

文档编号:

受控状态:

发 放 号:

AUTBUS 行业应用解决方案

板级通信应用方案

【V1.1】

编 制: _____

校 对: _____

审 核: _____

批 准: _____

北京神经元网络科技有限公司

Beijing Neuron Network Technology Co., LTD

修订历史记录

日期	版本	说明	作者
2024.04.11	V1.0	初始版本	肖磊
2024.11.06	V1.1	1. 更新 SPI 从模式对接方案 2. 增加双路 SPI 对接方案 3. 更新 AUTBUS 通信协议 4. 增加 AUTBUS 到 Host 心跳服务 5. 增加 AUTBUS Host 固件升级服务	单小明

Confidential-51y

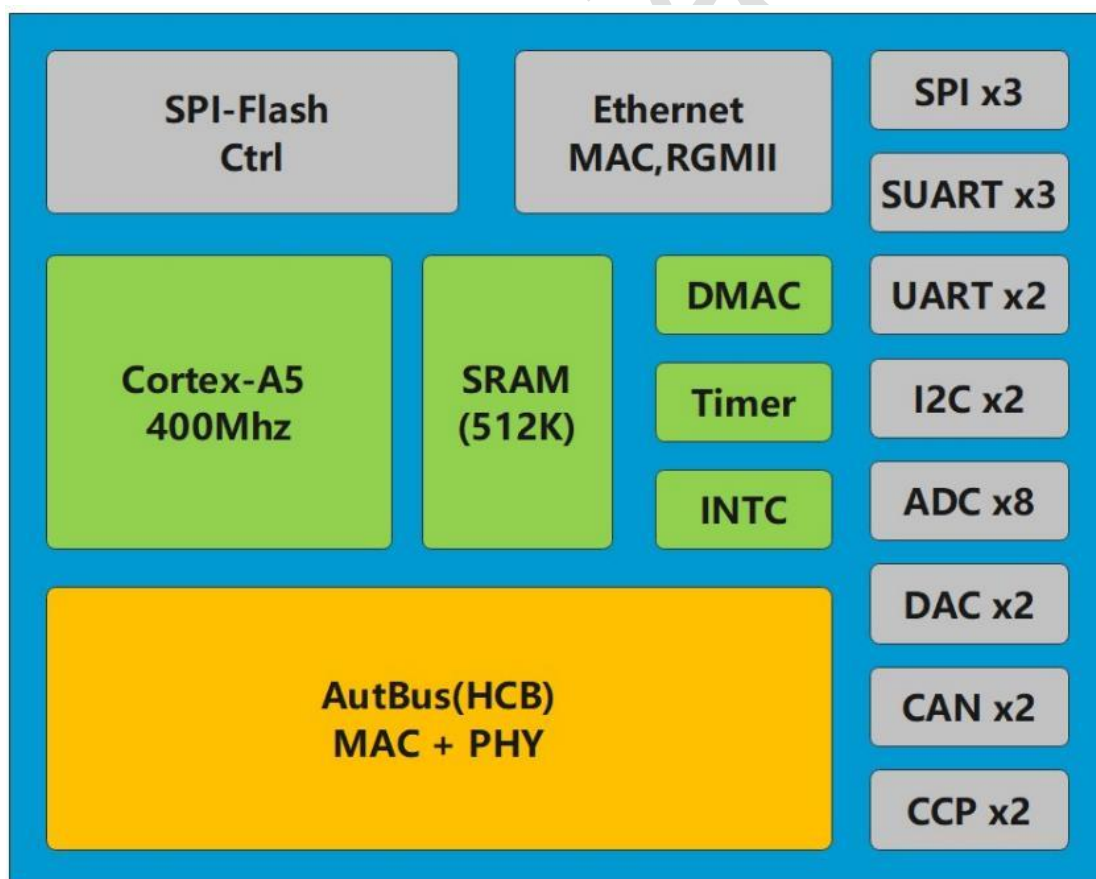
目录

一、概述.....	4
二、应用方案综述.....	5
三、板级集成方案.....	6
1. 硬件接口概述.....	6
2. 软件对接概述.....	11
3. 基础管理 IO 对接.....	11
4. SPI 从模式对接 Host.....	12
5. SPI 主模式对接 Host.....	20
6. 双路 SPI 对接 Host.....	25
7. RGMII 对接 Host.....	30
8. AUTBUS 通信协议.....	34
9. AUTBUS 系统配置服务.....	42
10. AUTBUS 状态管理服务.....	44
11. AUTBUS 网络时间同步服务.....	45
12. AUTBUS 到 Host 心跳服务.....	46
13. AUTBUS Host 固件升级服务.....	48
四、主要流程简介.....	51
1. 启动交互流程.....	51
2. AUTBUS 内部网络状态机流程.....	52
3. Host 与 AUTBUS 系统常用交互流程.....	53

一、概述

AUTBUS 是一种采用两线非桥接媒介的工业/行业现场总线；其在现场总线物理层的传输方案上，创新性的采用了基于 OFDM 技术，卷积+RS 级联码编码技术，高精度全网同步技术，将三者融合形成的时间敏感现场总线网络传输方案，其技术方案的核心部分已经申请国家标准并于 2022 年正式颁布（GB/T 42019-2022），对应的国际标准也于 2023 年正式颁布（IEC61158-type28）。

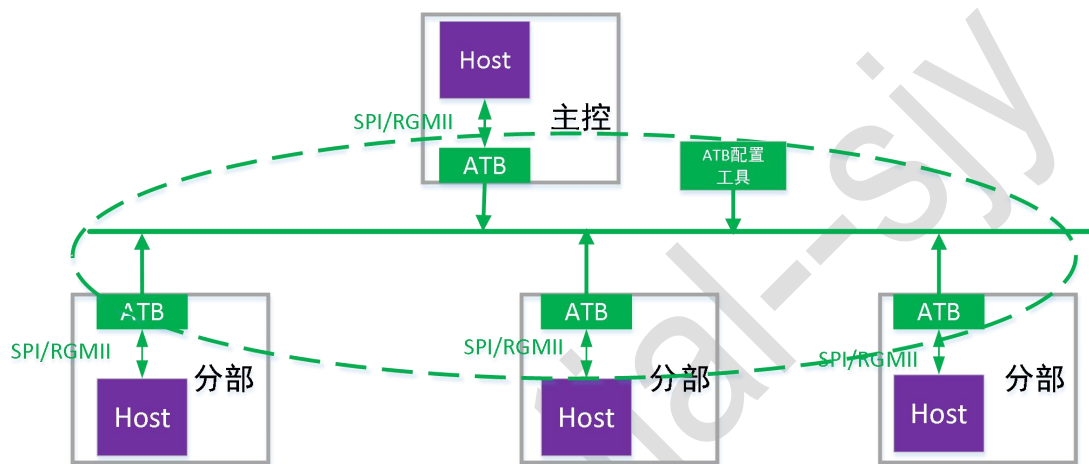
KY3001 芯片是一款以 AUTBUS 为核心 IP 的 SOC 芯片，其集控制和通信为一体，除了 AUTBUS 之外，还集成了 ARM A5 的 IP 核，还包含有多种类常用的外设 IP，如 ETH，CAN，UART，AD，PWM 等（如下图）；此外，芯片支持片外 flash（Nor 和 Nand 的主流类型均支持），内存 512KB，内部有硬件 Timer，DMA 等功能可供使用；



KY3001 芯片也配套 SDK 软件，除了提供基础的通信协议功能之外，还提供一个供客户二次开发的平台，供用户开发。

二、应用方案综述

AUTBUS 作为一个独特的现场层带宽总线技术，广泛的应用在智能工厂、工业机器人、船舶、轨道、智慧交通、石油化工、清洁能源、汽车电子和工程装备，军工等领域，为各个领域提供包括 AUTBUS 通信为核心的应用方案，其方案总体架构如下图：



如上图所示，此类应用方案是采用 AUTBUS 芯片在各个业务设备内集成的方式，其交付方式为固件+芯片；

上图中，Host 端（业务主控芯片）为用户的业务模块，AUTBUS 芯片端为通信模块，其各个 AUTBUS 的通信模块按照组网要求构成总线网络系统（绿色部分），此系统提供各个对应接入的业务模块的通信类服务，和应用扩展订制类服务（可选）。

其中，AUTBUS 芯片端与 Host 端在通信侧的对接方式有 SPI 或者 RGMII 两种方式（其中 SPI 也分为主模式对接和从模式对接两种）；

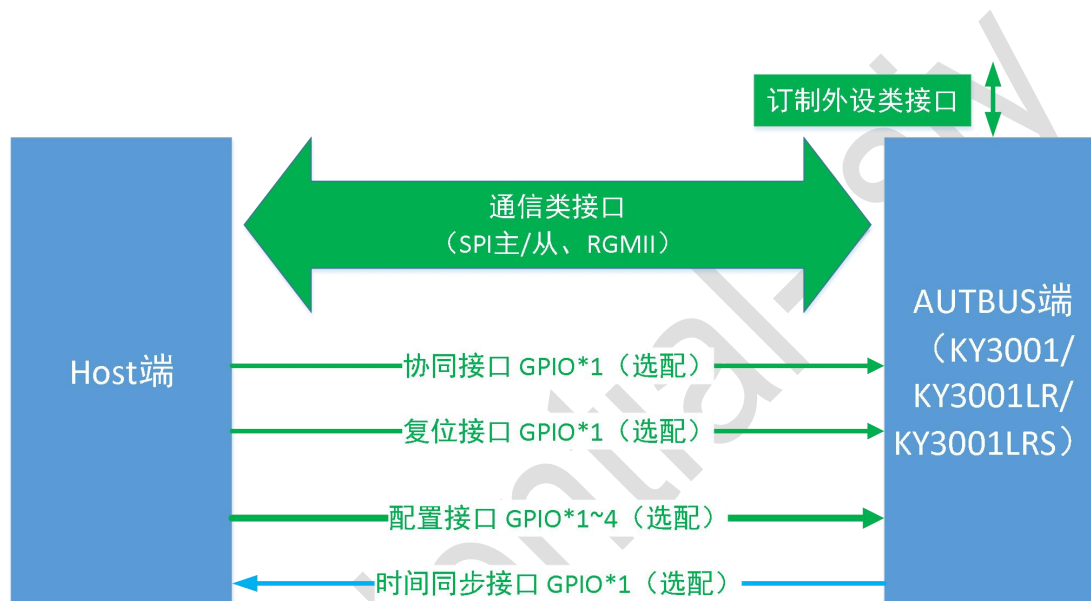
基于此，在方案中，AUTBUS 芯片端对外提供：

- （1）硬件各类接口和原理图参考设计；
- （2）通用级外设接口驱动和定义（二次开发或订制使用）；
- （3）方案级配置管理接口（复位 IO，配置 IO，协同 IO 等）机制和定义；
- （4）方案级通信接口（SPI 主/从模式，RGMII）驱动层交互机制；
- （5）AUTBUS 通信协议；
- （6）AUTBUS 系统状态机制；

- (7) 应用订制类服务机制；
- (8) AUTBUS 网络系统配置监看工具（网络配置，调试，运维）；

三、板级集成方案

1. 硬件接口概述



AUTBUS 端主要芯片型号有：KY3001，KY3001LR，LY3001LRS，不同型号芯片主要是尺寸不同，和封出来的引脚不同（功能范围不一样，详见 KY3001 芯片手册），所以在选择不同芯片的时候，其对应的接口方案选择会有一定约束，后续有具体详述；另外，对于 GPIO 类引脚的约定，此方案文档上不规定具体的引脚号，交由具体项目设计上根据总体引脚的规划和需求自行约定即可。

复位接口：此类接口为选配接口，对应 KY3001 系列芯片的复位引脚（RESET_IN），默认高位，低有效；一种使用方式是，将其做在板级复位方案上（配合板级看门狗机制联动之类），另一种方式是让 Host 端直接操作，管控 AUTBUS 的复位（先拉低，芯片进入复位态，后拉高，使其解复位）。

协同接口：此类接口为选配接口，使用芯片任意单个 GPIO 引脚（可约定），默认高位；其用于 Host 端与 AUTBUS 端的流程协同控制，由 Host 端发起，AUTBUS 端感知执行，一般被用来做启动流程顺序的控制；

配置接口：此类接口为选配接口，使用芯片任意一个或多个 GPIO 引脚（可

约定），默认高位；用于 Host 端对 AUTBUS 端固件的功能配置，比如对接通信接口的类型和配置，CN/TN 角色配置，用户订制的固件功能等；

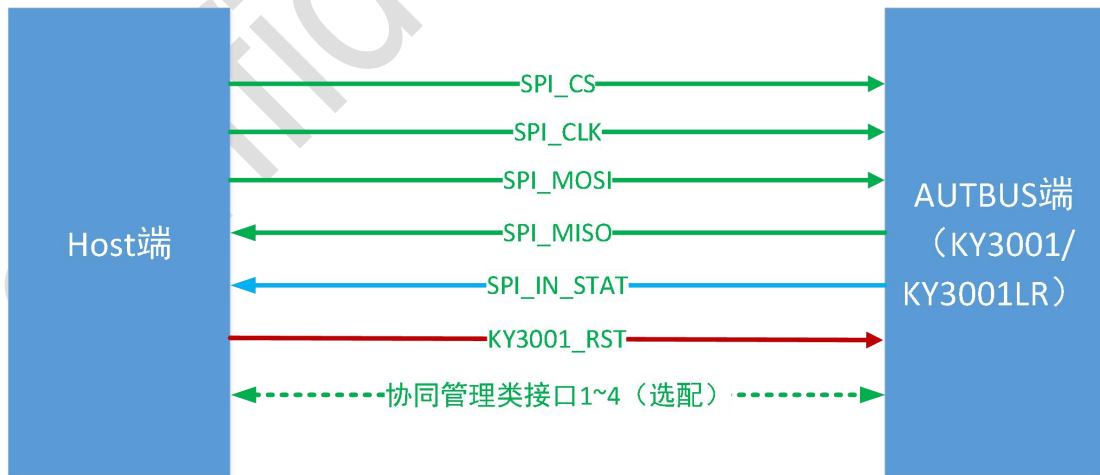
时间同步接口：此类接口为选配接口，使用芯片任意单个 GPIO 引脚（可约定），默认高位；这个接口是唯一由 AUTBUS 端对 Host 端进行单向输出的接口，其用于 AUTBUS 端向 Host 端提供实时的时间同步信号，由 AUTBUS 端周期性的发起，Host 端感知执行（具体详见“AUTBUS 网络时间同步服务”）；

订制外设接口：此类接口为选配接口，KY3001 系列芯片本身有丰富的外设接口可供订制使用（详见 KY3001 芯片手册）；通常意义上 AUTBUS 芯片端仅做通信功能使用，如果在设备板级方案上需要额外应用到芯片的外设功能，可订制对应的功能实现，并可结合其应用协议来配合使用；

通信接口：此类接口为方案必选接口，具体有三类，即 SPI 主模式接口，SPI 从模式接口，RGMII 接口；对于各个通信接口定义分别如下：

（1） SPI 从模式接口定义

SPI 从模式下，使用的是 KY3001 芯片系列的 SPI-1 接口（详见 KY3001 芯片手册），除了 SPI 的标准接口定义外，有一个额外的 SPI_IN_STAT 定义（此接口可选），用作通信状态的告知（有无 MISO 需求），一个 KY3001 复位引脚以及预留 1~4 个 GPIO 作为协同管理接口；

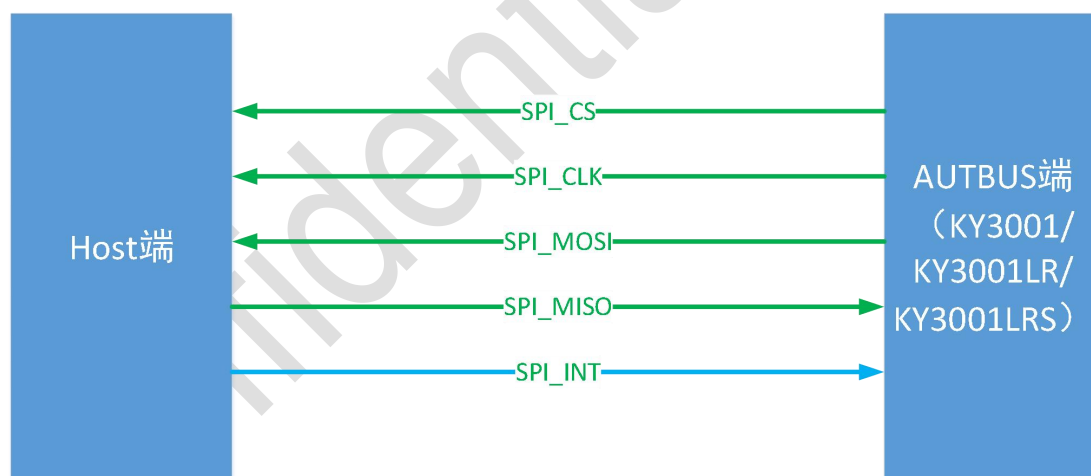


序号	信号名称	信号定义	KY3001 信号名称	方向	有效状态	空闲状态
1	SPI_CS	SPI 接口片选信号	ETH0_RXD3/SYS_GPIO36/S PI1_CS_B	从端输入	低电平	高电平

2	SPI_CLK	SPI 接口时钟信号	ETH0_RXD2/ SYS_GPIO37/SPI1_CLK	从端输入	低电平	高电平
3	SPI_MISO	SPI 接口从端数据 输出信号	ETH0_RXD1/SYS_GPIO38/S PI1_TX	从端输出	高电平/ 低电平	高电平
4	SPI_MOSI	SPI 接口从端数据 输入信号	ETH0_RXD0/SYS_GPIO39/S PI1_RX	从端输入	高电平/ 低电平	高电平
5	SPI_IN_STAT	SPI 接口收包指示 信号(可选)	CCP1_PWM3_P/SYS_GPIO1 2/CCP1_CAP3	从端输出	低电平	高电平
6	KY3001_RST	KY3001 复位信号	RESET_IN	从端输入	低电平	高电平
7	GPIO_RSV	协同管理类预留 GPIO (可选配)	按需要选配 1~4 个 GPIO	输入/ 输出	高电平/ 低电平	高电平/ 低电平

(2) SPI 主模式接口定义

SPI 主模式下,使用的是 KY3001 芯片系列的 SPI-0/2 接口(详见 KY3001 芯片手册),除了 SPI 的标准接口定义外,有一个额外的 SPI_IN_STAT 定义,是当 Host 有 MISO 需求时,对 AUTBUS 端发送的中断信号;

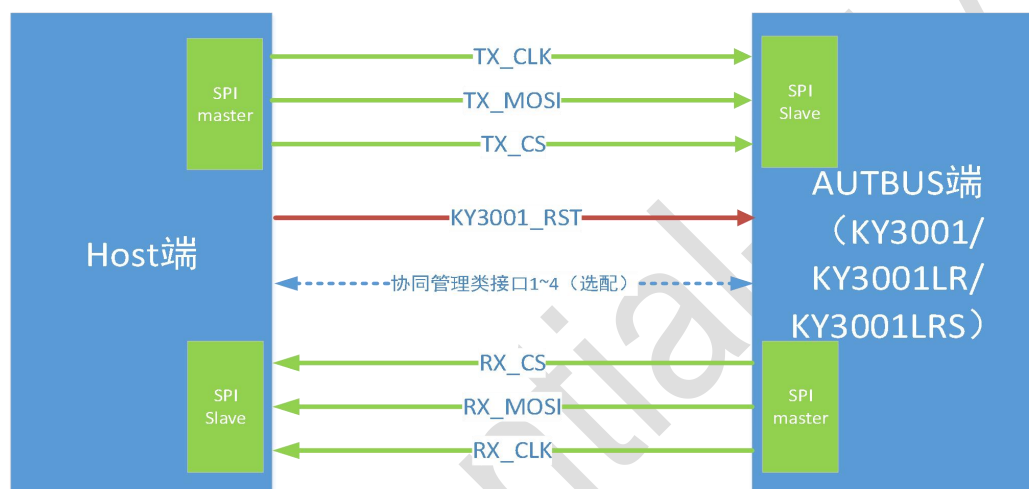


序号	信号名称	信号定义	方向	有效状态	空闲状态
1	SPI_CS	SPI 接口片选信号	从端输入	低电平	高电平
2	SPI_CLK	SPI 接口时钟信号	从端输入	低电平	高电平
3	SPI_MISO	SPI 接口从端数据输出信号	从端输出	高电平/低电平	高电平
4	SPI_MOSI	SPI 接口从端数据输入信号	从端输入	高电平/低电平	高电平

5	SPI_INT	SPI 接口中断信号	从端输出	下降沿	高电平
---	---------	------------	------	-----	-----

(3) 双路 SPI 接口定义

为了提升在 SPI 模式下对接通信效率，以 KY3001 系列芯片为基础，提出双路 SPI 通信对接的方案；此方案构造出一个双向异步的通信模式，即一路 SPI 专门发送到网络侧的数据，另一路 SPI 专门接收从网络侧过来的数据，收发流量间可做到互不影响；



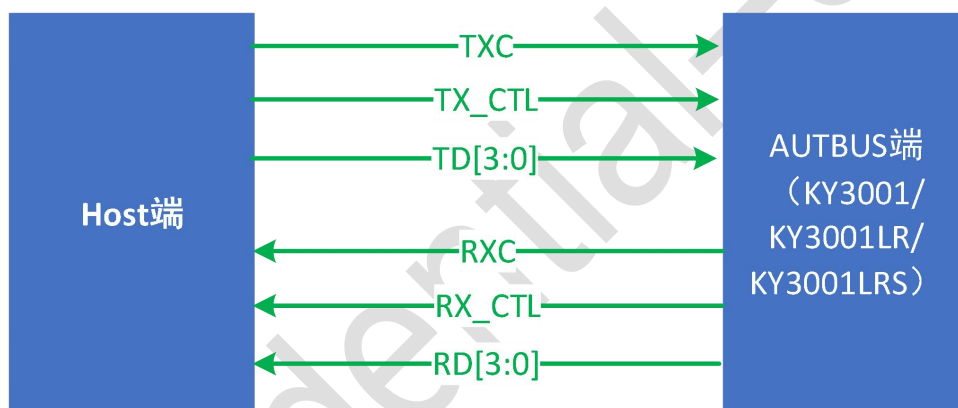
如上图所示，TX 方向，Host 端 SPI 为主，KY3001 为从，使用 CLK，MOSI，CS 信号线即可；RX 方向，也相同配置；

序号	信号名称	信号定义	KY3001 信号名称	方向	有效状态	空闲状态
1	SPI_TX_CS	SPI_TX 接口片选信号	ETH0_RXD3/SYS_GPIO36/ SPI1_CS_B	网络输入	低电平	高电平
2	SPI_TX_CLK	SPI_TX 接口时钟信号	ETH0_RXD2/ SYS_GPIO37/SPI1_CLK	网络输入	低电平	高电平
3	SPI_TX_MOSI	SPI_TX 接口数据输入信号	ETH0_RXD0/SYS_GPIO39/ SPI1_RX	网络输入	高电平/ 低电平	高电平
4	SPI_RX_CS	SPI_RX 接口片选信号	SPI0_CS0_B/SYS_GPIO58/ SPI2_CS3_B	网络输出	低电平	高电平
5	SPI_RX_CLK	SPI_RX 接口时钟信号	SPI0_CLK/SYS_GPIO52/ED M_TRST	网络输出	低电平	高电平
6	SPI_RX_MOSI	SPI_RX 接口数据输出信号	SPI0_TX/SYS_GPIO53/ED M_TMS	网络输出	高电平/ 低电平	高电平

7	SYS_STAT_N	系统协同管理接口信号 (选配参考,可扩展,1~4路 GPIO)	ETH0_RXD1/SYS_GPIO38/ SPI1_TX SPI0_RX/SYS_GPIO54/ED M_TDO CCP1_PWM3_P/SYS_GPIO12/CCP1_CAP3 CCP1_PWM3_N/SYS_GPIO13/SUART1_DOUT	系统间的 协同管理	高电平/ 低电平	高电平
8	KY3001_RST	KY3001 复位信号	RESET_IN	网络输入	低电平	高电平

(4) ETH-RGMII 接口定义

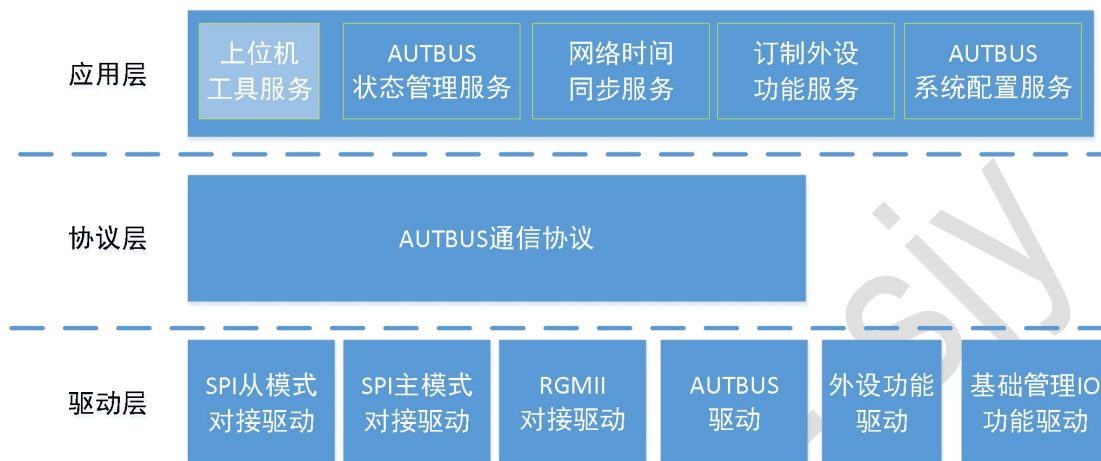
ETH-RGMII 接口下，使用的是 KY3001 芯片系列的 ETH0 接口（详见 KY3001 芯片手册）；



序号	信号名称	信号定义	方向	有效状态	空闲状态
1	TXC	HOST 发送时钟, 25M	HOST 端输出	高电平/低电平	高电平/低电平
2	TX_CTL	HOST 发送控制信号	HOST 端输出	高电平/低电平	低电平
3	TD[3:0]	HOST 发送数据	HOST 端输出	高电平/低电平	低电平
4	RXC	HOST 接收时钟, 25M	KY3001 端输入	高电平/低电平	高电平/低电平
5	RX_CTL	HOST 接收控制信号	KY3001 端输入	高电平/低电平	低电平
6	RD[3:0]	HOST 接收数据	KY3001 端输入	高电平/低电平	低电平

2. 软件对接概述

AUTBUS 端和 Host 端的软件对接，按照 AUTBUS 固件内部的软件架构，分为如下三个层级分别描述，即驱动层，协议层，应用层；如下图所示：



驱动层上，AUTBUS 芯片端除了提供芯片基础管理 IO 功能驱动之外，根据业务通信接口不同，会有不同的对接驱动（SPI 从/主，RGMII）；此外，还可能有用用户对于外设驱动功能驱动的订制；

协议层上，AUTBUS 通信协议，其具体提供：统一的帧格式和各字段的约定，按照节点 ID 寻址的通信方式，固件内服务的对接方式；

应用层上，在 AUTBUS 芯片端，除了上位机工具服务是专门对接 AUTBUS 上位机工具软件的之外，为 Host 端可提供的服务有：AUTBUS 系统配置服务，外设功能服务，网络时间同步服务，AUTBUS 状态管理服务；

具体的详见后续各个模块描述；

3. 基础管理 IO 对接

AUTBUS 端提供基础管理类 IO（GPIO 方式），供 Host 来配置和感知，其中就有：

（1）复位接口

此类接口为选配接口，对应 KY3001 系列芯片的复位引脚（RESET_IN），默认高位，低有效；一种使用方式是，将其做在板级复位方案上（配合板级看门

狗机制联动之类），另一种方式是让 Host 端直接操作，管控 AUTBUS 的复位。

复位操作为先拉低，使其进入复位态，然后拉高，解复位。

（2）协同接口

此类接口为选配接口，使用芯片任意单个 GPIO 引脚（可约定），默认高位；其用于 Host 端与 AUTBUS 端的流程协同控制，由 Host 端发起，AUTBUS 端感知执行，一般被用来做启动流程顺序的控制；

配置接口：此类接口为选配接口，使用芯片任意一个或多个 GPIO 引脚（可约定），默认高位；用于 Host 端对 AUTBUS 端固件的功能配置，比如对接通信接口的类型和配置，CN/TN 角色配置，用户订制的固件功能等；

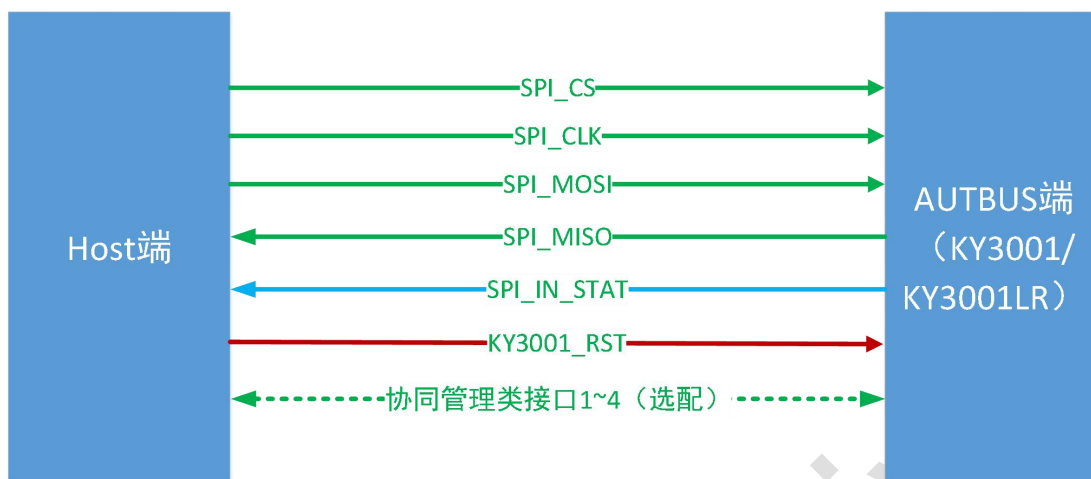
时间同步接口：此类接口为选配接口，使用芯片任意单个 GPIO 引脚（可约定），默认高位；这个接口是唯一由 AUTBUS 端对 Host 端进行单向输出的接口，其用于 AUTBUS 端向 Host 端提供实时的时间同步信号，由 AUTBUS 端周期性的发起，Host 端感知执行（具体详见“AUTBUS 网络时间同步服务”）；

订制外设接口：此类接口为选配接口，KY3001 系列芯片本身有丰富的外设接口可供订制使用（详见 KY3001 芯片手册）；通常意义上 AUTBUS 芯片端仅做通信功能使用，如果在设备板级方案上需要额外应用到芯片的外设功能，可订制对应的功能实现，并可结合其应用协议来配合使用；

4.SPI 从模式对接 Host

（1）接口说明

SPI 从模式下，使用的是 KY3001 芯片系列的 SPI-1 接口（详见 KY3001 芯片手册），除了 SPI 的标准接口定义外，有一个额外的 SPI_IN_STAT 定义（此接口可选），用作通信状态的告知（有无 MISO 需求），KY3001 复位引脚以及预留 1~4 个 GPIO 作为协同管理接口；



序号	信号名称	信号定义	KY3001 信号名称	方向	有效状态	空闲状态
1	SPI_CS	SPI 接口片选信号	ETH0_RXD3/SYS_GPIO36/ SPI1_CS_B	从端输入	低电平	高电平
2	SPI_CLK	SPI 接口时钟信号	ETH0_RXD2/ SYS_GPIO37/SPI1_CLK	从端输入	低电平	高电平
3	SPI_MISO	SPI 接口从端数据输出信号	ETH0_RXD1/SYS_GPIO38/ SPI1_TX	从端输出	高电平/ 低电平	高电平
4	SPI_MOSI	SPI 接口从端数据输入信号	ETH0_RXD0/SYS_GPIO39/ SPI1_RX	从端输入	高电平/ 低电平	高电平
5	SPI_IN_STAT	SPI 接口收包指示信号(可选)	CCP1_PWM3_P/SYS_GPIO 12/CCP1_CAP3	从端输出	低电平	高电平
6	KY3001_RST	KY3001 复位信号	RESET_IN	从端输入	低电平	高电平
7	GPIO_RSV	协同管理类预留 GPIO(可选配)	按需要选配 1~4 个 GPIO	输入/ 输出	高电平/ 低电平	高电平/ 低电平

SPI 通信接口共包含 5 根信号线，其中包括 SPI 4 根线和 1 根 GPIO 作为收包指示信号线；

GPIO 作为收包指示信号线，当 slave 端有包缓存待 host 接收时拉低，直到缓存读空时拉高；

GPIO_RSV 为协同管理类预留 GPIO 接口，建议预留 1~4 个 GPIO；

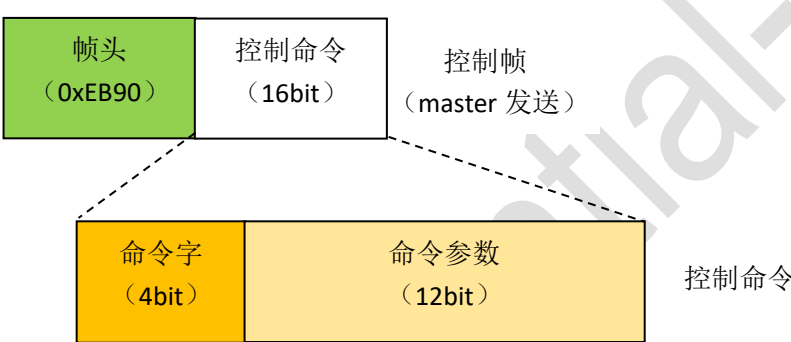
KY3001_RST 为 KY3001 芯片复位信号，用于 Host 复位 KY3001。

(2) SPI 物理层参数

SPI 物理层协议：摩托罗拉协议；
波特率：<=12.5Mbps；
字长：8bit；
极性配置：CPOL:1，CPHA:1；

(3) SPI 通信帧格式

控制帧格式



1) 控制帧用于对 slave 的操作控制和状态读取，下表为控制帧格式以及字段的描述；

SPI 通讯控制帧格式

字节偏移	参数名称	字节数	数据类型	备注
0~1	帧头	2Byte 小端	UINT16	0xEB90
2~3	控制命令	2Byte 小端	UINT16	见下表

SPI 通讯控制命令

命令字 (4bit)	命令含义	命令参数(12bit)
0x0	保留	保留
0x1	数据发送命令	12bit 发送数据帧长度
0x2	数据接收命令	12bit 最大接收数据帧长度

数据帧格式

帧头 (0xEB90)	帧长度 (16bit)	Payload 数据	CRC (16bit)	数据帧
----------------	----------------	---------------	----------------	-----

- 1) 业务层接收数据：驱动层接收 SPI 数据，处理后将数据内容发送至业务层。
- 2) 业务层发送数据：业务层将业务 payload 发送给驱动层，驱动层添加帧头，帧长度和 CRC 校验（可选），通过 SPI 发送。

SPI 通讯数据帧格式

字节偏移	参数名称	字节数	数据类型	备注
0~1	数据帧头	2Byte 小端	UINT16	0xEB90
2~3	帧长度	2Byte 小端	UINT16	N-6（业务 Payload 字节数）
4~N-3	消息负载	N-6	UINT08	——
N-2~N-1	CRC 校验（可选）	2Byte 小端	UINT16	2~N-3 字节的 CRC16 校验
注：N ∈ [10, 1024]				

CRC16 约定

CRC 校验多项式为： $CRC = X^{16} + X^{12} + X^5 + 1$ ，初始值 0x0000，结果异或值 0x0000。具体码表如下：

0x0000	0x1021	0x2042	0x3063	0x4084	0x50A5	0x60C6	0x70E7	0x8108
0x9129								
0xA14A	0xB16B	0xC18C	0xD1AD	0xE1CE	0xF1EF	0x1231	0x0210	0x3273
0x2252								
0x52B5	0x4294	0x72F7	0x62D6	0x9339	0x8318	0xB37B	0xA35A	0xD3BD
0xC39C								
0xF3FF	0xE3DE	0x2462	0x3443	0x0420	0x1401	0x64E6	0x74C7	0x44A4
0x5485								
0xA56A	0xB54B	0x8528	0x9509	0xE5EE	0xF5CF	0xC5AC	0xD58D	0x3653
0x2672								
0x1611	0x0630	0x76D7	0x66F6	0x5695	0x46B4	0xB75B	0xA77A	0x9719
0x8738								
0xF7DF	0xE7FE	0xD79D	0xC7BC	0x48C4	0x58E5	0x6886	0x78A7	0x0840

0x1861
0x2802 0x3823 0xC9CC 0xD9ED 0xE98E 0xF9AF 0x8948 0x9969 0xA90A
0xB92B
0x5AF5 0x4AD4 0x7AB7 0x6A96 0x1A71 0x0A50 0x3A33 0x2A12 0xDBFD
0xCBDC
0xFBFF 0xEB9E 0x9B79 0x8B58 0xBB3B 0xAB1A 0x6CA6 0x7C87 0x4CE4
0x5CC5
0x2C22 0x3C03 0x0C60 0x1C41 0xEDAE 0xFD8F 0xCDEC 0xDDCD 0xAD2A
0xBD0B
0x8D68 0x9D49 0x7E97 0x6EB6 0x5ED5 0x4EF4 0x3E13 0x2E32 0x1E51
0x0E70
0xFF9F 0xEFBE 0xDFDD 0xCFFC 0xBF1B 0xAF3A 0x9F59 0x8F78 0x9188
0x81A9
0xB1CA 0xA1EB 0xD10C 0xC12D 0xF14E 0xE16F 0x1080 0x00A1 0x30C2
0x20E3
0x5004 0x4025 0x7046 0x6067 0x83B9 0x9398 0xA3FB 0xB3DA 0xC33D
0xD31C
0xE37F 0xF35E 0x02B1 0x1290 0x22F3 0x32D2 0x4235 0x5214 0x6277
0x7256
0xB5EA 0xA5CB 0x95A8 0x8589 0xF56E 0xE54F 0xD52C 0xC50D 0x34E2
0x24C3
0x14A0 0x0481 0x7466 0x6447 0x5424 0x4405 0xA7DB 0xB7FA 0x8799
0x97B8
0xE75F 0xF77E 0xC71D 0xD73C 0x26D3 0x36F2 0x0691 0x16B0 0x6657
0x7676
0x4615 0x5634 0xD94C 0xC96D 0xF90E 0xE92F 0x99C8 0x89E9 0xB98A
0xA9AB
0x5844 0x4865 0x7806 0x6827 0x18C0 0x08E1 0x3882 0x28A3 0xCB7D
0xDB5C
0xEB3F 0xFB1E 0x8BF9 0x9BD8 0xABBB 0xBB9A 0x4A75 0x5A54 0x6A37
0x7A16
0x0AF1 0x1AD0 0x2AB3 0x3A92 0xFD2E 0xED0F 0xDD6C 0xCD4D 0xBDAA
0xAD8B
0x9DE8 0x8DC9 0x7C26 0x6C07 0x5C64 0x4C45 0x3CA2 0x2C83 0x1CE0
0x0CC1
0xEF1F 0xFF3E 0xCF5D 0xDF7C 0xAF9B 0xBFBA 0x8FD9 0x9FF8 0x6E17
0x7E36
0x4E55 0x5E74 0x2E93 0x3EB2 0x0ED1 0x1EF0

（5）SPI 通信流程机制

数据包缓存机制

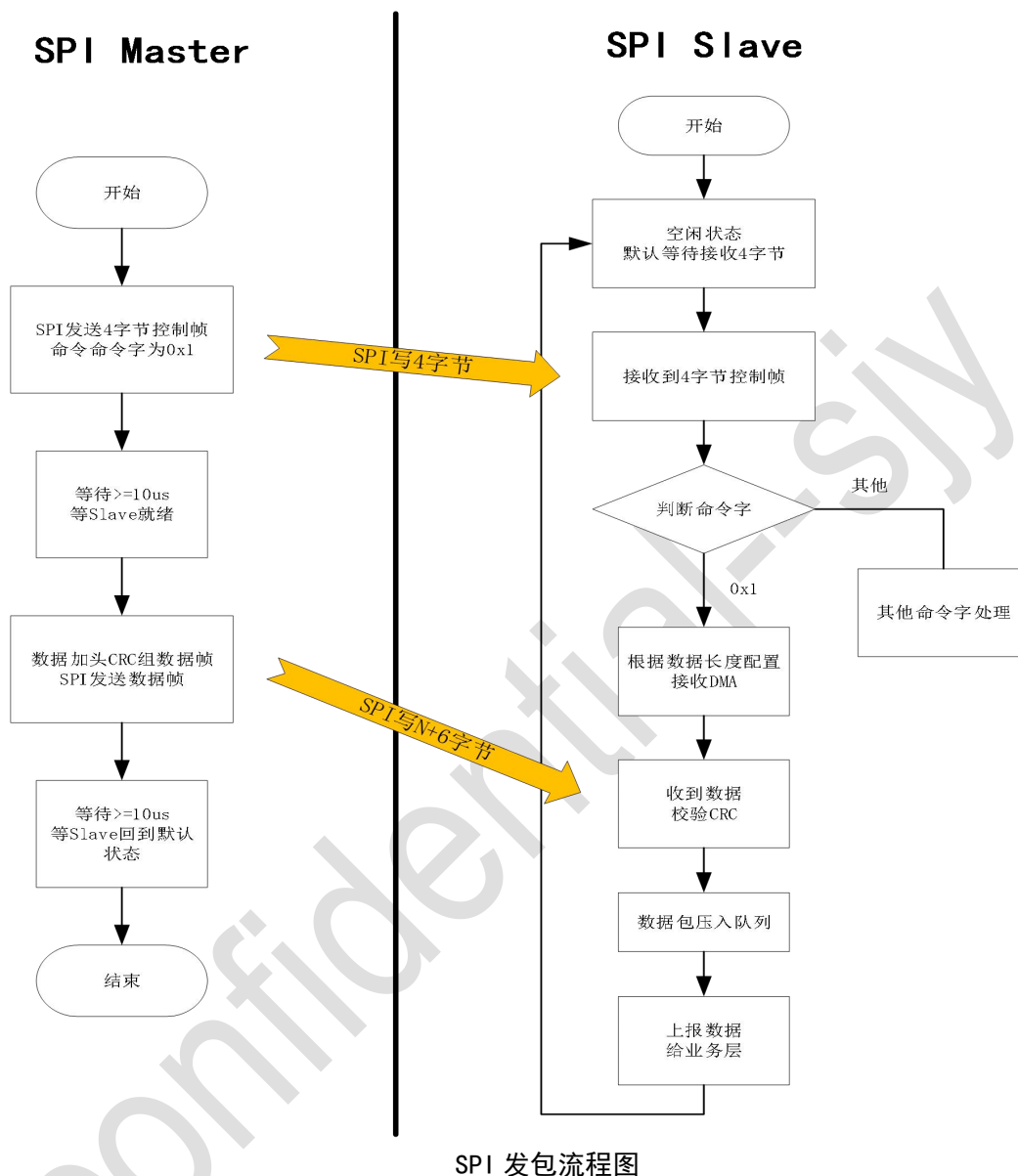
SPI Slave 收发双向均支持多包缓存，使用 FIFO 机制缓存消息数据，默认缓存 10 包，可根据实际情况调整。

流控机制

不支持流控机制，数据包缓存满后，将无法写入，做丢包处理。

Confidential-SLV

Host 发包过程



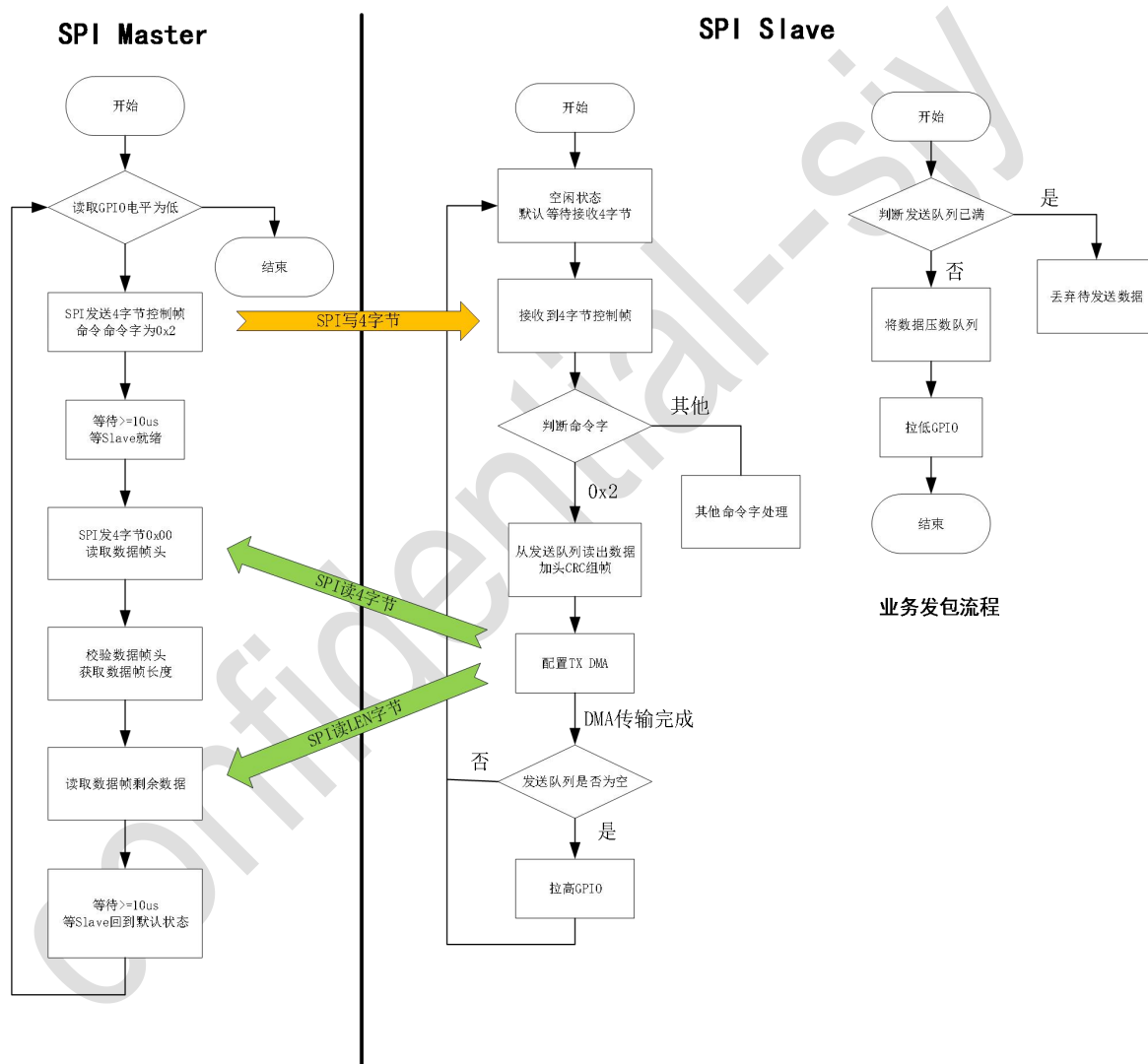
SPI 发包流程图

1) 发包过程如下：

- SPI slave 空闲状态下将 RX DMA 配置为接收长度 4 字节
- Host 在总线空闲时，先发送 0x1 控制命令，并传递发送数据长度，等待 $\geq 10\mu s$
- Slave 收到 4 字节的控制帧，如果命令字为 0x1，确认为 MOSI 过程，根据数据长度配置接收 RX DMA
- Host 直接对有效数据包打包组帧后，通过 SPI 发送，将数据写入 Slave 内存

- e) Slave 收到数据后，CRC 校验，校验失败丢弃（驱动默认不开启 CRC 校验），压入队列，通知业务层读取数据，回到默认接收 4 字节长度状态
- f) Host 等待 $\geq 10\mu s$ 后，发送流程结束

Host 收包过程



SPI 收包流程图

1) Slave 业务发包过程如下：

- a) Slave 业务层数据包下来，填入队列
- b) Slave 拉低 gpio，通知 host 收包

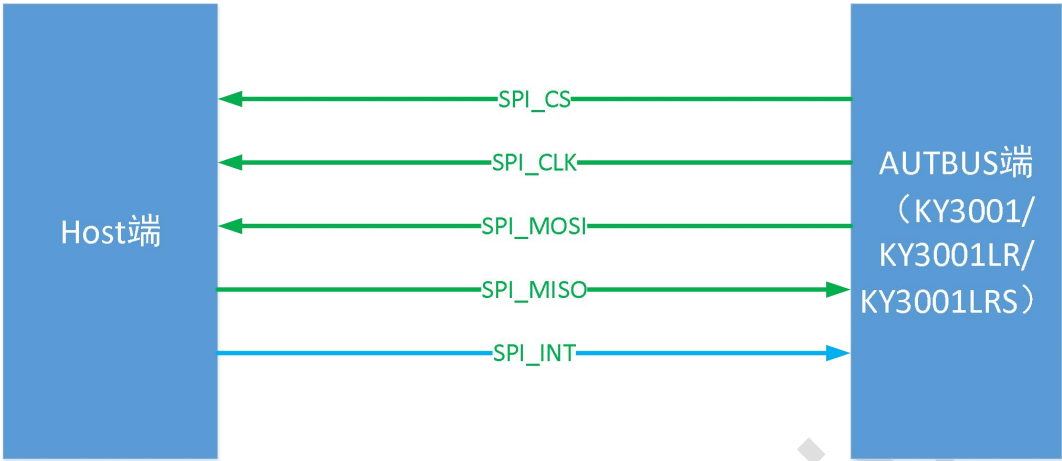
2) Host 收包过程如下:

- a) SPI slave 空闲状态下将 RX DMA 配置为接收长度 4 字节
- b) Host 收到下降沿中断, 或者在总线空闲时读 gpio, 电平为低认为有收包请求, 发送控制帧, 0x2 命令字
- c) Slave 收到 4 字节的控制命令帧, 如果命令字为 0x2, 确认为 MISO 过程
- d) Slave 从队列中取出一包数据, 按照数据帧格式打包好数据到指定内存地址, 配置并使能 TX DMA
- e) Host 等待 $\geq 10\mu\text{s}$ 后, 执行一次 SPI 读 4 字节操作, 读到数据帧头和数据帧长度
- f) Host 根据帧头中的长度 SPI 读取剩余的数据, 校验通过后, 通知业务层从收包队列读走数据
- g) Slave 在 TX DMA 传输结束后, 判断队列是否读空, 如果已读空, 将 gpio 拉高
- h) Host 等待 $\geq 10\mu\text{s}$ 后, 接收 1 包数据结束, 判断 gpio 电平是否为低, 如果为低则继续接收, 为高则接收结束

5.SPI 主模式对接 Host

(1) 接口说明

SPI 主模式下, 使用的是 KY3001 芯片系列的 SPI-0/2 接口(详见 KY3001 芯片手册), 除了 SPI 的标准接口定义外, 有一个额外的 SPI_IN_STAT 定义, 是当 Host 有 MISO 需求时, 对 AUTBUS 端发送的中断信号;



序号	信号名称	信号定义	方向	有效状态	空闲状态
1	SPI_CS	SPI 接口片选信号	从端输入	低电平	高电平
2	SPI_CLK	SPI 接口时钟信号	从端输入	低电平	高电平
3	SPI_MISO	SPI 接口从端数据输出信号	从端输出	高电平/低电平	高电平
4	SPI_MOSI	SPI 接口从端数据输入信号	从端输入	高电平/低电平	高电平
5	SPI_INT	SPI 接口中断信号	从端输出	下降沿	高电平

(2) SPI 物理层参数

SPI 物理层协议：摩托罗拉协议；

波特率：<=50Mbps；

字长：8bit；

极性配置：CPOL:1，CPHA:1；

(3) SPI 通信帧格式

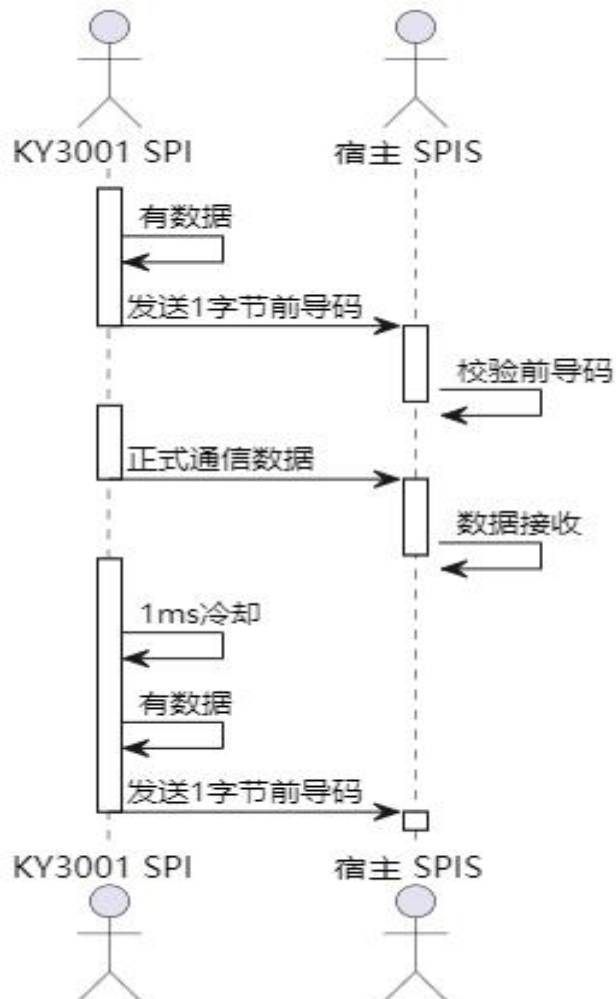


SPI 通讯数据帧格式

序号	参数名称	字节数	数据类型	备注
1~2	帧头	2	UINT16	0xEB90
3~N-2	消息负载	N-6	UINT08	——
N-1~N	CRC 校验	2	UINT16	3~N-2 字节的 CRC16 校验
注：N∈[10, 1024]				

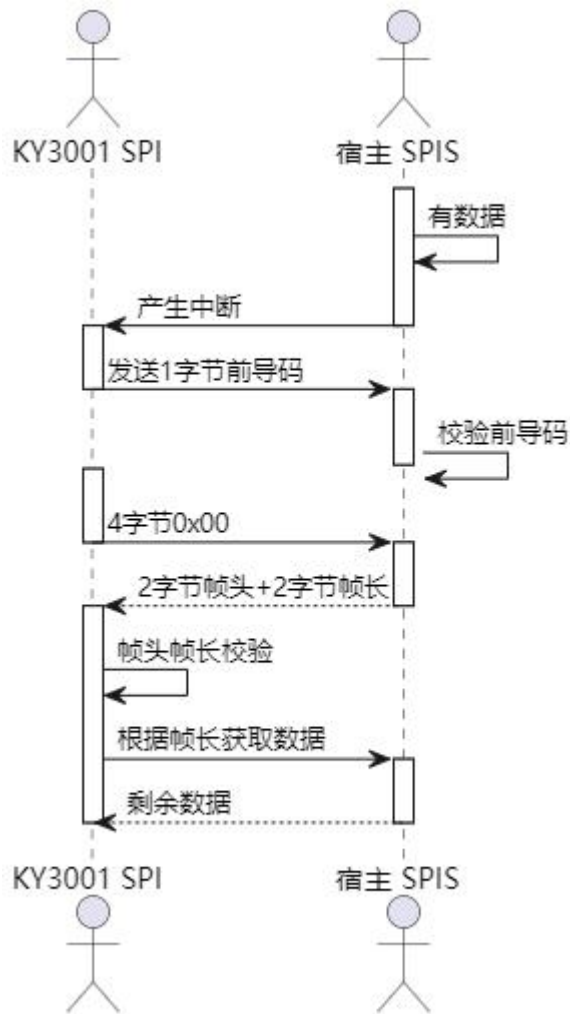
(4) SPI 通信流程机制

MOSI 流程



当主机有待发送数据时，主机向从机发送 1Byte 前导码，然后发送通信数据包，完成 1 次 MOSI 发送；消息框架有冷却功能，等冷却完成可进行下次 MOSI 发送。

MISO 流程

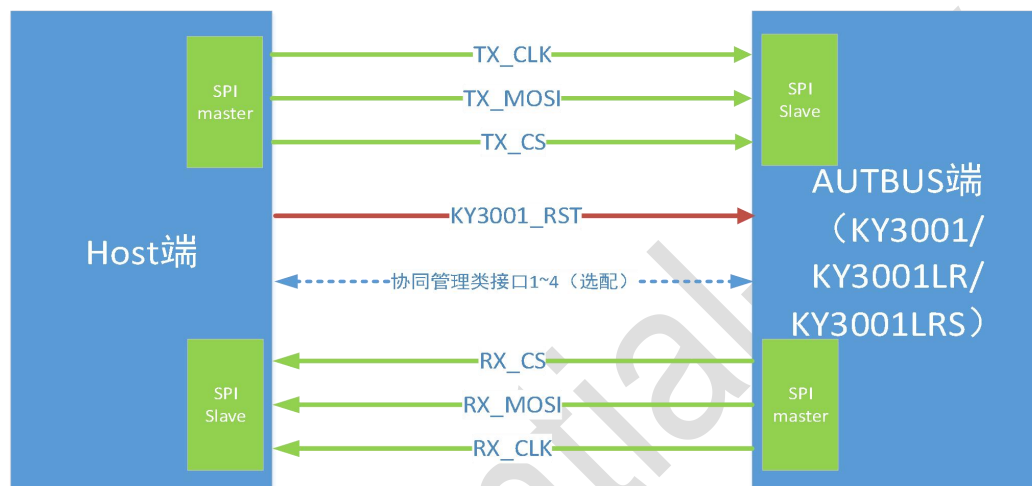


当从机有待发送数据时,通过外部中断通知主机,主机获取中断后发送 1Byte 前导码,然后发送 4Byte 0x00 获取帧头+校验,紧接着根据帧长发送对应字节 0x00 获取剩余数据。

6.双路 SPI 对接 Host

(1) 接口说明

为了提升在 SPI 模式下对接通信效率，以 KY3001 系列芯片为基础，提出双路 SPI 通信对接的方案；此方案构造出一个双向异步的通信模式，即一路 SPI 专门发送到网络侧的数据，另一路 SPI 专门接收从网络侧过来的数据，收发流量间可做到互不影响；



序号	信号名称	信号定义	KY3001 信号名称	方向	有效状态	空闲状态
1	SPI_TX_CS	SPI_TX 接口片选信号	ETH0_RXD3/SYS_GPIO36/ SPI1_CS_B	网络输入	低电平	高电平
2	SPI_TX_CLK	SPI_TX 接口时钟信号	ETH0_RXD2/ SYS_GPIO37/SPI1_CLK	网络输入	低电平	高电平
3	SPI_TX_MOSI	SPI_TX 接口数据输入信号	ETH0_RXD0/SYS_GPIO39/ SPI1_RX	网络输入	高电平/ 低电平	高电平
4	SPI_RX_CS	SPI_RX 接口片选信号	SPI0_CS0_B/SYS_GPIO58/ SPI2_CS3_B	网络输出	低电平	高电平
5	SPI_RX_CLK	SPI_RX 接口时钟信号	SPI0_CLK/SYS_GPIO52/ED M_TRST	网络输出	低电平	高电平
6	SPI_RX_MOSI	SPI_RX 接口数据输出信号	SPI0_TX/SYS_GPIO53/ED M_TMS	网络输出	高电平/ 低电平	高电平

7	SYS_STAT_N	系统协同管理接口信号 (选配参考, 可扩展, 1~4路 GPIO)	ETH0_RXD1/SYS_GPIO38/ SPI1_TX SPI0_RX/SYS_GPIO54/ED M_TDO CCP1_PWM3_P/SYS_GPIO 12/CCP1_CAP3 CCP1_PWM3_N/SYS_GPIO 13/SUART1_DOUT	系统间的 协同管理	高电平/ 低电平	高电平
8	KY3001_RST	KY3001 复位信号	RESET_IN	网络输入	低电平	高电平

如上图所示, TX 方向, Host 端 SPI 为主, KY3001 为从, 使用 CLK, MOSI, CS 信号线即可; RX 方向, 也相同配置;

(2) SPI 物理层参数

方案有两对 SPI 接口, 这里 SPI 的物理层配置各自描述;

SPI_TX 的参数

物理层协议: 摩托罗拉协议;

波特率: $\leq 12.5\text{Mbps}$;

字长: 8bit;

极性配置: CPOL:1, CPHA:1;

SPI_RX 的参数

物理层协议: 摩托罗拉协议;

波特率: $\leq 50\text{Mbps}$;

字长: 8bit;

极性配置: CPOL:1, CPHA:1;

(3) SPI 通信帧格式

数据帧格式

帧头 (0xeb90)	载荷长度 (16bit)	帧载荷	CRC (16bit可选)	Padding (补齐64字节)
----------------	-----------------	-----	------------------	---------------------

- 1) 两对 SPI 通信均有相同的通信帧格式, 和通信规则, 均遵从如下定义;
- 2) SPI 的每次通信均按照固定长度来执行, 即 64 字节 (可根据需求调整);

- 3) 每次通信均以指定帧头开始，后续表明实际载荷长度；
- 4) 不足 64 字节的单次通信，均需要打上 padding 补齐到 64 字节；

字节偏移	参数名称	字节数	数据类型	备注
0~1	帧头	2Byte 小端	UINT16	0xEB90
2~3	帧载荷长度	2Byte 小端	UINT16	N-6（业务 Payload 字节数） 或者 N-4（不计 CRC 的话）
4~N-3	帧载荷	N-6 N-4（不计 CRC 的话）	UINT08	——
N-2~N-1	CRC 校验（可选）	2Byte 小端	UINT16	2~N-3 字节的 CRC16 校验
Padding	补齐 64 字节使用	--	UINT08	补齐 64 字节使用
注：N 为除去 Padding 部分的帧长，是帧的实际内容部分， $N \in [10, 64]$				

SPI 通讯数据帧格式

CRC16 约定

CRC 校验多项式为： $CRC = X^{16} + X^{12} + X^5 + 1$ ，初始值 0x0000，结果异或值 0x0000。具体码表如下：

0x0000	0x1021	0x2042	0x3063	0x4084	0x50A5	0x60C6	0x70E7	0x8108
0x9129								
0xA14A	0xB16B	0xC18C	0xD1AD	0xE1CE	0xF1EF	0x1231	0x0210	0x3273
0x2252								
0x52B5	0x4294	0x72F7	0x62D6	0x9339	0x8318	0xB37B	0xA35A	0xD3BD
0xC39C								
0xF3FF	0xE3DE	0x2462	0x3443	0x0420	0x1401	0x64E6	0x74C7	0x44A4
0x5485								
0xA56A	0xB54B	0x8528	0x9509	0xE5EE	0xF5CF	0xC5AC	0xD58D	0x3653
0x2672								
0x1611	0x0630	0x76D7	0x66F6	0x5695	0x46B4	0xB75B	0xA77A	0x9719
0x8738								
0xF7DF	0xE7FE	0xD79D	0xC7BC	0x48C4	0x58E5	0x6886	0x78A7	0x0840
0x1861								
0x2802	0x3823	0xC9CC	0xD9ED	0xE98E	0xF9AF	0x8948	0x9969	0xA90A
0xB92B								
0x5AF5	0x4AD4	0x7AB7	0x6A96	0x1A71	0x0A50	0x3A33	0x2A12	0xDBFD

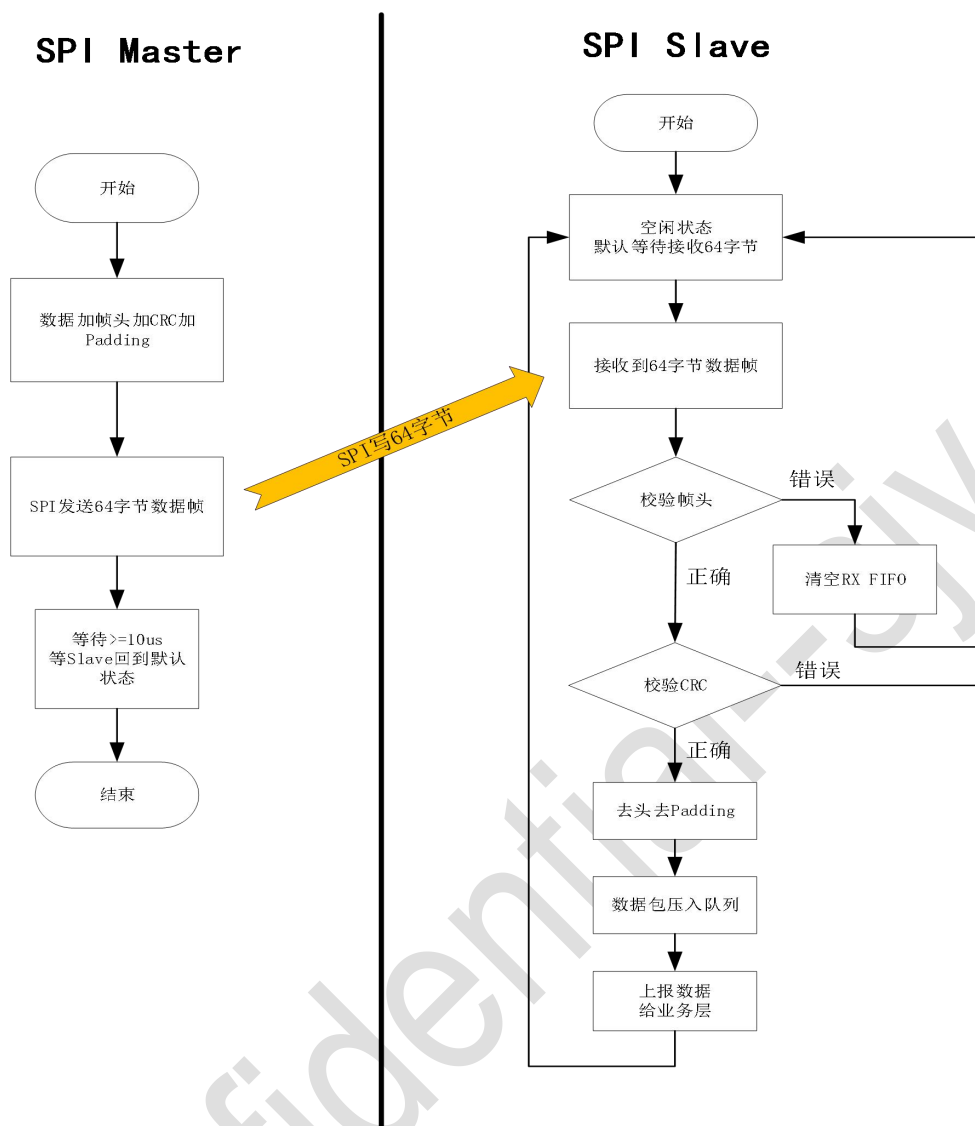
```

0xCBDC
0xFBFB 0xEB9E 0x9B79 0x8B58 0xBB3B 0xAB1A 0x6CA6 0x7C87 0x4CE4
0x5CC5
0x2C22 0x3C03 0x0C60 0x1C41 0xEDAE 0xFD8F 0xCDEC 0xDDCD 0xAD2A
0xBD0B
0x8D68 0x9D49 0x7E97 0x6EB6 0x5ED5 0x4EF4 0x3E13 0x2E32 0x1E51
0x0E70
0xFF9F 0xEFBE 0xDFDD 0xCFFC 0xBF1B 0xAF3A 0x9F59 0x8F78 0x9188
0x81A9
0xB1CA 0xA1EB 0xD10C 0xC12D 0xF14E 0xE16F 0x1080 0x00A1 0x30C2
0x20E3
0x5004 0x4025 0x7046 0x6067 0x83B9 0x9398 0xA3FB 0xB3DA 0xC33D
0xD31C
0xE37F 0xF35E 0x02B1 0x1290 0x22F3 0x32D2 0x4235 0x5214 0x6277
0x7256
0xB5EA 0xA5CB 0x95A8 0x8589 0xF56E 0xE54F 0xD52C 0xC50D 0x34E2
0x24C3
0x14A0 0x0481 0x7466 0x6447 0x5424 0x4405 0xA7DB 0xB7FA 0x8799
0x97B8
0xE75F 0xF77E 0xC71D 0xD73C 0x26D3 0x36F2 0x0691 0x16B0 0x6657
0x7676
0x4615 0x5634 0xD94C 0xC96D 0xF90E 0xE92F 0x99C8 0x89E9 0xB98A
0xA9AB
0x5844 0x4865 0x7806 0x6827 0x18C0 0x08E1 0x3882 0x28A3 0xCB7D
0xDB5C
0xEB3F 0xFB1E 0x8BF9 0x9BD8 0xABBB 0xBB9A 0x4A75 0x5A54 0x6A37
0x7A16
0x0AF1 0x1AD0 0x2AB3 0x3A92 0xFD2E 0xED0F 0xDD6C 0xCD4D 0xBDAA
0xAD8B
0x9DE8 0x8DC9 0x7C26 0x6C07 0x5C64 0x4C45 0x3CA2 0x2C83 0x1CE0
0x0CC1
0xEF1F 0xFF3E 0xCF5D 0xDF7C 0xAF9B 0xBFBA 0x8FD9 0x9FF8 0x6E17
0x7E36
0x4E55 0x5E74 0x2E93 0x3EB2 0x0ED1 0x1EF0

```

(4) SPI 通信流程

两对 SPI 通信均有相同的流程，在此介绍的流程对于两对 SPI 通信均通用；



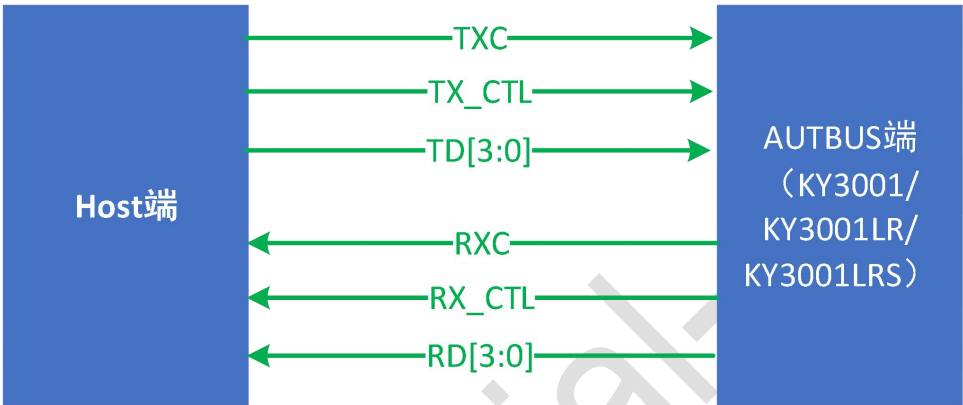
1) 如上图所示，一路 SPI 发送接收流程：

- SPI slave 空闲状态下将 RX DMA 配置为接收长度 64 字节
- SPI master 在总线空闲时，当有数据需要发送，驱动首先将数据加头，加 CRC，加 0 Padding，之后将 64Byte 数据帧通过 SPI 发送出去
- Slave 收到 64 字节的数据帧，先校验帧头和长度，校验失败做相应的后处理后回到等待接收 64 字节状态，校验成功后校验 CRC，CRC 校验失败则丢弃数据，回到等待接收 64 字节状态，校验成功后，将数据帧去头去 Padding，并压入队列，通知业务层读走数据，最后回到等待接收 64 字节状态
- SPI master 发完数据帧后等待 $\geq 10\mu s$ 冷却时间后，发送流程结束，可进行下次发送

7.RGMII 对接 Host

(1) 接口说明

ETH-RGMII 接口下，使用的是 KY3001 芯片系列的 ETH0 接口（详见 KY3001 芯片手册）；



序号	信号名称	信号定义	方向	有效状态	空闲状态
1	TXC	HOST 发送时钟，25M	HOST 端输出	高电平/低电平	高电平/低电平
2	TX_CTL	HOST 发送控制信号	HOST 端输出	高电平/低电平	低电平
3	TD [3:0]	HOST 发送数据	HOST 端输出	高电平/低电平	低电平
4	RXC	HOST 接收时钟，25M/125M	KY3001 端输入	高电平/低电平	高电平/低电平
5	RX_CTL	HOST 接收控制信号	KY3001 端输入	高电平/低电平	低电平
6	RD [3:0]	HOST 接收数据	KY3001 端输入	高电平/低电平	低电平

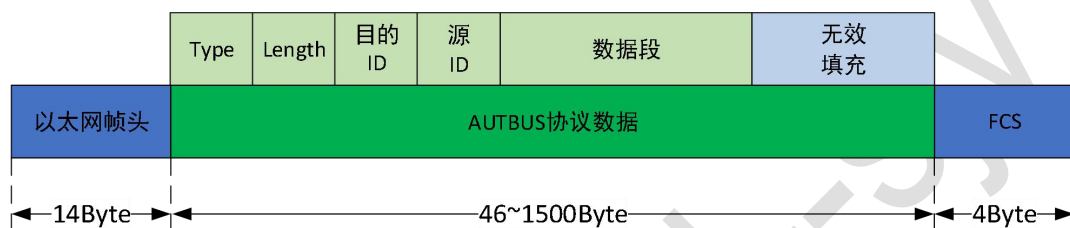
(2) 物理参数

ETH MAC 常规配置，带宽 100M/1000M 自适应（推荐 100Mbit/s）；

(3) ETH 帧格式

整体帧格式如下图，蓝色部分是 ETH 网的帧结构（遵从标准 ETH 帧格式），内部承载的绿色部分是 AUTBUS 的协议数据（格式详见“AUTBUS 通信协议”）；

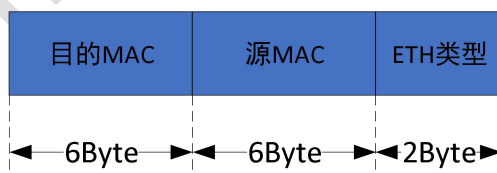
为了遵从标准 ETH 帧格式，对于 AUTBUS 的协议数据不足 46 字节的情况，驱动需要将其补足到 46 字节；



RGMI I 通讯数据帧格式

序号	参数名称	字节数	数据类型	备注
1~14	帧头	14	— —	— —
15~N-4	AUTBUS 协议数据	N-18	— —	— —
N-4~N	FCS	4	UINT32	FCS 校验
注：N ∈ [64, 1518]				
注：当协议层数据不足 46 时，无效填充字段需填写 00 _h ，满足以太网最小 64byte 要求				

进一步的，对于 14 字节的 ETH 帧头，有特殊定义如下：



以太网帧头格式

序号	参数名称	字节数	数据类型	备注
1~6	目的 MAC	6	— —	0x10-0x22-0x33-0x44-0x55-0x66（可订制）
7~12	源 MAC	6	— —	0x60-0x55-0x44-0x33-0x22-0x11（可订制）
13~14	ETH 类型	2	UINT16	0x8888

其中，对于目的 MAC 和源 MAC，根据各自情况可自定义；

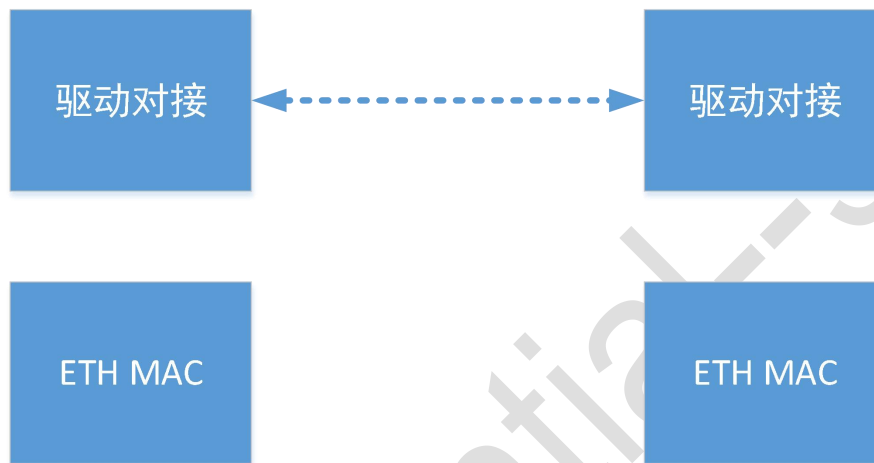
ETH 类型，则规定为 0x8888；

(4) ETH 通信机制

ETH 通信机制上，按照需求有几种场景：

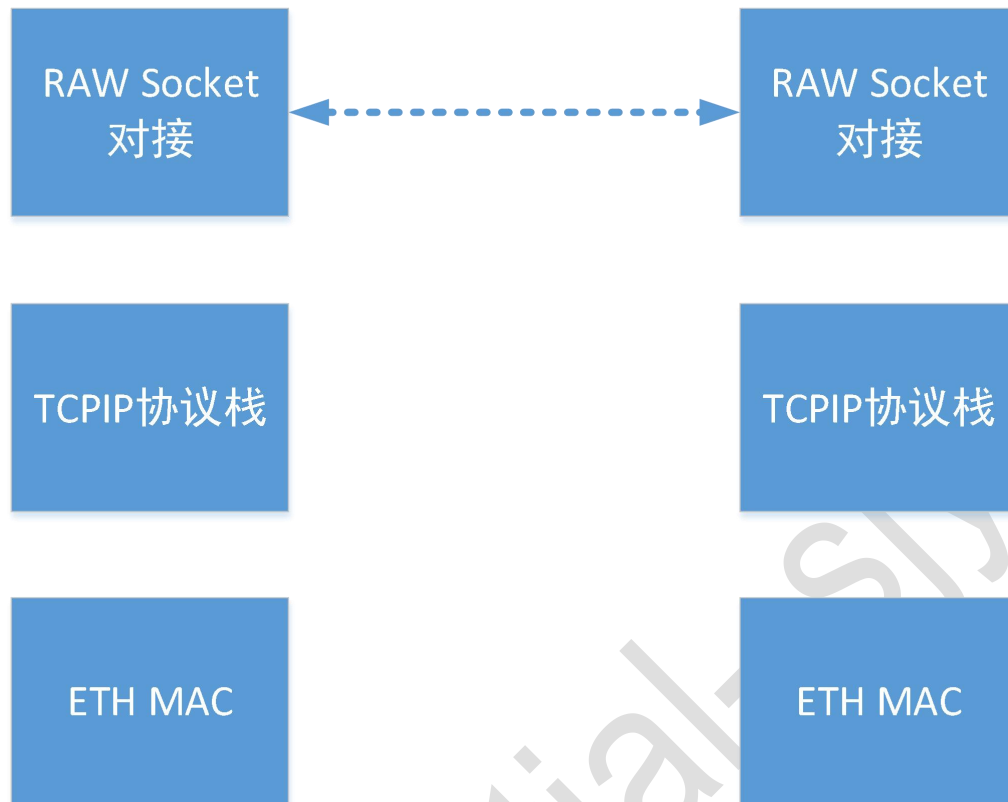
纯 MAC 驱动对接场景

此场景适用于对接双方均无 TCP/IP 协议栈情况，直接对接；



RAW SOCKET 对接场景

此场景适用于对接双方基于 TCP/IP 协议栈后使用原始套接字方式对接；



具体方式：

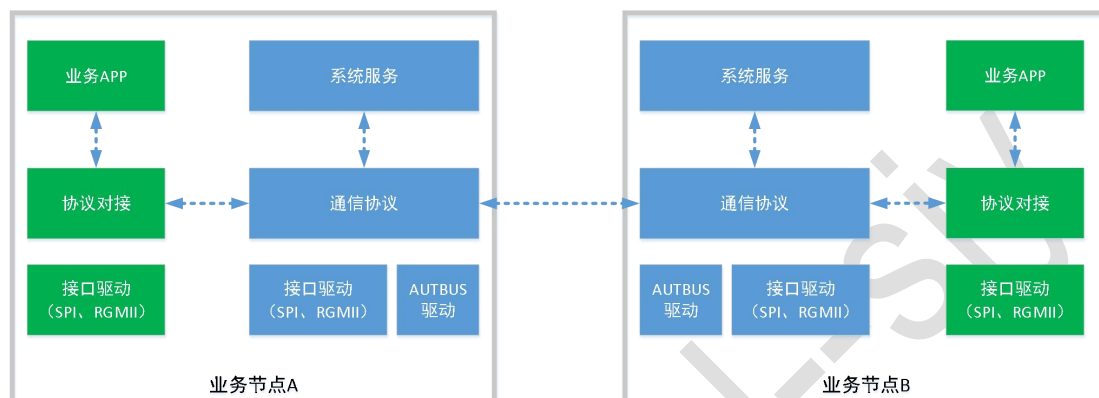
- (1) 使用 SOCK_RAW + PF_PACKET 模式，发送接收 ETH 数据帧；
- (2) 使用特殊 ETH 类型-（0x8888）；
- (3) 使用混杂方式（即可收非本机 MAC 的 ETH 数据帧）；

具体 socket 配置参见伪码如下：

```
sockfd = socket(PF_PACKET, SOCK_RAW, htons(0x8888));
req.ifr_flags |= IFF_PROMISC;
ioctl(eth_info_p->sockfd, SIOCSIFFLAGS, &req);
sll.sll_family = AF_PACKET;
sll.sll_ifindex = req.ifr_ifindex;
bind(sockfd, (struct sockaddr *)&sll, sizeof(sll));
```

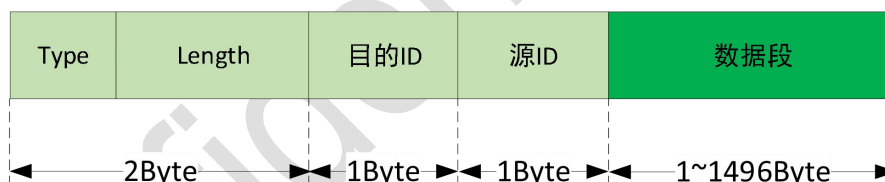
8. AUTBUS 通信协议

AUTBUS 端提供通信协议，和 Host 端做协议对接（Host 端需要有对应的协议对接模块供其自身业务 APP 使用，如下图），协议约定其帧格式，和基于 AUTBUS 通信系统通用的转发规则和策略，以及不同的服务类型；



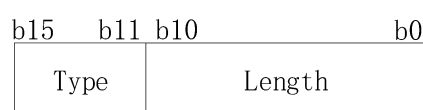
(1) 帧格式概述

帧格式主要字段如下：



序号	参数名称	字节数	数据类型	备注
1~2	Type + Length	2	UINT16	Type: 5bit; Length: 11bit, 3~N 字节数;
3	目的 ID	1	UINT08	目的 Autbus 通信 ID (均为逻辑 ID 可配置)
4	源 ID	1	UINT08	源 Autbus 通信 ID (均为逻辑 ID 可配置)
5~N	数据段	N - 4	— —	— —
注 1: $N \in [5, 1500]$				
注 2: 小端字节序传输				

Type Length 字段小端机器存储如下：



(2) 主要字段

TYPE

Type 区域划分如下表，分为系统管理区、业务协议区、扩展区。其中，系统管理区则是 AUTBUS 系统本身管理需求相关；业务协议区为业务转发承载需求相关；扩展区为后续协议扩展做留白。

序号	Type 索引范围	备注
1	00 _h	保留
2	01 _h to 08 _h	系统管理区
3	09 _h to 17 _h	业务协议区
4	18 _h to 1F _h	扩展区

各 type 类型如下：

序号	Type	备注
1	00 _h	Reserved
2	01 _h	设置类
3	02 _h	查询类
4	03 _h	事件类
5	04 _h	请求类
6	05 _h to 08 _h	Reserved
7	09 _h	通用业务类
8	0A _h	通用业务扩展类
9	0B _h to 17 _h	Reserved
10	18 _h to 1F _h	Reserved

对于每类 type，其子类型和可访问性如下：

Type	子类型	可访问性	备注
00h	00h	--	Reserved
01h (设置类)	00h	--	Reserved
	01h	L	设备 ID
	02h	L	重启 KY3001
	03h	L	切换配置
	04h	L	Reserved
	05h	L	开启/关闭心跳
	06h	L	设备 MAC
02h (查询类)	00h	--	Reserved
	01h	L	状态机
	02h	L	版本号
	03h	L	设备 ID
	04h	L	设备 MAC
	05h	L	读取 Flash
03h (事件类)	00h	--	Reserved
	01h	L	状态机
	02	L	授时
04h (请求类)	00h	--	Reserved
	01h	L	请求取消心跳检测
	02h	L	请求 Host 读取固件
05h to 08h	Reserved	--	Reserved
09h	--	R	通用业务类
0Ah	--	R	通用业务扩展 Autbus 时间戳
0Bh to 17h	--	--	Reserved
可访问性: R (remote) 可远程、L (local) 本地			

ID

ID 作为业务报文交互的寻址转发的凭据，网内每个节点均配置且不能重复，其定义规范如下：

ID	备注
0	保留
1-245	单播 ID
246-253	组播 ID
254	缺省 ID, 本地 type 使用

255	广播 ID
-----	-------

(3) 系统管理帧格式（主要部分）

设置 Type，ID 设置

Host 发送：

Type	Length	目的ID	源ID	子类型	参数1B
01h	04h	FEh	FEh	01h	设备ID

KY3001 返回：

Type	Length	目的ID	源ID	子类型	参数1B
01h	04h	FEh	FEh	81h	返回码

设置 Type，重启

Host 发送：

Type	Length	目的ID	源ID	子类型
01h	03h	FEh	FEh	02h

KY3001 返回：

Type	Length	目的ID	源ID	子类型	参数1B
01h	04h	FEh	FEh	82h	返回码

设置 Type，配置切换

Host 发送：

Type	Length	目的ID	源ID	子类型	参数1B
01h	04h	FEh	FEh	03h	配置索引

KY3001 返回:

Type	Length	目的ID	源ID	子类型	参数1B
01h	04h	FEh	FEh	83h	返回码

设置 Type，开启/关闭心跳

Host 发送:

Type	Length	目的ID	源ID	子类型	参数1B	参数2B小端
01h	06h	FEh	FEh	05h	01h开启/00h关闭	心跳间隔ms

KY3001 返回:

Type	Length	目的ID	源ID	子类型	参数1B
01h	04h	FEh	FEh	85h	返回码

说明:心跳帧为状态上报事件帧,心跳开启后,ky3001 按照设置的间隔(单位 ms,最小 10ms),周期上报状态事件

设置 Type，MAC 设置

Host 发送:

Type	Length	目的ID	源ID	子类型	参数6B
01h	09h	FEh	FEh	06h	设备MAC

KY3001 返回:

Type	Length	目的ID	源ID	子类型	参数1B
01h	04h	FEh	FEh	86h	返回码

查询 Type，状态查询

Host 发送:

Type	Length	目的ID	源ID	子类型
02h	03h	FEh	FEh	01h

KY3001 返回:

Type	Length	目的ID	源ID	子类型	参数1B	参数1B
02h	05h	FEh	FEh	81h	返回码	状态

查询 Type, ID 查询

Host 发送:

Type	Length	目的ID	源ID	子类型
02h	03h	FEh	FEh	03h

KY3001 返回:

Type	Length	目的ID	源ID	子类型	参数1B	参数1B	参数8B
02h	0Dh	FEh	FEh	83h	返回码	设备ID	保留

查询 Type, MAC 查询

Host 发送:

Type	Length	目的ID	源ID	子类型
02h	03h	FEh	FEh	04h

KY3001 返回:

Type	Length	目的ID	源ID	子类型	参数1B	参数6B
02h	0Ah	FEh	FEh	84h	返回码	设备MAC

查询 Type, 读取 Flash

Host 发送:

Type	Length	目的ID	源ID	子类型	参数1B	参数4B	参数4B
02h	0Ch	FEh	FEh	05h	保留	偏移	长度

KY3001 返回:

Type	Length	目的ID	源ID	子类型	参数1B	参数4B	参数4B	参数nB
02h	0Ch+n	FEh	FEh	85h	返回码	偏移	长度n	数据

说明:

1. Host 发送读取 flash 请求，参数为偏移和长度，均 4 字节小端，保留参数传 0。
2. 偏移起始值为 0x0，对应 host 在 KY3001 可读 flash 空间的起始地址，空间大小根据项目需求而定。
3. KY3001 返回内容包括：返回码（见返回码定义），偏移，长度，和数据，参数偏移和长度，为 4 字节小端，数据为从 flash 指定偏移和长度读取的数据内容。

事件 Type，状态事件

KY3001 发送:

Type	Length	目的ID	源ID	子类型	参数1B	参数1B
03h	05h	FEh	FEh	01h	00h	状态

事件 Type，授时事件

KY3001 发送:

Type	Length	目的ID	源ID	子类型	参数1B	参数8B
03h	0Ch	FEh	FEh	02h	00h	授时时间

请求 Type，取消心跳检测

KY3001 发送:

Type	Length	目的ID	源ID	子类型
04h	03h	FEh	FEh	01h

Host 返回:

Type	Length	目的ID	源ID	子类型	参数1B
04h	04h	FEh	FEh	81h	返回码

说明: 请求 host 取消心跳检测，例如防止升级时 MCU 将 KY3001 复位

请求 Type，请求 Host 读取固件

KY3001 发送：

Type	Length	目的ID	源ID	子类型	参数1B	参数4B
04h	08h	FEh	FEh	02h	下载状态	固件长度

Host 返回：

Type	Length	目的ID	源ID	子类型	参数1B
04h	04h	FEh	FEh	82h	返回码

下载状态：1 字节，0：下载成功，1：下载失败。

固件长度：4 字节小端，例如：长度为 0x101234Byte，存储格式为：0x34 0x12 0x10 0x00。

返回码定义

序号	码值	含义
1	00h	无故障
2	01h	类型错误
3	02h	长度错误
4	03h	ID 错误
5	04h	子类型错误
6	05h	参数错误
7	06h	执行错误
8	07h to FFh	保留

(4) 通用业务帧格式

通用业务帧

帧格式：

Type	Length	目的ID	源ID	业务数据
09h	XXh	XXh	XXh	...

说明：Length 字段为业务数据长度加上源/目的 ID 2 字节。

通用业务时间戳扩展帧

帧格式：

Type	Length	目的ID	源ID	源发送时间戳	目的接收时间戳	业务数据
0Ah	XXh	XXh	XXh	8Byte小端	8Byte小端	...

说明：

Length 字段为业务数据长度加上源/目的时间戳 16 字节，源/目的 ID 2 字节。

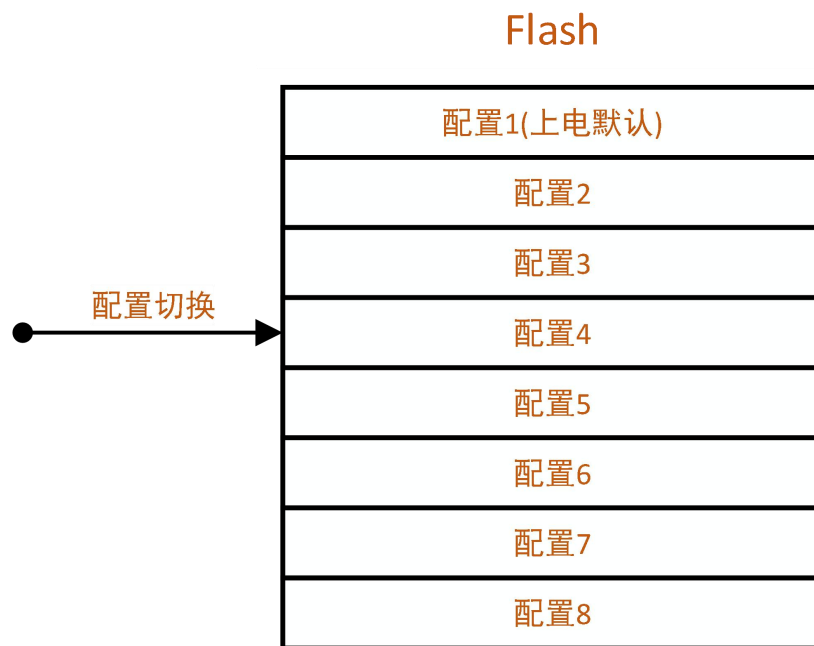
时间戳字段为 8Byte 小端格式，单位为 ns，由 KY3001 芯片增加，用于统计发送到接收的时间间隔，用户发送时预留 16 字节时间戳字段即可。

9. AUTBUS 系统配置服务

AUTBUS 端提供的系统配置服务，包含对本端 AUTBUS 节点的转发 ID 的配置，节点通信配置文件的切换（主节点专用），以及重启等配置等，此服务的获取方式是通过通信协议的指定帧格式去进行访问，具体详见“系统管理帧格式”；

转发 ID 由 Host 决定，此服务提供对本地节点进行转发 ID 的配置，进一步的，还可以提供组播 ID 的配置；

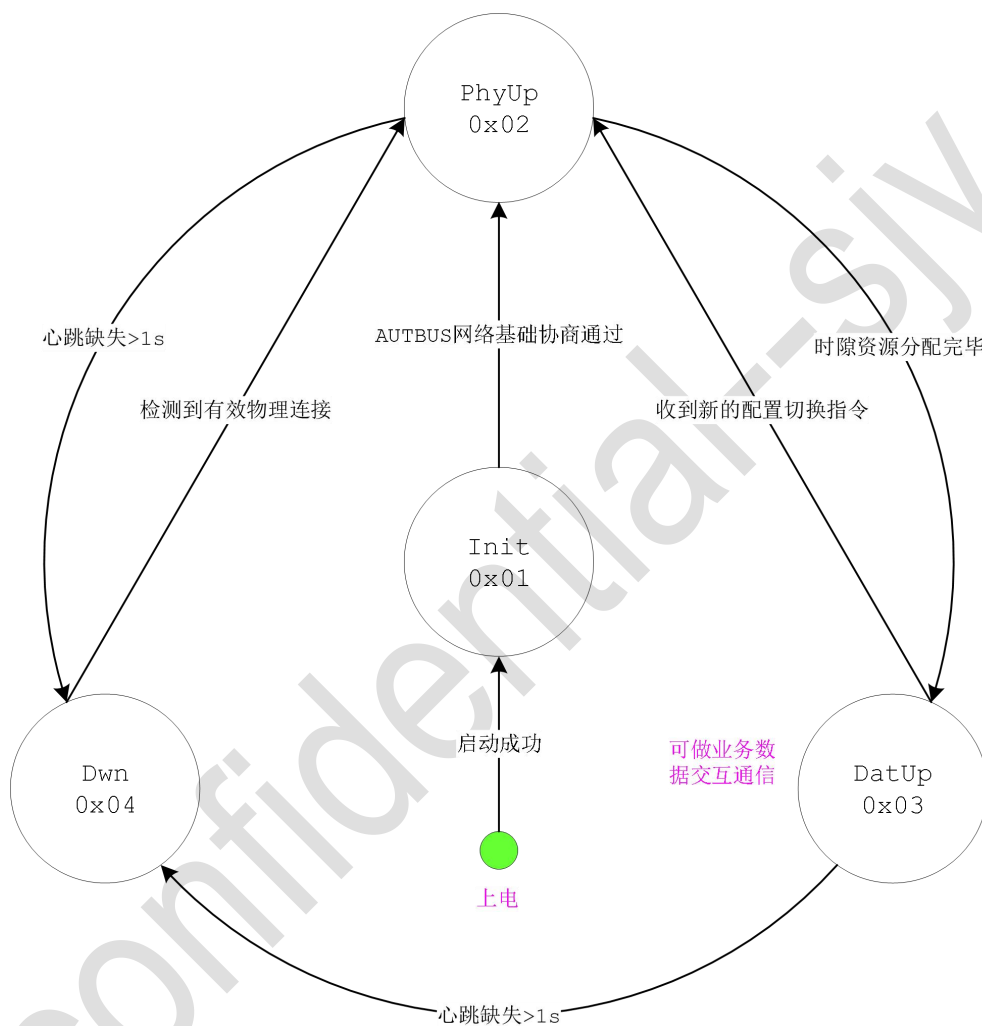
网络通信配置文件的切换功能如果要使用的话，预先要存入最多 8 份配置文件，然后编号，默认第一个配置文件启动，具体如下：



通过 autbus 组网的业务系统，需要根据业务场景的变化去动态切换时隙资源配置，其内部流程如下：

1. 通过离线配置方式，根据不同场景需求来形成多套资源配置文件，此配置文件会预先固化至指定 CN 节点的 flash 中；
2. 对应的系统通过下发 AUTBUS 设置配置切换帧，来触发本地的 AUTBUS-CN 节点激活指定配置文件，执行配置切换；
3. Autbus-CN 节点接收到配置切换帧后，网络切换状态为 PHY-UP 状态，并向其本地系统发送状态变更到 PHY-UP 的状态帧，广播发送配置切换的消息给所有 TN 节点，然后清除自己当前的配置资源表；
4. 所有 TN 节点收到配置切换的消息后，均回退到 PHY-UP 状态，给自己对应的核心 CPU 系统发送状态变更到 PHY-UP 的状态帧，然后清除自己的配置资源表，并且反馈给 CN 状态切换确认帧（AUTBUS 内部帧）；
5. 等收到所有节点的配置切换状态确认后，CN 节点开始进行新配置的装载流程和相关的配置协商处理；
6. 配置切换处理是如同初始化的资源配置一样的执行流程，即从 PHY-UP 到 DATA-UP 的过程；

状态机:



执行程序搬移、系统初始化及自检工作，完成上电初始化工作后转入 Init 状态，此阶段实测历时约 1.7s。

与 AUSBUS 网络建立物理层连通，通过 AUSBUS 网络基础协商（物理层同步，统一全网时间系统）后转入 PhyUp 状态，此阶段最大实测历时不超过 100ms。

44 / 55

根据网络资源配置文件（Init 状态->PhyUp 状态：利用默认网络资源配置文件；DatUp 状态->PhyUp 状态：利用通信指定的网络资源配置文件；Dwn 状态->PhyUp 状态：利用脱网前的网络配置文件）分配全网时隙资源，当建立正确的时隙后转入 DatUp 状态。

转入此状态时刻，立即启动网络授时功能，并完成转入 PhyUp 状态的首次 AUTBUS 网络授时，其后开展周期（10s）AUTBUS 网络授时。

此状态中，若网络心跳缺失超过 1s，则判定脱网，转入 Dwn 状态。

此状态的历时与节点数、网络配置时隙大小等因素相关，当网络节点数为 9 时，此状态的历时约为 500ms。

DatUp 状态

处于此状态时，可以视情开展业务数据的通信交互，也可以视情调用新的网络配置文件。

此状态中，延续 PhyUp 状态的网络授时功能。

此状态中，若收到新的配置切换指令，转入 PhyUp 状态。

此状态中，若网络心跳缺失超过 1s，则判定脱网，转入 Dwn 状态。

Dwn 状态

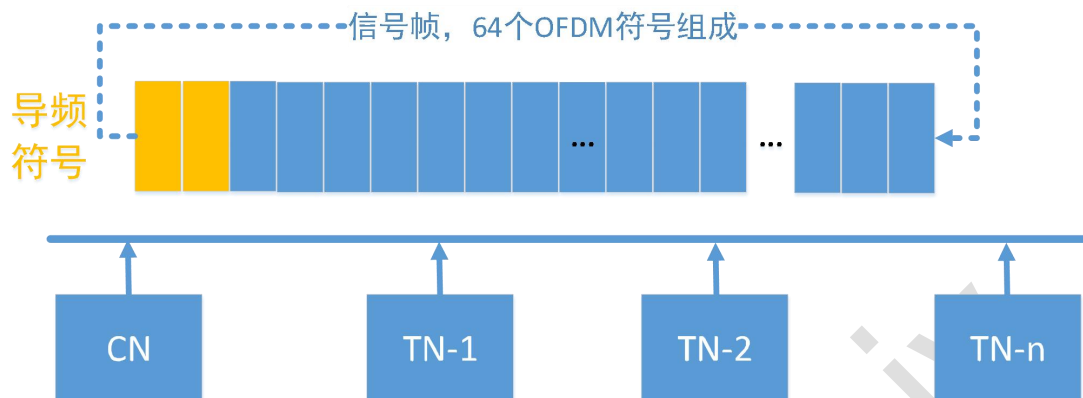
处于此状态时，不能进行业务数据的通信交互，不能进行网络重配。期间，停止网络授时功能，若检测到有效物理连接，则转入 PhyUp 状态，转换历时不超过 32ms。

11. AUTBUS 网络时间同步服务

AUTBUS 端内部，整网维护一个统一的 AUTBUS 时间，后续可提供以下几种服务（包括但不限于），具体如下：

- （1）PPS+TOD 方式来提供授时给 Host，其中 PPS 是 AUTBUS 整周期的对外的信号激励，TOD 方式为 AUTBUS 通信帧（授时事件）；
- （2）仅提供“PPS”给 Host，可设置 PPS 的周期（是以 AUTBUS 整周期为单位）；
- （3）仅提供“TOD”给 Host，可设置 TOD 发送的周期（默认 1s 发送）；

(4) 如果有其他需求可详细讨论；



AUTBUS 的物理层基于 OFDM 技术，其信号帧由导频符号和数据符号组成，两个连续的导频符号构成信号帧头，用来进行系统同步。随着 CN 节点周期性发送信号帧，所有 TN 节点就会随之周期性同步。同步利用的 AUTBUS 时钟源为 CN 本地晶振，所有 TN 节点的 AUTBUS 时钟与其对齐。

TN 节点只要正常上线，达到 PHY-UP 状态，就一直会维护和 CN 的同步，共享内部的 AUTBUS 网络时间。

Autbus 网络内部的时间用于网络同步使用，其由三个部分组成（类似时-分-秒结构）： $\text{frame_cnt} + \text{symbol_cnt} + \text{sample_cnt}$ 。其中： frame_cnt 和 symbol_cnt 的计时步长受网络配置影响，不同的网络配置， frame_cnt 和 symbol_cnt 的计时步长会发生变化； sample_cnt 的计时步长固定为 40ns，不受网络配置影响。故利用 Autbus 网络授时时，采用 sample_cnt 计数值， sample_cnt 为 8 字节长。

12. AUTBUS 到 Host 心跳服务

AUTBUS 提供向 Host MCU 周期发送心跳帧服务，使 Host MCU 能够确认 AUTBUS 芯片是否正常运行，并对异常情况做出相应的后处理，例如重启 AUTBUS 芯片。

Host 可以设置开启关闭心跳，AUTBUS 芯片也可以请求 Host 关闭心跳检测。

Host 也可以通过周期查询状态的方式确认 AUTBUS 芯片是否运行正常。

（1）开启/关闭心跳帧

Host 可以通过下发开启/关闭心跳的设置帧，来控制 AUTBUS 芯片开启或关闭心跳帧，开启时指定心跳间隔，帧格式如下：

Host 发送：

Type	Length	目的ID	源ID	子类型	参数1B	参数2B小端
01h	06h	FEh	FEh	05h	01h开启/00h关闭	心跳间隔ms

KY3001 返回：

Type	Length	目的ID	源ID	子类型	参数1B
01h	04h	FEh	FEh	85h	返回码

说明：心跳帧为状态上报事件帧，心跳开启后，ky3001 按照设置的间隔（单位 ms，最小 10ms），周期上报状态事件

（2）心跳帧

心跳帧即状态事件帧，通过周期上报状态事件作为心跳，帧格式如下：

KY3001 发送：

Type	Length	目的ID	源ID	子类型	参数1B	参数1B
03h	05h	FEh	FEh	01h	00h	状态

（3）请求取消心跳检测

在 AUTBUS 芯片升级等场景会出现较长时间不能上报心跳的情况，在该情况下 AUTBUS 芯片会提前向 Host MCU 发送取消心跳检测帧，并需要 Host 收到后关闭检测并回复，未及时回复有 10 次的超时重发机制。

请求取消心跳检测帧格式如下：

KY3001 发送：

Type	Length	目的ID	源ID	子类型
04h	03h	FEh	FEh	01h

Host 返回：

Type	Length	目的ID	源ID	子类型	参数1B
04h	04h	FEh	FEh	81h	返回码

说明：请求 host 取消心跳检测，例如防止升级时 MCU 将 KY3001 复位。

13. AUTBUS Host 固件升级服务

为简化 Host MCU 升级方案，可使用 AUTBUS Host MCU 固件升级服务，AUTBUS 上位机工具将 Host 固件暂存到 KY3001 Flash 中，并通知 Host 通过 SPI 读取固件进行升级。

（1）Flash 地址分配

- 1) KY3001 flash 预留 0x400000—0x5FFFFFFF 空间给 Host 固件使用。
- 2) Host 读取范围为 0x0—0x1FFFFFFF，对应 flash 的 0x400000—0x5FFFFFFF，其他 flash 空间 Host 不可读。

（2）下载方案

通过复用 sdk3.1 版本上位机的固件升级功能，将 Host 固件下载到 KY3001 Flash 的 0x400000 偏移的位置，根据固件头区分固件类型，并确认写固件偏移地址。

Host 固件下载前需要增加 16 字节头信息，用于描述文件类型和校验信息，下载时会一起存到 flash 中，方便下载过程的文件校验以及 host 主动读取固件时获取长度校验。

16 字节头格式如下，前 8 字节为“HostFirm”作为 Magic，之后是 4 字节固件长度，小端存储，不包含 16 字节头；最后是 4 字节 32 位 CRC 校验值，小端存储，

校验不包含 16 字节头，CRC 算法参考： `crc32.c`。

Magic	固件长度	固件校验值
8 字节字符：“HostFirm”	4 字节小端	4 字节小端

下载完成后，KY3001 发送请求 Host 读取固件消息，通知 Host mcu，host 收到后需要返回消息确认已收到，Host 根据该消息进行后续的读取固件升级处理。

请求 Host 读取固件帧格式：

KY3001 发送：

Type	Length	目的ID	源ID	子类型	参数1B	参数4B
04h	08h	FEh	FEh	02h	下载状态	固件长度

Host 返回：

Type	Length	目的ID	源ID	子类型	参数1B
04h	04h	FEh	FEh	82h	返回码

下载状态：1 字节，0：下载成功，1：下载失败。

固件长度：4 字节小端，例如：长度为 0x101234Byte，存储格式为：0x34 0x12 0x10 0x00。

Host 读取 KY3001 Flash 帧格式：

Host 发送：

Type	Length	目的ID	源ID	子类型	参数1B	参数4B	参数4B
02h	08h	FEh	FEh	05h	保留	偏移	长度

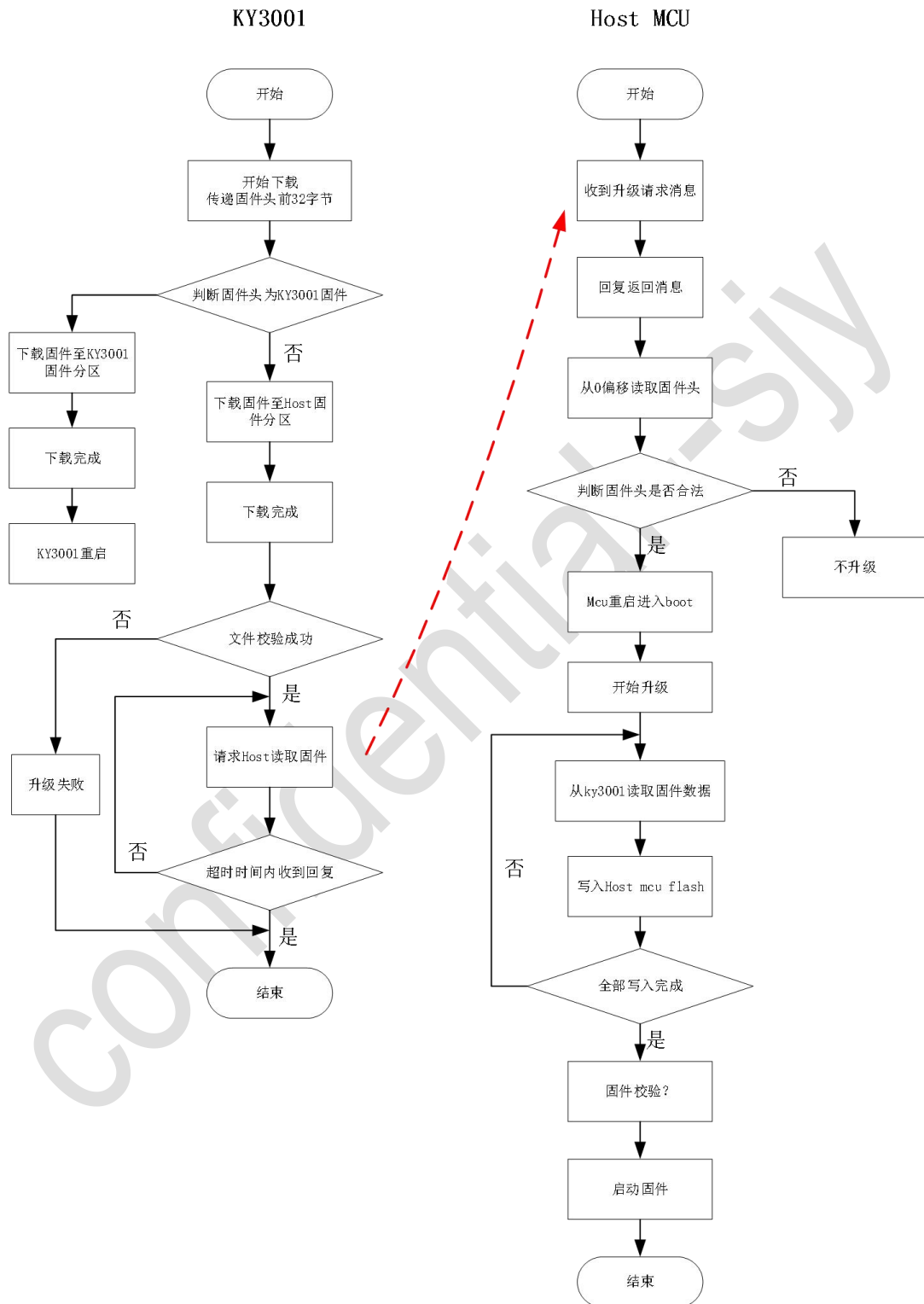
KY3001 返回：

Type	Length	目的ID	源ID	子类型	参数1B	参数4B	参数4B	参数nB
02h	0Bh+n	FEh	FEh	85h	返回码	偏移	长度n	数据

说明：

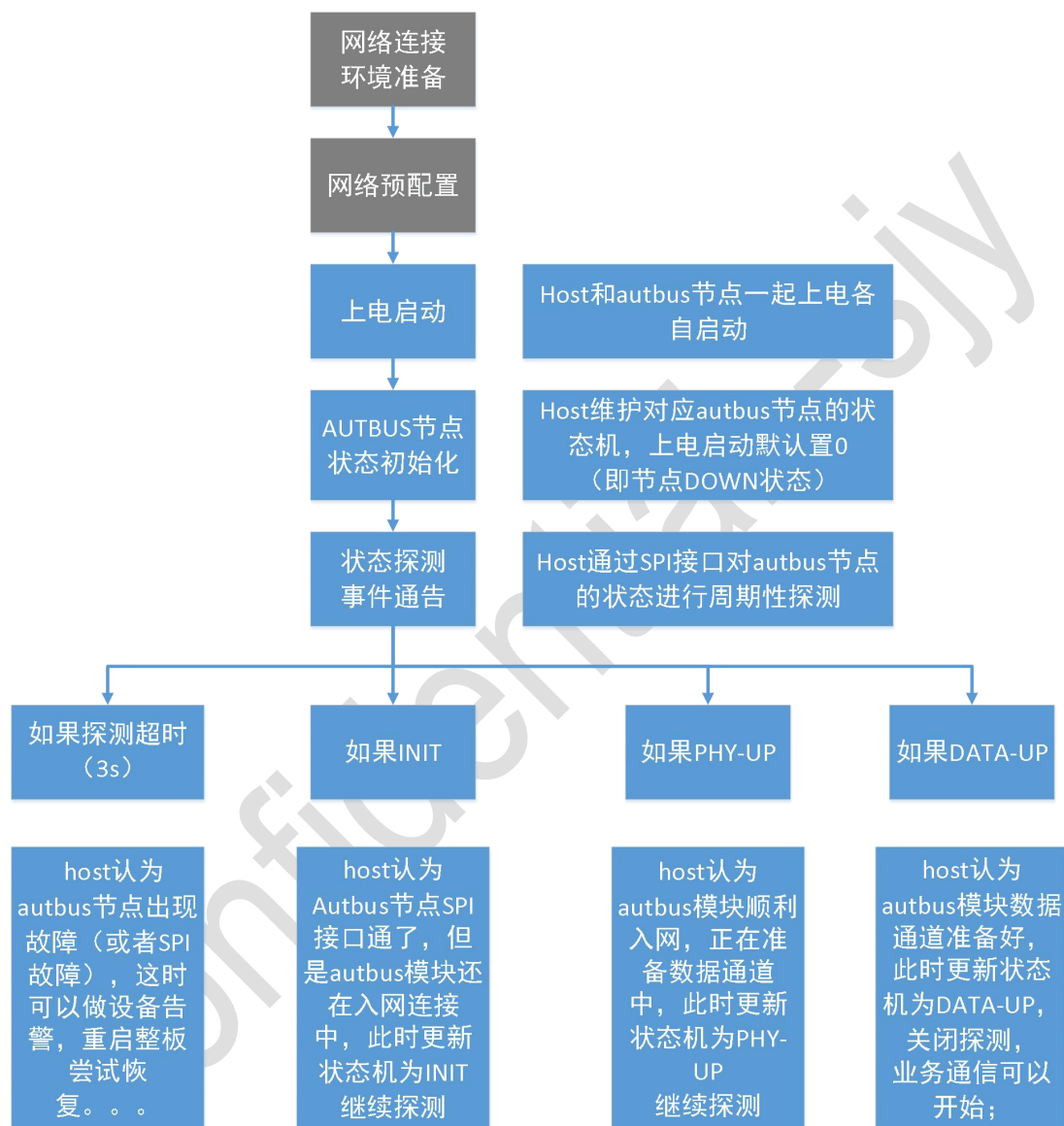
- 1) Host 发送读取 flash 请求，参数为偏移和长度，均 4 字节小端，保留参数传 0。
- 2) 偏移起始值为 0x0，对应 host 在 KY3001 可读 flash 空间的起始地址，空间大小为 0x200000，请求偏移+长度需要小于 0x200000，否则返回错误码。
- 3) KY3001 返回内容包括：返回码（见 AUTBUS 通信协议返回码定义章节），偏移，长度，和数据，参数偏移和长度，为 4 字节小端，数据为从 flash 指定偏移和长度读取的数据内容。

(3) 下载升级流程



四、主要流程简介

1. 启动交互流程



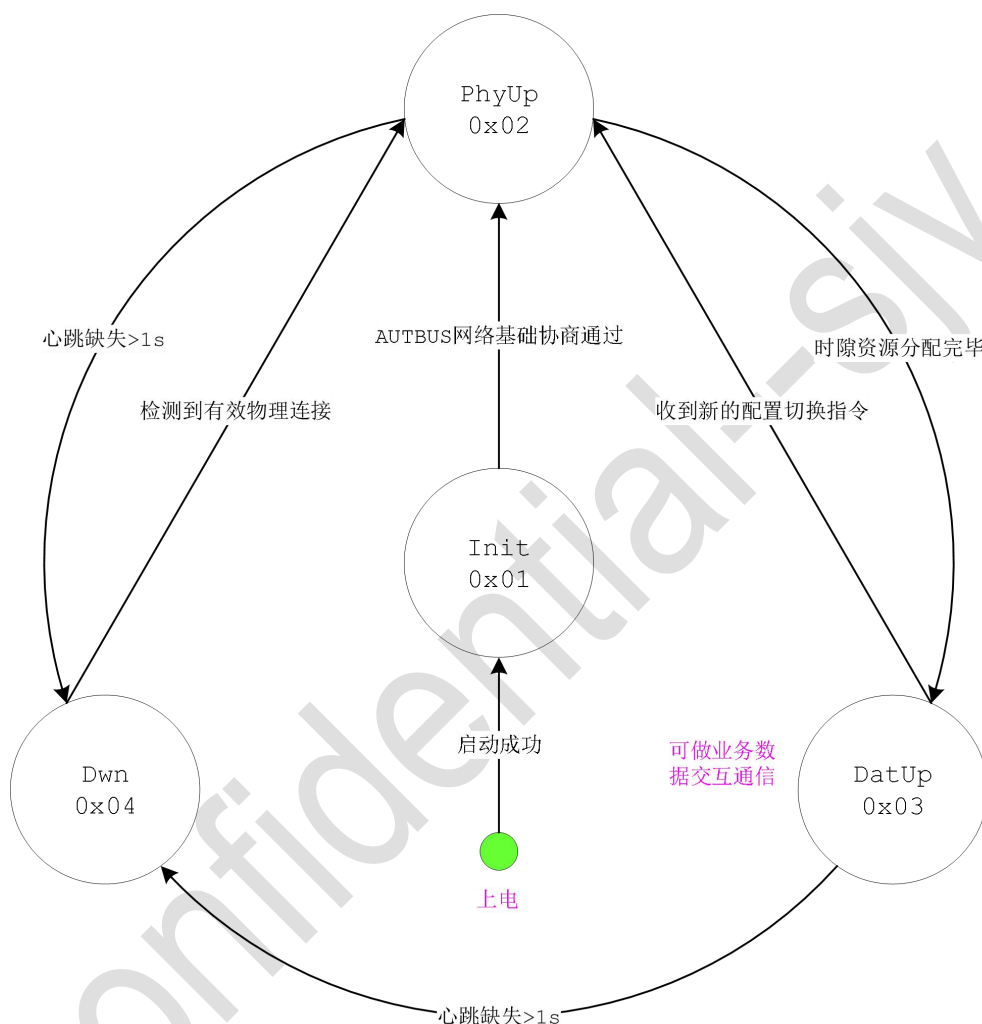
其中，对于网络预配置，是将固有调试好的网络配置文件烧录进主节点的flash中即可；

AUTBUS芯片和Host芯片同板时，启动是同时的，启动后，AUTBUS芯片为被动态，即不主动进行对接通信，由Host端主动发起状态探测来感知状态；

如果对接的是RGMII，则需要启动上注意顺序，需要Host端ETH RGMII

接口先使能之后，再对应初始化 AUTBUS，这个可以通过复位接口，和系统接口来进行启动流程控制，具体可以讨论；

2. AUTBUS 内部网络状态机流程



上电初始化

执行程序搬移、系统初始化及自检工作，完成上电初始化工作后转入 Init 状态，此阶段实测历时约 1.7s。

Init 状态

与 AUTBUS 网络建立物理层连通，通过 AUTBUS 网络基础协商（物理层同步，统一全网时间系统）后转入 PhyUp 状态，此阶段最大实测历时不超过 100ms。

PhyUp 状态

根据网络资源配置文件（Init 状态->PhyUp 状态:利用默认网络资源配置文

件；DatUp 状态→PhyUp 状态：利用通信指定的网络资源配置文件；Dwn 状态→PhyUp 状态：利用脱网前的网络配置文件）分配全网时隙资源，当建立正确的时隙后转入 DatUp 状态。

转入此状态时刻，立即启动网络授时功能，并完成转入 PhyUp 状态的首次 AUTBUS 网络授时，其后开展周期（10s）AUTBUS 网络授时。

此状态中，若网络心跳缺失超过 1s，则判定脱网，转入 Dwn 状态。

此状态的历时与节点数、网络配置时隙大小等因素相关，当网络节点数为 9 时，此状态的历时约为 500ms。

DatUp 状态

处于此状态时，可以视情开展业务数据的通信交互，也可以视情调用新的网络配置文件。

此状态中，延续 PhyUp 状态的网络授时功能。

此状态中，若收到新的配置切换指令，转入 PhyUp 状态。

此状态中，若网络心跳缺失超过 1s，则判定脱网，转入 Dwn 状态。

Dwn 状态

处于此状态时，不能进行业务数据的通信交互，不能进行网络重配。期间，停止网络授时功能，若检测到有效物理连接，则转入 PhyUp 状态，转换历时不超过 32ms。

3. Host 与 AUTBUS 系统常用交互流程

（1）Host 设置本地节点的 ID

Host 发送：

Type	Length	目的ID	源ID	子类型	参数1B
01h	04h	FEh	FEh	01h	设备ID

KY3001 返回：

Type	Length	目的ID	源ID	子类型	参数1B
01h	04h	FEh	FEh	81h	返回码

(2) Host 重启本地节点操作

一种是直接对其复位引脚（RESET_IN）进行直接操作，芯片复位；

另一种是发协议帧进行操作，如下：

Host 发送：

Type	Length	目的ID	源ID	子类型
01h	03h	FEh	FEh	02h

KY3001 返回：

Type	Length	目的ID	源ID	子类型	参数1B
01h	04h	FEh	FEh	82h	返回码

(3) Host 端进行状态查询

Host 发送：

Type	Length	目的ID	源ID	子类型
02h	03h	--	--	01h

KY3001 返回：

Type	Length	目的ID	源ID	子类型	参数1B	参数1B
02h	05h	--	--	81h	返回码	状态

(4) Host 端对本地节点的 ID 进行查询

Host 发送：

Type	Length	目的ID	源ID	子类型
02h	03h	FEh	FEh	03h

KY3001 返回:

Type	Length	目的ID	源ID	子类型	参数1B	参数1B	参数8B
02h	0Dh	FEh	FEh	83h	返回码	设备ID	保留

Confidential-sly