

图解 USB 协议之一 枚举过程

关键字: HD-USB12 协议分析仪 枚举

前不久花了 2300 大银败了个 HD-USB12 协议分析仪 (肉痛...)。不过还蛮好用, 以后做 USB 固件开发就方便多了。

看到 21ic 坛子里 [arthur0561](#) 大虾发了一系列 USB 协议入门总结的文章, 向他 (OR 她?) 表示一下敬意。我也准备用手头这台协议分析仪以实例和图解的方式重复叙述一下, 希望对初学 USB 的童鞋有些帮助。
废话少说, 首先图解一下 USB 枚举过程。

一、 总线复位及向默认地址 0 发送 GET_DESCRIPTOR 指令包, 请求设备描述。如 (图一) 所示:

Index	m:s.ms.us	Dur	Len	Err	PID	CRC	Packet
4	00:09.528.706						<Reset> / <Target Disconnected>
5	00:09.539.257						<Unreset> / <Target Connected>
6	00:09.549.258	43.01 ms					[44 SOF]
7	00:09.593.175	2.75 us			SETUP	02	DEV = 00 EP = 00
8	00:09.593.178	8.08 us	8		DATA0	94DD	GET_DESCRIPTOR
9	00:09.593.187	1.42 us			ACK		
10	00:09.593.191	4.77 us					[1 IN/NAK]
11	00:09.593.264	2.75 us					[1 SOF]
12	00:09.593.267	925.17 us					[125 IN/NAK]
13	00:09.594.257	2.75 us					[1 SOF]
14	00:09.594.260	93.10 us					[13 IN/NAK]
15	00:09.594.355	2.75 us			IN	02	DEV = 00 EP = 80
16	00:09.594.359	13.42 us	16		DATA1	1BB8	12 01 00 01 00 00 00 10 14 1B 66 06 00 01 00 00
17	00:09.594.373	1.42 us			ACK		
18	00:09.594.380	2.75 us			OUT	02	DEV = 00 EP = 00
19	00:09.594.383	2.75 us	0		DATA1		
20	00:09.594.387	1.42 us			ACK		

(图一)

1) Index[4 - 5]: 表示 USB 插入总线复位

欢迎访问avatar的电子博客: <http://blog.ednchina.com/avatar/>
Avatar 的邮箱: avatar_1210@tom.com

2) Index[7 - 8]: 表示主机向默认地址发送 GET_DESCRIPTOR 指令包, 详细信息也抓出来了, 如 (图二) 所示:

```
bmRequestType = 0x80
    D7: Direction
        1 = Device-to-Host
    D6...5: Type
        0 = Standard
    D4...0: Recipient
        0 = Device
bRequest = 0x06
    GET_DESCRIPTOR
wValue = 0x0100
    D15...8: Descriptor Type
        1 = DEVICE
    D7...0: Descriptor Index
        0
wIndex = 0x0000
wLength = 0x0040
```

(图二)

- 3) Index[15 - 17]: 表示设备向主机发送设备描述数据 Index[16]
- 4) Index[18 - 19]: 表示主机完成 GET_DESCRIPTOR 指令后, 给设备发送一个空应答

二、 再次复位总线及向设备发送 SET_ADDRESS 指令包, 设置设备地址。如 (图三) 所示:

Index	m:s.ms.us	Dur	Len	Err	PID	CRC	Packet
22	00:09.595.310						<Reset> / <Target Disconnected>
23	00:09.606.260						<Unreset> / <Target Connected>
24	00:09.616.260	15.00 ms					[16 SOF]
25	00:09.631.265	2.75 us			SETUP	02	DEV = 00 EP = 00
26	00:09.631.268	8.08 us	8		DATA0	25EB	SET_ADDRESS
27	00:09.631.277	1.42 us			ACK		
28	00:09.631.281	659.27 us					[89 IN/NAK]
29	00:09.631.943	2.75 us			IN	02	DEV = 00 EP = 80
30	00:09.631.946	2.75 us	0		DATA1		
31	00:09.631.950	1.42 us			ACK		

(图三)

- 1) Index[22 - 23]: 表示再次总线复位, 该复位自动完成, 不是手工插拔 USB 完成

欢迎访问avatar的电子博客: <http://blog.ednchina.com/avatar/>
Avatar 的邮箱: avatar_1210@tom.com

2) Index[25 - 27]: 表示主机向默认地址发送 SET_ADDRESS 指令包, 详细信息如 (图四) 所示:

```
bmRequestType = 0x00
    D7: Direction
        0 = Host-to-Device
    D6...5: Type
        0 = Standard
    D4...0: Recipient
        0 = Device
bRequest = 0x05
    SET_ADDRESS
wValue = 0x0001
    Address = 0x0001 = 1
wIndex = 0x0000
wLength = 0x0000
```

(图四)

3) Index[29 - 31]: 表示设备完成 SET_ADDRESS 指令后, 给主机发送一个空应答

三、 向第二步设定的设备地址发送 GET_DESCRIPTOR 指令包, 请求设备描述。如 (图五) 所示:

Index	m:s.ms.us	Dur	Len	Err	PID	CRC	Packet
33	00:09.677.143	2.75 us			SETUP	1D	DEV = 01 EP = 00
34	00:09.677.146	8.08 us	8		DATA0	F4E0	GET_DESCRIPTOR
35	00:09.677.154	1.42 us			ACK		
36	00:09.677.158	78.54 us					[11 IN/NAK]
37	00:09.677.262	2.75 us					[1 SOF]
38	00:09.677.265	960.14 us					[130 IN/NAK]
39	00:09.678.254	2.75 us					[1 SOF]
40	00:09.678.257	49.00 us					[7 IN/NAK]
41	00:09.678.309	2.73 us			IN	1D	DEV = 00 EP = 80
42	00:09.678.312	13.42 us	16		DATA1	1BB8	12 01 00 01 00 00 00 10 14 1B 66 06 00 01 00 00
43	00:09.678.326	1.42 us			ACK		
44	00:09.678.328	244.92 us					[33 IN/NAK]
45	00:09.678.576	2.75 us			IN	1D	DEV = 00 EP = 80
46	00:09.678.579	4.17 us	2		DATA0	8F3F	00 01
47	00:09.678.584	1.42 us			ACK		
48	00:09.678.586	2.75 us			OUT	1D	DEV = 01 EP = 00
49	00:09.678.589	2.75 us	0		DATA1		
50	00:09.678.592	1.42 us			ACK		

(图五)

欢迎访问avatar的电子博客: <http://blog.ednchina.com/avatar/>
Avatar 的邮箱: avatar_1210@tom.com

1) Index[33 - 35]: 表示主机向地址 01 发送 GET_DESCRIPTOR 指令包, 详细信息见 (图六)

```
bmRequestType = 0x80
    D7: Direction
        1 = Device-to-Host
    D6...5: Type
        0 = Standard
    D4...0: Recipient
        0 = Device
bRequest = 0x06
    GET_DESCRIPTOR
wValue = 0x0100
    D15...8: Descriptor Type
        1 = DEVICE
    D7...0: Descriptor Index
        0
wIndex = 0x0000
wLength = 0x0012
```

(图六)

- 2) Index[41 - 43]: 表示设备向主机发送设备描述数据 Index[42]
- 3) Index[45 - 47]: 表示设备向主机发送设备描述数据 Index[46]
- 4) Index[48 - 50]: 表示主机完成 GET_DESCRIPTOR 指令后, 给设备发送一个空应答

四、 向第二步设定的设备地址发送 GET_DESCRIPTOR 指令包, 请求配置描述。如 (图七) 所示:

Index	m.s.ms.us	Dur	Len	Err	PID	CRC	Packet
52	00:09.679.283	2.75 us			SETUP	1D	DEV = 01 EP = 00
53	00:09.679.286	8.08 us	8		DATA0	04AE	GET_DESCRIPTOR
54	00:09.679.295	1.42 us			ACK		
55	00:09.679.299	917.52 us					[125 IN/NAK]
56	00:09.680.244	2.75 us					[1 SOF]
57	00:09.680.247	2.75 us			IN	1D	DEV = 01 EP = 80
58	00:09.680.250	8.75 us	9		DATA1	B873	09 02 29 00 01 01 00 60 01
59	00:09.680.260	1.42 us			ACK		
60	00:09.680.262	2.75 us			OUT	1D	DEV = 01 EP = 00
61	00:09.680.265	2.75 us	0		DATA1		
62	00:09.680.268	1.42 us			ACK		

(图七)

欢迎访问avatar的电子博客: <http://blog.ednchina.com/avatar/>

Avatar 的邮箱: avatar_1210@tom.com

1) Index[52 - 54]: 表示主机向地址 01 发送 GET_DESCRIPTOR 指令包, 详细信息见 (图八)

```
bmRequestType = 0x80
    D7: Direction
        1 = Device-to-Host
    D6...5: Type
        0 = Standard
    D4...0: Recipient
        0 = Device
bRequest = 0x06
    GET_DESCRIPTOR
wValue = 0x0200
    D15...8: Descriptor Type
        2 = CONFIGURATION
    D7...0: Descriptor Index
        0
wIndex = 0x0000
wLength = 0x0009
```

(图八)

2) Index[57 - 59]: 表示设备向主机发送配置描述数据 Index[58]

3) Index[60 - 62]: 表示主机完成 GET_DESCRIPTOR 指令后, 给设备发送一个空应答

五、 重复向第二步设定的设备地址发送 GET_DESCRIPTOR 指令包, 取完所有设备描述及配置描述;

六、 向第二步设定的设备地址发送 SET_CONFIGURATION 指令包, 设置配置描述。如 (图九) 所示:

欢迎访问avatar的电子博客: <http://blog.ednchina.com/avatar/>

Avatar 的邮箱: avatar_1210@tom.com

Index	m:s.ms.us	Dur	Len	Err	PID	CRC	Packet
139	00:09.733.229	2.75 us			SETUP	1D	DEV = 01 EP = 00
140	00:09.733.232	8.08 us	8		DATA0	2527	SET_CONFIGURATION
141	00:09.733.241	1.42 us			ACK		
142	00:09.733.245	631.50 us					[86 IN/NAK]
143	00:09.733.879	2.75 us			IN	1D	DEV = 00 EP = 80
144	00:09.733.882	2.75 us	0		DATA1		
145	00:09.733.886	1.42 us			ACK		

(图九)

1) Index[139 - 141]: 表示主机向地址 01 发送 SET_CONFIGURATION 指令包, 详细信息见 (图十)

```
bmRequestType = 0x00
    D7: Direction
        0 = Host-to-Device
    D6...5: Type
        0 = Standard
    D4...0: Recipient
        0 = Device
bRequest = 0x09
    SET_CONFIGURATION
wValue = 0x0001
    Configuration = 0x0001 = 1
wIndex = 0x0000
wLength = 0x0000
```

(图十)

2) Index[143 - 145]: 表示设备完成 SET_CONFIGURATION 指令后, 给主机发送一个空应答

以上的 USB 枚举过程是根据现有特定设备捕获得到, 并不完整涉及到 USB 协议的所有指令包, 如 GET_STATUS 指令包、CLEAR_FEATURE 指令包、SET_FEATURE 指令包、SET_DESCRIPTOR 指令包、GET_CONFIGURATION 指令包、GET_INTERFACE 指令包、SET_INTERFACE 指令包、SYNCH_FRAME 指令包等。

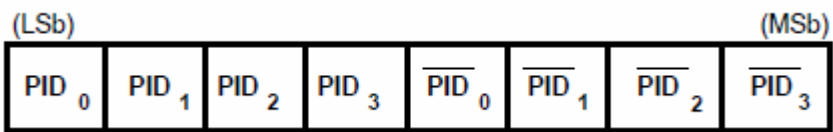
图解 USB 协议之二 数据包分析

感谢朋友们对我发的上一个帖子《图解 USB 协议之一 枚举过程》的支持，虽然手头的项目比较忙，周末还是抽时间写了这个专题的第二篇，不能让大家失望啊，哈！

本篇分析数据包，仍然使用“HD-USB12”USB 协议分析仪采集分析数据。分两部分内容：首先和童鞋们一起简单复习一下 USB 协议中的相关知识，然后用“HD-USB12”USB 协议分析仪采集几个实例进行分析。

一、协议知识

1) PID 格式，详见图 1



(图 1)

注：

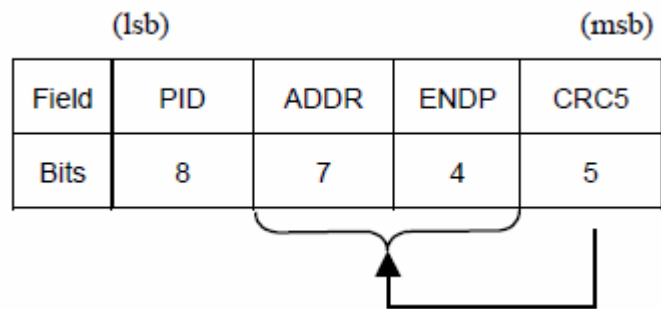
- a、上图是从低位到高位
- b、高四位是低四位取反
- c、PID 自校验

2) PID 分类，详见图 2

PID 类型	PID 名称	数值（由高位到低位）		备注
		十六进制	二进制	
Token	OUT	E1	11100001	
	IN	69	1101001	
	SOF	A5	10100101	
	SETUP	2D	00101101	
Data	DATA0	C3	11000011	
	DATA1	4B	01001011	
	DATA2	87	10000111	仅用于高速 USB
	MDATA	0F	00001111	仅用于高速 USB
Handshake	ACK	D2	11010010	
	NAK	5A	01011010	
	STALL	1E	00011110	
	NYET	96	10010110	
Special	PRE	3C	00111100	
	ERR	3C	00111100	
	SPLIT	78	01111000	仅用于高速 USB
	PING	B4	10110100	仅用于高速 USB

(图 2)

3) Token 包格式 (不包括 SOF), 详见图 3

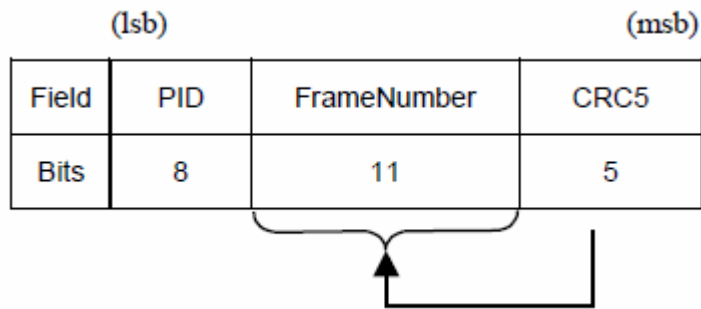


(图 3)

注:

- a、ADDR 表示设备地址
- b、ENDP 表示端点号
- c、CRC5 表示设备地址及端点号的校验和

4) SOF 包格式, 详见图 4

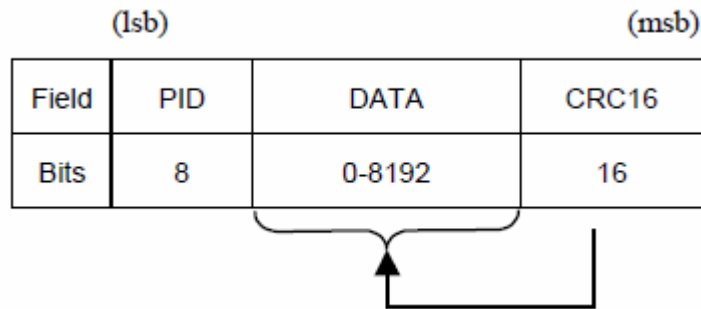


(图 4)

注:

- a、FrameNumber 表示帧号, 1 毫秒发送一帧
- b、CRC5 表示 FrameNumber 的校验和

5) Data 包格式, 详见图 5

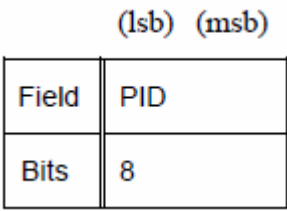


(图 5)

注:

- a、DATA 表示要传输的数据, 全速 USB 的最大长度为 1023Bytes, 高速 USB 的最大长度为 1024Bytes
- b、CRC16 表示 DATA 的校验和

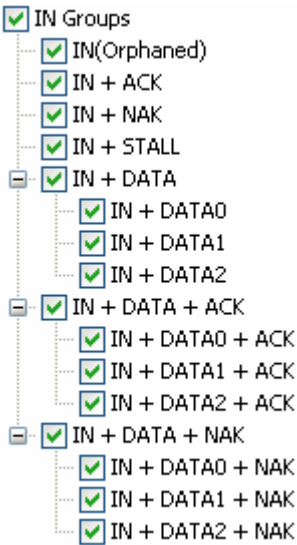
6) Handshake 包格式, 详见图 6



(图 6)

二、USB 传输包组（事务）实例分析

1) IN 包组分类（懒得自己总结，直接从 HD-USB12 的数据包过滤功能截取），详见图 7



(图 7)

2) IN 包组分析（通过 HD-USB12 监控软件仅过滤出 IN 包组数据），详见图 8

Index	ms.us	Dur	Len	Err	PID	CRC	Packet
691	00:21.197.117	457.98 us					[62 IN/NAK]
692	00:21.197.577	2.75 us			IN	1D	DEV = 00 EP = 80
693	00:21.197.581	1.42 us			STALL		
698	00:21.198.172	902.31 us					[123 IN/NAK]
700	00:21.199.104	224.08 us					[30 IN/NAK]
701	00:21.199.331	2.75 us			IN	1D	DEV = 00 EP = 80
702	00:21.199.334	13.69 us	16		DATA1	6FEE	06 00 FF 09 01 A1 02 15 00 26 FF 00 75 08 95 01
703	00:21.199.348	1.42 us			ACK		

(图 8)

- a、Index[691]: 表示主机向设备发送 IN 指令取数据，设备回应没数据；
- b、Index[692 - 693]: 表示主机向设备地址 0（DEV = 00）端点 0（EP = 80，最高位表示传输方向）发送 IN 指令取数据，设备回应端点停止（STALL）；
- c、Index[701 - 703]: 表示主机向设备地址 0（DEV = 00）端点 0（EP = 80，最高位表示传输方向）发送 IN 指令取数据，设备收到指令后向主机发送数据 Index[702]，主机正确收到数据后回应 ACK

欢迎访问avatar的电子博客: <http://blog.ednchina.com/avatar/>
Avatar 的邮箱: avatar_1210@tom.com

- d、Index[701]的内容, 详见图 9
- e、Index[702]的内容, 详见图 10
- f、Index[702]的数据发送时序, 详见图 11 (只截取了部分, 时序做的很漂亮, 看着也很清晰), 注意时序图的 Offset[0003]其传输的数据为 FF, 根据 USB 采用的传输编码 (NRZI) 前 6 位为 1 后插入 1 位 0, 因些第 7 位的传输要传输 2 位 (01), 其时间也占用两位的时间。(啊啊! 有时序功能就是清楚, 时序还有其它用途以后分解)

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ASCII
00	69	01	E8														i..

(图 9)

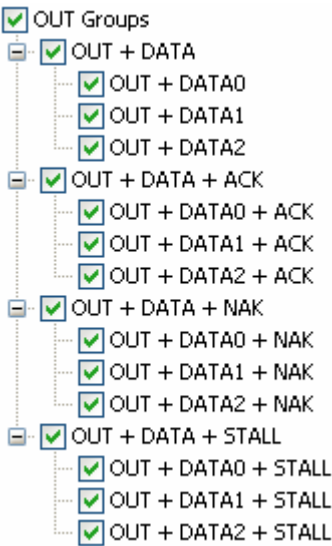
Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ASCII
00	4B	06	00	FF	09	01	A1	02	15	00	26	FF	00	75	08	95	K.....&.u..
10	01	EE	6F														...o

(图 10)

Offset	Time	Value	Timing(ns): [b0...b7]
			Timestamp = 00:21.199.334.466 Duration = 13.687 us
0000	687	4B	
0001	1354	06	
0002	2020	00	
0003	2687	FF	
0004	3437	09	

(图 11)

3) OUT 包组分类, 详见图 12 (直接从 HD-USB12 的数据包过滤功能截取)



(图 12)

4) OUT 包组分析 (通过 HD-USB12 监控软件仅过滤出 OUT 包组数据), 详见图 13

欢迎访问avatar的电子博客: <http://blog.ednchina.com/avatar/>
Avatar 的邮箱: avatar_1210@tom.com

Index	m.s.ms.us	Dur	Len	Err	PID	CRC	Packet
16	00:17.397.734	2.75 us			OUT	02	DEV = 00 EP = 00
17	00:17.397.737	2.75 us	0		DATA1		
18	00:17.397.740	1.42 us			ACK		
44	00:17.481.988	2.75 us			OUT	1D	DEV = 01 EP = 00
45	00:17.481.991	2.75 us	0		DATA1		
46	00:17.481.994	1.42 us			ACK		

(图 13)

- a、Index[16 - 18]: 表示主机向设备地址 0 (DEV = 00) 端点 0 (EP = 00, 最高位表示传输方向) 发送 Out 指令后, 发送数据 Index[17] (主机发出空回应所以 Index[17]没有数据), 设备收到数据后回应 ACK;
- b、Index[16]的内容, 详见图 14
- c、Index[17]的内容, 详见图 15

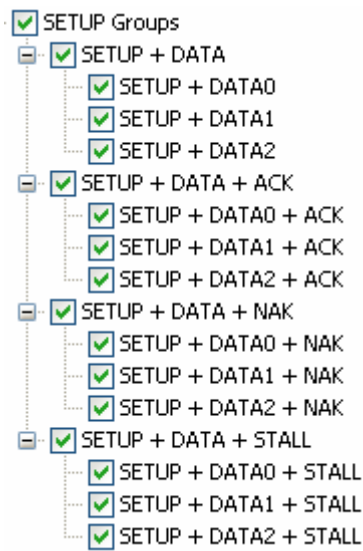
Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ASCII
00	E1	00	10														...

(图 14)

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ASCII
00	4B	00	00														K..

(图 15)

5) SETUP 包组分类, 详见图 16 (直接从 HD-USB12 的数据包过滤功能截取)



(图 16)

6) SETUP 包组分析 (通过 HD-USB12 监控软件仅过滤出 SETUP 包组数据), 详见图 17

欢迎访问avatar的电子博客: <http://blog.ednchina.com/avatar/>
Avatar 的邮箱: avatar_1210@tom.com

Index	m:s.ms.us	Dur	Len	Err	PID	CRC	Packet
23	00:17.435.629	2.75 us			SETUP	02	DEV = 00 EP = 00
24	00:17.435.632	8.08 us	8		DATA0	25EB	SET_ADDRESS
25	00:17.435.641	1.42 us			ACK		
31	00:17.480.572	2.75 us			SETUP	1D	DEV = 01 EP = 00
32	00:17.480.575	8.08 us	8		DATA0	F4E0	GET_DESCRIPTOR
33	00:17.480.584	1.42 us			ACK		

(图 17)

- a、Index[23 - 25]: 表示主机向设备地址 0 (DEV = 00) 端点 0 (EP = 00, 最高位表示传输方向) 发送 SETUP 指令后, 发送数据 Index[24], 设备收到数据后回应 ACK;
- b、Index[23]的内容, 详见图 18
- c、Index[24]的内容, 详见图 19

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ASCII
00	2D	00	10														-. .

(图 18)

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ASCII
00	C3	00	05	01	00	00	00	00	00	EB	25					%

(图 19)

以上数据包分析是根据现有特定设备捕获得到, 并不完整涉及到所有事务包组, 如 IN+DATA+NAK、OUT+DATA+NAK、SETUP+DATA+NAK 等等。

图解 USB 协议之三 USB 标准描述符

最近忙于公司的项目和本人的驾驶证考试，拖延了这个系列第三篇的完成，先向各位支持我的 DX 表示一下歉意。不过好在驾驶证的蝴蝶桩考试顺利通过，



自己先庆贺一下，嘿嘿。

最近做一个手持机的项目，第一次真正感受到了手头这个小设备的功力。事情是这样的：要将 n 年前写的基于 51 的固件代码移植到 AVR，问题出现了....，枚举不成功！立刻想到用协议分析仪看看.....（此处省略 168 字），结果当然是一目了然，几分钟轻松搞定！关于这次的经历，回头专门写一个帖子吧，先卖个关子，哈。

闲话少说，本篇图解 USB 标准描述符。

数据采集设备：HD-USB12 USB 协议分析仪。

还是先复习一下 USB 协议的相关内容。USB 通讯配置是通过描述符完成的，下面对 USB 标准描述符进行说明。

一、USB 标准描述符

1、标准设备描述符

偏移量	域	大小	值	描述
0	bLength	1	数字	描述符的大小=12H
1	bDescriptorType	1	常量	设备描述符类型=01H
2	bcdUSB	2	BCD 码	此设备与描述符兼容的 USB 设备说明版本号(BCD 码)
4	bDeviceClass	1	类	设备类码 如果此域的值为 0 则一个设置下每个接口指出它自己的类，并个接口各自独立工作。 如果此域的值处于 1~FEH 之间，则设备在不同的接口上支持不同的类。并这些接口可能不能独立工作。此值指出了，这些接口集体的类定义。 如果此域设为 FFH，则此设备的类由厂商定义。

欢迎访问avatar的电子博客：EET站点：http://forum.eet-cn.com/BLOG_zhou36_301.HTM

EDN站点：<http://blog.ednchina.com/avatar/>

Avatar 的邮箱：avatar_1210@tom.com

5	bDeviceSubClass	1	子类	子类码 这些码值的具体含义根据 bDeviceClass 域来看。 如 bDeviceClass 域为零，此域也须为零 如 bDeviceClass 域为 FFH，此域的所有值保留。
6	bDevicePortocol	1	协议	协议码 这些码的值视 bDeviceClass 和 bDeviceSubClass 的值而定。 如果设备支持设备基础上的类相关的协议，此码标志了设备类说明上的值。 如果此域的值为零，则此设备不在设备基础上支持设备类相关的协议。然而，它可能在接口基础上支持设备类相关的协议。 如果此域的值为 FFH，此设备使用厂商定义的协议。
7	bMaxPacketSize0	1	数字	端点 0 的最大包大小(仅 8,16,32,64 为合法值)
8	idVendor	2	ID	厂商标志（由 USB 标准付值）
10	idProduct	2	ID	产品标志（由厂商付值）
12	bcdDevice	2	BCD 码	设备发行号（BCD 码）
14	iManufacturer	1	索引	描述厂商信息的字串的索引。
15	iProduct	1	索引	描述产品信息的字串的索引。
16	iSerialNumber	1	索引	描述设备序列号信息的字串的索引。
17	bNumConfigurations	1	数字	可能的设置数

2、标准配置描述符

偏移量	域	大小	值	描述
0	bLength	1	数字	描述符的大小=09H
1	bDescriptorType	1	常量	配置描述符类型=02H
2	wTotalLength	2	数字	此配置信息的总长（包括配置，接口，端点和设备类及厂商定义的描述符）
4	bNumInterfaces	1	数字	此配置所支持的接口个数
5	bConfigurationValue	1	数字	在 SetConfiguration（）请求中用作参数来选定此配置。
6	iConfiguration	1	索引	描述此配置的字串描述符索引
7	bmAttributes	1	位图	配置特性： D7： 保留（设为一） D6： 自给电源 D5： 远程唤醒 D4..0： 保留（设为一）

欢迎访问avatar的电子博客：EET站点：http://forum.eet-cn.com/BLOG_zhou36_301.HTM

EDN站点：<http://blog.ednchina.com/avatar/>

Avatar 的邮箱：avatar_1210@tom.com

				一个既用总线电源又有自给电源的设备会在 MaxPower 域指出需要从总线取的电量。并设置 D6 为一。运行时期的实际电源模式可由 GetStatus(DEVICE) 请求得到。
8	MaxPower	1	mA	在此配置下的总线电源耗电量。以 2mA 为一个单位。

3、标准接口描述符

偏移量	域	大小	值	说明
0	bLength	1	数字	描述符的大小=09H
1	bDescriptorType	1	常量	接口描述符类型=04H
2	bInterfaceNumber	1	数字	接口号，当前配置支持的接口数组索引（从零开始）
3	bAlternateSetting	1	数字	可选设置的索引值。
4	bNumEndpoints	1	数字	此接口用的端点数量，如果是零则说明此接口只用缺省控制管道。
5	bInterfaceClass	1	类	类值 零值为将来的标准保留。 如果此域的值设为 FFH，则此接口类由厂商说明。 所有其它的值由 USB 说明保留。
6	bInterfaceSubClass	1	子类	子类码 这些值的定义视 bInterfaceClass 域而定。 如果 bInterfaceClass 域的值为零则此域的值必须为零。 bInterfaceClass 域不为 FFH 则所有值由 USB 所保留。
7	bInterfaceProtocol	1	协议	协议码：bInterfaceClass 和 bInterfaceSubClass 域的值而定。如果一个接口支持设备类相关的请求此域的值指出了设备类说明中所定义的协议。
8	iInterface	1	索引	描述此接口的字符串描述符的索引值。

4、标准端点描述符

偏移量	域	大小	值	说明
0	bLength	1	数字	描述符的大小=07H
1	bDescriptorType	1	常量	端点描述符类型=05H

欢迎访问avatar的电子博客：EET站点：http://forum.eet-cn.com/BLOG_zhou36_301.HTM

EDN站点：<http://blog.ednchina.com/avatar/>

Avatar 的邮箱：avatar_1210@tom.com

2	bEndpointAddress	1	端点	<p>此描述符所描述的端点的地址。此地址的编码如下：</p> <p>Bit 3..0 : 端点号。</p> <p>Bit 6..4 : 保留, 为零</p> <p>Bit 7: 方向, 如果控制端点则略。</p> <p>0: 出端点</p> <p>1: 入端点</p>
3	bmAttributes	1	位图	<p>此域的值描述的是在 bConfigurationValue 域所指的配置下端点的特性。</p> <p>Bit 1..0 : 传送类型</p> <p>00=控制传送</p> <p>01=同步传送</p> <p>10=批传送</p> <p>11=中断传送</p> <p>所有其它的位都保留。</p>
4	wMaxPacketSize	2	数字	<p>当前配置下此端点能够接收或发送的最大数据包的大小。</p> <p>对与同步传送此值用于为每帧的数据净负荷预留时间。而通道可能在实际运行时不需要预留的带宽。实际带宽可由设备通过一种非 USB 定义的机制汇报给主机。</p> <p>对于中断传送, 批传送, 控制传送, 端点可能发送较小的数据包。并且在结束传送后既有可能间隙时间来重启, 也有可能不需要这段时间。具体请参照第五章。</p>
6	bInterval	1	数字	<p>轮寻数据传送端点的时间间隙。</p> <p>此域的值对于批传送的端点及控制传送的端点忽略。对于同步传送的端点此域必需为 1。对于中断传送的端点此域值的范围为 1 到 255。</p>

5、语言描述符

偏移量	域	大小	值	描述
0	bLength	1	N+2	描述符的大小
1	bDescriptorType	1	常量	字串描述符类型=03H
2	wLANGID[0]	2	数字	语言标识 (LANGID) 码 0
...
N	wLANGID[x]	2	数字	语言标识 (LANGID) 码 X

6、UNICODE 字串描述符

偏移量	域	大小	值	描述
0	bLength	1	数字	描述符的大小
1	bDescriptorType	1	常量	字符串描述符类型=03H
2	bString	N	数字	UNICODE 编码的字符串

二、图解枚举过程中的描述符

1、枚举成功后主机得到的第一个描述符（设备描述符），见图 1

Index	m:s.ms.us	Dur	Len	Err	PID	CRC	Packet
32	00:03.738.227	2.75 us			SETUP	1D	DEV = 01 EP = 00
33	00:03.738.230	8.08 us	8		DATA0	F4E0	GET_DESCRIPTOR
34	00:03.738.239	1.42 us			ACK		
35	00:03.738.243	797.77 us					[109 IN/NAK]
36	00:03.739.069	2.75 us					[1 SOF]
37	00:03.739.072	297.06 us					[40 IN/NAK]
38	00:03.739.372	2.75 us			IN	1D	DEV = 00 EP = 80
39	00:03.739.375	13.42 us	16		DATA1	1BB8	12 01 00 01 00 00 00 10 14 1B 66 06 00 01 00 00
40	00:03.739.389	1.42 us			ACK		
41	00:03.739.391	239.44 us					[33 IN/NAK]
42	00:03.739.633	2.75 us			IN	1D	DEV = 00 EP = 80
43	00:03.739.637	4.17 us	2		DATA0	8F3F	00 01
44	00:03.739.641	1.42 us			ACK		
45	00:03.739.644	2.75 us			OUT	1D	DEV = 01 EP = 00
46	00:03.739.647	2.75 us	0		DATA1		
47	00:03.739.650	1.42 us			ACK		

(图 1) (本数据由 HD-USB12 USB 协议分析仪采集)

- a、Index[38 - 44]：设备发送到主机的描述，详见图 2。
- b、图 2 为设备描述符，其各字段含义参考上面标准设备描述符。

bLength = 0x12
bDescriptorType = 0x01
DEVICE
bcdUSB = 0x0100
bDeviceClass = 0x00
bDeviceSubClass = 0x00
bDeviceProtocol = 0x00
bMaxPacketSize0 = 16
idVendor = 0x1B14
idProduct = 0x0666
bcdDevice = 0x0100
iManufacturer = 0
iProduct = 0
iSerialNumber = 0
bNumConfigurations = 1

(图 2) (本数据由 HD-USB12 USB 协议分析仪采集)

2、枚举成功后主机得到的第二个描述符（配置描述符），见图 3

Index	Time (ms.us)	Dur	Len	Err	PID	CRC	Packet
49	00:03.740.095	2.73 us			SETUP	1D	DEV = 01 EP = 00
50	00:03.740.098	8.06 us	8		DATA0	04AE	GET_DESCRIPTOR
51	00:03.740.106	1.44 us			ACK		
52	00:03.740.111	909.10 us					[124 IN/NAK]
53	00:03.741.051	2.75 us					[1 SOF]
54	00:03.741.054	2.75 us			IN	1D	DEV = 01 EP = 80
55	00:03.741.057	8.75 us	9		DATA1	B873	09 02 29 00 01 01 00 60 01
56	00:03.741.067	1.42 us			ACK		
57	00:03.741.069	2.75 us			OUT	1D	DEV = 01 EP = 00
58	00:03.741.072	2.75 us	0		DATA1		
59	00:03.741.075	1.42 us			ACK		

(图 3) (本数据由 HD-USB12 USB 协议分析仪采集)

- a、Index[54 - 56]：设备发送到主机的描述，详见图 4。
- b、图 4 为配置描述符，其各字段含义参考上面标准配置描述符。

```
bLength = 0x09
bDescriptorType = 0x02
CONFIGURATION
wTotalLength = 41
bNumInterfaces = 1
bConfigurationValue = 1
iConfiguration = 0
bmAttributes = 0x60
D7 : Reserved (set to one)
0
D6 : Self-powered
1
D5 : Remote Wakeup
1
D4...0 : Reserved (reset to zero)
00000
bMaxPower = 2mA
```

(图 4) (本数据由 HD-USB12 USB 协议分析仪采集)

3、随后为配置描述符、接口描述符及端点描述符见图 5

欢迎访问avatar的电子博客：EET站点：http://forum.eet-cn.com/BLOG_zhou36_301.HTM
EDN站点：<http://blog.ednchina.com/avatar/>

Avatar 的邮箱：avatar_1210@tom.com

Index	m:s.ms.us	Dur	Len	Err	PID	CRC	Packet
61	00:03.742.180	2.75 us			SETUP	1D	DEV = 01 EP = 00
62	00:03.742.183	8.17 us	8		DATA0	A4F9	GET_DESCRIPTOR
63	00:03.742.192	1.42 us			ACK		
64	00:03.742.196	814.02 us					[111 IN/NAK]
65	00:03.743.041	2.75 us					[1 SOF]
66	00:03.743.044	281.56 us					[38 IN/NAK]
67	00:03.743.328	2.75 us			IN	1D	DEV = 00 EP = 80
68	00:03.743.331	13.42 us	16		DATA1	F484	09 02 29 00 01 01 00 60 01 09 04 00 00 02 03 00
69	00:03.743.345	1.42 us			ACK		
70	00:03.743.347	654.35 us					[90 IN/NAK]
71	00:03.744.030	2.75 us					[1 SOF]
72	00:03.744.033	2.75 us			IN	1D	DEV = 01 EP = 80
73	00:03.744.036	13.42 us	16		DATA0	6F22	00 00 09 21 01 01 00 01 22 25 00 07 05 82 03 40
74	00:03.744.050	1.42 us			ACK		
75	00:03.744.052	470.02 us					[64 IN/NAK]
76	00:03.744.525	2.75 us			IN	1D	DEV = 00 EP = 80
77	00:03.744.528	8.75 us	9		DATA1	B2FB	00 0A 07 05 02 03 40 00 0A
78	00:03.744.537	1.42 us			ACK		
79	00:03.744.540	2.75 us			OUT	1D	DEV = 01 EP = 00
80	00:03.744.543	2.75 us	0		DATA1		
81	00:03.744.546	1.42 us			ACK		

(图 5) (本数据由 HD-USB12 USB 协议分析仪采集)

- a、Index[67 - 78]：设备发送到主机的描述，详见图 6、图 7、图 8 及图 9。
- b、图 6 为配置描述符，其各字段含义参考上面标准配置描述符。

```
bLength = 0x09
bDescriptorType = 0x02
CONFIGURATION
wTotalLength = 41
bNumInterfaces = 1
bConfigurationValue = 1
iConfiguration = 0
bmAttributes = 0x80
D7 : Reserved (set to one)
0
D6 : Self-powered
1
D5 : Remote Wakeup
1
D4...0 : Reserved (reset to zero)
00000
bMaxPower = 2mA
```

(图 6) (本数据由 HD-USB12 USB 协议分析仪采集)

- c、图 7 为接口描述符，其各字段含义参考上面标准接口描述符。

欢迎访问avatar的电子博客：EET站点：http://forum.eet-cn.com/BLOG_zhou36_301.HTM

EDN站点：<http://blog.ednchina.com/avatar/>

Avatar 的邮箱：avatar_1210@tom.com

```
bLength = 0x09
bDescriptorType = 0x04
    INTERFACE
bInterfaceNumber = 0
bAlternateSetting = 0
bNumEndpoints = 2
bInterfaceClass = 0x03
bInterfaceSubClass = 0x00
bInterfaceProtocol = 0x00
iInterface = 0
```

(图 7) (本数据由 HD-USB12 USB 协议分析仪采集)

d、图 8 及图 9 为端点描述符，其各字段含义参考上面标准端点描述符。

```
bLength = 0x07
bDescriptorType = 0x05
    ENDPOINT
bEndpointAddress = 0x82
    D7 : Direction
        1 = IN endpoint
    D6...4 : Reserved
        000
    D3...0 : The endpoint number
        2
bmAttributes = 0x03
    D7...6 : Reserved
        00
    D5...4 : Usage Type
        00 = Data endpoint
    D3...2 : Synchronization Type
        00 = No Synchronization
    D1...0 : Transfer Type
        11 = Interrupt
wMaxPacketSize = 64
bInterval = 10
```

(图 8) (本数据由 HD-USB12 USB 协议分析仪采集)

```
bLength = 0x07
bDescriptorType = 0x05
    ENDPOINT
bEndpointAddress = 0x02
    D7 : Direction
        0 = OUT endpoint
    D6...4 : Reserved
        000
    D3...0 : The endpoint number
        2
bmAttributes = 0x03
    D7...6 : Reserved
        00
    D5...4 : Usage Type
        00 = Data endpoint
    D3...2 : Synchronization Type
        00 = No Synchronization
    D1...0 : Transfer Type
        11 = Interrupt
wMaxPacketSize = 64
bInterval = 10
```

欢迎访问**avatar**的电子博客：EET站点：http://forum.eet-cn.com/BLOG_zhou36_301.HTM

EDN站点：<http://blog.ednchina.com/avatar/>

Avatar 的邮箱：**avatar_1210@tom.com**

（图 9）（本数据由 HD-USB12 USB 协议分析仪采集）

以上描述符不包括高速 USB 的描述符，如速度配置描述符等，详细请参考 USB 标准。

图解 USB 协议之四 USB 枚举失败常见原因分析

本篇基于固件代码的开发及移植过程中可能引起的 USB 枚举失败的原因进行分析。

数据采集设备：“HD-USB12” 全速/低速 USB2.0 协议分析仪

一、如图 1 所示的 USB 枚举失败

- 1) Index[666 - 668]: 表示主机向默认地址发送 GET_DESCRIPTOR 指令包。
- 2) Index[670 - 671]: 表示设备默认地址端点不可用。
- 3) 由上我们可以推断出固件代码对 GET_DESCRIPTOR 指令包无响应。

Index	m:s.ms.us	Dur	Len	Err	PID	CRC	Packet
663	00:07.516.127						<Reset> / <Target Disconnected>
664	00:07.516.779						<Unreset> / <Target Connected>
665	00:07.517.778	49.01 ms					[50 SOF]
666	00:07.566.790	2.75 us			SETUP	02	DEV = 00 EP = 00
667	00:07.566.793	8.08 us	8		DATA0	94DD	GET_DESCRIPTOR
668	00:07.566.801	1.42 us			ACK		
669	00:07.566.806	173.04 us					[23 IN/NAK]
670	00:07.566.981	2.75 us			IN	02	DEV = 00 EP = 80
671	00:07.566.985	1.42 us			STALL		

(图 1) (本数据由 HD-USB12 USB 协议分析仪采集)

二、如图 2 所示的 USB 枚举失败

- 1) Index[697 - 699]: 表示主机向默认地址发送 SET_ADDRESS 指令包。
- 2) Index[701 - 703]: 表示设备完成 SET_ADDRESS 指令后，给主机发送一个空应答。
- 3) Index[705 - 710]: 表示主机向地址 DEC=01 重复发送 SET_DESCRIPTOR 指令包，但设备无任何应答，因此该事务包设备并没有接收到。
- 4) 由上我们可以推断出固件中设置地址的代码执行失败。

欢迎访问avatar的电子博客：EET站点：http://forum.eet-cn.com/BLOG_zhou36_301.HTM
EDN站点：<http://blog.ednchina.com/avatar/>

Avatar 的邮箱：avatar_1210@tom.com

Index	Time (ms.us)	Dur	Len	Err	PID	CRC	Packet
694	00:08.572.258						<Reset> / <Target Disconnected>
695	00:08.573.208						<Unreset> / <Target Connected>
696	00:08.574.207	24.01 ms					[25 SOF]
697	00:08.598.831	2.75 us			SETUP	02	DEV = 00 EP = 00
698	00:08.598.835	8.08 us	8		DATA0	25EB	SET_ADDRESS
699	00:08.598.843	1.42 us			ACK		
700	00:08.598.847	189.25 us					[26 IN/NAK]
701	00:08.599.039	2.75 us			IN	02	DEV = 00 EP = 80
702	00:08.599.043	2.75 us	0		DATA1		
703	00:08.599.046	1.42 us			ACK		
704	00:08.599.209	44.01 ms					[45 SOF]
705	00:08.643.757	2.75 us			SETUP	1D	DEV = 01 EP = 00
706	00:08.643.760	8.08 us	8		DATA0	F4E0	GET_DESCRIPTOR
707	00:08.643.773	2.75 us			SETUP	1D	DEV = 01 EP = 00
708	00:08.643.776	8.08 us	8		DATA0	F4E0	GET_DESCRIPTOR
709	00:08.643.789	2.75 us			SETUP	1D	DEV = 01 EP = 00
710	00:08.643.792	8.06 us	8		DATA0	F4E0	GET_DESCRIPTOR
711	00:08.644.216	514.09 ms					[515 SOF]
712	00:09.168.551						<Reset> / <Target Disconnected>

(图 2) (本数据由 HD-USB12 USB 协议分析仪采集)

三、如图 3 所示的 USB 枚举失败(图 3 为过滤掉 IN + NAK 及 SOF 包后的数据)

- 1) Index[694 - 696]: 表示主机向默认地址发送 SET_ADDRESS 指令包。但设备对该指令答，无应答。
- 2) 由上我们可以推断出固件代码对 SET_ADDRESS 指令包无响应或响应没发送空应答。

欢迎访问avatar的电子博客：EET站点：http://forum.eet-cn.com/BLOG_zhou36_301.HTM

EDN站点：<http://blog.ednchina.com/avatar/>

Avatar 的邮箱：avatar_1210@tom.com

Index	m.s.ms.us	Dur	Len	Err	PID	CRC	Packet
691	00:10.069.484						<Reset> / <Target Disconnected>
692	00:10.070.431						<Unreset> / <Target Connected>
694	00:10.096.100	2.75 us			SETUP	02	DEV = 00 EP = 00
695	00:10.096.103	8.06 us	8		DATA0	25EB	SET_ADDRESS
696	00:10.096.112	1.42 us			ACK		
11400	00:15.950.761						<Reset> / <Target Disconnected>
11401	00:15.951.567						<Unreset> / <Target Connected>
11403	00:15.994.460	2.75 us			SETUP	02	DEV = 00 EP = 00
11404	00:15.994.463	8.08 us	8		DATA0	94DD	GET_DESCRIPTOR
11405	00:15.994.472	1.42 us			ACK		
11409	00:15.994.759	2.75 us			IN	02	DEV = 00 EP = 80
11410	00:15.994.762	13.42 us	16		DATA1	1BB8	12 01 00 01 00 00 00 10 14 1B 66 06 00 01 00 00
11411	00:15.994.776	1.42 us			ACK		
11412	00:15.994.779	2.75 us			OUT	02	DEV = 00 EP = 00
11413	00:15.994.782	2.75 us	0		DATA1		
11414	00:15.994.785	1.42 us			ACK		
11416	00:16.005.625						<Reset> / <Target Disconnected>
11417	00:16.006.574						<Unreset> / <Target Connected>
11419	00:16.098.969	2.75 us			SETUP	02	DEV = 00 EP = 00
11420	00:16.098.972	8.08 us	8		DATA0	25EB	SET_ADDRESS
11421	00:16.098.981	1.42 us			ACK		
22022	00:21.893.552						<Reset> / <Target Disconnected>
22023	00:21.893.567						<Unreset> / <Target Connected>

(图 3) (本数据由 HD-USB12 USB 协议分析仪采集)

四、设备发送给主机的描述信息不完整，如少发送接口或端点信息等导致枚举失败。这要求分析所有采集到的数据才能发现问题。由于数据量较大，就不用图片说明了。

以上是根据本人开发过程中曾经碰到的问题进行的总结分析，在实际情况中枚举失败的原因很多，不可能一一列出，但只要有一台总线协议分析设备能抓取数据进行分析是很容易找到原因的。