|  |
| --- |
| Anitoa |
| ESF Protocol spec. |
| V1.0 |

|  |
| --- |
| R.L  2020/11/15 |

Update history

|  |  |  |  |
| --- | --- | --- | --- |
| V1.0 | 2020-11-15 |  | 初稿 |

目录

[1. 机器指令定义: 4](#_Toc56376991)

[1.1指令包格式 4](#_Toc56376992)

[1.1.1封包格式 4](#_Toc56376993)

[1.1.2 status代码 4](#_Toc56376994)

[1.2指令集 5](#_Toc56376995)

[1.2.1 CMD: 0x1 （set） 5](#_Toc56376996)

[1.2.2 CMD: 0x9 （flash program op） 5](#_Toc56376997)

[1.2.3 CMD: 0x4 （read） 8](#_Toc56376998)

[2. 运行命令流程 9](#_Toc56376999)

[2.1 program flash 9](#_Toc56377000)

# 1. 机器指令定义:

## 1.1指令包格式

### 1.1.1封包格式

Host send to device format

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0xAA | CMD | length | type | data | checksum | 0x17 | 0x17 |

Device reply host format

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0xAA | status | CMD | length | type | data | checksum | 0x17 | 0x17 |

Length: 包含type以及有效数据data部分的长度

Checksum: 从0xAA之后开始，计算到data最后一个字节，

＊如果checksum结果为0x17，则加一变为 0x18

　　　设备回复包中的status是设备对上本次收到的命令的执行状态。

　　　0x00 -- ACK, 表明命令已经被正确解析，执行

### 1.1.2 status代码

|  |  |
| --- | --- |
| 0 | ACK |
| 1 | 数据包长度错误 |
| 2 | CMD错误 |
| 3 | 图像行号越界 |
| 4 | checksum错误 |
| 5 | TYPE错误 |
| 6 | 数据段错误 |
| 7 | 参数超出范围 |
| 8 | 未定义错误 |
| 9 | 温控命令执行超时 |
| 10 | 图像命令执行超时 |

## 1.2指令集

### 1.2.1 CMD: 0x1 （set）

本组指令设备回复包，只有确认信息，无数据返回．

回复包以第一条为示例，不同指令回复仅type有差别。

**Type: 0x02**

ESF RESET

发送数据包：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| header | CMD | length | type | data | checksum |  |  |
| 0xAA | 0x1 | 2 | **0x2** | xx | xx | 0x17 | 0x17 |

设备回复包 :

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| header | status | CMD | length | type | data | checksum |  |  |
| 0xAA | ACK | 0x1 | 1 | **0x2** | ------ | xx | 0x17 | 0x17 |

### 1.2.2 CMD: 0x9 （flash program op）

本组指令设备回复包，包含确认信息，以及数据返回．

回复包格式以第一条为示例，不同指令type有差别，不在重复解释。

每条指令依据数据不同，回复length（长度）也会不同，不再重复解释。

返回的数据含义，参照每条指令后面的详细解释

***Type: 0x01***

Chip erase

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| header | CMD | length | type | data | checksum |  |  |
| 0xAA | 0x9 |  | 0x1 | Data (2 byte) | xx | 0x17 | 0x17 |

Data format: byte 0 --- bit[4:0] chip index

Byte 1 --- trim data: Ers\_HV

**Type: 0x02**

Sector erase

发送数据包：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| header | CMD | length | type | data | checksum |  |  |
| 0xAA | 0x9 |  | 0x2 | 4 byte | xx | 0x17 | 0x17 |

Data format:

Data 0 -- bit[4:0] chip index

Data 1 -- trim data: Ers\_HV

Data 2 -- address LSB

Data 3 -- address MSB

**Type: 0x03**

**Write packet**

发送数据包：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| header | CMD | length | type | data | checksum |  |  |
| 0xAA | 0x9 |  | 0x3 |  | xx | 0x17 | 0x17 |

Data format:

Byte [0] -- packet index (start from 0)

Byte [1] -- valid data length in current packet

Byte[2...x] -- data

**Type: 0x04**

**Program command**

发送数据包：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| header | CMD | length | type | data | checksum |  |  |
| 0xAA | 0x9 |  | 0x4 |  | xx | 0x17 | 0x17 |

Data format:

Byte [0] -- Flash chip idx

Byte [1] -- trim data PrgDig\_HV

Byte [2] -- trim data ENIP\_PrgDig

Byte [3] -- trim data ENBIAS\_PrgDig

Byte [4] -- length (u16) LSB

Byte [5] -- length (u16) MSB

Byte [6] -- Xaddr LSB

Byte [7] -- Xaddr MSB

Byte [8] -- Yaddr

**Type: 0x05**

**Programming test**

发送数据包：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| header | CMD | length | type | data | checksum |  |  |
| 0xAA | 0x9 |  | 0x5 |  | xx | 0x17 | 0x17 |

Data format:

Byte[0] -- chip index

Byte[1] -- target 1 byte data to be programmed

Byte[2] -- trim data: Ers\_HV

Byte[3] -- trim data: ENIP\_PRGDIG

Byte[4] -- trim data: ENBIAS\_PRGDIG

### 1.2.3 CMD: 0x4 （read）

**Type: 0x01**

Digital read test

发送数据包：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| header | CMD | length | type | data | checksum |  |  |
| 0xAA | 0x4 |  | **0x1** | 6 byte | xx | 0x17 | 0x17 |

Data format: data 0 -- chip index

Data 1 --- YAddr

Data 2 --- Xaddr LSB

Data 3 --- Xaddr MSB

Data 4 -- read length

设备回复包 :

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| header | status | CMD | length | type | data | checksum |  |  |
| 0xAA | ACK |  |  |  |  | xx | 0x17 | 0x17 |

**Type: 0x27**

Version info (版本信息)

发送数据包：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| header | CMD | length | type | data | checksum |  |  |
| 0xAA | 0x4 |  | 0x27 | Not care | xx | 0x17 | 0x17 |

设备回复包 :

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| header | status | CMD | length | type | data | checksum |  |  |
| 0xAA | ACK | 0x4 |  | **0x27** | 18 byte | xx | 0x17 | 0x17 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| host型号 | 主板本号 | 次版本号 | 年 | 月 | 日 | IMG 型号 | 主板本号 | 次版本号 | 年 | 月 | 日 | TEMP 型号 | 主板本号 | 次版本号 | 年 | 月 | 日 |

Data 解释：

# 2. 运行命令流程

## 2.1 program flash

**Write data packet**：

* cmd=0x09 type 0x3, 写入数据进入MCU buffer

**Program start**：

Cmd 0x09 type 0x4，指定flash chip， trim setting, address, data length

MCU会从之前写入的data packet顺序编程进入指定的flash中

**Programming test ：**

Cmd 0x09 type 0x5，指定flash chip，

MCU会从地址0开始编程指定的固定数据，长度为32 word

上位机发送 cmd 0x4， type 0x27 读**版本信息**时

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 型号 | 主板本号 | 次版本号 | 年 | 月 | 日 |

下位机返回数据段

**其中型号代码 为06**