

**ARM® Cortex®-M**  
**32-bit Microcontroller**

**NuMicro® Family**  
**Nu-Link Command Tool**  
**User Manual**

*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 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](http://www.nuvoton.com)

Table of Contents

1 概述.....4

1.1 支持芯片 ..... 4

2 NU-LINK COMMAND TOOL .....5

2.1 安装 Nu-Link Command Tool ..... 5

2.2 启动 Nu-Link Command Tool ..... 7

2.3 命令格式 ..... 8

2.3.1 Nu-Link ID.....8

2.3.2 选项 .....8

2.3.3 编程 APROM by Nu-Link Command Tool.....24

2.3.4 多个 Nu-Link 并行执行命令 .....24

3 定制的闪存编程工具 .....25

3.1 环境 ..... 25

3.2 Nu-Link Vendor UI 用法..... 25

4 特定的系列 .....28

4.1 支持特定系列..... 28

5 版本历史 .....29

List of Figures

图 1-1 Nu-Link Command Tool 系统框架图 ..... 4

图 2-1 语言选择对话框 ..... 5

图 2-2 安装向导对话框 ..... 5

图 2-3 安装路径选择对话框 ..... 5

图 2-4 快捷路径设置对话框 ..... 6

图 2-5 安装信息确认对话框 ..... 6

图 2-6 安装完成对话框 ..... 6

图 2-7 打开Windows命令提示符 ..... 7

图 2-8 使用Windows命令提示符启动 Nu-Link. exe ..... 7

图 2-10 使用 -l 命令显示Nu-Link ID ..... 9

图 2-11 使用 -r 命令读取 Config 0 ..... 11

图 2-12 使用 -w 命令写入 Config 0 值 ..... 14

图 2-13 使用 -v 命令验证 Config 0 值 ..... 16

图 2-14 使用 -e 命令擦除芯片 ..... 18

图 2-15 使用 -reset 命令重置目标芯片 ..... 19

图 2-16 使用 -p 命令来显示目标芯片的型号 ..... 20

图 2-17 使用 -version 命令来显示应用程序和Nu-Link ICE的固件版本 ..... 21

图 2-18 使用 -update 命令更新Nu-Link ICE的固件 ..... 22

图 2-19 使用 -disconnect 命令断开与目标芯片的联机 ..... 23

图 3-1 Nu-Link Vendor UI 窗口 ..... 25

图 3-2 选择 Bin 文件 ..... 26

图 3-3 由于命令长度太长显示错误信息 ..... 26

图 3-4 利用 Vendor UI 编程数据目标芯片 ..... 27

图 3-5 显示所有连接的Nu-Link设备ID ..... 27

## 1 概述

Nu-Link Command Tool是一个通过Nu-Link访问闪存嵌入在NuMicro® MCU的Win32控制面板应用程序。此外，它也提供了更新Nu-Link固件功能。所有这些功能的输入选项将在第2章介绍。

Nu-Link Command Tool，用户可以擦除，编程和转储闪存根据他们的要求。换句话说，用户可以创建自己定制的闪存编程工具。在第3章，一个简单的Windows应用程序介绍，以供参考。

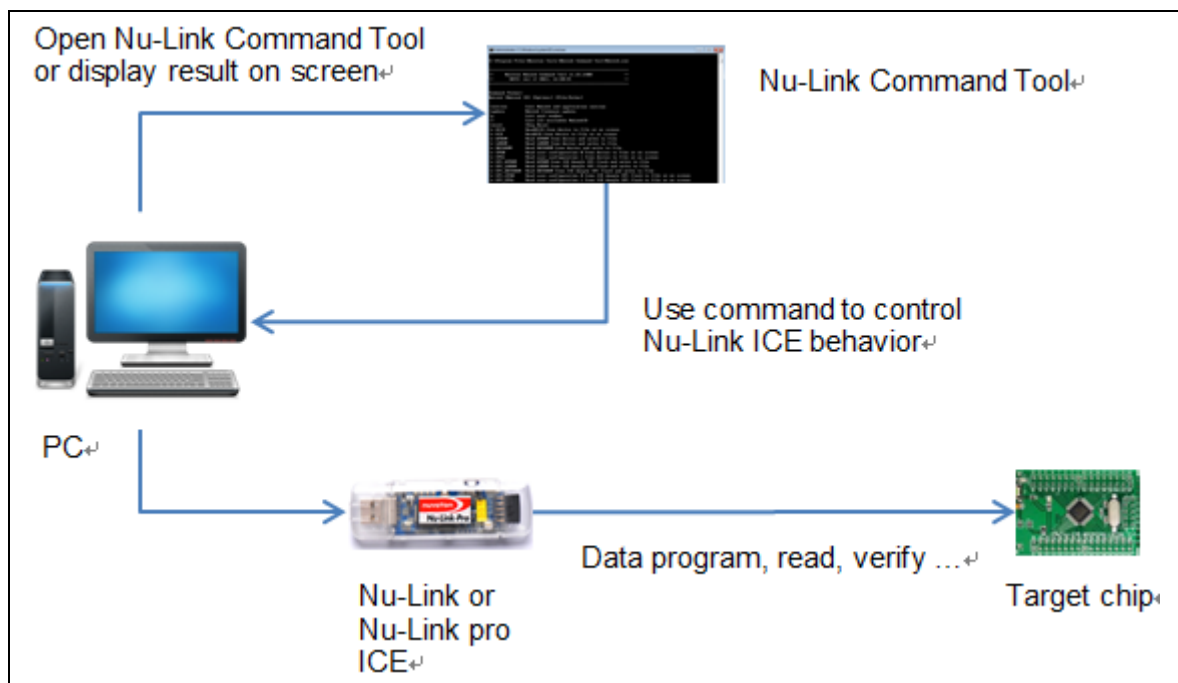


图 1-1 Nu-Link Command Tool 系统框架图

### 1.1 支持芯片

点击打开芯片的支持列表: [Link of supported devices.](#)

用户可以从 <http://www.nuvoton.com> 获取每个芯片的规格书。

## 2 NU-LINK COMMAND TOOL

### 2.1 安装 Nu-Link Command Tool

请执行安装工具安装NuLink Command Tool，并按照说明完成installation. 详细安装程序步骤如下。

1. 选择安装过程的语言。



图 2-1 语言选择对话框

2. 安装向导对话框中包含工具的版本号。

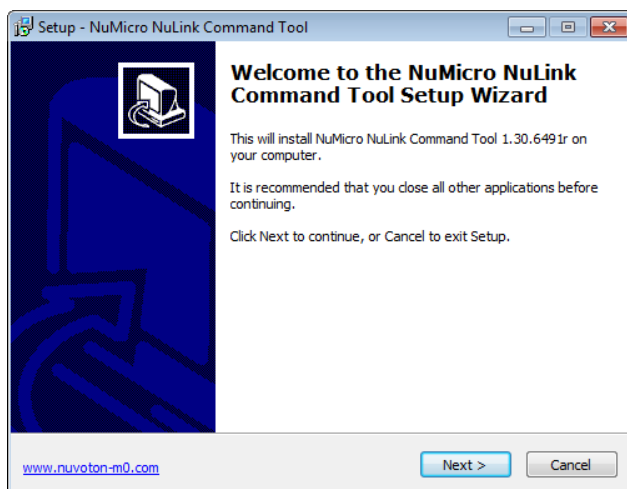


图 2-2 安装向导对话框

3. 浏览安装路径。

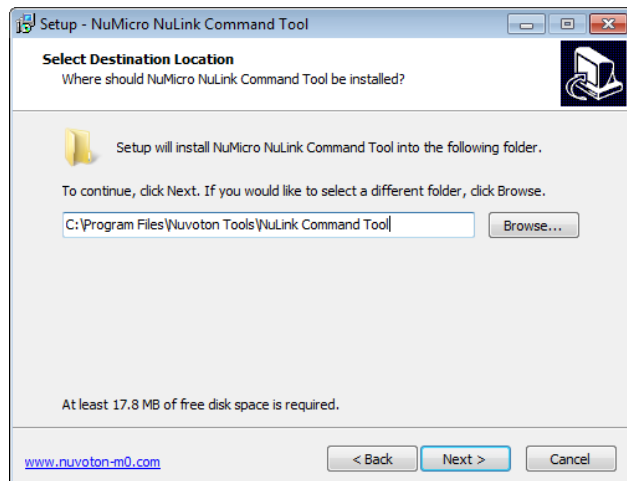


图 2-3 安装路径选择对话框

4. 选择要放置的“开始菜单”中的快捷路径。

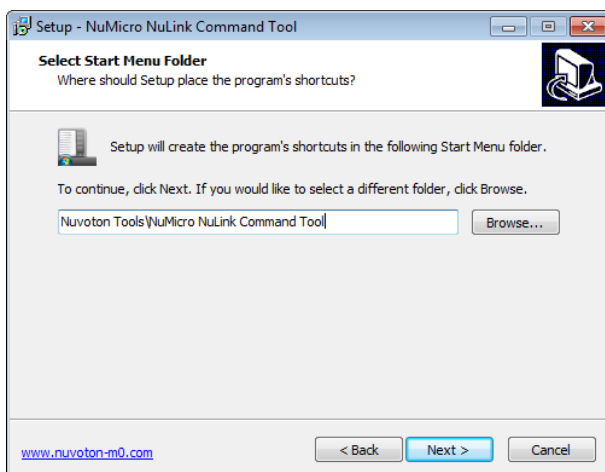


图 2-4 快捷路径设置对话框

5. 确认所有安装步骤的信息。

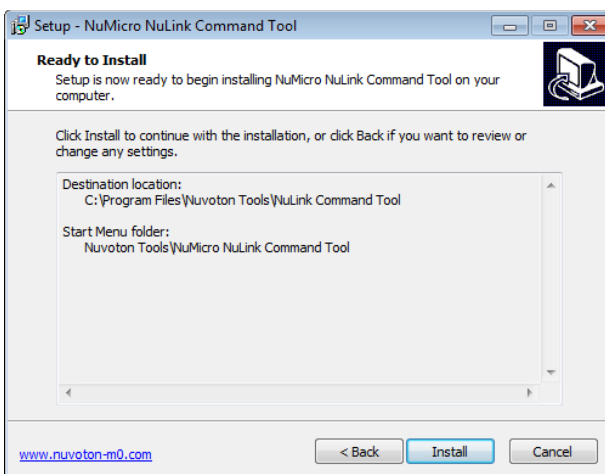


图 2-5 安装信息确认对话框

6. 完成 Nu-Link Command Tool 安装。

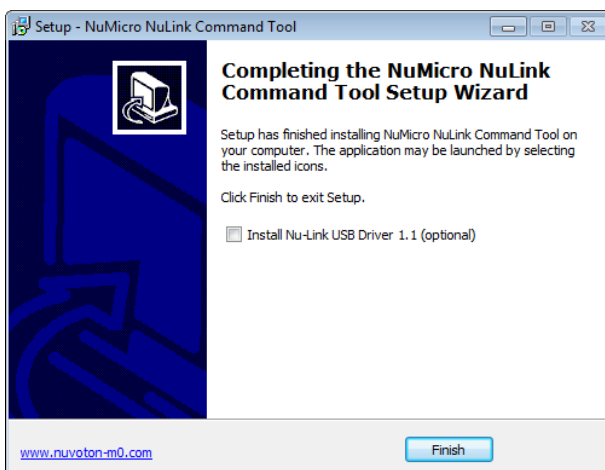


图 2-6 安装完成对话框

2.2 启动 Nu-Link Command Tool

要启动 Nu-Link Command Tool，请打开Windows命令提示符，然后改变当前目录到Nu-Link Command Tool的安装路径。键入“NuLink.exe”，然后按“Enter”键。Nu-Link Command Tool的简要说明会显示 在Windows命令提示符。

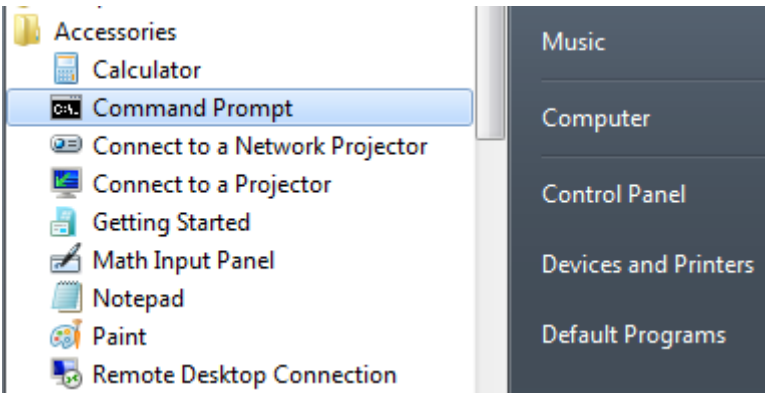


图 2-7 打开Windows命令提示符

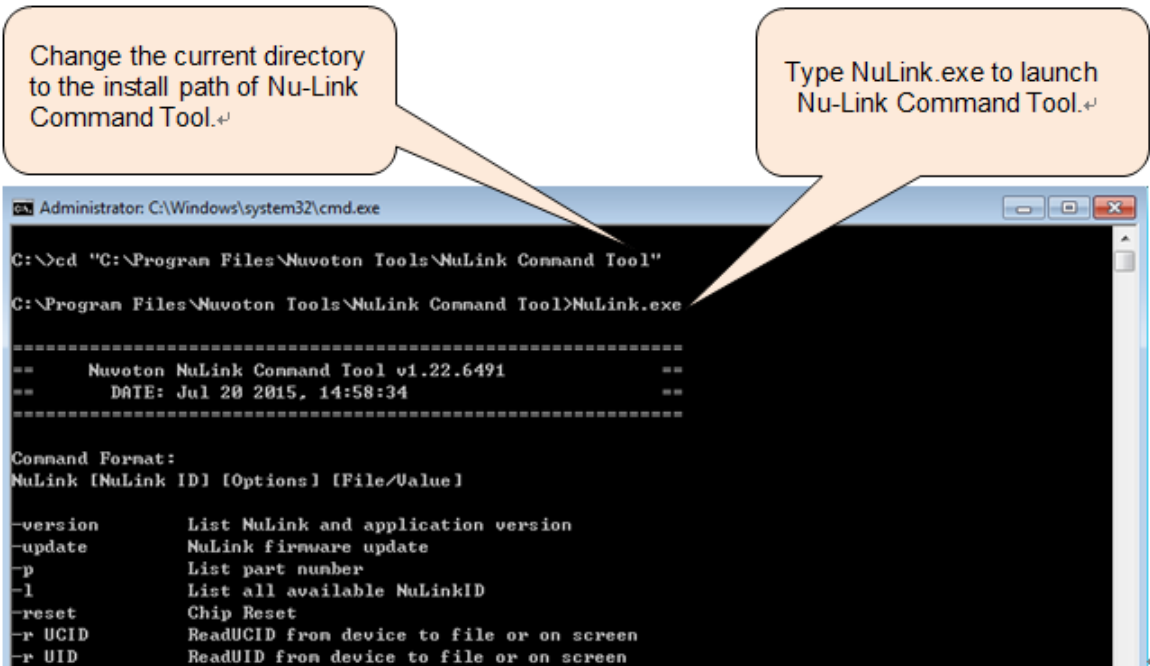


图 2-8 使用Windows命令提示符启动 Nu-Link. exe

2.3 命令格式

Nu-Link命令工具是基于控制台的命令工具。用户可以在Windows中打开一个命令提示符程序，然后执行的Nu-Link Command Tool，如下。

```
>NuLink [Nu-Link ID] [选项] [区块] [文件]
```

注：NuLink 是Nu-Link Command Tool的执行名称而且名称有区分大小写。

每个参数的使用在以下各节中描述。

2.3.1 Nu-Link ID

每个Nu-Link都有一个ID。用户可以选择相应的ID，以控制对应的Nu-Link ICE。

使用关键字参数“all\_nulink”代替Nu-Link ID，Nu-Link命令工具将为每个连接的Nu-Link对目标芯片执行相同的命令，并且所有连接的目标芯片都必须是相同的型号。

注：执行期间，过程消息将快速显示，并且仅在屏幕上显示最后执行的命令。用户可以使用“> file.txt”在文件中输出完整信息。

2.3.2 选项

Nu-Link 命令类型如下表格。

选项	描述
<a href="#">-disconnect</a>	断开与目标芯片的联机
<a href="#">-e</a>	擦除芯片的数据
<a href="#">-l</a>	显示所有可用Nu-Link ID
<a href="#">-p</a>	显示目标芯片的型号
<a href="#">-r</a>	读取芯片的数据
<a href="#">-reset</a>	重设目标芯片
<a href="#">-update</a>	升级Nu-Link ICE的固件
<a href="#">-v</a>	用文件或值来验证芯片上的数据
<a href="#">-version</a>	显示Nu-Link Command Tool当前版号和Nu-Link ICE的固件版号
<a href="#">-w</a>	写文件数据或值到芯片



### 2.3.2.1 显示所有可用 Nu-Link IDs

命令：显示所有可用 Nu-Link ID。

```
>NuLink -l
```

实例：

```
>NuLink -l
```

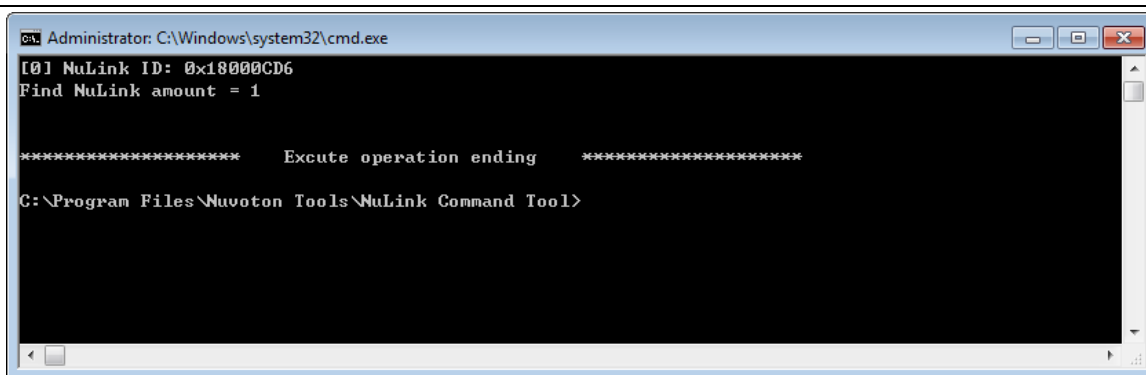


图 2-9 使用 -l 命令显示Nu-Link ID

### 2.3.2.2 读取芯片的数据

命令：读取目标芯片的区块数据。

```
>NuLink [Nu-Link ID] -r [区块] [文件]
```

[Nu-Link ID] 可选择的。

[区块] 必要项。参数名为以下所示和不区分大小写。

读取目标芯片的数据： CFG0, CFG1,CFG2, CFG3, LDROM, APROM, DATAROM, SPROM, SPROM2, SPROM3, KPROM, UID, UCID。

[文件] 必要项。但读取config0和config1是可选择的。

存放输出结果的文件名路径。

```
>NuLink [Nu-Link ID] -r SPIFLASH [文件] address [值1] size [值2] channel [值3]
```

[值1] 读取 SPI flash 的起始地址。

[值2] 读取 SPI flash 的大小。

[值3] SPIM channel。 可选择的。

注：

如果在PC上接上超过一个Nu-Link ICE而且没有指定Nu-Link ID， 命令可能无法正常运转。

实例：

在命令窗口显示芯片的UCID值。

用Nu-Link ID为0x12345678的ICE读取芯片的config0值显示在命令窗。

```
>NuLink 0x12345678 -r CFG0
```

读取芯片的 APROM 数据并写入到文件。

```
>NuLink -r APROM C:\APROM.bin
```

用Nu-Link ID为0x12345678的ICE读取芯片的APROM数据并写入到文件。

```
>NuLink 0x12345678 -r APROM C:\APROM.bin
```

所有连接目标芯片的ICE读取芯片的APROM数据并写入到文件。

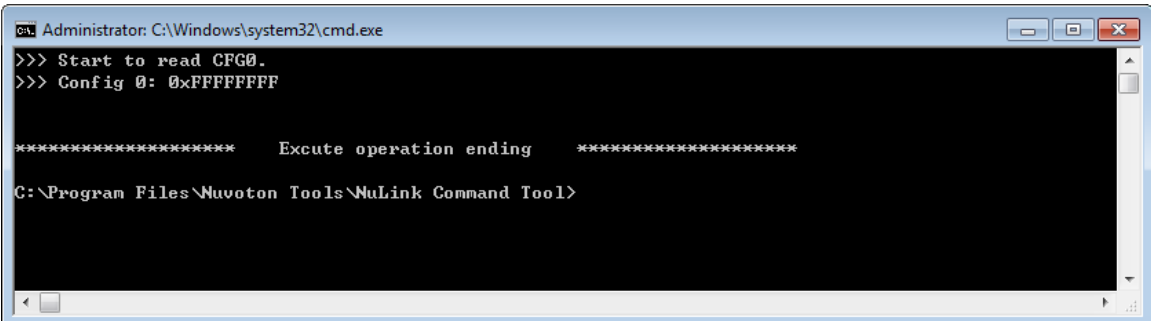
```
>NuLink all_nulink -r APROM C:\APROM.bin
```

读取芯片的SPI flash 数据 (范围0x0~0x1000) 并写入到文件。

```
>NuLink -r SPIFLASH C:\SPIFLASH.bin address 0x0 size 0x1000
```

读取芯片的config0值显示在命令窗口。

>**NuLink** -r CFG0



```

Administrator: C:\Windows\system32\cmd.exe
>>> Start to read CFG0.
>>> Config 0: 0xFFFFFFFF

***** Excute operation ending *****

C:\Program Files\Nuvoton Tools\NuLink Command Tool>
    
```

图 2-10 使用 -r 命令读取 Config 0

### 2.3.2.3 将数据写入芯片

命令：写数据到目标芯片区块。

**>NuLink** [Nu-Link ID] -w [区块] [文件/值] [offset] [偏移值]

[Nu-Link ID] 可选择的。

[区块] 必要项。参数名为以下所示和不区分大小写。

写入数据到目标芯片：CFG0, CFG1, CFG2, CFG3, LDROM, APROM, DATAROM, SPROM, SPROM2, SPROM3, SBKEY, IB。

[文件/值] 必要项。

[offset] 可选择的。偏移关键字。

[偏移值] 可选择的。写入数据到目标芯片的偏移值。

要写入数据的文件名路径。

**>NuLink** [Nu-Link ID] -w SPIFLASH [文件/值] address [值1] channel [值2] key0 [值3] key1 [值4]

[值1] 写入 SPI flash 的起始地址。

[值2] SPIM channel。可选择的。

[值3] 加密SPI flash 数据的金钥0。数值为0代表不加密写数据到SPI flash。可选择的。

[值4] 加密SPI flash 数据的金钥1。数值为0代表不加密写数据到SPI flash。可选择的。

**>NuLink** [Nu-Link ID] -w SN number [值1] address [值2]

[值1] 写入软件序列号的值。

[值2] 写入软件序列号的地址

**>NuLink** [Nu-Link ID] -w KPROM [值1] [值2] [值3] [值4] [值5] [值6]

[值1] 写入KPROM金钥0的值。

[值2] 写入KPROM金钥1的值。

[值3] 写入KPROM金钥2的值。

[值4] 金钥错误尝试的最大重新上电次数。

[值5] 每次上电的金钥错误尝试最大次数。

[值6] 资料防写。

值 0 = APROM and KPROM防写。

值 1 = APROM, KPROM and configuration防写。

值 2 = APROM, KPROM and SPROM防写。

值 3 = APROM, KPROM, SPROM and configuration防写。

值 其他 = APROM and KPROM防写。

注：

如果在PC上接上超过一个Nu-Link ICE而且没有指定Nu-Link ID，命令可能无法正常运作。

## 实例：

用Nu-Link ID为0x12345678的ICE写入0xFFFFFFFF值到目标芯片的config0。

**注：**配置值为32位数据长度。要为8051系列编程8位数据长度的配置值，用户需要将四个配置值组合为一个。

在8051系列中：Config0 = 0xDB, Config1 = 0xFB, Config2 = 0x67, Config3 = 0xFF, Config4 = 0x5F。

在命令配置值：CFG0 = 0xFF67FBDB, CFG1 = 0xFFFFFFFF5F。

```
>NuLink 0x12345678 -w CFG0 0xFFFFFFFF
```

将文件数据写入目标芯片的APROM。

```
>NuLink -w APROM C:\APROM.bin
```

用Nu-Link ID为0x12345678的ICE将文件数据写入目标芯片的APROM。

```
>NuLink 0x12345678 -w APROM C:\APROM.bin
```

所有连接目标芯片的ICE将文件数据写入目标芯片的APROM。

```
>NuLink all_nulink -w APROM C:\APROM.bin
```

将文件数据写入目标芯片的APROM。偏移值0x1000。

```
>NuLink -w APROM C:\APROM.bin offset 0x1000
```

将文件数据写入目标芯片的SPI flash 起始地址为0x0。

**注：**金钥0数值为0x123，金钥1数值为0x456。使用非0数值代表加密数据并写入到SPI flash。使用0数值代表不加密数据并写入到SPI flash。

```
>NuLink -w SPIFLASH C:\SPIFLASH.bin address 0x0 key0 0x123 key1 0x456
```

将软件序列号值0x18000000写入目标芯片地址0x100010。

```
>NuLink -w SN number 0x18000000 address 0x100010
```

写入0xFFFFFFFF值到目标芯片的config0。

```
>NuLink -w CFG0 0xFFFFFFFF
```

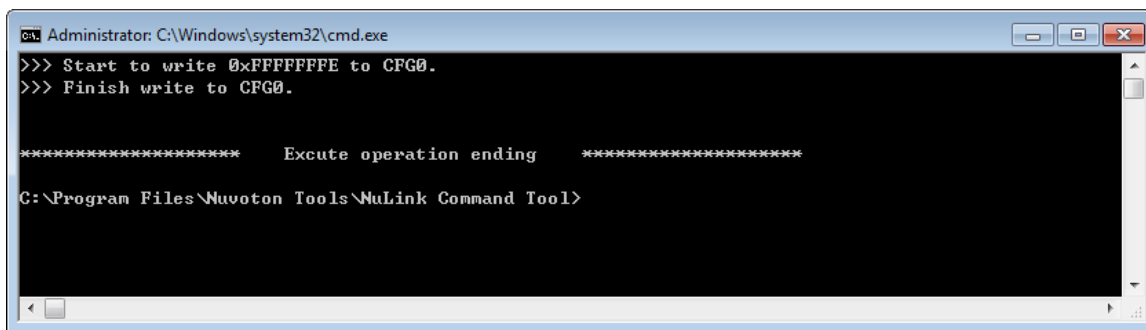


图 2-11 使用 -w 命令写入 Config 0 值

#### 2.3.2.4 验证芯片数据与源文件/值

**命令：**验证目标芯片数据与文件/值。

```
>NuLink [Nu-Link ID] -v [区块][文件/值][offset][偏移值]
```

[Nu-Link ID] 可选择的。

[区块] 必要项。参数名为以下所示和不区分大小写。

验证目标芯片的数据：CFG0, CFG1, CFG2, CFG3, LDROM, APROM, DATAROM, SPROM, SPROM2, SPROM3。

[文件/值] 必要项。

[offset] 可选择的。偏移关键字。

[偏移值] 可选择的。验证目标芯片数据的偏移值。

指定要用来验证数据的文件名路径。

```
>NuLink [Nu-Link ID] -v SPIFLASH [文件/值] address [值1] channel [值2]
```

[值1] 验证 SPI flash 的起始地址。

[值2] SPIM channel。 可选择的。

#### 注：

如果在PC上接上超过一个Nu-Link ICE而且没有指定Nu-Link ID，命令可能无法正常运转。

#### 实例：

用Nu-Link ID为0x12345678的ICE验证目标芯片的config0是否为0xFFFFFFFFE值。

```
>NuLink 0x12345678 -v CFG0 0xFFFFFFFFE
```

从文件数据验证目标芯片的LDROM数据。

```
>NuLink -v LDROM C:\LDROM.bin
```

从文件数据验证目标芯片的APROM数据偏移值0x1000。

```
>NuLink -v APROM C:\APROM.bin offset 0x1000
```

用Nu-Link ID为0x12345678的ICE从文件数据验证目标芯片的LDROM数据。

```
>NuLink 0x12345678 -v LDROM C:\LDROM.bin
```

所有连接目标芯片的ICE从文件数据验证目标芯片的LDROM数据。

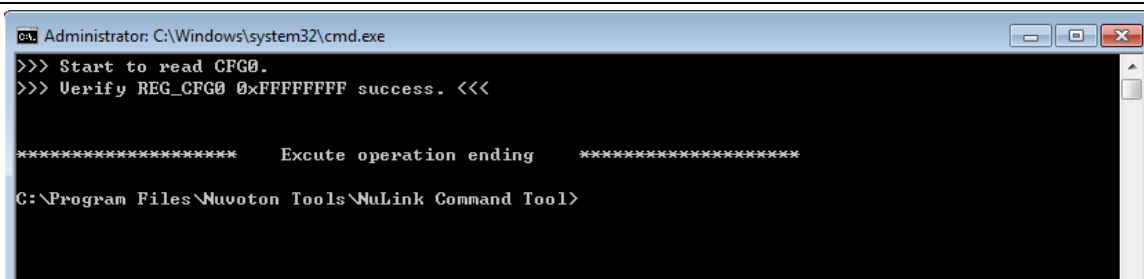
```
>NuLink all_nulink -v LDROM C:\LDROM.bin
```

从文件数据验证SPI flash 数据起始地址为0x0。

```
>NuLink -v SPIFLASH C:\SPIFLASH.bin address 0x0
```

验证目标芯片的 config0 是否为 0xFFFFFFFF 值。

>**NuLink** -v CFG0 0xFFFFFFFF



```

Administrator: C:\Windows\system32\cmd.exe
>>> Start to read CFG0.
>>> Uerify REG_CFG0 0xFFFFFFFF success. <<<

*****      Excute operation ending      *****

C:\Program Files\Nuvoton Tools\NuLink Command Tool>
    
```

图 2-12 使用 -v 命令验证 Config 0 值



### 2.3.2.5 擦除芯片/闪存ROM

**命令：**擦除目标芯片区块数据。

```
>NuLink [Nu-Link ID] -e [区块] [offset] [offset value]
```

[Nu-Link ID] 可选择的。

[区块] 必要项。参数名为以下所示和不区分大小写。

擦除目标芯片数据： all, LDROM, APROM, DATAROM, SPROM, SPROM2, SPROM3, KPROM。

[offset] 可选择的。偏移关键字。

[偏移值] 可选择的。擦除目标芯片数据的偏移值。

```
>NuLink [Nu-Link ID] -e SPIFLASH [文件/值] address [值1] size [值2] channel [值3]
```

[值1] 擦除 SPI flash 的起始地址。

[值2] 擦除 SPI flash 的大小。

[值3] SPIM channel。可选择的。

**注：**

如果在PC上接上超过一个Nu-Link ICE而且没有指定Nu-Link ID，命令可能无法正常运作。

**实例：**

擦除目标芯片的LDROM数据。

```
>NuLink -e LDROM
```

擦除目标芯片的APROM数据。

```
>NuLink -e APROM
```

擦除目标芯片的APROM数据偏移值0x1000。

```
>NuLink -e APROM offset 0x1000
```

擦除目标芯片的DATAROM数据。

```
>NuLink -e DATAROM
```

用Nu-Link ID为0x12345678的ICE擦除目标芯片的LDROM数据。

```
>NuLink 0x12345678 -e LDROM
```

用Nu-Link ID为0x12345678的ICE擦除目标芯片的APROM数据。

```
>NuLink 0x12345678 -e APROM
```

所有连接目标芯片的ICE擦除目标芯片的APROM数据。

```
>NuLink all_nulink -e APROM
```

用Nu-Link ID为0x12345678的ICE擦除目标芯片的DATAROM数据。

```
>NuLink 0x12345678 -e DATAROM
```

擦除SPI flash数据 (范围0x0~0x1000)。

```
>NuLink -e SPIFLASH address 0x0 size 0x1000
```

擦除所有连接目标芯片的整个数据。

```
>NuLink all_nulink -e all
```

增加型号参数来擦除解锁目标芯片。

```
>NuLink -e all M483KGCAE
```

擦除整个目标芯片数据包括config0, config1, config2, config3, LDROM, APROM, DATAROM, KPROM。

```
>NuLink -e all
```

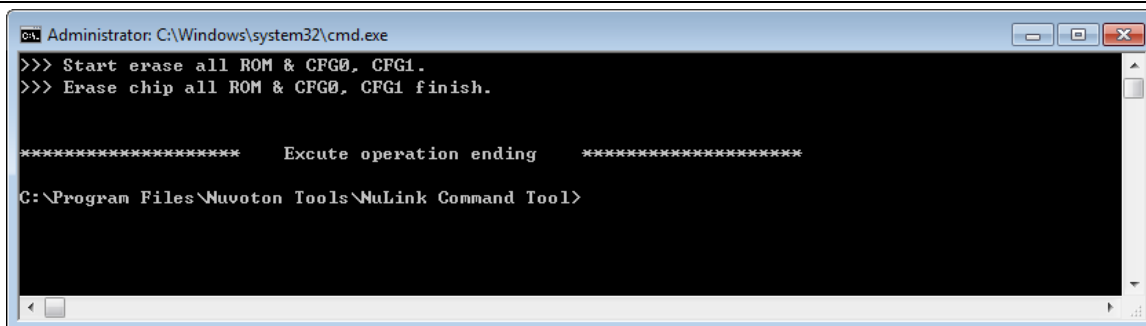


图 2-13 使用 -e 命令擦除芯片

### 2.3.2.6 重设目标芯片

**命令：**重设目标芯片。

```
>NuLink [Nu-Link ID] -reset
```

[Nu-Link ID] 可选择的。

**注：**

如果在PC上接上超过一个Nu-Link ICE而且没有指定Nu-Link ID，命令可能无法正常运作。

**实例：**

用Nu-Link ID为0x12345678的ICE重设目标芯片。

```
>NuLink 0x12345678 -reset
```

重设所有连接的目标芯片。

```
>NuLink all_nulink -reset
```

重设目标芯片。

```
>NuLink -reset
```

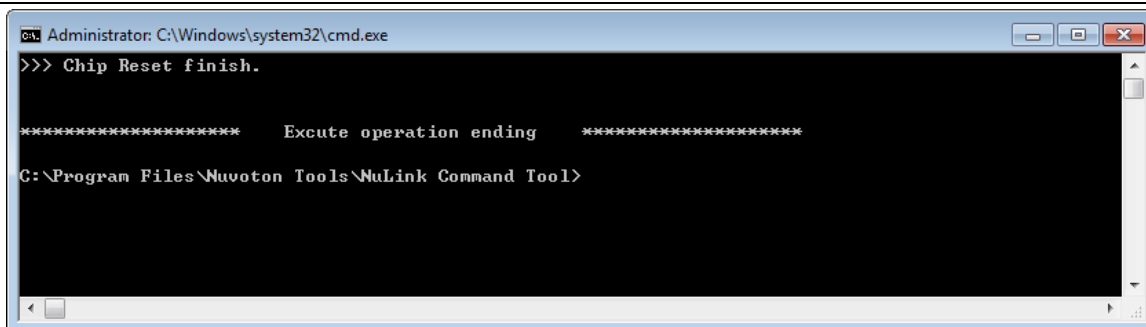


图 2-14 使用 -reset 命令重置目标芯片

### 2.3.2.7 显示目标芯片的型号

命令：显示目标芯片的型号。

```
>NuLink -p
```

注：

命令只支持一个Nu-Link ICE接上PC。

实例：

显示目标芯片的型号。

```
>NuLink -p
```

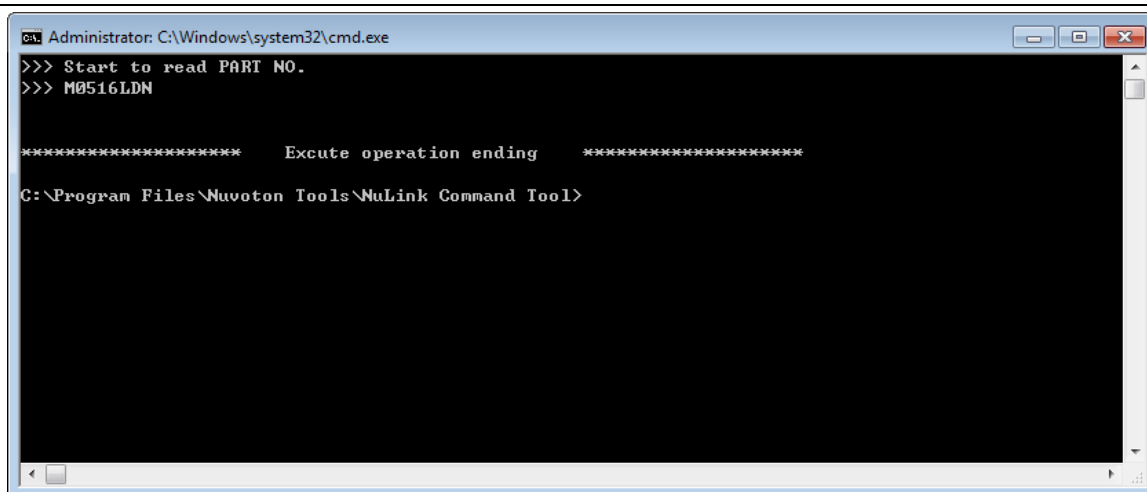


图 2-15 使用 -p 命令来显示目标芯片的型号

### 2.3.2.8 显示当前的应用程序和Nu-Link ICE的固件版本号

命令：显示当前的应用程序和Nu-Link ICE的固件版本号。

```
>NuLink -version
```

注：

命令只支持一个Nu-Link ICE接上PC。

Nu-Link ICE固件版号为当前接上PC的Nu-Link ICE的固件版本号。

当前应用程序版号为Nu-Link Command Tool应用程序的版号。

实例：

显示当前的应用程序和Nu-Link ICE的固件版本号。

```
>NuLink -version
```

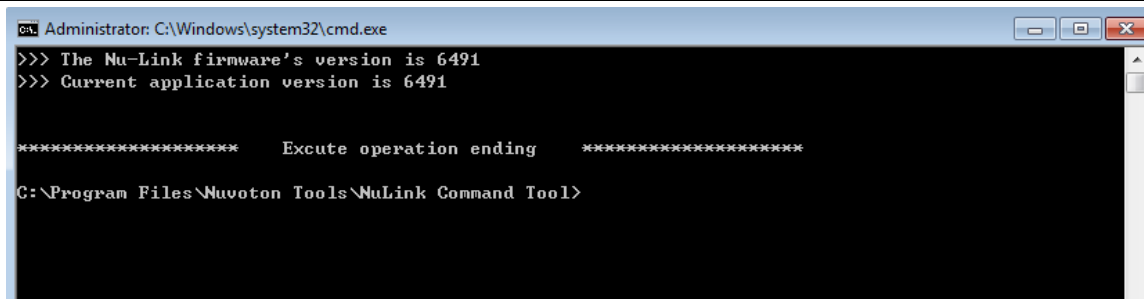


图 2-16 使用 -version 命令来显示应用程序和Nu-Link ICE的固件版本

### 2.3.2.9 Nu-Link ICE 固件更新

命令：Nu-Link ICE 固件更新。

```
>NuLink -update
```

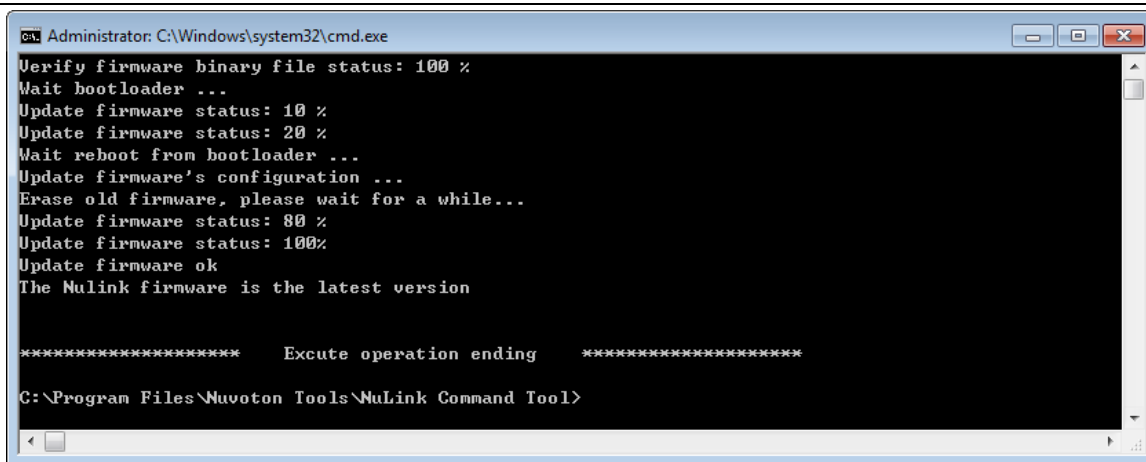
注：

命令只支持一个Nu-Link ICE接上PC。

如果Nu-Link ICE固件版号与当前应用程序版号不同时，执行该行命令，固件版号将被变更成当前应用程序版号。

实例：

```
>NuLink -update
```



```
Administrator: C:\Windows\system32\cmd.exe
Verify firmware binary file status: 100 %
Wait bootloader ...
Update firmware status: 10 %
Update firmware status: 20 %
Wait reboot from bootloader ...
Update firmware's configuration ...
Erase old firmware, please wait for a while...
Update firmware status: 80 %
Update firmware status: 100%
Update firmware ok
The Nulink firmware is the latest version

*****      Excute operation ending      *****

C:\Program Files\Nuvoton Tools\NuLink Command Tool>
```

图 2-17 使用 -update 命令更新Nu-Link ICE的固件

### 2.3.2.10 断开与目标芯片的联机

**Command:** 断开与目标芯片的联机。

```
>NuLink [Nu-Link ID] -disconnect
```

[Nu-Link ID] 可选择的。

**Remark:**

如果在PC上接上超过一个Nu-Link ICE而且没有指定Nu-Link ID，命令可能无法正常运作。  
将数据写入芯片后务必使用该命令。

**Example:**

用Nu-Link ID为0x12345678的ICE断开与目标芯片的联机。

```
>NuLink 0x12345678 -disconnect
```

断开所有连接的目标芯片的联机。

```
>NuLink all_nulink -disconnect
```

断开与目标芯片的联机。

```
>NuLink -disconnect
```

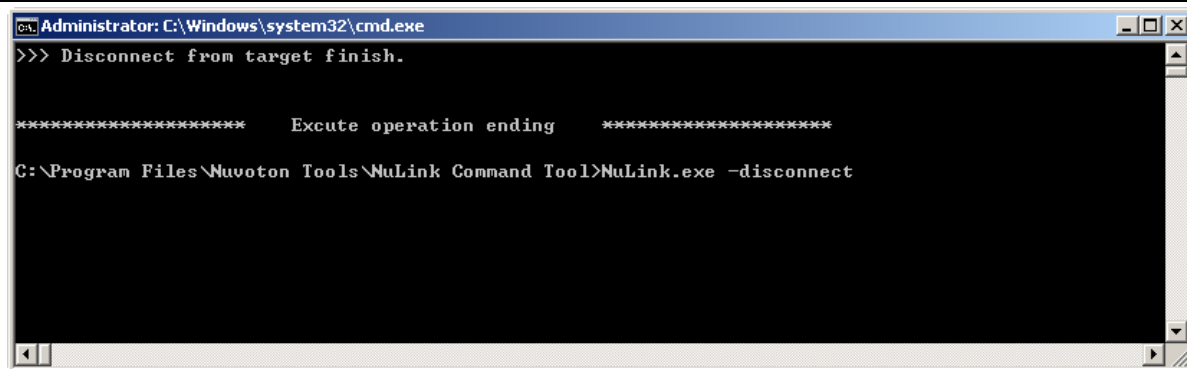


图 2-18 使用 -disconnect 命令断开与目标芯片的联机

2.3.3 编程 APROM by Nu-Link Command Tool

以下步骤是写“APROM.bin”文件数据到APROM目标芯片，验证和读出APROM数据的一个例子。最后，目标芯片将被重置。

> <b>NuLink</b> -e APROM
> <b>NuLink</b> -w APROM C:\APROM.bin
> <b>NuLink</b> -v APROM C:\APROM.bin
> <b>NuLink</b> -r APROM C:\R_APROM.bin
> <b>Nulink</b> -reset

注：在编程数据前，用户需要擦除芯片。

以下步骤是所有连接的目标芯片写“APROM.bin”文件数据到APROM目标芯片，验证和读出APROM数据的一个例子。最后，目标芯片将被重置。

> <b>NuLink</b> all_nulink -e APROM
> <b>NuLink</b> all_nulink -w APROM C:\APROM.bin
> <b>NuLink</b> all_nulink -v APROM C:\APROM.bin
> <b>NuLink</b> all_nulink -r APROM C:\R_APROM.bin
> <b>Nulink</b> all_nulink -reset

2.3.4 多个 Nu-Link 并行执行命令

请注意，以下项目必须提前做好。

- 1. 将所有 Nu-Link 和目标芯片连接到您的 PC。
- 2. 确认每个Nu-Link ID不同，每个目标芯片相同。
- 3. 如果已经安装，请卸载 Nuvoton USB 驱动程序。

下面是并行执行命令的步骤。

- 1. Nulink -parallel [目标芯片的型号]（例如：NuLink -parallel M483KGCAE）。
- 2. 添加您的命令在每个批处理文件中，并在每一个命令最后添加 -parallel 参数。
- 3. 运行每个批处理文件并行执行命令。

步骤 2给一个批处理文件示例：

```
NuLink 0x18000000 -e APROM -parallel
NuLink 0x18000000 -w APROM test.bin -parallel
NuLink 0x18000000 -v APROM test.bin -parallel
NuLink 0x18000000 -reset -parallel
```

注：并行执行命令不支持特定系列。



### 3 定制的闪存编程工具

本章介绍了一种基于Nu-Link Command Tool的Windows应用程序例子。这个程序调用的Nu-Link Command Tool来访问嵌入在NuMicro® MCU的内存FASH。它支持同时更新多个NuMicro® MCU和 支持动态扫描所有连接NuMicro® 的MCU。

示例源代码存储在安装文件夹。用户可以参考源代码来创建自己的定制闪存编程工具。

#### 3.1 环境

The Vendor UI 工具基于 Nu-Link Command Tool. 因此, 用户需要在执行Vendor UI之前先安装 Nu-Link Command Tool。

Nu-Link Command Tool 执行文件的默认路径是在 “C:\Program Files\Nuvoton Tools\Nu-Link Command Tool”。

预编译的执行版本路径为\NuLinkVendorUI\Debug\NuLinkVendorUI.exe”。

#### 3.2 Nu-Link Vendor UI 用法

以下说明步骤是用于执行NuLinkVendorUI。

步骤1:

连接 Nu-Link ICE to PC。

步骤2:

执行 NuLinkVendorUI.exe。

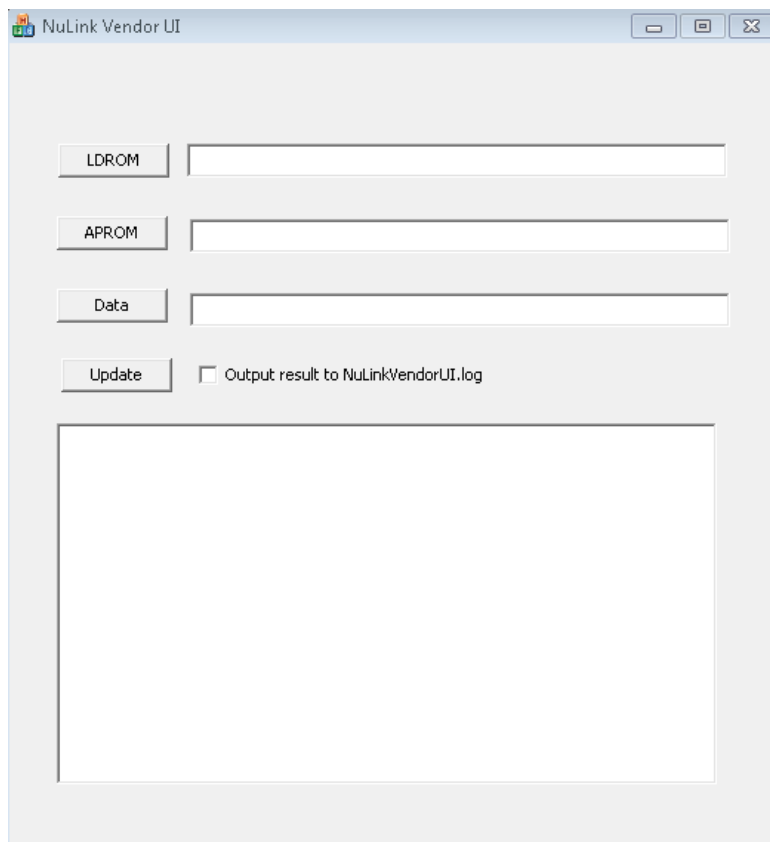


图 3-1 Nu-Link Vendor UI 窗口

步骤3:

按“LDROM”，“APROM”或“Data”按钮，从硬盘选择bin文件。

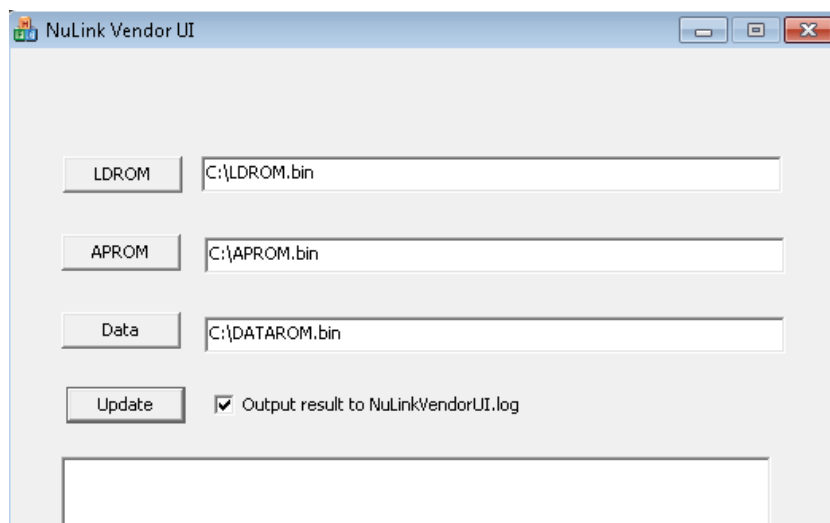


图 3-2 选择 Bin 文件

Cmd.exe的命令有长度的限制。对于Microsoft Windows XP或更高版本，可以使用在命令提示符下字符串的最大长度为8191个字符。对于Microsoft Windows2000或Windows NT4.0，可以使用在命令提示符下字符串的最大长度为2047个字符。

例如：如果命令长度超过限制，Cmd.exe 会显示“命令格式错误!!!”信息。

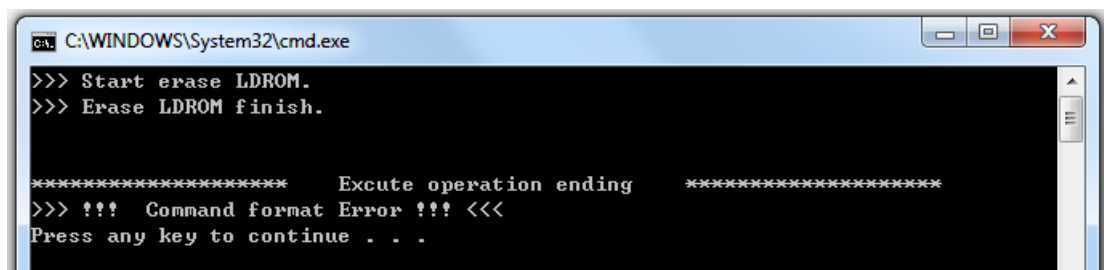


图 3-3 由于命令长度太长显示错误信息

步骤4:  
按“Update”按钮，并开始通过所有连接的Nu-Link设备更新程序。

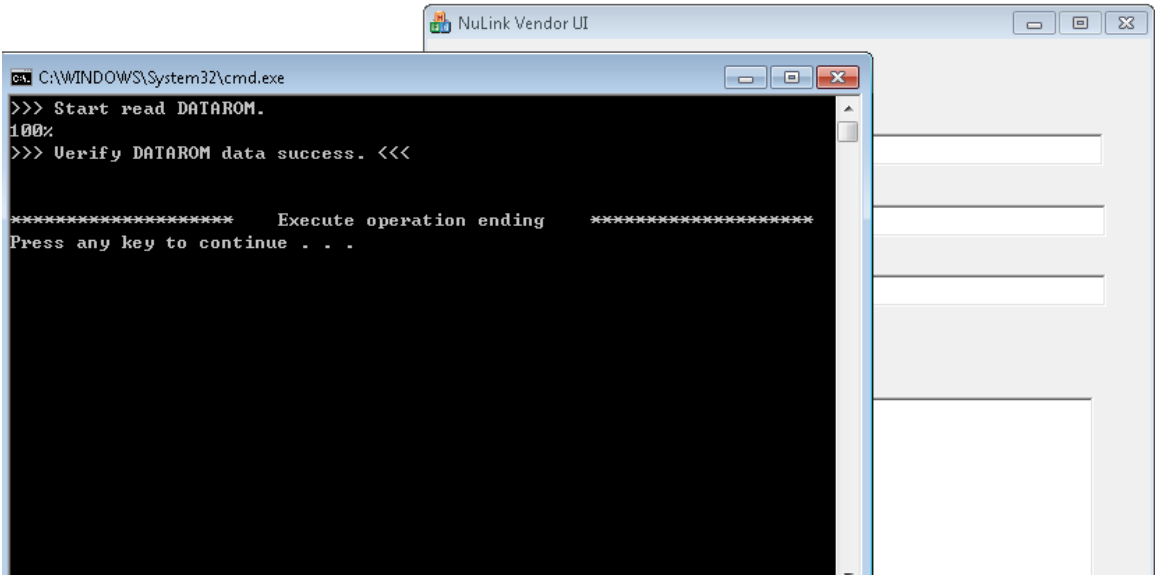


图 3-4 利用 Vendor UI 编程数据目标芯片  
所有连接的Nu-Link设备ID将被列在“Update”按钮下面的列表框中。

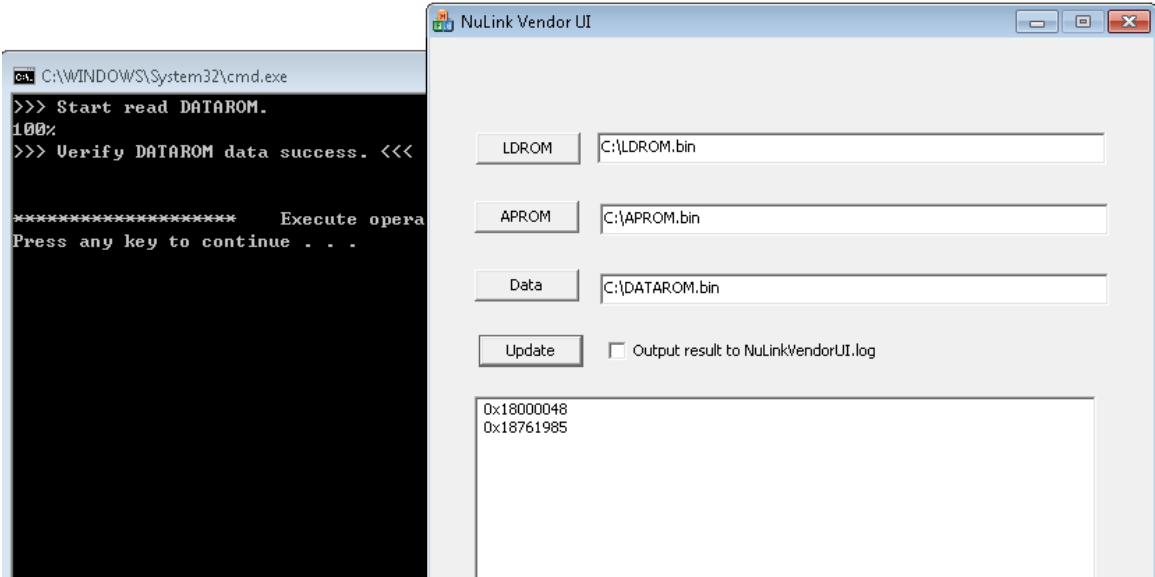


图 3-5 显示所有连接的Nu-Link设备ID  
每个bin文件的更新流程擦除 ->编程 ->验证。  
所以，上面的例子中的更新流程为擦除LDROM ->编程LDROM ->验证LDROM ->擦除APROM  
->编程APROM ->验证APROM ->擦除DATAROM ->编程DATAROM ->验证DATAROM。  
**注：**  
要记录更新流程的结果更新流程信息，启用“Output result”选项。

4 特定的系列

4.1 支持特定系列

对于特定系列，使用NuLink Command Tool有一些不同的命令格式，请按照说明获得以下用法。

- 1. 在Windows中打开命令提示程序，找到NuLink Command Tool的安装路径。
- 2. 查找要执行工具的特定文件名，并显示命令格式和示例。

执行文件名	芯片系列
NuLink_M2354.exe	M2354系列
NuLink_M2351_M261.exe	M2351系列和M261系列
NuLink_8051OT.exe	8051 1T系列
NuLink_KM1Mx.exe	KM1M7A/B/C系列和KM1M4B系列
NuLink_M460_M2L31.exe	M460系列, M2L31系列和M2U51系列
NuLink_M55M1.exe	M55M1系列

## 5 版本历史

日期	版本	描述
2015.07.21	1.30	初版
2016.03.16	1.31	新增转储数据的命令。
2016.07.22	2.00	新增断开与目标芯片连接的命令。
2017.01.13	2.01	新增SPROM、KPROM擦除、编程和验证的命令。
2017.06.23	2.02	新增 SPI flash 擦除、编程和验证的命令。
2018.06.22	2.04	支持M2351系列。
2018.07.25	2.05	更新SPI flash 命令格式内容。
2020.04.30	3.02	新增写入软件序列号到目标芯片的命令。
2020.06.29	3.03	1. 新增 all_nulink 参数对每个连接目标芯片执行相同的命令。 2. 更新Nu-Link Vendor UI内容。
2020.09.14	3.04	支持M2354系列。
2021.06.04	3.07	新增parallel参数对每个连接的目标执行相同的命令。
2021.12.06	3.08	支持M460系列。
2023.05.02	3.12	支持M2L31系列和更新命令选项。
2024.05.31	3.16	支持M55M1系列。

### 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.*