

AT32F407/437使用网口收发数据丢包问题处理

Questions: AT32F407/437 在使用 EMAC 网口进行以太网数据收发时，如果出现数据丢包，应该如何处理？

Answer:**1. 检查 Ethernet PHY 的时钟源**

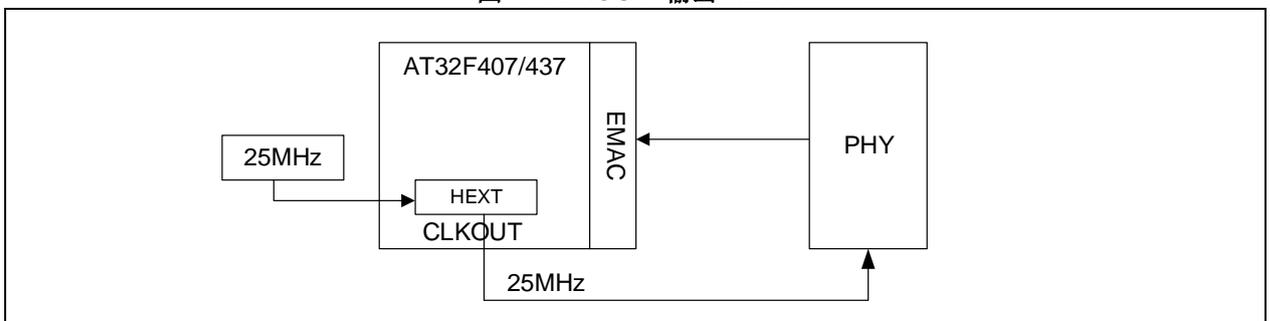
不能使用通过 PLL 分频 25MHz/50MHz 给 Ethernet PHY 供时钟，通过 PLL 的时钟可能达不到 Ethernet PHY 的时钟要求，导致概率出现丢包。如下介绍如何给 Ethernet PHY 供时钟。

AT32F407/437 使用 EMAC 时，根据需求需要给 Ethernet PHY 提供 25MHz/50MHz 时钟

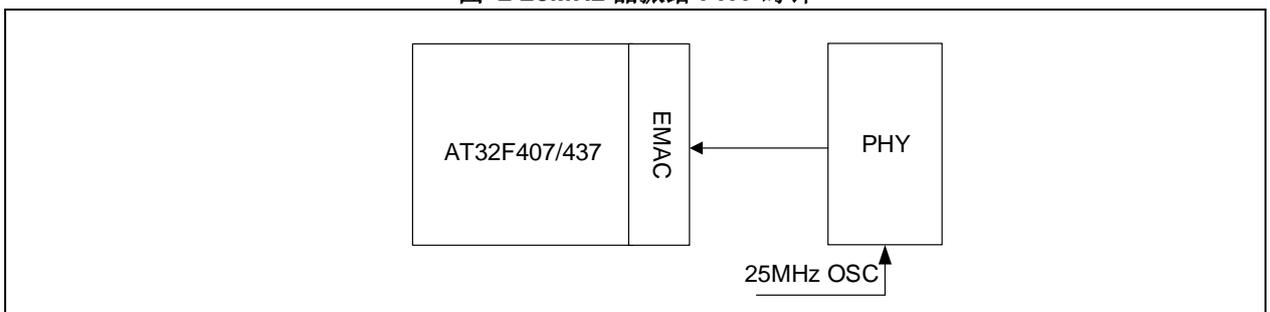
- 当需要给 Ethernet PHY 提供 25MHz 时钟时，有两种方法，如下图 1 和图 2。
- 当需要给 Ethernet PHY 提供 50MHz 时钟时，如下方法图 3，建议直接给 Ethernet PHY 挂 50 晶振

注意：不能使用通过 PLL 分频 25MHz/50MHz 给 Ethernet PHY 供时钟

选择 25MHz 晶振作为 AT32F407 外置时钟 HEXT，CLKOUT(PA8)时钟源选择 HEXT，CLKOUT 输出 25MHz 时钟作为 Ethernet PHY 的时钟源；

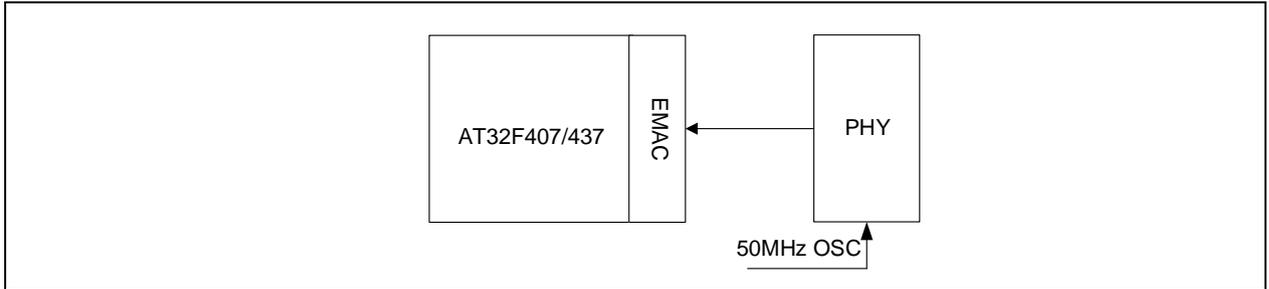
图 1 CLKOUT 输出 HEXT

选择 25MHz 晶振直接作为 Ethernet PHY 时钟。

图 2 25MHz 晶振给 PHY 时钟

选择 50MHz 晶振直接作为 Ethernet PHY 时钟

图 3 50MHz 晶振给 PHY 时钟



2. 检查 EMAC 接收和发送是否存在溢出和错误

A. 对于接收，检查是否有接收溢出：检查 EMAC_DMASTS 如下寄存器位是否置起：

位 4	OVF	0x0	rw1c	接收溢出 (Receive Overflow) 该位表示接收缓存在接收帧期间发生数据溢出。如果部分帧已转发到应用程序，RDES0[11]位会置起溢出状态
-----	-----	-----	------	---

如果发生 OVF 位置起，表示软件的处理速度太慢，不能及时读出缓冲区的数据，导致数据溢出。出现此种情况的处理方法：

- 增加接收缓冲区个数，此处会增加内存使用

```
#define EMAC_RXBUFNB      10 //收缓冲区个数
#define EMAC_TXBUFNB     10 //发送缓冲区个数
```

- 提高软件处理速度

此部分可以使用提高主频（在规格范围内），优化软件加快处理流程。

B. 对于发送，检查是否有发送错误，如下图是否会返回 ERROR

```
error_status emac_txpkt_chainmode(u16 FrameLength)
{
    /* Check if the descriptor is owned by the ETHERNET DMA (when set) or CPU (when reset) */
    if((dma_tx_desc_to_set->status & EMAC_DMATXDESC_OWN) != (u32)RESET)
    {
        /* Return ERROR: OWN bit set */
        return ERROR;
    }
}
```

如果此处返回错误，是因为软件写发送缓冲区的速度太快，EMAC 来不及发送完发送缓冲区的数据，出现此种情况时的处理方法：

- 增加发送缓冲区个数，此处会增加内存使用

```
#define EMAC_RXBUFNB      10 //收缓冲区个数
#define EMAC_TXBUFNB     10 //发送缓冲区个数
```

- 减慢软件写发送缓冲区的速度

如果使用 EMAC 速度是 100Mbps，理论上总线速度可以到 12.5MB/s，建议软件写缓冲区的速度不要超过此速度，因为总线上还有其它协议开销，实际的速度会比 12.5MB/s 要低。

- 发送时增加发送完成判断，以保证有足够缓冲区存放数据

如下图，每次发送完成之后判断下一个缓冲区是否发送完成，此处 timeout 可根据实际发送数据的多少进行修改。

```
error_status emac_txpkt_chainmode(u16 FrameLength)
{
    uint32_t timeout = 0xFFFF;
    /* Check if the descriptor is owned by the ETHERNET DMA (when set) or CPU (when reset) */
    if((dma_tx_desc_to_set->status & EMAC_DMATXDESC_OWN) != (u32)RESET)
    {
        /* Return ERROR: OWN bit set */
        return ERROR;
    }

    /* Setting the Frame Length: bits[12:0] */
    dma_tx_desc_to_set->ctrlsize = (FrameLength & EMAC_DMATXDESC_TBS1);

    /* Setting the last segment and first segment bits (in this case a frame is transmitted in one descriptor) */
    dma_tx_desc_to_set->status |= EMAC_DMATXDESC_LS | EMAC_DMATXDESC_FS;

    /* Set Own bit of the Tx descriptor Status: gives the buffer back to ETHERNET DMA */
    dma_tx_desc_to_set->status |= EMAC_DMATXDESC_OWN;
    /* When Tx Buffer unavailable flag is set: clear it and resume transmission */
    if(emac_dma_flag_get(EMAC_DMA_TBU_FLAG))
    {
        /* Clear TBUS ETHERNET DMA flag */
        emac_dma_flag_clear(EMAC_DMA_TBU_FLAG);
        /* Resume DMA transmission*/
        EMAC_DMA->tpd_bit.tpd = 0;
    }

    /* Update the ETHERNET DMA global Tx descriptor with next Tx decriptor */
    /* Chained Mode */
    /* Selects the next DMA Tx descriptor list for next buffer to send */
    dma_tx_desc_to_set=(emac_dma_desc_type*) (dma_tx_desc_to_set->buf2nextdescaddr);

    while((dma_tx_desc_to_set->status & EMAC_DMATXDESC_OWN) != (u32)RESET)
    {
        if((timeout --) == 0)
        {
            break;
        }
    }
    /* Return SUCCESS */
    return SUCCESS;
}
```

类型：MCU 应用

适用型号：AT32F407/437

主功能：EMAC

次功能：无

文档版本历史

日期	版本	变更
2022.11.16	2.0.0	最初版本

重要通知 - 请仔细阅读

买方自行负责对本文所述雅特力产品和服务的选择和使用，雅特力概不承担与选择或使用本文所述雅特力产品和服务相关的任何责任。

无论之前是否有过任何形式的表示，本文档不以任何方式对任何知识产权进行任何明示或默示的授权或许可。如果本文档任何部分涉及任何第三方产品或服务，不应被视为雅特力授权使用此类第三方产品或服务，或许可其中的任何知识产权，或者被视为涉及以任何方式使用任何此类第三方产品或服务或其中任何知识产权的保证。

除非在雅特力的销售条款中另有说明，否则，雅特力对雅特力产品的使用和/或销售不做任何明示或默示的保证，包括但不限于有关适销性、适合特定用途（及其依据任何司法管辖区的法律的对应情况），或侵犯任何专利、版权或其他知识产权的默示保证。

雅特力产品并非设计或专门用于下列用途的产品：（A）对安全性有特别要求的应用，例如：生命支持、主动植入设备或对产品功能安全有要求的系统；（B）航空应用；（C）航天应用或航天环境；（D）武器，且/或（E）其他可能导致人身伤害、死亡及财产损害的应用。如果采购商擅自将其用于前述应用，即使采购商向雅特力发出了书面通知，风险及法律责任仍将由采购商单独承担，且采购商应独立负责在前述应用中满足所有法律和法规要求。

经销的雅特力产品如有不同于本文档中提出的声明和/或技术特点的规定，将立即导致雅特力针对本文所述雅特力产品或服务授予的任何保证失效，并且不应以任何形式造成或扩大雅特力的任何责任。

© 2022 雅特力科技 保留所有权利