



# **G+ Mass Production Tool For GPL329xx 使用手册**

V1.0.1 – 2011 年 8 月 12 日



### Important Notice

Generalplus Technology reserves the right to change this documentation without prior notice. Information provided by Generalplus Technology is believed to be accurate and reliable. However, Generalplus Technology makes no warranty for any errors which may appear in this document. Contact Generalplus Technology to obtain the latest version of device specifications before placing your order. No responsibility is assumed by Generalplus Technology for any infringement of patent or other rights of third parties which may result from its use. In addition, Generalplus products are not authorized for use as critical components in life support devices/systems or aviation devices/systems, where a malfunction or failure of the product may reasonably be expected to result in significant injury to the user, without the express written approval of Generalplus.

# 目 录

页码

G+ MASS PRODUCTION TOOL FOR GPL329XX.....	1
使用手册.....	1
1 TOOL 简介.....	6
1.1 USB 设备端口映射对话框 .....	6
1.2 主窗口界面 .....	7
1.2.1 脚本配置 .....	7
1.2.2 设备枚举 .....	8
1.2.3 端口映射 .....	9
1.2.4 开始下载 .....	9
1.3 步骤配置介绍.....	10
1.3.1 读步骤.....	10
1.3.2 写步骤.....	11
1.3.3 代码区与数据区步骤 .....	12
1.3.4 Send Nand Database .....	13
1.3.5 代码区头文件解析步骤.....	14
1.3.6 引导区头文件解析步骤.....	15
1.3.7 BootHeader 烧录步骤 .....	16
1.3.8 BootLoader 烧录步骤.....	17
1.3.9 Chip Erase 步骤.....	18
1.3.10 Erase 步骤 .....	19
1.3.11 获取 Nand 信息步骤.....	20
1.3.12 主机端延迟步骤 .....	21
1.3.13 跳转步骤 .....	22
1.3.14 Nand Flush 步骤 .....	23
1.3.15 寄存器写步骤.....	24
1.3.16 Dram Calibration 步骤.....	25
1.3.17 用户自定义命令步骤 .....	25
1.3.18 Lua Script.....	27
1.4 执行（烧录） .....	28
1.4.1 单台烧录 .....	28



1.4.2	多台烧录 .....	29
1.5	CHECK FOR UPDATE(S) .....	30
2	烧录档存放规则 CONF (SCRIPTFILE) 和 BIN / HDB.....	34
2.1	后缀名.CONF 档存放规则 .....	34
2.2	后缀名.BIN/.HDB 档存放规则 .....	34



## Revision History

Revision	Date	By	Remark
1.0.1	07/18/2011	Phoebe	1. 支援 Lua Script 2. 输入值支援 10 进制和 16 进制 3. 添加 Auto Mode, 通过 Dram Calibration、Send Nand Database、App Header Parsing 步骤, 在 download 过程中使相关内容进行自动调整。
1.0.0	05/12/2011	Eric	1st Version

---

## 1 Tool 简介

---

当用 G+Code packer 包出 binary 文件或 hdb 文件后, 用户就可以呼叫 G+ Mass Production Tool For GPL329xx 来对指定的 memory type 进行烧录. 尽管如此, 用户需要先配置好整个烧录的流程, 即透过此软体进行下载步骤的配置.

所以, 在执行烧录的动作之前, 用户应确保下载脚本文件已正确创建, 此文件的默认创建路径在与可执行档平级的名为 **conf** 的文件夹下, 档名由用户自行拟定. 我们将在后续的章节中来更详细的描述此脚本文件的创建过程.

### 1.1 USB 设备端口映射对话框

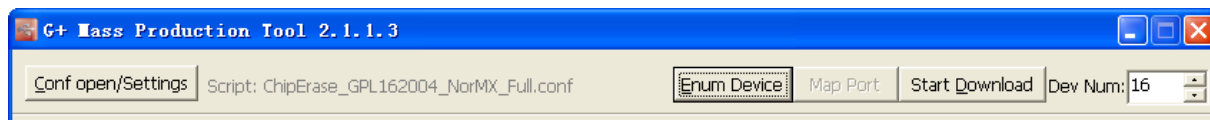
目前, 此 Tool 会透过这个设备端口映射对话框来对 PC 机上所装载的 Generalplus USB 设备进行控管, 假定一开始, Tool 没有检测到 GP USB 设备, 这时用户接入一台 GP USB 设备, Tool 如果枚举到此设备, 就会跳出一个对话框, 来询问用户是否要将此设备显示在 Tool 的主界面上, 同时刷新端口映射表. 若用户选是, 将会执行上述行为, 若选否, 那 Tool 将取消上述操作, 不会将 tool 枚举到的设备放到主界面的设备列表中, 当然端口映射表也不会随之更新. 下面就是端口映射表的所长的样子, 里面有棵树.



用户能透过此树来很清楚的查看到当前接入 PC 的所有设备状况，它可以很明确反映用户，说某个设备挂接在某个集线器的某个端口上，因为 tool 会在每个集线器的每个端口上打印出当前端口所接入的设备信息。有了这些信息，如果用户可以勾选对应的设备选项，表明用户想对此台设备对行烧录，当用户点击 **Set** 按钮后，如果用户选中的设备是能跟此 Tool 沟通的 USB 设备，那么此设备信息将会列在 Tool 的主界面的设备列表中。当然,如果用户选中的设备不是 GP USB 设备，那么，主界面的设备列表也不会将用户在端口映射表选中的设备信息显示出来。

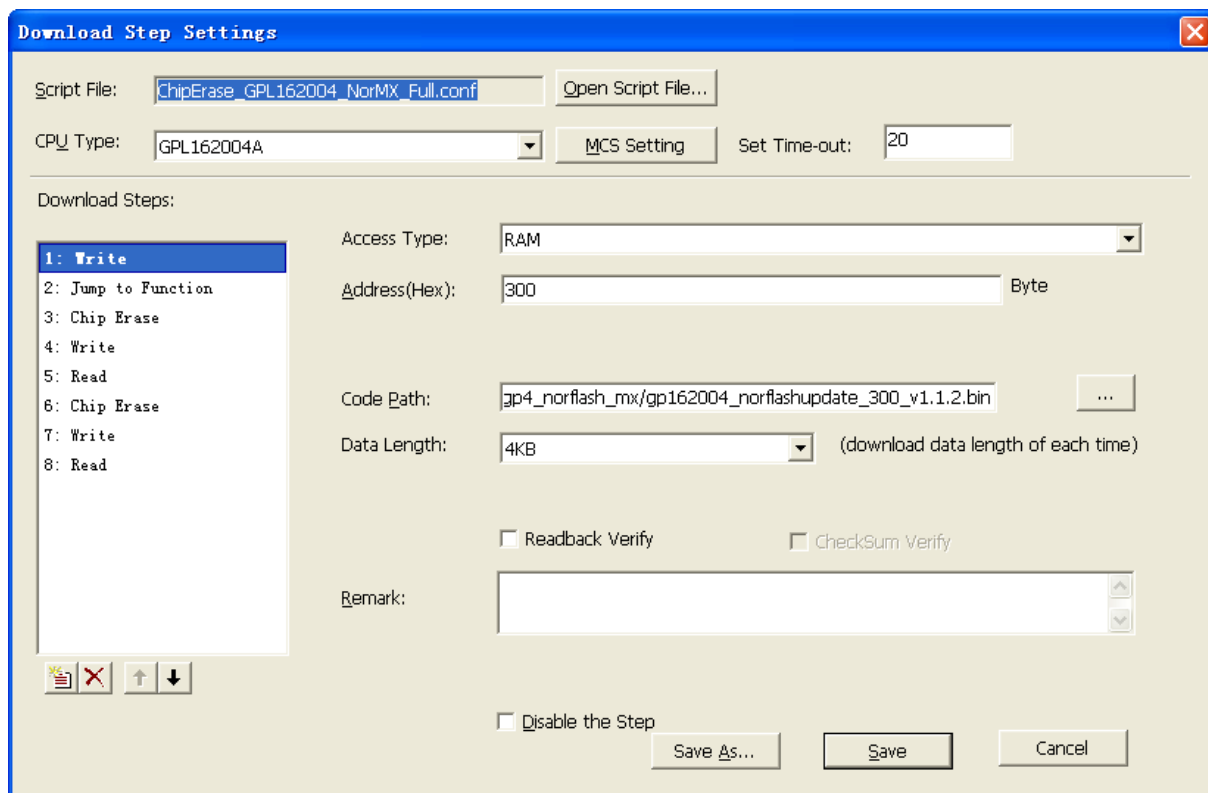
## 1.2 主窗口界面

当用户配置好端口映射表后，将会返回到主对话框，下面就是主对话框的界面：



### 1.2.1 脚本配置

点击 **conf open/Settings**，打开 **Download Step Settings** 对话框，可查看或进行下载脚本配置：



Open Script File: 选择已经配置好的脚本。

CPU Type: 选择对应设置的 CPU 类型。

MCS Setting: 进行 MCS 设置。

Download Steps: 打开已有脚本，可查看下载步骤；或创建下载步骤。双击各步骤，可对其进行重命名。右边界面则显示每个步骤的对应界面。

: 添加某个步骤；

: 删除选中步骤；

: 调整各个步骤的顺序。

Disable the Step: 使能步骤，勾选，则该步骤失效,实际执行过程中,会跳过它直接执行后面的步骤。

Save As: 将脚本另存为。

Save: 保存当前脚本。

Cancel: 取消当前操作，返回主操作界面。

## 1.2.2 设备枚举

Tool 内部有自动侦测并连接设备的功能，所以一般不会使用到按钮。但当用户不确定

Tool 本身自动连接功能是否有生效时，则可尝试点击该按钮，来检查设备是否被识别到。若跳出 “No Device be Found”，则表示未识别到设备。

### **1.2.3 端口映射**

弹出 USB Port Mapping 对话框。

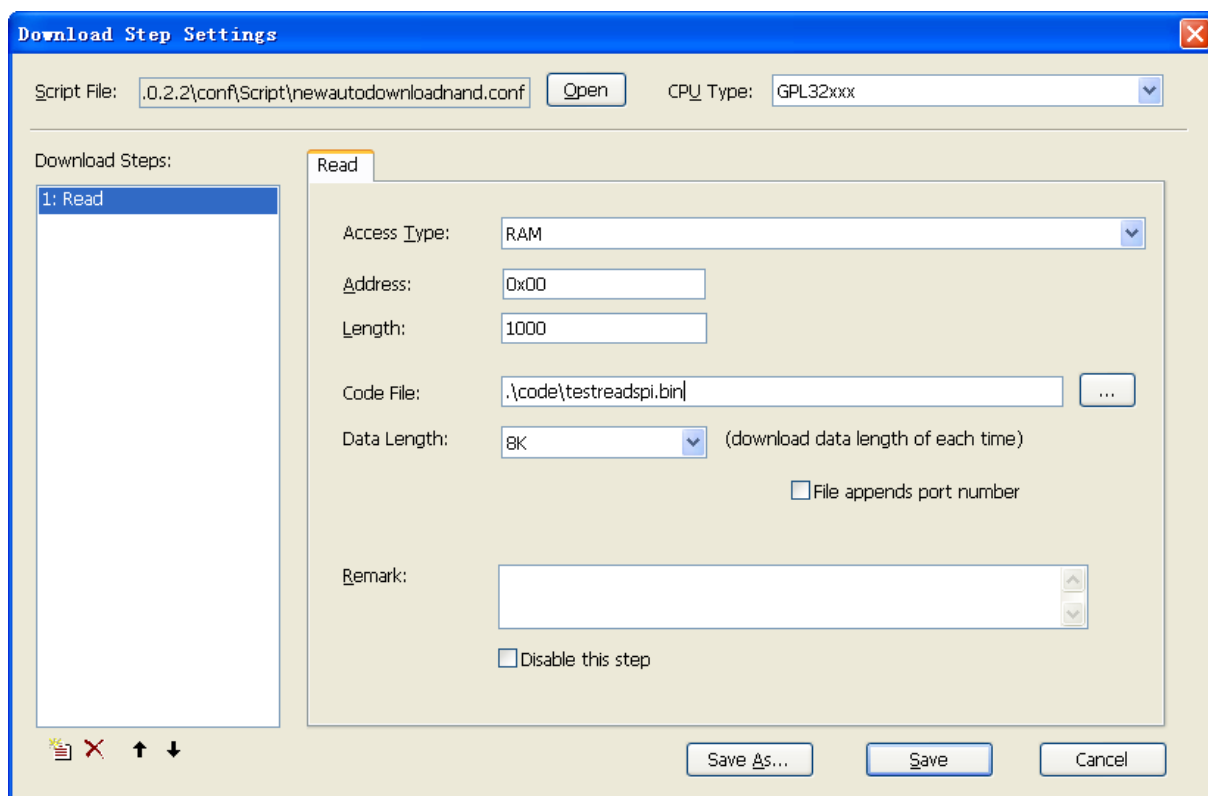
### **1.2.4 开始下载**

设置完成后，点击 “**Start Download**”，进入下载流程。Tool 会对设备列表中所有处于有效就绪状态的设备，按照脚本文件的配置进行烧录，主界面上将及时反映每台设备的烧录状况，如当前步骤信息及进度显示等，当烧录完成时，进度条显示为 **100%**且有 **Download finished** 字样。选中某设备双击，可查看该设备烧录的日志信息，也可以在 Tool 安装目录的 log 文件夹中查看日志信息。

下载过程中，点击 **Stop Download**，则终止下载。

## 1.3 步骤配置介绍

### 1.3.1 读步骤



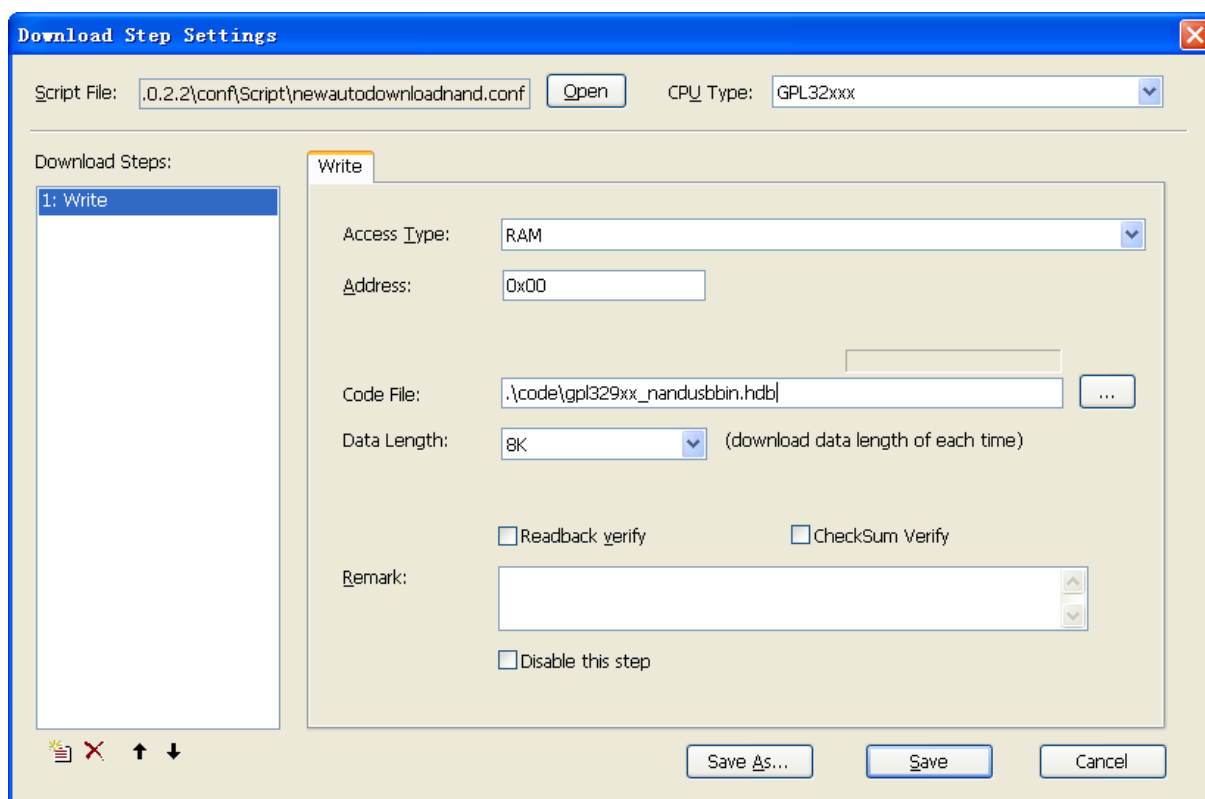
当用户在脚本配置画面左侧的步骤列表中切换到读这个步骤时，右边将会显示此步骤所需要填入的一些参数，这些参数主要包括存储器的访问的类型，目前主要支援 RAM, SPI Flash, Nor Flash, Nand Flash App Area 及 Nand Flash Data Area. 右边的组合框中将会列出以上这几种存储器介质. 下面将说明下每个参数所代表的意义：

- ✓ 存储器访问类型：在此组合框中的下拉列表，可以指定用户当前想要对哪种存储器读数据。
- ✓ 地址：让用户可以指定从存储器的那个位置开始读数据，地址输入支援 10 进制或 16 进制。注意，不同的存储器型别会对应不同的地址单位，默认为 Byte。
- ✓ 长度：指定要从选定的存储器的指定地址读取数据的总长度。注意，此处所填充的数据是十六

进制格式的。

- ✓ 文件：在磁碟上指定个文件，用于保存设备上指定的存储器中读取的数据信息
- ✓ 单笔数据长度：它表明每次找设备指定存储器中读取数据长度，因为当我们将上面长度设定很大值时，比如 1MB，而将此数据长度选为 4KB，那表示 Tool 会每次找设备要 4KB 的数据，而总长度是 1MB，所以 Tool 总共会找设备要 256 次数据，才能将所有数据读回来。

### 1.3.2 写步骤

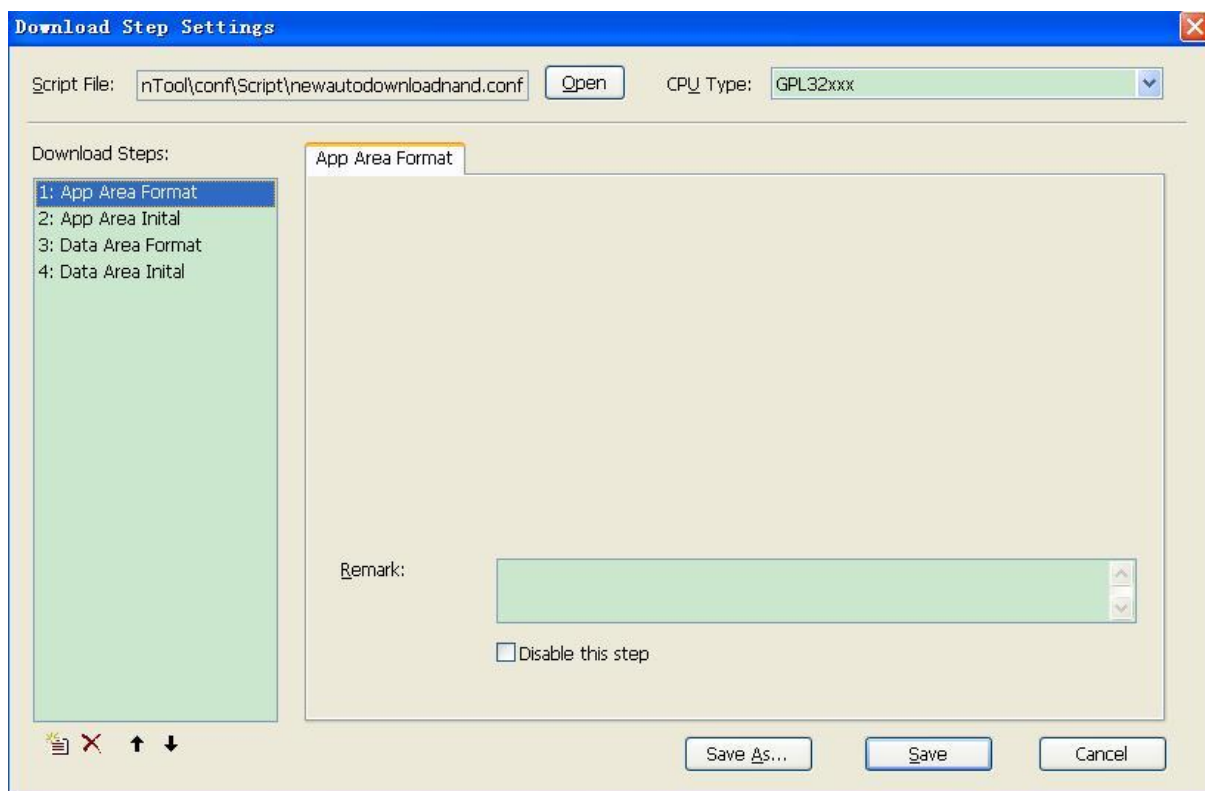


写步骤与读步骤在参数的配置上有很多相似的地方，比如存储器的选择，它同样会支援五种类型的存储器。不同的是，该步骤由用户指定的烧录文件可是二进制的 bin 文件也可以是由 Code Packer 包出来的 Header Data Block 的 hdb 文件。如果用户选择烧录的是扩展名为 hdb 的文件，此页面还会将该文件的所记录的 Tool ID 及 Tool 的版本号显示出来。这个主要是想提醒固件开发人员当前所使用的 hdb 档是否是他真正想要烧录的文件。

- ✓ 存储器访问类型：在此组合框中的下拉列表，可以指定用户当前想要对哪种存储器读数据。

- ✓ 地址: 让用户可以指定从存储器的那个位置开始写数据, 输入值支持 10 进制和 16 进制. 以 0x 开头为 16 进制, 否则为 10 进制. 注意, 不同的存储器型别会对应不同的地址单位, 默认为 Byte. (Byte, Sector, Block)
- ✓ 文件: 在磁碟上选取一个现有的文件, 稍后会将此文件中的内容写到设备上指定的存储器中
- ✓ 单笔数据长度: 它表明每次从 Tool 端发送指定数据长度到设备指定存储器中, 因为当我们将上面长度设定值时, 比如 1MB, 而将此数据长度选为 8KB, 那表示 Tool 每次发送 8KB 的数据到设备指定存储器中, 而总长度是 1MB, 所以 Tool 端总共会向设备发送 128 次数据, 才能完成本次烧录.

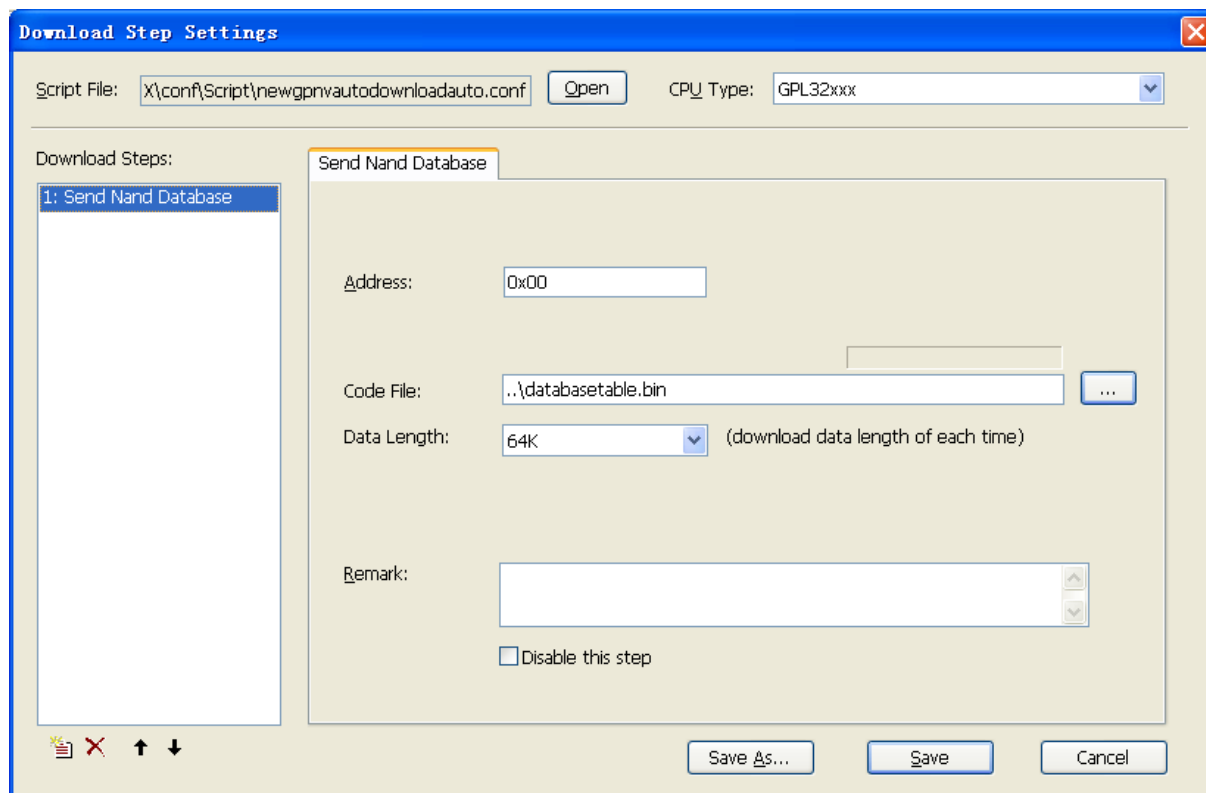
### 1.3.3 代码区与数据区步骤



这几个步骤主要是用于对 Nand 这颗存储器中的代码区或者数据区, 进行一个初始化工作, 或者对这些区域进行格式化操作. 这些步骤属于 Step 命令, 这几个 Step 命令有着相同的用户界面, 但是不同的 Step 命令步骤具有不同的涵义. 这些步骤在执行过

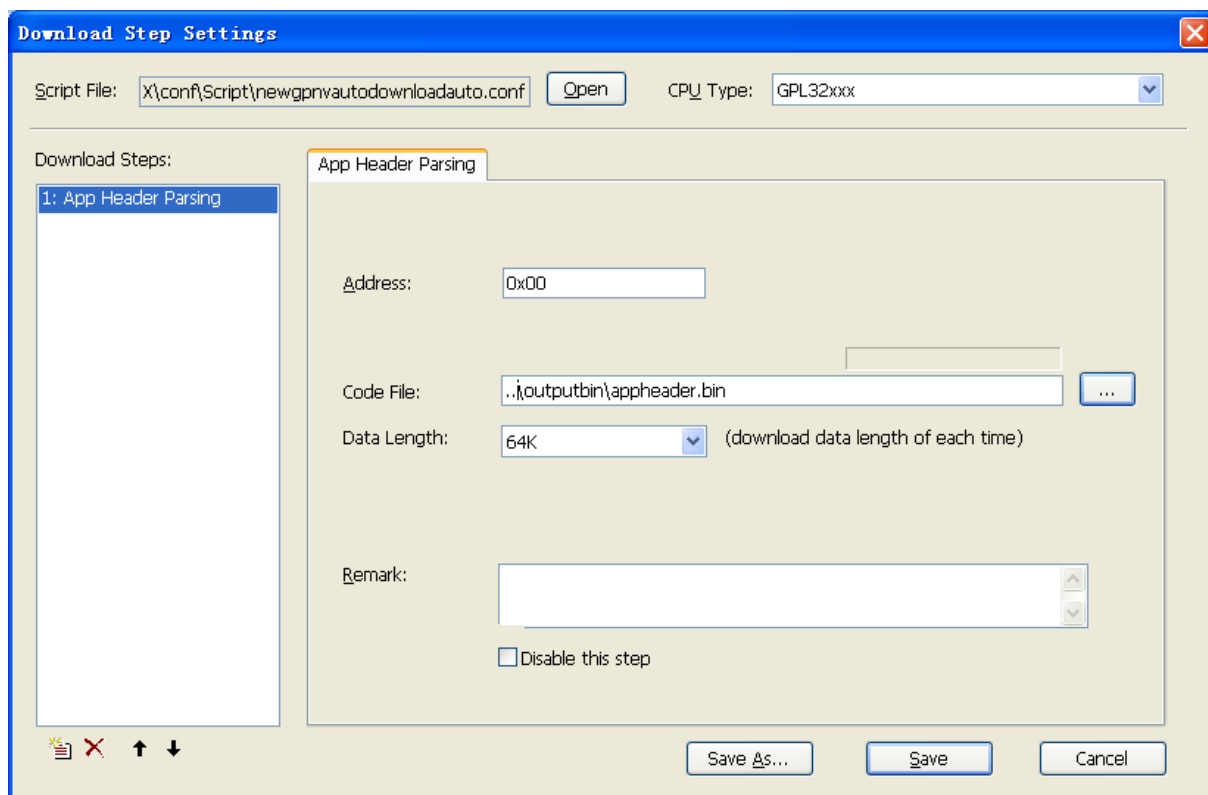
程中,常常会对设备进行一个 polling 的动作,会不停的询问当前的步骤执行的状况,看是否有成功完成.

### 1.3.4 Send Nand Database



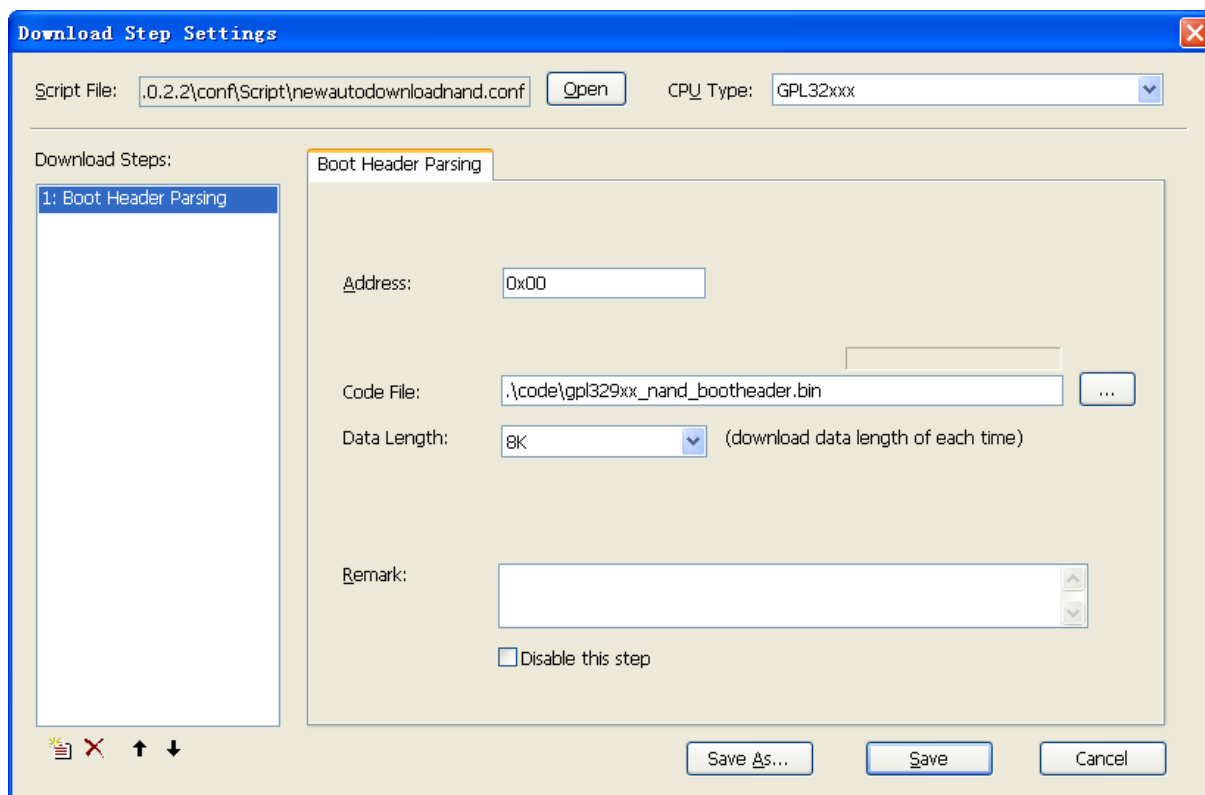
这个步骤主要是透过 0xFF55 这个命令向设备发送一个 DataBaseTable 的二进制 bin 文件,设备自动判断当前使用的 Nand 型号,并解析 bin 档,查询对应 Nand 的型号及相关信息。

### 1.3.5 代码区头文件解析步骤



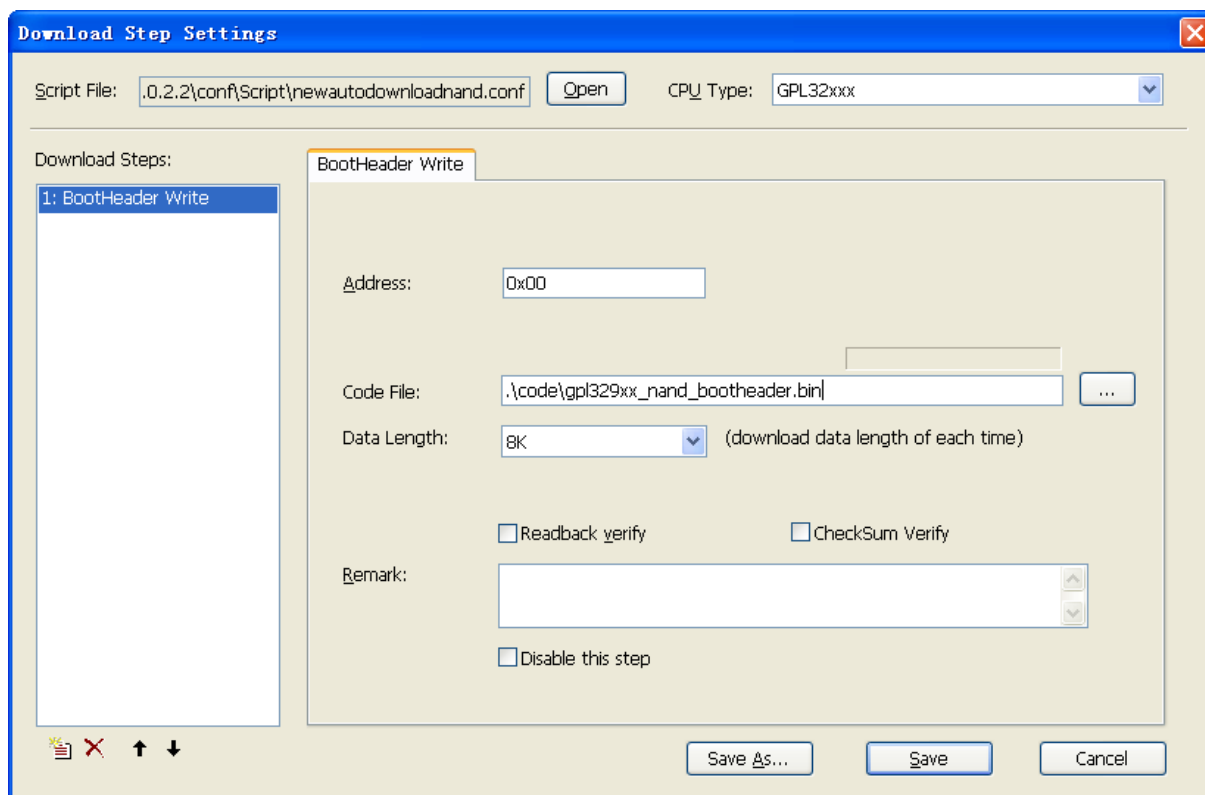
这个步骤主要是透过 0xFF56 这个命令向设备发送一个 AppHeader 的二进制 bin 文件, 设备收到命令后,自动解析 bin 文件,并根据解析出的结果自动调整写 App Area 时相关操作。

### 1.3.6 引导区头文件解析步骤



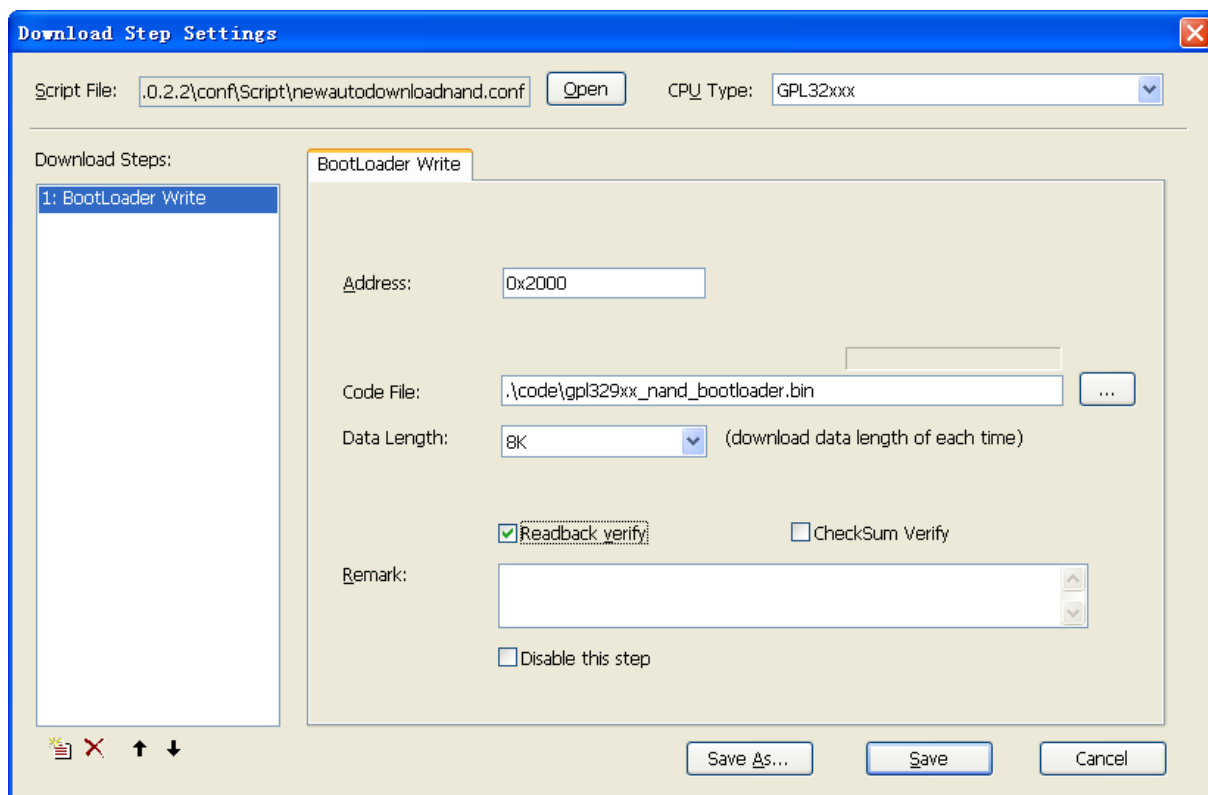
这个步骤主要是透过 0xFF51 这个命令向设备发送一个 BootHeader 的二进制 bin 文件，设备收到命令后，并对接收到 bin 文件进行解析，设备从而知道其引导区的 Block 总数及整个引导区的大小。

### 1.3.7 BootHeader 烧录步骤



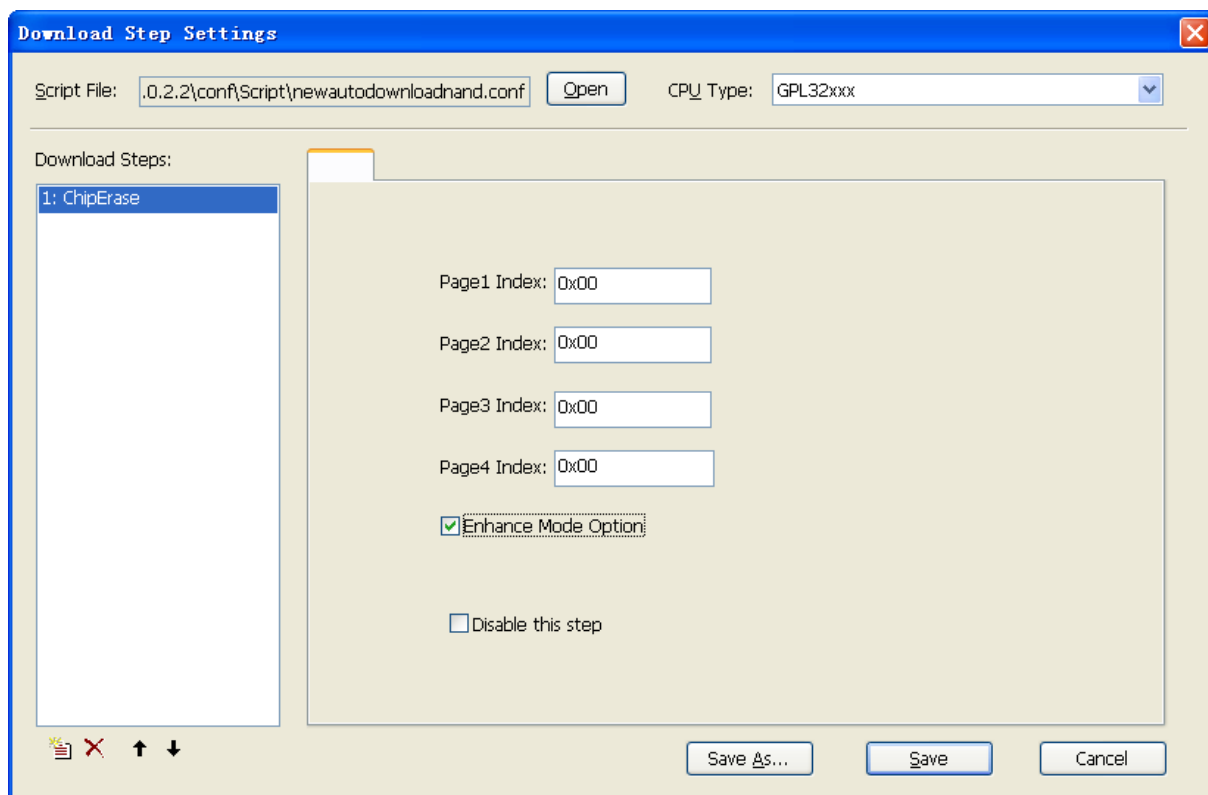
文件进行烧录，在上述的步骤中设备已经得知当前引导区的相关信息，此时这个步骤会将此 **BootHeader** 文件写到设备的引导区中。另外，该步骤还提供了两种写入数据正确性的验证方法：一个是 **Readback** 验证，另一个是 **Checksum** 验证。两种方法都可以检验当前步骤中的数据文件烧到设备中的正确性与否。客观上给固件开发人员提供了一个错误检查的窗口。因为往往下载成功，并不代表说真正写入到设备中的数据也是正确的，有可能因为地址的原因导致设备 **Boot** 后达不到预期效果或者根本都 **Boot** 不起来，所以以防万一，使用者可以勾选当前步骤所要进行哪种检验机制。另外，此命令也属于 **Step** 命令。

### 1.3.8 BootLoader 烧录步骤



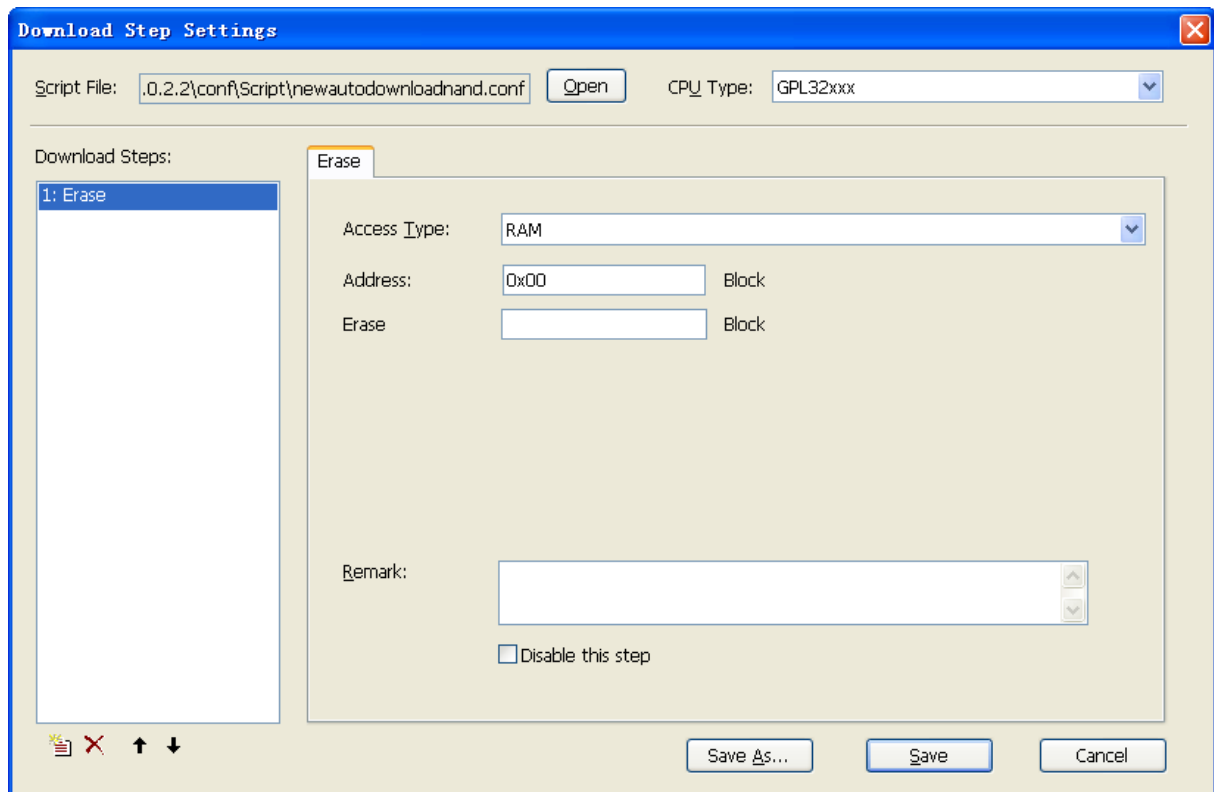
这个步骤与 BootHeader 烧录步骤很相似，这里就不再赘述，该步骤同样也属于 Step 命令。

### 1.3.9 Chip Erase 步骤



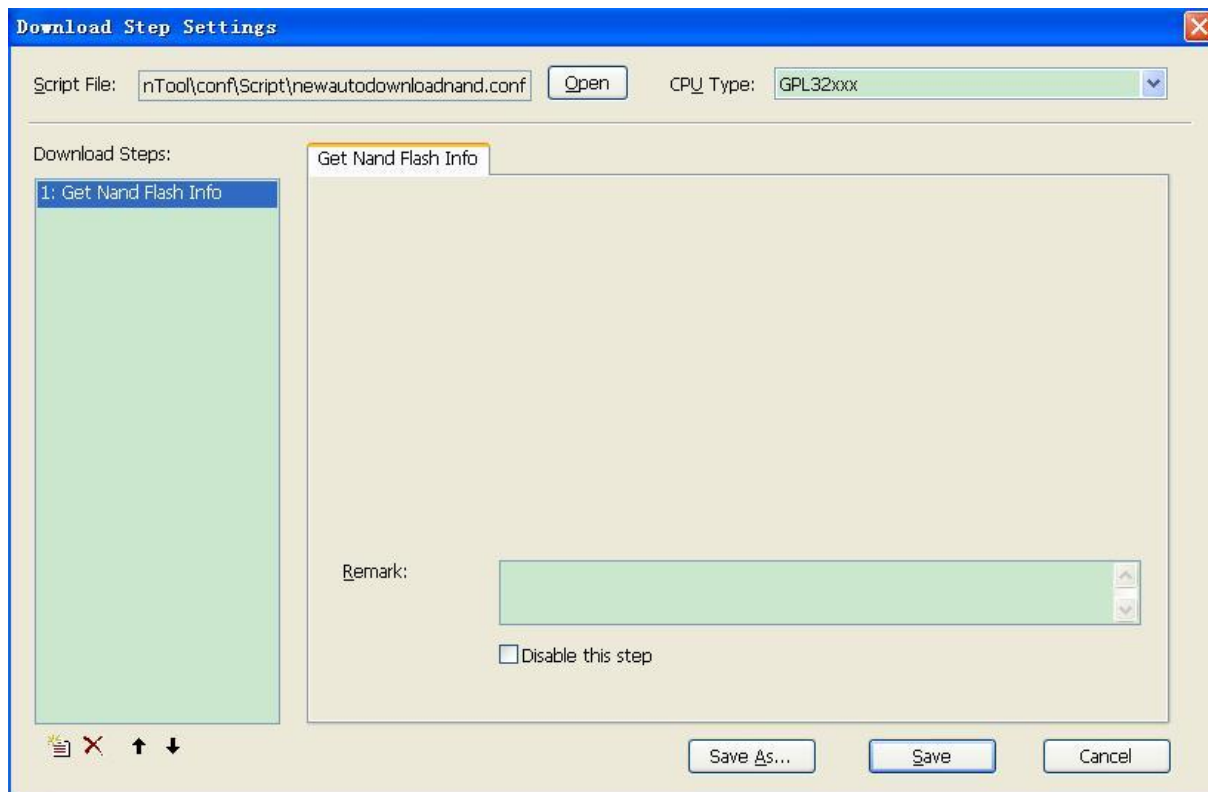
- ✓ 这个步骤主要是对 Nand Flash 的整个 Block 进行擦除动作，默认为普通擦除模式。
- ✓ 加强模式：勾选“Enhance Mode Option”，在 Page index 中输入想要检测的 Page 数，则会对指定 page 使用加强模式擦除。

### 1.3.10 Erase 步骤



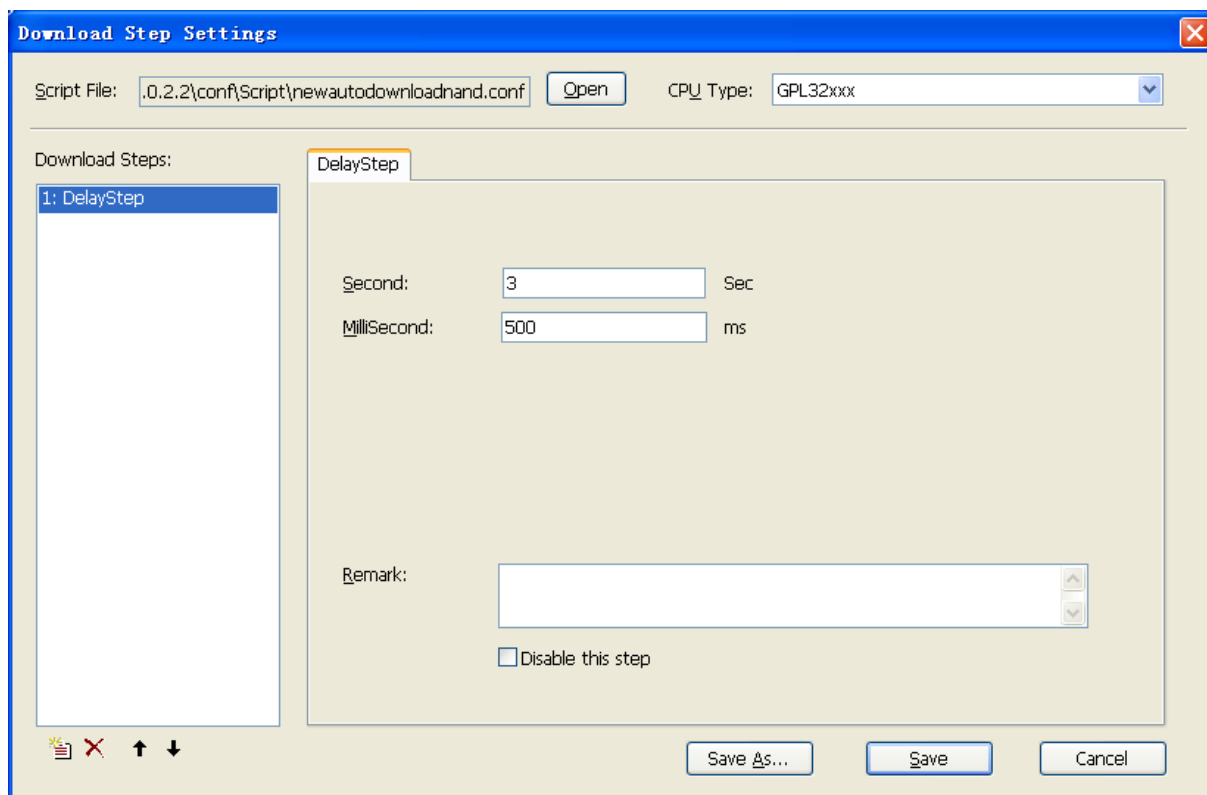
- ✓ 存储器访问类型: 在此组合框中的下拉列表, 可以指定用户当前想要进行擦除的存储器类型.
- ✓ 地址: 指定要擦除的起始 Block.
- ✓ Erase: 指定要擦除的 Block 的大小.

### 1.3.11 获取 Nand 信息步骤



当主机与设备进行通信时, Tool 就会透过执行此步骤, 来获取 nand 的有关资讯, 换句话说, 我们可以对 Nand 的相关属性进行监视, 并将这些信息用第三方软体进行输出, 如打印在 Uart 串口调试工具中.

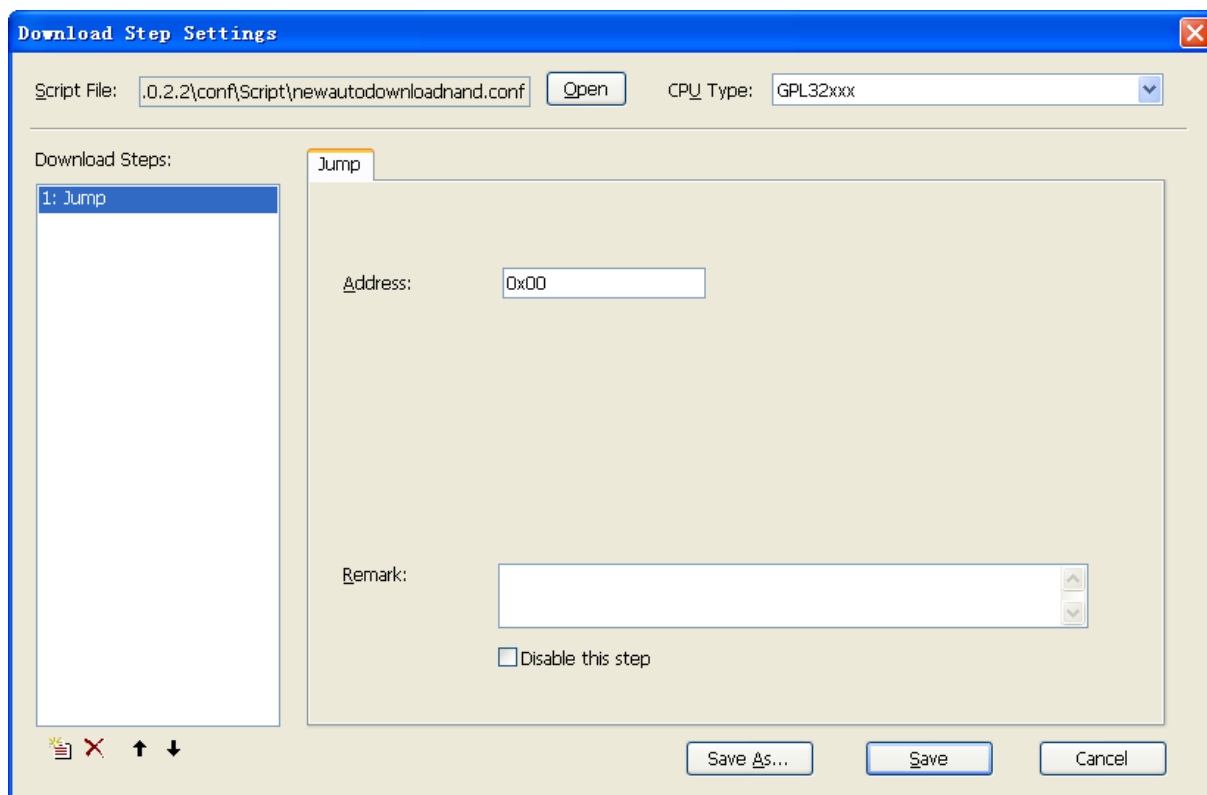
### 1.3.12 主机端延迟步骤



这个命令主要是用于 PC 端，并不会真正与 USB 设备进行沟通。

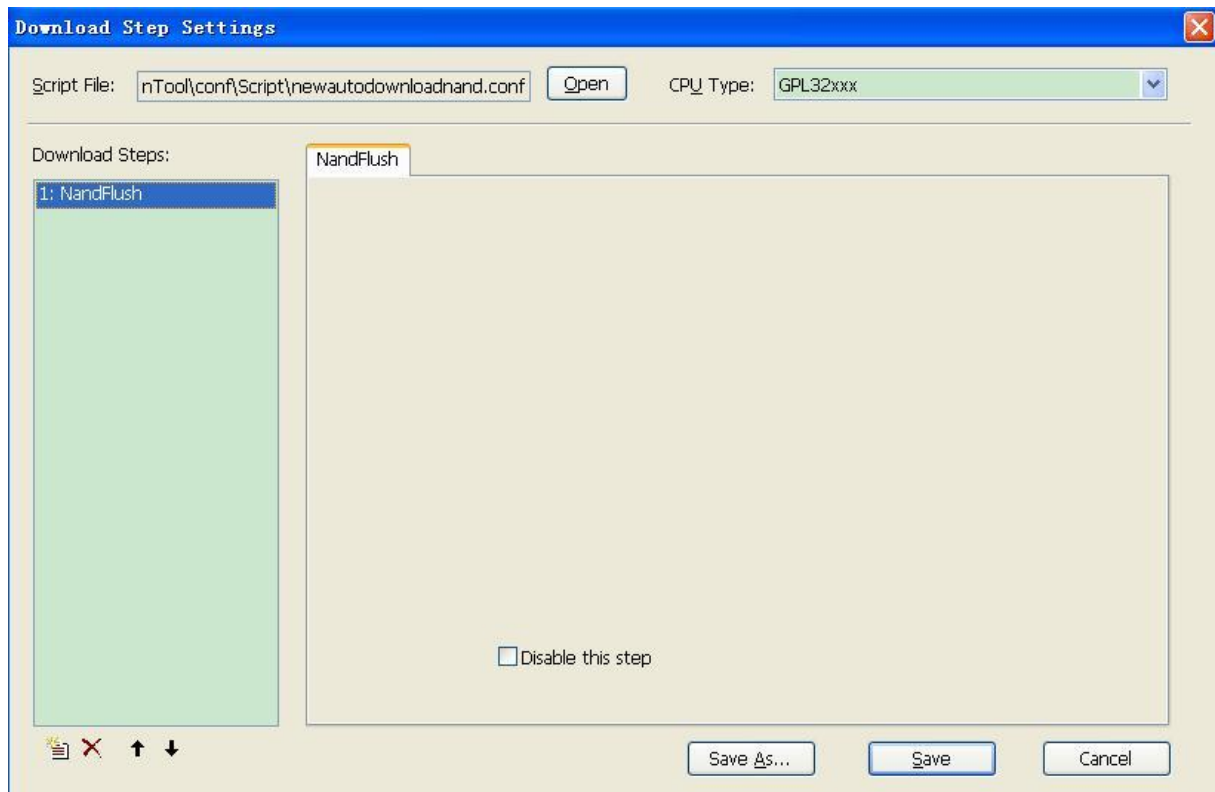
- ✓ 指定使用者想要延迟的时间。如上，延迟时间为  $3 \times 1000 + 500 = 3500$  ms。

### 1.3.13 跳转步骤



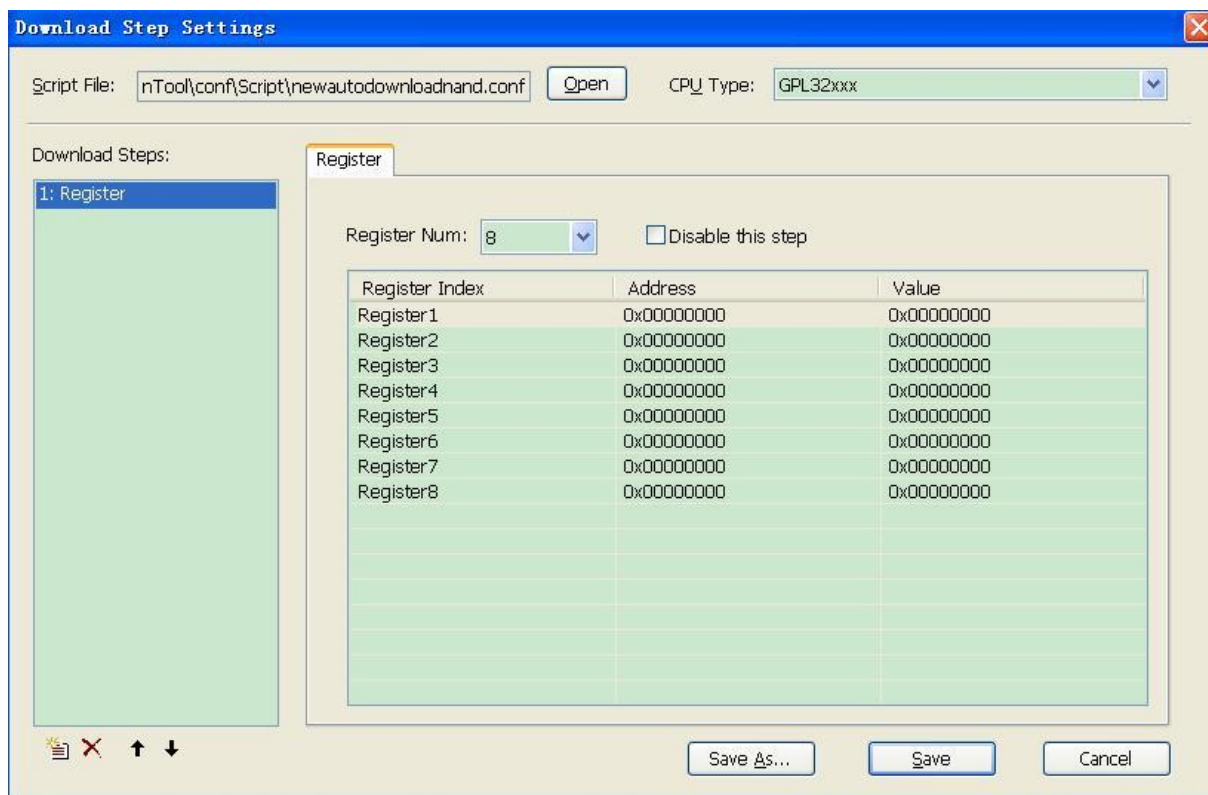
这个步骤主要是将设备的 PC 指针设定到指定的地址,使其跳转到这个内存地址,一般它是 iRam 或 SDRam 中的某个地址.

### 1.3.14 Nand Flush 步骤



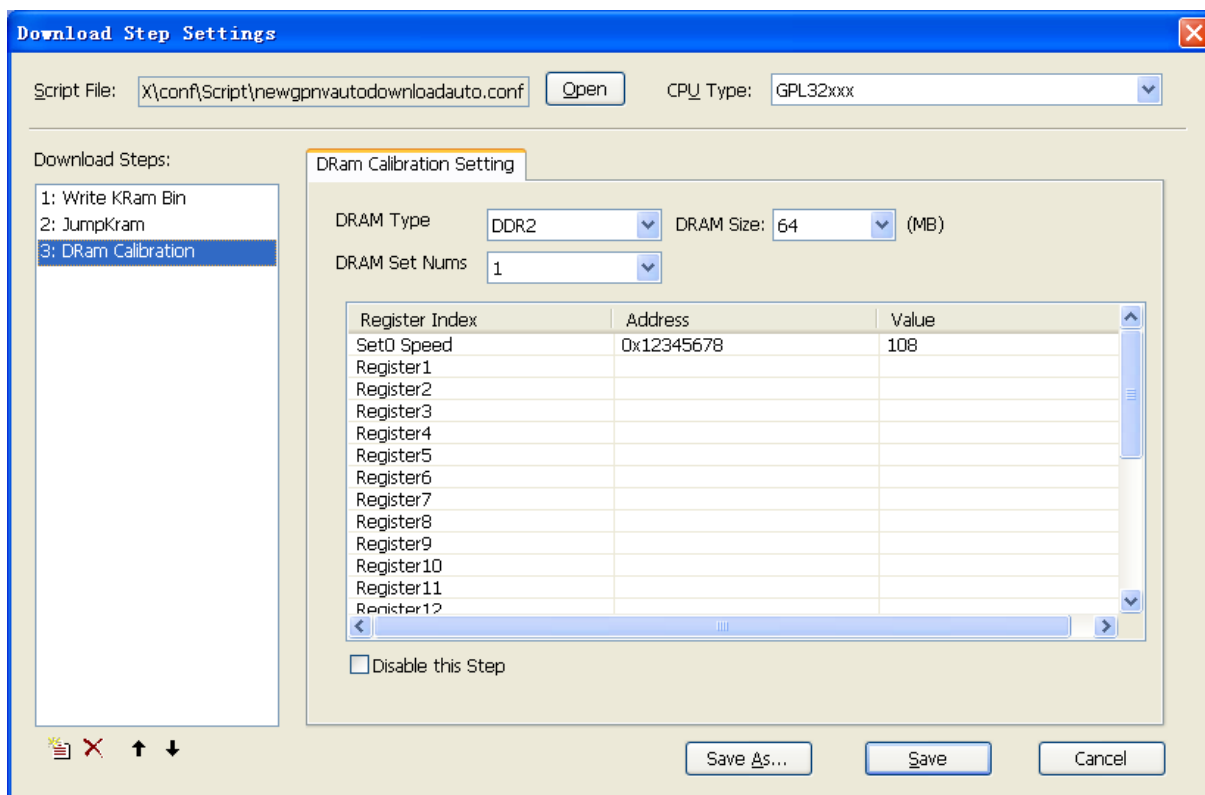
这个步骤主要是用于将缓冲区的数据刷新，使其对数据写入 Nand Flash 中。

### 1.3.15 寄存器写步骤



这个步骤主要是用于向指定的寄存器写入用户设定的值,Tool 会提供 31 组配置供用户设置,这也是目前支持的最大写入寄存器组数. 另外,我们提供了另一个步骤名称,其名字为读寄存器步骤. 当然,它与写寄存器步骤有很多相似之处,两个步骤均使用同一大套用户界面,惟一不同是,读寄存器步骤中的 **Value** 这一列是只读的,对使用者来说是不可编辑,因为它的值将是透过命令找设备要数据,并将读回的寄存器的数值在此显示出来.另外,当设备成功执行完此步骤,便可以在日志文件查看这些读回来的寄存器资讯.

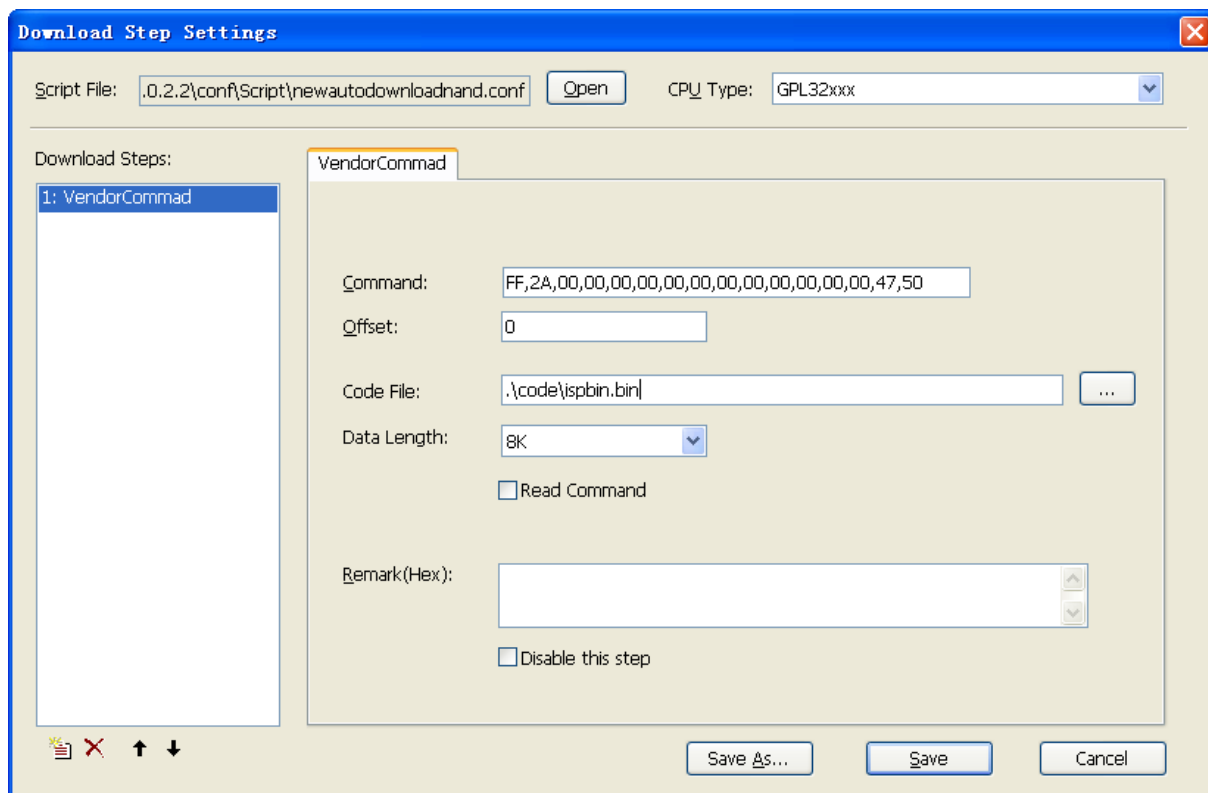
### 1.3.16 Dram Calibration 步骤



该步骤主要通过烧写 bin 档和 Jump 操作，并发送 DRAM Type、DRAM Size 和 SRAM Set Num 等信息给设备，设置在获取信息并解析后，让 IC 自动进行 Dram 校准操作。Tool 会在校准后读取相关寄存器的值并显示在界面中。

### 1.3.17 用户自定义命令步骤

用户自定义写命令：



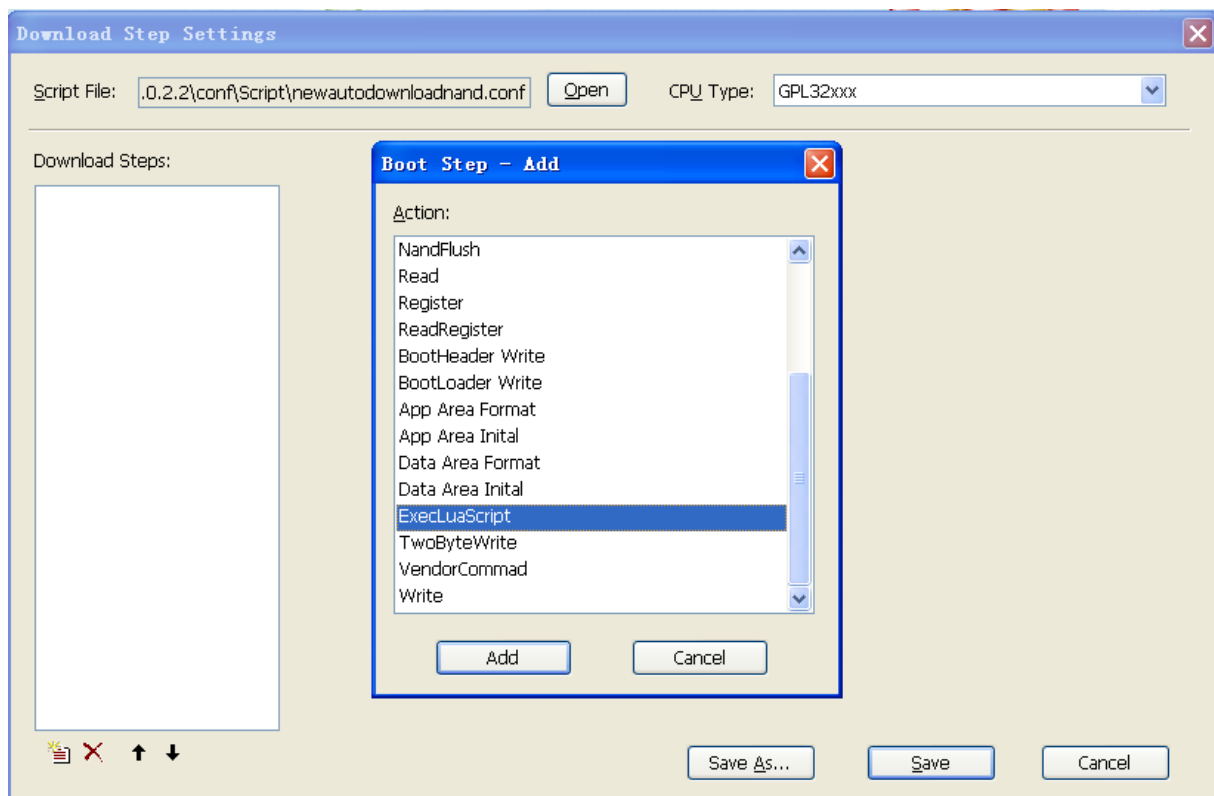
我们提供此步骤，主要是借此方式来适当增加弹性，也是希望最大限度的满足用户的需求。从而大的规范下对此步骤进行了一定程度的客制化。在这里,Tool 给用户提供了自定义写与自定义读两种用户界面相似却又功能不同的命令。一般而言, Tool 中其它步骤都可能透过此步骤来实现。所以一定程度上要求使用对其参数加以了解，尽量发挥此命令的功效;所以下面说下自定义写命令的相关参数及其用法:

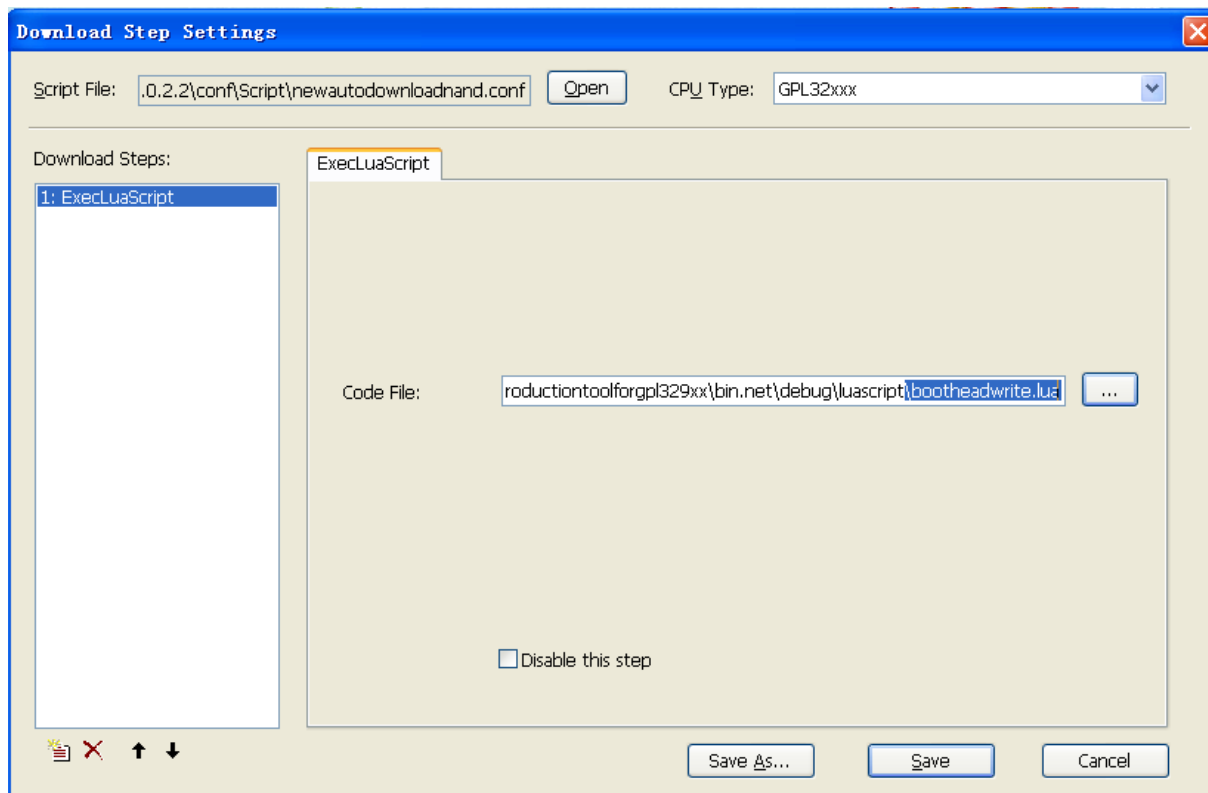
- ✓ 命令协议串: 用户可以在此栏位填入他自己想要送给设备的命令字符串，这一串字符将会直接作为 CBW 的 16 个字节的 数据送给设备，在 Tool 是不会对这 16 个字节进行任何形式的解析，当然我们假定此命令串的含义可由固件开发人员或其它使用者自行定制。command 的输入有格式限制，正确的输入方式是: 将鼠标光标定位到你要修改的某个 byte 的 command 的起始输入位置处，然后依次输入你要的 command。注意: 不能框选中 1 个 byte 的 command 再输入我们要的 command，也不能复制 command 再粘贴到其它位置，否则会造成输入的 command 串与预期的结果不一致。
- ✓ 偏移地址: 设置偏移地址的字节数。

- ✓ 烧录文件: 让用户可以指定将要烧录(或读回)的代码文件, 若此栏位为空, 则表明用户并不想烧录文件,而是作其它用途; 若不为空, 就说明使用者是要烧录 (或读回到) 此文件.
- ✓ 长度: 指定每一次读或写的数据长度.
- ✓ 读命令选项框: 主要是决定当前命令字符串是用作读还是写, 若勾选此选项, 表明当前配置是想透过此步骤来从设备读取数据, 否则, 它就是写入命令, 会根据用户设定的参数值,执行写入操作. 默认是不勾选的, 即写入步骤.

### 1.3.18 Lua Script

我们提供支援 Lua Script 的功能, 在 MP 工具中整合了对 Lua 脚本的解析, 用户可以将下载行为写为 Lua 脚本, 再将此 Lua 脚本以 Step 形式挂进 MP 工具即可。

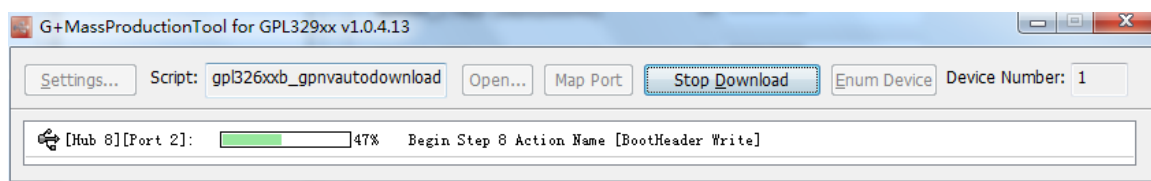




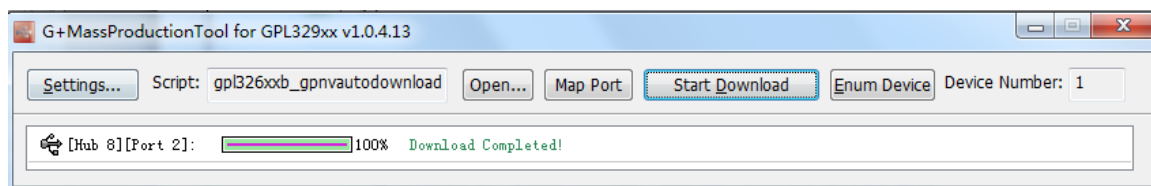
## 1.4 执行（烧录）

### 1.4.1 单台烧录

烧录步骤配置完成后，点击“Start Download”按钮，即 Download 开始，且会显示进度，下载过程中可以点击 Stop Download 按钮来停止下载。



当烧录完成之后，Stop Download 按钮的状态会自动变为 Start Download，且进度显示为 100%，用户可以直接换连另外的设备进行同样步骤的单台烧录

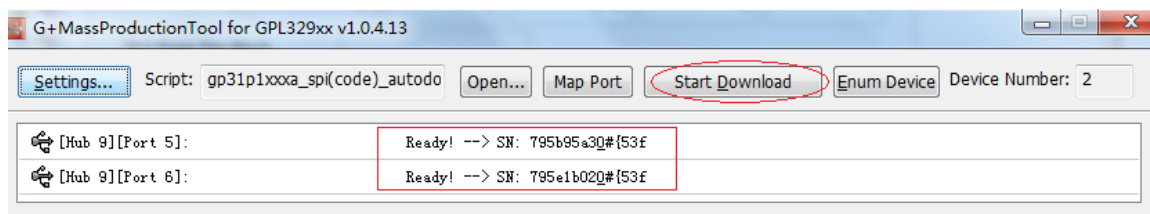


### 1.4.2 多台烧录

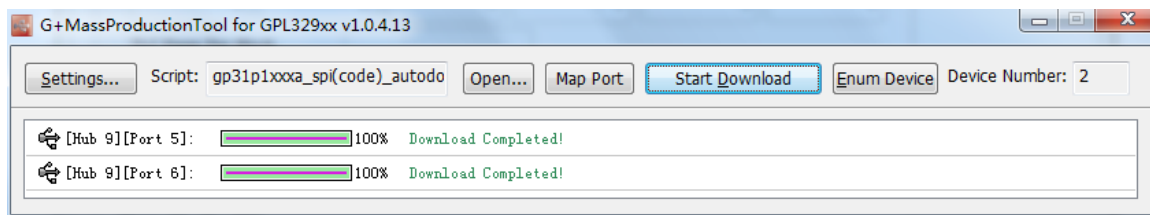
当用户需要多台同时烧录时，需遵循多台烧录准则。

1. 若用户是从 G+Code packer 包出 binary 文件或 hdb 文件后，再点击 download 呼叫出 G+ Mass Production Tool For GPL329xx。多台烧录步骤说明如下：

- (1) 开启 CodePacker 进行 DRAM calibration
- (2) CodePacker 中设定正确的 pack 参数后，完成 pack 功能
- (3) CodePacker 中按下 download 按钮，等待自动开启 MP 工具
- (4) MP 工具开启后，将 GP USB device 一台一台接入 Hub，并且依序上电（不能 Hub 串 Hub，Hub 本身是 Self power 要能提供 2A current，如 Belkin F5U234 (Hi-Speed USB 2.0 4-Port Hub)）
- (5) MP 工具 UI 显示全数的 device 后，再按下 StartDownload 按钮，烧录过程中 StartDownload 变为 StopDownload



(6) 等待所有 device 都完成烧录工作，StopDownload 会变回 StartDownload，且进度都显示为 100%完成（或可能出现烧录失败的情形），关闭所有 device 电源，重新执行步骤 4.，进行下一批烧录工作



请勿于过程中，进行任何 USB bus 的插拔

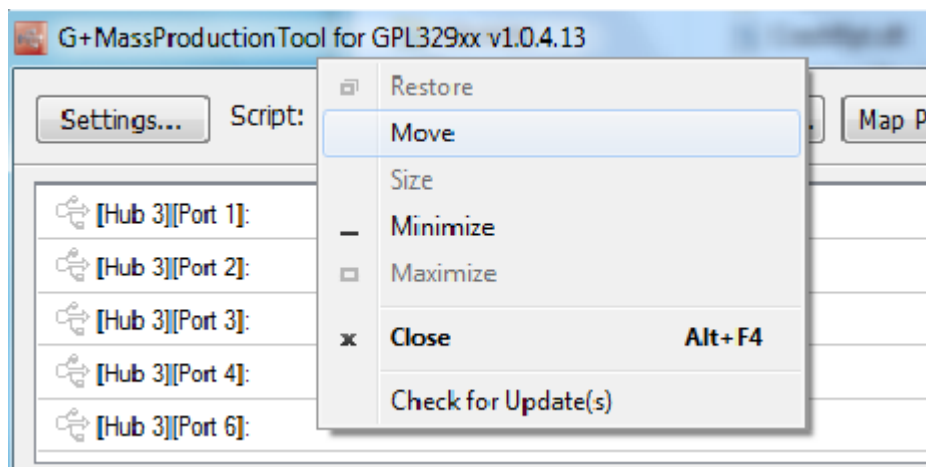
如有任何 device 完成或失败，请等待全数 device 烧录完成，才能关闭或拔除 device。

2. 若用户已有 conf、bin 等烧录档案，是直接开启的 G+ Mass Production Tool For GPL329xx 使用，那直接从上述步骤 4 开始执行多台烧录

## 1.5 Check for Update(s)

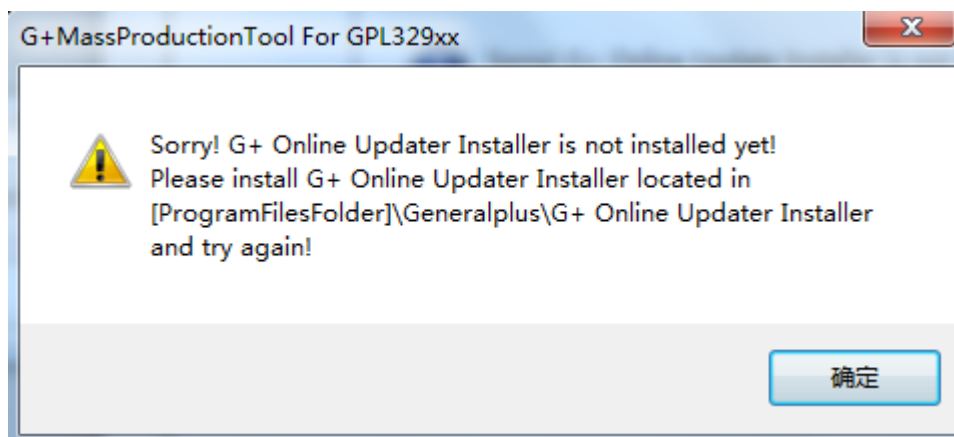
在工具栏上的 G+MassProductionTool For GPL329xx 图示上点击右键，即可看到此功能之按钮选项。此功能用于检查是否存在更新版本的工具可供使用，并允许拥有 A-Key 的用户直接下载新版工具使用。此功能的基本操作说明如下所示：

1. 在工具栏上 G+MassProductionTool For GPL329xx 图示点右键，即可看到 Check for Update(s)选项。

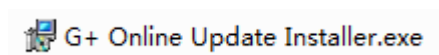


2. 点击此按钮执行 Check for Update(s)动作。

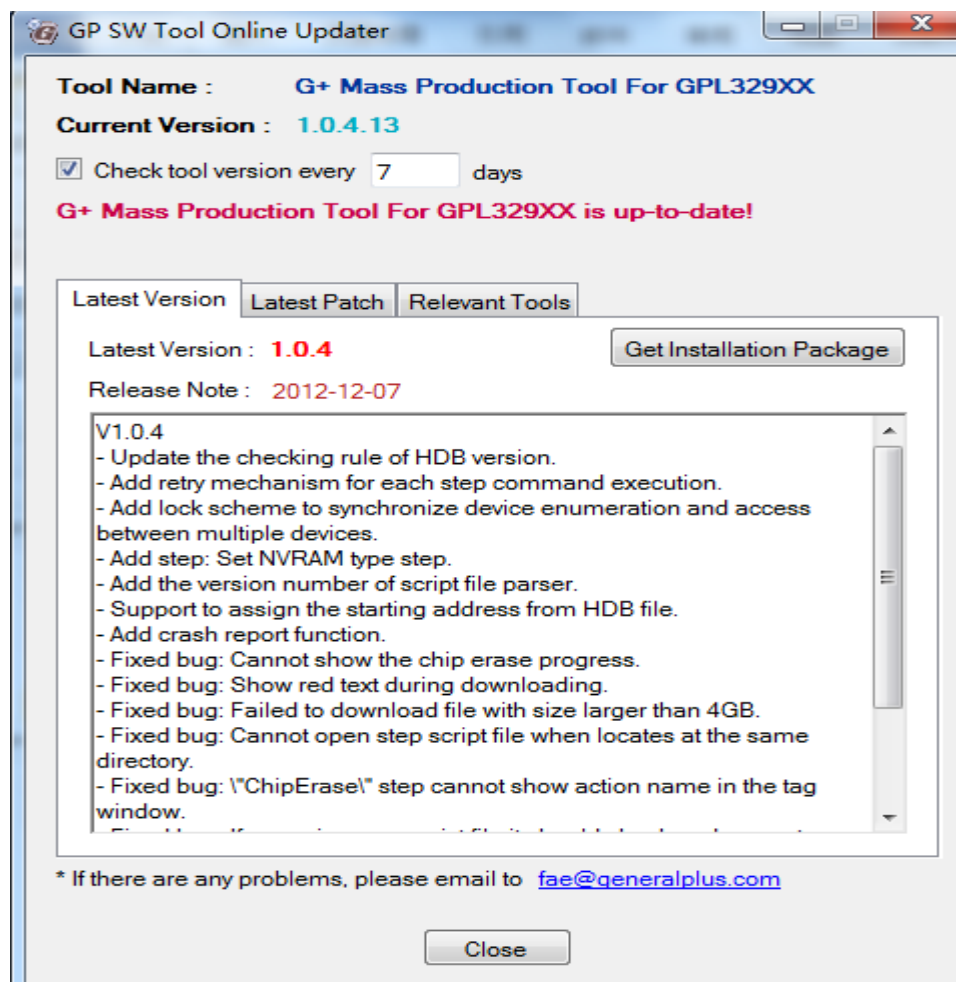
2.1 若尚未安装 GP SW Tool Online Updater 工具，会弹出以下讯息。



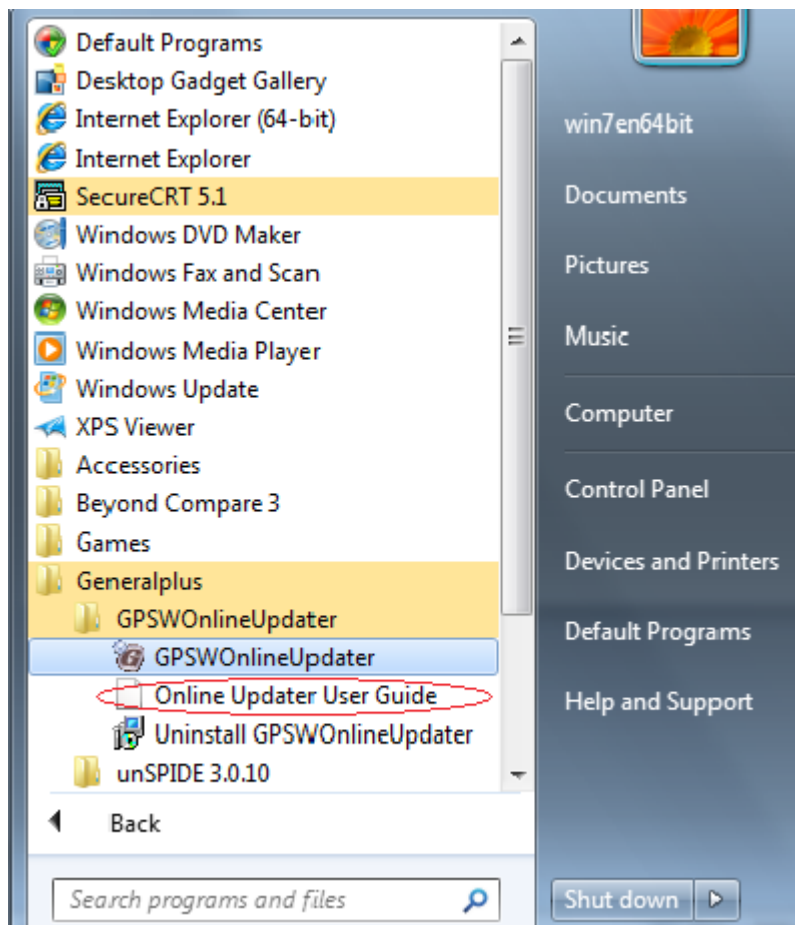
此时请到讯息中提到的目录下执行 G+ Online Update Installer.exe，并按照指示进行安装动作



2.2 若已安装 GP SW Tool Online Updater 工具，则会弹出此工具接口。可由工具接口上得知目前的工具版本以及最新版本等信息，也可以设置自动检查工具版本的周期。当 Tool 版本不是最新的时候，也允许拥有 A-Key 的用户透过 Get Installation Package 按钮，下载最新版本的工具来使用。



2.3 如果您想知道更多关于“G+ Online Updater”的使用，在您成功安装“GP SW Tool Online Updater”后，可以阅读手册“Online Updater User Guide.pdf”来了解使用方法。(路径: All Programs-> Generalplus -> GPSWOnlineUpdater -> Online Updater User Guide.pdf )



---

## 2 烧录档存放规则 Conf (ScriptFile) 和 bin / hdb

---

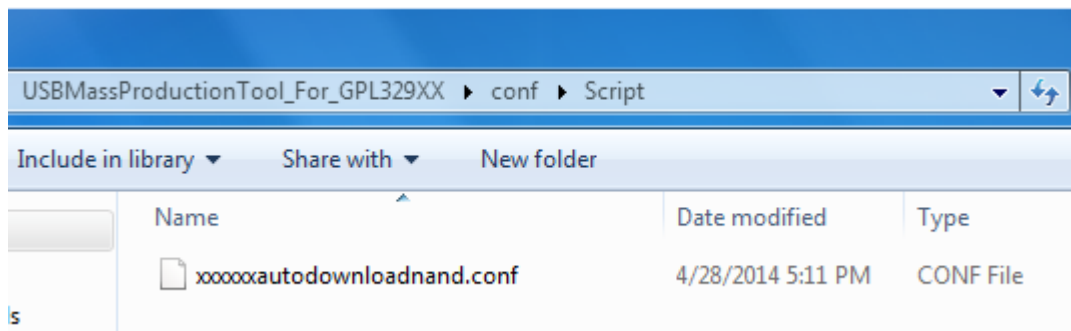
当用 G+Code packer 包出 binary 文件或 hdb 文件, 烧录用的 Conf (ScriptFile) 后, 因需要用户可能会对这些档案移动位置、或将 G+ Mass Production Tool For GPL329xx 目录移植再烧录, 这样操作很容易带来一个问题: 即移植后有些文件路径变更了导致烧录时无法找到。为了避免问题的发生, 建议用户对档案的存放使用如下的规则:

### 2.1 后缀名.conf 档存放规则

规则: 烧录用的 Conf (ScriptFile) 存放至

“USBMassProductionTool\_For\_GPL329XX\conf\Script” 目录下

举例: 如 “xxxxxautodownloadnand.conf” 是要烧录文件



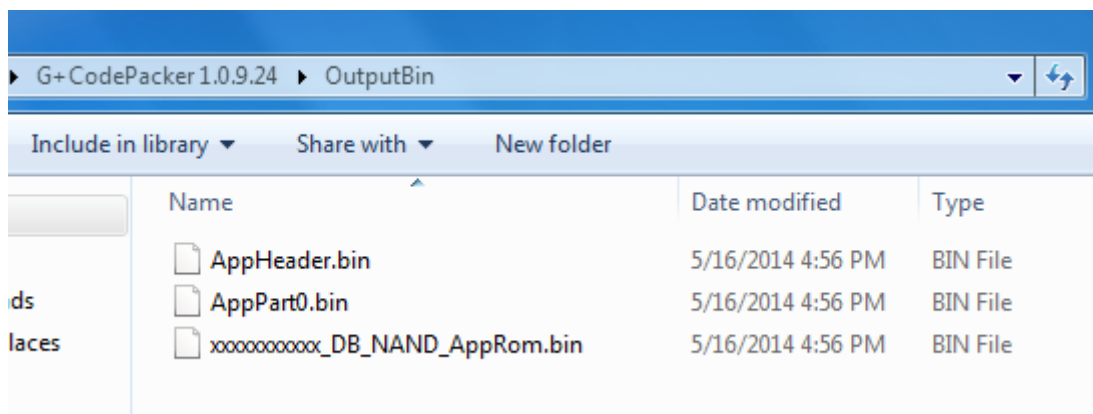
### 2.2 后缀名.bin/.hdb 档存放规则

规则: G+Code packer 包出相关的 binary 文件或 hdb 文件存放至

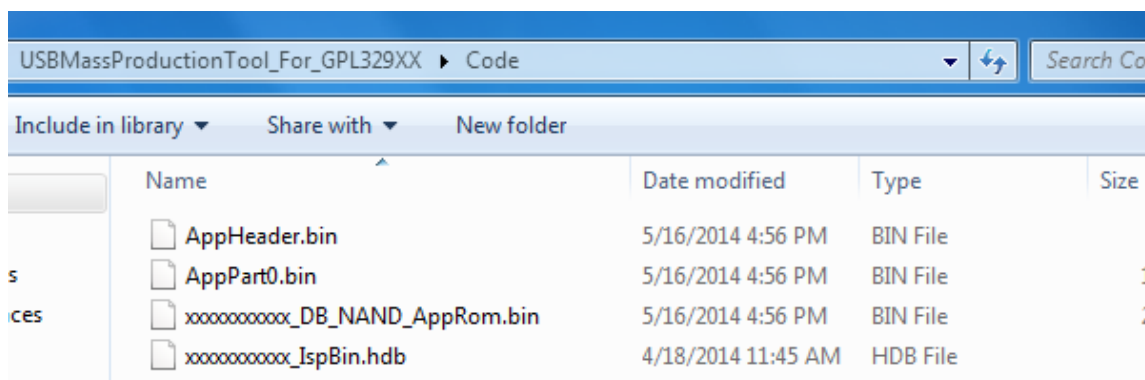
“USBMassProductionTool\_For\_GPL329XX\Code” 目录下

举例:

步骤 1: G+Code packer 包出 binary 文件 (.bin 档) 预设存放在 OutputBin 目录下 (如 zip 包解压后的目录 “G+CodePacker 1.0.9.24\OutputBin”)



步骤 2: 将步骤 1 中 OutputBin 目录下与烧录相关的 bin 档或者为用户已有的放置于其他位置的烧录 bin 档拷贝到 “USBMassProductionTool\_For\_GPL329XX\Code” 下，且将烧录用的 hdb 档也拷贝到此目录下，如图：



说明：hdb 档 codepacker 预设是存在于 “USBMassProductionTool\_For\_GPL329XX\Code”（如 zip 包解压后的目录 “G+CodePacker 1.0.9.24\USBMassProductionTool\_For\_GPL329XX\Code”）目录，若用户放置于其它地方也请移植至上图的目录中