

# P800Flash

## 量产型 Flash 编程器

UM01010101 V1.00 Date: 2018/03/14

产品用户手册

类别	内容
关键词	P800Flash、NandFlash、座烧、脱机、使用指南
摘 要	介绍 P800Flash 编程器烧录 NandFlash 使用方法

修订历史

版本	日期	原因
V1.00	2017/10/24	创建文档/lizengjia
V1.01	2017/10/31	编程器界面发生更改/lizengjia

## 目 录

1. P800Flash 编程器简介 .....	1
2. NandFlash 简单介绍 .....	2
2.1 NandFlash 的特殊性 .....	2
2.1.1 存在坏块 .....	2
2.1.2 易出现位翻转 .....	2
2.1.3 存在 Spare area (OOB 区) .....	2
2.1.4 多维的空间存储结构 .....	2
2.2 NandFlash 烧录的复杂性 .....	2
2.2.1 处理坏块 .....	2
2.2.2 计算 ECC .....	2
2.2.3 启动代码区的检查 .....	2
2.2.4 分区烧录(Partitioning) .....	2
2.2.5 非标准的坏块标志 .....	3
2.3 NandFlash 缩写名称介绍 .....	3
2.3.1 OOB .....	3
2.3.2 ECC .....	3
2.3.3 BBT .....	3
2.4 分区格式介绍 .....	3
2.4.1 .mbn 文件格式 .....	3
2.4.2 .def 文件格式 .....	3
2.5 NandFlash 编程器的烧录流程 .....	4
3. NandFlash 烧录方案信息表 .....	5
4. NandFlash 烧录功能介绍 .....	6
4.1 创建 .....	6
4.1.1 新建工程 .....	6
4.1.2 芯片选择 .....	7
4.1.3 烧录方式 .....	7
4.2 配置 .....	8
4.2.1 芯片选择 .....	8
4.2.2 通道配置 .....	9
4.2.3 组合配置 .....	9
4.2.4 烧写配置 .....	10
4.2.5 高级配置 .....	15
4.2.6 缓冲区 .....	19
4.2.7 校验和 .....	20
4.3 保存 .....	20
4.4 操作 .....	21
5. NandFlash 烧录基础操作流程 .....	23
5.1 方式一、调文件方案 .....	23
5.1.1 创建工程 .....	24
5.1.2 选择芯片 .....	24

5.1.3	烧录模式.....	25
5.1.4	调入文件.....	26
5.1.5	坏块管理方案选择.....	31
5.1.6	通道配置.....	32
5.1.7	保存工程.....	33
5.2	方式二、调入镜像数据文件+分区文件方案.....	34
5.2.1	创建工程.....	34
5.2.2	选择芯片.....	35
5.2.3	烧录模式.....	35
5.2.4	调分区文件.....	36
5.2.5	坏块管理方案选择.....	41
5.2.6	通道配置.....	42
5.2.7	保存工程.....	43
5.3	方式三、读母片方案.....	44
5.3.1	创建工程.....	44
5.3.2	选择芯片.....	45
5.3.3	烧录模式.....	46
5.3.4	母片分析.....	47
5.3.5	坏块管理方案选择.....	49
5.3.6	通道配置.....	50
5.3.7	保存工程.....	51
5.4	方式四、定制方案.....	52
5.4.1	用户定制流程.....	53
5.4.2	定制烧写方案要素.....	53
6	NandFlash 芯片烧录.....	54
6.1	量产烧录.....	54
6.2	芯片测试.....	54
6.3	日志文件.....	55
6.3.1	即时信息.....	55
6.3.2	Log 文件.....	55
7	免责声明.....	57

## 1. P800Flash 编程器简介

P800Flash 是广州致远电子有限公司历经十多年的编程技术积累，深耕打造的一款脱机、座烧、量产型的 Flash 编程器；它能支持各种封装的 EEPROM Flash、SPI NorFlash、Parallel NorFlash、NandFlash、SPI NandFlash、eMMC 等存储器。P800Flash 颠覆了传统烧录器的设计理念，积极创新，走上了属于自己的一条独特设计道路；它既能够实现对存储器芯片(包括了大容量的 eMMC 和 Nand Flash)快速、稳定批量烧录，又兼顾了各种便捷、智能和灵活的用户体验需求；其烧写效率高，性能优异，非常适合工厂批量生产、产品维修、研发实验室甚至自动化设备接入使用。



图 1.1 P800Flash 编程器

## 2. NandFlash 简单介绍

NandFlash 是目前市场上使用非常广泛的一种芯片，而由于 NandFlash 的本身工艺的特殊性，就已经决定了它与其他类型的 Flash 有着与众不同的区别。

### 2.1 NandFlash 的特殊性

#### 2.1.1 存在坏块

由于 NandFlash 生产工艺的原因，出厂芯片中会随机出现坏块。坏块在出厂时已经被初始化，并在特殊区域中标记为不可用，而在使用过程中如果出现坏块，也需要进行标记。

#### 2.1.2 易出现位翻转

NandFlash 由于本身硬件的内在特性，会导致（极其）偶尔的出现位反转的现象。如果位反转出现在关键文件上，会导致系统挂机；所以在使用 NandFlash 的同时，建议使用 ECC 算法确保可靠性。

#### 2.1.3 存在 Spare area (OOB 区)

正因为 NandFlash 有着坏块、易反转两项特殊地方，Spare area 就扮演作存放坏块标志、ECC 值以及芯片信息和文件信息的作用。

#### 2.1.4 多维的空间存储结构

NandFlash 一般由 Block、Page 结构组成，其中 Page 又由 Data area 和 Spare area(OOB 区)构成。所以在有的文件系统中就衍生出各种分区信息和扇区信息等。

### 2.2 NandFlash 烧录的复杂性

#### 2.2.1 处理坏块

由于 NandFlash 存在坏块，导致地址空间不是连续的。所以正确的处理坏块是保证 NandFlash 烧录后能够正常运行的关键。从大的方面来说，处理坏块常用的、最有效的主要为两种方法：跳过和替代。也有部分客户为了使用方便，并没有把坏块处理的概念引入，直接在坏块上写过，这种方法叫做硬拷贝，虽然简单，但容易产生很多不稳定的因素。我司编程器在出厂的时候为每一种 NandFlash 配置标准的坏块处理方式有跳过、保留替代和硬拷贝方式。

#### 2.2.2 计算 ECC

ECC 算法具有查错、纠错的功能，并且在 NandFlash 使用的大多数环境，需要带有 ECC 算法的；由于 ECC 算法比较多，每个算法个体又具有比较强的可变性(位组织，分段计算等)，而且在 Spare 区(OOB 区)存放的位置也不能统一，所以无法做成统一的算法。我司编程器在出厂的时候默认使用有几种的 ECC 算法，可供使用参考。

#### 2.2.3 启动代码区的检查

在启动区的代码要求存放在 NandFlash 对应块中是没有坏块的，一旦坏块出现，程序在 NandFlash 中启动就会失败。用户可以定制启动代码检查并报错功能。

#### 2.2.4 分区烧录(Partitioning)

NandFlash 在配合 CPU 一起运行的过程中，一些代码或者文件是需要从一个确定的地址读入运行的。所以在烧录过程中，烧录数据文件为多个，或者由多个数据文件合成的烧录档

案，需要进行分区并且定位烧录，这种方式我司称之为分区烧录模式，用户可以结合其 ECC 等功能，向我司定制。

### 2.2.5 非标准的坏块标志

一些用户为其系统的保密性，公司内部技术的兼容性，将坏块标志放在非标准的位置，或者使用非 0xFF 作为好块标志并带有其他意义(如只读)。如有这样的情况，请用户给予说明并定制算法。

## 2.3 NandFlash 缩写名称介绍

### 2.3.1 OOB

Out Of Band，即备用区（Spare Area），用于存放坏块标记、ECC 等值。

### 2.3.2 ECC

ECC 的全称是 Error Checking and Correction，是一种用于 NandFlash 的差错检测和修正算法。

### 2.3.3 BBT

Bad Block Table，即坏块表，记录坏块位置的表，方便系统快速找到坏块位置。

## 2.4 分区格式介绍

### 2.4.1 .mbn 文件格式

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
00000000h:	00	00	00	00	02	00	00	00	02	00	00	00	FF	FF	FF	FF	文件1.....
00000010h:	03	00	00	00	09	00	00	00	03	00	00	00	FF	FF	FF	FF	文件2.....
00000020h:	0A	00	00	00	D1	00	00	00	BE	00	00	00	FF	FF	FF	FF	文件3.....
00000030h:	D2	00	00	00	08	01	00	00	01	00	00	00	FF	FF	FF	FF	文件4.....
00000040h:	09	01	00	00	0E	01	00	00	05	00	00	00	FF	FF	FF	FF	文件5.....
00000050h:	0F	01	00	00	18	02	00	00	11	00	00	00	FF	FF	FF	FF	文件6.....
00000060h:	19	02	00	00	FF	03	00	00	02	00	00	00	FF	FF	FF	FF	文件7.....
00000070h:																	

起始块地址      结束块地址      烧写块数

图 2.4.1 .mbn 分区格式

### 2.4.2 .def 文件格式

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
00000000h:	47	52	4F	55	50	20	44	45	46	49	4E	45	32	00	00	00	GROUP_DEFINE2 标识符
00000010h:	01	00	00	00	00	00	00	00	09	00	00	00	02	00	00	00	.....
00000020h:	01	00	00	00	0A	00	00	00	1E	00	00	00	07	00	00	00	.....
00000030h:	01	00	00	00	1F	00	00	00	31	00	00	00	05	00	00	00	.....1.....
00000040h:	01	00	00	00	32	00	00	00	41	00	00	00	04	00	00	00	.....2...A.....
00000050h:	01	00	00	00	42	00	00	00	45	00	00	00	01	00	00	00	.....B...E.....
00000060h:	01	00	00	00	46	00	00	00	49	00	00	00	01	00	00	00	.....F...I.....
00000070h:	01	00	00	00	4A	00	00	00	4D	00	00	00	01	00	00	00	一行表示一个文件
00000080h:	01	00	00	00	4E	00	00	00	AD	00	00	00	50	00	00	00	.....N...-...P...
00000090h:	01	00	00	00	AE	00	00	00	0D	01	00	00	50	00	00	00	.....P...
000000a0h:	01	00	00	00	0E	01	00	00	CD	03	00	00	34	01	00	00	.....? ...4...
000000b0h:	01	00	00	00	CE	03	00	00	FF	03	00	00	01	00	00	00	.....? ..
000000c0h:	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
000000d0h:																	

起始块      结束块      烧写块数

图 2.4.2 .def 分区格式

## 2.5 NandFlash 编程器的烧录流程

编程器软件执行烧写 NandFlash 流程：

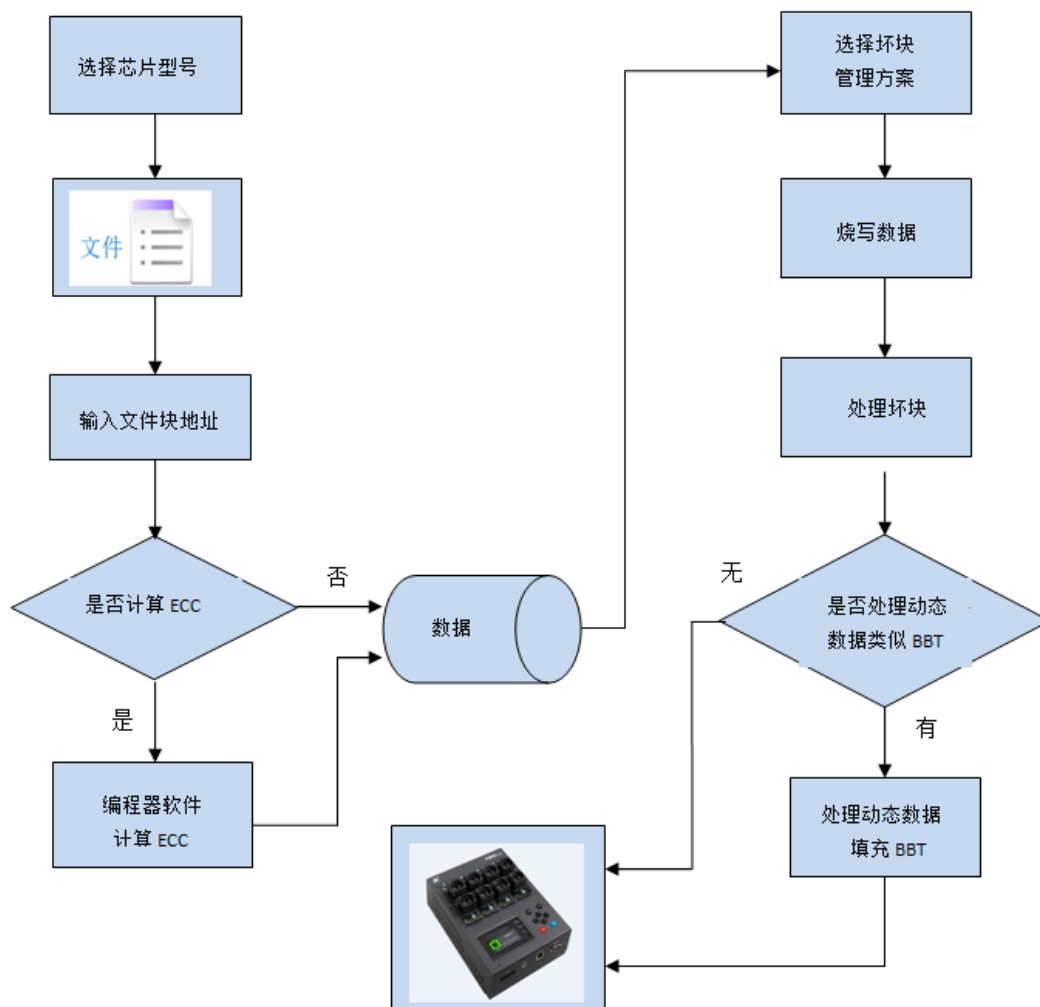


图 2.4.3 定制烧写流程



## 3. NandFlash 烧录方案信息表

申请人基本信息			
申请单位		联系人	
联系方式		邮箱	
器件厂家		器件型号	
Nand Flash 算法信息需求 (软件技术人员填写)			
一、您使用的是哪一种处理坏块方式?			
答: A.跳过坏块 B.替换表 C.硬拷贝方式 D.分区烧录模式 E.其他 ..... 若是使用了其他的坏块处理方案, 请问是否能提供详细的坏块管理文档说明? A.可以(提供文档说明) B.不可以			
二、您的数据文件的格式是什么? 有多少个烧写的数据文件? 请详细描述待烧录的数据文件在芯片存储空间的固定烧录地址。			
答: 1)、数据文件格式: 2)、数据文件个数: 3)、各个数据文件对应 IC 的地址:			
起始块	Flash 物理地址	烧写的块数量	文件名字
三、数据文件是否包含有 spare area(OOB)?			
答: A.包含 B.没有包含 ..... 若数据文件没有包含 spare area(OOB), 请问 spare area(OOB)是填充 0xff 还是填充 ECC? A.填充 0xff B.填充 ECC ..... 若数据文件没有包含 spare area(OOB), 并需要填充 ECC, 且要求由我们来计算的, 是否能提供 ECC 算法以及 C 源码。并且提供几个完整的页数据镜像文件 (含 spare area(OOB 区))。 A.能提供(ECC 算法以及 C 源码) B.不能提供 ..... 若数据文件没有包含 spare area(OOB), 需要填充 ECC 的, 请问每个文件是否使用相同的 ECC? A. 相同 B.不相同			
四、烧录的数据文件是否改变了坏块标志位, 如果改变了, 重新存放在什么地方?			
答: 没有改变 B.改变了, 存放地址为:			
五、烧录的数据文件是否有启动代码(boot)? 若有, 启动代码占用多少个 block? 启动代码允许有坏块吗? 如果有坏块, 是否不可以继续编程?			
答:;			
六、是否使用了文件系统? 例如 UBI 文件系统			
答: A.使用 B.不使用 ..... 如果使用, 您是否可以提供文件系统的描述 (代码, 文档)。请简要描述一下您要求的文件系统。 A.可以提供; B.不可以			
七、如果您有其他的要求我们没有涉及到的, 请您描述, 我们会努力满足您的需求。			

鉴于 NandFlash 烧录的特殊性, 我司定制 NandFlash 烧录方案的, 起价收费标准: ¥\_\_\_\_\_; 如需了解更多的信息, 请拨打 020-22644360 电话沟通或者发邮件 tools@zlg.cn 咨询。

## 4. NandFlash 烧录功能介绍

### 4.1 创建

点击【创建】按钮，可以创建工程，如图 4.1 所示：

P800Flash 编程器在烧录之前，要求用户必须先创建工程；创建工程，可以实时记录并保存当前用户的操作设置，待用户验证完成之后，批量生产时，打工工程，即可实现一键烧录，免去再次的繁琐的重复操作设置和重复的验证工作。



图 4.1 主界面

#### 4.1.1 新建工程

点击【创建】按钮后，进入新建工程，如图 4.1.1 所示的页面：

- ◆ 工程名称：点击输入框，即可输入新建的工程名称（注：使用字母或数字）。
- ◆ 工程类型：存放工程的路径，可以是本机、SD 卡、U 盘。

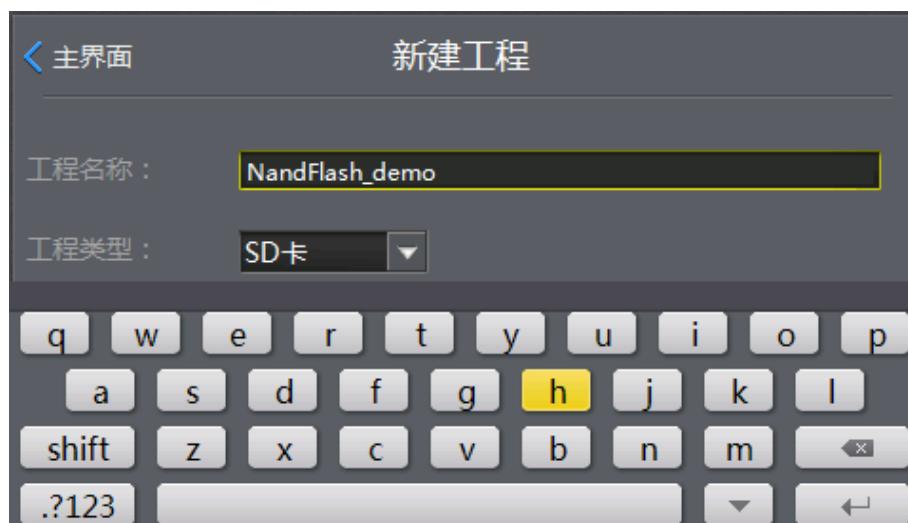


图 4.1.1 新建工程

### 4.1.2 芯片选择

点击【新建】，进入如图 4.1.2 所示芯片选择页面：

用户在【检索】编辑框中，输入需要烧录的芯片名称，点击对应的芯片之后，在【选中芯片】一栏中即可显示对应的芯片名称。

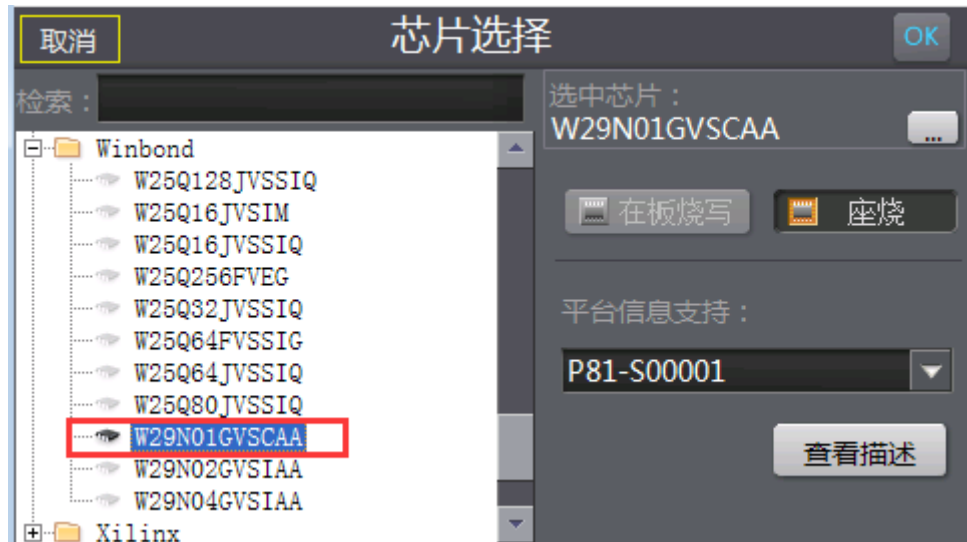


图 4.1.2 芯片选择

### 4.1.3 烧录方式

进入【芯片选择】页面，用户可以选择烧录方式，如图 4.1.3 所示的页面：

编程器提供有两种烧录方式：【在板烧录】和【座烧】；P800Flash 只支持座烧方式，即裸片烧录（芯片先烧录，后贴板）；并且通过【平台信息支持】，可以查询到当前选中芯片的所对应的适配器型号。

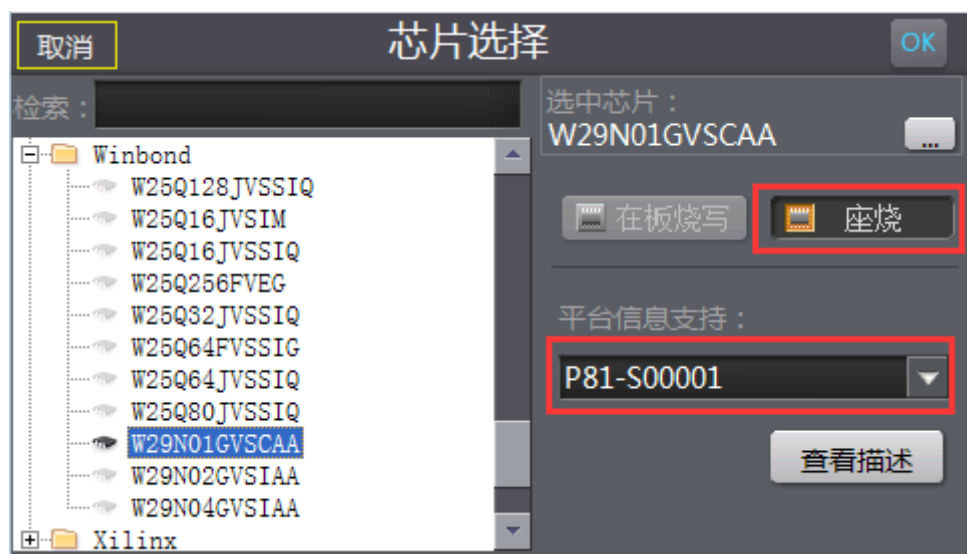


图 4.1.3 芯片选择

## 4.2 配置

点击【配置】，进入如图 4.1.4 所示配置芯片信息页面：

进入配置界面：页面包括【芯片选择】、【通道配置】、【组合配置】、【烧写配置】、【高级配置】、【校验和】、【系统设置】：



图 4.1.4 主界面



图 4.1.5 配置界面

### 4.2.1 芯片选择

点击【芯片选择】，进入如图 4.2.1 所示的页面：

芯片选择，可以选择芯片型号、烧录方式，并可以查询芯片对应的适配器等相关信息。在新建工程中，已选择了芯片型号、烧录方式和适配座型号，在该【芯片选择】配置无需再重复操作。

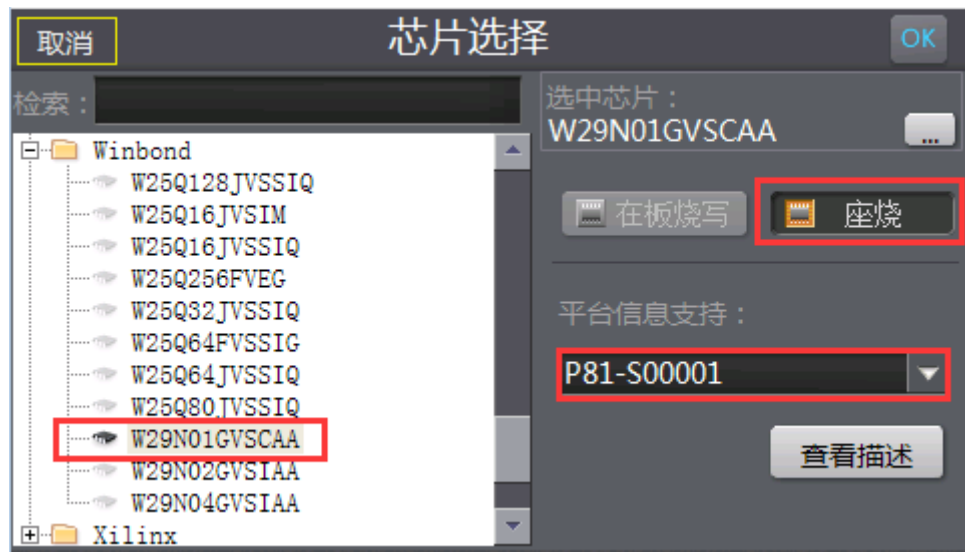


图 4.2.1 通道配置

#### 4.2.2 通道配置

点击【通道配置】，如图 4.2.2 所示的页面：

通道配置，用来使能对应芯片放置并烧录的通道，编程器只默认使能了第一通道。



图 4.2.2 通道配置

#### 4.2.3 组合配置

点击【组合配置】，进入如图 4.2.3 所示的页面：

组合配置是批处理动作，目的是简化在进行工厂批量生产环节的人工动作。客户可编辑所执行的操作步骤，达到一键实现的目的。（如上图在执行组合或者量产的时候按顺序对芯片执行：擦除、编程和校验操作）。

**重要说明：**NandFlash 由于其工艺的特殊性，烧录的方式与其他芯片有着很大的差异性，所以在“组合”或者“量产”烧录模式下，请按照默认配置即可！

编程器在“组合”或者“量产”下，烧录 NandFlash 正常流程如下：【扫描并记录整片芯片坏块】---->【执行整片擦除】---->【烧写一块】----->【校验一块】---->【烧写下一块】----->【校验下一块】.....。



图 4.2.3 组合操作

#### 4.2.4 烧写配置

点击【烧写配置】，进入如图 4.2.4 所示的页面：



图 4.2.4 操作选项

##### 1. 烧录模式：

烧录模式有两种：普通模式和高速模式：

- ◆ 【普通模式】：对存储 SD 卡没有要求，但烧写速度较慢；
- ◆ 【高速模式】：必须使用原厂提供的高速 SD 卡，烧写速度快；

##### 2. 文件调入：

文件调入方式有两种：调入多个数据文件和调入镜像数据文件+分区文件：

◆ 【调入多个数据文件】：根据用户分区地址，依次调入各个数据文件；

在【烧写配置】界面，点击【调入文件】，打开烧写的数据文件后，进入如图 4.2.5 所示的页面：



图 4.2.5 文件配置

➤ 起始地址和块地址：

用来设置烧写的数据文件在芯片起始固定地址；起始地址：用 16 进制显示，显示没有包含 OOB 区域的物理地址；起始块：十进制表示，表示芯片块地址；两个只需填写一个即可，两者会自动换算的；

➤ 文件自带 OOB 区域：

调入的数据文件，已经包含了 OOB 区域，则默认该设置；

➤ ECC 计算方案：

调入的数据文件并没有包含 OOB 区域的，那么就需要在 OOB 区域填充 0xFF 或者 ECC 计算数据，否则调入的数据文件在编程器软件的缓冲区发生错乱。

➤ 特殊处理：

在 ECC 计算方案的前提下，需要再继续定制方案；

➤ 自定义数据：

用户需求在 OOB 区域设置其他数据的，可以设置固定设置；自定义数据只能在 OOB 区域设置；

➤ 自定义页规格：

用来计算 ECC 数据使用的大页方式或者小页方式。

◆ 【调入镜像数据文件+分区文件】：用户已有分区文件（.mbn 或.def 格式）和镜像数据文件，可同时加载镜像数据文件和分区文件到编程器软件缓冲区内；

在【烧写配置】界面，点击【调入文件】，打开烧写的分区文件后，进入如图 4.2.6



所示的页面：

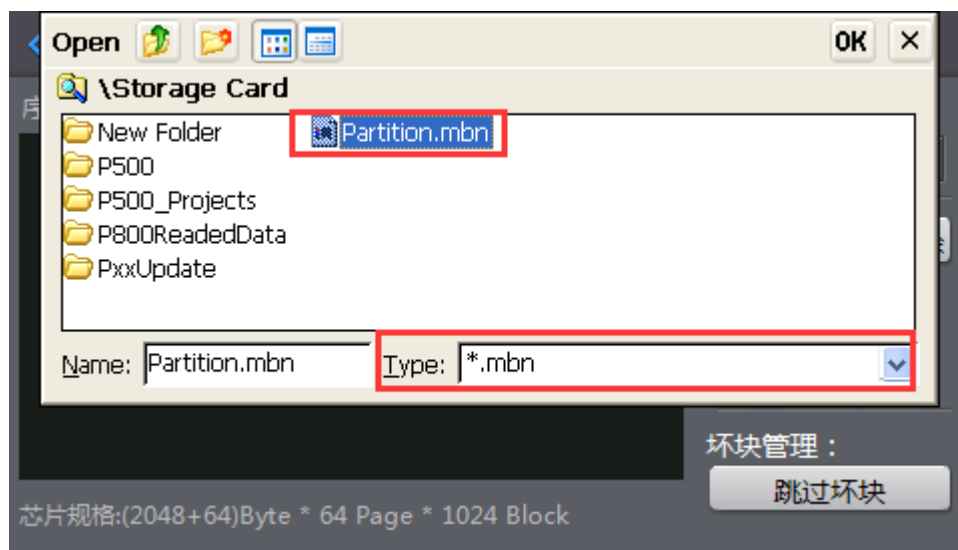


图 4.2.6 调入分区



图 4.2.7 调入烧写数据文件

### 3. 删除：

可以执行删除分区列表的文件；若是某些分区列表的文件不需要烧写的，选中该分区，执行【删除】，即可去掉该分区。

### 4. 导出分区：

导出分区列表的分区文件，编程器软件支持保存的文件格式为.mbn 和.def；

### 5. 坏块管理：

坏块管理方案种类众多，大部分很多都是用户定制的，点击编程器软件的【坏块管理】，进入如图 4.2.8 所示的页面：





图 4.2.8 坏块方案配置

- ◆ **【跳过坏块】**: 烧录时遇到坏块，直接跳到下一块烧录；
- ◆ **【通用 BBT】**: 属于定制类坏块管理方案；
- ◆ **【硬拷贝】**: 无论芯片是否有坏块，都会对应作一一拷贝，出错也不提示；
- ◆ **【通用替换表】**: 根据用户方案规则，预留芯片某些区域，用来替换烧写时遇到的坏块；

#### 6. 母片分析:

在【烧写配置】界面，点击【母片分析】，进入如图 4.2.9 所示的页面：

母片分析是一种读母片数据的烧录方案，用户在不清楚烧录方案的情况下，可通过使用母片分析方式作烧写测试：



图 4.2.9 母片分析

- ◆ **【文件】**: 分析的对象为芯片的镜像数据文件(即整个芯片容量的数据文件)；
- ◆ **【母片】**: 分析的对象为可以正常运行的芯片作为母片；注：分析时母片需要放置在编程器的第一个烧录座上；

- ◆ **【预设分区】**: 通过预设分区, 可以更加准确分析芯片的数据布局;
  - ✓ 序号: 显示预设分区的文件的序号, 从“0”开始;
  - ✓ 起始块: 每个分区文件烧写的起始块地址, 十进制显示;
  - ✓ 添加: 可以通过“添加”按钮, 预设每个分区文件的起始块;
  - ✓ 删除: 可以删除已经预设好的分区列表文件;
  - ✓ 导入: 用户已有分区文件, 可以直接导入, 无需依次“添加”; 编程器软件支持的分区文件格式有两种: .mbn 和 .def;
- ◆ **【Data】**: 勾选“Data”, 分析区域仅为 Data 区域; 同时勾选 Data、Spare, 那么分区的区域为所有区域数据;
- ◆ **【Spare】**: 勾选“Spare”, 分析区域仅为 Spare 区域同时勾选 Data、Spare, 那么分区的区域为所有区域数据;
- ◆ **【自动纠正错误位】**: 由于芯片极其容易出现位翻转的现象, 在读取母片数据时, 可以通过勾选该选项, 认为在安全范围之内, 出现个别位翻转, 属于正常现象, 软件自动修正为 0xFF;
- ◆ **【分析方式】**: 分析数据时, 用来判别是否把坏块也一起作分析处理;
  - ✓ 默认方式: 分析母片时, 遇到坏块, 直接跳过读取下一块;
  - ✓ 忽略坏块: 分析母片时, 遇到坏块, 把坏块数据一起读取出来;
  - ✓ 特征码: 定制的方案;
- ◆ **【开始分析】**: 点击“开始分析”, 即可开始分析母片数据;

## 7. 分区列表;

分区列表用来确定每个数据文件在芯片的起始块地址和烧写的块数; 量产烧写时, 编程器将会严格执行分区列表的设置进行烧写!:

- ◆ **【序号】**: 从“0”开始, 排序显示数据名字;
- ◆ **【起始块】**: 每个数据文件的烧写起始块地址, 十进制显示;
- ◆ **【块数】**: 每个数据文件要烧写的块数, 十进制显示;
- ◆ **【块限制】**: 可以控制每个数据文件的在该分区内允许出现的坏块数量, 十进制显示; “0”表示不允许, 其他数值表示允许出现的坏块数量;



图 4.2.10 分区列表

4.2.5 高级配置

点击【高级配置】，进入如图 4.2.11 所示的页面：

【高级配置】有以下各个子功能：【检测坏块】、【坏块识别】、【校验配置】、【擦除配置】、【读取配置】和【常用配置】



图 4.2.11 高级配置

1. 检测坏块：

可全面扫描芯片的坏块，并通过保存日记，记录当前测试的状况。



图 4.2.11 坏块检测

## 2. 坏块识别:

坏块识别的方式有多种，编程器默认判断的方式为识别芯片的坏块标志位是否为 0xFF;

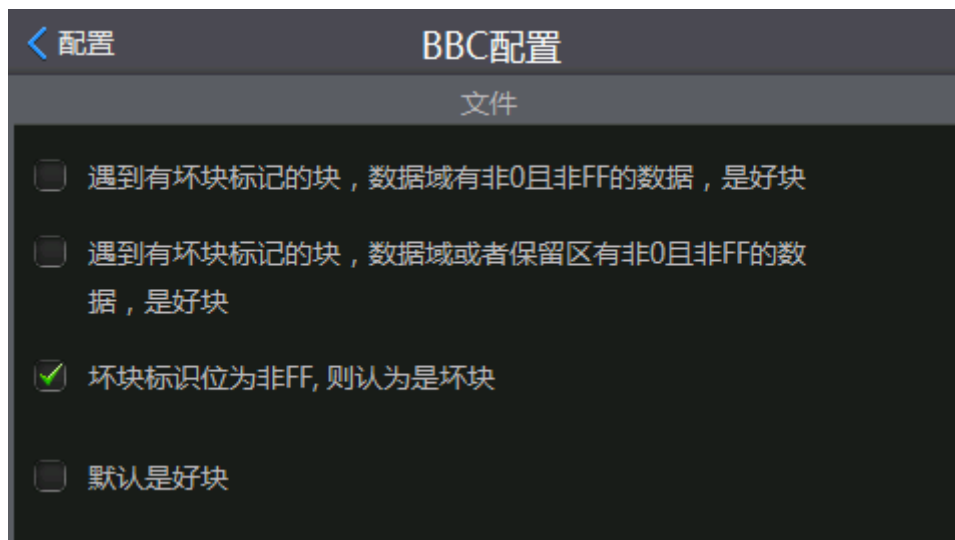


图 4.2.12 坏块识别

## 3. 校验配置:



图 4.2.13 校验配置

- ◆ **【校验允许容错位】**：由于 NandFlash 的生产工艺特殊性，芯片是极其容易出现位翻转的现象，通过设置“校验允许容错位”，即可在“组合”或者“量产”下通过数据校验；编程器软件有默认的设置数值，用户也可以根据自己的 ECC 算法的纠错能力手动配置。
- ◆ **【组合/量产的校验失败后】**：这个选项由用户烧录方案决定；编程器软件默认为校验失败后，直接报错并停止烧录。

#### 4. 擦除配置：

用来设置对芯片的擦除操作。



图 4.2.14 擦除配置

- ◆ **【全片擦除】**：用来整片擦除芯片；擦除前，会先扫描整个芯片的坏块，坏块不做擦除处理；
- ◆ **【分区列表擦除】**：按照分区列表设置的地址范围擦除，擦除前，会先扫描整

个芯片的坏块，坏块不做擦除处理；

- ◆ **【强制擦除】**：勾选“强制擦除”和全片擦除，会对整个芯片进行擦除，无需扫面坏块
- ◆ **【擦除失败不报错，只是提示】**：在执行擦除动作时，某些块可能出现无法擦除，那么可以直接跳到下一块继续擦除，直到芯片执行完擦除操作；

#### 5. 读取配置：



图 4.2.14 读取配置

- ◆ **【全片读取】**：读取的地址范围为整个芯片的容量，包括坏块等；
- ◆ **【分区列表读取】**：读取的地址范围为分区列表设置的地址范围，包括坏块等；

#### 6. 常用配置：



图 4.2.15 常用配置

- ◆ **【自动管脚检测】**：在量产模式下，通过管脚检测来判断取、放芯片；
- ◆ **【检测期间 ID】**：读取芯片的 ID 号，可以用来判断芯片的型号；
- ◆ **【量产时不烧空片】**：该功能只在“普通模式”下使用，勾选之后，若是缓冲

区的任意一个块有空页(0xFF)，则跳过该页，不作烧录处理；

- ◆ 【全片至少可用好块】：一个芯片出现的坏块的数量是有限的，可以通过设置该数值，来筛选一下不良的芯片；

#### 4.2.6 缓冲区

可以显示当前“烧写”或者“读取”的数据，通过输入某一块的某一页的，即显示当前一个页的数据；



图 4.2.9 数据显示

**重要说明：**编程器软件支持双缓冲区功能，把烧写的数据和读取芯片的数据独立分开，既可避免用户因误点击【读取】的违规操作，导致批量生产事故，又可将烧写的数据和读取的数据作对比验证，安全而便捷。

【保存文件】功能，用户既可以保存“烧写”缓冲区的数据，也可以保存“读取”缓冲区数据；且用户可以任意设置保存的文件长度以及保存的路径！

注意：用户想要烧录读取芯片的数据，就必须先保存读取到缓冲区的数据，然后再前往在【烧写配置】界面上，执行调入文件操作！



图 4.2.10 保存文件

#### 4.2.7 校验和

计算数据校验和，数据校验和的值在工程文件下显示。



图 4.2.16 常用配置

- ◆ **【SUM】**: 计算数据的累加和；
- ◆ **【CRC32】**: 通过 CRC32 算法，计算数据校验和；
- ◆ **【MD5】**: 通过 MD5 算法，计算数据校验和；

#### 4.3 保存

点击【保存】，用来保存工程，如图 4.3 所示的页面：

通过【创建】工程，设置【配置】完成之后，点击【保存】，即可保存用户当前所有的设置；用户批量生产时，直接打开已创建的工程，便可实现一键烧录。

**重要说明：**保存工程文件时强烈建议“计算数据校验和”；数据校验和的值，会在工程



文件下面显示，可方便工厂核对校准！



图 4.3 主界面

#### 4.4 操作

点击【操作】，用来对芯片执行烧录，如图 4.4 所示的页面：

进入【操作】界面：页面包括【擦除】、【查空】、【烧写】、【校验】、【组合】、【量产】，如图 2.3 所示：



图 4.4 主界面



图 4.5 操作界面

- ◆ **【读取】**：对芯片的数据进行读取操作。
- ◆ **【擦除】**：根据配置选项中配置的是全片擦除还是分区列表擦除在这里点击擦除的时候会对使能通道的芯片进行擦除芯片内容。
- ◆ **【查空】**：检测当前的芯片是否全部是芯片的默认数据。
- ◆ **【烧写】**：根据配置选项中设置的烧录区域中的内容烧录到芯片对应的位置。
- ◆ **【校验】**：把缓冲区的数据，与芯片的数据进行比对。
- ◆ **【组合】**：根据组合配置中所设置的步骤按顺序执行上述的操作。
- ◆ **【量产】**：多个组合的执行过程。当完成一次组合之后检测芯片是否重新放置之后再次执行组合操作。

## 5. NandFlash 烧录基础操作流程

### 5.1 方式一、调文件方案

该调文件方案，可支持调入一个数据文件或者多个数据文件；若用户是有多个数据文件一起调入缓冲区的，请务必提供各个数据文件在芯片 Flash 空间的固定烧录地址，即分区文件(Partitioning)。

注：烧录之前，需要务必先把烧录文件保存在 SD 卡上！

**案例操作演示：**本调入方案的操作流程指南，仅供参考：

本方案提供的 NandFlash 烧录要素：

- ◆ 烧写的芯片型号 W29N01GVSI AA，芯片规格是：(2048+64)\*64\*1024；
- ◆ 提供有 7 个自带有 OOB 区域(spare area)的烧写数据文件；
- ◆ 提供有分区文件，7 个文件都有固定的起始块地址；
- ◆ 采用的坏块管理策略：跳过坏块'

7 个自带有 OOB 区域的烧录数据文件：

File_1.bin	2015/8/13 15:27	BIN 文件	264 KB
File_2.bin	2015/8/13 15:28	BIN 文件	396 KB
File_3.bin	2015/8/13 15:28	BIN 文件	25,080 KB
File_4.bin	2015/8/13 15:28	BIN 文件	132 KB
File_5.bin	2015/8/13 15:29	BIN 文件	660 KB
File_6.bin	2015/8/13 15:29	BIN 文件	2,244 KB
File_7.bin	2015/8/13 15:29	BIN 文件	264 KB

图 5.1 数据文件

数据文件分区，十进制显示：

序号	文件名	起始块	结束块	块数
0	File_1.bin	0	2	2
1	File_2.bin	3	9	3
2	File_3.bin	10	209	190
3	File_4.bin	210	264	1
4	File_5.bin	265	270	5
5	File_6.bin	271	536	17
6	File_7.bin	537	1023	2

图 5.2 分区文件

### 5.1.1 创建工程

点击【创建】，即可创建工程，如图 5.1.1 所示：



图 5.1.1 主界面

点击【创建】，进入【新建工程】页面，新建工程名称和工程类型，如图 5.1.2 所示：

【工程名称】：工程名称由字母和数字组成；

【工程类型】：存放的工程可以在本机、SD 卡或者 U 盘上；

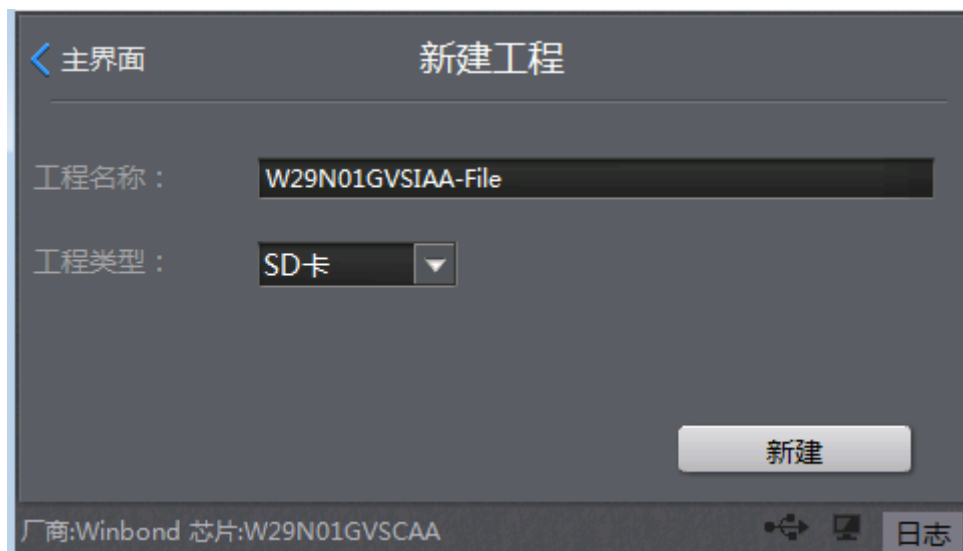


图 5.1.2 新建工程

### 5.1.2 选择芯片

完成输入【工程名称】之后，点击【新建】，进入【芯片选择】页面，如图 5.1.3 所示：

选择对应芯片型号、烧录方式和适配座型号。

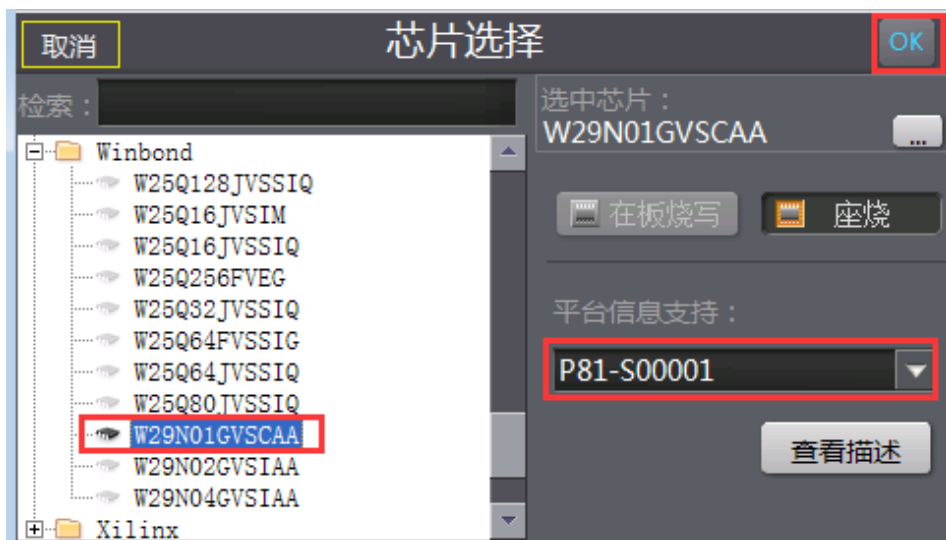


图 5.1.3 选择芯片

### 5.1.3 烧录模式

完成【芯片选择】操作之后，点击右上角的【OK】，进入【配置】页面，如图 5.1.4 所示：

点击【烧写配置】，编程器提供有两种可选择的烧录模式：【高速模式】和【普通模式】：

- ◆ 【高速模式】：必须使用原厂提供的高速 SD 卡，特点烧录速度快；
- ◆ 【普通模式】：对 SD 卡要求不高，缺点是烧录速度慢；

**使用注意：**

- 【高速模式】必须使用原厂提供的 SD 卡，可实现快速烧录，提高生产的效率。
- 由于两种烧录模式的 SD 卡存储数据方式不同，切换【烧录模式】，编程器软件会清空缓冲区数据和分区列表，用户必须要重新调入烧录的数据文件。



图 5.1.4 烧写配置



图 5.1.5 烧录模式

#### 5.1.4 调入文件

选择好【烧录模式】之后，在【烧写配置】页面，点击【调入文件】，开始调入用户的数据文件到编程器软件的缓冲区上。



图 5.1.6 调入文件

进入选择文件的界面，在文件类型(Type)处，默认选择格式：All file(\*.\*), 可显示所有的文件。

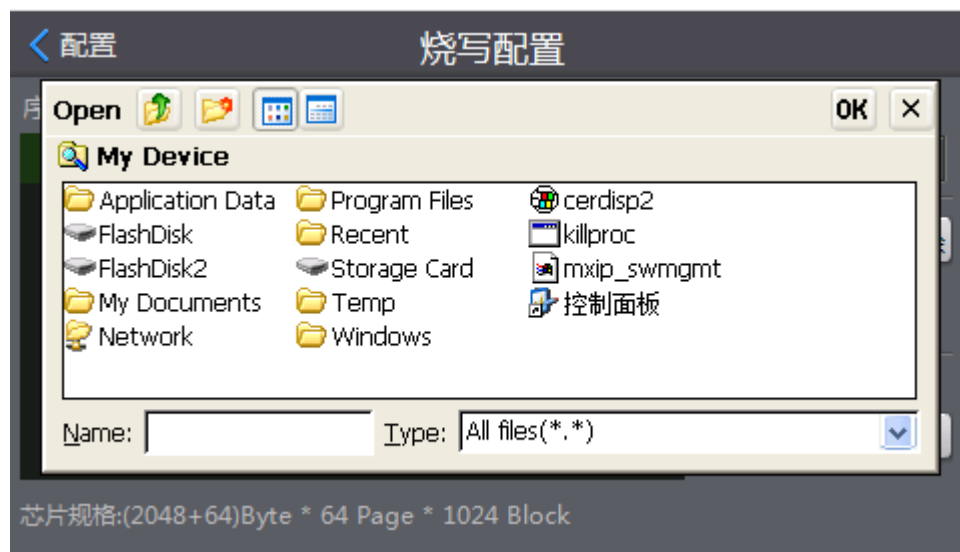


图 5.1.7 选择文件的格式类型

打开保存在 SD 路径下的数据文件，并依次调入待烧录的数据文件。

1. 调入第一个数据文件：

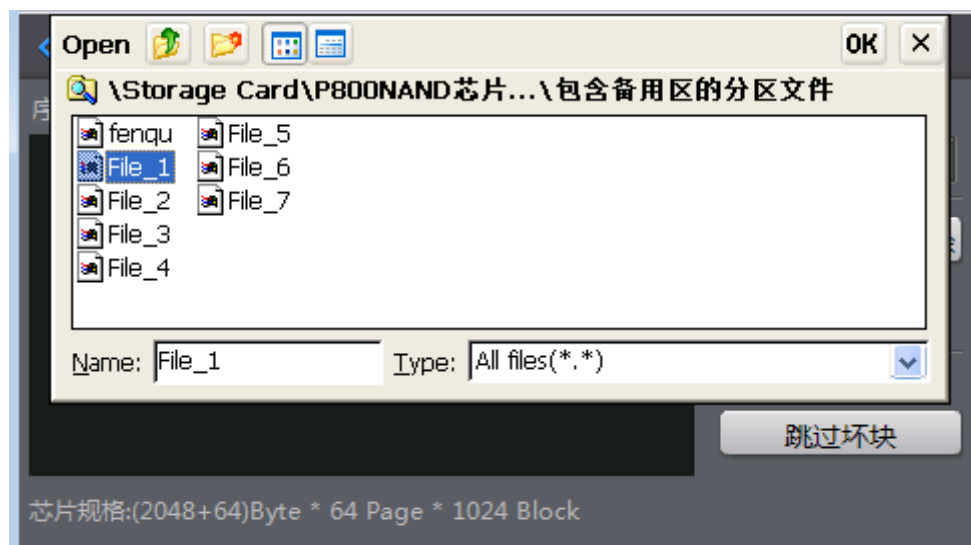


图 5.1.8 选择【File\_1】数据文件

**重要说明：**数据文件调入之后，会进入【文件配置】界面，用户可根据自己的烧录方案选择以下三种方式：

- ◆ 若数据文件已经包含了 OOB 区域（Spare area），则默认【文件自带 OOB 区域】；
- ◆ 若数据文件并没有包含 OOB 区域，则需要用户填充 0xFF 数据或者选择对应的【ECC 计算方案】；
- ◆ 若是烧录文件并没有包含 OOB 区域，且用户在编程器软件上也没有找到相对应的【ECC 计算方案】，则需要做软件定制。

\*该芯片的页规格是：2048+64，其中 2048 为芯片的数据区域，64 为芯片的 OOB 区域（Spare area）。

该操作演示的数据文件已包含了 OOB 数据，编程器软件按照默认选择【文件自带 OOB

区域】即可。

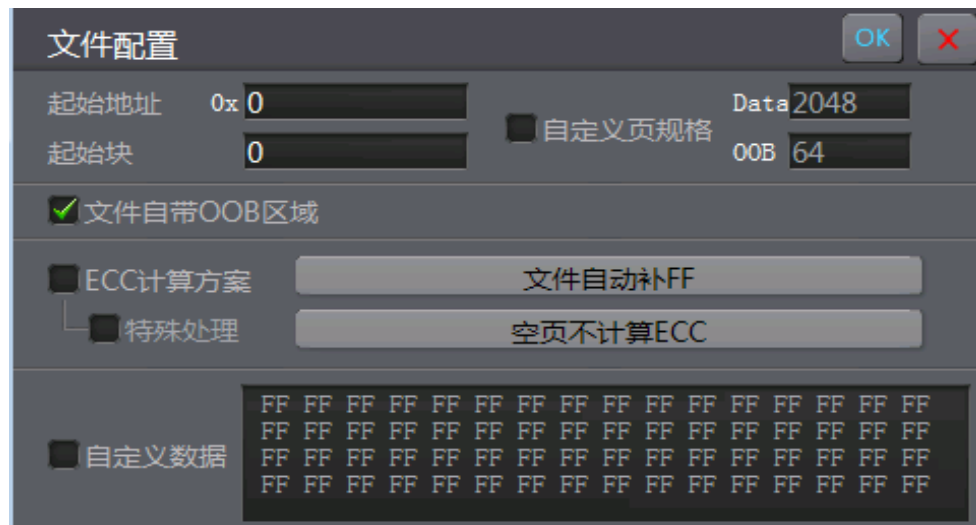


图 5.1.9 文件包含了 OOB 区域

设置数据文件的起始块地址：

- ◆ 起始地址：该地址是没有包含 OOB 区域数据的物理地址；
- ◆ 起始块：该地址是芯片的实际的块地址，十进制显示模式。

这两个起始地址编程器软件会自动换算的，用户只需填写其中一个即可。

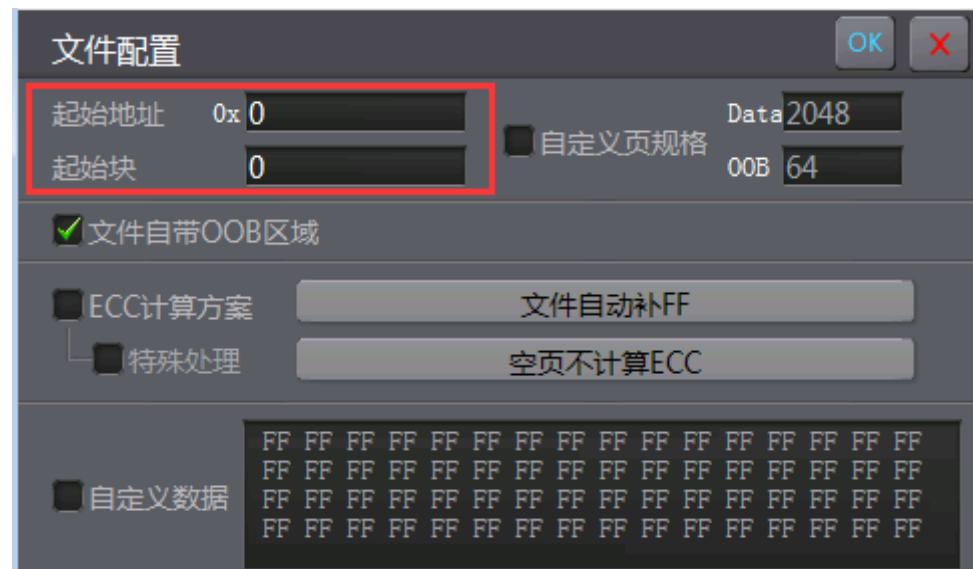


图 5.1.10 设置起始块地址

点击右上角的【OK】，第一个数据文件调入完毕，分区列表上就可显示出数据文件名字、起始块地址和烧写的块数：





图 5.1.11 第一个数据文件调入完毕

## 2. 调入第二个数据文件：

依据第一次调入数据文件过程，大部分的操作设置相一致，不同之处需要修改数据文件的起始块地址。

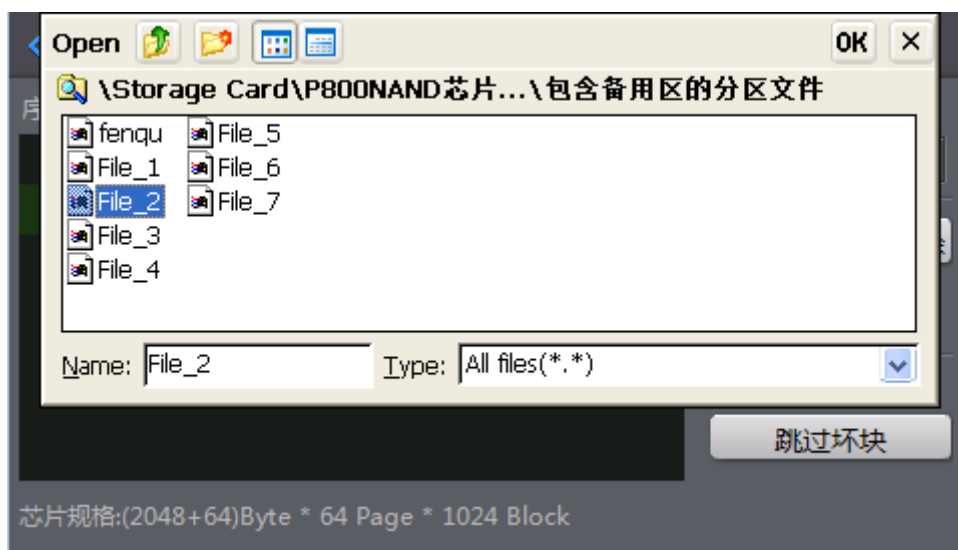


图 3.1.12 选择【File\_2】数据文件

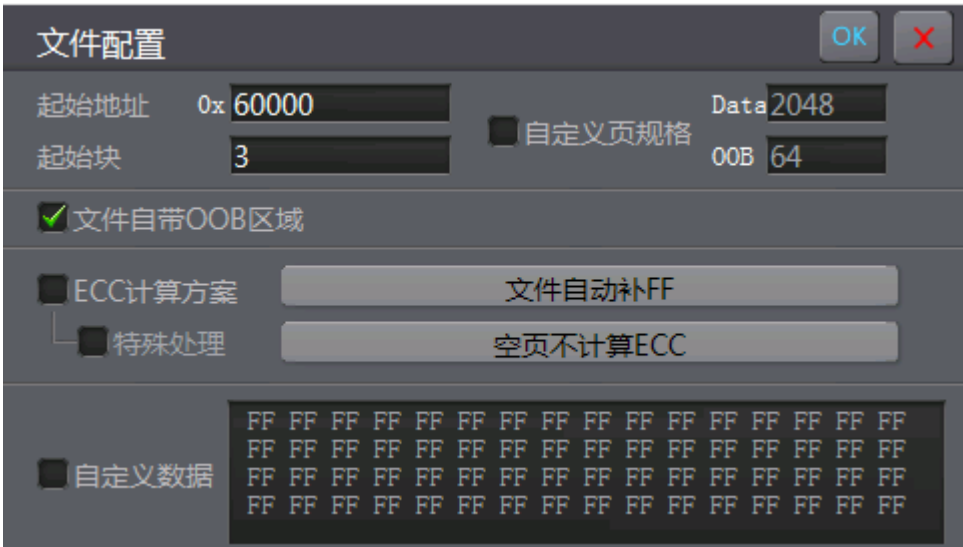


图 3.1.13 设置【File\_2】数据文件的起始块地址

点击右上角的【OK】，第二个数据文件也调入完毕：



图 3.1.14 【File\_2】数据文件调入完毕

3. 依次调入其他数据文件：

按照上述操作方式，依次把剩下的数据文件都调入到缓冲区上。在【烧写配置】界面上，就会相应显示调入的文件名、起始块和烧写的块数。



图 3.1.15 所有的数据文件依次调入完毕

5.1.5 坏块管理方案选择

完成【调入文件】之后，需要选择对应的坏块管理方案；  
本操作演示采用的坏块管理方案为跳过坏块方式，所以在【坏块管理】处选择：跳过坏块。



图 5.1.16 选择坏块管理方案

点击左上角的【配置】，则完成数据文件调入和坏块管理方案的相关配置。



图 5.1.17 返回配置，完成相关配置

5.1.6 通道配置

在【配置】页面，点击【通道配置】，即可设置通道选择，编程器默认只使能了第一个通道，用户批量生产时，需求同时烧录多个芯片，则需要把需求的通道使能上。



图 5.1.18 通道配置



图 5.1.19 通道使能

### 5.1.7 保存工程

完成【配置】设置后，返回【主界面】，首先要【保存】当前操作的工程文件；

**重要说明：**计算校验和会花费一定时间，为了保证工程文件数据的准确性，建议作计算校验和；数据校验的值，会在工程文件下面显示，可方便工厂核对校准！

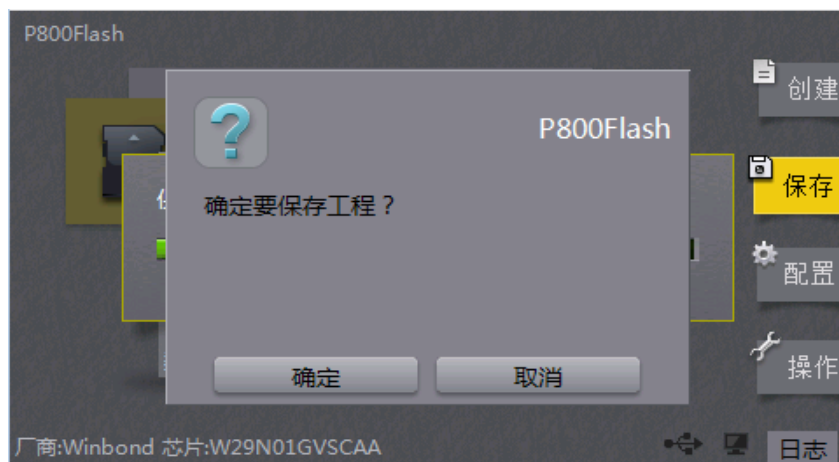


图 5.1.22 保存工程

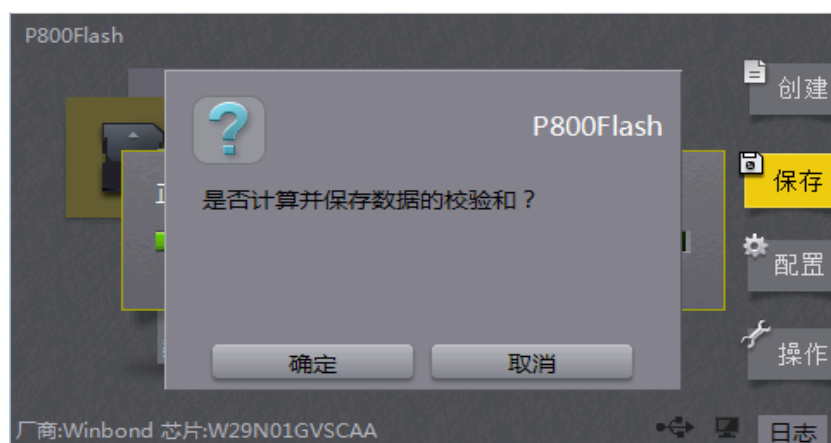


图 5.1.23 计算校验和



图 5.1.24 数据校验和

## 5.2 方式二、调入镜像数据文件+分区文件方案

用户提供芯片的烧录镜像数据文件和分区文件；镜像数据文件一般为.bin 格式，分区文件目前编程器软件只支持了.mbn 和.def 两种格式。

注：烧录之前，请先把镜像数据文件和分区文件保存在 SD 卡上！

**案例操作演示：**本调入方案的操作流程指南，仅供参考：

提供的 NandFlash 烧录方案要素：

- ◆ 烧写的芯片型号 W29N01GVSI AA，芯片规格是：(2048+64)\*64\*1024；
- ◆ 提供 132M 的镜像数据文件；
- ◆ 提供.mbn 格式的分区文件；
- ◆ 采用的坏块管理策略：跳过坏块'

镜像数据文件“Data.bin”和分区文件“Partition.mbn”



	Data.bin	2017/2/14 9:04	BIN 文件	135,168 KB
	Partition.mbn	2017/2/14 10:45	MBN 文件	1 KB

图 5.2 数据文件和分区文件

### 5.2.1 创建工程

点击【创建】，即可创建工程，如图 5.2.1 所示：



图 5.2.1 主界面

点击【创建】，进入【新建工程】页面，新建工程名称和工程类型，如图 5.2.2 所示：

【工程名称】：工程名称由字母和数字组成；

【工程类型】：存放的工程可以在本机、SD 卡或者 U 盘上；

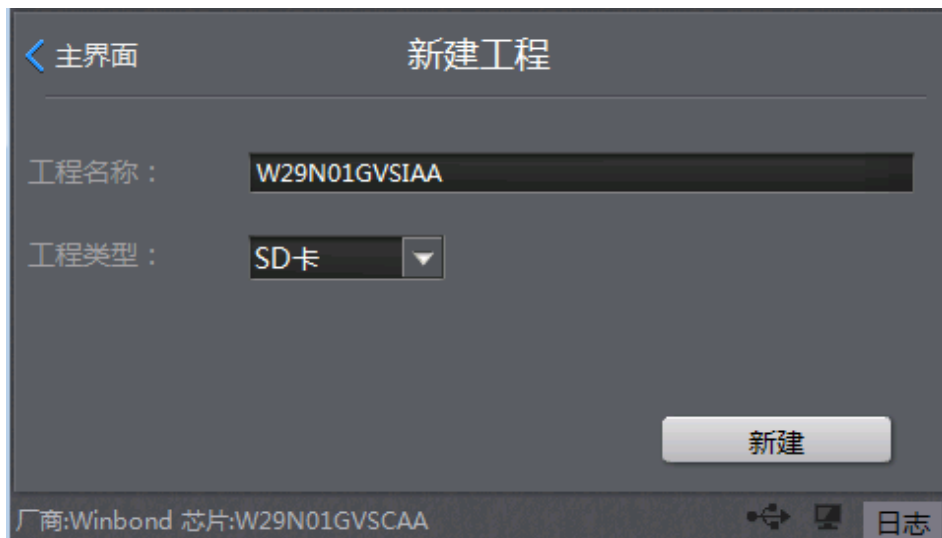


图 5.2.2 新建工程

### 5.2.2 选择芯片

完成输入【工程名称】之后，点击【新建】，进入【芯片选择】页面，如图 5.2.3 所示：选择对应芯片型号、烧录方式和适配座型号。

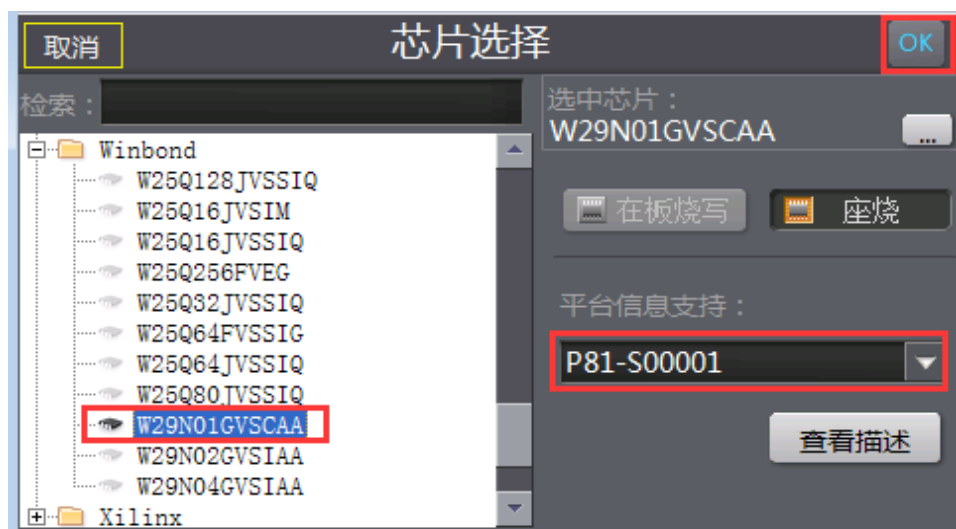


图 5.2.3 选择芯片

### 5.2.3 烧录模式

完成【芯片选择】操作之后，点击右上角的【OK】，进入【配置】页面，如图 5.2.4 所示：

点击【烧写配置】，编程器提供有两种可选择的烧录模式：【高速模式】和【普通模式】：

- ◆ 【高速模式】：必须使用原厂提供的高速 SD 卡，特点烧录速度快；
- ◆ 【普通模式】：对 SD 卡要求不高，缺点是烧录速度慢；

**使用注意：**

- 【高速模式】必须使用原厂提供的 SD 卡，可实现快速烧录，提高生产的效率。
- 由于两种烧录模式的 SD 卡存储数据方式不同，切换【烧录模式】，编程器软件会

清空缓冲区数据和分区列表，用户必须要重新调入烧录的数据文件。



图 5.2.4 烧写配置



图 5.2.5 烧录模式

#### 5.2.4 调分区文件

在【烧写配置】页面，点击【调入文件】按钮：

注：调入的文件应为 “Partition.mbn” 分区文件。





图 5.2.6 点击调入文件

进入“选择文件”的界面，首先更改设置文件类型(Type)，选择的格式类型：\*.mbn：

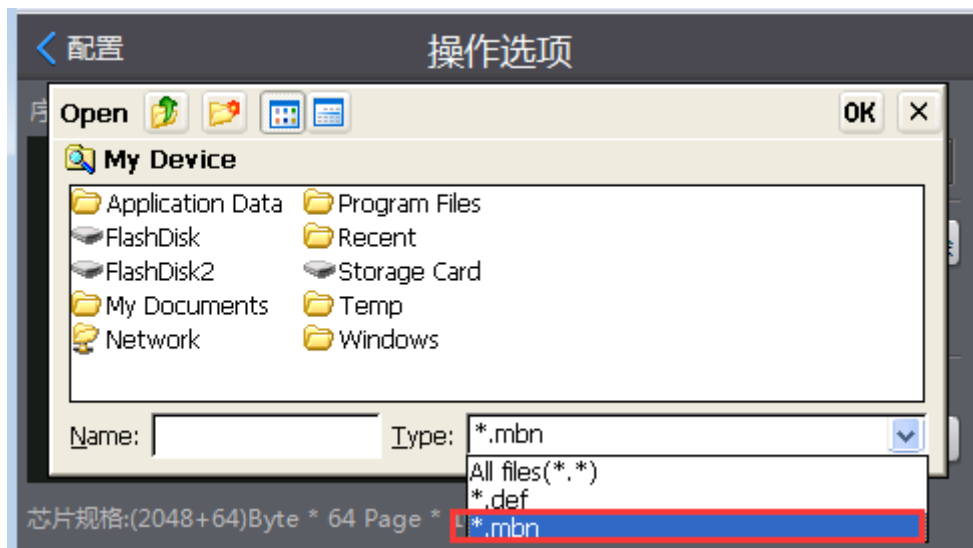


图 5.2.7 选择“文件格式”

在 SD 卡上选中“Partition.mbn” 分区文件：



图 5.2.8 选择分区文件

“Partition.mbn” 分区文件选好之后，弹出“文件路径设置”对话框，从【分区】看到，分区文件已被选择：



图 5.2.9 完成加载分区文件

“分区”已经完成选择，接着需要加载“Data.bin”数据文件：

**重要说明：**数据文件和分区文件在同一个文件夹内，并且文件名字一致，则编程器软件会自动选中数据文件！



图 5.2.10 加载分区文件

需在在 SD 卡上重新选择文件类型(Type)，类型为 All file(\*.\*)，即显示所有的文件；选中要烧写的镜像数据文件“Data.bin”：

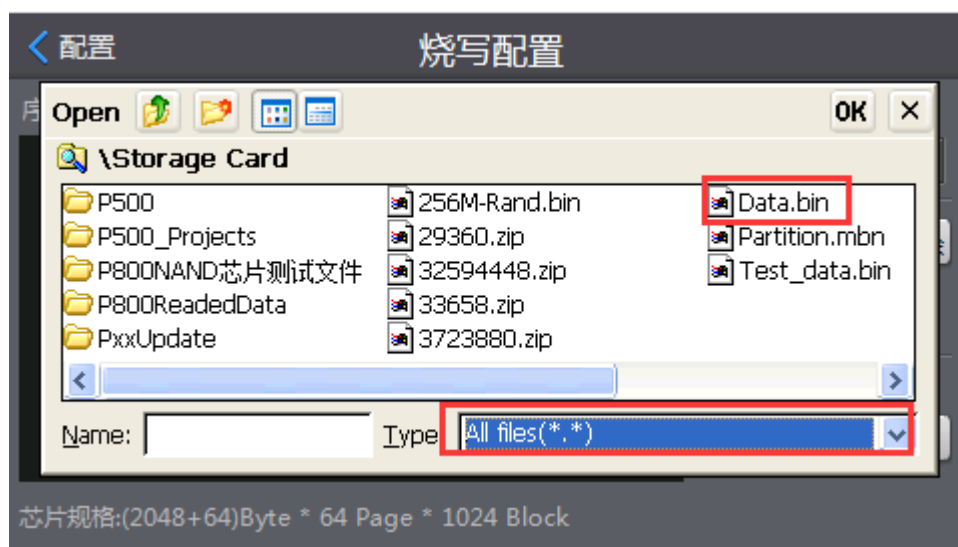


图 5.2.10 选中要烧写的镜像数据文件“Data.bin”

分区文件和镜像数据文件都完成选择之后，点击【确定】按钮，数据就会自动加载到编程器软件的缓冲区上，而分区文件也相应地加载到【烧写配置】的分区列表上：



图 5.2.11 “Partition.mbn”、“Data.bin” 完成选择



图 5.2.12 生成缓冲区数据

返回到【烧写配置】，可看到分区文件加载到分区列表上；



图 5.2.13 生成分区列表

5.2.5 坏块管理方案选择

完成【调入文件】之后，需要选择对应的坏块管理方案；  
本操作演示采用的坏块管理方案为跳过坏块方式，所以在【坏块管理】处选择：跳过坏块。



图 5.2.14 选择坏块管理方案

点击左上角的【配置】，则完成数据文件和分区文件的调入，坏块管理方案等配置。



图 5.2.15 返回配置，完成文件和坏块管理方案的配置

5.2.6 通道配置

在【配置】页面，点击【通道配置】，即可设置通道选择，编程器默认只使能了第一个通道，用户批量生产时，需求同时烧录多个芯片，则需要把需求的通道使能上。

NandFlash 烧写，目前只支持同步操作模式；



图 3.2.16 通道配置



图 3.2.17 通道使能

### 5.2.7 保存工程

完成【配置】设置后，返回【主界面】，首先要【保存】当前操作的工程文件；

**重要说明：**计算校验和会花费一定时间，为了保证工程文件数据的准确性，建议作计算校验和；数据检验的值，会在工程文件下面显示，可方便工厂核对校准！

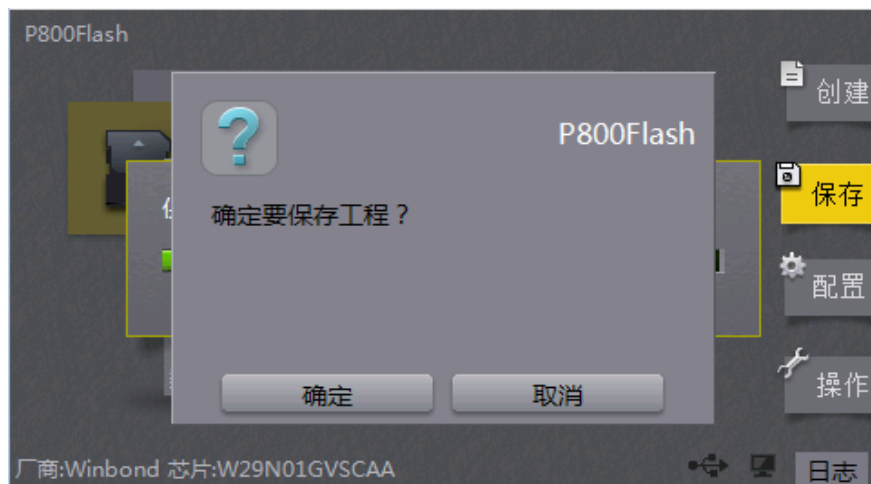


图 5.1.22 保存工程

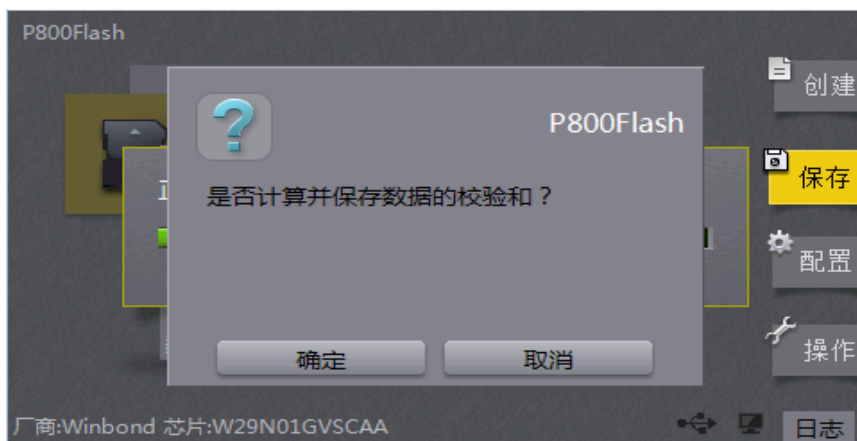


图 5.1.23 计算校验和



图 5.1.24 数据校验和

### 5.3 方式三、读母片方案

读母片方案，则“母片分析功能”，读母片方案分有两种方式：

- ◆ 分析芯片：提供能正常运行的芯片作为母片，把母片放置在编程器的第一个烧录座上，编程器对芯片内的数据进行读取分析，分析完之后，即可点击量产烧录。
- ◆ 分析文件：提供镜像数据文件，编程器对数据文件进行分析，分析完之后，即可点击两次烧录；

**重要说明：**以芯片进行分析的，最好能确保母片是没有坏块的，这样母片分析会更加准确，成功率更高；以镜像数据文件进行分析的，请务必先把烧录文件保存在 SD 卡上。

#### 5.3.1 创建工程

点击【创建】，即可创建工程，如图 5.3.1 所示：





图 5.3.1 主界面

点击【创建】，进入【新建工程】页面，新建工程名称和工程类型，如图 5.3.2 所示：

【工程名称】：工程名称由字母和数字组成；

【工程类型】：存放的工程可以在本机、SD 卡或者 U 盘上；

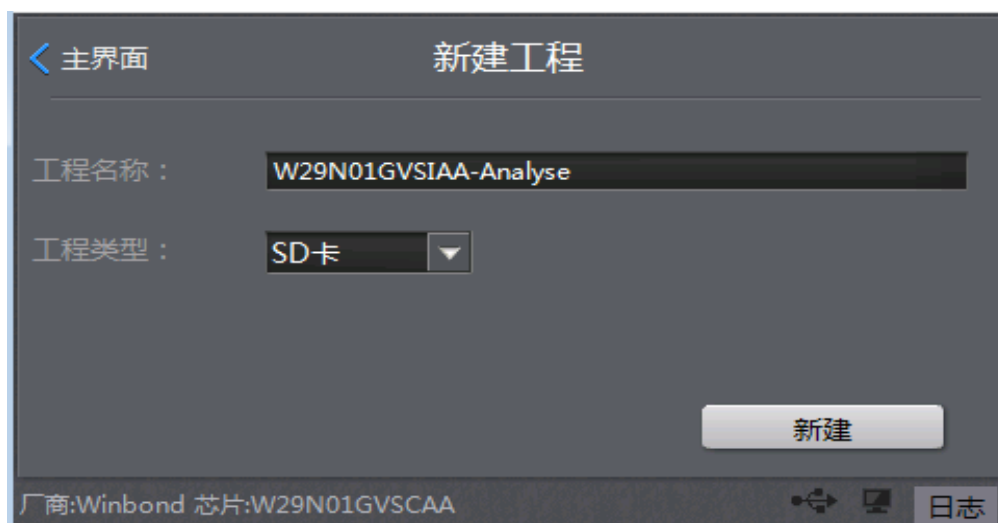


图 5.3.2 新建工程

### 5.3.2 选择芯片

完成输入【工程名称】之后，点击【新建】，进入【芯片选择】页面，如图 5.3.3 所示：选择对应芯片型号、烧录方式和适配座型号。

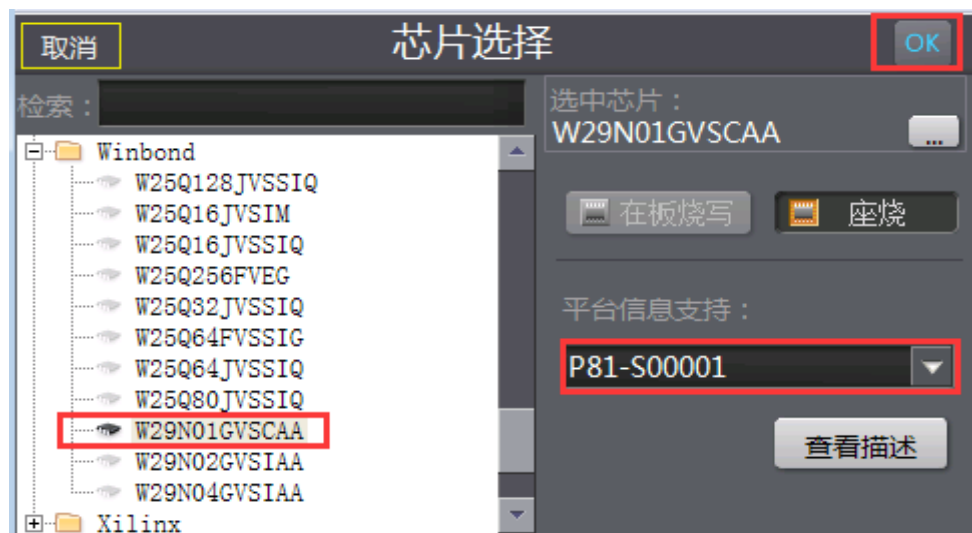


图 5.3.3 选择芯片

### 5.3.3 烧录模式

完成【芯片选择】操作之后，点击右上角的【OK】，进入【配置】页面，如图 5.1.4 所示：

点击【烧写配置】，编程器提供有两种可选择的烧录模式：【高速模式】和【普通模式】：

- ◆ 【高速模式】：必须使用原厂提供的高速 SD 卡，特点烧录速度快；
- ◆ 【普通模式】：对 SD 卡要求不高，缺点是烧录速度慢；

**使用注意：**

- 【高速模式】必须使用原厂提供的 SD 卡，可实现快速烧录，提高生产的效率。
- 由于两种烧录模式的 SD 卡数据存储方式不同，切换【烧录模式】必须要重新调入烧录文件。



图 5.3.4 烧写配置



图 5.3.5 烧录模式

### 5.3.4 母片分析

在【烧写配置】界面，点击【母片分析】，编程器进入母片分析界面，如图 5.3.7 所示：



图 5.3.6 点击【母片分析】

**重要说明：**分析的对象是芯片的，请选择“母片”；分析的对象为数据文件的，请选择“文件”；本操作演示使用的是芯片，所以选择了“母片”。



图 5.3.7 母片分析界面

**重要说明：**

- ◆ 若用户不清楚母片的文件分区情况下，可以按照默认设置，直接点击【开始分析】分许母片数据，但前提必须保证母片是没有坏块的；
- ◆ 若用户清楚了解母片的文件分区情况，可以通过先预设分区，设置好每个分区的起始块地址；在预设分区环境下，母片分析的成功率会更高一点；

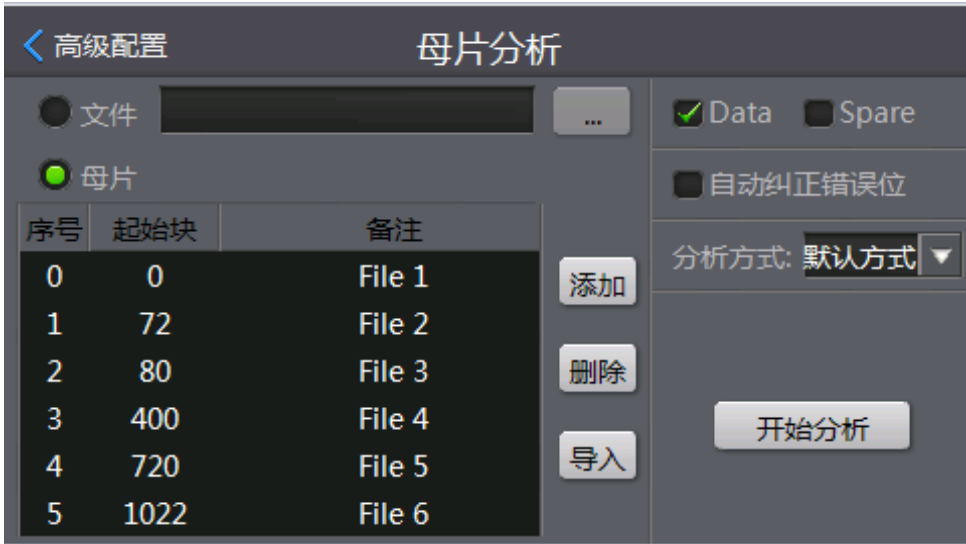


图 5.3.8 预设分区

点击【开始分析】，编程器会智能分析芯片的数据；

母片分析完成之后，读取的数据自动导入到编程器软件的缓冲区上，文件分区布局也导入到【烧写配置】的分区列表上：



图 5.3.9 完成母片分析



图 5.3.10 烧写配置的分區列表

5.3.5 坏块管理方案选择

完成【母片分析】之后，需要选择对应的坏块管理方案；  
本操作演示采用的坏块管理方案为跳过坏块方式，所以在【坏块管理】处选择：跳过坏块。



图 5.3.12 选择坏块管理方案

点击【烧写配置】界面左上角的【配置】，则完成坏块管理方案等配置。



图 5.3.13 返回配置

5.3.6 通道配置

在【配置】页面，点击【通道配置】，即可设置通道选择，编程器默认只使能了第一个通道，用户批量生产时，需求同时烧录多个芯片，则需要把需求的通道使能上。

NandFlash 烧写，目前只支持同步操作模式；



图 5.3.14 通道配置



图 5.3.15 通道使能

### 5.3.7 保存工程

完成【配置】设置后，返回【主界面】，首先要【保存】当前操作的工程文件；

**重要说明：**计算校验和会花费一定时间，为了保证工程文件数据的准确性，建议作计算校验和；数据检验的值，会在工程文件下面显示，可方便工厂核对校准！

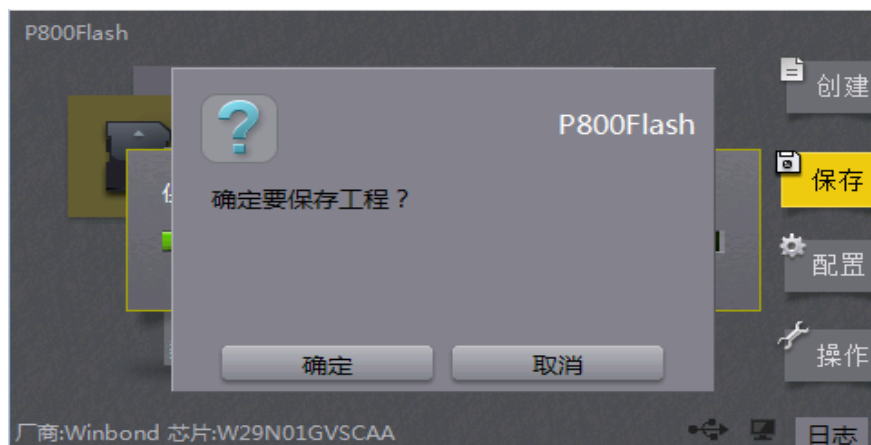


图 5.1.22 保存工程

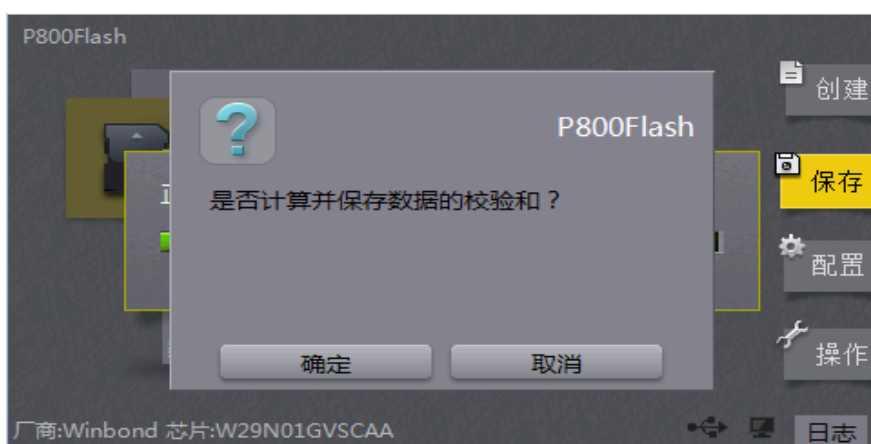


图 5.1.23 计算校验和



图 5.1.24 数据校验和

## 5.4 方式四、定制方案

由于 NandFlash 的特殊性，NandFlash 烧录与一般的 Flash 有所不同。当前绝大多数的 NandFlash 烧录，都存在着定制成分，采取简单的烧录手段是不能解决问题的，需要编程器厂商、用户、方案设计者三方共同合作定制烧录方案。具体操作流程见如下：



### 5.4.1 用户定制流程

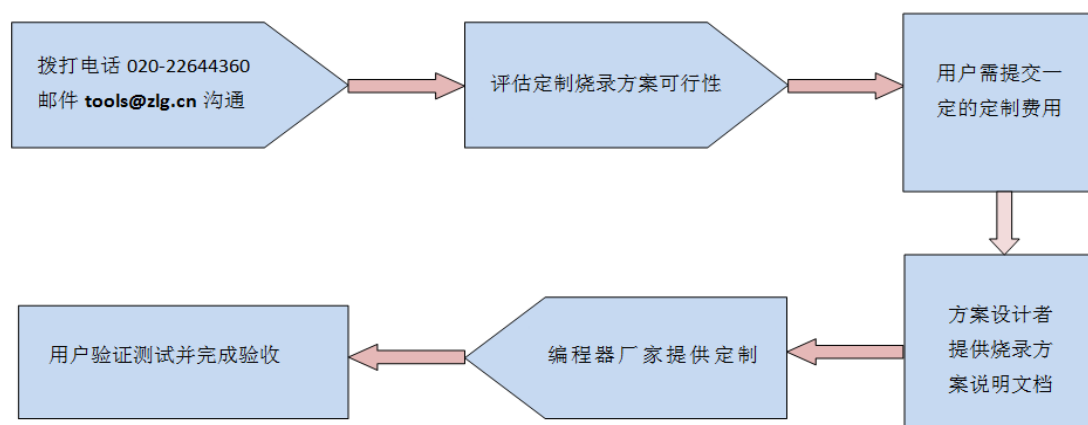


图 5.4.1 定制流程

重要说明：评估定制烧录方案可行性前，客户需要先详细填写《NandFlash 烧录方案信息表》，提供相关的说明文档，提供一个熟悉烧录方案的软件工程师作为技术接口人。

### 5.4.2 定制烧写方案要素

编程器烧录 NandFlash，除了编程器能对芯片正常烧写数据外，编程器软件还需要的清楚以下四大要素：

#### 1. 分区(Partitioning):

定义分区的实质是定义数据会如何写入 NandFlash，不同内容的数据写到对应的地址中。一般用户会有多个区，比如 boot、kernel、fs、user 等分区。

分区的描述：分区的地址范围（起始块、结束块），镜像文件大小（Image Size）。

#### 2. ECC:

ECC 存在于 NAND 每页的备用区（Spare Area）中，它允许外部系统发现主区的数据是否有误。在大多数情况下，ECC 算法可以纠正误码。NandFlash 在使用中也可能会出现坏块，所以 ECC 是非常有必要的。

不同的用户可能会使用不同的 ECC 算法，一般来说 ECC 算法由处理器供应商提供，如果编程器软件中无这个 ECC 算法，则需要用户提供 ECC 算法源代码。如果用户不使用调入文件，而是使用读母片的方式烧录，并且无动态数据，则可以不考虑 ECC 算法，因为母片中的备用区已计算好 ECC，直接将母片的备用区拷贝至其他芯片即可。

#### 3. 坏块管理:

坏块处理策略定义了遇到坏块时算法应该如何处理，基本的坏块处理策略有：跳过坏块、替换表，定制的比如 BBT 等！

#### 4. 动态数据:

动态数据是根据每颗芯片的具体情况写入不一样的数据，这些数据必须是有固定算法的，否则无法烧写。烧写这些动态数据则需要定制。

## 6. NandFlash 芯片烧录

### 6.1 量产烧录

点击【操作】，进入操作页面，选择【量产】，即可实现批量生产



图 61 主界面



图 6.2 量产烧录

### 6.2 芯片测试

为了保证用户批量烧录芯片的安全性，建议用户先烧录几颗芯片，然后上板测试验证；验证通过了，先小批量生产测试，再大批量投入生产！



图 6.3 量产烧录

### 6.3 日志文件

操作日记可以实时记录该工程文件下的所有日期的烧写信息,用户通过查看日记的烧录信息,可以跟踪或者追溯先前的烧录情况!

#### 6.3.1 即时信息

当烧录失败或者烧录过程中需要知道当前的烧写信息,如上图所示,可以点击右下角的日志按钮直接查看即时的烧写信息。



图 6.4 即时信息

#### 6.3.2 Log 文件

在烧录工程文件目录下,记录并保存着完整的操作日记文件,用户通过查阅日记文件即可查看所有时间日期的烧录信息:

路径如下: \Storage Card\P500\_Projects\NandFlash-demo\Temp\LogInfos 里面分别记

录有 4 个烧录头的烧录信息！

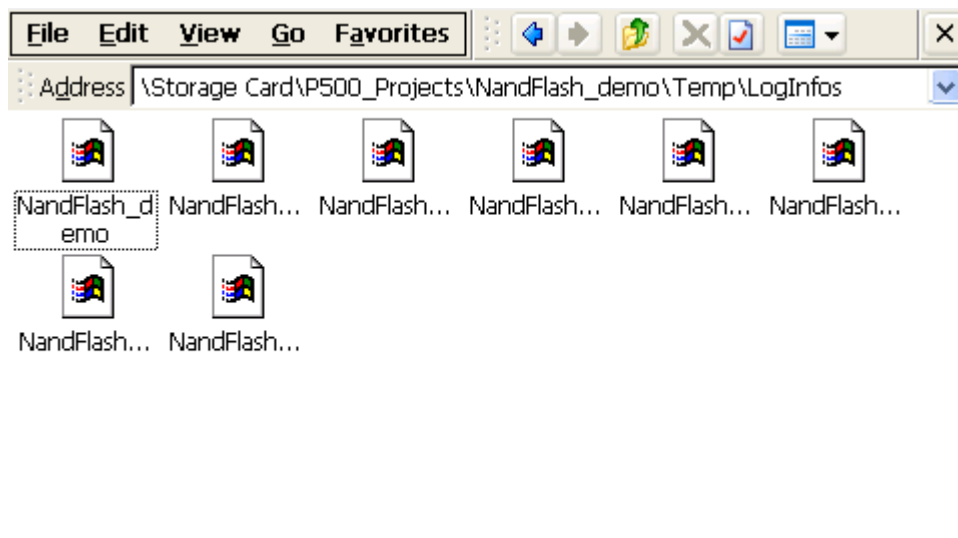


图 6.5 Log 文件

## 7. 免责声明

此使用手册的著作权属于广州致远电子有限公司。任何个人或者是单位，未经广州致远电子有限公司同意，私自使用此手册进行商业往来，导致或产生的任何第三方主张的任何索赔、要求或损失，包括合理的律师费，由您赔偿，广州致远电子有限公司与合作公司、关联公司不承担任何法律责任。

广州致远电子有限公司特别提醒用户注意：广州致远电子有限公司为了保障公司业务发展和调整的自主权，广州致远电子有限公司拥有随时自行修改此手册而不需通知用户的权利，如有必要，修改会以通告形式公布于广州致远电子有限公司网站重要页面上。