裕太微电子
MotorComm

JULY 22, 2020

YT8614 应用说明

四口 10/100/1000 以太网物理层收发器

版本 0.2

苏州裕太微电子有限公司

苏州市高新区科技城科灵路 78 号 4 号楼 201 室
上海市浦东新区盛夏路 570 号 201 室
www.motor-comm.com

版权声明

Copyright Statement

免责声明

Disclaimer

本文档仅提供阶段性信息，所含内容将/可根据产品的实际情况随时更新，恕不另行通知。如因文档使用不当造成直接或间接损失，裕太微电子不承担任何责任。

This document only provides periodic information, and its contents will/may be updated from time to time according to actual situation of Motorcomm’s products without further notice. Motorcomm will not take any responsibility for any direct or indirect losses caused due to improper use of this document.

Revision History

Revision	Release Date	Summary
0.1	2020.04.17	MPW 版本初始化配置
0.2	2020.07.15	NTO 版本初始化配置；基本功能

目录

1 简介

YT8614 是一款支持一路 QSGMII，四个 10/100/1000 UTP 电口和四个 SGMII/Fiber 口。。

YT8614 需外接 3.3v 和 1.2v 两路电源。

本文主要介绍它的一些功能的配置用法，还包括客户使用过程中经常遇到的问题，包括：硬件上某些特殊的应用如何实现，软件上特定的功能如何配置寄存器。其它的，包括：芯片介绍，请参考 datasheet；电路和 layout 请参考 reference schematic， schematic and layout check list 等。

2 缩略语

YT: 裕太微电子，代指本公司。

PHY: physical layer, 物理层芯片，本文用来代指以太网物理层芯片，即 YT8614 芯片。

MAC: Media Access Control, 数据链路层，本文用来代指与 PHY 相连的上层接口芯片，通常为集成 MAC 功能的 switch 芯片或 CPU 芯片。

DUT: Device under test, 被测对象，本文一般用来代指 YT8614 芯片。

LP: Link Partner，对端相连芯片，本文一般用来代指与 DUT 通过网线，光纤或 SMA 等线相连接的对端芯片。

UTP: un-twisted pair,非屏蔽双绞线，用来与其它 PHY 芯片相连的接口，本文用来代指以太网电口（或称 RJ45），包括 1000/100/10BT 三种速率。

Fiber: 光纤，用来与其它 PHY 芯片相连的接口。本文用来代指以太网光口，包括 1000BX 和 100FX 两种速率。

RGMII: Reduced Giga Media Impendent Interface , 是 PHY 与 MAC 并行接口，在本产品中不存在。

MDIO/SMI: Management Data Input Output, 是由 MDC/MDIO 组成的接口，用来读写 PHY 的内部寄存器。

Mii: Media independent interface，早期以太网 PHY 芯片与 MAC 相连的接口，只适用于百兆/十兆速率，在 YT8614 中已经不存在。但因为 802.3 中规定的 PHY 标准寄存器命名为 MII 寄存器，所以本文中沿用称呼标准寄存器为 Mii 寄存器，记为 mii_reg。

Ext: 由于 MII 寄存器数量有限，仅有 32 个。为满足更多功能需求，phy 芯片配置了更多的寄存器，称为扩展寄存器 extended register,记为 ext_reg。

MMD: MDIO Managable Device, 通过 IEEE802.3 标准 Clause45 规定的 MDIO 接口进行管理的寄存器，称为 MMD 寄存器，记为 mmd_reg。

SGMII: Serial giga media independent interface。一种用来与 MAC 相连的接口。

QSGMII：Quad SGMII。

SDS: Serdes, 串行解串总线的统称，上述的 SGMII 和 Fiber 都属于 SDS 总线。相关的寄存器也称为 SDS 寄存器。

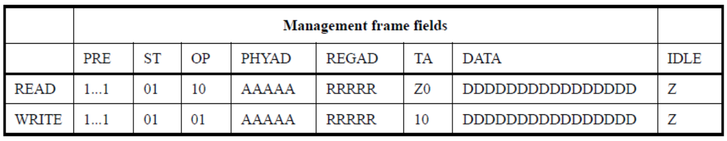
3 寄存器类型及访问方法

3.1 寄存器的类型

YT PHY 内部有三种类型的寄存器：MII 寄存器（MII register，以下简写为 mii_reg），扩展寄存器（extended register，以下简写为 ext_reg）和 MMD 寄存器（简写为 mmd_reg）。

mii_reg：寄存器地址为 0~0x1f，遵从 802.3 clause 22 的标准定义方式进行访问。其 MDC/MDIO 协议如下：

Table 22–12—Management frame format



	Management frame fields							
	PRE	ST	OP	PHYAD	REGAD	TA	DATA	IDLE
READ	1...1	01	10	AAAAA	RRRRR	Z0	DDDDDDDDDDDDDDDD	Z
WRITE	1...1	01	01	AAAAA	RRRRR	10	DDDDDDDDDDDDDDDD	Z

ext_reg：由于 mii_reg 只有 0~0x1f，即 64 个。不能满足产品对寄存器数目的要求，就通过访问两个 mii_reg 0x1e, 0x1f 来扩展了更多的寄存器。访问方式为：

将要访问的 ext_reg 地址，写入 mii_reg 0x1e.

读 mii_reg 0x1f 得到值，即为 ext_reg 内的值. 写值到 mii_reg 0x1f，即将 ext_reg 寄存器内容改为写入的值.

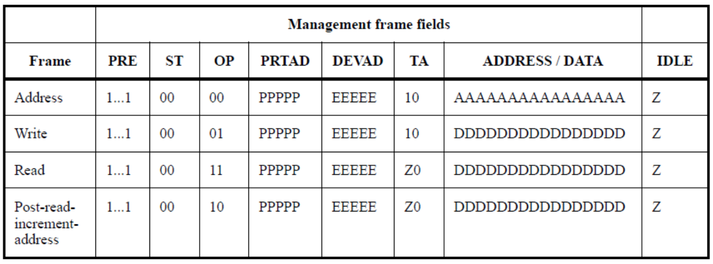
例如：读取 ext_reg0x1000 的值：write mii_reg0x1e 0x1000;
read mii_reg0x1f;

写 ext_reg0x1000 的值为 0x3456：
write mii_reg0x1e 0x1000;
write mii_reg0x1f 0x3456;

mmd_reg：为了扩展更多的寄存器，以适应更高速率的以太网。MMD 寄存器被创造出来。遵从 802.3 clause 45 的标准定义方式进行访问。MMD 寄存器有两种访问方式：间接访问和直接访问。

间接访问：其协议与 MII 访问不同，它是通过两条指令实现 MMD 寄存器的读或写，即先发 address 指令，再发 Write 或 Read 指令实现，具体协议如下：

Table 45–153—Extensions to management frame format for indirect access



	Management frame fields							
Frame	PRE	ST	OP	PRTAD	DEVAD	TA	ADDRESS / DATA	IDLE
Address	1...1	00	00	PPPPP	EEEEE	10	AAAAAAAAAAAAAAAA	Z
Write	1...1	00	01	PPPPP	EEEEE	10	DDDDDDDDDDDDDDDD	Z
Read	1...1	00	11	PPPPP	EEEEE	Z0	DDDDDDDDDDDDDDDD	Z
Post-read-increment-address	1...1	00	10	PPPPP	EEEEE	Z0	DDDDDDDDDDDDDDDD	Z

直接访问：即通过 MII 寄存器（mii_reg 0xd, mii_reg 0xe，其具体含义见下表）来访问 MMD 寄存器。

比如：

读 MMD 3 的 0x5 寄存器，其指令为：

write mii_reg 0xd 0x3; write mii_reg 0xe 0x5; write mii_reg 0xd 0x4003; read mii_reg 0xe

写 MMD 7 的 0x3c 寄存器值为 0x6，其指令为：

write mii_reg 0xd 0x7; write mii_reg 0xe 0x3c; write mii_reg 0xd 0x4007; write mii_reg 0xe 0x6

MII 0Dh: MMD access control register

Bit	Symbol	Access	Default	Description
15:14	Function	RW	2’b0	00 = Address 01 = Data, no post increment 10 = Data, post increment on reads and writes 11 = Data, post increment on writes only
13:5	Reserved	RO	9’b0	Always 0
4	DEVAD	RW	5’b	MMD register device address.

:0			0	00001 = MMD1 00011 = MMD3 00111 = MMD7

MII 0Eh: MMD access data register

Bit	Symbol	Access	Default	Description
15:0	Address data	RW	16’b0	If register 0xD bits [15:14] are 00, this register is used as MMD DEVAD address register. Otherwise, this register is used as MMD DEVAD data register as indicated by its address register.

3.2 通过 MII 寄存器访问其它类型寄存器

按上面所讲，无论 ext_reg 还是 mmd_reg 都可以通过 mii_reg 来访问。

假设函数 read_mii_reg(phy_addr, reg_addr) 与 write_mii_reg(phy_addr, reg_addr, data)表示读写指定 phy 地址的指定 mii 寄存器地址。即：

Mii 寄存器的读写:

函数名：**read_mii_reg**(phy_addr, reg_addr)

输入参数：phy_addr, reg_addr

返回值：读取到的 mii reg_addr 寄存器的值

函数名：**write_mii_reg**(phy_addr, reg_addr, data)

输入参数：phy_addr, reg_addr, data

返回值：无

Ext 寄存器的读写:

函数名：**read_ext_reg**(phy_addr, reg_addr)

输入参数：phy_addr, reg_addr

返回值：读取到的 Ext 寄存器 reg_addr 内的值

```
def read_ext_reg(phy_addr, reg_addr):
write_mii_reg(phy_addr, 0x1e, reg_addr)
d = read_mii_reg(phy_addr, 0x1f)
        return d
```

函数名：**write_ext_reg**(phy_addr, reg_addr, data)

输入参数：phy_addr, reg_addr, data

返回值：无

```
def write_ext_reg(phy_addr, reg_addr, data)
        write_mii_reg(phy_addr, ox1e, reg_addr)
        write_mii_reg(phy_addr, 0x1f, data)
```

MMD 寄存器的读写：

函数名：**read_mmd_reg**(phy_addr, device, reg_addr)

输入参数：phy_addr, device, reg_addr

返回值：读取到的 MMD 寄存器的值

```
def read_mmd_reg(phy_addr, device, reg_addr):
        write_mii_reg(phy_addr, 0x0d, device)
        write_mii_reg(phy_addr, 0x0e,addr)
        write_mii_reg(phy_addr, 0x0d, 0x4000+device)
        d = read_mii_reg(phy_addr, 0x0e)
        return d
```

(说明，例如 MMD3 0x01,其中 3 即为 MMD 寄存器的器件地址；0x01 为 MMD 的寄存器地址)

函数名：**write_mmd_reg**(phy_addr, device, reg_addr, data)

输入参数：phy_addr, device, reg_addr, data

返回值：无

```
def write_mmd_reg(phy_addr, device, reg_addr, data)
        write_mii_reg(phy_addr, 0x0d, device)
        write_mii_reg(phy_addr, 0x0e,reg_addr)
        write_mii_reg(phy_addr, 0x0d, 0x4000+device)
        write_mii_reg(phy_addr, 0x0e, data)
```

3.3 寄存器地址空间

YT8614 按寄存器所属模块区分：1. 通用寄存器(COMMON register)：2. 以太网电口寄存器（UTP register)：3. QSGMII 寄存器（QSGMII register)：4. SGMII/光口寄存器（SGMII_register)：

COMMON 空间寄存器：都属于 ext_reg，且其起始地址为 0xa000。一般地用法是，

Write_ext_reg(phy_addr, 0xa000, 0x2): 选择访问 SDS（包括 SGMII/Fiber）寄存器空间。

Write_ext_reg(phy_addr, 0xa000, 0x0): 选择访问 UTP 寄存器空间。

UTP 空间寄存器：分 utp_mii_reg, utp_ext_reg(起始地址 0x0~0x2xx，不与 common 寄存器地址冲突), mmd_reg。

UTP MII 寄存器: utp_ext_reg0x100 bit1 为 0，这是默认值 ，选择 UTP MII 寄存器，IEEE1000/100/10BT 的速度，自协商能力等，判断 UTP 连接的速度与双工等寄存器，都在 UTP mii_reg 中。

LDS MII 寄存器: utp_ext_reg0x100 bit1 手动设为 1，选择跟 LDS 相关的 MII 寄存器。

UTP MMD 寄存器: utp_mmd_reg，寄存器定义见 802.3 定义和芯片手册。

UTP EXT 寄存器: utp_ext_reg，寄存器定义见 802.3 定义和芯片手册。

QSGMII 空间寄存器：分 qsgmii_mii_reg, qsgmii_ext_reg。主要用来配置 QSGMII 相关的功能。

SGMII 空间寄存器：分 sgmii_mii_reg, ext_reg。主要用来配置 SGMII/Fiber 相关的功能。

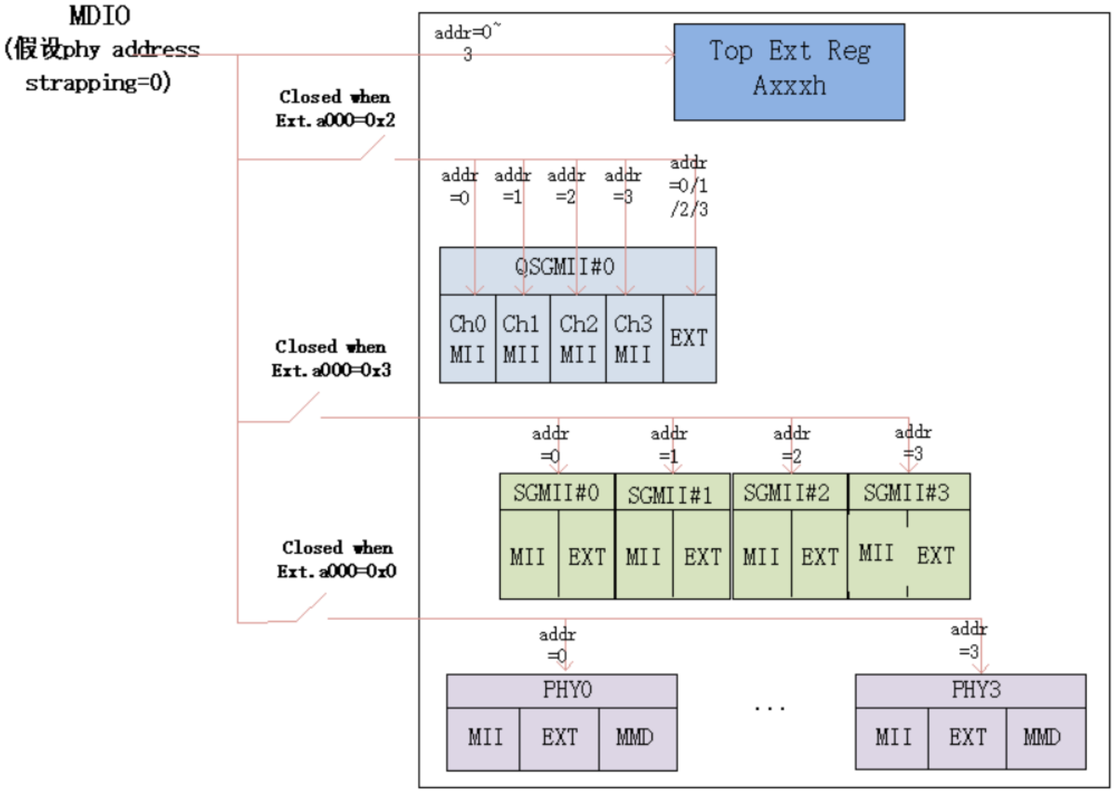
在通过 MDIO 访问寄存器时，各个寄存器对应的 phy 地址（不是寄存器地址），由下式决定：

PhyAddr = PhyBaseAddr + PhyOffsetAddr

其中，PhyBaseAddr 是由芯片的 Pin21 (phyad0)，Pin22(phyad1)，Pin137(phyad2)，Pin136(phyad3)，Pin135(phyad4)的上下拉决定的，芯片默认是 pulldown。

PhyOffsetAddr 见下图各 UTP/QSGMII/SGMII 通道。

比如，通过 power-on-strapping 设定的 PhyBaseAddr 为 0x0，则各模块的 PhyAddr（即下图中 addr）如下：



读取上图中 UTP2 mii_reg0x1:

Write_ext_reg(0x0, 0xa000, 0x0) #写 phy 地址 0x0 的通用扩展寄存器 0xa000 为 0x0,选择 UTP 空间。
Read_mii_reg(0x2, 0x1) #读 phy 地址 0x2 的 mii 寄存器 0x1。

写上图中 QSGMII0 channel1 mii_reg0x4 为 0x1234：

Write_ext_reg(0x0, 0xa000, 0x2) #写 phy 地址 0x0 的通用扩展寄存器 0xa000 为 0x2,选择 QSGMII 空间。
Write_mii_reg(0x1, 0x4, 0x1234) #写 phy 地址 0x1（QSGMII0 channel1）的 mii 寄存器 0x4 的值为 0x1234。

写上图中 SGMII2 ext_reg0xa 为 0xab94：

Write_ext_reg(0x0, 0xa000, 0x3) #写 phy 地址 0x0 的通用扩展寄存器 0xa000 为 0x3,选择 SGMII 空间。
Write_ext_reg(0x2, 0xa, 0xab94) #写 phy 地址 0x2（SGMII2）的 ext 寄存器 0xa 的值为 0xab94。

4 工作模式及配置方法

4.1 内部模块 vs 工作模式

YT8614 支持 6 种工作模式，可由 POS 配置成相应模式。不同的工作模式调用内部不同的模块。它们的对应关系如下：

工作模式 / 内部模块状态	QSGMII				PHY 0	PHY 1	PHY 2	PHY 3	SGMII 0	SGMII 1	SGMII 2	SGMII 3
	ch 0	ch 1	ch 2	ch 3								
QSGMII x 1 + Combo x 4	Y (QSGMII0)				Y	Y	Y	Y	Y	Y	Y	Y
QSGMII x 1 + Copper x 4	Y (QSGMII0)				Y	Y	Y	Y	-	-	-	-
QSGMII x 1 + Fiber x 4	Y (QSGMII0)				-	-	-	-	Y	Y	Y	Y
QSGMII x 1 + SGMAC x 4	Y (QSGMII0)				-	-	-	-	Y	Y	Y	Y
SGMII x 4 + Copper x 4	-				Y	Y	Y	Y	Y	Y	Y	Y
Fiber x 4 + Copper x 4	-				Y	Y	Y	Y	Y	Y	Y	Y

4.2 硬件配置工作模式

工作模式由 POS 决定，详细请参考 datasheet。

5 初始化配置

上电硬复位后，初始化配置 PHY 相关寄存器以优化 10BT 连接下的 QSGMII 性能，如下：

Write_ext_reg(PhyBaseAddr, reg0xa000, 0x2) #切换到 QSGMII 的地址空间

Write_ext_reg(PhyBaseAddr, reg0xe, 0x7c80) #配置 QSGMII0 的扩展寄存器 0xe

Write_ext_reg(PhyBaseAddr, reg0xa000, 0x0) #切换回 UTP 的地址空间

6 软件功能配置

6.1 UTP（电口）配置

6.1.1 UTP 寄存器读写方式

UTP 寄存器的读写，需要先选择 UTP 地址空间，再对相应的 UTP 口（根据口对应的 PhyAddr）的寄存器进行读/写操作，比如：

write_ext_reg(PhyBaseAddr, 0xa000, 0x0) #选择 UTP 地址空间

1. read_utp_mii_reg(PhyBaseAddr+0, 0x1) #读取 UTP port0 的 mii_reg0x1 的值
2. write_utp_mii_reg(PhyBaseAddr+1, 0x0, 0x9140) #写 UTP port1 的 mii_reg0x0 的值为 0x9140
3. read_utp_ext_reg(PhyBaseAddr+4, 0x27) #读取 UTP port4 的 ext_reg0x27 的值
4. write_utp_ext_reg(PhyBaseAddr+4, 0x27, 0x2010) #写 UTP port4 的 ext_reg0x27 的值为 0x2010

注意：

- 对电口寄存器操作一定要确保当前选择的是 UTP 地址空间。
- 每个电口都有独立的一套 mii, ext 和 mmd 寄存器，对电口寄存器操作一定要指定电口的 PhyAddr，即 PhyBaseAddr+PhyOffsetAddr（port0~port3 的 PhyOffsetAddr 分别为 0，1，2，3）。

6.1.2 PHY ID

上电硬复位后，可以读取 utp mii_reg0x2,0x3 获取 PHY ID。YT8614 PHY ID 为 0x4f51e899。

Phy MII 02h: PHY identification register1				
Bit	Symbol	Access	Default	Description
15:0	Phy_Id	RO	0x4F51	Bits 3 to 18 of the Organizationally Unique Identifier
Phy MII 03h: PHY identification register2				
Bit	Symbol	Access	Default	Description
15:10	Phy_Id	RO	0xE8	Bits 19 to 24 of the Organizationally Unique Identifier
9:4	Type_No	RO	0x9	6 bits manufacturer's type number. It's 0x8 in YT8618, and 0x9 in YT8614.
3:0	Revision_No	RO	0x9	4 bits manufacturer's revision number.

6.1.3 10/100/1000BT 的速度和双工

默认 YT8614 不需做任何配置，YT8614 的自协商能力是打开的，1000/100/10BT 能力都是打开的。

判断 PHY 当前的连接速度和双工？有两种方法：

通过标准的 PHY MII 寄存器来判断当前是否 link up（utp_mii_reg0x1[2]），然后通过 phy_utp_mii_reg0x4, 0x5, 0x9 等标准寄存器和算法来得到当前 PHY 的连接速度和双工。

为了更方便地得到当前 PHY 的连接与否，速度与双工，YT PHY 将这些信息放在 phy_utp_mii_reg0x11(只读寄存器)内，系统只需读取 0x11 寄存器，对应地配置 MAC，即可正常通信。

其中的几个 bit 含义如下（或参照芯片手册）：

Bit 15-Bit14 ： speed mode， 11---系统保留；10---1000M；01---100M；00---10M

Bit 13 : Duplex， 1---Full duplex；0---half duplex

Bit 10 : Link status, 1---link up; 0---link down

注：要先判断 bit 10 为 1，表示连接建立，再去判断 bit 15,14,13 得到速度和双工。

Phy MII 11h: PHY specific status register 0x11				
Bit	Symbol	Access	Default	Description
15:14	Speed_mode	RO	0x0	These status bits are valid only when bit11 is 1. Bit11 is set when Auto-Negotiation is completed or Auto-Negotiation is disabled. 11 = Reserved 10 = 1000 Mbps 01 = 100 Mbps 00 = 10 Mbps
13	Duplex	RO	0x0	This status bit is valid only when bit11 is 1. Bit11 is set when Auto-Negotiation is completed or Auto-Negotiation is disabled. 1 = Full-duplex 0 = Half-duplex

12	Page Received real-time	RO	0x0	1 = Page received 0 = Page not received
11	Speed and Duplex Resolved	RO	0x0	When Auto-Negotiation is disabled, this bit is set to 1 for force speed mode. 1 = Resolved 0 = Not resolved
10	Link status real-time	RO	0x0	1 = Link up 0 = Link down

6.1.4 只保留 1000BT 能力

YT8614 电口不支持 Force 配置千兆，可通过配置自协商能力，达到关闭百兆、十兆，只保留千兆能力。相关寄存器是 UTP MII 的 0x04、0x09 寄存器。

上电复位后，进行如下寄存器配置初始化动作：

write utp_mii_reg0x04: 0x1C01 #将 Bit8、7、6、5 位的值由默认的 1 改为 0，分别关闭了 100M_Full、100M_Half、10M_Full、10M_Half 的自协商能力。其余位的值不做处理。若其他位的值在其他功能中有改变，请按改变后的赋值。

Set bit mii_reg0x00[15] #置位 mii 寄存器 0x00 的 Bit15，进行软复位，使得上述寄存器设置生效。软复位后，YT8614 会重新启动 link up 过程。(软复位不会清寄存器值，只会使其生效)

Phy MII 04h: Auto-Negotiation advertisement				
Bit	Symbol	Access	Default	Description
15	NEXT_Page	RW	0x0	If 1000BASE-T is advertised, the required next pages are automatically transmitted. This bit must be set to 0 if no additional next page is needed. 1 = Advertise 0 = Not advertised
14	Ack	RO	0x0	Always 0.
13	Remote_Fault	RW	0x0	1 = Set Remote Fault bit 0 = Do not set Remote Fault bit
12	Extended_NEXT_Page	RW	0x1	1 = Local device supports transmission of extended next pages 0 = Local device does not support transmission of extended next pages.
11	Asymmetric_Pause	RW	0x1	1 = Asymmetric Pause 0 = No asymmetric Pause
10	Pause	RW	0x1	1 = MAC PAUSE implemented 0 = MAC PAUSE not implemented
9	100BASE-T4	RO	0x0	1 = Able to perform 100BASE-T4 0 = Not able to perform 100BASE-T4 Always 0
8	100BASE-TX_Full_Duplex	RW	0x1	1 = Advertise 0 = Not advertised Needs Soft Reset action to take effect
7	100BASE-TX_Half_Duplex	RW	0x1	1 = Advertise 0 = Not advertised Needs Soft Reset action to take effect

6	10BASE-Te_Full_Duplex	RW	0x1	1 = Advertise 0 = Not advertised Needs SoftReset action to take effect
5	10BASE-Te_Half_Duplex	RW	0x1	1 = Advertise 0 = Not advertised Needs SoftReset action to take effect
4:0	Selector_Field	RW	0x1	Selector Field mode. 00001 = IEEE 802.3

Phy MII 09h: MASTER-SLAVE control register

Bit	Symbol	Access	Default	Description
15:13	Test mode	RW	0x0	The TX_TCLK signals from the RX_CLK pin is for jitter testing in test modes 2 and 3. Needs soft reset to take effect 000 = Normal Mode 001 = Test Mode 1 - Transmit Waveform Test 010 = Test Mode 2 - Transmit Jitter Test (MASTER mode) 011 = Test Mode 3 - Transmit Jitter Test (SLAVE mode) 100 = Test Mode 4 - Transmit Distortion Test 110, 111 = Reserved
12	Master/Slave Manual configuration Enable	RW	0x0	1 = Manual MASTER/SLAVE configuration 0 = Automatic MASTER/SLAVE configuration. Needs soft reset to take effect
11	Master/Slave configuration	RW	0x0	This bit is ignored if bit[12] is 0. 1 = Manual configuration as MASTER 0 = Manual configuration as SLAVE. Needs soft reset to take effect
10	Port Type	RW	0x0	This bit is ignored if bit[12] is 1. 1 = Prefer multi-port device (MASTER) 0 = Prefer single port device (SLAVE) Needs soft reset to take effect
9	1000BASE-T Full	RW	0x1	1 = Advertise 0 = Not advertised Needs soft reset to take effect
8	1000BASE-T Half-	RW	0x0	1 = Advertise 0 = Not advertised (default) Needs soft reset to take effect
7:0	Reserved	RW	0x0	Write as 0, ignore on read.

6.1.5 Loopback（回环）模式

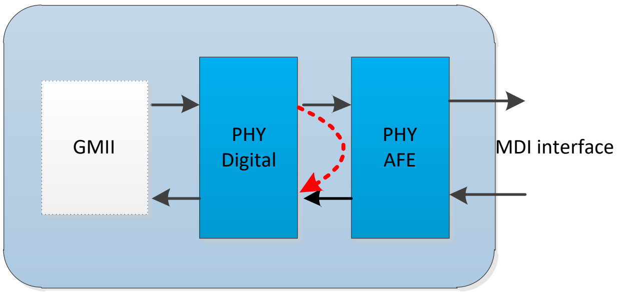
1、Internal loopback 情况下，utp 的 analog 电路被 bypass，直接将芯片内部 GMII 接口发送的数据从 utp phy 的数字电路回环回去，也可以称作 digital loopback，如下图所示。以 port 0 为例，配置方法如下：

10M：write_utp_mii_reg(PhyBaseAddr+0, 0x0,0x4100)

100M: write_utp_mii_reg(PhyBaseAddr+0, 0x0,0x6100)

1000M: write_utp_mii_reg(PhyBaseAddr+0, 0x0,0x4140)

注意：一般的寄存器配置不会被软复位清掉，但 UTP mii register 0x0 里的 internal loopback 和 power down 2 个 bit 会被软复位清掉，因此在使能这两个功能后不要做软复位。



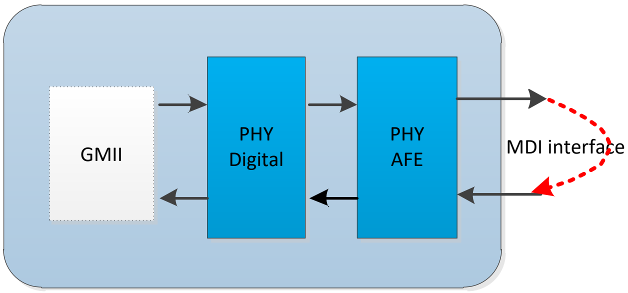
2、在 External Loopback 模式下，AFE 发送的信号由回环线直接回到 AFE 的接收电路。回环线将网线的 pair1&2 与 pair3&6 连接，pair4&5 与 pair7&8 连接，如下图所示。以 port 0 为例，配置方法如下：

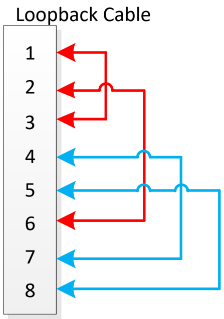
10M：write_utp_ext_reg(PhyBaseAddr+0, 0x27[15],1’b0)关闭 sleep 模式；
write_utp_ext_reg(PhyBaseAddr+0, 0xa[4],1’b1);
write_utp_mii_reg(PhyBaseAddr+0, 0x0,0x8100)

100M: write_utp_ext_reg(PhyBaseAddr+0, 0x27[15],1’b0)关闭 sleep 模式;
write_utp_ext_reg(PhyBaseAddr+0, 0xa[4],1’b1),
write_utp_mii_reg(PhyBaseAddr+0, 0x0,0xa100)

1000M: write_utp_ext_reg(PhyBaseAddr+0, 0x27[15],1’b0)关闭 sleep 模式;
write_utp_ext_reg(PhyBaseAddr+0, 0xa[4],1’b1)
write_utp_mii_reg(PhyBaseAddr+0, 0x0,0x8140)

注意：上述配置的最后一个 mii register x0，因为 bit15 置为 1，实现了软复位，使得前面的配置生效并发起 external loopback 机制，此时应保证 external loopback 线已经插上，否则会导致 external loopback 无法连接，这时需要再一次软复位才可以。

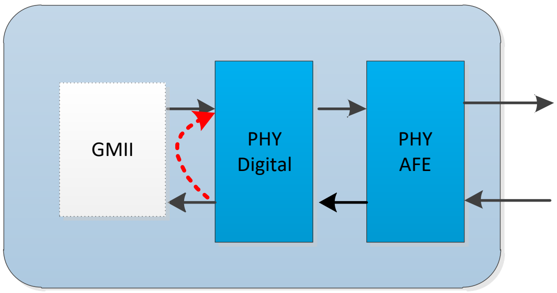




3、在 Remote Loopback 模式下，芯片内部 UTP GMII RX 接收的数据被直接注入 GMII TX。如下图所示。以 port 0 为例，以 port 0 为例，配置方法如下：

10M:　write_utp_ext_reg(PhyBaseAddr+0, 0xa[5], 1’b1)

```
          write_utp_mii_reg(PhyBaseAddr+0, 0x9, 0x0)
          write_utp_mii_reg(PhyBaseAddr+0, 0x4, 0xc41)
           write_utp_mii_reg(PhyBaseAddr+0, 0x0, 0x9140)
100M:     write_utp_ext_reg(PhyBaseAddr+0, 0xa[5], 1’b1)
           write_utp_mii_reg(PhyBaseAddr+0, 0x9, 0x0)
           write_utp_mii_reg(PhyBaseAddr+0, 0x4, 0xd01)
           write_utp_mii_reg(PhyBaseAddr+0, 0x0, 0x9140),
1000M: write_utp_ext_reg(PhyBaseAddr+0, 0xa[5], 1’b1)
            write_utp_mii_reg(PhyBaseAddr+0, 0x9, 0x200)
            write_utp_mii_reg(PhyBaseAddr+0, 0x4, 0xc01)
            write_utp_mii_reg(PhyBaseAddr+0, 0x0, 0x9140)
```



6.1.6 Smart downgrade（自动降速）配置

对于以太网 PHY，当网线插上时，默认情况下是从双方支持的最高速率开始尝试连接。有时由于外界干扰或连接线太长或质量太差，导致高速率连接不上。当反复尝试连接超过一定次数时，PHY 可以选择自动降速来完成连接。Smart speed，或者叫 auto-downgrade，就是指 PHY 的这个自动降低连接速度的功能。

比如双方都按 1000BT 的能力发起自协商，但网线超过最大连接距离（比如到 150 米），双方经过 5 次尝试失败（每次 link up 或 link down 维持时间小于 2 秒），仍无法连接到 1000BT，此时具备自动降速功能的一方会发起 100BT 的自协商，尝试 100BT 的连接。当然若 100BT 经过 5 次尝试，仍不能连接，PHY 会降到 10BT 再尝试。

注：

1. 如果对端口的快速插拔，导致短时间内的不断 link up/down，也会触发 smart downgrade 机制。
2. 此功能只发生在连接建立时间内，若连接并维持住，PHY 不会根据丢包等自动降速。

以上是 YT8614 的默认行为。如果不需要此功能，则需要将此功能通过寄存器禁用。即，将寄存器 phy mii_reg0x14 bit5 清 0，然后做 software reset (将 phy mii_reg0x0 bit15 置 1)使其生效。描述如下：

Phy MII 14h: Speed Auto Downgrade Control Register				
Bit	Symbol	Access	Default	Description
15:12	Reserved	RO	0x0	Reserved
11	Reserved	RW	0x1	Reserved
10	Reserved	RW	0x0	Reserved
9	Reserved	RW	0x0	Reserved
8	Reserved	RW	0x0	Reserved
7:6	Reserved	RO	0x0	Reserved

5	En_speed_downgrade	RW	0x1	When this bit is set to 1, the PHY enables smart-speed function. Writing this bit requires a software reset to update. This bit will be set to 1'b0 in UTP_TO_FIBER_FORCE and UTP_TO_FIBER_AUTO mode; else set to 1'b1, only take effect after software reset
4:2	Reserved	RW	0x3	Reserved
1	Reserved	RW	0x0	Reserved
0	Reserved	RO	0x0	Reserved

注：若要禁用 smart speed，需要网线两端的 PHY 都禁用才可以。否则任一边自动降速都会导致连接降速。

6.1.7 Sleep（自动睡眠）配置

UTP 处于不连接状态超过一定时间后（约 40 秒），PHY 会自动进入 sleep 状态。在 sleep 状态下，PHY 会自动关闭内部部分电路，以达到省电的功能。进入 sleep 后，PHY 会定期发送 Pulse 信号，并打开信号侦测的功能。一旦收到对方发过来的信号，超过信号侦测的门限，PHY 会立刻打开相关电路进入正常工作状态。

Sleep 功能默认是打开的，此功能不会影响 PHY 的正常连接。如果关闭 sleep 功能也不会有任何副作用。

关闭此功能的寄存器： write_utp_ext_reg0x27[15]: 1’b0。

Phy EXT 27h: Sleep Control1

Bit	Symbol	Access	default	Description
15	En_sleep_sw	RW	0x1	1 = enable sleep mode: PHY will enter sleep mode and close AFE after unplug cable for a timer;
14	Pllon_in_slp	RO	0x0	1 = keep PLL on in sleep mode;
13	Slp_pulse_sw	RW	0x1	when PHY enter sleep,
12	En_upd_afe_sbs	RW	0x0	When AFE control is changed, no matter it's triggered by sleep control logic or normal work mode change,
11:6	Reserved	RO	0x0	Reserved
5	Sleeping	RO	0x0	1 = PHY is slept;
4	Gate_25m	RO	0x0	Not used.
3:0	Slp_state	RO	0x0	FSM state of internal sleep control logic.

6.1.8 LDS 和 4 对线百兆长距离

YT8614 支持在 4 对 UTP 线上实现长距离（>400 米 cat5e）百兆以太网的信号传输。当然，前提是对方也必须支持这种模式（通常也是 YT 系列型号，目前包括 YT8511, YT8521S）。除去上面提到的区别外，与传统百兆以太网（100BT）相比，4 对线长距离以太网（暂命名为 4pair_LRE_100M）还有以下特点：

通过 LDS（Link Discovery Signaling）协商的机制实现双方支持能力的交互（10/100/100BT，4pair_LRE_100M）。

在 LDS 协商过程中检测到连接的线长，根据双方能力和线长，决定连接到哪种速率，在 LDS enable，并且所有能力都打开的情况下，最终速率与线长的关系为：

线长	速率
< 200m	10/100/1000BT 自协商
>= 200m	4pair_LRE_100M

注：

1. 整个过程全部由 PHY 自动实现，MAC 不参与。MAC 通过 polling PHY 寄存器得到当前的状态。
2. 默认情况下，PHY 的所有能力（1000BT, 100BT, 10BT, 4pair_LRE_100M）都打开，但 LDS 不开，所以跟传统的 GE 一样，不会进 4pair_LRE_100M 模式。

3. 在 LDS 打开的情况下，若对端是传统 10/100/1000 以太网，DUT 也能够与其对连到传统以太网模式。
4. 在长线条件下，线长检测精度通常<5%。

综上所述，在 YT8614 与 YT8614 相连情况下，若要进入 4pair_LRE_100M 模式，只要：

1. 连线超过 200m（极限线长约 500m）
2. 选择 DUT LDS MII 地址空间
3. 双方 LDS enable，并 software reset
4. 通过 LDS 或 PHY MII_reg0x1 (Latch low)/PHY MII_reg0x11[10]，判断 DUT 已连接。
5. 读 DUT LDS MII_reg0xb，判断当前连到 4pair_LRE_100M 还是 10/100/1000BT:
6. 若 4pair_LRE_100M: 则配置 MAC 到百兆
7. 若 10/100/1000BT: 则选择 PHY MII 地址空间，再通过 PHY MII_reg0x11 判断速度和双工，配置 MAC
8. 正常通信

相关寄存器配置：

LDS 的相关寄存器放在 LDS MII 寄存器中，与 PHY MII 寄存器属于不同的地址空间，其通过 PHY ext reg 0x100 来选择，其含义为：

Phy EXT 100h: PHY/LDS sel

Bit	Symbol	Access	Default	Description
15:3	Reserved	RO	0x0	reserved
2	Acc_ctrl_ovrd_en	RW	0x1	Access control override enble; 1'b1: override LRE register access; 1'b0: Normal operation
1	Acc_ctrl_ovrd_val	RW	0x1	1'b1: Access IEEE registers; 1'b0: Access LRE registers
0	Acc_ctrl_val	RO	0x1	1'b1: IEEE register is active; 1'b0: LRE register is active

LDS 打开/关闭的寄存器：

Lds mii 00h: LRE control

Bit	Symbol	Access	Default	Description
15	Reset	RW SC	0x0	PHY Software Reset. Writing 1 to this bit causes immediate PHY reset. Once the operation is done, this bit is cleared automatically.; 1'b0: Normal operation; 1'b1: PHY reset
14	Loopback	RW	0x0	Loopback control; 1'b0: disable loopback; 1'b1: enable loopback
13	Restart_LDS	RW SC	0x0	1'b1: restart LDS process
12	LDS_Enable	RW	0x0	1'b1: LDS enabled; 1'b0: LDS disabled
11	Power_down	RW	0x0	1 = Power down; 0 = Normal operation; When the port is switched from power down to; normal operation, software reset and Auto-; Negotiation are performed even bit[15] RESET and bit[9] RESTART_AUTO_NEGOTIATION are not set by the user.
10	Isolate	RW	0x0	Isolate phy from MII/GMII/RGMII: PHY will not respond to RGMII TXD/TX_CTL, and present high impedance on RXD/RX_CTL.; 1'b0: Normal mode; 1'b1: Isolate mode
9:6	Speed_selection	RW	0x0	4'b0000: 10Mbps; 4'b1000: 100Mbps; Others: reserved
5:4	Pair_selection	RW	0x0	2'b00: 1 pair connection; 2'b01: 2 pair connections; 2'b10: 4 pair connections; 2'b11: reserved
3	M/S_selection	RW	0x0	1'b1: manually force local device to master, when reg0.12 = 0; 1'b0: manually force local device to slave, when reg0.12 = 0
2	Force auto negotiation	RW	0x0	1'b1: manually force local device to auto negotiation state, when reg0.12 = 0
1:0	Reserved	RO	0x0	Reserved. Write as 0, ignore on read

线长检测的长度：

Lds mii 0Ah: LDS expansion				
Bit	Symbol	Access	Default	Description
15	Downgrade_ability	RO	0x0	1'b1: LDS speed downgrade
14	Master/Slave	RO	0x0	1 = Local PHY configuration resolved to Master; 0 = Local PHY configuration resolved to Slave
13:12	Connections_pairs	RO	0x0	Number of pairs; 2'b00: 1 pair; 2'b01: 2 pairs; 2'b10: 4 pairs; 2'b11: reserved
11:0	Estimated_cable_length	RO	0x0	

判断 DUT 当前是否已连接：

Lds mii 01h: LRE status				
Bit	Symbol	Access	Default	Description
15:14	Reserved	RO	0x0	Ignore on read
13	100Mbps_1-pair capable	RO	0x0	1'b1: 100Mbps 1-pair capable; 1'b0: Not 100Mbps 1-pair capable
12	100Mbps_4-pair capable	RO	0x1	1'b1: 100Mbps 4-pair capable; 1'b0: Not 100Mbps 4-pair capable
11	100Mbps_2-pair capable	RO	0x0	1'b1: 100Mbps 2-pair capable; 1'b0: Not 100Mbps 2-pair capable
10	10Mbps_2-pair capable	RO	0x0	1'b1: 10Mbps 2-pair capable; 1'b0: Not 10Mbps 2-pair capable
9	10Mbps_1-pair capable	RO	0x0	1'b1: 10Mbps 1-pair capable; 1'b0: Not 10Mbps 1-pair capable
8	Extended_Status	RO	0x1	Whether support EXTended status register in 0Fh; 0: Not supported; 1: Supported
7	Reserved	RO	0x1	
6	Mf_Preamble_Suppression	RO	0x1	1'b0: PHY will not accept management frames with preamble suppressed; 1'b1: PHY will accept management frames with preamble suppressed
5	LDS_Complete	RO	0x0	1'b1: LDS auto-negotiation complete; 1'b0: LDS auto-negotiation not complete
4	Support_IEEE_802.3 _PHY	RO	0x1	1'b1: Support IEEE 802.3 PHY operation; 1'b0: Not Support IEEE 802.3 PHY operation
3	LDS_Ability	RO	0x1	1'b1: LDS auto-negotiation capable; 1'b0: Not LDS auto-negotiation capable
2	Link_Status	RO	0x0	Link status; 1'b0: Link is down; 1'b1: Link is up
1	Jabber_Detect	RO LH	0x0	10Baset jabber detected; 1'b0: no jabber condition detected; 1'b1: Jabber condition detected
0	Extended_Capability	RO LH	0x1	To indicate whether support EXTended registers, to access from address register 1Eh and data register 1Fh; 1'b0: Not supported; 1'b1: Supported

判断当前连接是 4pair_LRE_100M 还是 10/100/1000BT：

Lds mii 0Bh: LDS Results				
Bit	Symbol	Access	Default	Description
15:6	Reserved	RO	0x0	
5	4-pair_100M	RO	0x0	1'b1: local PHY configuration resolved to 4-pair 100M

4	Auto_negotiation	RO	0x0	1'b1: local PHY configuration resolved to AN
3	1-pair_100M	RO	0x0	1'b1: local PHY configuration resolved to 1-pair 100M
2	1-pair_10M	RO	0x0	1'b1: local PHY configuration resolved to 1-pair 10M
1	2-pair_100M	RO	0x0	1'b1: local PHY configuration resolved to BR 100M
0	2-pair_10M	RO	0x0	1'b1: local PHY configuration resolved to BR 10M

判断当前连接是 10BT/100BT/1000BT：

Phy MII 11h: PHY specific status register

Bit	Symbol	Access	Default	Description
15:14	Speed_mode	RO	0x0	These status bits are valid only when bit11 is 1. Bit11 is set when Auto-Negotiation is completed or Auto-Negotiation is disabled. 11 = Reserved 10 = 1000 Mbps 01 = 100 Mbps 00 = 10 Mbps
13	Duplex	RO	0x0	This status bit is valid only when bit11 is 1. Bit11 is set when Auto-Negotiation is completed or Auto-Negotiation is disabled. 1 = Full-duplex 0 = Half-duplex
12	Page Received real-time	RO	0x0	1 = Page received 0 = Page not received
11	Speed and Duplex Resolved	RO	0x0	When Auto-Negotiation is disabled, this bit is set to 1 for force speed mode. 1 = Resolved 0 = Not resolved
10	Link status real-time	RO	0x0	1 = Link up 0 = Link down
9:7	Reserved	RO	0x0	Reserved
6	MDI Crossover Status	RO	0x0	This status bit is valid only when bit11 is 1. Bit11 is set when Auto-Negotiation is completed or Auto-Negotiation is disabled. The bit value depends on register 0x10 “PHY specific function control register” bits6~bit5 configurations. Register 0x10 configurations take effect after software reset. 1 = MDIX 0 = MDI
5	Wirespeed downgrade	RO	0x0	1 = Downgrade 0 = No Downgrade
4	Reserved	RO	0x0	Reserved
3	Transmit Pause	RO	0x0	This status bit is valid only when bit11 is 1. Bit11 is set when Auto-Negotiation is completed. This bit indicates MAC pause resolution. This bit is for information purposes only and is not used by the device. When in force mode, this bit is set to be 0. 1 = Transmit pause enabled 0 = Transmit pause disabled

2	Receive Pause	RO	0x0	This status bit is valid only when bit[11] is 1. Bit[11] is set when Auto-Negotiation is completed. This bit indicates MAC pause resolution. This bit is for information purposes only and is not used by the device. When in force mode, this bit is set to be 0. 1 = Receive pause enabled 0 = Receive pause disabled
1	Polarity Real Time	RO	0x0	1 = Reverted polarity 0 = Normal polarity
0	Jabber Real Time	RO	0x0	1 = Jabber 0 = No jabber

详细的寄存器操作为：

1. Write ext_reg0xa000: 0x0 #选择 PHY&LDS 地址空间
2. Write utp_ext_reg0x100: 0x4 #选择 LDS 地址空间
3. Write utp_mii_reg0x0: bit12 置 1 #打开 LDS 能力
4. Write utp_mii_reg0x0: bit15 置 1 #软复位
5. 等待一段时间或 polling mii_reg0x1[2]变为 1 #表示已连接
6. Read utp_mii_reg0xb: bit5 若为 1：则为 4pair_LRE_100M。将 MAC 配置为百兆全双工，结束。
   bit5 若为 0，且 bit4 为 1，则连到 10/100/1000BT，然后：
7. write utp_ext_reg0x100: 0x6 #选择 PHY 地址空间
8. read utp_mii_reg0x11: bit 15, 14, 13 #判断速度双工
9. 配置 MAC 到相同速率，结束

6.1.9 VCT（线长检测）配置

YT8614 进入含 copper 的一种模式，比如：Qsgmii x1 + Copper x4，然后进行 VCT 测试；

1，把 common ext reg0xa000 写 0x0； #访问 phy 的地址空间

2，把 phy utp ext reg0x27 的 bit15 清零； #关闭 sleep 模式

3，把 phy utp ext reg0x80[0]置'1'； #进入 VCT 测试

4，当 phy utp ext reg0x84 的 bit[15]变为 0，vct 测试完成；

5，读取 phy utp ext reg0x84[7:0]，判断 pair 内的 open（0xff），short（0xaa），load（0x0）状态；

建议先校准一下环境，把待测网线的另一端接 load（pair0~3 的每个 pair 的 p 和 n 之间接 100ohm 的电阻），得到的 phy ext reg0x84[7:0]为 0x0，符合预期后再进行后续测试。

UTP EXT 84h: vct_mon0				
7:6	Self_st_3	RO	2'b0	Intra pair status of channel 3. 00: normal, pair impedance matches. 01: error happened during last VCT test, the error may be RX channel is busy, no near-end echo was detected, or the location of far-end echo was not farther than that of near-end echo and the gap between near-end echo and far-end echo is not small; 10: pair is short; 11: pair is open.
5:4	Self_st_2	RO	2'b0	Intra pair status of channel 2.See bit7~6 for detail.
3:2	Self_st_1	RO	2'b0	Intra pair status of channel 1.See bit7~6 for detail.
1:0	Self_st_0	RO	2'b0	Intra pair status of channel 0.See bit7~6 for detail.

6，读取 phy ext reg0x87~8a 得到网线出问题的点，单位为 cm(load 状态得到的值为 0，此状态主要做校准环境用)；

UTP EXT 87h: vct_mon3				
Bit	Symbol	Access	default	Description
15:00	Self_dmg_loc_0	RO	0x0	The intra pair damage location of channel 0. In unit cm.
UTP EXT 88h: vct_mon4				
Bit	Symbol	Access	default	Description
15:00	Self_dmg_loc_1	RO	0x0	The intra pair damage location of channel 1. In unit cm.
UTP EXT 89h: vct_mon5				
Bit	Symbol	Access	default	Description
15:00	Self_dmg_loc_2	RO	0x0	The intra pair damage location of channel 2. In unit cm.
UTP EXT 8Ah: vct_mon6				
Bit	Symbol	Access	default	Description
15:00	Self_dmg_loc_3	RO	0x0	The intra pair damage location of channel 3. In unit cm.

6.1.10 Template（电口指标）配置

YT8614 template 测试命令如下：
初始化：
进入一个含 copper 的模式，比如：Qsgmii x1 + Copper x4，然后对每个 port 分别进行 template 测试

1000BT:

```
    Write_comm_ext_reg0xa000: 0x0        #选择 UTP 地址空间
    write_utp_ext_reg0x27: 0x2026        #disable sleep
    write_utp_mii_reg0x10: 0x2           #force MDI
    write_utp_mii_ reg0x0: 0x8140        # #phy reset,auto-nego disable

    Test Mode 1，Transmit waveform test
    write_utp_mii_reg0x9: 0x2200
    write_utp_mii_reg0x0: 0x8140
     Test Mode 2，Transmit Jitter test (master mode)
    write_utp_mii_reg0x9: 0x5a00
    write_utp_mii_reg0x0: 0x8140
     Test Mode 3，Transmit Jitter test (slave mode)
    write_utp_mii_reg0x9: 0x7200
    write_utp_mii_ reg0x0: 0x8140
     Test Mode 4，Transmit distortion test
    write_utp_mii_reg0x9: 0x8200
    write_utp_mii_ reg0x0: 0x8140
```

100BT：

```
    Write_comm_ext_reg0xa000: 0x0            #选择 UTP 地址空间
    Write utp ext reg0x41:0x33               #rise/fall adjust
    Write utp ext reg0x42:0x66
    Write utp ext reg0x43:0xaa
```

```
Write utp ext reg0x44:0xd0d
Write_utp_ext_reg0x27: 0x2026          #disable sleep
Write_utp_mii_reg_0x10: 0x2            #force MDI
Write_utp_mii_reg_0x0: 0xa100          # PHY reset,auto-nego disable,100M
```

10BTe

```
Write_comm_ext_reg0xa000: 0x0 #选择 UTP 地址空间
Write_utp_ext_reg0x27: 0x2026          #disable sleep
Write_utp_mii_reg_0x10: 0x2            #force MDI
Write_utp_mii_reg0x0: 0x8100           # phy reset,auto-nego disable,10M

Write_utp_ext_reg0xa: 0xa09     #packet with all ones, 10MHz sine wave, for harmonic test
Write_utp_ext_reg0xa: 0xa0a     #pseudo random, for TP_idle/Jitter/Different voltage test
Write_utp_ext_reg0xa: 0xa0b     #normal link pulse only
Write_utp_ext_reg0xa: 0xa0c     #5MHz sine wave
Write_utp_ext_reg0xa: 0xa0d     #Normal mode
```

上述设置能通过 template 测试。考虑到不同的板级设计，PHY 内部也有丰富的配置方便用户调整。其中常用的几项调整为：

调整+/-Vout Diffferential Output Voltage:
1000BT，可调 UTP_ext_reg0x51，其中 bit[11:9]为粗调，[3:0]为细调
100BT，可调 UTP_ext_reg0x57，其中 bit[14:12]为粗调，[11:8]为细调
10BTe，可调 UTP_ext_reg0x57，其中 bit[6:4]为粗调，[3:0]为细调
以上值增加则输出幅度增大，粗调每增加 1 则幅度增大约 25%，细调每增加 1 则幅度增大约 1/16。

6.1.11 SYNC-E（时钟输出）配置

通过 common ext reg0xa006(SyncE0) or 0xa00e(SyncE1)进行 SyncE 操作；

1，bit[15]enable SyncE function；

2，bit[14]决定在 SyncE 没有 lock 时，是否依然输出 clk；

3，在没有 link 依然需要输出时，可以选择 pll 作为时钟源，同时需要把 bit[14]置'1'。如果需要始终输出，还需要把 sleep 功能关掉，否则进入 sleep 后，PLL clk 会关闭，这样就没有 clk 输出了。

4，bit[13][10]组合决定输出 clk freq；

5，bit[9:0]表示 SyncE 恢复的时钟源头。如果同时选择了多个源头，以优先级高的为准，源头的优先级从 bit[9]到 bit[0]逐步降低。如果没有选择任何一个源头，则以 bit[0]对应的 Phy0 作为时钟源头。

6，SyncE 驱动能力调整，可通过 common ext reg0xa020[13:12]： 00 为最低档，11 为最高档

7，phyaddr0/sync_lock0 和 phyaddr1/sync_lock1 为复用 pin，当它们做地址 pin pull down，做 sync_lock 指示时高电平有效；当它们做地址 pin pull up，做 sync_lock 时低电平有效。Sync_lock 的电平指示在 sync lock 上后输出

8，当选择 utp recovered rx clock 作为源头时，当 utp link 在 1000M slave mode/ 100M / 10M 时，源头来自 utp recovered rx clock；当 utp link 在 1000M master mode 时，源头来自本地 PLL。

Common EXT A006: SYNCE0 cfg				
Bit	Symbol	Access	Default	Description

15	sync_clk_en_0	RW	0x0	1=enable to output the synce_clk0 to the PAD SYNCE0
14	bp_sync_lock_gating_0	RW	0x0	When synce_clk0 is not locked, this bit controls whether to output the synce_clk0 or crystal clock to the PAD SYNCE0 1: to output; 0: to not output.
13	sel_synce_125m_0	RW	0x0	bit13 and bit10 controls the synce_clk0's frequence and source: {sel_synce_125m_0, sel_clk_25m_xtl_0}, 2'b00: output 25MHz syncE clock; 2'b01: output 31.25MHz syncE clock; 2'b10: output 125MHz syncE clock; 2'b11: output 25MHz reference clock.
12:11	reserved	RO	0x0	always 0.
10	sel_clk_25m_xtl_0	RW	0x0	refer to bit13 sel_synce_125m_0 for detail.
9	sel_rclk_sds1_0	RW	0x0	bit9:0, select the source of the synce_clk0 from the recovered RX clocks. For YT8618, the recovered RX clocks include that from 8 port copper PHYs and 2 5G serdes PHYs; for YT8614, the recovered RX clocks include that from 4 port copper PHY, 4 port Fiber/SGMII 1.25G serdes PHY and one 5G serdes PHY. MSB has the higher priority, for example, bit9:0=280h means to output the sds1's RX recovered clock; bit9:0=030h means to output the port 6's RX recovered clock. sds1, for YT8618, means the 5G serdes connected to port 4~7; for YT8614, means none. sds0, means the 5G serdes connencted to copper port 0~3. phy7, for YT8618, means the copper port 7; for YT8614, means the fiber/SGMII port 3. phy6, for YT8618, means the copper port 6; for YT8614, means the fiber/SGMII port 2. phy5, for YT8618, means the copper port 5; for YT8614, means the fiber/SGMII port 1. phy4, for YT8618, means the copper port 4; for YT8614, means the fiber/SGMII port 0. phy3, means the copper port 3. phy2, means the copper port 2. phy1, means the copper port 1. phy0, means the copper port 0.
8	sel_rclk_sds0_0	RW	0x0	
7	sel_rclk_phy7_0	RW	0x0	
6	sel_rclk_phy6_0	RW	0x0	
5	sel_rclk_phy5_0	RW	0x0	
4	sel_rclk_phy4_0	RW	0x0	
3	sel_rclk_phy3_0	RW	0x0	
2	sel_rclk_phy2_0	RW	0x0	
1	sel_rclk_phy1_0	RW	0x0	
0	sel_rclk_phy0_0	RW	0x0	

Common EXT A00E: SYNCE1 cfg

Bit	Symbol	Access	Default	Description
15	sync_clk_en_1	RW	0x0	1=enable to output the synce_clk1 to the PAD SYNCE1
14	bp_sync_lock_gating_1	RW	0x0	When synce_clk1 is not locked, this bit controls whether to output the synce_clk1 or crystal clock to the PAD SYNCE1 1: to output; 0: to not output.
13	sel_synce_125m_1	RW	0x0	bit13 and bit10 controls the synce_clk1's frequence and source: {sel_synce_125m_1, sel_clk_25m_xtl_1}, 2'b00: output 25MHz syncE clock; 2'b01: output 31.25MHz syncE clock; 2'b10: output 125MHz syncE clock; 2'b11: output 25MHz reference clock.
12:11	reserved	RO	0x0	always 0.
10	sel_clk_25m_xtl_1	RW	0x0	refer to bit13 sel_synce_125m_1 for detail.
9	sel_rclk_sds1_1	RW	0x0	bit9:0, select the source of the synce_clk1 from the recovered RX clocks. For YT8618, the recovered RX clocks include that from 8 port copper PHYs and 2 5G serdes PHYs; for YT8614, the recovered RX clocks include that from 4 port copper PHY, 4 port Fiber/SGMII 1.25G serdes PHY and one 5G serdes PHY. MSB has the higher priority, for example, bit9:0=280h means to output
8	sel_rclk_sds0_1	RW	0x0	
7	sel_rclk_phy7_1	RW	0x0	
6	sel_rclk_phy6_1	RW	0x0	
5	sel_rclk_phy5_1	RW	0x0	

4	sel_rclk_phy4_1	RW	0x0	the sds1's RX recovered clock; bit9:0=030h means to output the port 6's RX recovered clock. sds1, for YT8618, means the 5G serdes connected to port 4~7; for YT8614, means none. sds0, means the 5G serdes connencted to copper port 0~3. phy7, for YT8618, means the copper port 7; for YT8614, means the fiber/SGMII port 3. phy6, for YT8618, means the copper port 6; for YT8614, means the fiber/SGMII port 2. phy5, for YT8618, means the copper port 5; for YT8614, means the fiber/SGMII port 1. phy4, for YT8618, means the copper port 4; for YT8614, means the fiber/SGMII port 0. phy3, means the copper port 3. phy2, means the copper port 2. phy1, means the copper port 1. phy0, means the copper port 0.
3	sel_rclk_phy3_1	RW	0x0	
2	sel_rclk_phy2_1	RW	0x0	
1	sel_rclk_phy1_1	RW	0x0	
0	sel_rclk_phy0_1	RW	0x0	

6.1.12 Interrupt（中断）配置

通过 common ext reg0xa005[11]可修改 interrupt 的极性：1: INT_N is active LOW; 0: INT_N is active HIGH.

1. UTP/fiber/SGMII link up/down interrupt

- 通过 common ext reg0xa010 打开 link up/down interrupt 的 mask；
- 当产生 Interrupt 后通过读取 common ext reg0xa011 获取哪个 port 发生了 link up/down interrupt.

Common EXT A010: PHY Link Up/Down INTn Mask

Bit	Symbol	Access	default	Description
15	phy7_link_up_int_mask	RW	0x0	the interrupt mask bit for EXT A011 phy7_link_up_int. 1: to enable the interrupt.
14	phy6_link_up_int_mask	RW	0x0	the interrupt mask bit for EXT A011 phy6_link_up_int. 1: to enable the interrupt.
13	phy5_link_up_int_mask	RW	0x0	the interrupt mask bit for EXT A011 phy5_link_up_int. 1: to enable the interrupt.
12	phy4_link_up_int_mask	RW	0x0	the interrupt mask bit for EXT A011 phy4_link_up_int. 1: to enable the interrupt.
11	phy3_link_up_int_mask	RW	0x0	the interrupt mask bit for EXT A011 phy3_link_up_int. 1: to enable the interrupt.
10	phy2_link_up_int_mask	RW	0x0	the interrupt mask bit for EXT A011 phy2_link_up_int. 1: to enable the interrupt.
9	phy1_link_up_int_mask	RW	0x0	the interrupt mask bit for EXT A011 phy1_link_up_int. 1: to enable the interrupt.
8	phy0_link_up_int_mask	RW	0x0	the interrupt mask bit for EXT A011 phy0_link_up_int. 1: to enable the interrupt.
7	phy7_link_down_int_mask	RW	0x0	the interrupt mask bit for EXT A011 phy7_link_down_int. 1: to enable the interrupt.
6	phy6_link_down_int_mask	RW	0x0	the interrupt mask bit for EXT A011 phy6_link_down_int. 1: to enable the interrupt.
5	phy5_link_down_int_mask	RW	0x0	the interrupt mask bit for EXT A011 phy5_link_down_int. 1: to enable the interrupt.
4	phy4_link_down_int_mask	RW	0x0	the interrupt mask bit for EXT A011 phy4_link_down_int. 1: to enable the interrupt.
3	phy3_link_down_int_mask	RW	0x0	the interrupt mask bit for EXT A011 phy3_link_down_int. 1: to enable the interrupt.

2	phy2_link_down_int_mask	RW	0x0	the interrupt mask bit for EXT A011 phy2_link_down_int. 1: to enable the interrupt.
1	phy1_link_down_int_mask	RW	0x0	the interrupt mask bit for EXT A011 phy1_link_down_int. 1: to enable the interrupt.
0	phy0_link_down_int_mask	RW	0x0	the interrupt mask bit for EXT A011 phy0_link_down_int. 1: to enable the interrupt.

Common EXT A011: PHY Link Up/Down INTn Status

Bit	Symbol	Access	default	Description
15	phy7_link_up_int	RO	0x0	For YT8618, it's the UTP #7 link up interrupt status; for YT8614, it's the 1.25G SerDes #3 link up interrupt status.
14	phy6_link_up_int	RO	0x0	For YT8618, it's the UTP #6 link up interrupt status; for YT8614, it's the 1.25G SerDes #2 link up interrupt status.
13	phy5_link_up_int	RO	0x0	For YT8618, it's the UTP #5 link up interrupt status; for YT8614, it's the 1.25G SerDes #1 link up interrupt status.
12	phy4_link_up_int	RO	0x0	For YT8618, it's the UTP #4 link up interrupt status; for YT8614, it's the 1.25G SerDes #0 link up interrupt status.
11	phy3_link_up_int	RO	0x0	It's the UTP #3 link up interrupt status.
10	phy2_link_up_int	RO	0x0	It's the UTP #2 link up interrupt status.
9	phy1_link_up_int	RO	0x0	It's the UTP #1 link up interrupt status.
8	phy0_link_up_int	RO	0x0	It's the UTP #0 link up interrupt status.
7	phy7_link_down_int	RO	0x0	For YT8618, it's the UTP #7 link down interrupt status; for YT8614, it's the 1.25G SerDes #3 link down interrupt status.
6	phy6_link_down_int	RO	0x0	For YT8618, it's the UTP #6 link down interrupt status; for YT8614, it's the 1.25G SerDes #2 link down interrupt status.
5	phy5_link_down_int	RO	0x0	For YT8618, it's the UTP #5 link down interrupt status; for YT8614, it's the 1.25G SerDes #1 link down interrupt status.
4	phy4_link_down_int	RO	0x0	For YT8618, it's the UTP #4 link down interrupt status; for YT8614, it's the 1.25G SerDes #0 link down interrupt status.
3	phy3_link_down_int	RO	0x0	It's the UTP #3 link down interrupt status.
2	phy2_link_down_int	RO	0x0	It's the UTP #2 link down interrupt status.
1	phy1_link_down_int	RO	0x0	It's the UTP #1 link down interrupt status.
0	phy0_link_down_int	RO	0x0	It's the UTP #0 link down interrupt status.

2. UTP interrupt

A，通过 common ext reg0xa012[3:0]打开 UTP0~3 的 mask;

B，把 UTP0~3 端口的 MII reg0x12 的 Interrupt mask 根据需要置’1’;

C，当发生中断时，可以通过 common ext reg0xa013 获得哪个端口产生了中断，然后读取对应端口的 MII reg0x13，获得 Interrup 的详细信息。然后再读取一次 common ext reg0xa013,把它的中断信息清掉，细节参考 common ext reg0xa012[15]描述。

Common EXT A012: UTP Legacy Interrupt Mask

Bit	Symbol	Access	default	Description
15	en_clr_all_phys_intn	RW	0x0	1: reading EXT 0xA013 will clear itself and all UTP legacy interrupts' status in MII 0x13; 0: to clear EXT 0xA013, UTP legacy interrupts' status in MII 0x13 should be read clear first, then read clear 0xA013.
14:8	Reserved	RW	0x0	not used.
7	utp7_legacy_int_mask	RW	0x0	the interrupt mask bit for EXT 0xA013 utp7_legacy_int. 1: to enable the interrupt.

6	utp6_legacy_int_mask	RW	0x0	the interrupt mask bit for EXT 0xA013 utp6_legacy_int. 1: to enable the interrupt.
5	utp5_legacy_int_mask	RW	0x0	the interrupt mask bit for EXT 0xA013 utp5_legacy_int. 1: to enable the interrupt.
4	utp4_legacy_int_mask	RW	0x0	the interrupt mask bit for EXT 0xA013 utp4_legacy_int. 1: to enable the interrupt.
3	utp3_legacy_int_mask	RW	0x0	the interrupt mask bit for EXT 0xA013 utp3_legacy_int. 1: to enable the interrupt.
2	utp2_legacy_int_mask	RW	0x0	the interrupt mask bit for EXT 0xA013 utp2_legacy_int. 1: to enable the interrupt.
1	utp1_legacy_int_mask	RW	0x0	the interrupt mask bit for EXT 0xA013 utp1_legacy_int. 1: to enable the interrupt.
0	utp0_legacy_int_mask	RW	0x0	the interrupt mask bit for EXT 0xA013 utp0_legacy_int. 1: to enable the interrupt.
Common EXT A013: UTP Legacy Interrupt Status				
Bit	Symbol	Access	default	Description
15:8	Reserved	RO	0x0	always 0.
7	utp7_legacy_int	RO	0x0	the UTP legacy interrupt status bit for EXT 0xA013 utp7_legacy_int. It's asserted when UTP #7 MII 0x13 is not 0. When EXT 0xA012 bit15 is 1, it's read clear; else, UTP #7 MII 0x13 should be read clear first, then it will be cleared when read out.
6	utp6_legacy_int	RO	0x0	the UTP legacy interrupt status bit for EXT 0xA013 utp6_legacy_int. It's asserted when UTP #6 MII 0x13 is not 0. When EXT 0xA012 bit15 is 1, it's read clear; else, UTP #6 MII 0x13 should be read clear first, then it will be cleared when read out.
5	utp5_legacy_int	RO	0x0	the UTP legacy interrupt status bit for EXT 0xA013 utp5_legacy_int. It's asserted when UTP #5 MII 0x13 is not 0. When EXT 0xA012 bit15 is 1, it's read clear; else, UTP #5 MII 0x13 should be read clear first, then it will be cleared when read out.
4	utp4_legacy_int	RO	0x0	the UTP legacy interrupt status bit for EXT 0xA013 utp4_legacy_int. It's asserted when UTP #4 MII 0x13 is not 0. When EXT 0xA012 bit15 is 1, it's read clear; else, UTP #4 MII 0x13 should be read clear first, then it will be cleared when read out.
3	utp3_legacy_int	RO	0x0	the UTP legacy interrupt status bit for EXT 0xA013 utp3_legacy_int. It's asserted when UTP #3 MII 0x13 is not 0. When EXT 0xA012 bit15 is 1, it's read clear; else, UTP #3 MII 0x13 should be read clear first, then it will be cleared when read out.
2	utp2_legacy_int	RO	0x0	the UTP legacy interrupt status bit for EXT 0xA013 utp2_legacy_int. It's asserted when UTP #2 MII 0x13 is not 0. When EXT 0xA012 bit15 is 1, it's read clear; else, UTP #2 MII 0x13 should be read clear first, then it will be cleared when read out.
1	utp1_legacy_int	RO	0x0	the UTP legacy interrupt status bit for EXT 0xA013 utp1_legacy_int. It's asserted when UTP #1 MII 0x13 is not 0. When EXT 0xA012 bit15 is 1, it's read clear; else, UTP #1 MII 0x13 should be read clear first, then it will be cleared when read out.
0	utp0_legacy_int	RO	0x0	the UTP legacy interrupt status bit for EXT 0xA013 utp0_legacy_int. It's asserted when UTP #0 MII 0x13 is not 0. When EXT 0xA012 bit15 is 1, it's read clear; else, UTP #0 MII 0x13 should be read clear first, then it will be cleared when read out.

Phy MII 12h: Interrupt Mask Register				
Bit	Symbol	Access	Default	Description
15	Auto-Negotiation Error INT mask	RW	0x0	1 = Interrupt enable

14	Speed Changed INT mask	RW	0x0	1 = Interrupt enable
13	Duplex changed INT mask	RW	0x0	1 = Interrupt enable
12	Page Received INT mask	RW	0x0	1 = Interrupt enable
11	Link Failed INT mask	RW	0x0	1 = Interrupt enable
10	Link Succeed INT mask	RW	0x0	1 = Interrupt enable
9:7	reserved	RW	0x0	No used.
6	WOL INT mask	RW	0x0	1 = Interrupt enable
5	Wirespeed downgraded INT mask	RW	0x0	1 = Interrupt enable
4:2	Reserved	RW	0x0	No used.
1	Polarity changed INT mask	RW	0x0	1 = Interrupt enable
0	Jabber Happened INT mask	RW	0x0	1 = Interrupt enable

Phy MII 13h: Interrupt Mask Register

Bit	Symbol	Access	Default	Description
15	Auto-Negotiation Error INT	RW	0x0	Error can take place when any of the following
14	Speed Changed INT	RW	0x0	1 = Speed changed
13	Duplex changed INT	RW	0x0	1 = duplex changed
12	Page Received INT	RW	0x0	1 = Page received
11	Link Failed INT	RW	0x0	1 = Phy link down takes place
10	Link Succeed INT	RW	0x0	1 = Phy link up takes place
6	WOL INT	RW	0x0	1 = PHY received WOL magic frame.
5	Wirespeed downgraded INT	RW	0x0	1 = speed downgraded.
4:2	Reserved	RW	0x0	No used.
1	Polarity changed INT	RW	0x0	1 = PHY revered MDI polarity
0	Jabber Happened INT	RW	0x0	1 = 10BaseT TX jabber happened

3. QSGMII link up/down interrupt

- 在用到 QSGMII0 的模式，需要观测它的 link up/down 情况，可以通过 common ext reg0xa014 打开 interrupt mask；
- 当发生 QSGMII0 link up/dow 时，可以通过 common ext reg0xa015 获取中断状态信息。

Common EXT A014

Bit	Symbol	Access	default	Description
15	q1_ch3_link_up_int_mask	RO	0x0	the interrupt mask bit for EXT 0xA015 q1_ch3_link_up_int. 1: to enable the interrupt.
14	q1_ch2_link_up_int_mask	RO	0x0	the interrupt mask bit for EXT 0xA015 q1_ch2_link_up_int. 1: to enable the interrupt.
13	q1_ch1_link_up_int_mask	RO	0x0	the interrupt mask bit for EXT 0xA015 q1_ch1_link_up_int. 1: to enable the interrupt.
12	q1_ch0_link_up_int_mask	RO	0x0	the interrupt mask bit for EXT 0xA015 q1_ch0_link_up_int. 1: to enable the interrupt.
11	q0_ch3_link_up_int_mask	RO	0x0	the interrupt mask bit for EXT 0xA015 q0_ch3_link_up_int. 1: to enable the interrupt.
10	q0_ch2_link_up_int_mask	RO	0x0	the interrupt mask bit for EXT 0xA015 q0_ch2_link_up_int. 1: to enable the interrupt.
9	q0_ch1_link_up_int_mask	RO	0x0	the interrupt mask bit for EXT 0xA015 q0_ch1_link_up_int. 1: to enable the interrupt.
8	q0_ch0_link_up_int_mask	RO	0x0	the interrupt mask bit for EXT 0xA015 q0_ch0_link_up_int. 1: to enable the interrupt.
7	q1_ch3_link_down_int_mask	RO	0x0	the interrupt mask bit for EXT 0xA015

				q1_ch3_link_down_int. 1: to enable the interrupt.
6	q1_ch2_link_down_int_mask	RO	0x0	the interrupt mask bit for EXT 0xA015 q1_ch2_link_down_int. 1: to enable the interrupt.
5	q1_ch1_link_down_int_mask	RO	0x0	the interrupt mask bit for EXT 0xA015 q1_ch1_link_down_int. 1: to enable the interrupt.
4	q1_ch0_link_down_int_mask	RO	0x0	the interrupt mask bit for EXT 0xA015 q1_ch0_link_down_int. 1: to enable the interrupt.
3	q0_ch3_link_down_int_mask	RO	0x0	the interrupt mask bit for EXT 0xA015 q0_ch3_link_down_int. 1: to enable the interrupt.
2	q0_ch2_link_down_int_mask	RO	0x0	the interrupt mask bit for EXT 0xA015 q0_ch2_link_down_int. 1: to enable the interrupt.
1	q0_ch1_link_down_int_mask	RO	0x0	the interrupt mask bit for EXT 0xA015 q0_ch1_link_down_int. 1: to enable the interrupt.
0	q0_ch0_link_down_int_mask	RO	0x0	the interrupt mask bit for EXT 0xA015 q0_ch0_link_down_int. 1: to enable the interrupt.

Common EXT A015

Bit	Symbol	Access	default	Description
15	q1_ch3_link_up_int	RO	0x0	It's the QSGMII #1 channel #3 link up interrupt status.
14	q1_ch2_link_up_int	RO	0x0	It's the QSGMII #1 channel #2 link up interrupt status.
13	q1_ch1_link_up_int	RO	0x0	It's the QSGMII #1 channel #1 link up interrupt status.
12	q1_ch0_link_up_int	RO	0x0	It's the QSGMII #1 channel #0 link up interrupt status.
11	q0_ch3_link_up_int	RO	0x0	It's the QSGMII #0 channel #3 link up interrupt status.
10	q0_ch2_link_up_int	RO	0x0	It's the QSGMII #0 channel #2 link up interrupt status.
9	q0_ch1_link_up_int	RO	0x0	It's the QSGMII #0 channel #1 link up interrupt status.
8	q0_ch0_link_up_int	RO	0x0	It's the QSGMII #0 channel #0 link up interrupt status.
7	q1_ch3_link_down_int	RO	0x0	It's the QSGMII #1 channel #3 link down interrupt status.
6	q1_ch2_link_down_int	RO	0x0	It's the QSGMII #1 channel #2 link down interrupt status.
5	q1_ch1_link_down_int	RO	0x0	It's the QSGMII #1 channel #1 link down interrupt status.
4	q1_ch0_link_down_int	RO	0x0	It's the QSGMII #1 channel #0 link down interrupt status.
3	q0_ch3_link_down_int	RO	0x0	It's the QSGMII #0 channel #3 link down interrupt status.
2	q0_ch2_link_down_int	RO	0x0	It's the QSGMII #0 channel #2 link down interrupt status.
1	q0_ch1_link_down_int	RO	0x0	It's the QSGMII #0 channel #1 link down interrupt status.
0	q0_ch0_link_down_int	RO	0x0	It's the QSGMII #0 channel #0 link down interrupt status.

6.1.13 Fast link down 配置

YT8614 支持 fast link down 功能，但默认是关闭的，以 port0 为例，打开步骤如下：

1、使能 link down 中断

2、打开 fast link down 功能，write_utp_ext_reg(PhyBaseAddr+0, 0x34[15],1’b0)

3、设定 fast link down 时间，write_utp_ext_reg(PhyBaseAddr+0, 0x37[15:13],3’b0)，时间选项如下：

Bit	Symbol	Access	default	Description
15:13	fld_timer_sel	RW	0x0	fast link down timer sel. 3'b000=0ms; 3'b001=5ms; 3'b010=10ms; 3'b011=20ms; 3'b100=40ms; 3'b101=80ms; 3'b110=160ms; 3'b111=320ms

6.1.14 SNR 读取（当前连接质量指示）

YT8614 的 utp ext reg 0x5A, 0x5B, 0x5C, 0x5D 内的 bit[14:0]值分别表征了 4 对 MDI 线的 SNR，以 Port0 为例，读取步骤如下：

步骤一，write_utp_ext_reg(PhyBaseAddr+0, 0x59[15],1'b1)

步骤二，write_ext_reg(PhyBaseAddr+0, 0xa080[10],1'b0)

步骤三，分别读取 phy ext 扩展寄存器 0x5A,0x5B, 0x5C, 0x5D 的 bit[14:0]

MSE(0) = read_utp_ext_reg(PhyBaseAddr+0, 0x5A)

MSE(1) = read_utp_ext_reg(PhyBaseAddr+0, 0x5B)

MSE(2) = read_utp_ext_reg(PhyBaseAddr+0, 0x5C)

MSE(3) = read_utp_ext_reg(PhyBaseAddr+0, 0x5D)

将每个值读至少 100 次，并取平值，得到每对 MDI 线上稳定 MSE

步骤四，将上述稳定 MSE 值，按照如下公式计算 SNR：

千兆：10*log10(29696/mse)

百兆：10*log10(32768/mse)

要达到稳定连接，SNR 值应大于 24.73dB (千兆，对应 MSE 约为 100)； 19.13dB (百兆，对应 MSE 约 400)；

附千兆情况下几个 MSE 与 SNR 的对应表：

MSE	SNR
118	24
187	22
297	20
374	19
470	18
592	17
746	16

6.1.15 Idle error counter（1000BT）

在 1000BT 连接情况下，线上一直在发 idle 信号，PHY 内部也有检查 idle 码型是否有错的机制，并把错误的个数放到 utp_mii_reg0xa[7:0]中。如果读到此统计值有错，那在数据传输情况下也会有错码。

Phy MII 0Ah: MASTER-SLAVE status register				
Bit	Symbol	Access	Default	Description
15	Master/Slave_cfg_error	RO RC SWC LH	0x0	This register bit will clear on read, rising of MII 0.12 and rising of AN complete. 1 = Master/Slave configuration fault detected 0 = No fault detected
14	Master/Slave	RO	0x0	This bit is not valid unless register 0x1 bit5 is 1. 1 = Local PHY configuration resolved to Master 0 = Local PHY configuration resolved to Slave
13	Local Receiver Status	RO	0x0	1 = Local Receiver OK 0 = Local Receiver not OK
12	Remote Receiver	RO	0x0	1 = Remote Receiver OK 0 = Remote Receiver not OK
11	Link Partner	RO	0x0	This bit is not valid unless register 0x1 bit5 is 1. 1 = Link Partner supports 1000BASE-T half duplex 0 = Link Partner does not support 1000BASE-T half duplex
10	Link Partner	RO	0x0	This bit is not valid unless register 0x1 bit5 is 1. 1 = Link Partner supports 1000Base-T full duplex 0 = Link Partner does not support 1000Base-T full duplex

9:8	Reserved	RO	0x0	Reserved
7:0	Idle Error Count	RO RC	0x0	MSB of Idle Error Counter. The register indicates the idle error count since the last read operation performed to this register. The counter pegs at 11111111 and does not roll over.

6.1.16 Big error counter

当 UTP 出现错包，丢包或断线的时候，可能是 UTP 连接通路质量不好（比如线太长或太差），或者出现了短时的大干扰。前者可以通过读 SNR 相关寄存器获取，后者通过 big error counter 或者 clipping 寄存器得到。前者说明出现了较大的 burst error，后者说明在 ADC 处出现了饱和的现象。出现该现象时，不一定会有丢包发生。其步骤为：

1. 把 common ext reg0xa000 写为 0x0
2. 把 UTP_ext_reg0xa080[10]清零
3. 把 UTP_ext_reg0x59[14:13]置 1
4. 读取 UTP_ext_reg0x69~0x70 的值， 分 channel 0, 1, 2, 3 来判断是否发生 big error

UTP EXT F8h: Clock gating ctrl

Bit	Symbol	Access	default	Description
10	En_gate_UTPdbg	RW	0x1	gating UTPdbg module when it is diable

UTP EXT 59h: debug cfg

Bit	Symbol	Access	default	Description
14	Cnt_err_auto	RW	0x0	1 = monitor big slicer error after 1000BT training done;
13	Cnt_clp_auto	RW	0x0	1 = monitor big ADC output after 1000BT training done;

UTP EXT 69h: debug mon16

Bit	Symbol	Access	default	Description
15:0	Cnt_err0_big_pre	RO RC	0x0	It's fixed to 0 when EXT 59h bit14 cnt_err_auto is set, otherwise, it's the channel 0's big slicer error counter before the latest assertion of RX_DV.
7:0	Cnt_err0_big	RO RC	0x0	The counter of Channel 0's slicer error that larger than the EXT 59h bit7~0 err_big_th.

UTP EXT 6Ah: debug mon17

Bit	Symbol	Access	default	Description
15:0	Cnt_err1_big_pre	RO RC	0x0	Same as EXT 69h, except it's channel 1's big error counter.
7:0	Cnt_err1_big	RO RC	0x0	Same as EXT 69h, except it's channel 1's big error counter.

UTP EXT 6Bh: debug mon18

Bit	Symbol	Access	default	Description
15:0	Cnt_err2_big_pre	RO RC	0x0	Same as EXT 69h, except it's channel 2's big error counter.
7:0	Cnt_err2_big	RO RC	0x0	Same as EXT 69h, except it's channel 2's big error counter.

UTP EXT 6Ch: debug mon19

Bit	Symbol	Access	default	Description

15:0	Cnt_err3_big_pre	RO RC	0x0	Same as EXT 69h, except it's channel 3's big error counter.
7:0	Cnt_err3_big	RO RC	0x0	Same as EXT 69h, except it's channel 3's big error counter.
UTP EXT 6Dh: debug mon20				
Bit	Symbol	Access	default	Description
15:0	Cnt_clp0_pre	RO RC	0x0	It's fixed to 0 when EXT 59h bit13 cnt_clp_auto is set, otherwise, it's the channel 0's big ADC output counter before the latest assertion of RX_DV.
7:0	Cnt_clp0	RO RC	0x0	The counter of Channel 0's ADC output that larger than the EXT 58h bit11~8 adc_clp_th.
UTP EXT 6Eh: debug mon21				
Bit	Symbol	Access	default	Description
15:0	Cnt_clp1_pre	RO RC	0x0	Same as EXT 69h, except it's channel 1's ADC output clipping counter.
7:0	Cnt_clp1	RO RC	0x0	Same as EXT 69h, except it's channel 1's ADC output clipping counter.
UTP EXT 6Fh: debug mon22				
Bit	Symbol	Access	default	Description
15:0	Cnt_clp2_pre	RO RC	0x0	Same as EXT 69h, except it's channel 2's ADC output clipping counter.
7:0	Cnt_clp2	RO RC	0x0	Same as EXT 69h, except it's channel 2's ADC output clipping counter.
UTP EXT 70h: debug mon23				
Bit	Symbol	Access	default	Description
15:0	Cnt_clp3_pre	RO RC	0x0	Same as EXT 69h, except it's channel 3's ADC output clipping counter.
7:0	Cnt_clp3	RO RC	0x0	Same as EXT 69h, except it's channel 3's ADC output clipping counter.

6.2 QSGMII 配置

6.2.1 QSGMII 寄存器读写方式

QSGMII 寄存器的读写，需要先选择 QSGMII 地址空间，再对相应的 QSGMII 口（根据 QSGMII 对应的 PhyAddr）的寄存器进行读/写操作，比如：

write_ext_reg(PhyBaseAddr, 0xa000, 0x2)　#选择 QSGMII 地址空间

1、read_qsgmii_mii_reg(PhyBaseAddr+0, 0x1)　#读取 QSGMII channel 0 的 mii_reg0x1 的值

2、write_qsgmii_mii_reg(PhyBaseAddr+1, 0x0, 0x9140)　#写 QSGMII channel 1 的 mii_reg0x0 的值为 0x9140

3、read_qsgmii_ext_reg(PhyBaseAddr+0, 0x27)　#读取 QSGMII 0 的 ext_reg0x27 的值

4、write_qsgmii_ext_reg(PhyBaseAddr+0, 0x27, 0x2010)　#写 QSGMII 0 的 ext_reg0x27 的值为 0x2010

注意：

- 对 QSGMII 寄存器操作一定要确保当前选择的是 QSGMII 地址空间。
- QSGMII 每个 channel 都有独立的一套 mii 寄存器，对 QSGMII 的 channel mii 寄存器操作一定要指定 QSGMII channel 的 PhyAddr，即 PhyBaseAddr+PhyOffsetAddr（port0~port3 的 PhyOffsetAddr 分别为 0，1，2，3）。

- QSGMII 还有一套 ext 寄存器，4 个 channel 共用一套 ext 寄存器，所以对应于 QSGMII channel 的不同 PhyAddr 是访问的同一套寄存器。

6.2.2 AN（自协商）配置

QSGMII 包含 4 个 channel 的 SGMII，分别对应 4 个 UTP 或 Fiber。QSGMII 会把与之对应的 UTP 或 Fiber 的连接信息，包括 link up/down, speed and duplex，通过 AN (自协商)机制传递给 MAC 端的 QSGMII。

QSGMII 的自协商是通过每个 channel 的 qsgmii_mii_reg0x0[12]来单独设置：

- 打开自协商：write_qsgmii_mii_reg0x0[12]:1'b1，默认是打开的。
- 关闭自协商：write_qsgmii_mii_reg0x0[12]:1'b0。

6.2.3 位同步，连接和速率

QSGMII 四个 channel 的状态（位同步、连接和速率），都是通过各自 channel 的 mii reg 0x11 获取的。

- 位同步：read_qsgmii_mii_reg0x11[0] = 1'b1，表示已经同步；1'b0 表示未同步。
- Link state：read_qsgmii_mii_reg0x11[10] = 1'b1，表示 link up；1'b0 表示 link down。
- Speed：read_qsgmii_mii_reg0x11[15:14] = 2'b10，表示 1000M；2'b01，表示 100M，2'b00，表示 10M。
- Duplex：read_qsgmii_mii_reg0x11[13] = 1'b1，表示 full duplex； 1'b0 表示 half duplex。

Sds MII 11h: Sds specific status register				
Bit	Symbol	Access	Default	Description
15:14	Speed_mode	RO	0x0	These status bits are valid only when bit11 is 1. Bit11 is set when Auto-Negotiation is completed or Auto-Negotiation is disabled.
13	Duplex	RO	0x0	This status bit is valid only when bit11 is 1. Bit11 is set when Auto-Negotiation is completed or Auto-Negotiation is disabled.
12:11	Pause	RO	0x0	Pause to mac
10	Link status real-time	RO	0x0	1 = Link up
9	Rx_lpi_active	RO	0x0	rx lpi is active
8	Duplex_error	RO	0x0	realtime duplex error
7	En_flowctrl_rx	RO	0x0	realtime en_flowctrl_rx
6	En_flowctrl_tx	RO	0x0	realtime en_flowctrl_tx

5:4	Ser_mode_cfg	RO	0x0	realtime serdes working mode.
				00: SG_MAC;
				01: SG_PHY;
				10: FIB_1000;
				11: FIB_100.

3:1	Xmit	RO	0x0	realtime transmit statemachine.
				001: Xmit Idle;
				010: Xmit Config;
				100: Xmit Data.

0	Syncstatus	RO	0x0	realtime syncstatus

6.2.4 PRBS/BERT

配置步骤如下：,

1. 将 DUT 自身的 QSGMII TX 与 RX 相连（或者 DUT 的 SGMII 与另一端 Link partner 的 QSGMII 相连）。

2. 配置 DUT（或 DUT 与 link partner）寄存器来设置要发送和接收的 PRBS 码型，支持的码型包括：PRBS7, PRBS10, PRBS31 及自定义。

3. 配置 DUT（或 DUT 与 link partner）寄存器打开 PRBS 和 BERT 功能。

4. 读取 DUT（或 DUT 与 link partner）寄存器以检查 QSGMII 是否同步上，且无误码

相关内部寄存器的各 bit 说明如下：

5G Sds EXT 05h: sds prbs cfg1		**0x05**		
Bit	Symbol	Access	Default	Description
15	En_prbs	RW	0x0	enable TX PRBS or self-defined pattern, the pattern type is determined by bit7:5 test_mode.
14	En_bert	RW	0x0	enable Bit Error Rate Test;
13	Prbs_invert	RW	0x0	TX error injection function, it has highest polarity than prbs_err_once and prbs_err_cont. 1: to send polarity inverted PRBS or self-defined pattern;
12	Prbs_err_cont	RW	0x0	TX error injection function, it has the lowest polarity.1: to inject error on TX pattern continuously and periodic, the period is controlled by prbs_err_rate.
11:10	Prbs_err_rate	RW	0x0	It’s valid only when prbs_err_cont is 1. 2'b00: inject one error every 1024 cycles; 2'b01: inject one error every 2048 cycles; 2'b10: inject one error every 4096 cycles; 2'b11: inject one error every 8192 cycles;
9	Prbs_err_once	RW	0x0	TX error injection function.At the rising of this bit, to send polarity inverted PRBS or self-defined pattern for one cycle;
8	Test_mode_prbs31	RW	0x0	1: PRBS31, bit7:5 test_mode[2:0] has no effect.0: not PRBS31, bit7:5 test_mode[2:0] take effect then.
7:5	Test_mode	RW	0x0	Control the TX pattern in test mode: 3'h0, PRBS7; 3'h1, PRBS10; 3'h2, Fix pattern, the fix pattern is controlled by Ext.6;3'h3, 010101…; 3'h4, 00110011…; 3'h5, 00000_11111_00000_11111…; 3'h6, 0000_0000_00_1111_1111_11…; 3'h7, Increase pattern, 0->1023->0->1023…
4	Duration_check_en	RW	0x0	enable fixed number of bits check define in sds EXT BA/BB
3:0	Reserved	RO	0x0	Reserved
5G Sds EXT 06h: sds prbs cfg2		0x06		
Bit	Symbol	Access	Default	Description
15:10	Reserved	RO	0x0	Reserved
9:0	Fix_pattern	RW	0x0	fix pattern transmited in test_mode 2
5G Sds EXT 08h: sds prbs mon1		0x08		
Bit	Symbol	Access	Default	Description
15:9	Fix_pattern	RO	0x0	Reserved
8	Err	RO	0x0	Err flag after bert has synchronized
7:1	Reserved	RO	0x0	Reserved
0	Bitsync	RO	0x0	real time synchronization status
5G Sds EXT 09h: sds prbs mon2		0x09		

Bit	Symbol	Access	Default	Description
15:0	Err_cnt_15_0	RO	0x0	real time lowest 16 bits received error bit counter
5G Sds EXT 0Ah: sds prbs mon3		0x0A		
Bit	Symbol	Access	Default	Description
15:0	Err_cnt_31_16	RO	0x0	real time highest 16 bits received error bit counter

自发自收 PRBS，配置如下：

write qsgmii_ext_reg 0x5 0xc000/0xc020/0xc100 (设置发送码型为 PRBS7/PRBS10/PRBS31，开始发送，并打开 BERT)

read qsgmii_ext_reg 0x8 (若为 1，表示 bit sync)

read qsgmii_ext_reg 0x9 (读出的值为接收 bit error 的统计，低 16 位)

read qsgmii_ext_reg 0xa (读出的值为接收 bit error 的统计，高 16 位)

作为接收端接收 PRBS，配置如下：

write qsgmii_ext_reg 0x5 0x4000 /0x4020/0x4100 (设置接收码型为 PRBS7/10/31，不发送码型，只打开 BERT/打开 bit error cnt)

read qsgmii_ext_reg 0x8 (若为 1，表示 bit sync)

read qsgmii_ext_reg 0x9 (读出的值为接收 bit error 的统计，低 16 位)

read qsgmii_ext_reg 0xa (读出的值为接收 bit error 的统计，高 16 位)

6.2.5 发送幅度配置

QSGMII_ext_reg0xa1[15:12]代表 QSGMII 驱动能力。可以用来调节 QSGMII 的发送幅度。默认值为 4’b1010。值越大，发送幅度越高。

5G Sds EXT A1h: analog cfg2				
Bit	Symbol	Access	default	Description
15:12	Tx_driver_stg2	RW	0xA	TX driver stage2 amplitude control bit<3:2>: 00 +0mA 01: +2.5mA 10:+2.5mA 11+5mA bit<1:0>: 00 +0mA 01: +0.625mA 10: +1.25mA 11:+2.5mA
11	Tx_driver_stg1	RW	0x1	TX driver stage1 amplitude control
10:8	Reserved	RO	0x0	Reserved
7:6	vb_tx_term	RW	0x2	tx output common mode voltage when tx power down 00: 0.7V 01:0.8V 10:0.9V 11:1V
5	tx_vamp_post	RW	0x0	TX driver post stage1 amplitude control.
4	Reserved	RO	0x0	always 0.
3:1	tx_driver_post	RW	0x7	TX driver stage2 de-emphasize control bit<2>: 0 +0mA 1: +2.5mA bit<1:0>: 00 +0mA 01: +0.625mA 10: +1.25mA 11:+2.5mA
0	Tx_pd	RW	0x0	power down analog tx

6.2.6 发送预加重配置

YT8614 QSGMII 有两个预加重：main 和 post；可通过 QSGMII ext_reg0xa1 设置；

初始化设置：

PHY X (X 为指定需要配置的 PHY 地址)

write ext reg 0xa007 0xf004 (设置 PHY Serdes 为 QSGMII 模式)

write ext reg 0xa000 0x2 (选择 SGMII SDS 寄存器空间)

具体的 bit 位如下：

read qsgmii_ext_reg 0xa1 #读取 sds ext_reg0xa1 的寄存器值，并修改如下几个 bit，并重新写入该寄存器(write ext reg 0xa1 xxxx)：

Main cursor: 3 个 bit, write qsgmii_ext_reg 0xa1[15:14]:2'b11 最大，2'b00 最小;

Post cursor: 1 个 bit, write qsgmii_ext_reg 0xa1[5]，Post_Enable；

Post cursor: 3 个 bit, write qsgmii_ext_reg 0xa1[2:1]: 2'b11 最大，2'b00 最小；

注意: qsgmii_ext_reg 0xa1 的其它位请保留原值不要变

6.2.7 Loopback（回环）模式

1、Internal loopback 情况下，SERDES 的 analog 电路会被 bypass，直接将 GMII 发送的数据从 SERDES 的数字电路回环回去，也可以称作 digital loopback，如下图所示。配置方法如下：

Write_qsgmii_ext_reg(PhyBaseAddr+0, 0x2[10], 1'b1)QSGMII 4 个 channel 设为 internal loopback

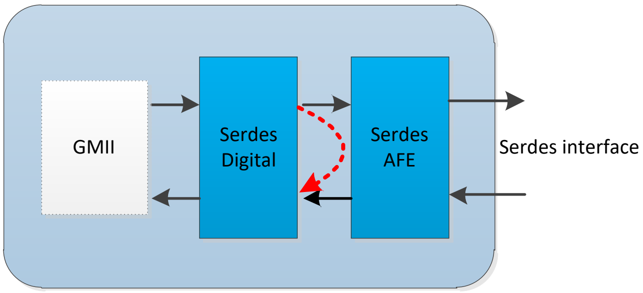
另外还可分别配置 QSGMII 每个 channel internal loopback：

Channel 0: Write_qsgmii_mii_reg(PhyBaseAddr+0, 0x0[14], 1'b1)

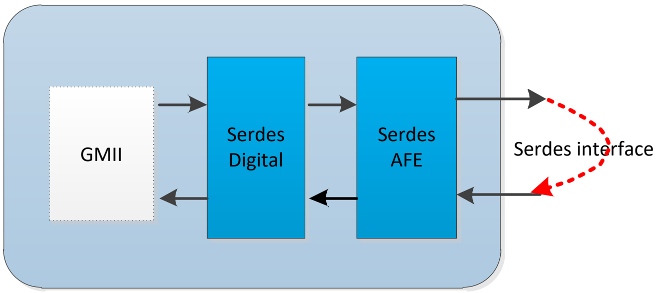
Channel 1: Write_qsgmii_mii_reg(PhyBaseAddr+1, 0x0[14], 1'b1)

Channel 2: Write_qsgmii_mii_reg(PhyBaseAddr+2, 0x0[14], 1'b1)

Channel 3: Write_qsgmii_mii_reg(PhyBaseAddr+3, 0x0[14], 1'b1)



2、在 External Loopback 模式下，SERDES 发送的信号由回环线直接回到 SERDES 的接收电路。回环线将 SERDES 的发送和接收互连即可，不需要配寄存器。



3、在 Remote Loopback 模式下，芯片内部的 GMII RX 接收的数据被直接注入 GMII TX。寄存器在 common 地址空间，配置方法如下：

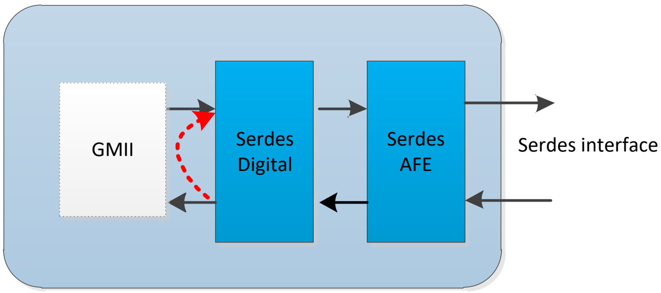
分别配置 QSGMII 每个 channel remote loopback：

Channel 0: Write_ext_reg(PhyBaseAddr+0, 0xa00d[0], 1'b1)

Channel 1: Write_ext_reg(PhyBaseAddr+1, 0xa00d[1], 1'b1)

Channel 2: Write_ext_reg(PhyBaseAddr+2, 0xa00d[2], 1'b1)

Channel 3: Write_ext_reg(PhyBaseAddr+3, 0xa00d[3], 1'b1)



6.3 SGMII 配置

6.3.1 SGMII 寄存器读写方式

SGMII 寄存器的读写，需要先选择 SGMII 地址空间，再对相应的 SGMII port 的寄存器进行读/写操作，比如：

write_ext_reg(PhyBaseAddr, 0xa000, 0x3)　　#选择 SGMII 地址空间

1、read_sgmii_mii_reg(PhyBaseAddr+0, 0x1)　　#读取 sgmii port0 的 mii_reg0x1 的值

2、write_sgmii_mii_reg(PhyBaseAddr+1, 0x0, 0x9140)　#写 sgmii port1 的 mii_reg0x0 的值为 0x9140

3、read_sgmii_ext_reg(PhyBaseAddr+3, 0x27)　　#读取 sgmii port3 的 ext_reg0x27 的值

4、write_sgmii_ext_reg(PhyBaseAddr+3, 0x27, 0x2010)　　#写 sgmii port3 的 ext_reg0x27 的值为 0x2010

注意：

- 对 SGMII 寄存器操作一定要确保当前选择的是 SGMII 地址空间。
- 每个 SGMII 都有独立的一套 mii 和 ext 寄存器，对 SGMII 寄存器操作一定要指定 SGMII 的 PhyAddr，即 PhyBaseAddr+PhyOffsetAddr（port0~port3 的 PhyOffsetAddr 分别为 0，1，2，3）。

6.3.2 AN（自协商）配置

SGMII 分为 SGMII_PHY 模式和 SGMII_MAC 模式。在 Qsgmii x1 + Sgmii(MAC) x4 mode 中是 SGMII_MAC，其余模式中都是 SGMII_PHY。

SGMII_PHY 需要把与之对应的 UTP 或 Fiber 的连接信息，包括 link up/down, speed and duplex，通过 SGMII 的 AN (自协商)机制传递给 SGMII_MAC，最终实现 SGMII_MAC，SGMII_PHY 与网口速率保持一致。

SGMII 可以通过 write_qsgmii_mii_reg0x0[12]:1'b1 打开自协商。

SGMII 也可以关闭自协商进入强制（force）模式。

- 对于 SGMII_PHY，只需要 write_sgmii_mii_reg0x0[12]:1'b0 关闭自协商。SGMII_PHY 传递的速率和双工模式由对应的 UTP/Fiber 决定。
- 对于 SGMII_MAC，除了 write_sgmii_mii_reg0x0[12]:1'b0 关闭自协商，还需要设置速率和双工模式。具体操作如下：
- Force SGMII transfer 1000FULL:　write_sgmii_mii_reg0x0: 0x8140
- Force SGMII transfer 100FULL:　write_sgmii_mii_reg0x0: 0xa100
- Force SGMII transfer 100HALF:　write_sgmii_mii_reg0x0: 0xa000
- Force SGMII transfer 10FULL:　write_sgmii_mii_reg0x0: 0x8100

- Force SGMII transfer 10HALF： write_sgmii_mii_reg0x0: 0x8000

6.3.3 位同步，连接和速率

SGMII 的状态（位同步、连接和速率），是通过 sgmii mii reg 0x11 获取的。

- 位同步：read_qsgmii_mii_reg0x11[0] = 1’b1，表示已经同步；1’b0 表示未同步。
- Link state：read_qsgmii_mii_reg0x11[10] = 1’b1，表示 link up；1’b0 表示 link down。
- Speed：read_qsgmii_mii_reg0x11[15:14] = 2’b10，表示 1000M；2’b01，表示 100M，2’b00，表示 10M。
- Duplex：read_qsgmii_mii_reg0x11[13] = 1’b1，表示 full duplex； 1’b0 表示 half duplex。

Sds MII 11h: Sds specific status register				
Bit	Symbol	Access	Default	Description
15:14	Speed_mode	RO	0x0	These status bits are valid only when bit11 is 1. Bit11 is set when Auto-Negotiation is completed or Auto-Negotiation is disabled.
13	Duplex	RO	0x0	This status bit is valid only when bit11 is 1. Bit11 is set when Auto-Negotiation is completed or Auto-Negotiation is disabled.
12:11	Pause	RO	0x0	Pause to mac
10	Link status real-time	RO	0x0	1 = Link up
9	Rx_lpi_active	RO	0x0	rx lpi is active
8	Duplex_error	RO	0x0	realtime duplex error
7	En_flowctrl_rx	RO	0x0	realtime en_flowctrl_rx
6	En_flowctrl_tx	RO	0x0	realtime en_flowctrl_tx

5:4	Ser_mode_cfg	RO	0x0	realtime serdes working mode.
				00: SG_MAC;
				01: SG_PHY;
				10: FIB_1000;
				11: FIB_100.

3:1	Xmit	RO	0x0	realtime transmit statemachine.
				001: Xmit Idle;
				010: Xmit Config;
				100: Xmit Data.

0	Syncstatus	RO	0x0	realtime syncstatus

6.3.4 PRBS/BERT

具体步骤为：

1. 将 DUT 自身的 SGMII TX 与 RX 相连（或者 DUT 的 SGMII 与另一端 Link partner 的 SGMII 相连）。
2. 配置 DUT（或 DUT 与 link partner）寄存器来设置要发送和接收的 PRBS 码型，支持的码型包括：PRBS7, PRBS10, PRBS31 及自定义。
3. 配置 DUT（或 DUT 与 link partner）寄存器打开 PRBS 和 BERT 功能。
4. 读取 DUT（或 DUT 与 link partner）寄存器以检查 SGMII 是否同步上，且无误码。

相关内部寄存器的各 bit 说明如下：

Sds EXT 05h: sds prbs cfg1		**0x05**		

Bit	Symbol	Access	Default	Description
15	En_prbs	RW	0x0	enable TX PRBS or self-defined pattern, the pattern type is determined by bit7:5 test_mode.
14	En_bert	RW	0x0	enable Bit Error Rate Test;
13	Prbs_invert	RW	0x0	TX error injection function, it has highest polarity than prbs_err_once and prbs_err_cont. 1: to send polarity inverted PRBS or self-defined pattern;
12	Prbs_err_cont	RW	0x0	TX error injection function, it has the lowest polarity.1: to inject error on TX pattern continuously and periodic, the period is controlled by prbs_err_rate.
11:10	Prbs_err_rate	RW	0x0	It's valid only when prbs_err_cont is 1. 2'b00: inject one error every 1024 cycles; 2'b01: inject one error every 2048 cycles; 2'b10: inject one error every 4096 cycles; 2'b11: inject one error every 8192 cycles;
9	Prbs_err_once	RW	0x0	TX error injection function.At the rising of this bit, to send polarity inverted PRBS or self-defined pattern for one cycle;
8	Test_mode_prbs31	RW	0x0	1: PRBS31, bit7:5 test_mode[2:0] has no effect.0: not PRBS31, bit7:5 test_mode[2:0] take effect then.
7:5	Test_mode	RW	0x0	Control the TX pattern in test mode: 3'h0, PRBS7; 3'h1, PRBS10; 3'h2, Fix pattern, the fix pattern is controlled by Ext.6;3'h3, 010101…; 3'h4, 00110011…; 3'h5, 00000_11111_00000_11111…; 3'h6, 0000_0000_00_1111_1111_11…; 3'h7, Increase pattern, 0->1023->0->1023…
4	Duration_check_en	RW	0x0	enable fixed number of bits check define in sds EXT BA/BB
3:0	Reserved	RO	0x0	Reserved
Sds EXT 06h: sds prbs cfg2		0x06		
Bit	Symbol	Access	Default	Description
15:10	Reserved	RO	0x0	Reserved
9:0	Fix_pattern_9_0	RW	0x0	fix pattern transmited in test_mode 2
Sds EXT 08h: sds prbs mon1		0x08		
Bit	Symbol	Access	Default	Description
15:9	Reserved	RO	0x0	Reserved
8	Err	RO	0x0	Err flag after bert has synchronized
7:1	Reserved	RO	0x0	Reserved
0	Bitsync	RO	0x0	real time synchronization status
Sds EXT 09h: sds prbs mon2		0x09		
Bit	Symbol	Access	Default	Description
15:0	Err_cnt_15_0	RO	0x0	real time lowest 16 bits received error bit counter
Sds EXT 0Ah: sds prbs mon3		0x0A		

Bit	Symbol	Access	Default	Description
15:0	Err_cnt_31_16	RO	0x0	real time highest 16 bits received error bit counter

自发自收 PRBS，配置如下：

write ext reg 0x5 0xc000/0xc020/0xc100 (设置发送码型为 PRBS7/PRBS10/PRBS31，开始发送，并打开 BERT)

read ext reg 0x8 (若为 1，表示 bit sync)

read ext reg 0x9 (读出的值为接收 bit error 的统计，低 16 位)

read ext reg 0xa (读出的值为接收 bit error 的统计，高 16 位)

作为接收端接收 PRBS，配置如下：

write ext reg 0x5 0x4000 /0x4020/0x4100 (设置接收码型为 PRBS7/10/31，不发送码型，只打开 BERT/打开 bit error cnt)

read ext reg 0x8 (若为 1，表示 bit sync)

read ext reg 0x9 (读出的值为接收 bit error 的统计，低 16 位)

read ext reg 0xa (读出的值为接收 bit error 的统计，高 16 位)

6.3.5 发送预加重配置

YT8614 SGMII 有两个预加重：main 和 post；可通过 SDS ext_reg0xa1 设置；

初始化设置：

PHY X (X 为指定需要配置的 PHY 地址)

write ext reg 0xa007 0xf003 (设置 PHY Serdes 为 SGMII 模式)

write ext reg 0xa000 0x3 (选择 SGMII SDS 寄存器空间)

具体的 bit 位如下：

read ext reg 0xa1 #读取 sds ext_reg0xa1 的寄存器值，并修改如下几个 bit，并重新写入该寄存器 (write ext reg 0xa1 xxxx)：

Main cursor: 3 个 bit, write ext_reg 0xa1[15:13]: 3’b111 最大，3’b000 最小;

Post cursor: 3 个 bit, write ext_reg 0xa1[5]: 1’b1 最大，1’b0 最小;

注意: ext_reg 0xa1 的其它位请保留原值不要变。

6.3.6 接收的最大幅值

YT8521SH SGMII 标称允许接收的单端最大幅值(R_Y2)是 600mV，差分最大幅值(VRX_DIFFFp-p)是 1200mV。从仿真的结果来看，YT8521SH 允许接收更高的幅值，单端最大幅值（R_Y2）达到 800mV，差分最大幅值（VRX_DIFFFp-p）达到 1600mV。

6.3.7 输出极性反转配置

PHY 的 SGMII 支持+/-极性反转，包括发送和接收两方向，分别的配置如下：

HSIP、HSIN：sgmii_ext_reg0x02 Bit4,默认为 0,；该 bit 为 1，则 HSIP 、HSIN 极性互换

HSOP、HSON：sgmii_ext_reg0x02 Bit5,默认为 0,；该 bit 为 1，则 HSOP 、HSON 极性互换

6.3.8 Loopback（回环）模式

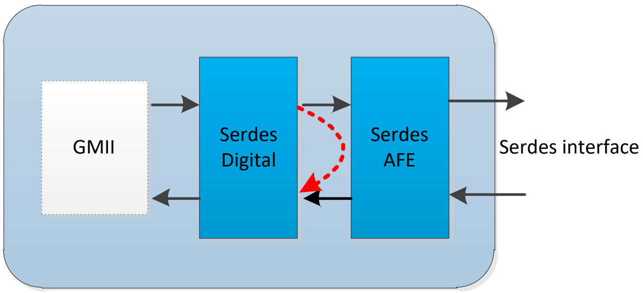
1、Internal loopback 情况下，SERDES 的 analog 电路会被 bypass，直接将 GMII 发送的数据从 SERDES 的数字电路回环回去，也可以称作 digital loopback，如下图所示。配置方法如下：

Port 0: Write_sgmii_mii_reg(PhyBaseAddr+0, 0x0[14], 1’b1)

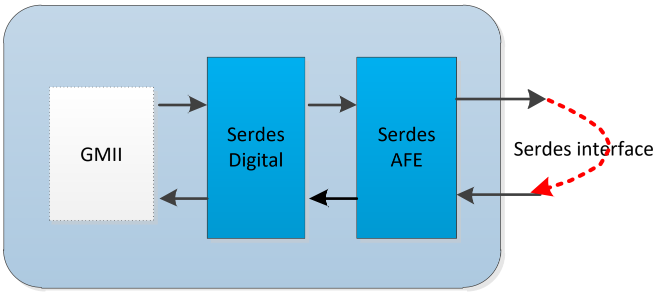
Port 1: Write_sgmii_mii_reg(PhyBaseAddr+1, 0x0[14], 1’b1)

Port 2: Write_sgmii_mii_reg(PhyBaseAddr+2, 0x0[14], 1’b1)

Port 3: Write_sgmii_mii_reg(PhyBaseAddr+3, 0x0[14], 1’b1)



2、在 External Loopback 模式下，SERDES 发送的信号由回环线直接回到 SERDES 的接收电路。回环线将 SERDES 的发送和接收互连即可，不需要配寄存器。



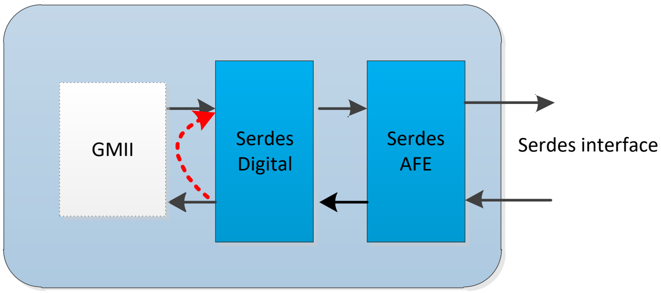
3、在 Remote Loopback 模式下，芯片内部的 GMII RX 接收的数据被直接注入 GMII TX。寄存器在 common 地址空间，配置方法如下：

Port 0: Write_ext_reg(PhyBaseAddr+0, 0xa009[8], 1’b1)

Port 1: Write_ext_reg(PhyBaseAddr+1, 0xa009[9], 1’b1)

Port 2: Write_ext_reg(PhyBaseAddr+2, 0xa009[10], 1’b1)

Port 3: Write_ext_reg(PhyBaseAddr+3, 0xa009[11], 1’b1)



6.4 Fiber 配置

6.4.1 Fiber 寄存器读写方式

Fiber 地址空间与 SGMII 一样，寄存器的读写，需要先选择 SGMII 地址空间，再对相应的 Fiber 口的寄存器进行读/写操作，比如：

write_ext_reg(PhyBaseAddr, 0xa000, 0x3) #选择 SGMII 地址空间

read_sgmii_mii_reg(PhyBaseAddr+0, 0x1) #读取 fiber port0 的 mii_reg0x1 的值

write_sgmii_mii_reg(PhyBaseAddr+1, 0x0, 0x9140) #写 fiber port1 的 mii_reg0x0 的值为 0x9140

read_sgmii_ext_reg(PhyBaseAddr+3, 0x27) #读取 fiber port3 的 ext_reg0x27 的值

write_sgmii_ext_reg(PhyBaseAddr+3, 0x27, 0x2010) #写 fiber port3 的 ext_reg0x27 的值为 0x2010

注意：

- 对 fiber 寄存器操作一定要确保当前选择的是 SGMII 地址空间。
- 每个 fiber 都有独立的一套 mii 和 ext 寄存器，对 fiber 寄存器操作一定要指定 SGMII 的 PhyAddr，即 PhyBaseAddr+PhyOffsetAddr（port0~port3 的 PhyOffsetAddr 分别为 0，1，2，3）。

6.4.2 速率和双工

Fiber auto sensing 功能默认是打开的，即 Fiber 会自动检测插入的是百兆还是千兆光模块来适配到相应的连接速度。Fiber 当前状态，可以通过对应的 sgmii port mii reg 0x11 读到：

A. Link state：bit 10。1’b1 表示当前是 link up 的。

B. speed：bit 15:14。2b’10: 1000M， 2b’01: 100M.

C. duplex: bit 13。1’b1 表示 full duplex

Sds MII 11h: Sds specific status register

Bit	Symbol	Access	Default	Description
15:14	Speed_mode	RO	0x0	These status bits are valid only when bit10 is 1. Bit10 is set when Auto-Negotiation is completed or Auto-Negotiation is disabled. 10---1000M；01---100M；
13	Duplex	RO	0x0	This status bit is valid only when bit10 is 1. Bit10 is set when Auto-Negotiation is completed or Auto-Negotiation is disabled. 1: full duplex; 0: half duplex
12:11	Pause	RO	0x0	Pause to mac
10	Link status real-time	RO	0x0	1 = Link up
9	Rx_lpi_active	RO	0x0	rx lpi is active
8	Duplex_error	RO	0x0	realtime duplex error
7	En_flowctrl_rx	RO	0x0	realtime en_flowctrl_rx
6	En_flowctrl_tx	RO	0x0	realtime en_flowctrl_tx
5:4	Ser_mode_cfg	RO	0x0	realtime serdes working mode
3:1	Reserved	RO	0x0	Reserved
0	Syncstatus	RO	0x0	realtime syncstatus

6.4.3 Fiber 强制 speed 说明

Fiber auto sensing 功能默认是打开的，强制 speed 首先要关闭 fiber auto sensing 功能，然后强制 speed 为 1000BX 或者 100FX，配置如下：

Port0:

write_sgmii_ext_reg(PhyBaseAddr+0, 0xa5[15], 1’b0) 关闭 autosensing；

write_ext_reg(PhyBaseAddr+0, 0xa009[0], 1’b1) 强制为 1000BX，0xa009 在 common 地址空间。

write_ext_reg(PhyBaseAddr+0, 0xa009[0], 1’b0) 强制为 100FX

write_sgmii_mii_reg(PhyBaseAddr+0, 0x0, 0x9140)

Port1:

write_sgmii_ext_reg(PhyBaseAddr+1, 0xa5[15], 1’b0) 关闭 autosensing；

write_ext_reg(PhyBaseAddr+1, 0xa009[1], 1’b1) 强制为 1000BX， 0xa009 在 common 地址空间。

write_ext_reg(PhyBaseAddr+1, 0xa009[1], 1’b0) 强制为 100FX

write_sgmii_mii_reg(PhyBaseAddr+1, 0x0, 0x9140)

Port2:

write_sgmii_ext_reg(PhyBaseAddr+2, 0xa5[15], 1’b0) 关闭 autosensing；

write_ext_reg(PhyBaseAddr+2, 0xa009[2], 1’b1) 强制为 1000BX， 0xa009 在 common 地址空间。

write_ext_reg(PhyBaseAddr+2, 0xa009[2], 1’b0) 强制为 100FX

write_sgmii_mii_reg(PhyBaseAddr+2, 0x0, 0x9140)

Port3:

write_sgmii_ext_reg(PhyBaseAddr+3, 0xa5[15], 1’b0) 关闭 autosensing；

write_ext_reg(PhyBaseAddr+3, 0xa009[3], 1’b1) 强制为 1000BX， 0xa009 在 common 地址空间。

write_ext_reg(PhyBaseAddr+3, 0xa009[3], 1’b0) 强制为 100FX

write_sgmii_mii_reg(PhyBaseAddr+3, 0x0, 0x9140)

6.4.4 4UTP to 4Fiber 模式说明

该模式下，UTP 的 10/00/1000 能力是随跟 Fiber 状态而自动变化，且 Fiber 不 link，UTP 是不会 link 的。具体如下：

A. Fiber link down：UTP 内部自动进入 power down 模式，即使和 Link partner 对接，也不会 link 上；

B. Fiber link 在 1000BX：UTP 自动配置 mii reg 4 = 0x1c01, reg 9 = 0x200，表示只有千兆的能力，并与对端 UTP 协商；

C. Fiber link 在 100FX: UTP 自动配置 mii reg 4 = 0x1d81, reg 9 = 0x0，表示只有 100M full/half 的能力，并与对端 UTP 协商；

在 4UTP-4Fiber 模式下，Fiber auto sensing 功能默认也是打开的，即 Fiber 会自动检测插入的是百兆还是千兆光模块来适配到相应的连接速度。需要强制 speed，请参考 6.4.3

LED 灯配置

6.4.5 Loopback（回环）模式

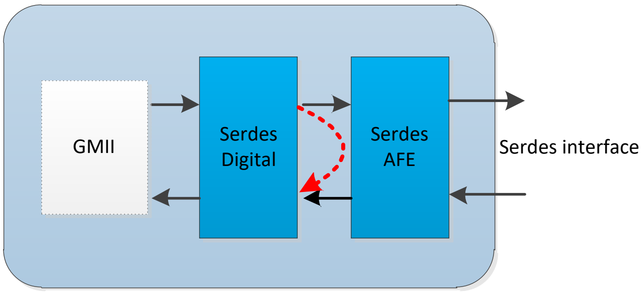
1、Internal loopback 情况下，SERDES 的 analog 电路会被 bypass，直接将 GMII 发送的数据从 SERDES 的数字电路回环回去，也可以称作 digital loopback，如下图所示。配置方法如下：

Port 0: Write_sgmii_mii_reg(PhyBaseAddr+0, 0x0[14], 1’b1)

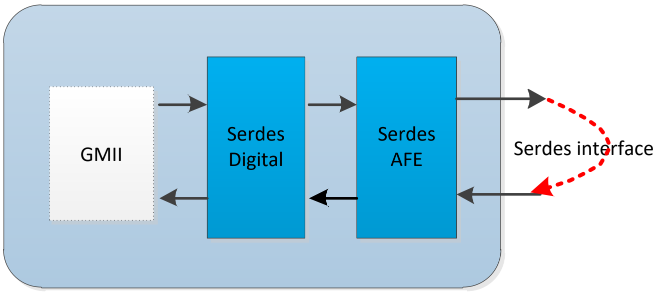
Port 1: Write_sgmii_mii_reg(PhyBaseAddr+1, 0x0[14], 1’b1)

Port 2: Write_sgmii_mii_reg(PhyBaseAddr+2, 0x0[14], 1’b1)

Port 3: Write_sgmii_mii_reg(PhyBaseAddr+3, 0x0[14], 1’b1)

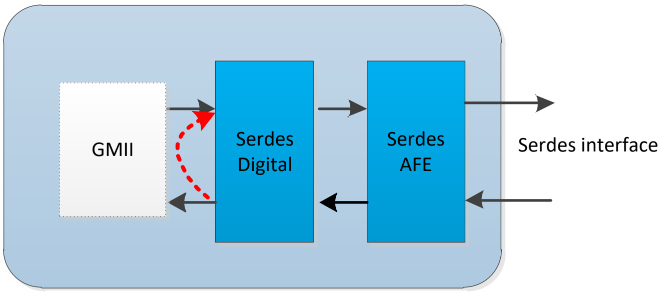


2、在 External Loopback 模式下，SERDES 发送的信号由回环线直接回到 SERDES 的接收电路。回环线将 SERDES 的发送和接收互连即可，不需要配寄存器。



3、在 Remote Loopback 模式下，芯片内部的 GMII RX 接收的数据被直接注入 GMII TX。寄存器在 common 地址空间，配置方法如下：

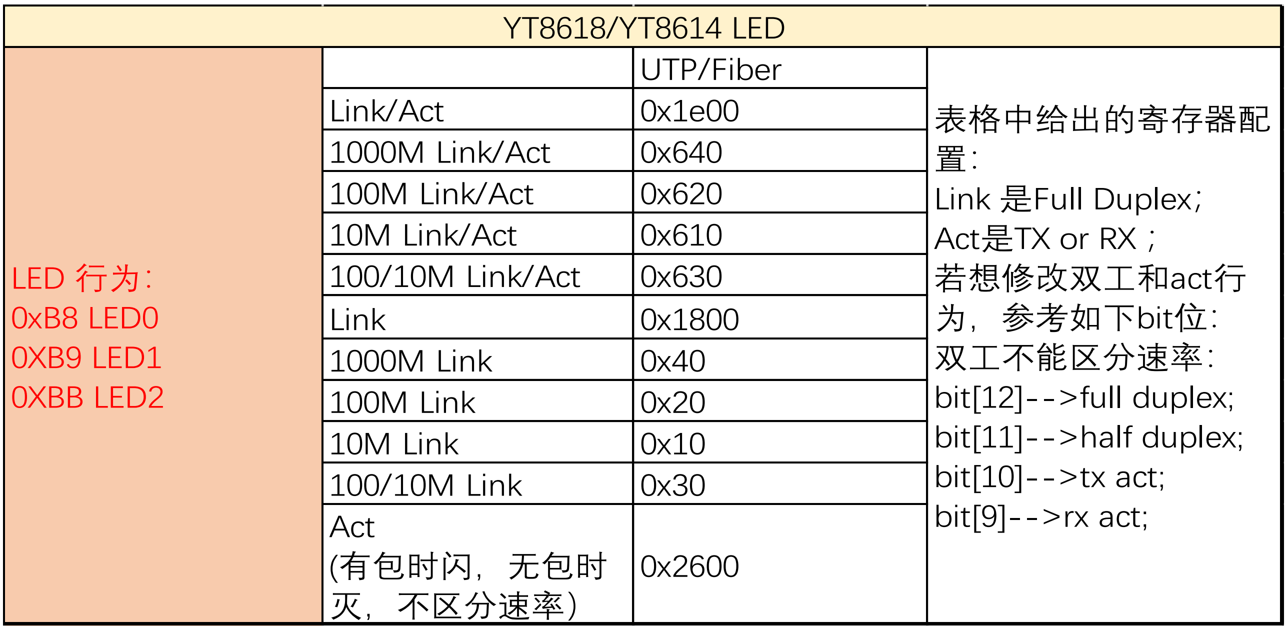
Port 0: Write_ext_reg(PhyBaseAddr+0, 0xa009[8], 1’b1)
Port 1: Write_ext_reg(PhyBaseAddr+1, 0xa009[9], 1’b1)
Port 2: Write_ext_reg(PhyBaseAddr+2, 0xa009[10], 1’b1)
Port 3: Write_ext_reg(PhyBaseAddr+3, 0xa009[11], 1’b1)



6.5 LED 灯配置

erial LED，default LED_Clock 推出 36bit（LED0-LED35），per port 支持最多 3 个灯，占用 24bit（LED12-LED35），有 12bit（LED0-LED11）无效；

0xA001	Bit[11]	[0]=active low	选择LED active 电平
		[1]=active high	
	Bit[10]	[0]=Disable LED2	单独开启/关闭其中一组LED
		[1]=Enable LED2	
	Bit[9]	[0]=Disable LED1	
		[1]=Enable LED1	
	Bit[8]	[0]=Disable LED0	
		[1]=Enable LED0	
0xA004	Bit[5:4]	[00]=per port 1 LED	MDI: [link/act]； FIBER: [link/act]
		[01]=per port 2 LED	MDI: [speed1000/act] [speed100(10)/act]； FIBER: [speed1000/act] [speed100/act]
		[10]=per port 3LED	MDI: [speed1000/act] [speed100/act] [speed10/act]; FIBER: [speed1000/act] [speed100/act] [Disable]
		[11]=per port 3LED（default）	MDI: [link/act] [speed1000] [speed100] FIBER: [link/act] [speed1000] [speed100]
	Bit[6]	[0]=Disable serial LED_data	
		[1]=Enable serial LED_data	
	Bit[7]	[0]=Enable serial LED_clock	
		[1]=Disable serial LED_clock	
	Bit[8]	[0]=Enable serial LED	
		[1]=Disable serial LED	
0xA007	Bit[4]	[0]=Enable serial LED	可以通过power on strapping 控制
		[1]=Disable serial LED	
0xBA	Bit[3:2]	[00]=4Hz	Blink Mode2
		[01]=8Hz（default）	
		[10]=16Hz	
		[11]=32Hz	
	Bit[1:0]	[00]=4Hz	Blink Mode1
		[01]=8Hz	
		[10]=16Hz（default）	
		[11]=32Hz	
0xB7	Bit[15]	[0]=Blink Mode1	选择 Blinking Mode
		[1]=Blink Mode2	
	Bit[8]	[1]=LED2 force_En	LED force Mode
	Bit[7:6]	[00]=force LED2 OFF	
		[01]=force LED2 ON	
		[10]=force LED2 Blink at Blink Mode1	
		[11]=force LED2 Blink at Blink Mode2	
	Bit[5]	[1]=LED1 force_En	
	Bit[4:3]	[00]=force LED1 OFF	
		[01]=force LED1 ON	
		[10]=force LED1 Blink at Blink Mode1	
		[11]=force LED1 Blink at Blink Mode2	
	Bit[2]	[1]=LED0 force_En	
	Bit[1:0]	[00]=force LED0 OFF	
		[01]=force LED0 ON	
		[10]=force LED0 Blink at Blink Mode1	
		[11]=force LED0 Blink at Blink Mode2	



YT8618/YT8614 LED			
LED 行为： 0xB8 LED0 0XB9 LED1 0XBB LED2		UTP/Fiber	表格中给出的寄存器配置： Link 是Full Duplex； Act是TX or RX ； 若想修改双工和act行为，参考如下bit位： 双工不能区分速率： bit[12]-->full duplex; bit[11]-->half duplex; bit[10]-->tx act; bit[9]-->rx act;
	Link/Act	0x1e00	
	1000M Link/Act	0x640	
	100M Link/Act	0x620	
	10M Link/Act	0x610	
	100/10M Link/Act	0x630	
	Link	0x1800	
	1000M Link	0x40	
	100M Link	0x20	
	10M Link	0x10	
	100/10M Link	0x30	
	Act (有包时闪，无包时灭，不区分速率)	0x2600	

默认 LED12-LED35 对应的寄存器和端口顺序如下，可以修改任意 LED 对应任意端口；

PHY/Mode		YT8618 QSGMII*2==>8*UTP		YT8614 QSGMII==>4*Combo	
寄存器	LED_MODE	2’b11	PortN	2’b11	PortN
0xa077	LED35	0x0000	P0 LED0	0x0000	P0 LED0
0xa076	LED34	0x0010	P1 LED0	0x0010	P1 LED0
0xa075	LED33	0x0020	P2 LED0	0x0020	P2 LED0
0xa074	LED32	0x0030	P3 LED0	0x0030	P3 LED0
0xa073	LED31	0x0040	P4 LED0	0x0001	P0 LED1
0xa072	LED30	0x0050	P5 LED0	0x0011	P1 LED1
0xa071	LED29	0x0060	P6 LED0	0x0021	P2 LED1
0xa070	LED28	0x0070	P7 LED0	0x0031	P3 LED1
0xa06f	LED27	0x0001	P0 LED1	0x0002	P0 LED2
0xa06e	LED26	0x0011	P1 LED1	0x0012	P1 LED2
0xa06d	LED25	0x0021	P2 LED1	0x0022	P2 LED2
0xa06c	LED24	0x0031	P3 LED1	0x0032	P3 LED2
0xa06b	LED23	0x0041	P4 LED1	0x0040	Fiber0 LED0
0xa06a	LED22	0x0051	P5 LED1	0x0050	Fiber1 LED0
0xa069	LED21	0x0061	P6 LED1	0x0060	Fiber2 LED0
0xa068	LED20	0x0071	P7 LED1	0x0070	Fiber3 LED0
0xa067	LED19	0x0002	P0 LED2	0x0041	Fiber0 LED1
0xa066	LED18	0x0012	P1 LED2	0x0051	Fiber1 LED1
0xa065	LED17	0x0022	P2 LED2	0x0061	Fiber2 LED1
0xa064	LED16	0x0032	P3 LED2	0x0071	Fiber3 LED1
0xa063	LED15	0x0042	P4 LED2	0x0042	Fiber0 LED2
0xa062	LED14	0x0052	P5 LED2	0x0052	Fiber1 LED2
0xa061	LED13	0x0062	P6 LED2	0x0062	Fiber2 LED2
0xa060	LED12	0x0072	P7 LED2	0x0072	Fiber3 LED2

YT8618		Default LED 顺序
0xa004 bit[5:4]	[10][11]=per port 3LED LED0 & LED1 & LED2	P7LED0…→P1 LED0 →P0 LED0→…P7LED1…→P1 LED1 →P0 LED1→…P7LED2…→P1 LED2 →P0 LED2
	[01]=per port 2 LED LED0 & LED1	P7LED0…→P1 LED0 →P0 LED0→…P7LED1…→P1 LED1 →P0 LED1
	[00]=per port 1 LED LED0	P7LED0…→P1 LED0 →P0 LED0

YT8614		Default LED 顺序
0xa004 bit[5:4]	[10][11]=per port 3LED LED0 & LED1 & LED2	Fiber3LED0…→Fiber1 LED0 →Fiber0 LED0→…Fiber3LED1…→Fiber1 LED1 →Fiber0 LED1→…Fiber3LED2…→Fiber1 LED2 →Fiber0 LED2→Fiber3LED0…→ P1 LED0 →P0 LED0→…P3LED1…→P1 LED1 →P0 LED1→…P3LED2…→P1 LED2 →P0 LED2
	[01]=per port 2 LED LED0 & LED1	Fiber3LED0…→Fiber1 LED0 →Fiber0 LED0→…Fiber3LED1…→Fiber1 LED1 →Fiber0 LED1→P3LED0…→P1 LED0 →P0 LED0→…P3LED1…→P1 LED1 →P0 LED1
	[00]=per port 1 LED LED0	Fiber3LED0…→Fiber1 LED0 →Fiber0 LED0→P3LED0…→P1 LED0 →P0 LED0

6.6 其它维测寄存器

6.6.1 用户自配置寄存器

因为 DUT 寄存器一般只在下电或硬复位后才会清掉，如果 DUT 工作中遇到异常，客户想确认发生异常时是否发生了掉电或硬复位，就需要一个指定的寄存器，在初始化时写入值，在发生异常时读取此值。此寄存器为 common 扩展寄存器 ext_reg0xa0ff，其默认值为 0x0。

6.6.2 PLL 锁定指示

DUT 内部的 PLL 在时钟信号出现很大频偏时，会发生失锁，从而导致内部工作异常。查看其指示的步骤如下：

步骤一，把 common ext reg0xa000 写 0x0；
步骤二，把 UTP_ext_reg0x53[7]置 1；
步骤三，check UTP_ext_reg0x56[1:0]，若这 2 位不同时为 0，则 PLL unlock。
步骤四，把 common ext reg0xa000 写 0x2；
步骤五，设置 serdes ext reg0xa0[10]为 1；
步骤六，check serdes ext reg0xa0[1:0]，若这 2 为不同时为 0，则 PLL unlock；
备注：
对 UTP 或 serdes 进行访问，首先需要确认是否当前模式是 UTP 还是 serdes 模式；
对输入的 clk，一般偏差到 10MHz 以上才会出现 PLL unlock。

UTP EXT 53h: analog cfg4				
Bit	Symbol	Access	default	Description
7	en_vco_check	RW	0x0	enable or disable VCO_fast or VCO_slow check circuit

UTP EXT 56h: analog cfg7				
Bit	Symbol	Access	default	Description
1	Pll_vco_h	RO	0x0	PLL fast indicator
0	Pll_vco_l	RO	0x0	PLL slow indicator

Sds EXT A0h: analog cfg1				
Bit	Symbol	Access	default	Description
10	Pll_en_vco_check	RW	0x0	enable or disable VCO_fast or VCO_slow check circuit
1	Pll_vco_h	RO	0x0	pll_vco_high flag
0	Pll_vco_l	RO	0x0	pll_vco_low flag

6.6.3 包生成器和收发包统计

YT8614 内部包含 4 个包生成器和 4 个收发包统计，分别位于芯片内部的 4 个数据通道上，寄存器均在 common 地址空间，下面以包生成器和收发包统计 0 介绍配置方法。

包生成器和收发包统计 0: 0xA0A0 ~ 0xA0B9

包生成器和收发包统计 1: 0xA1A0 ~ 0xA1B9

包生成器和收发包统计 2: 0xA2A0 ~ 0xA2B9

包生成器和收发包统计 3: 0xA3A0 ~ 0xA3B9

6.6.3.1 包生成器

1. 先设定包生成器发送包的方向，是 upstream media 还是 downstream media。寄存器地址 0xA0B7[13]，默认为 downstream media 方向。

Bit	Symbol	Access	default	Description
13	u0_pkgen_txdir_sel	RW	0x0	0=to downstream or to UTP in utp2fib; 1=to upstream or to fiber in utp2fib

芯片每种模式下，upstream media 和 downstream media 对应如下：

Chip Mode	**Downstream media 方向** 0xA0B7[13]=1’b0	**Upstream media 方向** 0xA0B7[13]=1’b1
QSGMII x 1 + Combo x 4	对应的 Combo port	对应的 QSGMII channel
QSGMII x 1 + Copper x 4	对应的 Copper port	对应的 QSGMII channel
QSGMII x 1 + Fiber x 4	对应的 Fiber port	对应的 QSGMII channel
QSGMII x 1 + SGMAC x 4	对应的 SGMAC port	对应的 QSGMII channel
SGMII x 4 + Copper x 4	对应的 Copper port	对应的 SGMII port
Fiber x 4 + Copper x 4	对应的 Copper port	对应的 Fiber port

2. 设置 packet generator 的发送包的长度: write_ext_reg 0xA0A1[15:0]
3. 设置发送包的个数: write_ext_reg 0xA0A2[15:0]，值为 0 表示一直发包。
4. 设置 IPG 长度：write_ext_reg 0xA0A0[7:4]，不用时可以不设置。
5. 还可以设置 packet DA/SA 地址，不用时可以不设置。
6. 使能该功能，write_ext_reg 0xA0B7[12] = 1’b1, 关闭该功能，write_ext_reg 0xA0B7[12] = 1’b0
7. 设置 DA 的低 8 位, write_ext_reg 0xA0B[15:8], 则 DA 地址为 00 00 00 00 00 value
8. 设置 SA 的低 8 位, write_ext_reg 0xA0B7[7:0], 则 SA 地址为 00 00 00 00 00 value
9. 打开 packet generator:
10. write_ext_reg 0xA0A0[14:13] = 3’b00, 配置 generator
11. write_ext_reg 0xA0A0[12] = 1’b1, 打开 packet generator 开始发包.
12. Polling read_ext_reg 0xA0A0[12], 为 1’b0 时，表示发完了设置的包个数。包个数设置为 0 时，write_ext_reg 0xA0A0[12] = 1’b0, 表示停止发包,
13. 关闭 packet generator:
14. write_ext_reg 0xA0A0[14] = 1’b1，关闭 packet generator clock
15. write_ext_reg 0xA0A0[13, 12] = 2’b10，关闭 generator

6.6.3.2 收发包统计

1、先设定收发包统计的 data source，是 upstream media 还是 downstream media。寄存器地址 0xA0B7[15:14]，默认为 downstream media。

15	u0_pkgchk_rxsrc_sel	RW	0x0	control RX checker's data source. =0, RX checker checks the downstream media's GMII RX data or in UTP2Fiber mode, the UTP's GMII RX data, =1, it checks upstream media's GMII RX data or in UTP2Fiber mode, the Fiber's GMII RX data
14	u0_pkgchk_txsrc_sel	RW	0x0	When ext.A0b7.10 is 0, it controls TX checker's data source. =0, TX checker checks the downstream media's GMII TX data or in UTP2Fiber mode, the UTP's GMII TX data; =1, it checks upstream media's GMII TX data or in UTP2Fiber mode, the Fiber's GMII TX data. It is not valid when ext.A0b7.10 is 1.

芯片每种模式下，upstream media 和 downstream media 分别对应如下：

Chip Mode	Downstream media 0xA0B7[15:14]=2'b00	Upstream media 0xA0B7[15:14]=2'b11
QSGMII x 1 + Combo x 4	对应的 Combo port	对应的 QSGMII channel
QSGMII x 1 + Copper x 4	对应的 Copper port	对应的 QSGMII channel
QSGMII x 1 + Fiber x 4	对应的 Fiber port	对应的 QSGMII channel
QSGMII x 1 + SGMAC x 4	对应的 SGMAC port	对应的 QSGMII channel
SGMII x 4 + Copper x 4	对应的 Copper port	对应的 SGMII port
Fiber x 4 + Copper x 4	对应的 Copper port	对应的 Fiber port

2、打开 packet checker: write_ext_reg 0xA0A0[15:14]=2'b10

3、关闭 packet checker: write_ext_reg 0xA0A0.[15:14]=2'b01

4、查看 checker 对应包长的收发包的统计结果:

common reg 0xA0A3~0xA0A8，代表从选定的 media 收到的正确包的数量，寄存器为 read clear

common reg 0xA0A9~0xA0AC，代表从选定的 media 收到的 crc error 包数量，寄存器为 read clear

common reg 0xA0AD~0xA0B2，代表从选定的 media GMII 收到的正确包的数量，寄存器为 read clear

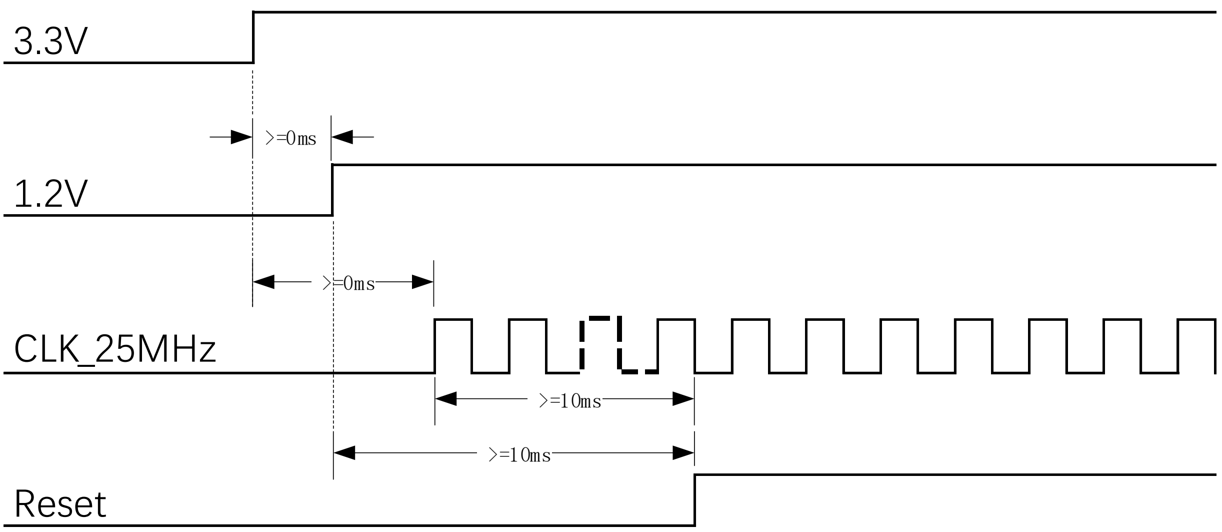
common reg 0xA0B3~0xA0B6，代表从选定的 media GMII 收到的 crc error 包数量，寄存器为 read clear

7 硬件电路相关

7.1 上电复位时序要求

YT8614 芯片支持多种模式。

它的上电时序要求如下：



说明：

1. 其中 1.2V 和 CLK_25MHz 并不要求先后，但要求 1.2V 和 CLK_25MHz 都稳定并保持至少 10ms 后再释放 Reset 信号。

7.2 PHY 地址分配要求

一颗 YT8614 会占用 8 个 mdio phy 地址，假定 POS(power on strapping)的 PhyBaseAddr=0x0, YT8614 会占用 8 个 phy 地址：0x0 ~ 0x3 供用户访问内部寄存器使用，详细请查看 3.3 节描述，0x4~0x7 被芯片内部测试占用，用户不能使用，所以用户在同一 MDIO 总线最多只能挂 4 颗 YT8614：POS 地址设定分别为 0x0,0x8,0x10,0x18。

7.3 快速上下电的要求

YT8614 快速上下电的要求和条件，根据实验和仿真，结论如下：

1、当 3.3V 电压掉到 0~2V，再上升回 3.3V，芯片会正常工作；

2、当 1.2V 电压掉到 0~0.4V，再上升到 1.2V, 芯片会正常工作；

3、当 3.3V 电源掉到 2V~2.97V，再上升到 3.3V，或者 1.2V 电源掉到 0.4~1.08V,再上升到 1.2V, 需要硬件复位芯片 reset pin 脚，才能保证芯片工作正常。

7.4 硬复位后延时进行 MDIO 操作要求

硬复位从低变高后，芯片需要在一段时间内完成内部的复位和 power-on-strapping 工作，因此一般要求在硬复位释放至少 100ms 后再进行 MDIO 操作。

另外，需要注意的是 MDIO/MDC 管脚也复用作 I2C，在上电复位生效后的约 100us 内，MDC/MDIO 线上会发一段 I2C 的波形（约 2ms）来侦测是否挂有 I2C 设备（比如 EEPROM），若收不到回应，则转为 MDC/MDIO 功能。上述复位 100ms 后进行 MDIO 操作完全可以避开这段时间。

7.5 外接参考时钟的抖动要求

通常可以接无源晶体在 XTL_IN 和 XTL_OUT 管脚，产生 DUT 的参考时钟。若要外接时钟信号，具体接法参见 datasheet。要求外灌 25MHz 的时钟抖动范围为：

12k~10MHz rms jitter < 0.5ps

（其它的要求，比如幅度，占空比等参见 datasheet）

7.6 外接参考时钟源切换操作要求

在外接参考时钟的情况下，如果有多个时钟源，并且有时钟源切换的要求，那对切换过程的要求如下：

1. 因为切换过程中，时钟的行为及对芯片造成的影响无法预测，而硬复位比软复位更为彻底，从系统可靠性角度看，建议切换时钟源后用硬复位将 PHY 复位。

2. 硬复位的时序，请参考本应用说明的复位时序要求(即 RESET 信号应该在时钟稳定后保持为低至少 10ms，再拉高/释放)。

3. 硬复位释放后，会清掉之前软件所配的所有的寄存器设置。MDIO 需要等至少 100ms 之后再进行操作，软件可通过 MDIO 将所需的寄存器重新配置（一般与上电初始化的寄存器配置相同）。

7.7 数据传输通路延时

系统实现精准时钟同步功能时，需要知道数据从 MDI 接口到 QSGMII/SGMII 接口，或从 QSGMII/SGMII 到 MDI 接口，所花的时间。包括时间的绝对值，和时间的抖动范围(Variation)。在各模式下，其值如下：

mode	direction	speed	basic delay(ns)	variation(ns)
sgmii<->utp	sgmii_rx->utp_tx	1000M	296	16
		100M	672	104
	utp_rx->sgmii_tx	1000M	368	24
		100M	688	72
qsgmii<->utp	qsgmii_rx->utp_tx	1000M	280	12
		100M	648	104
	utp_rx->sgmii_tx	1000M	344	24
		100M	664	72
qsgmii<->1000bx	qsgmii_rx->1000bx_tx	1000M	216	16
	1000bx_rx->qsgmii_tx	1000M	208	16
qsgmii<->100fx	qsgmii_rx->100fx_tx	100M	640	72
	100fx_rx->qsgmii_tx	100M	628	72

8 PCB 设计规则

8.1 Power & GND

芯片电源的入口要有 10uF 储能电容;如果电源走线较长,建议在电源走线中间也增加一个 10uF 储能电容。
芯片的每个电源 pin 要有 100nF 滤波电容;
选择低 ESR 的 bead，降低 IR drop 的影响;
EPAD 需要通过尽量多的过孔与 GND 接触良好。

8.2 MDI

MDI 走线差分阻抗要求 100ohm ±10%;
从芯片到变压器的 MDI 走线长度不要超过 6 inch;
同一个 port，4 对 MDI 走线长度相差不要超过 800mil;
同一对 MDI，P 和 N 的走线长度相差不要超过 5mil;
每对 MDI 走线之间的距离要大于 30mil;
MDI 走线应尽量避免过孔和换层。

8.3 QSGMII

QSGMII 差分阻抗要求 100ohm ±10%;
QSGMII 走线上应尽量避免过孔，最多不能超过 2 个;
QSGMII 走线长度不要超过 10 inch;
差分对中 P 和 N 的走线长度相差不要超过 5 mil;
差分对与其他信号的距离要大于 30 mil。

8.4 SGMII

1、 SGMII 差分阻抗要求 100ohm ±10%;
2、 SGMII 走线上应尽量避免过孔，最多不能超过 2 个;
3、 SGMII 走线长度不要超过 10 inch;
4、 差分对中 P 和 N 的走线长度相差不要超过 5 mil。

8.5 EMC Design Guide

尽量将变压器靠近芯片、RJ45 靠近变压器，以缩短 MDI 走线长度，提升 EMC 性能。

在芯片和变压器之间的 MDI 走线上预留 0ohm 电阻串联。如需要升 surge 性能，可以将 0ohm 替换成 1~2ohm 电阻。

如需提升 ESD 性能，可以在芯片和变压器之间的 MDI 走线上预留 5pF 电容并联到 GND。