
Ford Motor (China) Company

SDL 完整集成 usbmuxd 协议说明文档

Version <1.1>

免责声明

本文档中的内容仅供参考。福特汽车中国对本服务内容的错误或遗漏概不负责。在任何情况下，福特汽车中国均不对因使用本文档而产生或与之相关的任何特殊，直接，间接，间接或偶然的损害赔偿或任何损害负责，无论是在合同，疏忽，其他侵权行为中服务或服务的内容。福特汽车中国保留随时对本文档内容进行补充，删除或修改的权利，恕不另行通知。

Disclaimer

The contents contained in this document are for general information purposes only. Ford Motor China assumes no responsibility for errors or omissions in the contents on the Service.

In no event shall Ford Motor China be liable for any special, direct, indirect, consequential, or incidental damages or any damages whatsoever, whether in an action of contract, negligence or other tort, arising out of or in connection with the use of the Document or the contents of the Document.

Ford Motor China reserves the right to make additions, deletions, or modification to the contents on the Service at any time without prior notice.



本作品采用[知识共享署名-相同方式共享 4.0 国际许可协议](https://creativecommons.org/licenses/by-sa/4.0/)进行许可。

This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).

编制	Beyondsoft	日期	2017-11-24	版权	署名-相同方式共享 4.0 国际
审核	Ford	日期	2017-11-24	管理	Ford

Project Name: usbmuxd	Version: <1.0.0>
SDL 完整集成 usbmuxd 协议说明文档	Date: <24/11/2017>
<document identifier>	

修改历史

版本	日期	说明
1.0.0	2017-11-24	初版，完成文档架构，填写文档内容

Project Name: usbmuxd	Version: <1.0.0>
SDL 完整集成 usbmuxd 协议说明文档	Date: <24/11/2017>
<document identifier>	

目录

1 引言.....	2
1.1 背景.....	2
1.2 内容.....	2
1.3 适用范围.....	2
1.4 术语.....	2
1.5 参考资料.....	2
2 框架结构.....	3
2.1 网络拓扑图.....	3
2.2 sdl_core 和手机 app 关系包图.....	3
3 SDL_CORE USBMUXD 模块类图.....	3
4 时序图.....	4
4.1 sdl_core usbmuxd 模块初始化、连接及收发数据时序图.....	4
5 USBMUXD 协议 SDL_CORE 端接口函数.....	5
5.1 int usbmuxd_get_device_list().....	5
5.2 int usbmuxd_device_list_free().....	5
5.3 int usbmuxd_connect().....	6
5.4 int usbmuxd_disconnect().....	6
6 USBMUXD 协议 IPHONE 端接口函数.....	7
6.1 (void)ioFrameChannel.....	7
6.2 (BOOL)ioFrameChannel.....	7
6.3 (void)ioFrameChannel.....	8
6.4 (void)ioFrameChannel.....	8
6.5 (void)sendFrameOfType.....	8
6.6 (void)readFramesOverChannel.....	9
6.7 (void)readPayloadOfSize.....	9
6.8 (void)readAndDiscardDataOfSize.....	10
7 USBMUXD 协议帧格式说明.....	10



Project Name: usbmuxd	Version: <1.0.0>
SDL 完整集成 usbmuxd 协议说明文档	Date: <24/11/2017>
<document identifier>	

1 引言

1.1 背景

OSX 系统上，苹果有一个服务，叫 usbmuxd，这个服务主要用于在 USB 协议上实现多路 TCP 连接，将 USB 通信抽象为 TCP 通信，苹果的 iTunes，XCode，都直接或者间接地用到了这个服务，libimobiledevice 已经将苹果的 usbmuxd 服务实现跨平台并且开源了。只要在 PC 上提供 usbmuxd 服务，就可以让 iDevice 通过 TCP 和 PC 端的 sdl_core 通信。本文的讲的是 sdl_core 和 Iphone 如何集成 usbmuxd 协议，以方便开发人员快速的使用。

1.2 内容

本文档描述的主要内容包括：usbmuxd 协议接口函数，usbmuxd 协议帧格式说明以及 usbmuxd 的集成说明。

1.3 适用范围

本文档主要面向开发人员，指导开发人员如何去应用 usbmuxd 协议，提高开发效率，减少重复工作。

1.4 术语

缩写	含义
HMI	人机交互界面
sdl_core	SmartDeviceLink

1.5 参考资料

<https://github.com/libimobiledevice/libimobiledevice>

<https://github.com/libimobiledevice/libusbmuxd>

<https://github.com/libimobiledevice/libplist>

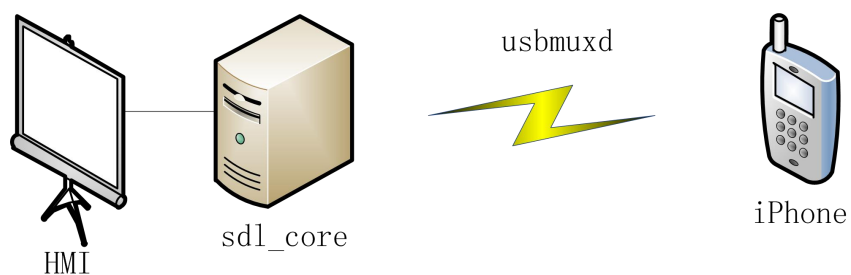


本作品采用[知识共享署名-相同方式共享 4.0 国际许可协议](https://creativecommons.org/licenses/by-sa/4.0/)进行许可

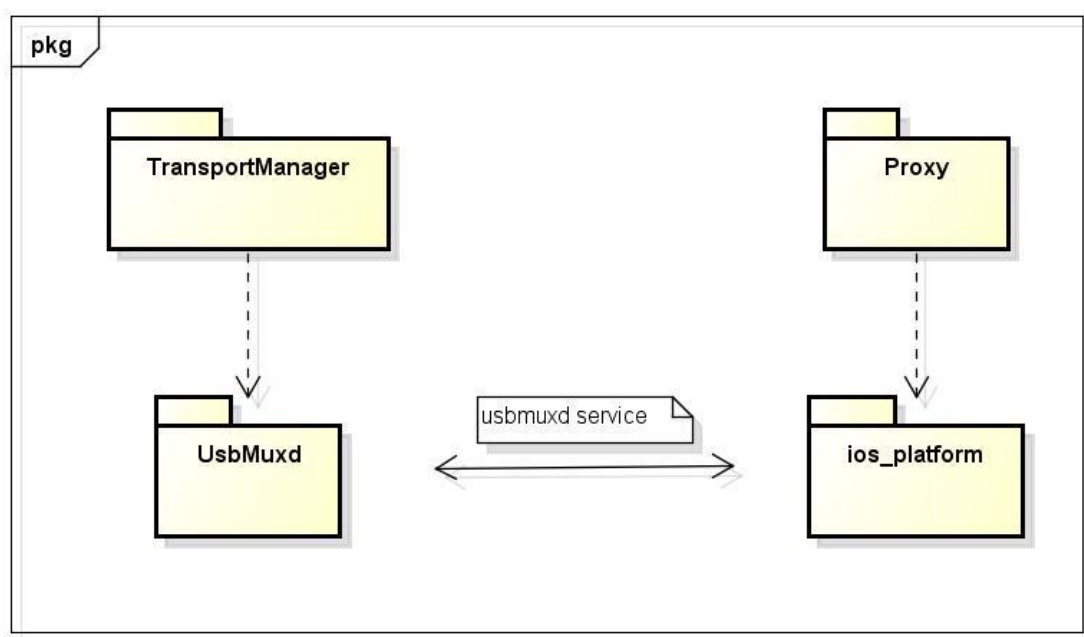
Project Name: usbmuxd	Version: <1.0.0>
SDL 完整集成 usbmuxd 协议说明文档	Date: <24/11/2017>
<document identifier>	

2 框架结构

2.1 网络拓扑图



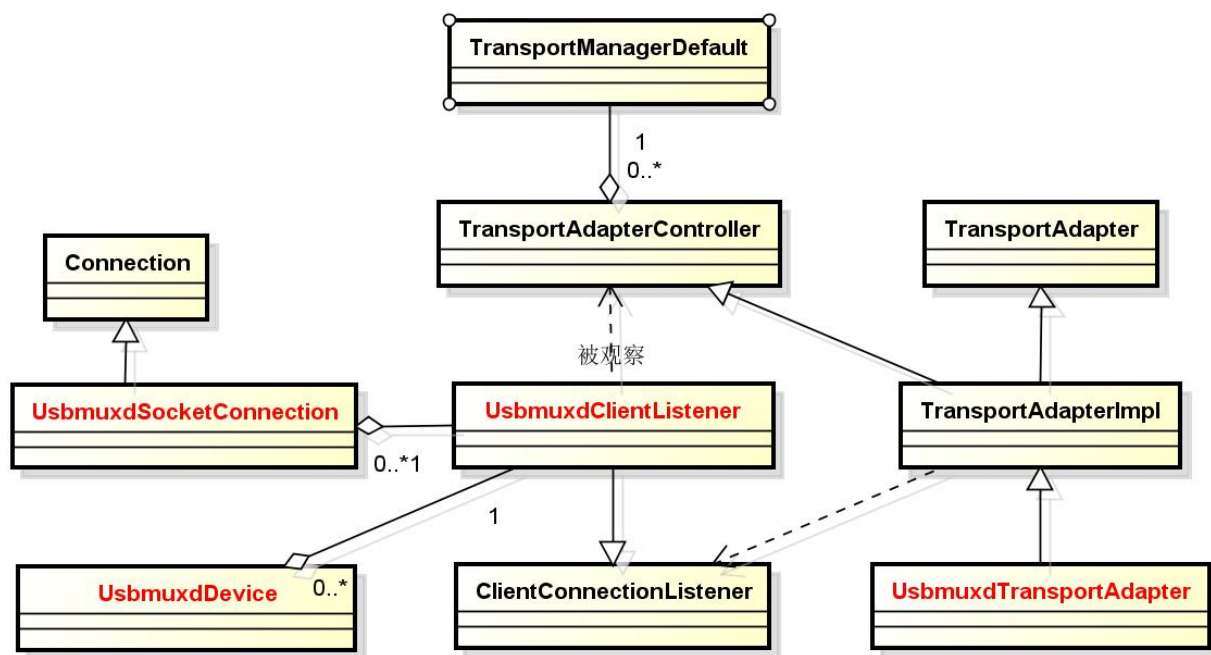
2.2 sdl_core 和手机 app 关系包图



3 sdl_core usbmuxd 模块类图



Project Name: usbmuxd	Version: <1.0.0>
SDL 完整集成 usbmuxd 协议说明文档	Date: <24/11/2017>
<document identifier>	

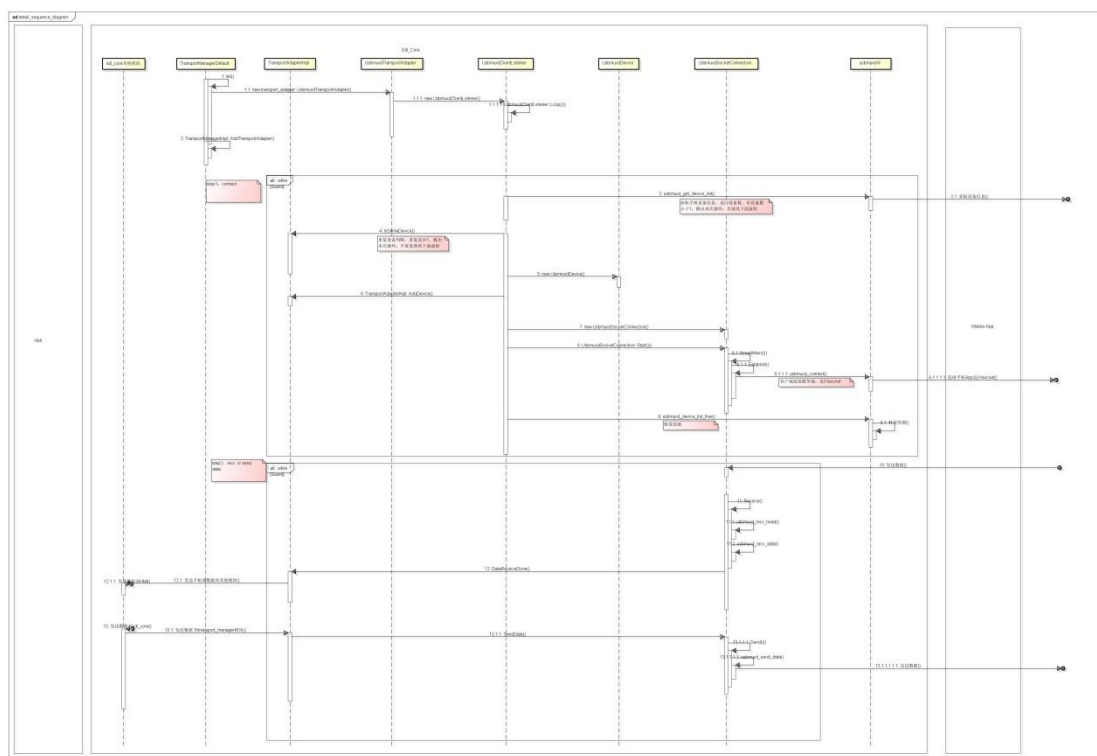


4 时序图

4.1 sdl_core usbmuxd 模块初始化、连接及收发数据时序图



Project Name: usbmuxd	Version: <1.0.0>
SDL 完整集成 usbmuxd 协议说明文档	Date: <24/11/2017>
<document identifier>	



5 usbmuxd 协议 sdl_core 端接口函数

5.1 int usbmuxd_get_device_list()

int usbmuxd_get_device_list(usbmuxd_device_info_t **device_list)

描述:

获取设备列表

参数:

usbmuxd_device_info_t **device_list): 设备列表详细信息的指针

参见:

usbmuxd.h、usbmuxd-proto.h

返回值:

设备列表数量

注释:

无

5.2 int usbmuxd_device_list_free()

int usbmuxd_device_list_free(usbmuxd_device_info_t **device_list)



本作品采用[知识共享署名-相同方式共享 4.0 国际许可协议](https://creativecommons.org/licenses/by-sa/4.0/)进行许可

Project Name: usbmuxd	Version: <1.0.0>
SDL 完整集成 usbmuxd 协议说明文档	Date: <24/11/2017>
<document identifier>	

描述:

释放设备列表资源

参数:

usbmuxd_device_info_t **device_list): 设备列表详细信息的指针

参见:

usbmuxd.h、usbmuxd-proto.h

返回值:

成功返回 0，错误返回-1

注释:

无

5.3 int usbmuxd_connect()

int usbmuxd_connect(const int handle, const unsigned short tcp_port)

描述:

与设备建立 TCP 连接

参数:

const int handle 连接设备的句柄

const unsigned short tcp_port 服务器端口

参见:

usbmuxd.h、usbmuxd-proto.h

返回值:

成功返回 socket 句柄，错误返回-1

注释:

无

5.4 int usbmuxd_disconnect()

int usbmuxd_disconnect(int sfd)

描述:

断开设备连接

参数:

int sfd: socket 句柄

参见:

usbmuxd.h、usbmuxd-proto.h

返回值:

成功返回 0，错误返回-1

注释:

无



本作品采用[知识共享署名-相同方式共享 4.0 国际许可协议](https://creativecommons.org/licenses/by-sa/4.0/)进行许可

Project Name: usbmuxd	Version: <1.0.0>
SDL 完整集成 usbmuxd 协议说明文档	Date: <24/11/2017>
<document identifier>	

6 usbmuxd 协议 Iphone 端接口函数

6.1 (void)ioFrameChannel

(void)ioFrameChannel:(PTChannel*)channel didReceiveFrameOfType:(uint32_t)type tag:(uint32_t)tag payload:(PTData*)payload

描述:

收到数据包

参数:

channel 连接通道

type 数据包类型

tag 数据包标签

payload 收到的数据包

参见:

SDLUSBMUXDChannel.h

返回值:

无

注释:

无

6.2 (BOOL)ioFrameChannel

(BOOL)ioFrameChannel:(PTChannel*)channel shouldAcceptFrameOfType:(uint32_t)type tag:(uint32_t)tag payloadSize:(uint32_t)payloadSize

述:

是否接受收到的数据

参数:

channel 连接通道

type 数据类型

payloadSize 数据包大小

参见:

SDLUSBMUXDChannel.h

返回值:

接受返回YES，不接受返回NO

注释:



本作品采用[知识共享署名-相同方式共享 4.0 国际许可协议](https://creativecommons.org/licenses/by-sa/4.0/)进行许可

Project Name: usbmuxd	Version: <1.0.0>
SDL 完整集成 usbmuxd 协议说明文档	Date: <24/11/2017>
<document identifier>	

无

6.3 (void)ioFrameChannel

(void)ioFrameChannel:(PTChannel*)channel didEndWithError:(NSError*)error

描述:

连接通道已关闭

参数:

channel 连接通道

error 关闭错误提示

参见:

SDLUSBMUXDChannel.h

返回值:

无

注释:

无

6.4 (void)ioFrameChannel

**(PTChannel*)channel
fromAddress:(PTAddress*)address**

didAcceptConnection:(PTChannel*)otherChannel

描述:

监听到新的连接

参数:

channel 连接通道

otherChannel 新的连接通道

address 新的连接地址

参见:

SDLUSBMUXDChannel.h

返回值:

无

注释:

无

6.5 (void)sendFrameOfType

**(void)sendFrameOfType:(uint32_t)frameType
withPayload:(dispatch_data_t)payload**

**tag:(uint32_t>tag
overChannel:(dispatch_io_t)channel**



本作品采用[知识共享署名-相同方式共享 4.0 国际许可协议](https://creativecommons.org/licenses/by-sa/4.0/)进行许可

Project Name: usbmuxd	Version: <1.0.0>
SDL 完整集成 usbmuxd 协议说明文档	Date: <24/11/2017>
<document identifier>	

callback:(void(^)(NSError *error))callback

描述:

发送数据包

参数:

frameType 数据包类型

tag 数据包标签

payload 数据包

channel 连接通道

callback 发送完毕后回调方法

参见:

SDLUSBMUXDChannel.h

返回值:

无

注释:

无

6.6 (void)readFramesOverChannel

(void)readFramesOverChannel:(dispatch_io_t)channel onFrame:(void(^)(NSError *error,uint32_t type,uint32_t tag,uint32_t payloadSize,dispatch_block_t resumeReadingFrames))onFrame

描述:

从连接通道读取收到的数据包

参数:

channel 连接通道

onFrame 读取到的数据包

参见:

SDLUSBMUXDChannel.h

返回值:

无

注释:

无

6.7 (void)readPayloadOfSize

(void)readPayloadOfSize:(size_t)payloadSize overChannel:(dispatch_io_t)channel callback:(void(^)(NSError *error, dispatch_data_t contiguousData,const uint8_t *buffer,size_t bufferSize))callback



本作品采用[知识共享署名-相同方式共享 4.0 国际许可协议](https://creativecommons.org/licenses/by-sa/4.0/)进行许可

Project Name: usbmuxd	Version: <1.0.0>
SDL 完整集成 usbmuxd 协议说明文档	Date: <24/11/2017>
<document identifier>	

描述:

从缓冲区读取数据包

参数:

payloadSize 数据包大小

channel 连接通道

callback 读取到数据包后回调函数

参见:

SDLUSBMUXDChannel.h

返回值:

无

注释:

无

6.8 (void)readAndDiscardDataOfSize

**(void)readAndDiscardDataOfSize:(size_t)size overChannel:(dispatch_io_t)channel
callback:(void(^)(NSError *error, BOOL endOfStream))callback**

描述:

从通道丢弃数据包

参数:

size 数据包大小

channel 连接通道

callback 完成后回调函数

参见:

SDLUSBMUXDChannel.h

返回值:

无

注释:

无

7 usbmuxd 协议帧格式说明

sdl_core 和 Iphone 手机 app 发送和接收数据必须按照特定的帧格式进行解析和打包。

帧格式: 帧头+数据内容

帧头内容: length (真实数据内容长度, 4 字节) + 真实数据内容

帧头:

```
struct usbmuxd_header_real {
```



本作品采用[知识共享署名-相同方式共享 4.0 国际许可协议](https://creativecommons.org/licenses/by-sa/4.0/)进行许可

Project Name: usbmuxd	Version: <1.0.0>
SDL 完整集成 usbmuxd 协议说明文档	Date: <24/11/2017>
<document identifier>	

```

uint32_t headerframe; //Frame of header identification: &&&&
uint32_t version; // protocol version
uint32_t message;    // message type
uint32_t tag;        // responses to this query will echo back this tag
uint32_t length;     //data length

} __attribute__((__packed__));

```

