

MAC2MAC 在 Jacinto7 Soc 中的应用

Fredy Zhang; Joe Shen; Tao Han

EP FAE

摘要

Jacinto™ 7 TDA4VM 是目前 TI 最新一代的汽车处理器，面向 ADAS 和自动驾驶车辆 (AV) 应用，并且建立在 TI 在 ADAS 处理器市场十余年领先地位中积累的广泛市场知识基础之上。TDA4VM 以业界领先的功耗/性能比为传统和深度学习算法提供高性能计算，并具有很高的系统集成度，从而使支持集中式 ECU 或多种传感器的高级汽车平台实现可扩展性和更低的成本。

TI 最新一代的 Jacinto™ 7 TDA4VM 同时集成了以太网交换机 (Ethernet Switch) CPSW9G 和 CPSW2G，为节点间的数据交换提供了更高的吞吐量，同时也给系统的集成带来了便捷性。通常 CPSW2G 在 MCU 域使用，CPSW9G 在主域使用。CPSW9G 对外提供 8 个 PORT，因此可以提供对外 8 个设备的连接，可以方便的对汽车系统内各个节点的数据进行交换。在实际的应用场景中：一方面，单个 TDA4VM 的 CPSW2G 通常与主域进行隔离，可单独运行，不能通过 RAM 直接访问。这种情况下，CPSW2G 和 CPSW9G 可以方便地通过 MAC2MAC 进行互联；另一方面，同一个板子上，当单个 TDA4VM SOC 算例不足地情况下或要连接其它处理器等外部连接情况下，可以非常方便地，通过 MAC2MAC 进行连接。显而易见，这样做的好处是去掉了外部的 PHY，从而降低了系统的成本，减少了板子面积，也降低了数据交换的时延。因此，MAC2MAC 的连接广泛使用在 TI Jacinto7 处理器中。

本手册旨在对 Ethernet 框架快速概述，重点介绍了 TI Jacinto7 CPSW 在 MAC2MAC 中的解决方案。

修改记录

Version	Date	Author	Notes
1.0	May 2021	Fredy Zhang	First release

目录

1. Ethernet (以太网) 网口结构及 CPSW 介绍	3
1.1. Ethernet(以太网) 网口结构	3
1.2. Ethernet(以太网) 集成结构	3
1.3. TI CPSW	4
2. JACINTO7 MAC2MAC 解决方案.....	5
2.1. 应用需求	5
2.2. RGMII 解决方案.....	8
2.3. SGMII 解决方案	9
3. MAC2MAC 实现	9
3.1. RGMII 实现.....	9
3.2. SGMII 实现	10
3.3. MAC2MAC 调试	11
4. 总结.....	11
5. 参考.....	11

图

图 1. Ethernet 网口结构.....	3
图 2. CPU 集成 MAC 和 PHY	3
图 3. PHY 采用独立芯片.....	4
图 4. MAC 和 PHY 采用独立芯片	4
图 5. CWSW Architecture.....	5
图 6. 典型的 RGMII 连接	6
图 7. Serdes0 Overview.....	6
图 8. MAC 和 PHY 采用独立芯片	7
图 9. SOC 间 MAC2MAC 互联.....	8
图 10. 外部 Soc 或设备与 TDA4VM 连接	8
图 11. RGMII MAC2MAC	9
图 12. SGMII MAC2MAC	9

表

表 1. SGMII PORT SERDES 对应表	7
---	----------

1. Ethernet (以太网) 网口结构及 CPSW 介绍

1.1. Ethernet(以太网) 网口结构

网口通常由三部分组成：CPU、MAC 和 PHY。它的简图如下图 1 所示，CPU 通常基层 DMA 控制器，这里虚线表示处理器常会使用到 DMA 来存取数据。

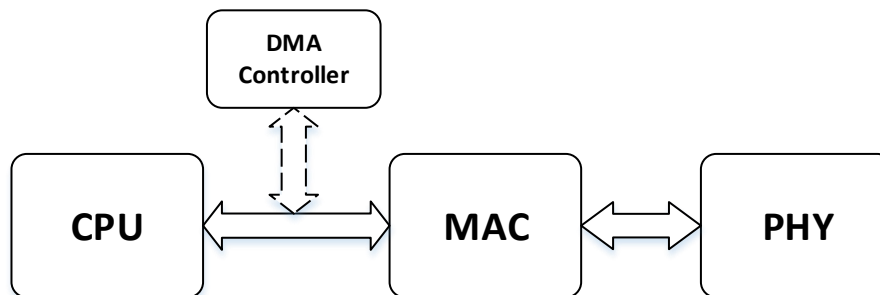


图 1. Ethernet 网口结构

CPU 是计算单元。Ethernet(以太网)的 MAC(Media Access Control, 即媒体访问控制子层协议)是属于 OSI 模型中的数据链路层。用来定义网络的位置，也即物理地址。PHY(Physical Layer)属于 OSI 模型中的最底层物理层，负责链路建立、维护和断开。IEEE-802.3 标准定义了以太网 PHY。PHY 的常用接口类型有 MII/GMII/RMII/RGMII/SGMII/SGMII 等。

1.2. Ethernet(以太网) 集成结构

上面我们提到了网口通常由三部分组成：CPU、MAC 和 PHY。PHY 整合了模拟的硬件。MAC 是全数字器件。考虑到芯片面积，数字和模拟混合的原因，通常有下面三种集成结构：

1. CPU 集成 MAC 和 PHY 其结构如图 2 所示下：

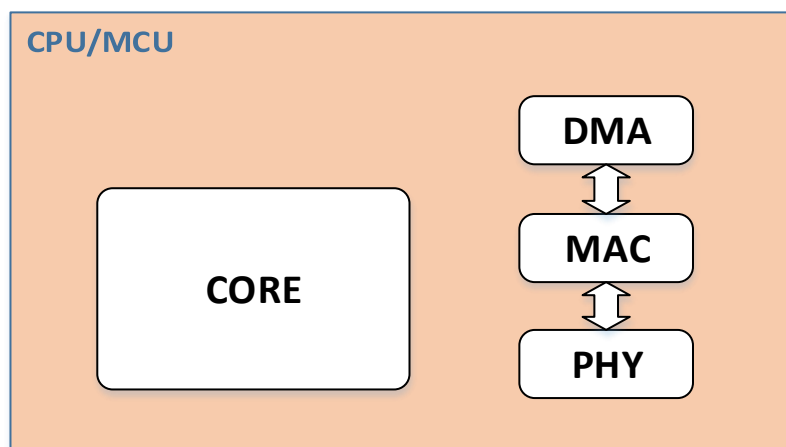


图 2. CPU 集成 MAC 和 PHY

2. CPU 集成 MAC, PHY 采用独立芯片，其结构如图 3 所示：

Overwrite this text with the Lit. Number

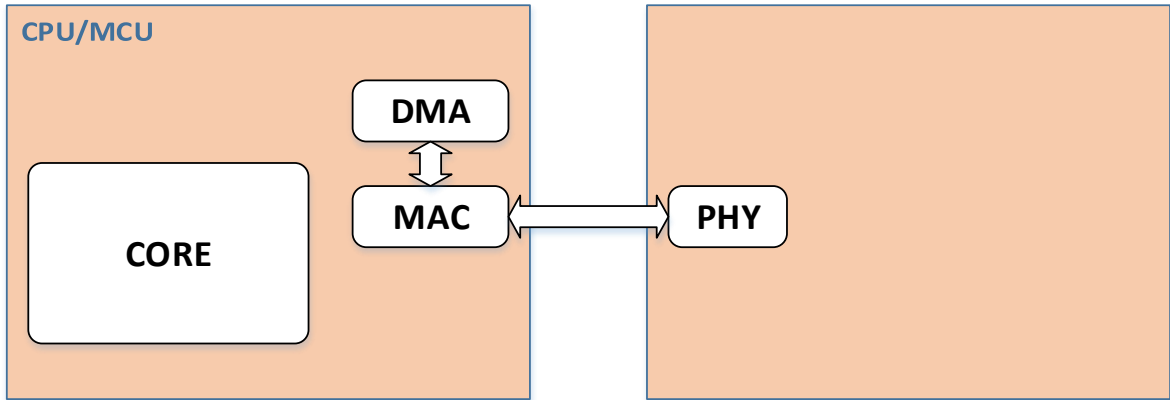


图 3. PHY 采用独立芯片

3. MAC 和 PHY 采用独立芯片，其结构如图 4 所示：

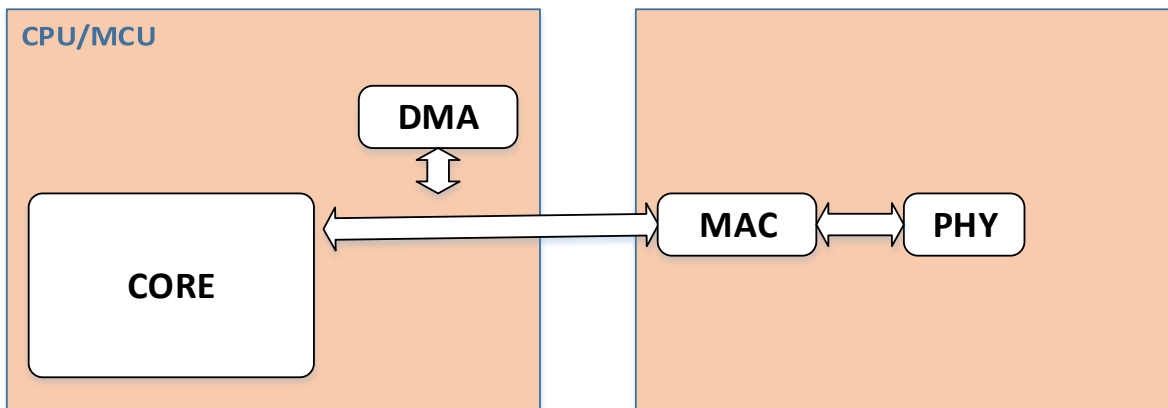


图 4. MAC 和 PHY 采用独立芯片

上面三种结构，大部分处理器将 MAC 集成在了 SOC 内部，只有极少数将 MAC 和 PHY 集成进了 CPU。因此，第二种比较常见，第一种比较少见。

1.3. TI CPSW

TI 的 JACINTO7 里面集成了 CPSW9G 和 CPSW2G。CPSW 是 TI Common Platform Ethernet Switch 的简称。其中 CPSW9G 有 9 个 PORT，一个 CPU PORT，是对外可以提供 8 个 PORT 的交换机。CPSW2G 提供一个 CPU PORT，对外提供一个 PORT。所以在 TI 的 Jacinto7 可以方便地在 CPSW2G 和 CPSW9G 直接在板子上去掉 PHY，直接进行 MAC 互联。另外，TI 的多个 SOC 可以方便地利用 PCIE Switch 和 Ethernet Switch 进行互联。

2. JACINTO7 MAC2MAC 解决方案

2.1. 应用需求

在 Jacinto7 的应用中，由于 Jacinto7 采用多核异构的架构，不仅集成 ARM A72 核，还集成了 TI 的 C7000 和 C6000 DSP，ARM R5F MCU 核等等。多核异构的这种架构不仅给客户提供了优异的性能。同时，也给设计带来了很大的灵活性和便捷性。MAC2MAC 的互联方案，去掉了 PHY，不仅节约了系统成本，而且还减少了通信地时延和板子面积。因此，在当前的系统设计中被广泛采用。

CPSW 的架构如图 5 所示，从 MAC 接口类型来看，我们通用的 RGMII(Reduced Gigabit Media Independent Interface (RGMII), SGMII (Serial GMII) 均可以支持 1000M 以太网。

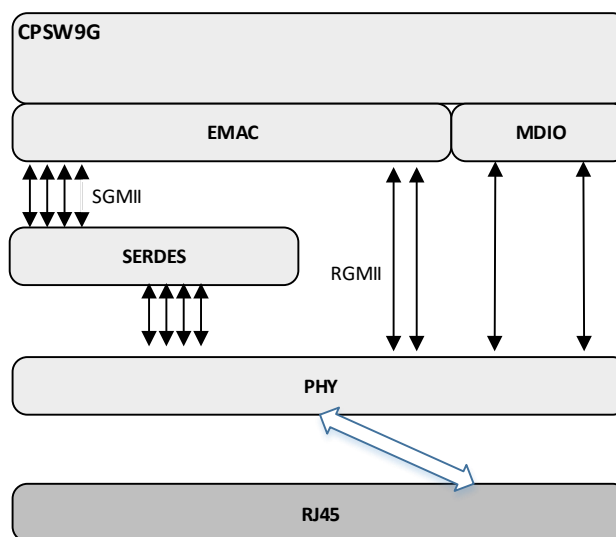


图 5. CWSW Architecture

RGMII 即 Reduced GMII，是 RGMII 的简化版本，时钟频率仍旧为 125MHz，TX/RX 数据宽度从 8 为变为 4 位，为了保持 1000Mbps 的传输速率不变，RGMII 接口在时钟的上升沿和下降沿都采样数据。在参考时钟的上升沿发送 GMII 接口中的 TXD[3:0]/RXD[3:0]，在参考时钟的下降沿发送 GMII 接口中的 TXD[7:4]/RXD[7:4]。RGMII 同时也兼容 100Mbps 和 10Mbps 两种速率，此时参考时钟速率分别为 25MHz 和 2.5MHz。典型的 RGMII 的应用连接如图 6 所示：

Overwrite this text with the Lit. Number

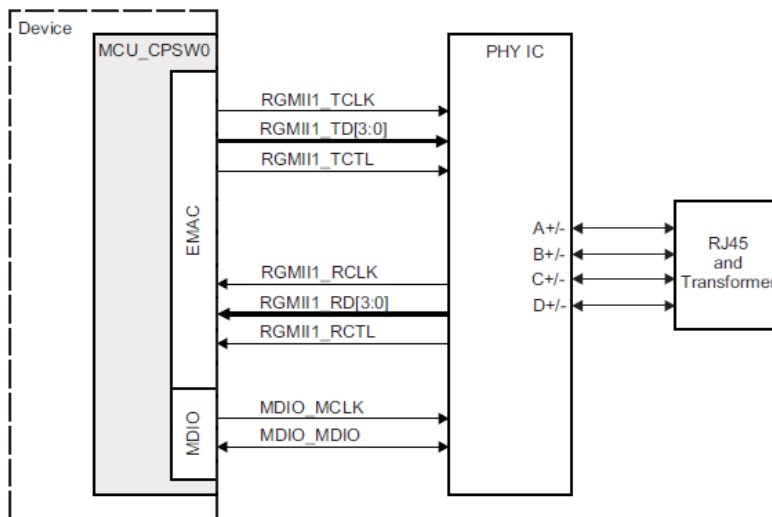


图 6. 典型的 RGMII 连接

SGMII 即 Serial GMII，串行 GMII，收发各一对差分信号线，时钟频率 625MHz，在时钟信号的上升沿和下降沿均采样，参考时钟 RX_CLK 由 PHY 提供，是可选的，主要用于 MAC 侧没有时钟的状况，通常状况下，RX_CLK 不使用，收发均可以从数据中恢复出时钟。大多数 MAC 芯片的 SGMII 接口均可以配置成 SerDes 接口（在物理上兼容，只需配置寄存器便可），直接外接其它模块，而不须要 PHY 层芯片，此时时钟速率仍旧是 625MHz。如 TRM CPSW 图 7 所示，通常 SGMII 可以配置成 Serdes 接口。

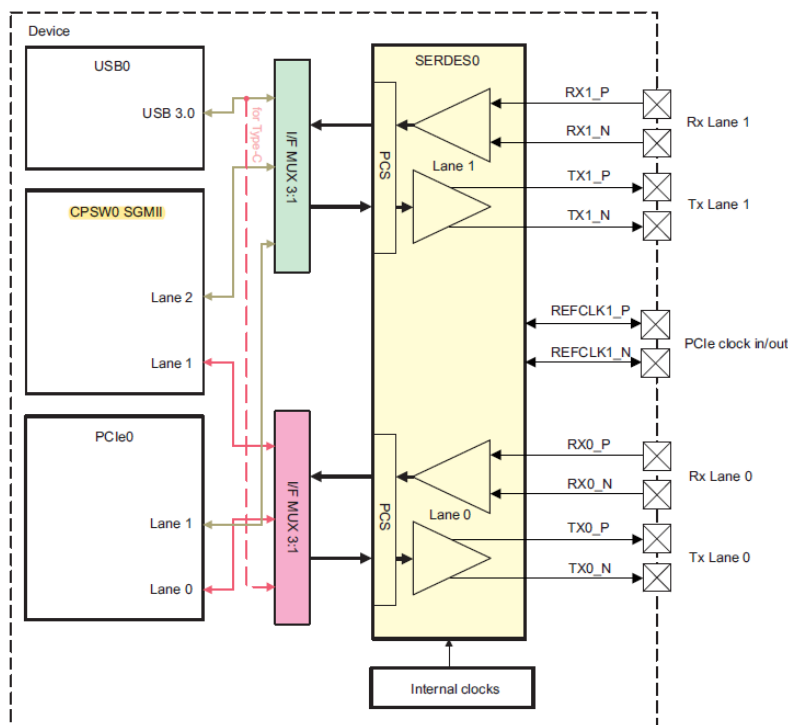


图 7. Serdes0 Overview

this text with the Lit. Number

这里需要注意的是不同的 SGMII 的 PORT 对应不同的 SERDES LANE，TDA4VM CPSW9G 对应的表格如下表 1 所示：

表 1. SGMII PORT SERDES 对应表

SGMII PORT	SERDES - LANE
PORT 1	SERDES 0 - LAN 0
PORT 2	SERDES 0 - LAN 1
PORT 3	SERDES 1 - LAN 0
PORT 4	SERDES 1 - LAN 1
PORT 5	SERDES 4 - LAN 0
PORT 6	SERDES 4 - LAN 1
PORT 7	SERDES 4 - LAN 2
PORT 8	SERDES 4 - LAN 3

因此，MAC2MAC 互联通常也有两种方式，基于 RGMII 的 MAC2MAC 互联和基于 SGMII 的 MAC2MAC 互联。通常由如下三种应用场景。下面我们分别举例子说明具体的应用场景。

1. Jacinto7 里面划分了 ASIL D 的 MCU 域。CPSW2G 通常划分给 MCU 域来使用。CPSW9G 通常给主域 A72 和 R5F 等核使用。MCU 域由于功能安全的要求，通常使用 AUTOSAR OS。除了内部的 Internal Data Path. 从外部，也可以非常方便地通过 MAC2MAC 与交换机相连接入网络。系统的架构图 8 所示：

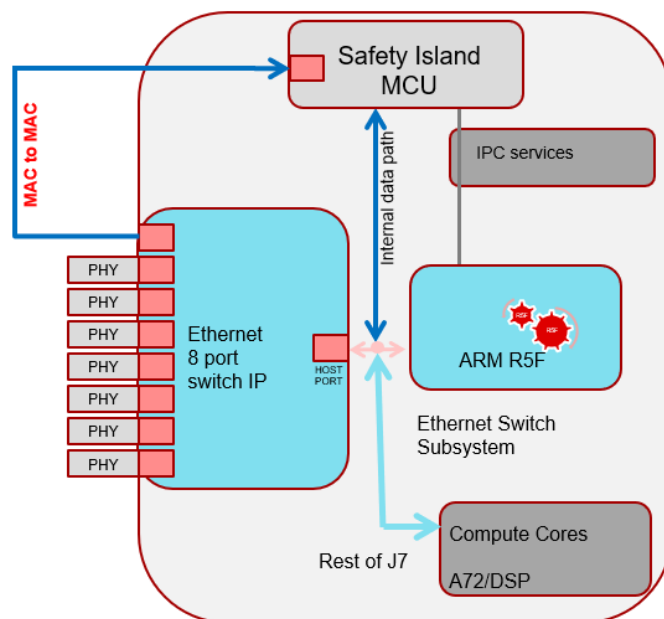


图 8. MAC 和 PHY 采用独立芯片

2. 当单颗 Jacinto7 的处理器算力无法满足系统要求时，由于 TI 内部集成了 Ethernet Switch 和 PCIE Switch, 因此可以非常方便地通过 MAC2MAC 形成多颗 Jactinto7 SOC 板上级联，并进行互联互通。系统框图如图 9 所示：

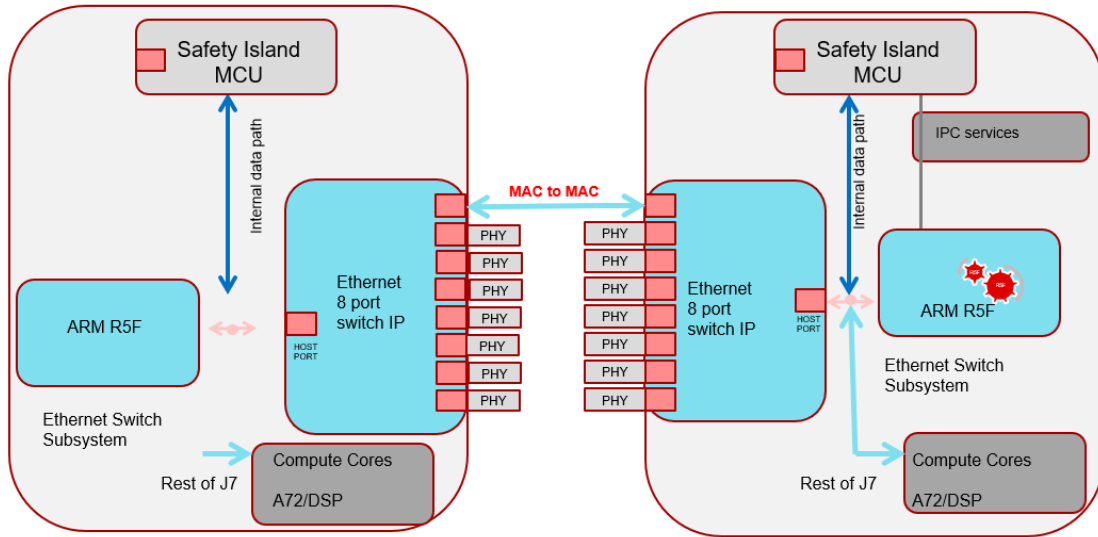


图 9. SOC 间 MAC2MAC 互联

- 3. 除了以上两种情况，我们还可以和外部的网络接口、SOC 通过 MAC2MAC 互联。系统框图如图 10 所示：

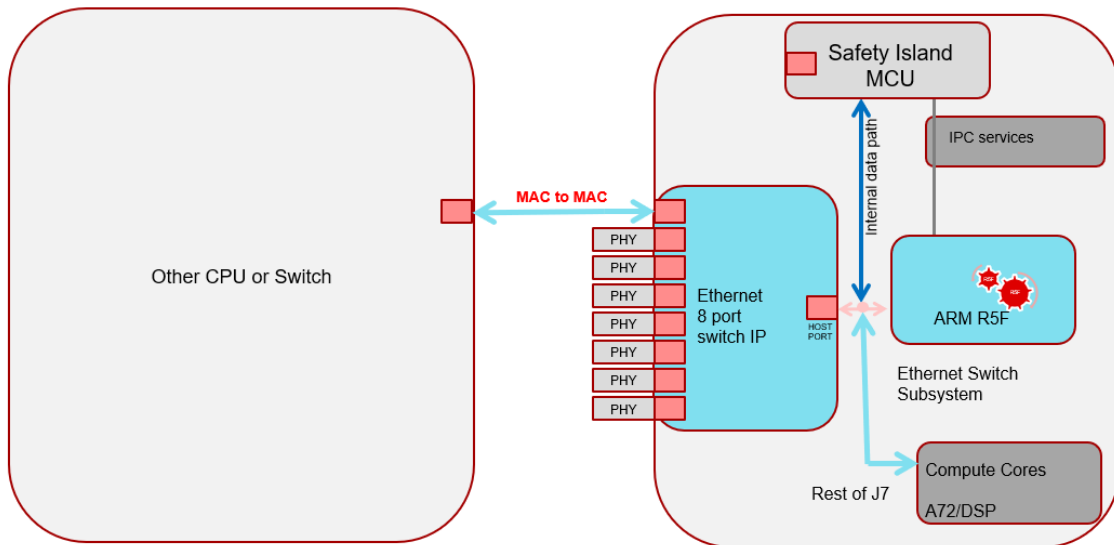


图 10. 外部 Soc 或设备与 TDA4VM 连接

2.2. RGMII 解决方案

这里我们先来看一个基于 CPSW2G 和 CPSW9G 互联的例子。内部 MAC 与 MAC 之间通过 RGMII 互联。RGMII 这种方式相对比 SGMII 比较简单，MAC 跟 MAC 直接相连就行。其连接如图 11 所示：

this text with the Lit. Number

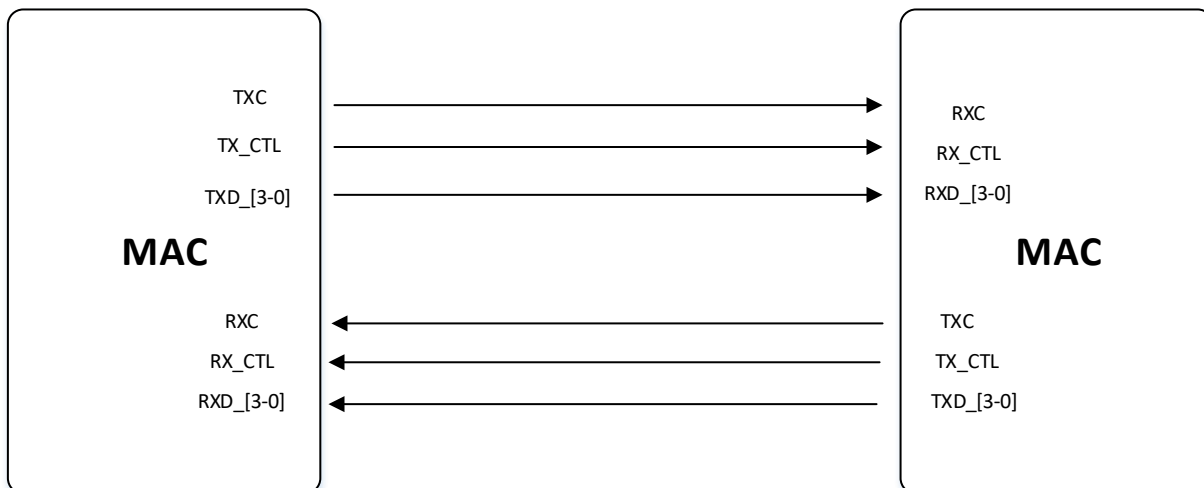


图 11. RGMII MAC2MAC

2.3. SGMII 解决方案

这里我们举一个 SGMII 互联的例子。在两颗 TDA4VM 之间通过 SGMII MAC2MAC 形成互联。SGMII 相比于 RGMII 需要增加 SERDES 的配置。系统框图如图 11 所示。

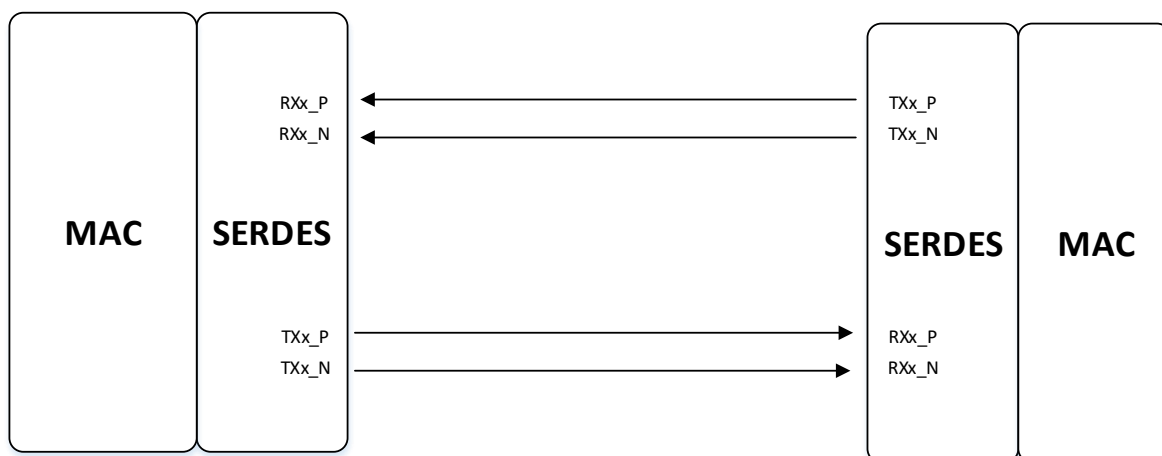


图 12. SGMII MAC2MAC

3. MAC2MAC 实现

3.1. RGMII 实现

这里给出一个 TI 的 CPSW2G 和 CPSW9G 互联的一个例子。CPSW2G 由 A72 控制，CPSW9G 由 Main R5F 控制，通过 MAC2MAC 将 CPSW2G 和 CPSW9G 互联。配置模式 RGMII 全双工 1000M (RGMII-Full Duplex-1000M)。在 TI 默认的 PSDKRA 里面，CPSW2G 和 CPSW9G 各有一个 Linux 以太网的节点。MAC 正常会连接 PHY，因此我们的主要配置就是去掉 PHY。

Overwrite this text with the Lit. Number

CPSW2G No PHY 模式的 DTB (arch/arm64/boot/dts/ti/k3-j721e-common-proc-board.dts) 主要改动如下:

1. 去掉 MDIO 的配置:

```
// &davinci_mdio {
//     phy0: ethernet-phy@0 {
//         reg = <0>;
//         ti,rx-internal-delay = <DP83867_RGMIIDCTL_2_00_NS>;
//         ti,fifo-depth = <DP83867_PHYCR_FIFO_DEPTH_4_B_NIB>;
//     };
// };
```

2. CPSW PORT 的配置去掉 PHY。

```
&cpsw_port1 {
    phy-mode = "rgmii-rxid";
    fixed-link {
        speed = <1000>;
        full-duplex;
    };
    // phy-handle = <&phy0>;
};
```

CPSW9G No PHY 配置模式方法如下: CPSW9G(RGMII3)

1. 打开对应 port
2. 把对应 port 的 phyAddr 配成 CPSW_PHY_INVALID_PHYADDR
3. 配置对应 port 的工作模式和速率 (RGMII-FULLDUPLEX-1000M)

关键代码 PATCH (CPSW9G-MAC-RGMII.patch) [下载地址](#)。

3.2. SGMII 实现

这里给出一个 TI 的 CPSW9G 到 CPSW9G 两颗 TDA4VM 互联的一个例子。CPSW9G 由 TI 的 ETHFW 进行管理。通过 SGMII MAC2MAC 将一个 TDA4VM 的 CPSW9G 和连接到另外一个 TDA4VM 的 CPSW9G。在 SGMII 的例子里,相比于 RGMII,我们还要配置 SERDES。配置流程如下:

1. 设置 SERDES 的 clock 和对 SERDES 进行配置。
2. 打开对应的 PORT。
3. 把对应的 PORT 的 phyAddr 配置成 CPSW_PHY_INVALID_PHYADDR。
4. 配置对应的模式 master 或 slave。

CPSW9G SERDES 对应情况如下所示:

```
* PORT 1 -SERDES 0 Lan0
* PORT 2 -SERDES 0 Lan1
* PORT 3 -SERDES 1 Lan0
* PORT 4 -SERDES 1 Lan1
* PORT 5 -SERDES 4 Lan0
```

this text with the Lit. Number

```
* PORT 6 -SERDES 4 Lan1
* PORT 7 -SERDES 4 Lan2
* PORT 8 -SERDES 4 Lan3
```

SGMII4 进行配置 MAC 的关键代码 (sgmii4_configure_patch.patch) 请从这里[下载](#).

3.3. MAC2MAC 调试

在调试过程中难免遇到问题，下面给出一些调试建议。

1. RGMII 调试

- a. 确认寄存器 CTRLMMR_ENETx_CTRL 对应 PORT 模式设置 PORT_MODE_SEL 是 RGMII。
- b. 确认对应 port 的 phyAddr 配置成 CPSW_PHY_INVALID_PHYADDR
- c. 确认寄存器 CPSW_SS_RGMIIx_STATUS_REG 对应的 RGMII FULLDUPLEX/SPEED/LINK 设置，确保 Link is up。

2. SGMII 调试

- a. 确认寄存器 CTRLMMR_ENETx_CTRL 对应 PORT 模式设置 PORT_MODE_SEL 是 SGMII。
- b. 确认对应 port 的 phyAddr 配置成 CPSW_PHY_INVALID_PHYADDR
- c. 确认 CPSW_SGMII_CONTROL_REG_j 寄存器的 MASTER mode 配置
- d. 确认 CPSW_SGMII_STATUS_REG_j 寄存器 SERDES PLL LOCK and LINK 状态。

下载检查 [CPSW2G](#) (j7_devmem2_cpsw2g_check_status.sh) 和 [CPSW9G](#) (j7_devmem2_cpsw9g_check_status.sh) 的脚本。

4. 总结

MAC2MAC 的连接广泛使用在 TI Jacinto7 处理器中。常用的解决方案有两种，一种是基于 RGMII 的解决方案，另外一种是基于 SGMII 的解决方案。本文重点针对这两个解决方案进行了阐述。同时也对应用场景和 MAC2MAC 的实现进行了详细介绍。

针对这两种方案，RGMII 的方案相比 SGMII 的方案：使用的 IO 数量会多一些，内部连接少了 SERDES，因此配置也会容易一些。这两种方案都可以支持全双工 1000M 以太网。使用 SGMII 的方案的时候要注意 SERDES 的资源是否冲突。

在实际的应用中，在 IO 口资源充裕的情况下，我们推荐优先使用基于 RGMII 的 MAC2MAC 解决方案。这种方案由于少了 SERDES 的部分，配置简单，易于使用。

5. 参考

1. http://software-dl.ti.com/processor-sdk-linux/esd/docs/latest/linux/Foundational_Components/Kernel/Kernel_Drivers/Network/CPSW.html
2. https://en.wikipedia.org/wiki/OSI_model
3. <https://www.ti.com/lit/zip/spru11>

重要声明和免责声明

TI 提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他安全、安保或其他要求。这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的应用。严禁对这些资源进行其他复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 TI 的销售条款 (<https://www.ti.com.cn/zh-cn/legal/termsofsale.html>) 或 [ti.com.cn](https://www.ti.com.cn) 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

邮寄地址：上海市浦东新区世纪大道 1568 号中建大厦 32 楼，邮政编码：200122
Copyright © 2021 德州仪器半导体技术（上海）有限公司