

USB 设备请求

目录

1 标准设备请求	1
1.1 读取状态请求 GetStatus	2
1.2 清除特性请求 ClearFeature	3
1.3 设定特性请求 SetFeature	4
1.4 设置地址请求 SetAddress	4
1.5 读取描述符请求 GetDescriptor	4
1.6 设置描述符请求 SetDescriptor	5
1.7 读取配置请求 GetConfiguration	5
1.8 设置配置请求 SetConfiguration	5
1.9 读取接口请求 GetInterface	5
1.A 设置接口请求 SetInterface	5
1.B 同步帧请求 SynchFrame	6

表格目录

表格 1 11 种标准设备描述符	2
表格 2 USB 请求数据格式	2
表格 3 GetStatus 请求读取设备数据格式	3
表格 4 GetStatus 请求读取设备返回数据格式	3
表格 5 GetStatus 读取接口请求数据格式	3
表格 6 GetStatus 读取接口请求返回数据格式	3
表格 7 GetStatus 读取端点请求数据格式	3
表格 8 读取端点请求返回数据格式	3
表格 9 ClearFeature 请示数据格式	3
表格 10 SetFeature 请求数据格式	4
表格 11 SetAddress 请求数据格式	4
表格 12 GetDescriptor 请求数据格式	4
表格 13 SetDescriptor 请求数据格式	5
表格 14 GetConfiguration 请求数据格式	5
表格 15 SetConfiguration 请求数据格式	5
表格 16 SetInterface 请求数据格式	6
表格 17 SynchFrame 请求数据格式	6

1 标准设备请求

USB 设备的信息和功能都存在它的设备描述符中，要想得到这些信息，或是修改这些信息，就要对描述符进行读写操作，完成这些读写的行为称为 **设备请求**。**设备请求**实际上就是主机对设备的读写的。设备请求分为

两类，一类是标准的，必需实现的，称为**标准设备请求**，共 11 个。另一类是设备制造者自定义的，称为**自定义设备请求**。

11 种标准的 USB 设备请求列表如下：

表格 1 11 种标准设备描述符

USB 请求	USB 请求号	功能描述
<u>GetStatus</u>	0x00	读取 USB 设备、接口或端点状态
<u>ClearFeature</u>	0x01	清楚或禁止 USB 设备、接口或端点的某些特性
<u>SetFeature</u>	0x03	设置或使能 USB 设备、接口或端点的某些特性
<u>SetAddress</u>	0x05	把分配给设备的地址告诉它
<u>GetDescriptor</u>	0x06	读取设备描述符
<u>GetDescriptor</u>	0x07	更新已有的描述符或添加新的描述符
<u>GetConfiguration</u>	0x08	读取当前的配置值
<u>SetConfiguration</u>	0x09	为 USB 设备选择一个合适的配置
<u>GetInterface</u>	0x0A	读取 USB 指定接口的当前可替换设置值
<u>SetInterface</u>	0x0B	为 USB 指定接口选择一个合适的可替换设置
<u>SynchFrame</u>	0x0C	读取 USB 同步端点所指定的帧序号

这十一个 USB 设备请求由主机通过控制管道发出，USB 设备必须全部支持，最低也要返回一个空响应。

当 USB 设备连接到 USB 主机上的时候，USB 主机首先通过标准的 USB 设备请求和 USB 功能设备进行通信。USB 主机读取描述符，从而获得该 USB 功能设备的产品信息和功能配置。当然也可以使用自定义的请求，来实现简单的控制。

如我们所知，USB 的控制传输分为三个阶段，即建立阶段、数据阶段和状态阶段。建立阶段由一个 Setup 令牌包、一个 DATA0 数据包和一个握手包（即一个建立事务）组成。其 DATA0 数据包恒为 8 字节长度（详情请参 <http://www.usr.cc/thread-51673-1-1.html> 的 7、9 页及 <http://www.usr.cc/thread-51676-1-1.html> 的第二页 1.4 节）。这 8 字节的位置就用于发送 USB 请求数据，其格式如下：

表格 2 USB 请求数据格式

偏移量	变量	大小	数值	描述
0	bmRequestType	1 字节	位图	第 7 位：数据方向位 0=主机到设备 1=设备到主机 第 6-5 位：类型 0=标准 1=类 2=厂商自定义 3=保留 第 4-0 位：接收者 0=对设备的请求 1=对接口的请求 2=对管道（端口的请求） 3=其他 4-31=保留
1	bRequest	1 字节	数值	请求类别
2	wValue	2 字节	数值	不同请求含义不同
4	wIndex	2 字节	数值	不同请求含义不同
6	wLength	2 字节	数值	表示需要有多少数据返回

其中，bRequest 代表不同的 USB 请求号，其值可见于表格 1 11 种标准设备描述符。

1.1 读取状态请求 GetStatus

GetStatus 请求用于 USB 主机读取 USB 设备、接口或端点的状态。

1) 获取设备状态

建立阶段中，建立事务中的数据格式：

表格 3 GetStatus 请求读取设备数据格式

bmRequestType	bmRequest	wValue	wIndex	wLength
1000 000B	0x00	0x0000	0x0000	0x0002

其中 wLength 字段为 2，表示要求设备返回 2 字节数据。下面是设备返回数据格式：

表格 4 GetStatus 请求读取设备返回数据格式

D7	D6	D5	D4	D3	D2	D1	D0
保留 置 0						远程唤醒	自供电
D15	D14	D13	D12	D11	D10	D9	D8
保留 置 0							

其中 D0 表示 USB 设备时否为自供电，为 1 表示自供电，为 0 表示供线供电。D1 表示是否支持远程唤醒，0 表示已禁用，1 表示已启用。

2) 读取接口状态

建立事务中的数据格式：

表格 5 GetStatus 读取接口请求数据格式

bmRequestType	bmRequest	wValue	wIndex	wLength
1000 0001B	0x00	0x0000	接口号	0x0002

设备返回的两个字节数据的格式：

表格 6 GetStatus 读取接口请求返回数据格式

D7	D6	D5	D4	D3	D2	D1	D0
保留 置 0							
D15	D14	D13	D12	D11	D10	D9	D8
保留 置 0							

读取接口请求返回的数据全是 0。

3) 读取端点状态

建立事务中的数据格式：

表格 7 GetStatus 读取端点请求数据格式

bmRequestType	bmRequest	wValue	wIndex	wLength
1000 0010B	0x00	0x0000	端点	0x0002

返回两字节的数据格式：

表格 8 读取端点请求返回数据格式

D7	D6	D5	D4	D3	D2	D1	D0
保留 置 0							停止
D15	D14	D13	D12	D11	D10	D9	D8
保留 置 0							

D0 位表示满意端点的是否停止，0 表示未被停止，1 表示被停止。

1.2 清除特性请求 ClearFeature

作用：清除或禁止 USB 设备、接口或端点的某些特性，该请求无数据阶段。ClearFeature 请求的定义格式如表格 9 所示：

表格 9 ClearFeature 请示数据格式

bmRequestType	bmRequest	wValue	wIndex	wLength
0000 0000B	0x01	特性选择符	0	0
0000 0001B	0x01	特性选择符	接口	0

bmRequestType	bmRequest	wValue	wIndex	wLength
0000 0010B	0x01	特性选择符	端点	0

其中 wValue 字段指明与 ClearFeature 请求的接收方相对应的设备、接口或端点的特性选择符值，用来清除或禁止相应的特性。如果 ClearFeature 请求的接收方是设备描述符，则该字段只能使用设备特性选择符的值；其它描述符同样道理。

- USB 当前支持的标准特性选择符有如下几个：
- ENDPOINT_HALT：其接收方是端点，值为 0，对应于端点的停止特性。
- DEVICE_REMOTE_WAKEUP：其接收方是设备，值为 1，对应于 USB 设备的远程唤醒功能。
- TEST_MODE：其接收方是设备，值为 2，用于 USB 设备的测试模式，它不能由 ClearFeature 请求来清除或禁止。

现在，USB 协议中还没有用于接口的特性选择符，另外 ClearFeature 请求只在 USB 设备处于地址状态和配置状态时才有效。

请求没有数据阶段。

1.3 设定特性请求 SetFeature

此请求用于设定 USB 的设备、接口或端点的特性值，和 ClearFeature 请求相对应。SetFeature 请求同样没有数据阶段。其请求格式如下：

表格 10 SetFeature 请求数据格式

bmRequestType	bmRequest	wValue	wIndex	wLength
0000 0000B	0x03	特性选择符	0	特性选择符
0000 0001B	0x03	特性选择符	接口	-
0000 0010B	0x03	特性选择符	端点	-

当 wValue 字段为 TEST_MODE 时，wIndex 字段的高字节指明设备测试选择符，低字节必须置 0。USB 协议当前支持的测试选择符如下：

- Test_J：值为 0x01
- Test_K：值为 0x02
- Test_SE0_NAK：值为 0x03
- Test_Packet：值为 0x04
- Test_Force_Enable：值为 0x05。

对于高速 USB 设备，在缺省的 USB 状态、地址状态和配置状态下，都必须支持 SetFeature 请求。

1.4 设置地址请求 SetAddress

SetAddress 请求主要用于在 USB 设备上电时候为其分配一个惟一的设备地址，该请求无数据阶段。SetAddress 请求的定义格式如下：

表格 11 SetAddress 请求数据格式

bmRequestType	bmRequest	wValue	wIndex	wLength
0000 0000B	0x05	设备地址	0	0

其中 wValue 字段为设备地址，范围是 0~127。

1.5 读取描述符请求 GetDescriptor

GetDescriptor 请求用于 USB 主机读取设备的指定的描述符，在该请求的数据阶段，USB 设备将向 USB 主机返回指定的描述符。GetDescriptor 请求的定义格式如下：

表格 12 GetDescriptor 请求数据格式

bmRequestType	bmRequest	wValue	wIndex	wLength
1000 0000B	0x06	类型和索引	0 或语言 ID	描述符长度

其中，wValue 字段的高位指明了主机所读取描述符的类型值，可选择设备描述符、配置描述符、字符串描述符、接口描述符、端点描述符、设备限定描述符和其他速率配置描述符。wValue 字段的低位字节指明描述符的索引值，只对配置描述符和字符串描述符有效。

wIndex 字段用于表示字符串描述符的语言 ID，如果用于其他描述符则应置 0。

wLength 指定了 GetDescriptor 请求所返回的字节数。

1.6 设置描述符请求 SetDescriptor

SetDescriptor 请求执行描述符的设置，与 GetDescriptor 执行相反的操作。

其数据格式如下：

表格 13 SetDescriptor 请求数据格式

bmRequestType	bmRequest	wValue	wIndex	wLength
0000 000B	0x07	类型和索引	0 或语言 ID	描述符长度

其中，wValue 字段的高位指明了主机所读取描述符的类型值，可选择设备描述符、配置描述符、字符串描述符、接口描述符、端点描述符、设备限定描述符和其他速率配置描述符。wValue 字段的低位字节指明描述符的索引值，只对配置描述符和字符串描述符有效。

wIndex 字段用于表示字符串描述符的语言 ID，如果用于其他描述符则应置 0。

wLength 指定了所设定的描述符的长度。

1.7 读取配置请求 GetConfiguration

GetConfiguration 请求主要用于主机读取 USB 设备当前的配置值，在 GetConfiguration 请求的数据阶段，USB 设备将向 USB 主机返回 1 字节的配置值。

其请求数据格式如下：

表格 14 GetConfiguration 请求数据格式

bmRequestType	bmRequest	wValue	wIndex	wLength
1000 000B	0x08	0	0	1

该请求在数据阶段会收到一个字节的返回值，USB 设备处于不同状态时，GetConfiguration 请求具有不同的响应：

地址状态：返回 0。

配置状态：配置描述符中的 bConfigurationValue 字段值。

缺省状态：无效。

1.8 设置配置请求 SetConfiguration

该请求为 USB 设备设置一个合适的配置值。它没有数据阶段，要设置的配置值在 Setup 事务的 8 字节数据包中，占用的是其 wValue 字段。请求数据格式如下：

表格 15 SetConfiguration 请求数据格式

bmRequestType	bmRequest	wValue	wIndex	wLength
0000 000B	0x09	配置值	0	0

wValue 字段的低位字节指明 USB 设备的配置值，高字节保留。

1.9 读取接口请求 GetInterface

GetInterface 请求主要用于 USB 主机读取指定接口的设置值，也就是接口描述符中的 bAlternateSetting 字段的值。在数据阶段 bAlternateSetting 字段被返回，是一个字节的数据。

该请求的数据格式如下：

bmRequestType	bmRequest	wValue	wIndex	wLength
1000 000B	0x0A	0	接口	1

1.A 设置接口请求 SetInterface

设置接口请求与读取接口请求操作对象相同，也是 bAlternateSetting 字段，所要设置的值在 Setup 事务的 8 字节数据包中传输，无数据阶段。该请求数据格式如下：

表格 16 SetInterface 请求数据格式

bmRequestType	bmRequest	wValue	wIndex	wLength
0000 000B	0x0B	设置值	接口	0

1.B 同步帧请求 SynchFrame

SynchFrame 请求用于设置并报告端点的同步帧号，用于同步传输，只适用于同步端点。在 SynchFrame 请求的数据阶段，USB 设备将向 USB 主机返回两字节的帧号数据。

SynchFrame 请求的数据格式如下：

表格 17 SynchFrame 请求数据格式

bmRequestType	bmRequest	wValue	wIndex	wLength
1000 000B	0x0C	0	端点	2

该请求只在配置状态下有效，它在数据阶段返回两个字节。