

AUTOSAR4.2.2 基于Can的通信栈

普华汽车电子事业部
iSOFT AE BU

2021年5月

普华基础软件股份有限公司

iSOFT INFRASTRUCTURE SOFTWARE CO.,
LTD



1

Can通信栈简介



2

CanDrv



3

CanIf



4

PduR



5

Com



Part 1

Can通信栈简介

1.1 Can通信栈概述



1.2 Can通信栈框架

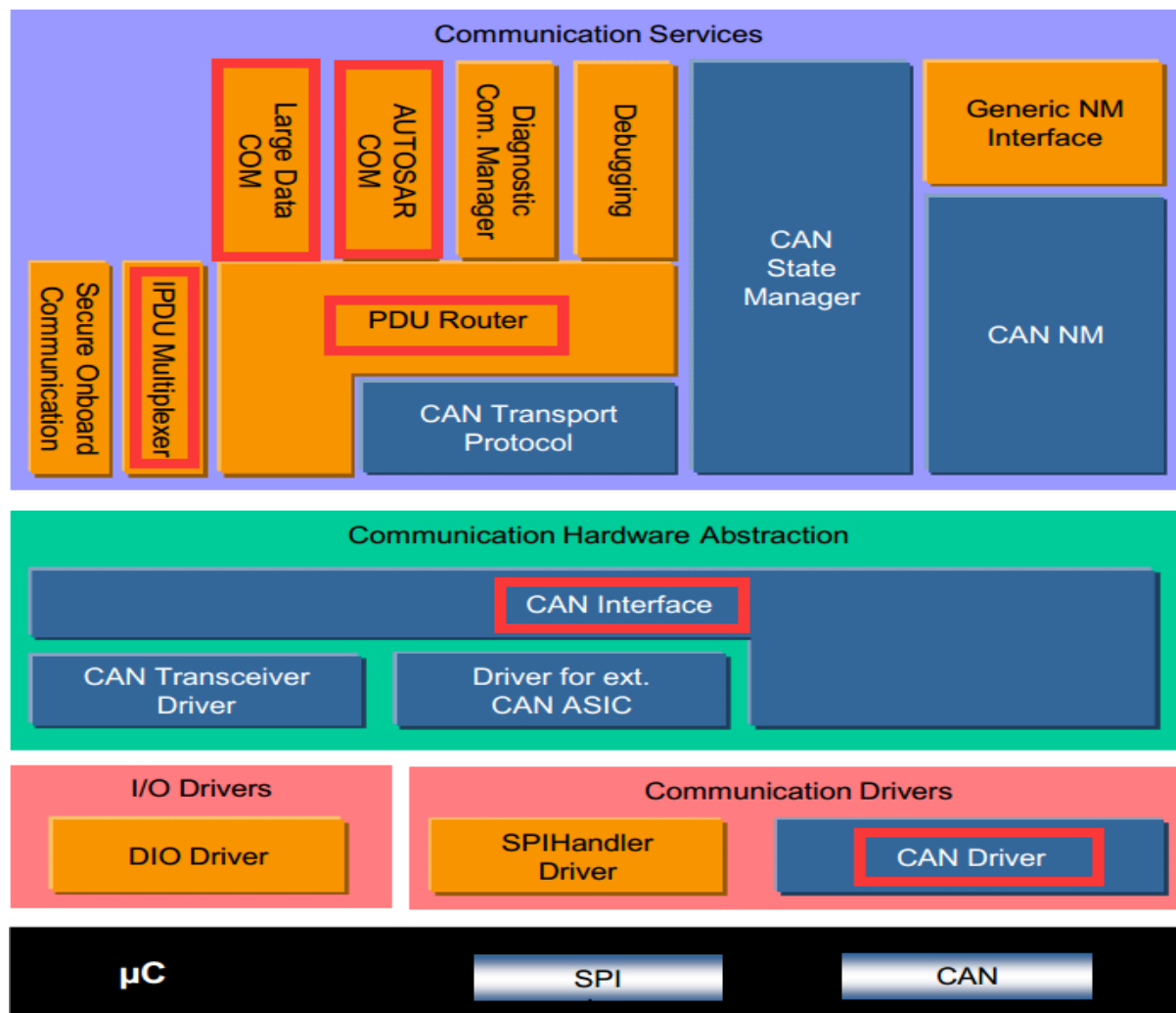


1.1 Can通信概述

- CAN通信栈是在CAN总线上完成Signal的接收/发送功能、Signal的路由转发以及报文的路由转发等功能。

1.2 通信栈框架

- AUTOSAR中基于Can的通信栈框架如图所示。



Part 2

CanDrv简介

2.1 CanDrv通信栈概述



2.2 CanDrv功能简介



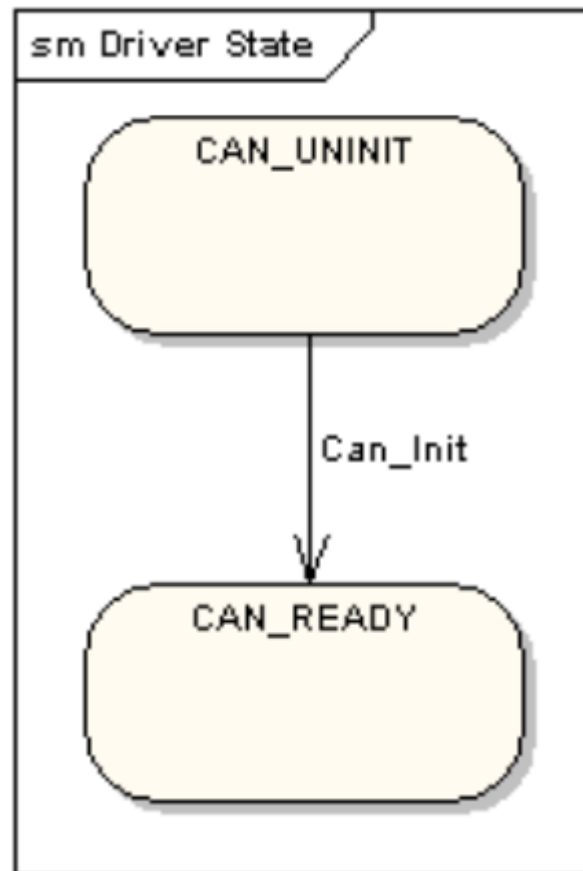
2.1 CanDrv概述

- CAN Driver处于通信最底层，与硬件进行交互，将不同类型硬件所进行的操作封装为统一的API接口提供给上层，隔离硬件与上层服务。
- CAN Driver包含与硬件相关的各配置项，将物理层发生的事件通过回调函数的方式通知上层，控制CAN控制器的行为及状态，并提供与收发报文相关的服务。

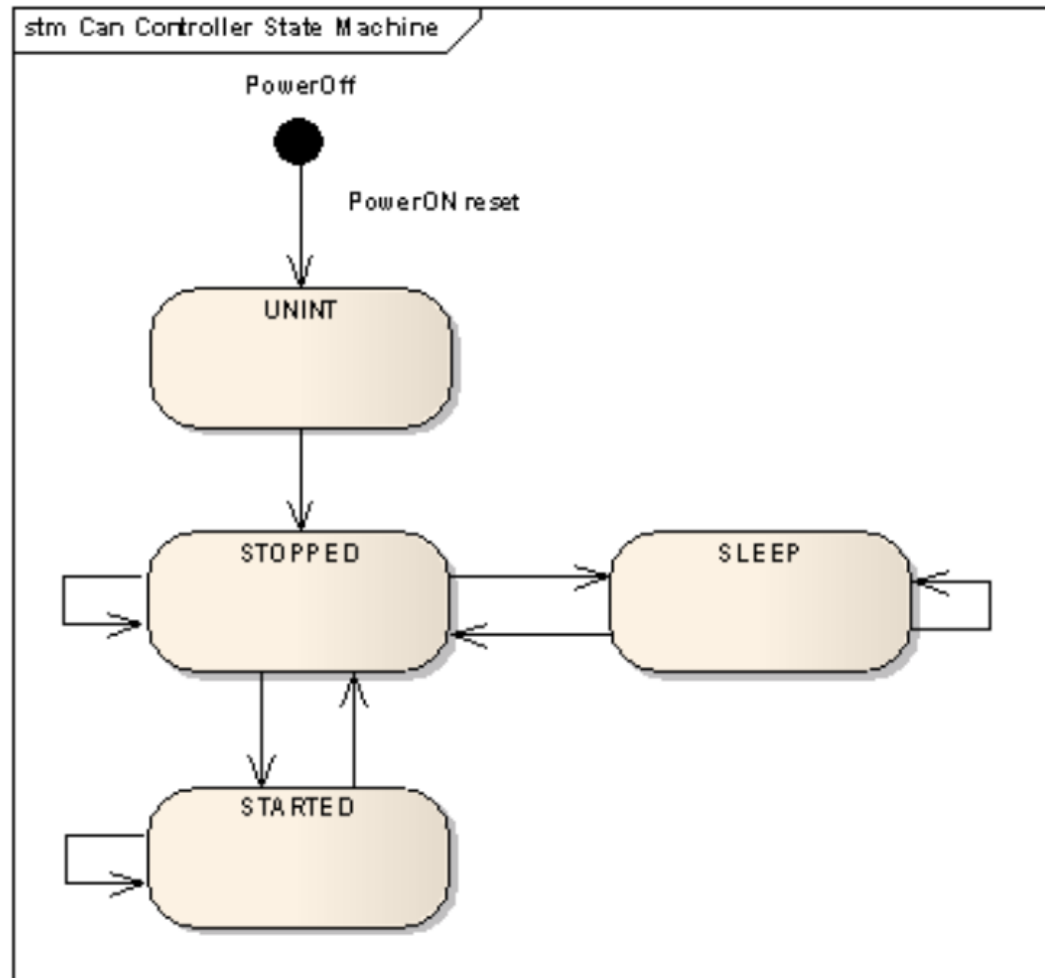
2.2 CanDrv功能

- CAN驱动API命名规则
- CAN驱动的状态机
- CAN Controller状态机
- CAN模块/Controller的初始化
- L-PDU的发送/接收
- CAN驱动唤醒
- 通知机制
- 错误分类

2.2.2 驱动状态机



2.2.3 Controller状态机

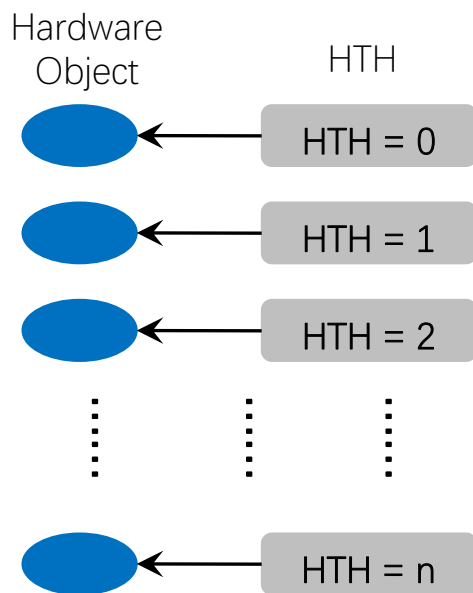


2.2.4 CanDrv初始化

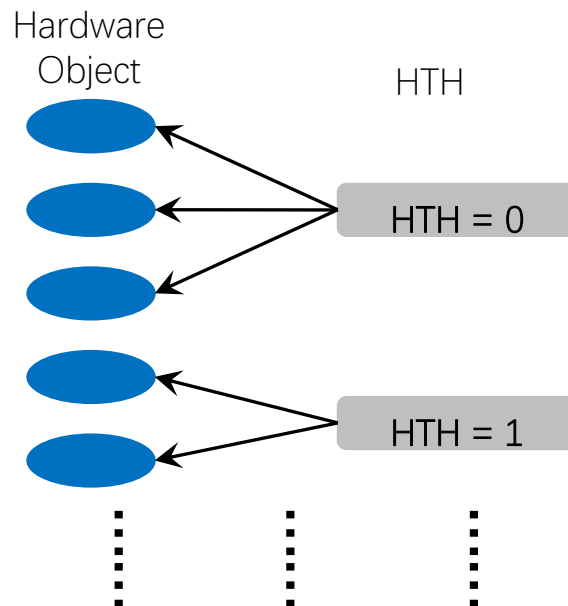
- Can_Init在CAN模块中别的API之前被调用（除Can_GetVersionInfo）。
- 初始化内容：
 - ① 静态变量，包括各种flag；
 - ② 整体硬件单元的一般寄存器设置；
 - ③ 每个CAN Controller的特定设置；

2.2.5 PDU的收发

- 发送时可选择多路发送Multiplexed Transmission:



非多路发送



多路发送

- 确保收发数据的一致性。

2.2.6 驱动唤醒

➤ 唤醒有两种方式：中断和轮询。

➤ 与EcuM模块交互：

① EcuM_CheckWakeup;

② EcuM_SetWakeupEvent;

2.2.7 通知机制

➤ 通知机制有两种方式：

① 中断

② 轮询

➤ 与CanIf模块交互(除唤醒与EcuM交互外)：

① CanIf_ControllerBusOff;

② CanIf_RxIndication;

③ CanIf_TxConfirmation;

.....

Part 3

CanIf简介

3.1 CanIf概述



3.2 CanIf功能简介



3.1 CanIf概述

- CAN Driver处于通信最底层，与硬件进行交互，将不同类型硬件所进行的操作封装为统一的API接口提供给上层，隔离硬件与上层服务。
- CAN Driver包含与硬件相关的各配置项，将物理层发生的事件通过回调函数的方式通知上层，控制CAN控制器的行为及状态，并提供与收发报文相关的服务。

3.2 CanIf功能

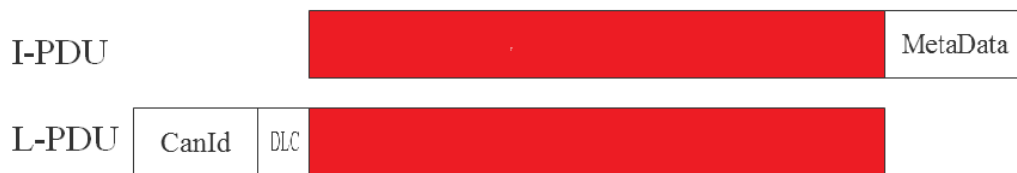
- Hardware object handles (HOH)
- 动态/静态L-PDU以及MetaData概念
- 接收L-PDU的BasicCAN和FullCAN概念
- L-PDU的发送
- L-PDU的接收
- CAN Controller模式控制
- PDU模式控制
- 软件滤波
- DLC检测
- 对于多个CAN驱动的支持
- 错误分类

3.2.1 HOH

- HOH为CAN邮箱的抽象，关联CAN Controller中1-n个具体邮箱。
- HOH分为Hardware Receive Handle（HRH）和Hardware Transmit Handle（HTH）。

3.2.2 静态/动态L-PDU及MetaData概念

- 这里动态指的是L-PDU的CanId在运行时会改变。
- 对于发送L-PDU改变其CanId的途径有两种：
 - ① 通过调用CanIf_SetDynamicTxId接口来实现；
 - ② 通过MetaData来实现。
- MetaData长度为0-4个byte，在I-PDU数据域后面（小端顺序）



- 发送L-PDU的真正CanId计算方式：

$$(\text{CanIfTxPduCanId} \ \& \ \text{CanIfTxPduCanIdMask}) \mid (\text{MetaData} \& (\sim \text{CanIfTxPduCanIdMask}))$$

3.2.2 静态/动态L-PDU及MetaData概念

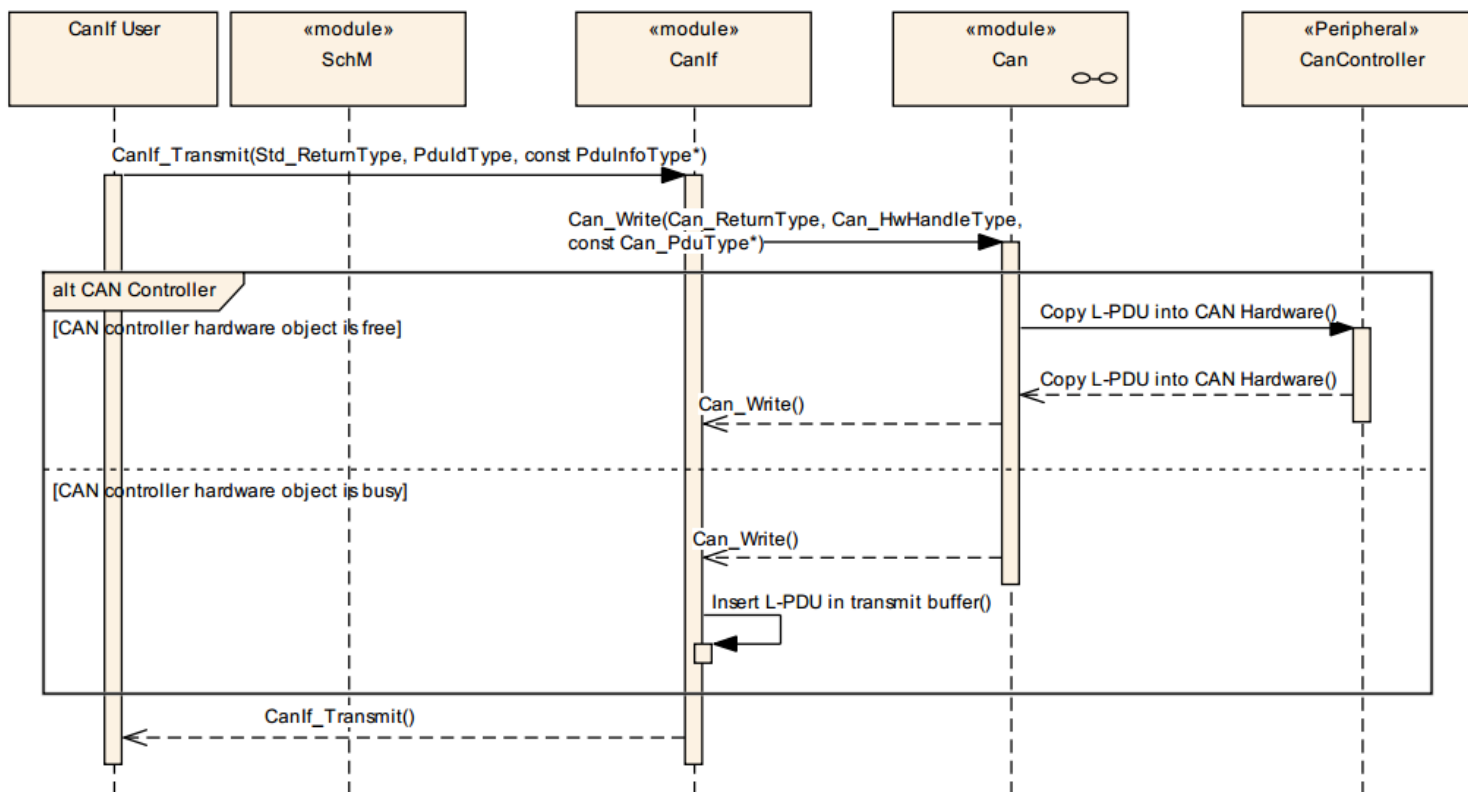
- 其中CanIfTxPduCanId由配置的CanId或者 CanIf_SetDynamicTxId决定。Rx L-PDU配置了MetaData时，将接收到的CanId作为MetaData放在I-PDU数据域之后，传给上层模块，Pdu长度为数据域长度+MetaData长度。

3.2.3 BasicCAN和FullCAN概念

- FullCAN指接收邮箱只收一个CanId，通过硬件滤波实现；
- BasicCAN指接收邮箱能接收多个CanIds。
- 由CAN驱动中设置邮箱的滤波掩码决定。如果是BasicCAN，在CanIf中需要进行软件滤波。

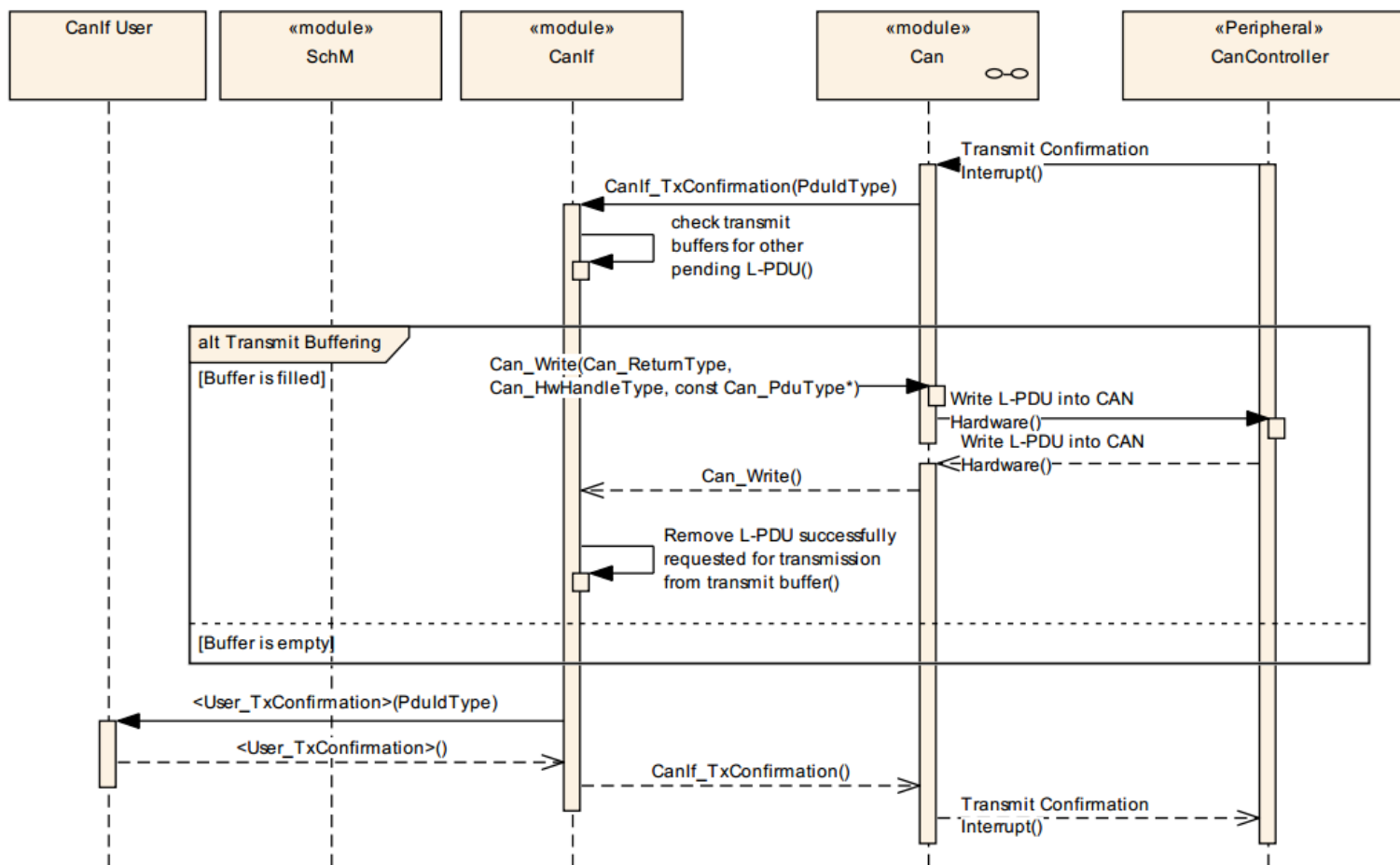
3.2.4 L-PDU的发送

➤ L-PDU发送请求

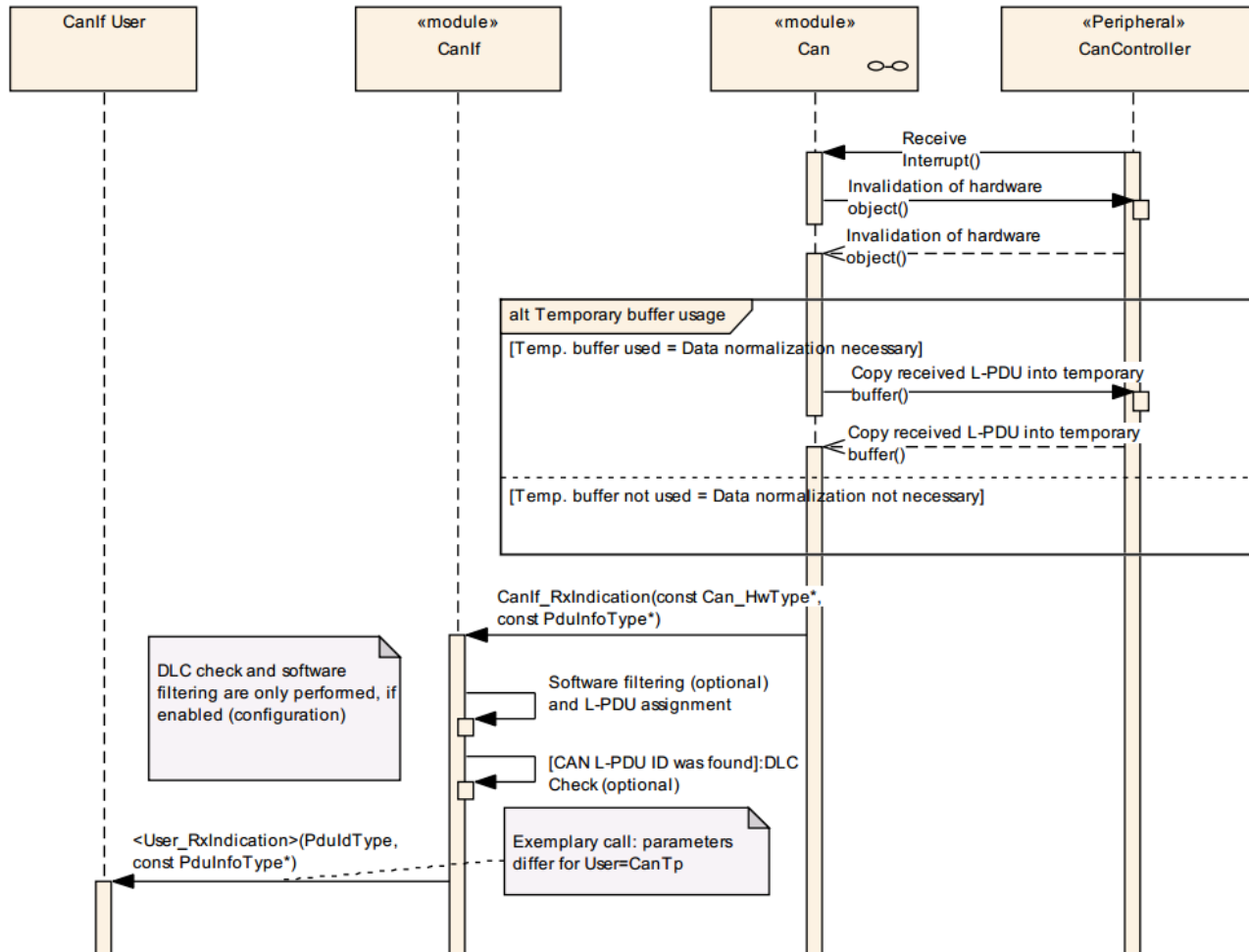


3.2.5 L-PDU的发送

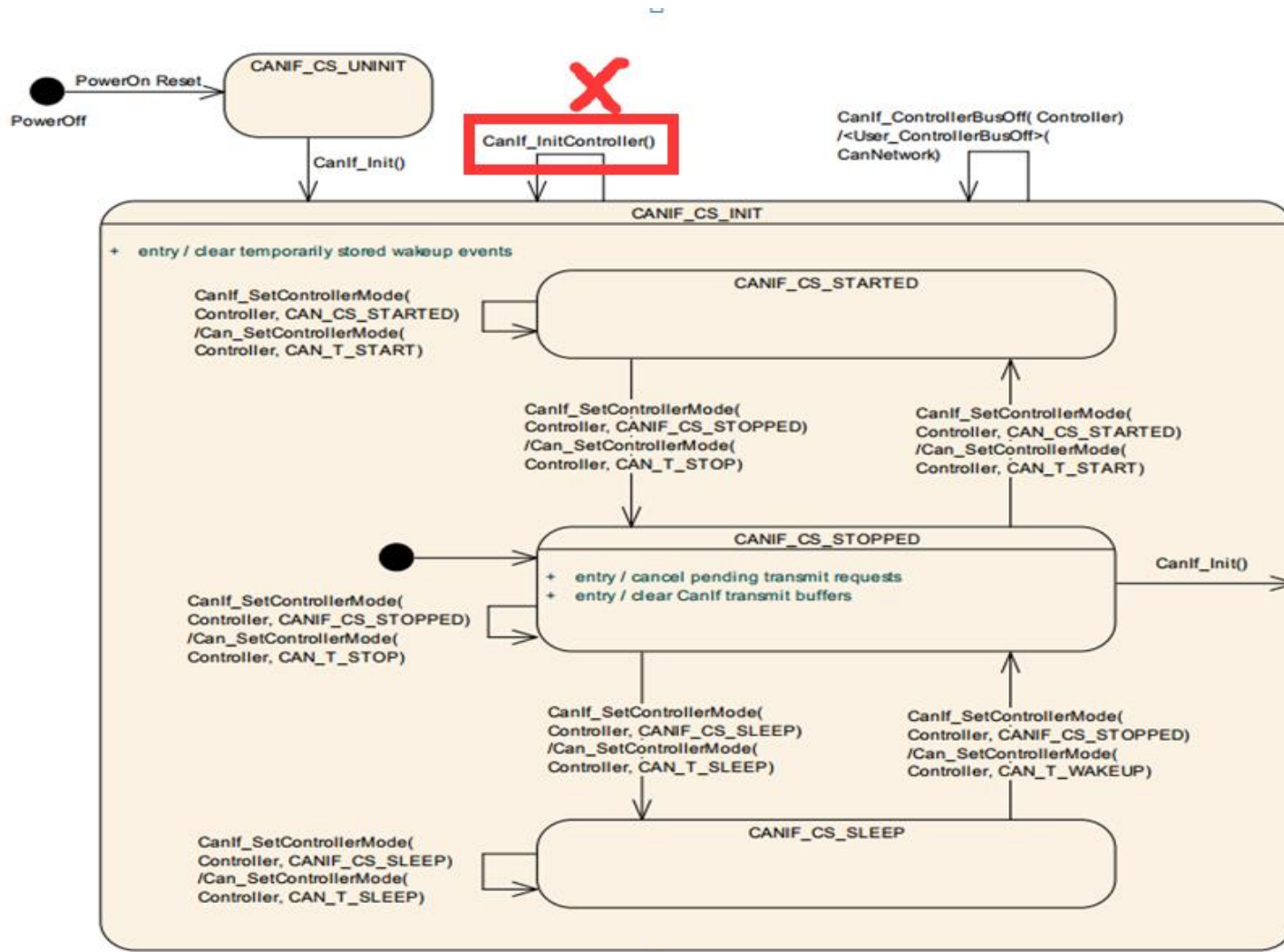
➤ L-PDU发送确认（带Tx Buffer）



3.2.6 L-PDU的接收



3.2.7 Controller模式控制



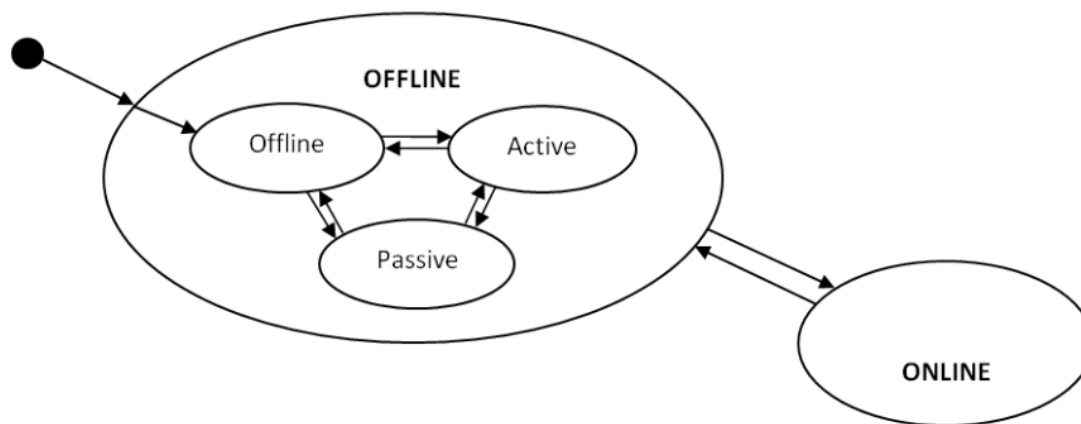
3.2.8 PDU模式控制

➤ PDU模式可看作为Controller模式为STARTED时的细分。

➤ 分别对应4种模式：

CANIF_ONLINE、CANIF_OFFLINE、CANIF_TX_OFFLINE、
CANIF_TX_OFFLINE_ACTIVE;

➤ 其中，CANIF_TX_OFFLINE_ACTIVE模式用作特殊情况（如diagnosis passive mode），该模式下不真正发送L-PDU,但会通过TxConfirmation通知上层模块发送完成。



3.2.9 软件滤波

➤ 软件滤波分2部分：

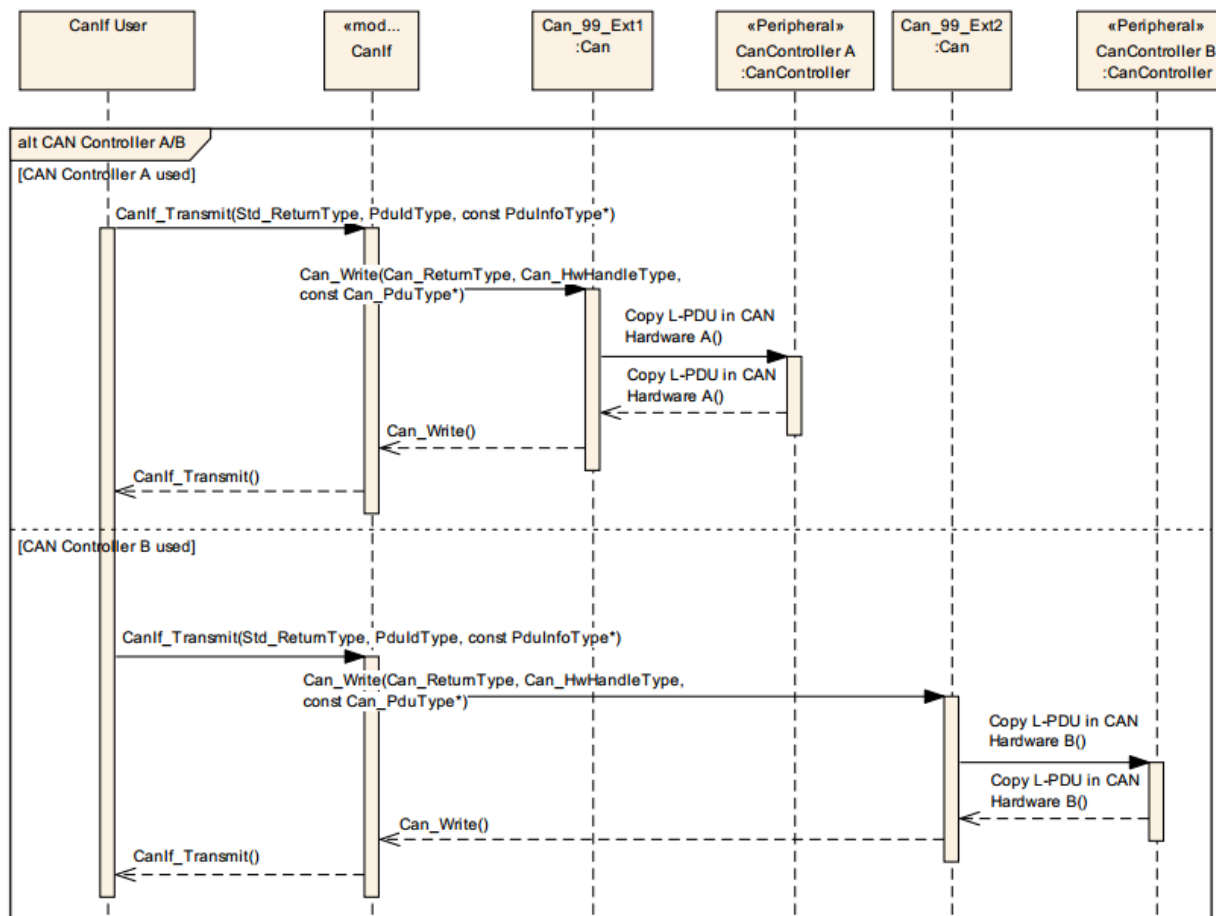
- ① HRH滤波：因为一般硬件邮箱都是通过邮箱掩码按bit位来进行滤波，这里实现基于上下限HRH滤波作为补充。
- ② 通过HRH滤波之后，根据接收到的CanId找到匹配的Rx L-PDU（目前代码实现LINEAR滤波方式）。

3.2.10 DLC检测

- 对于Rx L-PDU要实现DLC检测功能，需要配置该PDU的最小长度。只有当接收到的PDU长度 \geq 配置的最小长度时，DLC检测通过。

3.2.11 多CAN驱动的支持

➤ 这里涉及到CAN驱动API的命名规则。



Part 4

PduR简介

4.1 PduR概述



4.2 PduR功能简介



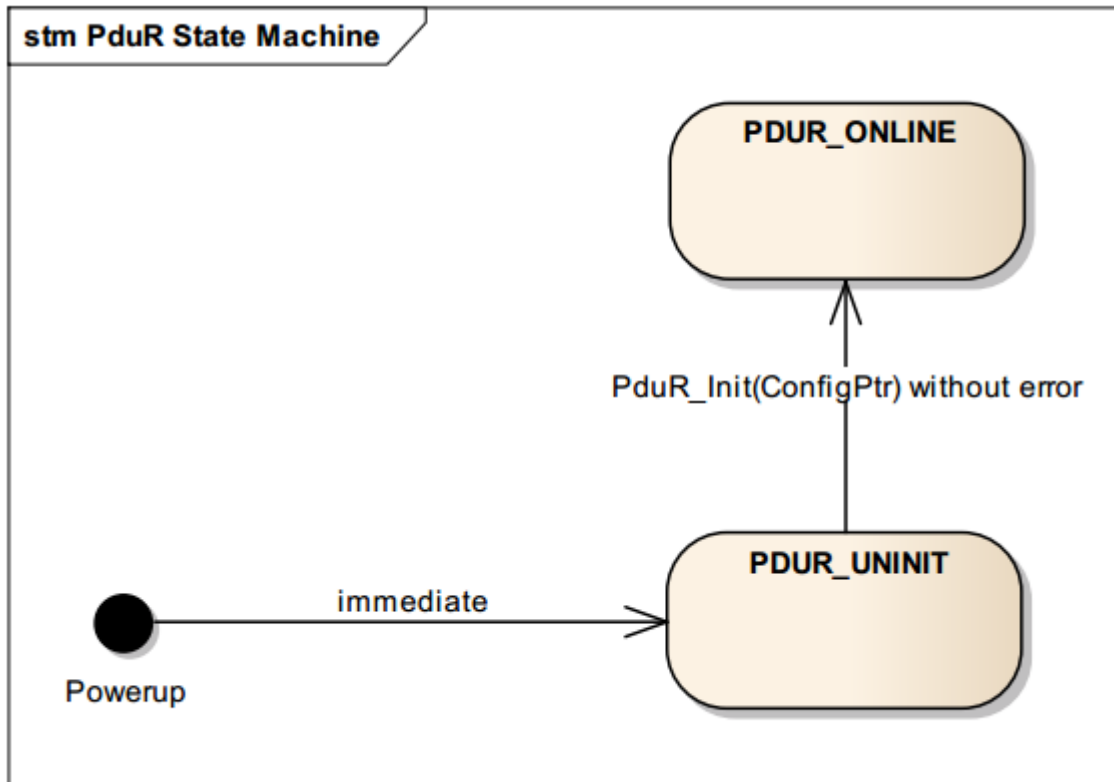
4.1 PduR概述

- PDU Router主要为通讯接口模块（CANIF）、传输协议模块（CAN TP、J1939 TP）、诊断通讯管理模块（DCM、J1939DCM）以及通讯模块（COM、LDCOM）以及IPDUM、SECOC等模块提供基于I-PDU的路由服务。
- PDU Router基础软件模块在通信栈中起着承上启下的功能，为上层服务基础软件模块和应用屏蔽了网络细节，使得上层基础软件模块和应用不用关心应用运行于哪种总线网络之上。同时，PDU Router提供了基于I-PDU的网关功能，使得不同总线之间的通讯成为可能。

4.1 PduR功能

- PDUR状态管理
- Routing Table路由表
- I-PDU的路由功能（核心）
- Zero Cost Operation概念
- Routing path groups概念
- 错误分类

4.2.1 PduR状态机



4.2.2 Routing Table路由表

- Routing Table由1-n个Routing Paths组成，每个Routing Path由1个Src Pdu和1-n个Dest Pdus组成。
- 只有当PDUR配置为Zero Cost Operation时才可不需路由表。

4.2.3 I-PDU的路由功能

- I-PDU的Routing Path根据路由方向分为3种：接收路由、发送路由、网关路由，其中接收路由和网关路由可合并。
- 接收路由：
 - ① IF路由，Src Pdu关联PDUR下层IF模块（如CanIf），Dest Pdu关联到PDUR上层模块（如Dcm）；
 - ② TP路由，Src Pdu关联PDUR下层TP模块（如CanTp），Dest Pdu中只能关联到一个PDUR上层模块（如Dcm）。

4.2.3 I-PDU的路由功能

➤ 发送路由：

- ① IF路由，Src Pdu关联PDUR上层模块（如Com），Dest Pdu关联到PDUR下层模块（如CanIf）；
- ② TP路由，Src Pdu关联PDUR上层模块（如Dcm），Dest Pdu关联到PDUR下层模块（如CanTp），只有单帧才可1:n TP传输；

4.2.3 I-PDU的路由功能

➤ 网关路由：

- ① IF路由，Src Pdu关联PDUR下层模块（如CanIf），Dest Pdu关联到PDUR下层模块（如CanIf、linIf）；
- ② TP路由，Src Pdu关联PDUR下层模块（如CanTp），Dest Pdu关联到PDUR下层模块（如CanTp、linTp），TP网关时可以通过配置阈值实现“gateway on the fly”；

➤ 注意：TP路由不能与IF路由混合。

4.2.4 Zero Cost Operation概念

- 当PDUR上下层模块唯一对应，不需要网关功能，这时PDUR可以选择“零消耗”模式，该模式下“相当于没有PDUR模块”。
- 在该模式下，COM与CANIF唯一对应，DCM与CANTP唯一对应，J1939DCM与J1939TP唯一对应，PDU的传输“透过”PDUR模块。

4.2.5 Routing path groups概念

- Routing path groups的概念理解为一组Routing path不大恰当，应理解为一组Dest Pdus。

A routing path group is a group of I-PDUs that can be disabled and enabled during runtime. The group contains the destination I-PDUs and not the routing path itself.

- PduR_EnableRouting/ PduR_DisableRouting可使能/不使能Routing path group中包含的Dest Pdus。
- 对于不属于任何Routing path group的Dest Pdus，其状态一直为Enable，不会改变。

4.2.6 错误分类

➤ 开发错误

<i>Type or error</i>	<i>Related error code</i>	<i>Value [hex]</i>
Invalid configuration pointer	PDUR_E_INIT_FAILED	0x00
Null pointer has been passed as an argument	PDUR_E_PARAM_POINTER	0x09

➤ 运行错误（也是通过DET接口报错）

<i>Type of error</i>	<i>Related error code</i>	<i>Value [hex]</i>
Loss of a PDU instance (buffer overrun in gateway operation)	PDUR_E_PDU_INSTANCES_LOST	0x0a
API service used without module initialization or PduR_Init called in any state other than PDUR_UNINIT	PDUR_E_INVALID_REQUEST	0x01
Invalid PDU identifier	PDUR_E_PDU_ID_INVALID	0x02
TP module rejects a transmit request for a valid PDU identifier	PDUR_E_TP_TX_REQ_REJECTED	0x03
If the routing table is invalid that is given to the PduR_EnableRouting or PduR_DisableRouting functions	PDUR_E_ROUTING_PATH_GROUP_ID_INVALID	0x08

Part 5

Com简介

5.1 Com概述



5.2 Com功能简介



5.1 Com概述

- AUTOSAR COM模块主要处理信号的接收和发送功能，并为RTE层提供信号接收和发送接口函数。
- AUTOSAR COM集成了OSEK COM3.0.3标准的大部分功能，并在此基础上扩展了信号网关，信号组等功能。

5.2 Com功能

- 将signals封装到I-PDUs中发送
- 将接收到的I-PDUs解包成signals, 供RTE获取
- signals的网关功能
- signal groups的网关功能
- I-PDU通信控制, 通过Com_IpduGroupControl实现
- 发送请求触发多次发送 (Direct/Mixed发送模式下)
- Minimum Delay Timer (MDT) I-PDU发送最小间隔时间
- Deadline monitoring (DM) 收发signals的超时监控

5.2 Com功能

- 接收signals过滤，发送signals滤波计算TMC
- 各种通知机制
- 提供signals初始值/无效值/更新bit机制
- 字节大小端选择
- 有符号数据类型signals接收时符号位扩展
- 每个发送I-PDU可配1-2种发送模式
- 支持大数据I-PDU，动态长度I-PDU收发
- 支持I-PDU counters，支持通信保护（发送时基于PDUR1:n实现）

5.2.1 Signal类型

- 信号种类：一般信号signal，信号组signal group，组信号group signal，描述信号description signal；
- 支持的信号types

Range	BOOLEAN
	FLOAT32
	FLOAT64
	SINT16
	SINT32
	SINT64
	SINT8
	UINT16
	UINT32
	UINT64
	UINT8
	UINT8_DYN
	UINT8_N

5.2.2 Signal对齐方式

➤ 信号封装/解封涉及到byte的大小端

➤ 小端

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	7	6	5	4	3	2	1	0
Byte 1	15 ← 2	14 ← 1	13 ← 0 LSB	12	11	10	9	8
Byte 2	23 ← 10	22 ← 9	21 ← 8	20 ← 7	19 ← 6	18 ← 5	17 ← 4	16 ← 3
Byte 3	31	30	29	28	27	26	25	24 MSB ← 11
Byte 4	39	38	37	36	35	34	33	32

➤ 大端

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	7	6	5	4	3	2	1	0
Byte 1	15	14	13 MSB ← 11	12 ← 10	11 ← 9	10 ← 8	9 ← 7	8 ← 6
Byte 2	23 ← 5	22 ← 4	21 ← 3	20 ← 2	19 ← 1	18 ← 0 LSB	17	16
Byte 3	31	30	29	28	27	26	25	24
Byte 4	39	38	37	36	35	34	33	32

5.2.3 Signal过滤

- 接收到的signal值为-3，该信号bit size为10，类型为sint16
- 从I-PDU中解析出来的数据为1111111101b，需要扩展为111111111111111101b
- 滤波方式：
 - ALWAYS
 - NEVER
 - MASKED_NEW_EQUALS_X
 - MASKED_NEW_DIFFERS_X
 - MASKED_NEW_DIFFERS_MASKED_OLD
 - NEW_IS_WITHIN
 - NEW_IS_OUTSIDE
 - ONE EVERY_N

5.2.3 Signal过滤

➤ Tx/Rx signal滤波不同的作用

- ① Rx signal滤波是过滤掉不想要的signal值；
- ② Tx signal滤波为计算出各个signal的TMC值，每个Tx I-PDU中所有signal的TMCs。决定其TMS，根据TMS选择I-PDU的发送模式（当配置了2种发送模式时）。

5.2.4 Transfer Property

➤ 发送信号的传输属性Com Transfer Property

- ① PENDING
- ② TRIGGERED
- ③ TRIGGERED_ON_CHANGE
- ④ TRIGGERED_ON_CHANGE_WITHOUT_REPETITION
- ⑤ TRIGGERED_WITHOUT_REPETITION

5.2.5 Update Bit

- 当信号配置了update bit时，在更新发送信号时需要置1；只有当接收信号的update bit置1，才会执行正常信号接收流程，否则该信号被舍弃。

5.2.6 Signal网关

- Rx I-PDU中signal/group signal/ source description signal网关到Tx I-PDU中signal/group signal/dest description signal;
- Rx I-PDU中signal group网关到Tx I-PDU中signal group, 其包含的group signals顺序要相同。

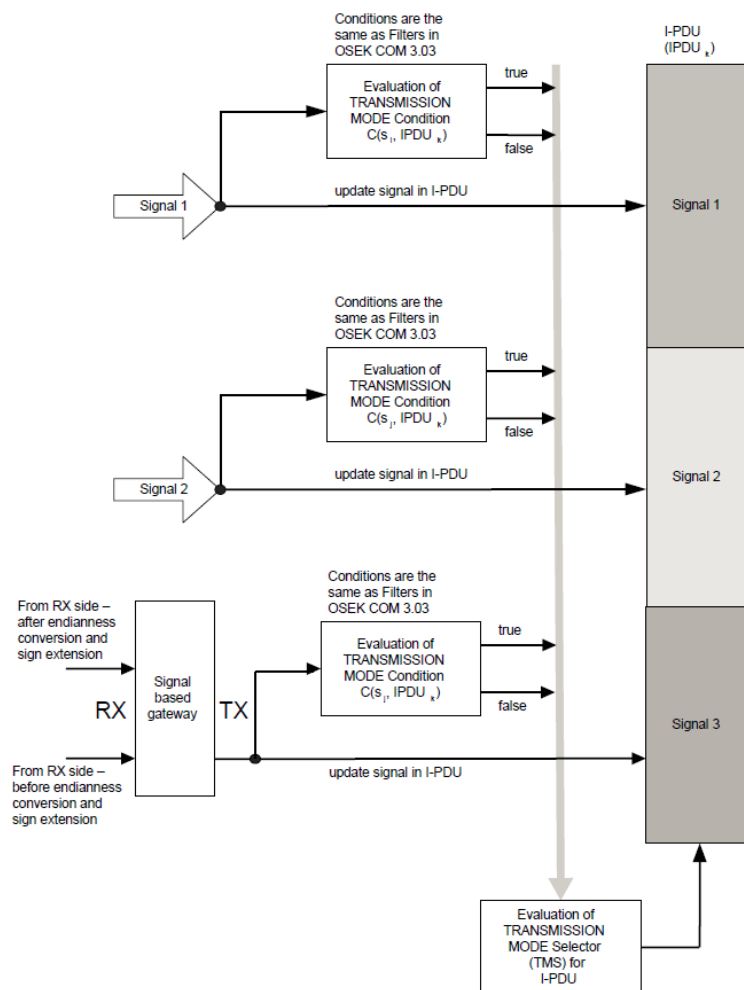
5.2.7 IPDU的发送模式及方式

- 发送I-PDU可配置1-2种发送模式：

ComTxModeTrue和ComTxModeFalse，当有2种模式时根据TMS来选择哪种模式进行发送；

- 每种发送模式中可选择4种发送方式：

- ① DIRECT
- ② MIXED
- ③ NONE
- ④ PERIODIC



5.2.8 IPDU的序列控制

➤ 发送I-PDU:

- ✓ I-PDU发送请求成功后（调用PduR_ComTransmit返回E_OK），counter加1。对于通过Com_TriggerTransmit发送I-PDU,其counter不加1。

➤ 接收I-PDU:

- ✓ I-PDU initialized by Com_Init or reinitialized by Com_IpduGroupControl with parameter Initialize set to true, 这时任何counter值的I-PDU都接收，并其该counter加1作为下一个接收I-PDU的期望counter;

5.2.9 IPDU的序列控制

- 如果收到的counter与期望的counter不匹配，调用配置的通知接口
ComIPduCounterErrorNotification;
- counter值的翻转，假如counter配置为4个bit，期望counter为15之后翻转到0;
- counter阈值及匹配算法

举例：期望counter为5，阈值为2，则当接收到的counter为5/6/7都是OK的。

5.2.10 IPDU通信保护

- 通信保护功能的实现依赖于I-PDU序列控制功能，这时counter的阈值ComIPduCounterThreshold必须配为0。
- 发送I-PDU
 - ✓ 通过PDUR中1:n（2-3）路由实现。
- 接收I-PDU
 - ✓ 当成功接收ComIPduReplicationQuorum个I-PDU时（counter通过，PDU数据完全一致）才执行I-PDU正常接收流程。

Part 6

LdCom简介

6.1 LdCom概述



6.2 LdCom发送操作



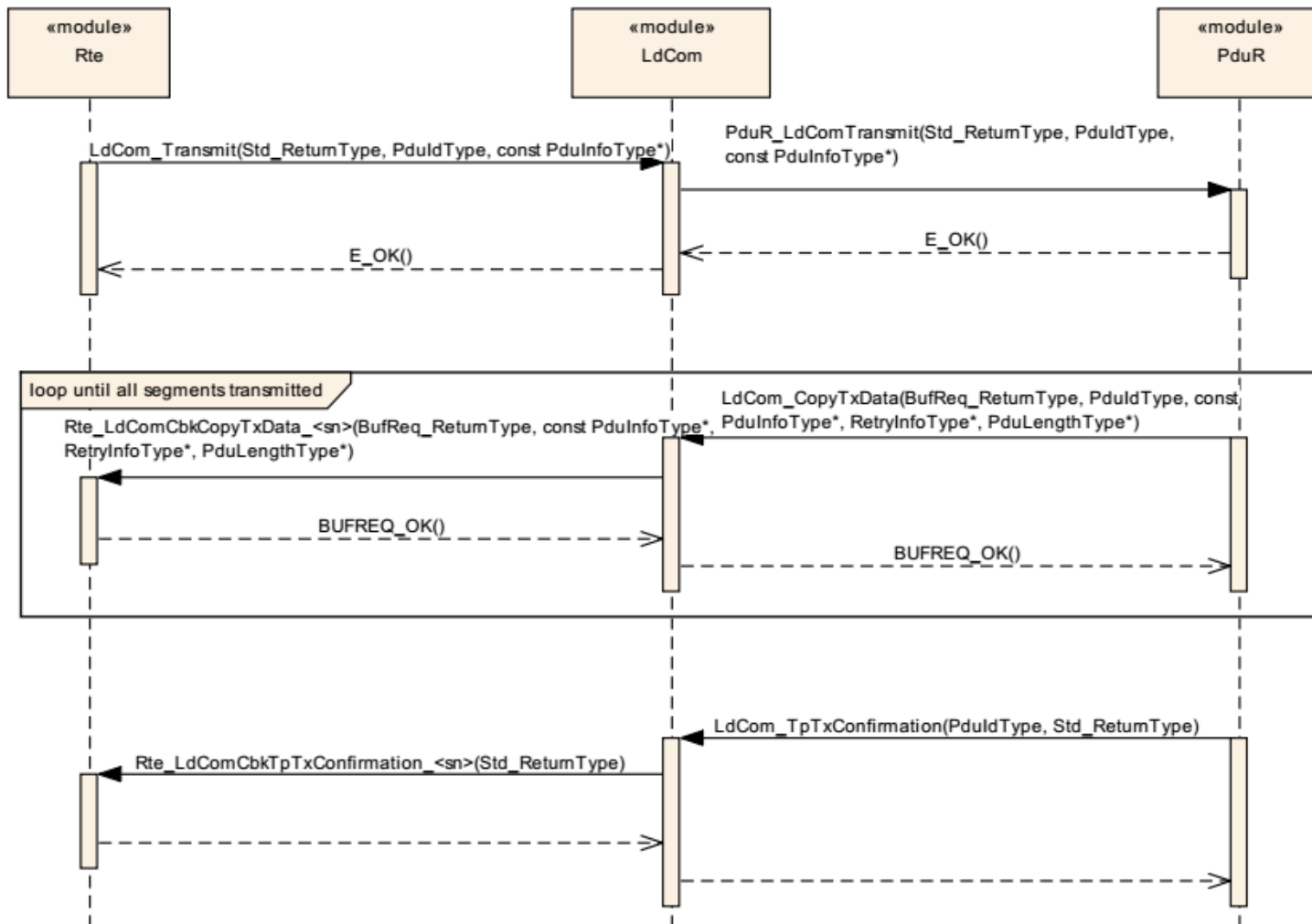
6.3 LdCom接收操作



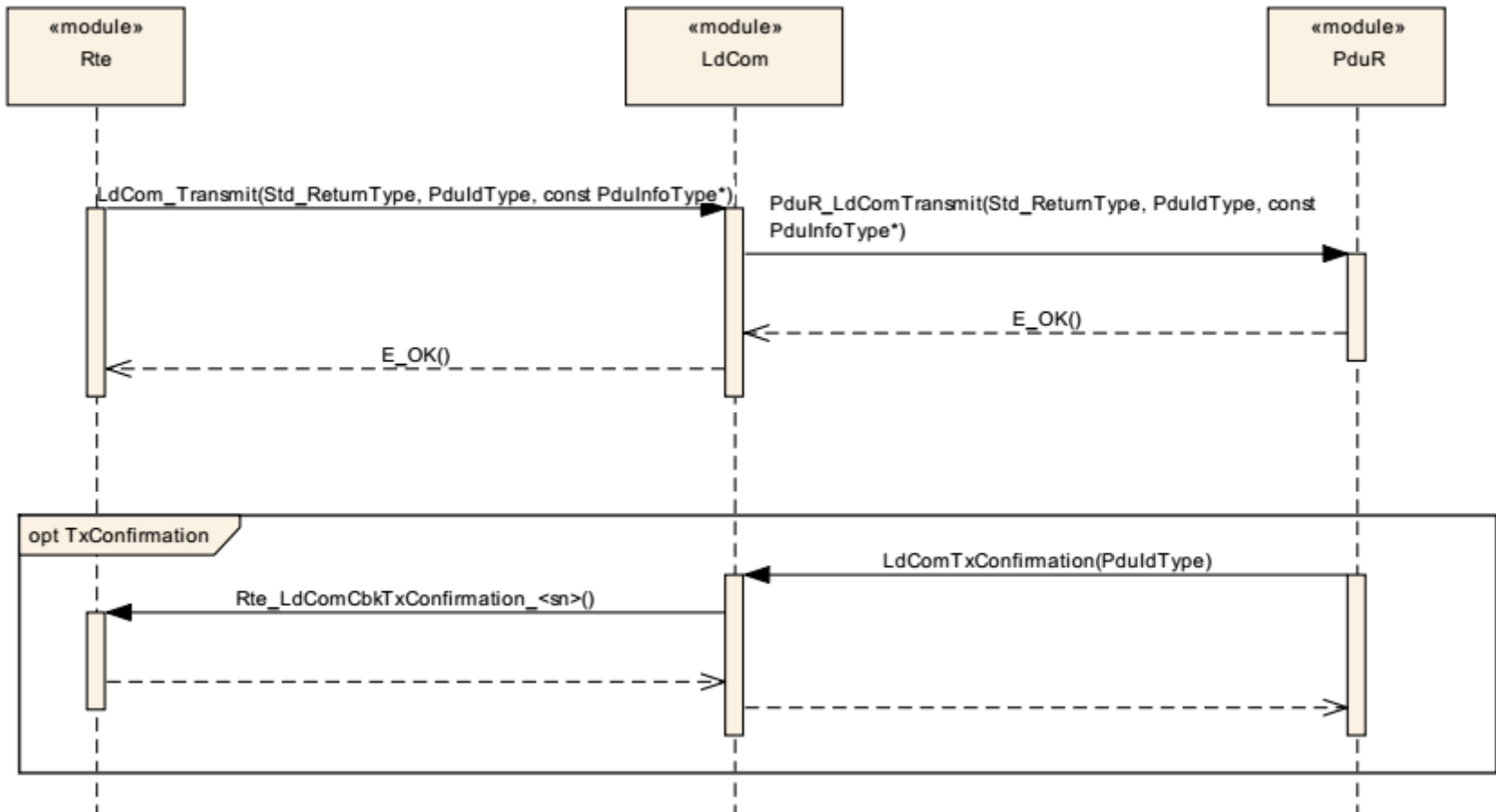
6.1 LdCom概述

- AUTOSAR LDCOM模块可认为是功能及其简单的COM模块，在RTE与PDUR之间起过渡作用。
- AUTOSAR LDCOM中每个PDU只包含一个字节对齐的signal，LD (large data) 这里指的是signal特性。
- PDU报文的封装与解析、发送的时机在应用中实现。

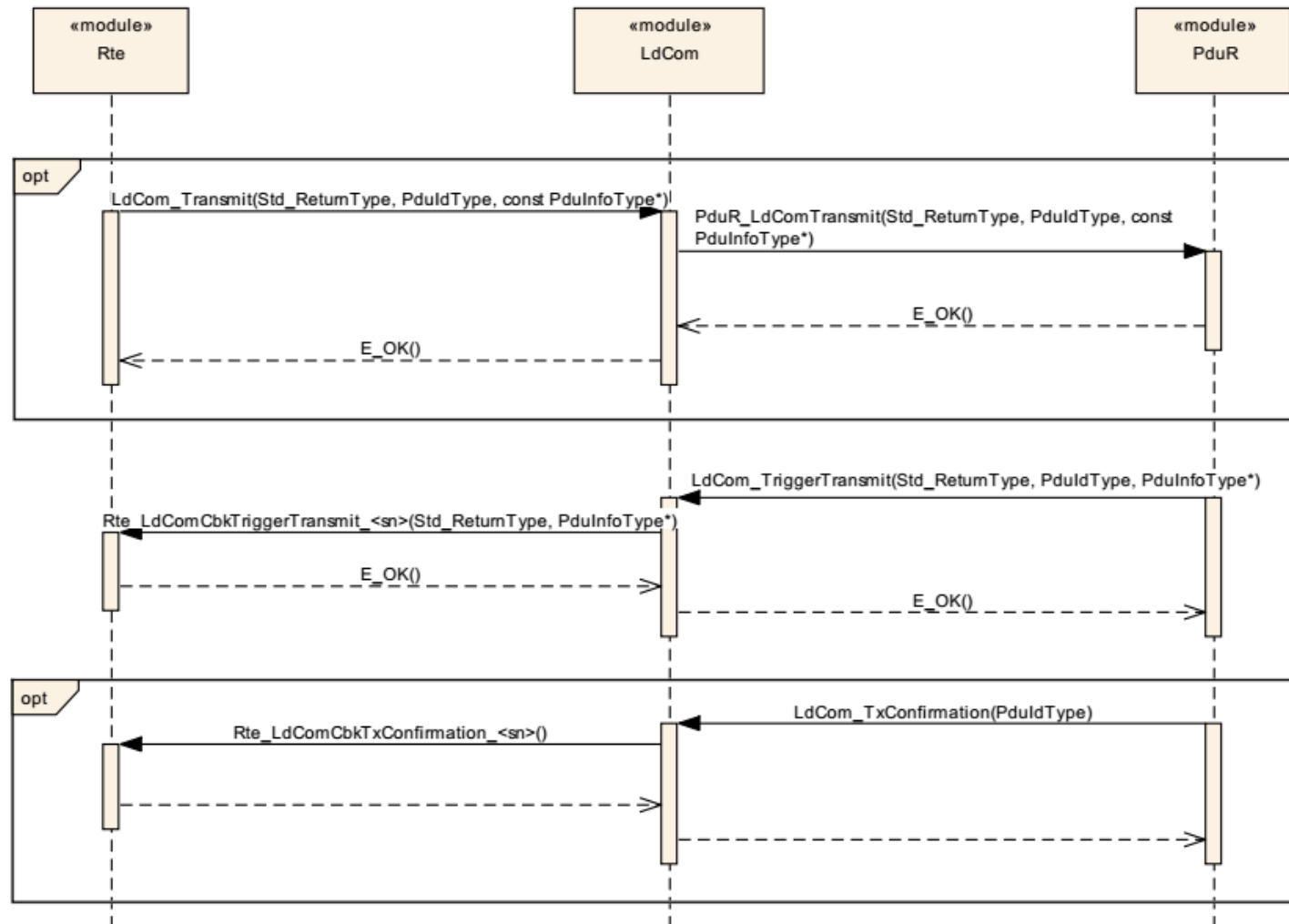
6.2 LdCom发送——TP API



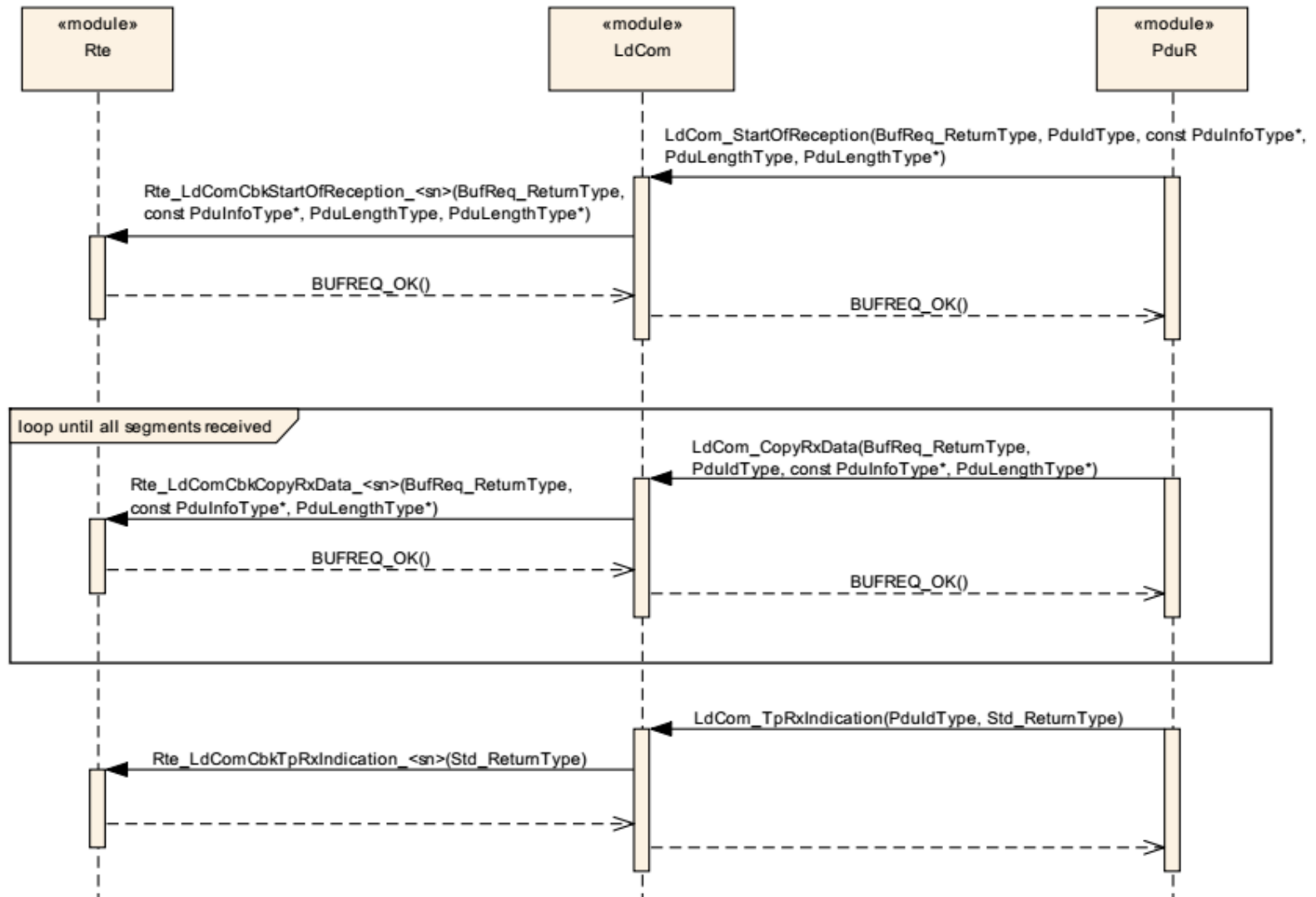
6.2 LdCom发送——IF API



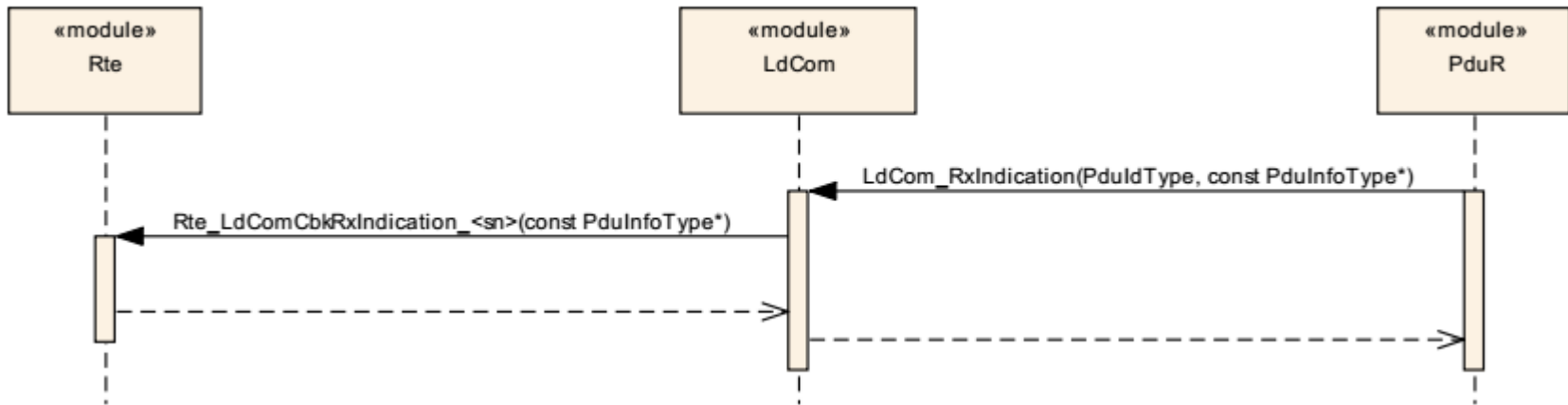
6.2 LdCom发送——Trigger Transmit



6.3 LdCom接收——TP API



6.3 LdCom接收——IF API



感谢关注!

责任 创新 卓越 共享



网址: www.i-soft.com.cn

信箱: Marketing@i-soft.com.cn

热线: 400-650-9325