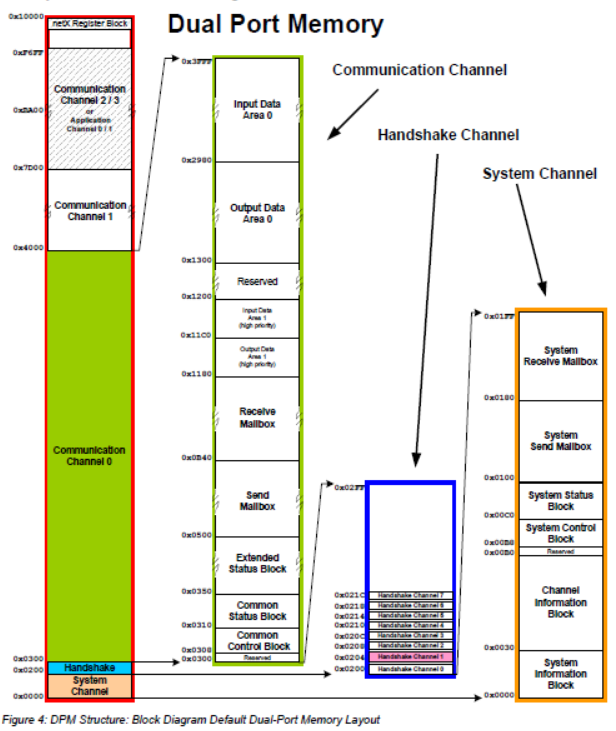
**操作双端口内存 - 关于非周期报文**

文档：netX Dual-Port Memory Interface DPM 17 EN.pdf

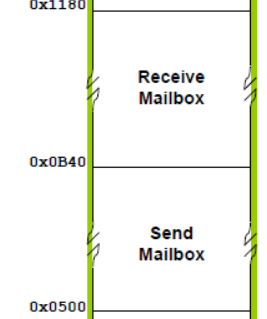
**1. 如何在HOST CPU上实现非周期数据的读写操作。**

（如果HOST CPU上的驱动是直接操作双端口内存DPM，而非移植所提供的Toolkit代码。）

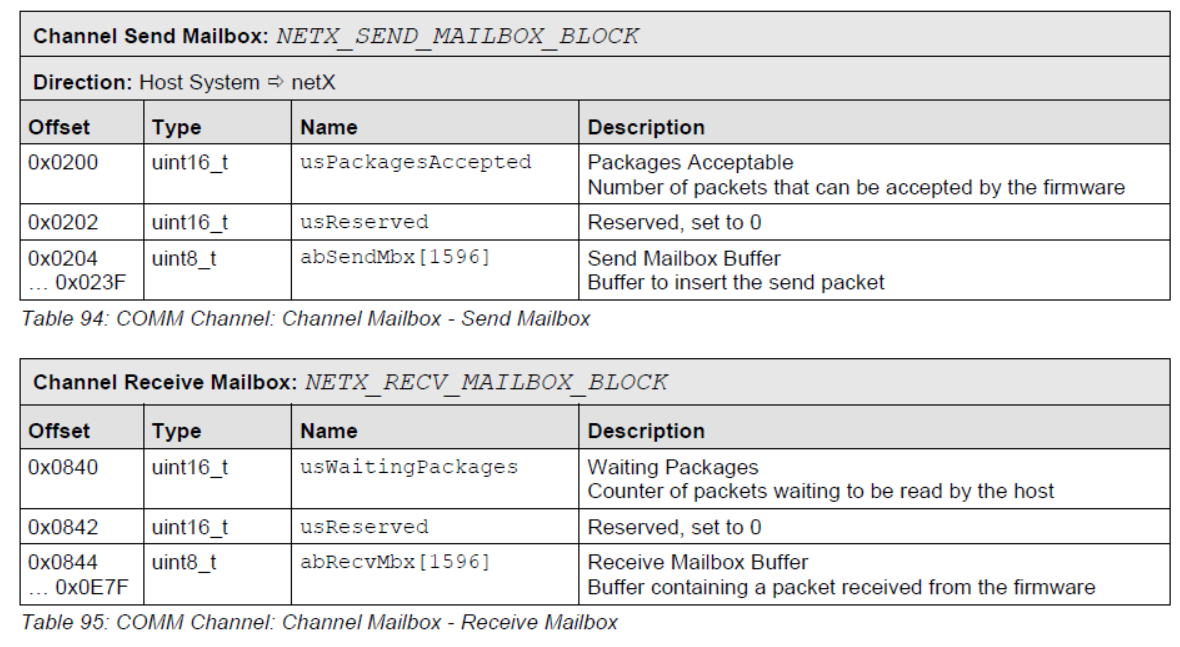
首先，需要了解双端口内存DPM的结构。netX Dual-Port Memory Interface DPM 17 EN.pdf文档。



如果是实现**通讯通道**非周期报文的操作，则主要看send mailbox和receive mailbox。偏移量地址：0x0500和0x0B40。



详细，第72页，

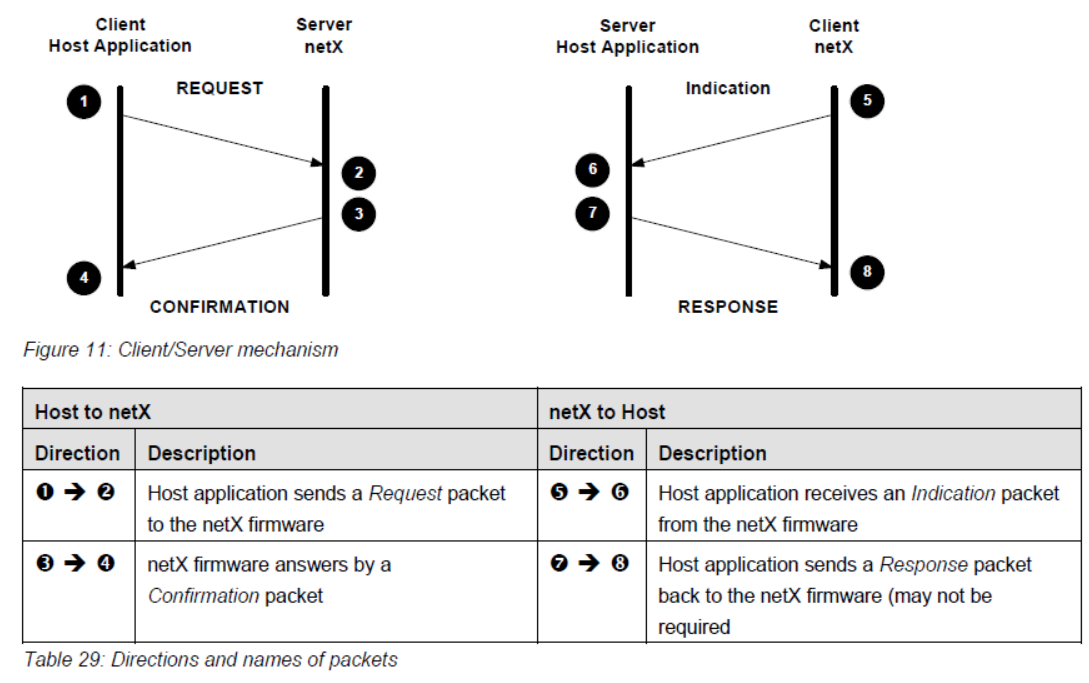


这里（上面截图）的偏移地址对应到整个双端口内存的偏移地址需要再加上0x0300。

发送非周期报文，则是发送到0x204(即0x504)。

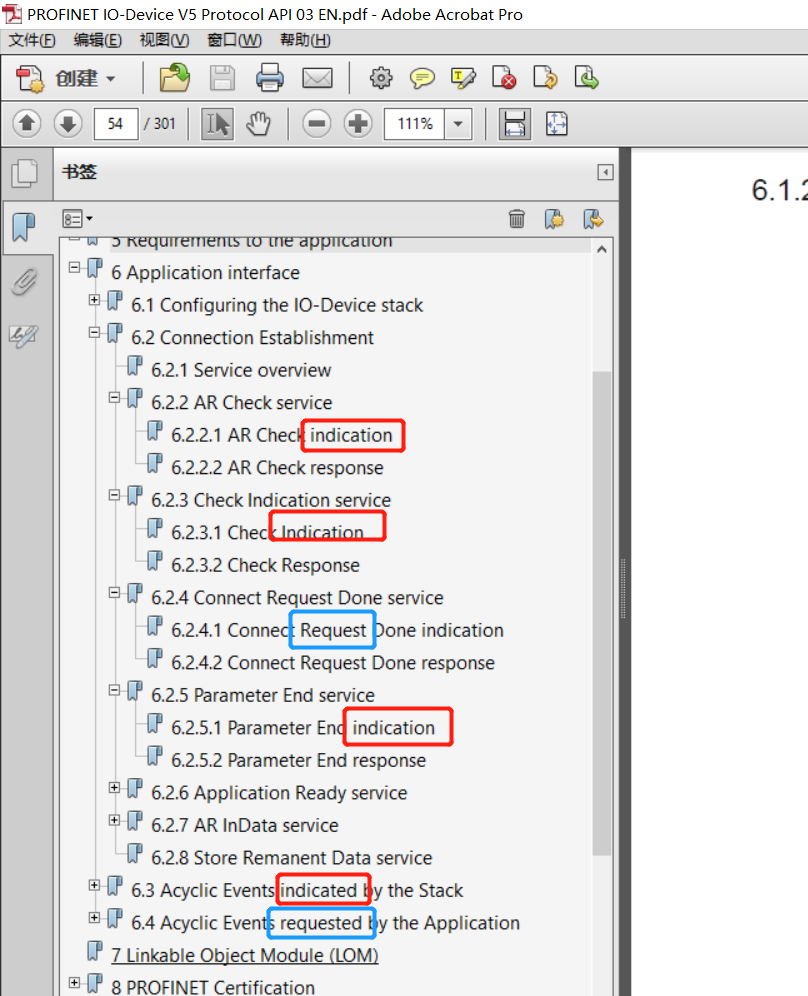
每个报文的结构，则需要先了解一下，Chapter 4.1 Non-Cyclic Data Transfer via Mailbox and Packets。

1. 首先，**非周期的报文有2种**：一种是**REQ/CNF**，MCU主动发报文给协议栈Firmware（先发报文给send mailbox，再读receive mailbox，看是否发送报文正确），去获取报文的；一种是**IND/RES**（协议栈Firmware会在receive mailbox中产生报文，MCU的应用程序要去读，然后再发送RES报文给send mailbox，通知协议栈报文已经读走或者将协议栈需要的信息反馈回），协议栈主动发送报文给MCU的，一般是 对方设备（主站或者从站）发给协议栈或者是网络状态，协议栈通知 MCU 的。

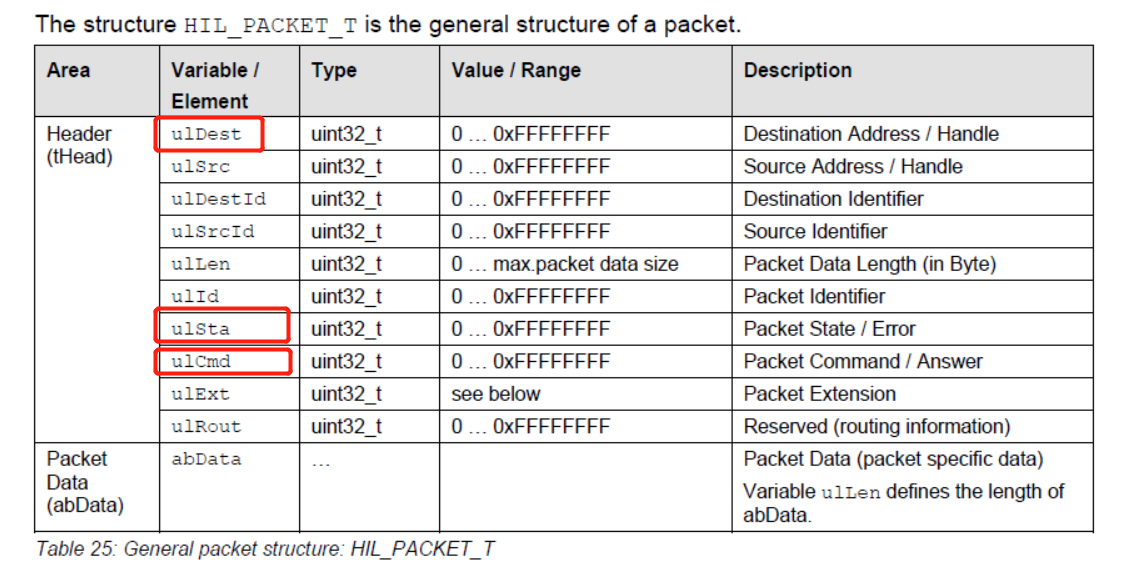


在每个 API 文档都会有关于 **REQ/CNF** 和**IND/RES**的描述。

比如：Profinet从站的API 文档，如下，**request**报文就是**REQ/CNF** ；Indication报文就是**IND/RES**。



1. 非周期报文的格式，



主要的三个参数：

ulDest： 通讯通道是:0x00000020; 系统通道是:0x00000000。

ulSta: 报文的状态和错误代码。

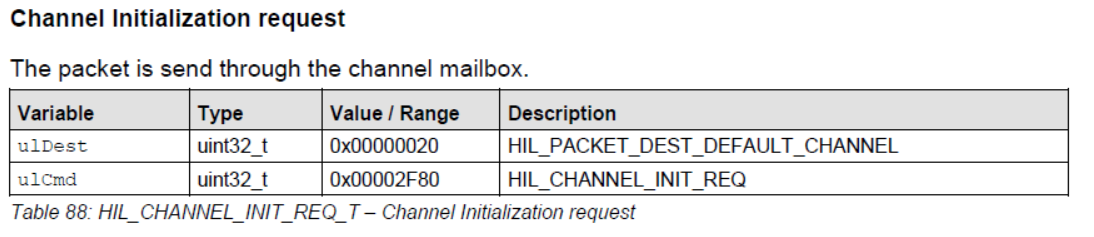
ulCmd: 每个报文的命令是不一样的。

1. **REQ/CNF** 报文， MCU 每次主动发request报文给协议栈（PutPacket/send mailbox）之后，都要再去读一下Confirm 报文（GetPacket/ receive mailbox）,判断 ulCmd是否+1，ulSta是否为0。如果ulSta是“非0“，那么request的报文就是有问题的，根据ulSta的值，错误代码，查看具体的解释。有些错误代码在 API文档中能找到，有些在文档 Hilscher status and error codes API 05 EN.pdf中能找到。
2. **IND/RES**报文，需要MCU 在一开始的时候 发送了命令0x2F10，Register Application ，MCU才会收到Indication 报文。 针对不同的协议，有些Indication报文还需要专门的注册register，需要看 协议的API文档。

**举例1**，

基于netX90，（如果是其它的netX产品，可以看相应的文档）

netX Dual-Port Memory packet-based services netX 90 4000 4100 API 05 EN.pdf文档中的Chapter 4.5 Channel Initialization通讯通道初始化命令。



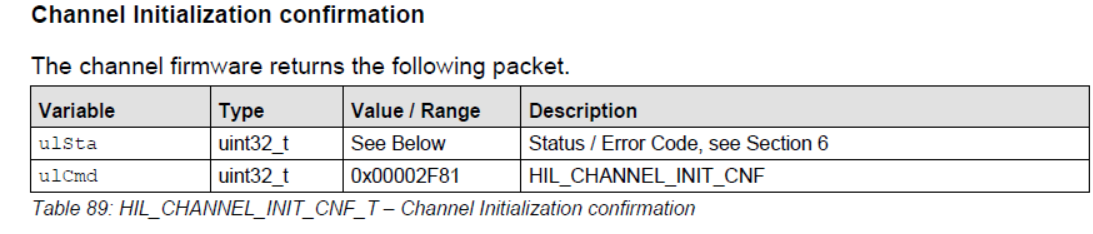
每个报文的命令ulCmd 是不一样的。

ulDest 一般都是0x00000020（通讯通道）。

ulSrc, ulDestId，ulSrcId，可以都是0。

ilId也是0。

HOST CPU发送完这个命令之后（send mailbox），还需要去读Receive mailbox，看返回的报文是否是ulCmd+1, ulSta = 0x00000000。



注意：在操作mailbox的时候，要先判断握手。