代碼的校驗值。

[Calculate CRC32 according the Image Size] :

依據原代碼的位元組數計算出 CRC32 校驗值。

[Verify the calculated CRC32 with Pre-calculated CRC32] :

比對計算出來的校驗值和預先寫入的校驗值是否相同。

本示例代碼將使用 GenCRC.bat 來執行 srec_cat.exe 的相關命令，GenCRC.bat 的內容和在專案內的配置如圖1-3所示。

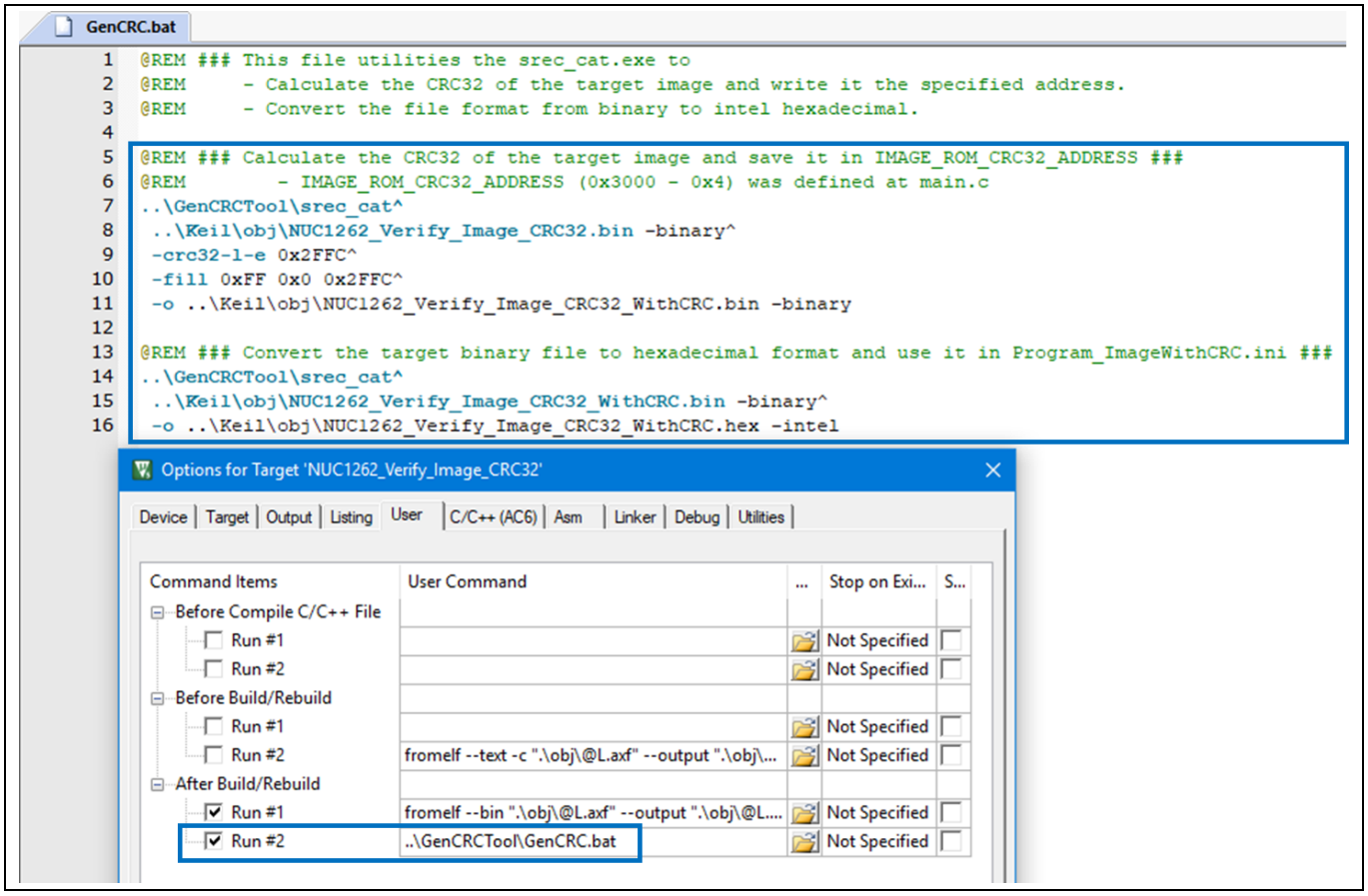

圖 1-3 GenCRC.bat 內容和配置

在編譯專案並產生原代碼的 binary 格式文件後，將會執行 GenCRC.bat 內的第一個 srec_cat 命令。此命令會先做原代碼 CRC32 校驗值的計算，並將此校驗值配置到指定的地址 (如: 0x2FFC)，之後便可產生一個新的 binary 格式文件，包含原代碼和其 CRC32 校驗值。而位於原代碼之後和存放校驗值之間的區域將被配置為全 0xFF。第二個 srec_cat 命令則是此新的 binary 格式文件轉成 intel HEX 格式的文件，其目的是為了可直接由 Keil IDE 做燒寫原代碼和校驗值到指定的位置。實現 Keil IDE 做燒寫所用到的文件為 ProgramTargetImageWithCRC.ini，其內容和在專案內的配置如圖1-4所示。

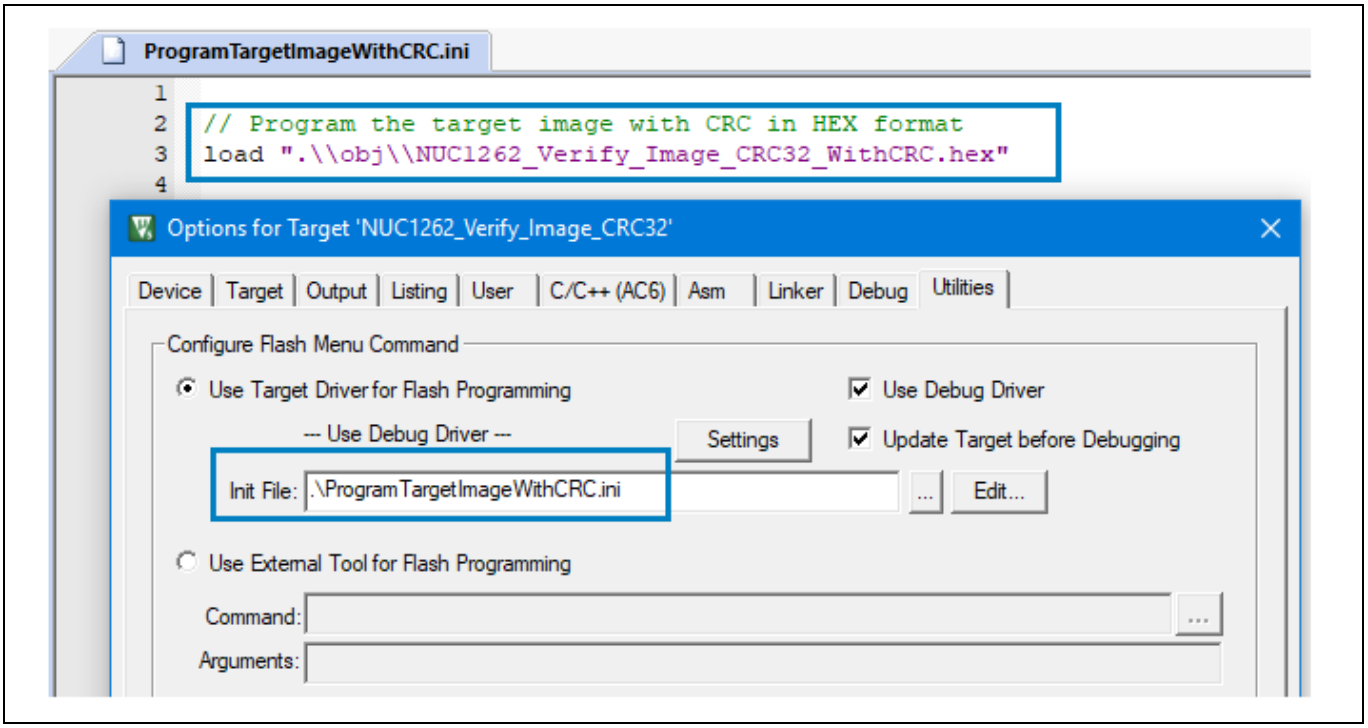

圖 1-4 ProgramTargetImageWithCRC.ini 內容和配置

1.2 執行結果

本示例代碼執行編譯和燒錄的結果如圖1-5所示。執行 GenCRC.bat 內的兩次 srec_cat 命令和燒寫 ProgramTargetImageWithCRC.ini 內指定的 HEX 文件。

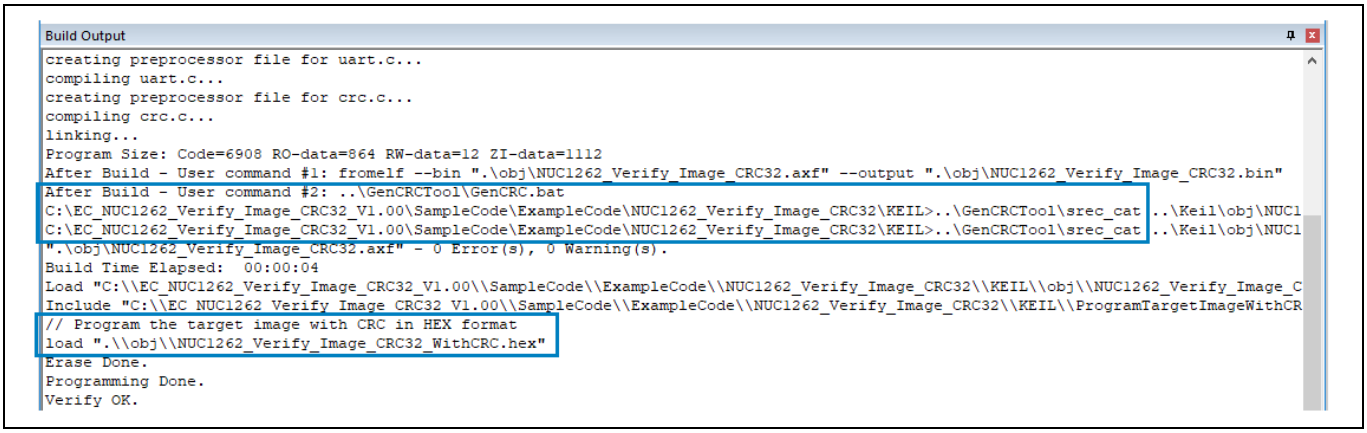

圖 1-5 執行編譯和燒錄結果

圖1-6說明開啟 NUC1262_Verify_Image_CRC32_WithCRC.bin 文件，可看到原代碼資料位元組數為 0x1e68 和其 CRC32 校驗值為 0x129c3e3f。

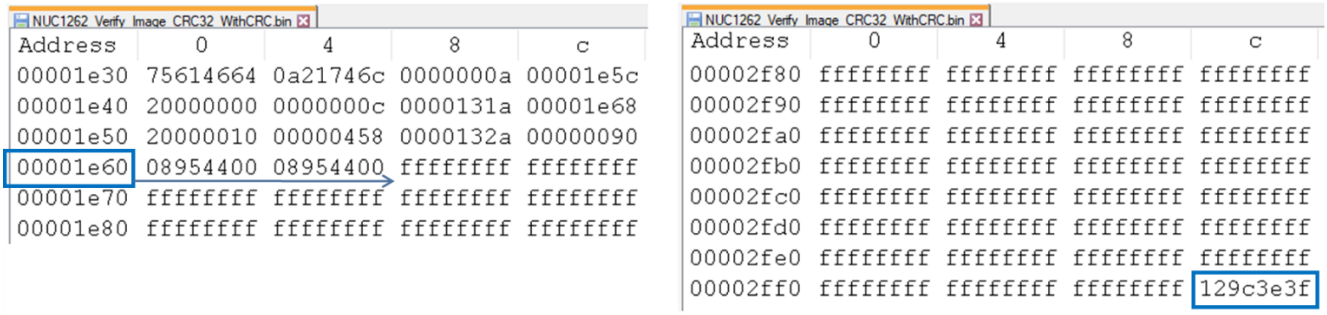

圖 1-6 原代碼 CRC32 校驗值

執行結果會由 PC UART 打印出原代碼的位元組數，CRC32 校驗值和依據原代碼位元組數計算得到的 CRC32 校驗值等資訊，如圖1-7所示。

```
[ Perform CRC32 calculation to verify image integrity ]
    # CPU @ 72000000 Hz.
    # The target Image ROM size is 0x1e68 (7784).
    # The pre-calculated CRC32 value 0x129c3e3f is stored in 0x00002ffc.

Start to perform HW CRC to claculate the Image ROM CRC32 checksum......
        - CRC32 checksum: 0x129c3e3f

[Compare Image ROM checksum ... PASS]
```

圖 1-7 執行結果

2 代碼介紹

計算原代碼位元組數的函式，代碼位於 main.c。

```
/**
  * @brief      Get Image Size
  * @param[in]  None
  * @return     Image ROM size (bytes)
  * @details    This function gets the target image size according to the Image RO and RW size.
  */
static uint32_t GetImageSize(void)
{
    extern uint32_t Image$$ER_RO$$Length; // Execution Region ER_RO can be found in NUC1262_Verify_Image_CRC32.map
    extern uint32_t Image$$ER_RW$$Length; // Execution Region ER_RW can be found in NUC1262_Verify_Image_CRC32.map
    uint32_t u32ROSize, u32RWSize;

    u32ROSize = (uint32_t)&Image$$ER_RO$$Length;
    u32RWSize = (uint32_t)&Image$$ER_RW$$Length;

    //printf("\tRO: %d, RW: %d.\n", u32ROSize, u32RWSize);

    return (u32ROSize + u32RWSize);
}
```

初始化系統後，經由 IMAGE_CRC32_ADDRESS 地址可取得原代碼 CRC32 校驗值，再和使用 CRC 模組計算出來的 CRC32 校驗值做比對。代碼位於 main.c。

```
    ……
 #define IMAGE_CRC32_ADDRESS         (0x3000 - 0x4) // 0x2FFC, the flash address to store pre-calculated CRC32
    ……
int main(void)
{
    volatile uint32_t addr;
    uint32_t u32ImageROMSize;
    uint32_t u32PreChks, u32CalChks;

    /* Unlock protected registers */
    SYS_UnlockReg();
```

```
    /* Init System, peripheral clock and multi-function I/O */
    SYS_Init();

    /* Init UART0 for printf */
    UART0_Init();

    /* Get the target image size and pre-calculated CRC32 checksum */
    u32ImageROMSize = GetImageSize();
    u32PreChks      = inpw(IMAGE_CRC32_ADDRESS);

    printf("\n\n[ Perform CRC32 calculation to verify image integrity ]\n");
    printf("    # CPU @ %d Hz.\n", SystemCoreClock);
    printf("    # The target Image ROM size is 0x%x (%d).\n", u32ImageROMSize,
u32ImageROMSize);
    printf("    # The pre-calculated CRC32 value 0x%08x is stored in 0x%08x.\n",
u32PreChks, IMAGE_CRC32_ADDRESS);
    printf("\n");



    printf("Start to perform HW CRC to claculate the Image ROM CRC32 checksum......\n");

    /* Configure CRC controller for CRC32 calculation */
    CRC_Open(CRC_32, (CRC_WDATA_RVS | CRC_CHECKSUM_RVS | CRC_CHECKSUM_COM), 0xFFFFFFFF,
CRC_CPU_WDATA_32);

    /* Calculate the Image ROM CRC32 checksum */
    for(addr=0; addr<u32ImageROMSize; addr+=4)
    {
        CRC_WRITE_DATA(inpw(addr));
    }
    u32CalChks = CRC_GetChecksum();
    printf("\t- CRC32 checksum: 0x%08x\n", u32CalChks);

    if(u32PreChks == u32CalChks)
    {
        printf("\n[Compare Image ROM checksum ... PASS]\n");
    }
    else
    {
        printf("\n[Compare Image ROM checksum ... WRONG]\n");
        printf("\tPre-calculated checksum:  0x%08x\n", u32PreChks);
```

```
        printf("\tCalculate CRC32 checksum: 0x%08x\n", u32CalChks);
    }

    /* Disable CRC function */
    CRC->CTL &= ~CRC_CTL_CRCEN_Msk;

    while(1) {}
}
```

3 軟體與硬體需求

- **軟體需求**
  - BSP 版本
    - NUC1262_Series_BSP_CMSIS_V3.00.001
  - IDE 版本
    - Keil uVersion 5.28
- **硬體需求**
  - 電路元件
    - NuMaker-NUC1262SE V1.0
  - 示意圖
    - 將 UART0 TX (PB.13) pin 腳連接到 PC UART RX，以顯示示例代碼的執行結果

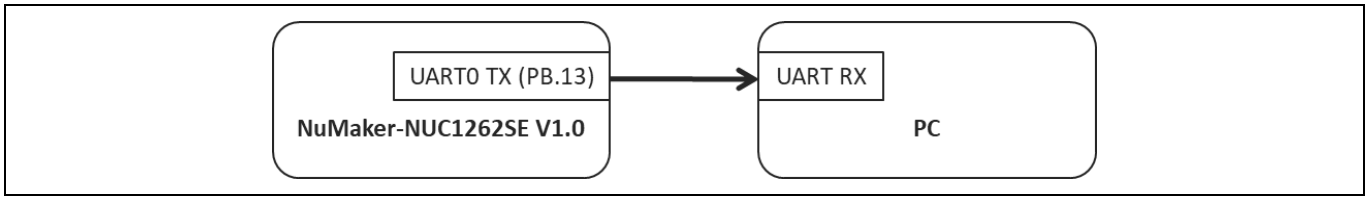

圖 3-1 NUC1262 UART0 TX 和 PC UART RX 連接

4 目錄資訊

EC_NUC1262_Verify_Image_CRC32_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

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 NUC1262_Verify_Image_CRC32.uvprojx。
2. 進入編譯模式介面
   a. 編譯
   b. 下載代碼至記憶體
   c. 進入 / 離開除錯模式
3. 進入除錯模式介面
   a. 執行代碼

6 修訂紀錄

Date	Revision	Description
2021.09.24	1.00	1. 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.