

Design Guide: TIDA-020071

适用于 Jacinto 7 处理器的 100/1000Base-T1 4 × 汽车以太网 PHY 扩展参考设计



说明

该参考设计通过 RGMII 以太网扩展连接器与 Jacinto™ 7 处理器 EVM 板连接。通过 TI 的汽车以太网物理层 (PHY) 添加四个汽车以太网连接。该设计展示了采用 TI DP83TC818S-Q1 100Mbps 和 DP83TG721S-Q1 1000Mbps 单线对以太网 (SPE) PHY 的实现方案。耦合网络用于将 12V 电压耦合到数据线上。

资源

TIDA-020071	设计文件夹
DP83TC817S-Q1、DP83TC818S-Q1	产品文件夹
DP83TG720R-Q1、DP83TG721S-Q1	产品文件夹
TPS74701-Q1、TPS7B82-Q1	产品文件夹
TPS4H160-Q1	产品文件夹

特性

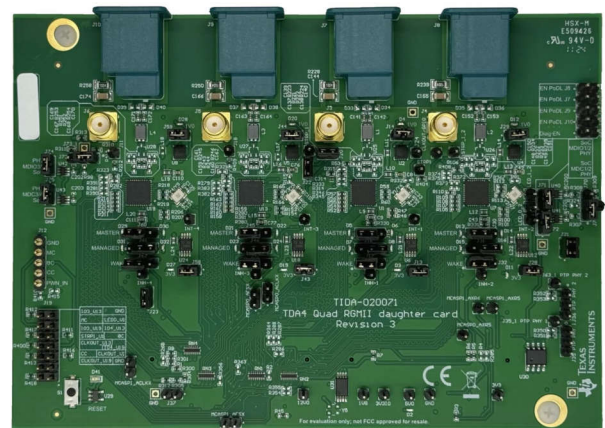
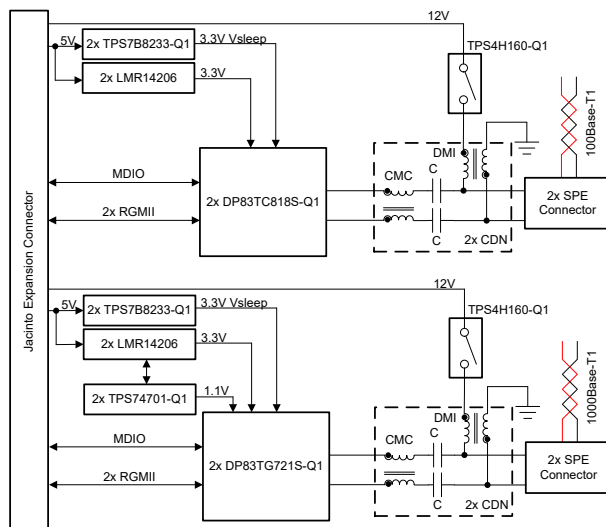
- 通过 RGMII 连接到 Jacinto™ 7 处理器的 100Base-T1 和 1000Base-T1 汽车以太网
- 可以选择为全部四个以太网 PHY 提供相同的时钟
- 用于组装 TI BAW 振荡器的选项
- 适用于 12V 电源耦合且符合汽车标准的滤波器网络
- 用于存储识别信息的电路板 ID EEPROM

应用

- 高级驾驶辅助系统 (ADAS)
- 车身电子装置和照明
- 信息娱乐系统与仪表组
- 软件定义车辆



请咨询 TI E2E™ 支持专家



1 系统说明

Jacinto™ 7 EVM 是可帮助开发人员围绕 Jacinto™ 7 系列处理器编写软件和开发硬件的开发和评估平台。该 EVM 提供了该系统的主要部分，从而为开发人员提供了与 Jacinto™ 7 处理器相关的大多数通用工程所需的基本资源。除了提供的基本资源，还可通过扩展卡增加其他功能。

该参考设计通过 TI 的汽车以太网 PHY 添加了四个汽车以太网连接。DP83TC818S-Q1 汽车以太网 PHY 支持 100Mbps 链路速度，符合 IEEE 802.3bw 和 OA 100BASE-T1 标准，支持 IEEE 1588v2、802.1AS 时间同步，支持 IEEE 1722 CRF 数据包解码和音频/视频桥接 (AVB) 媒体时钟生成功能，并支持 IEEE 802.1AE MACsec。DP83TG721 汽车以太网 PHY 支持 1000Mbps 链路速度，符合 IEEE 802.3bp 标准，支持 IEEE 1588v2、802.1AS 时间同步，并具有 AVB IEEE 1722 媒体时钟生成功能。

1.1 主要系统规格

TIDA-020071 设计的主要特性包括：

- 100Base-T1 和 1000Base-T1 汽车以太网
- 12V 可以耦合在汽车以太网电缆上
- DP83TC818S-Q1 1.0V 由 PHY 的内部 LDO 供电
- 可以选择通过时钟缓冲器从 TI BAW 振荡器向全部四个以太网 PHY 提供相同的时钟
- 用于组装 TI BAW 振荡器的选项
- 用于存储识别信息的电路板 ID EEPROM

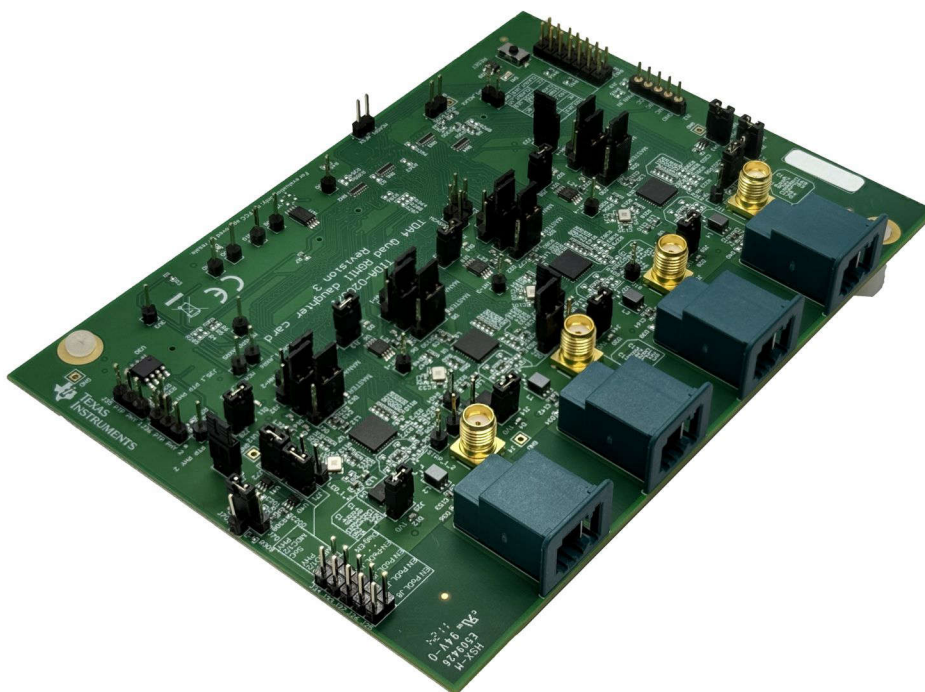


图 1-1. TIDA-020071 板

2 系统概述

图 2-1 展示了系统包含一个电路板，一侧通过扩展连接器连接到 Jacinto™ 7 处理器 EVM，另一侧通过双绞线电缆连接到最多四个链路伙伴。

Jacinto™ 7 EVM 通过扩展连接器为降压转换器和 LDO 提供 5V 电源，从而实现 3.3V Vsleep。该 EVM 还为差模电感器提供 12V 电压，从而将电力耦合到汽车以太网电缆。

2.1 方框图

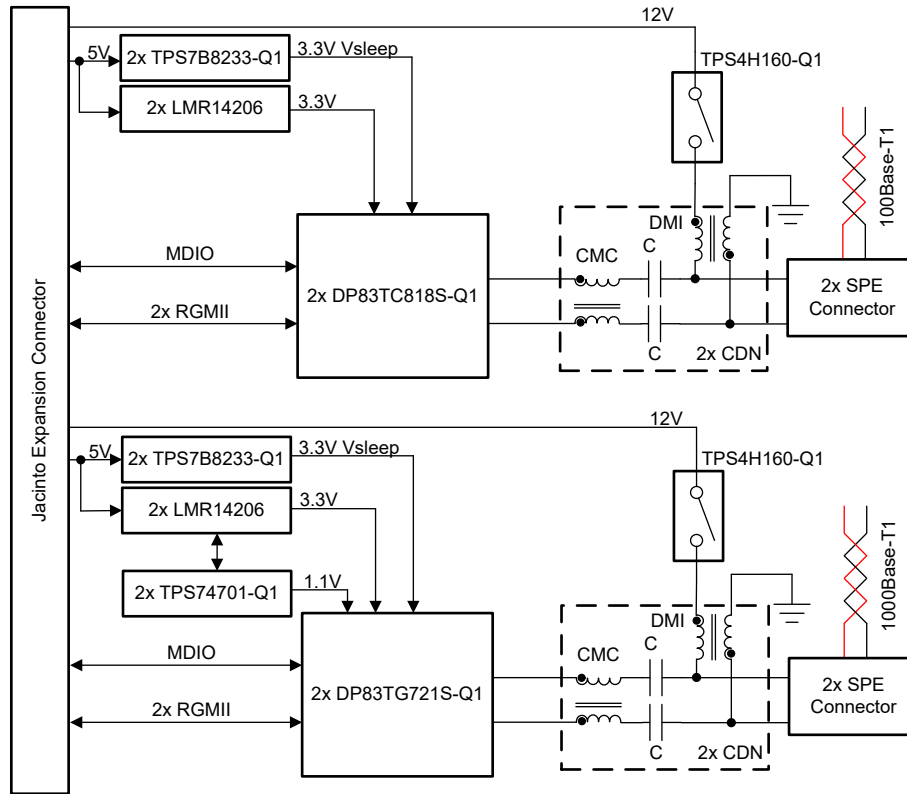


图 2-1. TIDA-020071 方框图

2.2 设计注意事项

该参考设计展示了包含单线对以太网 (SPE) PHY 的以太网接口实现，并提供了通过同一电缆为链路伙伴供电的选项。这是数据线供电 (PoDL) 的一部分。尽管 PoDL 在 IEEE802.3bu 中针对 100/1000Base-T1 进行了标准化，但该参考设计并不遵循 IEEE802.3bu，而是为了满足汽车要求。在汽车领域，汽车 OEM 定义了系统中使用哪些 ECU 和传感器模块，因此不需要进行检测和分类。此假设基于这样一个事实，即网络在车辆的生命周期内不会发生变化。这些电路板专为满足汽车要求而设计，可实现具有成本效益且轻便的实施。

由于电力和数据使用同一根电缆进行传输，传统的保险丝不再适用，因此强烈建议集成高侧开关 (HSS)，以保护链路伙伴免受损坏。Jacinto™ 7 EVM 上由 LM5175 降压/升压转换器生成用于将电力耦合到数据线的 12V 电源。该器件已经支持逐周期电流限制、输出过压保护 (OVP) 和热关断等保护功能。该参考设计包含 TPS4H160-Q1 (高侧开关)，其中包括可调节电流限制、对 GND 短路保护、热关断和电感负载负电压钳位。

2.3 主要产品

本节介绍此参考设计中的重要器件。有关完整的详细信息，请参阅各个器件的产品页面和数据表。

2.3.1 DP83TC818S-Q1 (汽车 SPE PHY)

DP83TC818S-Q1 器件是一款符合 IEEE 802.3bw 标准的汽车以太网物理层收发器。它提供发送和接收数据所需的所有物理层功能，以及 xMII 接口灵活性。DP83TC818S-Q1 符合 Open Alliance EMC 和非屏蔽单根双绞线电缆互操作规范。DP83TC818S-Q1 支持 OA TC-10 低功耗睡眠特性，具有唤醒转发功能，可在无需通信时降低系统功耗。

DP83TC818S-Q1 将 IEEE 802.1AE 线速安全性与身份验证和可选加密支持相集成，可确保网络通信安全。DP83TC818S-Q1 支持多达 16 个安全关联 (SA) 通道，并支持 SAK 自动翻转和扩展数据包编号。DP83TC818S-Q1 提供入口分类来过滤不需要的数据包，并支持 WAN MACsec 以实现端到端安全性。

DP83TC818S-Q1 集成了 IEEE 1588v2、802.1AS 硬件时间戳和分数 PLL，可实现高度精确的时间同步。分数 PLL 可实现挂钟的频率和相位同步 (无需外部 VCXO)，并可生成音频和其他 ADAS 应用所需的各种时间同步频率。PHY 还集成了 IEEE 1722 CRF 解码，可为 AVB 和其他音频应用生成媒体时钟和位时钟。

2.3.2 TPS7B8233-Q1 (3.3V Vsleep 超低 IQ 低压降稳压器)

在汽车电池连接应用中，低静态电流 (I_Q) 对于省电和延长电池寿命而言至关重要。对于始终开启的系统，必须要实现超低 I_Q 。

TPS7B82-Q1 是一款旨在 3V 至 40V (45V 负载突降保护) 宽输入电压范围内运行的低压降线性稳压器。TPS7B82-Q1 的工作电压低至 3V，因此可在冷启动以及启动和停止情况下继续工作。该器件在轻负载时的典型静态电流仅为 2.7 μ A，非常适合用于为待机系统中的微控制器 (MCU) 和 CAN 或 LIN 收发器供电。

这些器件具有集成的短路和过流保护功能。该器件可在 -40°C 至 +125°C 的环境温度下运行，结温范围为 -40°C 至 +150°C。此外，该器件采用了热传导封装，即使整个器件散热较多，也能实现持久运行。由于这些特性，该器件旨在用作各种汽车应用的电源。

2.3.3 TPS74701-Q1 (1.0V 电源轨低压降稳压器)

TPS74701-Q1 低压降 (LDO) 线性稳压器可面向多种应用提供易于使用的稳健型电源管理设计。用户可编程软启动通过减少容性浪涌电流，更大限度地减少了输入电源上的应力，并且单调启动旨在为多种不同类型的处理器和应用特定集成电路 (ASIC) 供电。借助使能输入和电源正常输出，可通过外部稳压器轻松进行时序控制，因此可为各种具有特殊启动要求的应用配置满足其时序控制要求的设计。

该器件还具有高精度的参考电压电路和误差放大器，可在整个负载、线路、温度和过程范围内提供 0.95% 精度。该器件在使用大于或等于 2.2 μ F 的任何类型的电容器时均能保持稳定，并完全符合 AEC-Q100 标准。

2.3.4 CDC6CE025000-Q1 (BAW 振荡器)

德州仪器 (TI) 的体声波 (BAW) 是一种微谐振器技术，能够将高精度 BAW 谐振器直接集成到具有超低抖动时钟电路的封装中。与其他硅基制造工艺一样，BAW 完全由 TI 工厂设计和制造。

CDC6x 器件是一款超低抖动固定频率振荡器，融合了 BAW 作为谐振器源。该器件根据特定运行模式进行出厂编程，包括频率、电压、输出类型和功能引脚。CDC6x 带有高性能分数分频器，能够产生指定范围内的任何频率，提供可满足所有频率需求的单个器件系列。

凭借高性能时钟、机械稳定性、灵活性和小型封装选项，此器件非常适用于电信、数据以及企业网络和工业应用中使用的高速 SERDES 内的参考时钟和核心时钟。

2.3.5 TPS4H160-Q1 (高侧开关)

TPS4H160-Q1 器件是一款受到全面保护的四通道智能高端开关，具有四个集成的 160mΩ NMOS 功率 FET。

该器件包含丰富的诊断功能以及高精度电流检测功能，从而能够对负载实施智能控制。该器件可从外部调节电流限值以限制浪涌或过载电流，从而提升整个系统的可靠性。

3 系统设计原理

本节提供有关该参考设计的不同部分的详细信息。

3.1 以太网 PHY

该电路板支持各种 TI 汽车 SPE PHY。该电路板支持 100Base-T1 和 1000Base-T1。默认情况下，该板使用两个 100Base-T1 系统，每个系统包含 DP83TC818S-Q1 和两个 DP83TG721S-Q1 SPE PHY (其地址如表 3-1 所示)。汽车以太网 PHY DP83TC818S-Q1 和 DP83TG721S-Q1 以及其他 TI SPE PHY 引脚对引脚兼容，因此只需进行少量 BOM 更改即可互换。

表 3-1. 以太网 PHY 地址

以太网 PHY	指示符	IEEE 标准	PHY 地址
DP83TC818S-Q1	U13	100Base-T1	0x5
DP83TC818S-Q1	U19	100Base-T1	0x8
DP83TG721S-Q1	U1	1000Base-T1	0x0
DP83TG721S-Q1	U7	1000Base-T1	0x4

3.1.1 以太网 PHY 电源

DP83TC818S-Q1 能在宽 I/O 电源电压范围 (3.3V、2.5V 或 1.8V) 内运行。该板具有 3.3V 的 IO 电源电压以连接各种能提供 3.3V I/O 电压的基板。DP83TC818S-Q1 也需要 1.0V 电源轨。图 3-1 所示的实现原理图允许使用 DP83TC818S-Q1 集成 LDO 来产生所需的 1.0V 电压。不需要电源时序控制。有关最新的电源器件建议，请查看并遵循 DP83TC818S-Q1 具有 TC10、IEEE802.1AS、IEEE802.1AE MACsec 和 AVB 时钟生成功能的精密安全型 100BASE-T1 汽车以太网数据表。

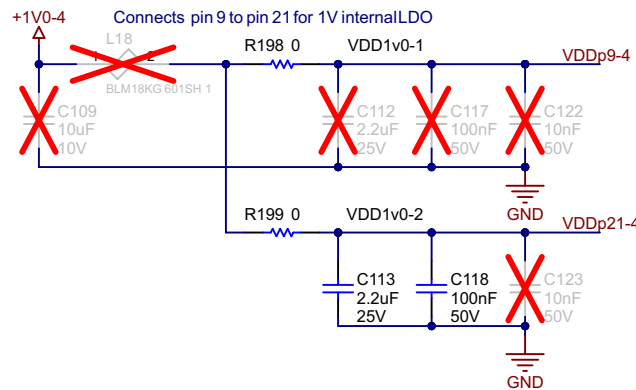


图 3-1. 以太网 PHY 1.0V 电源轨原理图

3.1.2 以太网 PHY 时钟源

图 3-2 中所示的原理图将 TI 的 BAW 振荡器与时钟缓冲器结合使用，能够为该参考设计上的全部四个以太网 PHY 提供相同的时钟源。该实现方案有助于同步，随后链路伙伴可使用以太网 PHY 的恢复时钟功能来减少时间同步系统的抖动。

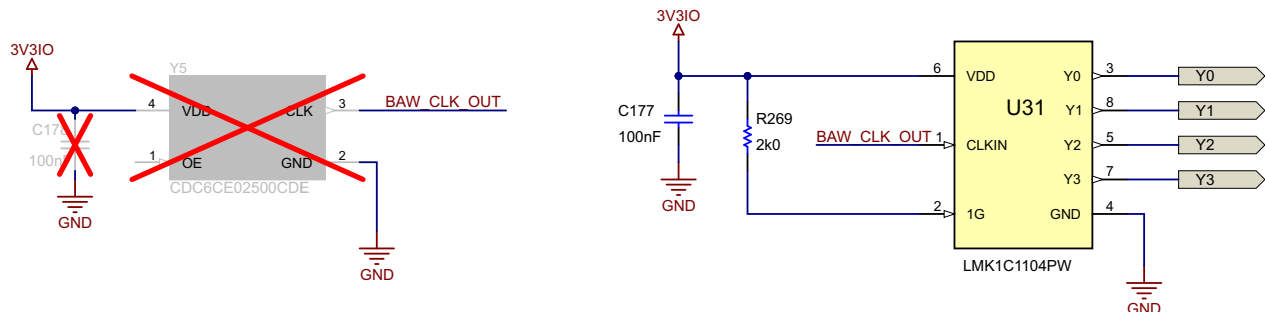


图 3-2. 以太网 PHY 时钟原理图

3.2 电源耦合网络

选择耦合网络时需要考虑两个主要因素：链路伙伴的以太网数据速率和功耗。表 3-2 展示了每个以太网 PHY 以及相应的以太网标准和数据频率。

表 3-2. 以太网信号频率

以太网 PHY	IEEE 标准	数据速率	符号速率
—	10Base-T1S	10Mbps	12.5MHz
DP83TC812-Q1	100Base-T1	100Mbps	66.6MHz
DP83TC813-Q1			
DP83TC814-Q1			
DP83TC817-Q1			
DP83TC818-Q1			
DP83TG720-Q1	1000Base-T1	1000Mbps	750MHz
DP83TG721-Q1			

根据元件的饱和特性，每个耦合网络在保持所需阻抗的同时可以支持的最大电流是一定的。电感器的行为不是完美的，电感器会散热，传递非常高的频率，并在流过多电流时达到饱和。整个电路中的所有元件和电缆都具有寄生电容和电感。了解链路伙伴从电源线获取的最大功率并选择能够在给定电压下提供该功率的元件非常重要。计算最大功率，作为链路伙伴侧功耗的最坏情况。

表 3-3 提供了耦合网络设计建议。TI 建议根据所需的频率范围、额定电流和温度选择其中某个网络。

表 3-3. 适用于 PoDL 的 CDN 滤波器设计

供应商	以太网速度	元件	电流额定值	温度额定值
TDK	1000Base-T1	CMC : ACT1210G-800 DMI : ADL32VHR-3R9M	540mA	150°C
Murata	1000Base-T1	CMC : DLW32MH101XK2	—	—
TDK	100Base-T1	CMC : ACT1210L-201 DMI : ADL32VHR-180M	370mA	150°C
Murata	100Base-T1	CMC : DLW32MH201XK2	—	—

3.2.1 高侧开关

TPS4H160-Q1 四通道高侧开关与跳线 J25、J26、J27 和 J33 配合使用，用于启用或禁用 12V 到数据线的耦合。安装跳线以启用输出。该开关还提供对 GND 短路和热关断保护。

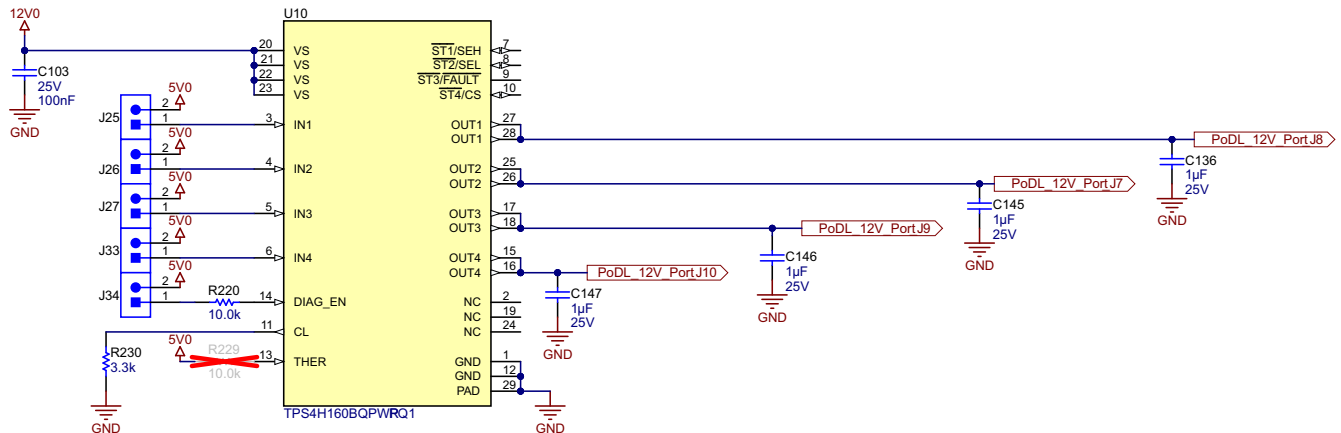


图 3-3. 高侧开关原理图

4 硬件、软件、测试要求和测试结果

本节介绍了在极简设置下演示汽车以太网功能的基本硬件和软件要求。该设置允许测量 IP 网络上可实现的最大带宽。

4.1 硬件要求

表 4-1 列出了所需的硬件。

表 4-1. 极简设置所需的硬件

数量	器件说明	器件型号
1 ×	4 × 汽车以太网 PHY 参考设计	TIDA-020071
1 ×	Jacinto 7 SoM 板	J721EXSOMXEVM
1 ×	用于 Jacinto 7 处理器的通用处理器板	J721EXCPXEVM
1 ×	100Mbps/s 汽车到标准以太网媒体转换器	DP83TC812EVM-MC
1 ×	1000Mbps/s 汽车到标准以太网媒体转换器	DP83TG721EVM-MC
1 ×	TE Connectivity MATEnet SPE 电缆	—
1 ×	RJ45 电缆	—
1 ×	12V、5A 电源	—

4.2 软件要求

本节介绍极简设置的软件要求。适用于 Jacinto 7 处理器的软件开发套件 (SDK) 用于在测试期间控制和监测此参考设计 (如图 4-1 所示) 上的以太网 PHY。

将汽车以太网 PHY 驱动程序添加到 Linux SDK，从而使 Jacinto 7 处理器可以识别以太网 PHY。另请参阅[如何将 Linux 驱动程序集成到您的系统中](#)应用手册。

Jacinto 7 SDK v10 的以下代码块中显示了此特定板的 Linux 器件树叠加层，其中 PHY 配置为 PHY 地址 0x0、0x4、0x5 和 0x8。有关其他 SDK 版本，请参阅 SDK 文档。

```

/dts-v1/;
/plugin/;

#include <dt-bindings/gpio/gpio.h>
#include <dt-bindings/net/ti-dp83867.h>
#include "k3-pinctrl.h"

&{/} {
    aliases {
        ethernet1 = "/bus@100000/ethernet@c000000/ethernet-ports/port@1";
        ethernet2 = "/bus@100000/ethernet@c000000/ethernet-ports/port@2";
        ethernet3 = "/bus@100000/ethernet@c000000/ethernet-ports/port@3";
        ethernet4 = "/bus@100000/ethernet@c000000/ethernet-ports/port@4";
    };
};

&cpsw0 {
    status = "okay";
    pinctrl-names = "default";
    pinctrl-0 = <&rgmii1_pins_default
        &rgmii2_pins_default
        &rgmii3_pins_default
        &rgmii4_pins_default>;
};

&cpsw0_port1 {
    status = "okay";
    phy-handle = <&cpsw9g_phy0>;
    phy-mode = "rgmii-rxid";
    mac-address = [00 00 00 00 00 00];
    phys = <&cpsw0_phy_gmii_sel 1>;
};

&cpsw0_port2 {
    status = "okay";
}
    
```



```

phy-handle = <&cpsw9g_phy4>;
phy-mode = "rgmii-rxid";
mac-address = [00 00 00 00 00 00];
phys = <&cpsw0_phy_gmii_sel 2>;
};

&cpsw0_port3 {
status = "okay";
phy-handle = <&cpsw9g_phy5>;
phy-mode = "rgmii-rxid";
mac-address = [00 00 00 00 00 00];
phys = <&cpsw0_phy_gmii_sel 3>;
};

&cpsw0_port4 {
status = "okay";
phy-handle = <&cpsw9g_phy8>;
phy-mode = "rgmii-rxid";
mac-address = [00 00 00 00 00 00];
phys = <&cpsw0_phy_gmii_sel 4>;
};

&cpsw9g_mdio {
status = "okay";
pinctrl-names = "default";
pinctrl-0 = <&mdio0_pins_default>;
bus_freq = <1000000>;
#address-cells = <1>;
#size-cells = <0>;

cpsw9g_phy0: ethernet-phy@0 {
reg = <0>;
};
cpsw9g_phy4: ethernet-phy@4 {
reg = <4>;
};
cpsw9g_phy5: ethernet-phy@5 {
reg = <5>;
};
cpsw9g_phy8: ethernet-phy@8 {
reg = <8>;
};
};

&main_pmx0 {
mdio0_pins_default: mdio0-pins-default {
pinctrl-single,pins = <
J721E_IOPAD(0x1bc, PIN_OUTPUT, 0) /* (V24) MDIO0_MDC */
J721E_IOPAD(0x1b8, PIN_INPUT, 0) /* (V26) MDIO0_MDIO */
>;
};

rgmii1_pins_default: rgmii1-pins-default {
pinctrl-single,pins = <
J721E_IOPAD(0x4, PIN_INPUT, 4) /* (AC23) PRG1_PRU0_GPO0.RGMII1_RD0 */
J721E_IOPAD(0x8, PIN_INPUT, 4) /* (AG22) PRG1_PRU0_GPO1.RGMII1_RD1 */
J721E_IOPAD(0xc, PIN_INPUT, 4) /* (AF22) PRG1_PRU0_GPO2.RGMII1_RD2 */
J721E_IOPAD(0x10, PIN_INPUT, 4) /* (AJ23) PRG1_PRU0_GPO3.RGMII1_RD3 */
J721E_IOPAD(0x1c, PIN_INPUT, 4) /* (AD22) PRG1_PRU0_GPO6.RGMII1_RXC */
J721E_IOPAD(0x14, PIN_INPUT, 4) /* (AH23) PRG1_PRU0_GPO4.RGMII1_RX_CTL */
J721E_IOPAD(0x30, PIN_OUTPUT, 4) /* (AF24) PRG1_PRU0_GPO11.RGMII1_TD0 */
J721E_IOPAD(0x34, PIN_OUTPUT, 4) /* (AJ24) PRG1_PRU0_GPO12.RGMII1_TD1 */
J721E_IOPAD(0x38, PIN_OUTPUT, 4) /* (AG24) PRG1_PRU0_GPO13.RGMII1_TD2 */
J721E_IOPAD(0x3c, PIN_OUTPUT, 4) /* (AD24) PRG1_PRU0_GPO14.RGMII1_TD3 */
J721E_IOPAD(0x44, PIN_OUTPUT, 4) /* (AE24) PRG1_PRU0_GPO16.RGMII1_TXC */
J721E_IOPAD(0x40, PIN_OUTPUT, 4) /* (AC24) PRG1_PRU0_GPO15.RGMII1_TX_CTL */
>;
};

rgmii2_pins_default: rgmii2-pins-default {
pinctrl-single,pins = <
J721E_IOPAD(0x58, PIN_INPUT, 4) /* (AE22) PRG1_PRU1_GPO0.RGMII2_RD0 */
J721E_IOPAD(0x5c, PIN_INPUT, 4) /* (AG23) PRG1_PRU1_GPO1.RGMII2_RD1 */
J721E_IOPAD(0x60, PIN_INPUT, 4) /* (AF23) PRG1_PRU1_GPO2.RGMII2_RD2 */
J721E_IOPAD(0x64, PIN_INPUT, 4) /* (AD23) PRG1_PRU1_GPO3.RGMII2_RD3 */
J721E_IOPAD(0x70, PIN_INPUT, 4) /* (AE23) PRG1_PRU1_GPO6.RGMII2_RXC */
J721E_IOPAD(0x68, PIN_INPUT, 4) /* (AH24) PRG1_PRU1_GPO4.RGMII2_RX_CTL */
>;
};

```

```

        J721E_IOPAD(0x84, PIN_OUTPUT, 4) /* (AJ25) PRG1_PRU1_GPO11.RGMII2_TD0 */
        J721E_IOPAD(0x88, PIN_OUTPUT, 4) /* (AH25) PRG1_PRU1_GPO12.RGMII2_TD1 */
        J721E_IOPAD(0x8c, PIN_OUTPUT, 4) /* (AG25) PRG1_PRU1_GPO13.RGMII2_TD2 */
        J721E_IOPAD(0x90, PIN_OUTPUT, 4) /* (AH26) PRG1_PRU1_GPO14.RGMII2_TD3 */
        J721E_IOPAD(0x98, PIN_OUTPUT, 4) /* (AJ26) PRG1_PRU1_GPO16.RGMII2_TXC */
        J721E_IOPAD(0x94, PIN_OUTPUT, 4) /* (AJ27) PRG1_PRU1_GPO15.RGMII2_TX_CTL */
    };
};

rgmii3_pins_default: rgmii3-pins-default {
    pinctrl-single,pins = <
        J721E_IOPAD(0xb0, PIN_INPUT, 4) /* (AF28) PRG0_PRU0_GPO0.RGMII3_RD0 */
        J721E_IOPAD(0xb4, PIN_INPUT, 4) /* (AE28) PRG0_PRU0_GPO1.RGMII3_RD1 */
        J721E_IOPAD(0xb8, PIN_INPUT, 4) /* (AE27) PRG0_PRU0_GPO2.RGMII3_RD2 */
        J721E_IOPAD(0xbc, PIN_INPUT, 4) /* (AD26) PRG0_PRU0_GPO3.RGMII3_RD3 */
        J721E_IOPAD(0xc8, PIN_INPUT, 4) /* (AE26) PRG0_PRU0_GPO6.RGMII3_RXC */
        J721E_IOPAD(0xc0, PIN_INPUT, 4) /* (AD25) PRG0_PRU0_GPO4.RGMII3_RX_CTL */
        J721E_IOPAD(0xdc, PIN_OUTPUT, 4) /* (AJ28) PRG0_PRU0_GPO11.RGMII3_TD0 */
        J721E_IOPAD(0xe0, PIN_OUTPUT, 4) /* (AH27) PRG0_PRU0_GPO12.RGMII3_TD1 */
        J721E_IOPAD(0xe4, PIN_OUTPUT, 4) /* (AH29) PRG0_PRU0_GPO13.RGMII3_TD2 */
        J721E_IOPAD(0xe8, PIN_OUTPUT, 4) /* (AG28) PRG0_PRU0_GPO14.RGMII3_TD3 */
        J721E_IOPAD(0xf0, PIN_OUTPUT, 4) /* (AH28) PRG0_PRU0_GPO16.RGMII3_TXC */
        J721E_IOPAD(0xec, PIN_OUTPUT, 4) /* (AG27) PRG0_PRU0_GPO15.RGMII3_TX_CTL */
    >;
};

rgmii4_pins_default: rgmii4-pins-default {
    pinctrl-single,pins = <
        J721E_IOPAD(0x100, PIN_INPUT, 4) /* (AE29) PRG0_PRU1_GPO0.RGMII4_RD0 */
        J721E_IOPAD(0x104, PIN_INPUT, 4) /* (AD28) PRG0_PRU1_GPO1.RGMII4_RD1 */
        J721E_IOPAD(0x108, PIN_INPUT, 4) /* (AD27) PRG0_PRU1_GPO2.RGMII4_RD2 */
        J721E_IOPAD(0x10c, PIN_INPUT, 4) /* (AC25) PRG0_PRU1_GPO3.RGMII4_RD3 */
        J721E_IOPAD(0x118, PIN_INPUT, 4) /* (AC26) PRG0_PRU1_GPO6.RGMII4_RXC */
        J721E_IOPAD(0x110, PIN_INPUT, 4) /* (AD29) PRG0_PRU1_GPO4.RGMII4_RX_CTL */
        J721E_IOPAD(0x12c, PIN_OUTPUT, 4) /* (AG26) PRG0_PRU1_GPO11.RGMII4_TD0 */
        J721E_IOPAD(0x130, PIN_OUTPUT, 4) /* (AF27) PRG0_PRU1_GPO12.RGMII4_TD1 */
        J721E_IOPAD(0x134, PIN_OUTPUT, 4) /* (AF26) PRG0_PRU1_GPO13.RGMII4_TD2 */
        J721E_IOPAD(0x138, PIN_OUTPUT, 4) /* (AE25) PRG0_PRU1_GPO14.RGMII4_TD3 */
        J721E_IOPAD(0x140, PIN_OUTPUT, 4) /* (AG29) PRG0_PRU1_GPO16.RGMII4_TXC */
        J721E_IOPAD(0x13c, PIN_OUTPUT, 4) /* (AF29) PRG0_PRU1_GPO15.RGMII4_TX_CTL */
    >;
};
};
};

```

除了添加 PHY 驱动程序，还需要调整以下 *器件树* 以启用以太网接口：

使用以下命令更改链接的固件 (r5f0_0-fw)：

```
ln -sfm /usr/lib/firmware/ti-ipc/j721e/ipc_echo_test_mcu2_0_release_strip.xer5f /lib/firmware/j7-main-r5f0_0-fw
```

电路板上电后，使用以下终端命令确认 PHY 地址 (phy[x]) 和 eth 端口 (eth[y])：

```
dmesg | grep mdio
```

```

davinci_mdio c000f00.mdio: phy[0]: device c000f00.mdio:00, driver TI DP83TG721CS1.0
davinci_mdio c000f00.mdio: phy[4]: device c000f00.mdio:04, driver TI DP83TG721CS1.0
davinci_mdio c000f00.mdio: phy[5]: device c000f00.mdio:05, driver TI DP83TC818CS2.0
davinci_mdio c000f00.mdio: phy[8]: device c000f00.mdio:08, driver TI DP83TC818CS2.0
am65-cpsw-nuss c000000.ethernet eth1: PHY [c000f00.mdio:00] driver [TI DP83TG721CS1.0] (irq=POLL)
am65-cpsw-nuss c000000.ethernet eth2: PHY [c000f00.mdio:04] driver [TI DP83TG721CS1.0] (irq=POLL)
am65-cpsw-nuss c000000.ethernet eth3: PHY [c000f00.mdio:05] driver [TI DP83TC818CS2.0] (irq=POLL)
am65-cpsw-nuss c000000.ethernet eth4: PHY [c000f00.mdio:08] driver [TI DP83TC818CS2.0] (irq=POLL)

```

4.3 测试设置

图 4-1 展示了测试设置方框图。该设置包括 Jacinto 7 处理器 EVM、TIDA-020071、DP83TC812EVM-MC、DP83TG721EVM-MC、以太网开关、标准 PC 和电缆组件。

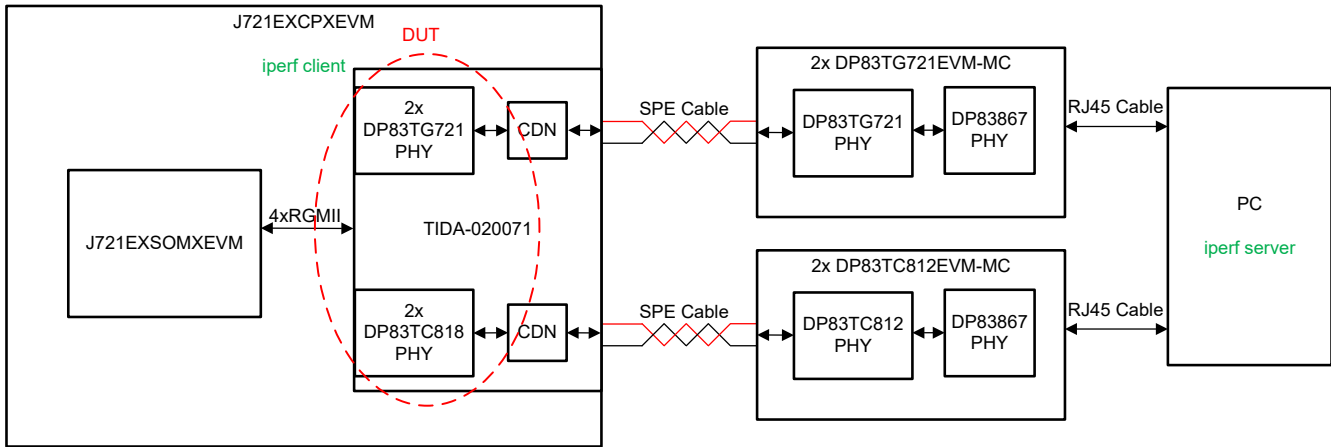


图 4-1. IP 网络带宽测试设置

iPerf3 是一种用于测量网络性能和带宽的开源工具，已集成到 TI 的 SDK 中。将 IP 地址分配给以太网端口后，通过以下命令使用 iPerf3。iPerf3 可用于单独测试每个接口，或者在不同端口上创建多个服务器实例时可以建立多个客户端连接以同时测试多个接口。

在 PC (服务器) 上运行以下命令：

```
iperf3 -s
```

此外，在 Jacinto 7 (客户端) 上运行以下命令：

```
iperf3 -c 192.168.1.1 (the IP address of the server)
```

4.4 测试结果

以下是从 Jacinto 7 EVM 捕获的代码块。在 100Mb/s 的连接下，可以看到成功传输了 114MB (112MB) 的数据，带宽与网络端口广播的速度 (100Mb/s) 非常接近。在 1000Mb/s 设置中，可以看到成功传输了 1.10GB 的数据，带宽与网络端口广播的速度 (1000Mb/s) 非常接近。

以下代码块显示了 100Mb/s 连接：

```
root@j721e-evm:~# iperf3 -c 192.168.1.1
Connecting to host 192.168.1.1, port 5201
[ 5] local 192.168.1.237 port 32836 connected to 192.168.1.1 port 5201
[ ID] Interval      Transfer    Bitrate    Retr  Cwnd
[ 5] 0.00-1.00    sec  12.6 MBytes  106 Mbits/sec  0    249 KBytes
[ 5] 1.00-2.00    sec  10.9 MBytes  91.7 Mbits/sec  0    249 KBytes
[ 5] 2.00-3.00    sec  11.4 MBytes  95.9 Mbits/sec  0    249 KBytes
[ 5] 3.00-4.00    sec  10.9 MBytes  91.7 Mbits/sec  0    249 KBytes
[ 5] 4.00-5.00    sec  11.4 MBytes  95.9 Mbits/sec  0    249 KBytes
[ 5] 5.00-6.00    sec  11.4 MBytes  95.9 Mbits/sec  0    249 KBytes
[ 5] 6.00-7.00    sec  10.9 MBytes  91.7 Mbits/sec  0    249 KBytes
[ 5] 7.00-8.00    sec  11.4 MBytes  95.9 Mbits/sec  0    249 KBytes
[ 5] 8.00-9.00    sec  10.9 MBytes  91.7 Mbits/sec  0    249 KBytes
[ 5] 9.00-10.00   sec  11.4 MBytes  95.9 Mbits/sec  0    249 KBytes
-----
[ ID] Interval      Transfer    Bitrate    Retr
[ 5] 0.00-10.00   sec  114 MBytes  95.2 Mbits/sec  0
[ 5] 0.00-10.05   sec  112 MBytes  93.8 Mbits/sec
iperf Done.
```

以下代码块显示了 1000M/s 连接：

```
root@j721e-evm:~# iperf3 -c 192.168.1.1
Connecting to host 192.168.1.1, port 5201
[ 5] local 192.168.1.81 port 52240 connected to 192.168.1.1 port 5201
[ ID] Interval      Transfer    Bitrate    Retr  Cwnd
[ 5] 0.00-1.00    sec  114 MBytes  957 Mbits/sec  0    362 KBytes
```

```

[ 5] 1.00-2.00 sec 112 MBytes 936 Mbits/sec 10 365 KBytes
[ 5] 2.00-3.00 sec 112 MBytes 944 Mbits/sec 10 369 KBytes
[ 5] 3.00-4.00 sec 112 MBytes 942 Mbits/sec 20 342 KBytes
[ 5] 4.00-5.00 sec 112 MBytes 943 Mbits/sec 10 366 KBytes
[ 5] 5.00-6.00 sec 113 MBytes 945 Mbits/sec 20 315 KBytes
[ 5] 6.00-7.00 sec 112 MBytes 940 Mbits/sec 10 363 KBytes
[ 5] 7.00-8.00 sec 112 MBytes 938 Mbits/sec 20 264 KBytes
[ 5] 8.00-9.00 sec 113 MBytes 947 Mbits/sec 10 346 KBytes
[ 5] 9.00-10.00 sec 112 MBytes 936 Mbits/sec 10 368 KBytes

[ ID] Interval      Transfer      Bitrate      Retr
[ 5] 0.00-10.00 sec 1.10 GBytes 943 Mbits/sec 120
[ 5] 0.00-10.04 sec 1.10 GBytes 938 Mbits/sec
iperf Done.
  
```

5 设计和文档支持

5.1 设计文件

5.1.1 原理图

要下载原理图，请参阅 [TIDA-020071](#) 中的设计文件。

5.1.2 BOM

要下载物料清单 (BOM)，请参阅 [TIDA-020071](#) 中的设计文件。

5.2 工具与软件

工具

ETHERNET-SW	以太网 PHY Linux 驱动程序和工具
J721EXCPXEVMM	用于 Jacinto™ 7 处理器的通用处理器板
J721EXSOMXEVMM	TDA4VM 和 DRA829V 模块上系统 (SoM)

软件

PROCESSOR-SDK-J721E	适用于 DRA829 和 TDA4VM Jacinto™ 处理器的软件开发套件
-------------------------------------	---

5.3 文档支持

- 德州仪器 (TI)，[如何将 Linux 驱动程序集成到您的系统中](#) 应用手册
- 德州仪器 (TI)，[DP83TC818S-Q1 具有 TC10、IEEE802.1AS、IEEE802.1AE MACsec 和 AVB 时钟生成功能的精密安全型 100BASE-T1 汽车以太网数据表](#)
- 德州仪器 (TI)，[DP83TG721x-Q1 具有高级 TSN 和 AVB 的 1000BASE-T1 汽车以太网 PHY 数据表](#)
- 德州仪器 (TI)，[DP83TC811、DP83TC812、DP83TC814、DP83TG720 硬件过渡文档](#) 应用手册
- 德州仪器 (TI)，[TDA4：定制电路板启动指南](#) 应用手册
- 德州仪器 (TI)，[高速接口布局指南](#) 应用手册

5.4 支持资源

[TI E2E™ 中文支持论坛](#) 是工程师的重要参考资料，可直接从专家处获得快速、经过验证的解答和设计帮助。搜索现有解答或提出自己的问题，获得所需的快速设计帮助。

链接的内容由各个贡献者“按原样”提供。这些内容并不构成 TI 技术规范，并且不一定反映 TI 的观点；请参阅 TI 的[使用条款](#)。

5.5 商标

Jacinto™, E2E™, TI E2E™, and FPD-Link™ are trademarks of Texas Instruments.

所有商标均为其各自所有者的财产。

6 作者简介

Yannik MUENDLER 是德州仪器 (TI) 汽车 ADAS 系统工程团队的一名系统工程师。在这一职位上，他在汽车行业参考设计开发方面发挥着关键作用。Yannik 在高性能计算（采用 TI 的 Jacinto™ 7 处理器）以及单线对以太网和 FPD-Link™ 技术等通信接口方面拥有深厚的专业知识积累。他丰富的知识和经验也为其工作提供了强有力的支持。

此外，他拥有德国兰茨胡特应用技术大学的电气工程硕士学位。

重要声明和免责声明

TI“按原样”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功能安全、信息安全、监管或其他要求。

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](#) 或 [ti.com](#) 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。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2024，德州仪器 (TI) 公司