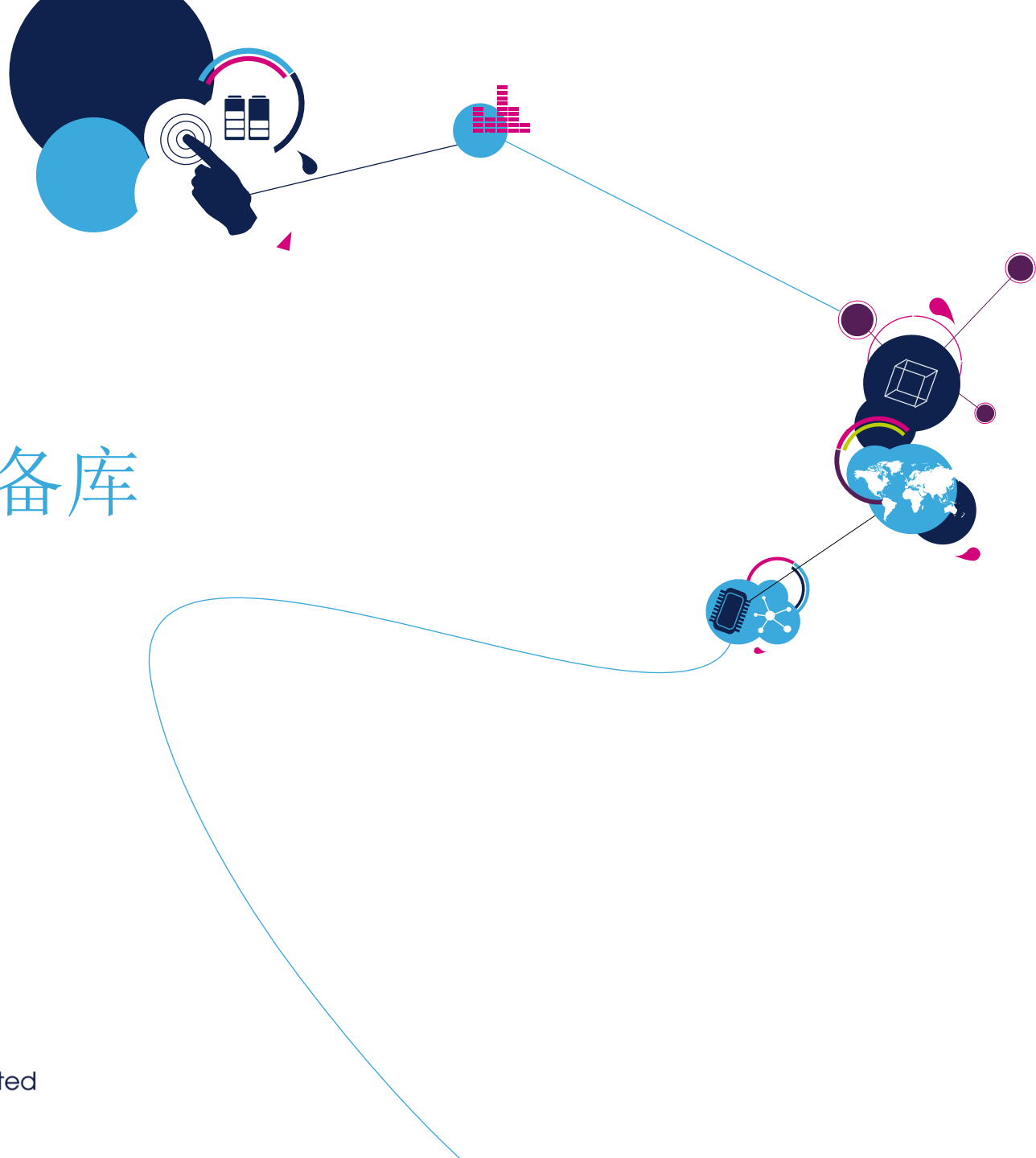


USB IP设备库

Lilian YAO



- 项目结构
 - 库函数、USB库函数
 - 类相关文件
 - 应用相关文件
- 库代码结构
 - 初始化
 - USB中断处理流程
 - 回调接口函数
- 如何基于已有例程做自己的应用裁剪
 - 需要用到的EP及其静态配置（硬件收发缓冲区的配置、端点类型、端点地址）
 - EP0上特定类相关命令、用户自定义命令的处理
 - 非0EP上如何进行数据收发

USB IP设备库有三类

3

USB(+) IP		
	Legacy library	Cube library
F0x2	STSW-STM32092	STM32CubeF0
F103	STSW-STM32121	STM32CubeF1
F3		STM32CubeF3
L1		STM32CubeL1
L0		STM32CubeL0

STSW-STM32092 STM32F0x2xx USB FS device library (UM1717)

● Active

STSW-STM32121

STM32F10x, STM32L1xx and STM32F3xx USB full speed device library (UM0424)

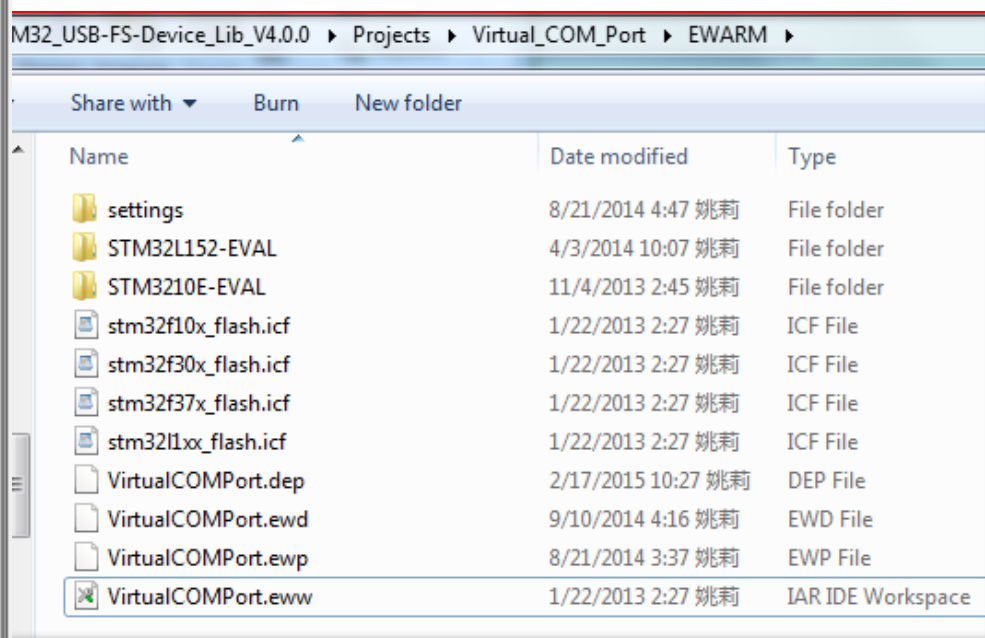
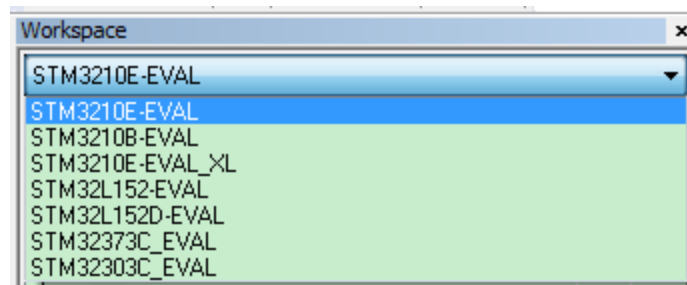
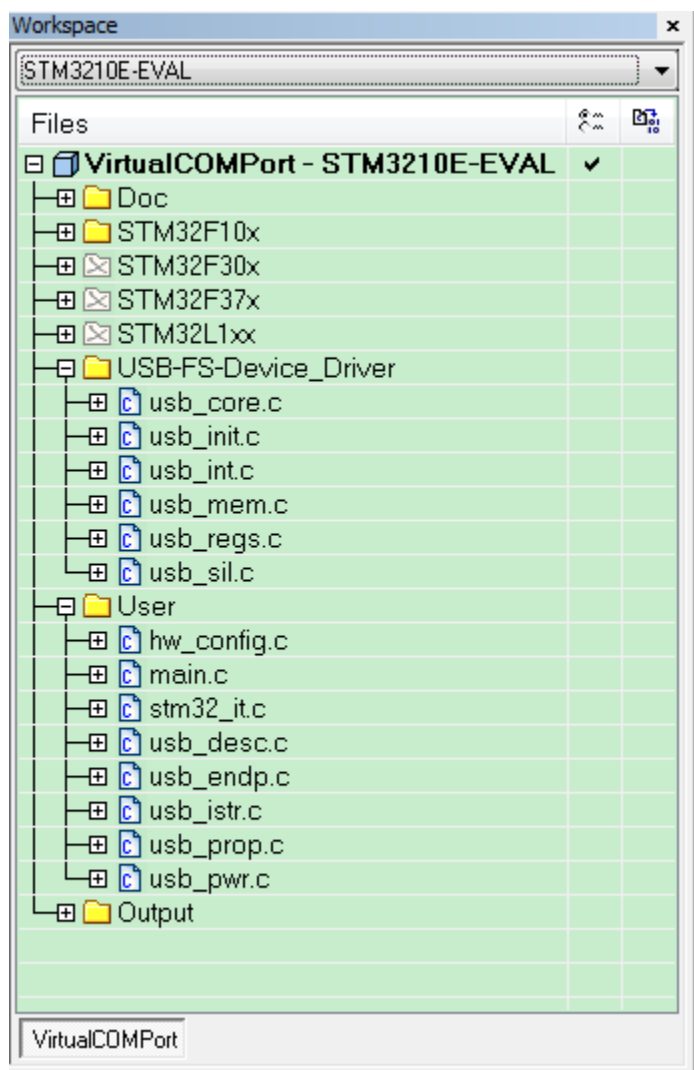
● Active

USB IP设备库有三个

4

USB(+) IP		
	Legacy library	Cube library
F0x2	STSW-STM32092	STM32CubeF0
F103	STSW-STM32121 VCP demo	STM32CubeF1
F3		STM32CubeF3
L1		STM32CubeL1
L0		STM32CubeL0

- **STSW-STM32121**中运行在STM3210E-EVAL板上的VCP例程





切换到IAR Workbench窗口...

Device_Property @ <usb_prop.c>

7

- Virtual_Com_Port **init**
 - Main() → USB_Init() → **pProperty->Init();**
- Virtual_Com_Port **Reset**
 - USB_Istr() → ISTR_RESET: : **Device_Property.Reset();**
- Virtual_Com_Port **Status_In**
 - In0_Proecss() → 该IN transaction是Control transfer的IN方向/status stage时: 调用(***pProperty->Process_Status_IN()**); 说明该control transfer是一个带OUT方向数据阶段的控制传输
 - 《VCP demo》“SET_LINE_CODING” request → USART_Config()
- Virtual_Com_Port **Status_Out**
 - Out0_Process() → 该OUT transaction是Control transfer的OUT方向/status stage时: 调用(***pProperty->Process_Status_OUT()**); 说明该control transfer是一个带IN方向数据阶段的控制传输

Device_Property @ <usb_prop.c> Conti.

8

- Virtual_Com_Port **NoData_Setup**

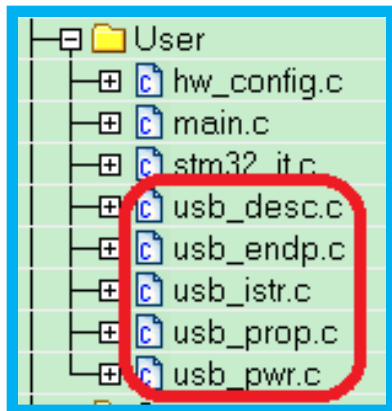
- Setup0_Proecss() → 不带数据阶段的、类相关的命令时：调用(*pProperty->Class_NoData_Setup())
- 《VCP demo》“SET_COMM_FEATURE”, “SET_CONTROL_LINE_STATE” request → 直接返回_SUCCESS; 其他命令返回_UNSUPPORT

- Virtual_Com_Port **Data_Setup**

- Setup0_Proecss() → 带数据阶段的、类相关的命令时：调用(*pProperty->Class_Data_Setup())
- 《VCP demo》“GET_LINE_CODING”, “SET_LINE_CODING”

应用相关的USB文件

9



需要修改的USB文件	用户操作
Usb_desc.c	设备描述符
Usb_endp.c	应用中用到的每个非0端点的OUT和IN transaction的处理
Usb_istr.c	
Usb_prop.c	Device_Property的每个成员函数的实现（至少前6个）
Usb_pwr.c	在PowerOn()中设置应用关心的USB中断
Usb_prop.h	注释掉需要实现的User_Standard_Requests中的成员函数的默认NOP_Process
Usb_conf.h	应用用到的单向物理端点数目；packet buffer描述表的位置、各自的硬件收发缓冲区地址、应用关心的USB中断；注释掉非0端点上默认NOP_Process

wlstr = _GetISTR() 查看中断触发源

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTR	PMA OVR	ERR	WKUP	SUSP	RESET	SOF	ESOF	Reserved				DIR	EP_ID[3:0]		
r	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0					r	r	r	r

#if (IMR_MSK & ISTR_SOF)

```
if (wlstr & ISTR_SOF & wInterrupt_Mask)
{
    _SetISTR((uint16_t)CLR_SOF);
    bIntPackSOF++;
}
```

#ifdef SOF_CALLBACK

SOF_Callback();

#endif

}

#endif**#if (IMR_MSK & ISTR_RESET)**

```
if (wlstr & ISTR_RESET & wInterrupt_Mask)
{
    _SetISTR((uint16_t)CLR_RESET);
    Device_Property.Reset();
}
```

#ifdef RESET_CALLBACK

RESET_Callback();

#endif

}

#endif**#if (IMR_MSK & ISTR_CTR)**

```
if (wlstr & ISTR_CTR & wInterrupt_Mask)
{
```

CTR_LP();

#ifdef CTR_CALLBACK

CTR_Callback();

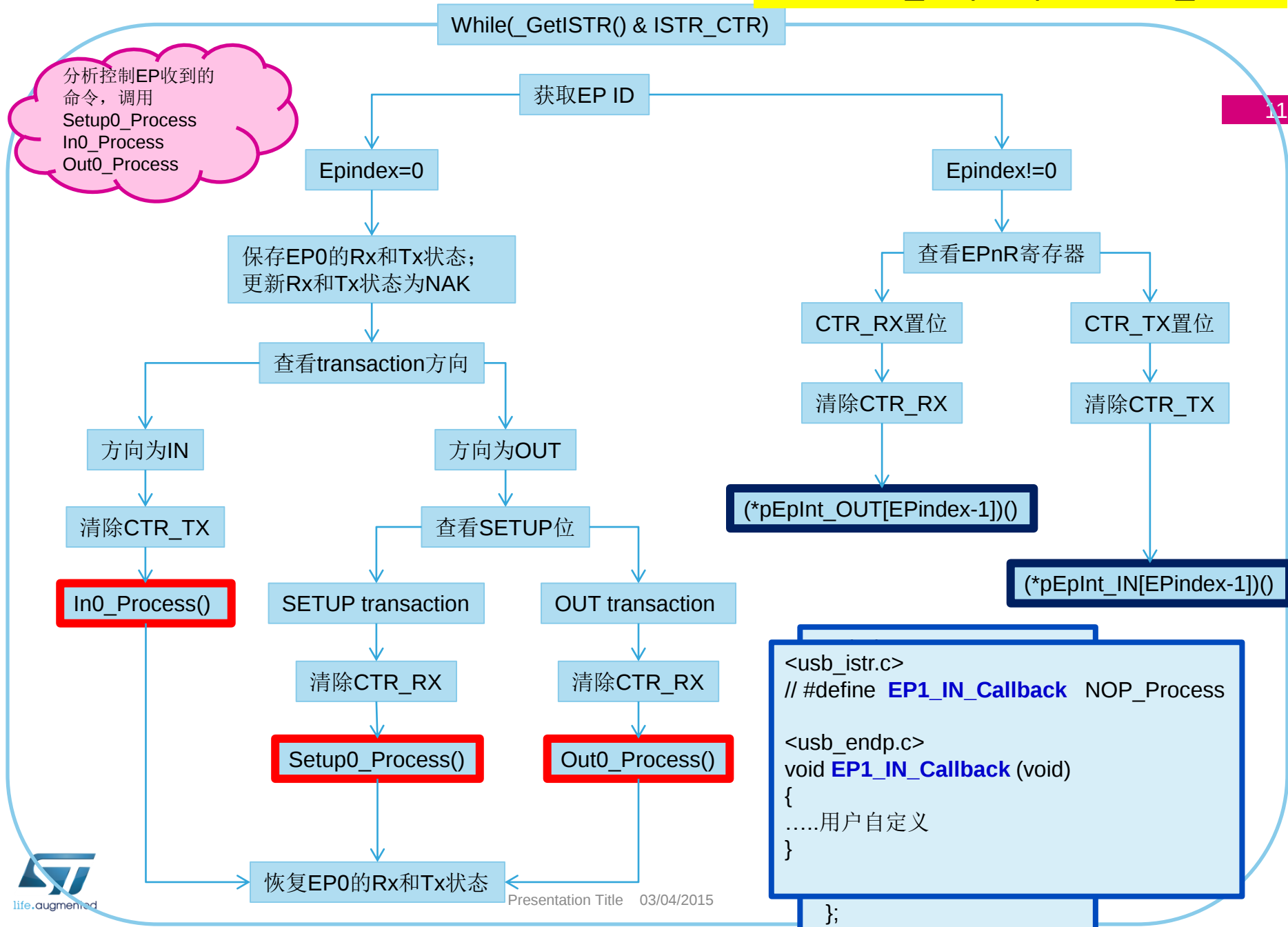
#endif

}

#endif

@ <usb_conf.h>

```
#define IMR_MSK (CNTR_CTRM | CNTR_WKUPM |
CNTR_SUSPM | CNTR_ERRM | CNTR_SOFM \
| CNTR_ESOFM | CNTR_RESETM)
```



USB transactions

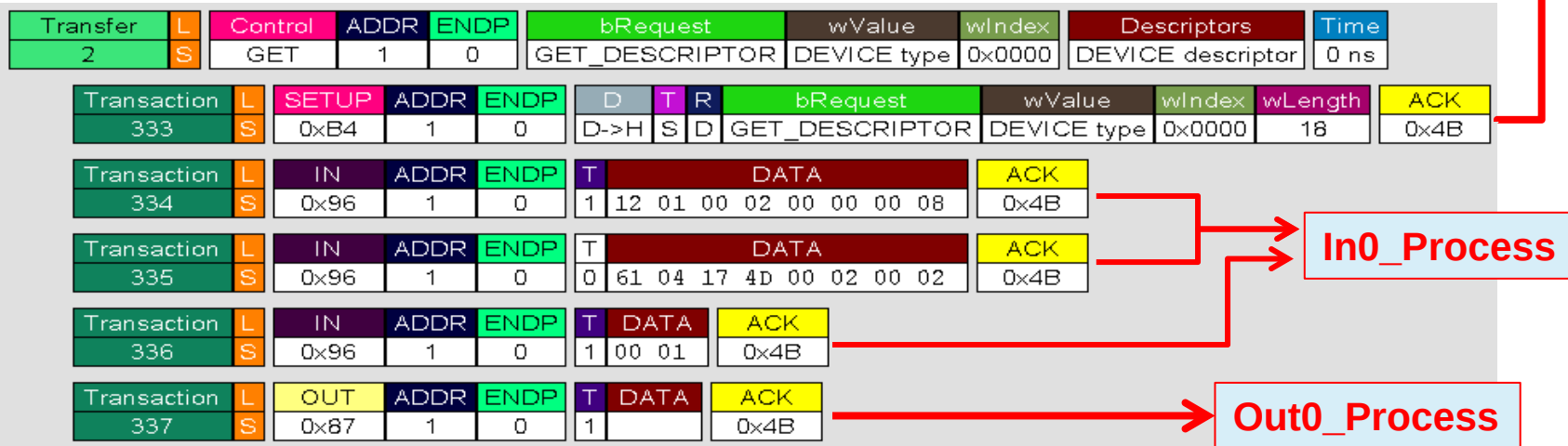
- Transaction可以分成三类

- Setup transaction: 主机用来向设备发送控制命令
- Data IN transaction: 主机用来从设备读取数据
- Data OUT transaction: 主机用来向设备发送数据

总是由主机发起

- Transaction的packet组成

- Token packet: 总是由主机发出
- Data packet: 包含此次transaction的数据负载
- 可选的Handshake packet



USB IP设备库有三个

14

USB(+) IP		
	Legacy library	Cube library
F0x2	STSW-STM32092	STM32CubeF0
F103	STSW-STM32121 VCP demo	STM32CubeF1
F3		STM32CubeF3
L1		STM32CubeL1
L0		STM32CubeL0

CONFIDENTIALITY OBLIGATIONS:

THIS DOCUMENT CONTAINS SENSITIVE INFORMATION.

IT IS CLASSIFIED "MICROCONTROLLERS, MEMORIES & SECURE MCUs (MMS) RESTRICTED AND ITS DISTRIBUTION IS SUBMITTED TO ST/MMS AUTHORIZATION

AT ALL TIMES YOU SHOULD COMPLY WITH THE FOLLOWING SECURITY RULES:

- DO NOT COPY OR REPRODUCE ALL OR PART OF THIS DOCUMENT
- KEEP THIS DOCUMENT LOCKED AWAY
- FURTHER COPIES CAN BE PROVIDED ON A "NEED TO KNOW BASIS", PLEASE CONTACT YOUR LOCAL ST SALES OFFICE OR DOCUMENT WRITTER.

Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of STMicroelectronics.

The ST logo is registered trademark of STMicroelectronics
All other names are the property of their respective owners

© 2015 STMicroelectronics - All Rights Reserved

STMicroelectronics group of companies Australia - Brazil - Canada - China - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States.

www.st.com