ETHERNET-

Interfaces Overview and Debug/Bring-Up Tips

For QorIQ Product Families – T-Series and LS-Series

Patrick B. Billings
DN Applications Support
19.Apr.2018

SECURE CONNECTIONS FOR A SMARTER WORLD

Introduction

- The content in this presentation is focused on the various Ethernet related interfaces supported by Digital Networking (DN) Group at NXP Semiconductors.

- The presentation is NOT a deep dive of the architecture but merely an overview of the implementations and Ethernet startup hurdles that may be encountered when designing with particular DN SoCs.

- The information shared in this presentation is intended for public consumption.

Agenda

- **Overview**
- Supported Parallel and Serial Data Interfaces
- PHY and PHYless Considerations
- Management Bus and Controller
- RCW and Signal Integrity Considerations
- Documentation : SoC and DPAAx Reference Manuals
- U-Boot command line check for link status
- Summary / Conclusion

Overview – Ethernet Data Controllers

The NXP QorIQ product families support the Ethernet protocol with a variety of controllers for the data path and the PCS/PHY management path.

Parallel data interfaces are MII, RMII, GMII, and RGMII.

Serial data interfaces are SGMII, OC-SGMII (Overclocked), QSGMII, XAUI, XFI, USXGMII, XLAUI, CAUI-1/2/4 (with some backplane implementations as well).

For the T-series, the main Ethernet controller is DPAA1-FMAN-mEMAC.

For the LS-series, the main Ethernet controllers are eTSEC 2.x, PPFE, DPAA1-FMAN-mEMAC, and DPAA2-WRIOP-mEMAC.

Overview – Ethernet Management Controllers

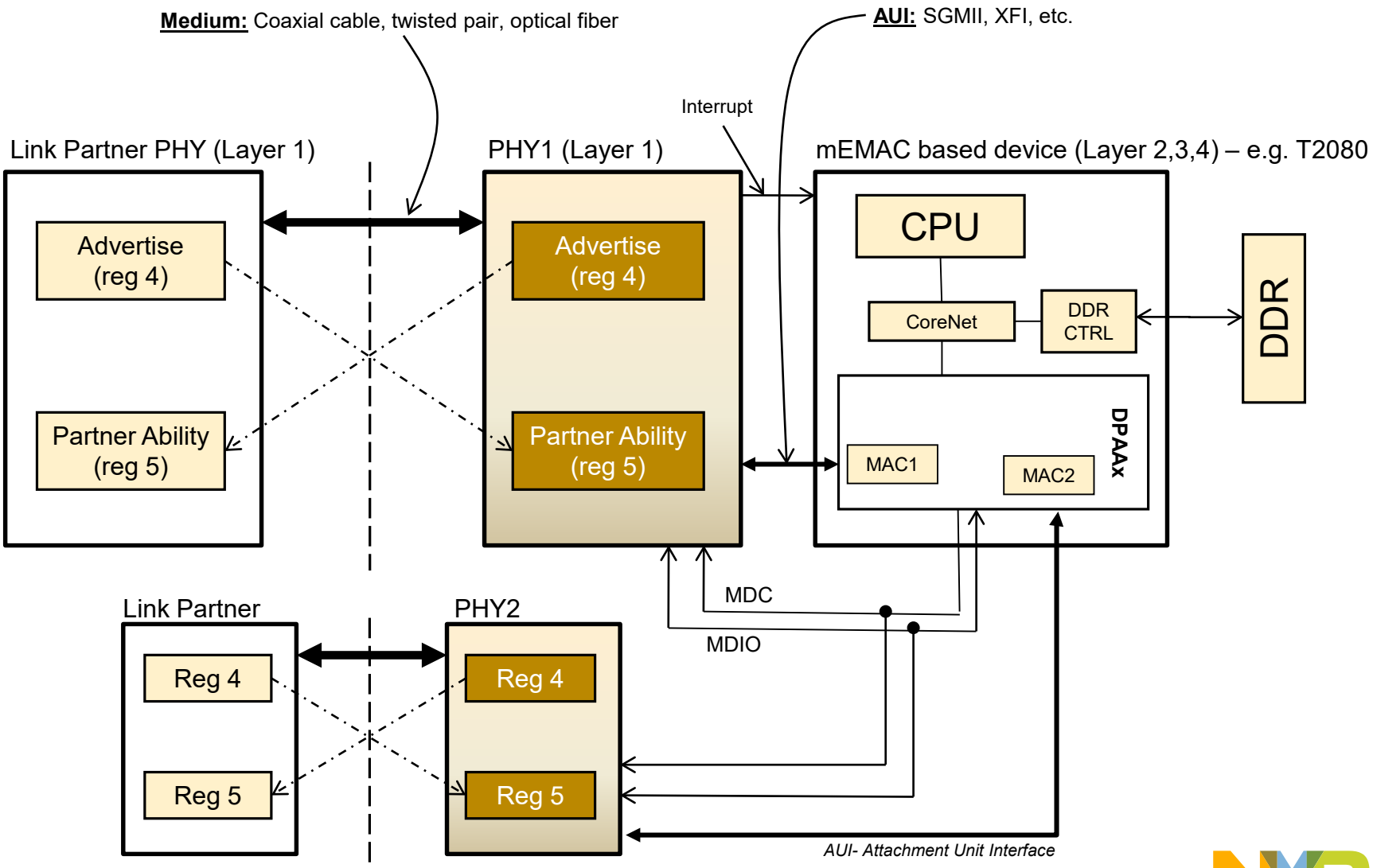
The Ethernet management controllers support either IEEE 802.3 Clause 22 or Clause 45 or both.

External PHY registers are accessed via a specified MAC or a dedicated controller (e.g. EMI1 or EMI2).

Internal TBI and PCS registers are accessed via registers in the associated MAC.

Clause 22 has been associated with 10/100/1000Mbps PHYs and Clause 45 with 10GBps PHYs but this is not a hard restriction. The Aquantia PHYs support 10G/5G/2.5G/1G/100M and use the Clause 45 MDIO register set. Our 4-Port Gigabit Ethernet Copper PHY [*F104S8A*] supports Clause 45 for the Energy Efficient Ethernet (EEE) registers but Clause 22 for all other registers.

Overview – Data Paths and Management Bus



FMAN MDIO Memory Map (e.g., T2080)

Table 5-16. FMan base address in SoC Memory map

Type	FMan base address in SoC
FMan_v3 in T2080	FMan: 0x40_0000

0xE_0000–0xE_07FF	2 KB	EMAC1 (2.5/1Gbps) (mEMAC[1])	6.4.2/65-5
0xE_0800–0xE_0FFF	2KB	MACsec on EMAC1	6.4.2/65-5
0xE_1000–0xE_1FFF	4 KB	MDIO-1 for EMAC1[2]	6.4.2/65-5
0xE_2000–0xE_27FF	2 KB	EMAC2 (10/2.5/1Gbps) (mEMAC)	6.4.2/65-5
0xE_2800–0xE_2FFF	2 KB	MACsec on EMAC2	6.4.2/65-5
0xE_3000–0xE_3FFF	4 KB	MDIO-2 for EMAC2	6.4.2/65-5

WRIOP MDIO Memory Map (e.g., LS2088)

- MAC1 mdio controller registers @ 8C0_7000
- MAC2 mdio controller registers @ 8C0_B000
- MAC3 mdio controller registers @ 8C0_F000
- MAC4 mdio controller registers @ 8C1_3000
- MAC5 mdio controller registers @ 8C1_7000
- MAC6 mdio controller registers @ 8C1_B000
- MAC7 mdio controller registers @ 8C1_F000
- MAC8 mdio controller registers @ 8C2_3000
- MAC9 mdio controller registers @ 8C2_7000
- MAC10 mdio controller registers @ 8C2_B000
- MAC11 mdio controller registers @ 8C2_F000
- MAC12 mdio controller registers @ 8C3_3000
- MAC13 mdio controller registers @ 8C3_7000
- MAC14 mdio controller registers @ 8C3_B000
- MAC15 mdio controller registers @ 8C3_F000
- MAC16 mdio controller registers @ 8C4_3000

For external MDIO, we use offsets for EMI1 and EMI2, but programming is the same.

- EMI1 mido registers @ 8B9_6000
- EMI2 mdio registers @ 8B9_7000

The formula for the specific DPAA2 MAC MDIO controller register offset is 8C0_0000 + (Mac Port #)*(0x4000) + 0x3000.

Agenda

- Overview
- **Supported Parallel and Serial Data Interfaces**
- PHY and PHYless Considerations
- Management Bus and Controller
- RCW and Signal Integrity Considerations
- Documentation : SoC and DPAAx Reference Manuals
- U-Boot command line check for link status
- Summary / Conclusion

NXP Supported Physical Interfaces (SoC dependent)

Parallel

- MII (10/100 Mbps)
  - 4 data pins, 2.5MHz/25MHz clk
- RMII (10/100 Mbps)
  - 2 data pins, 5MHz/50MHz clk
- GMII (10/100Mbps & 1Gbps)
  - 8 data pins, 125MHz clk
  - Source Synchronous
  - MII fallback for lower speeds
- RGMII (10/100Mbps & 1Gbps)
  - 4 data pins, 125MHz clk
  - Sample Rising & Falling Clk Edges for 1G; only rising edge for 10/100

Serial (differential signal pair)

- SGMII (10/100Mbps &1Gbps)
  - 1.25G SerDes lane, 8b/10b
- OC-SGMII (2.5Gbps)
  - 3.125G SerDes lane, 8b/10b
- QSGMII (4x 1Gbps)
  - 5G SerDes lane, 4 MACs
- XAUI (10Gbps)
  - Four 3.125G SerDes lanes, 8b/10b
- XFI (10Gbps)
  - 10.3125G SerDes lane, 64b/66b
- FUTURE
  - USXGMII (10.3125G SerDes Lane): auto-neg for 100M,1G,2.5G,5G,10G
  - CAUI-1/2/4 (25G SerDes Lane): 25G, 50G, 100G
  - XLAUI (x4 10.3125G SerDes lanes): 40G

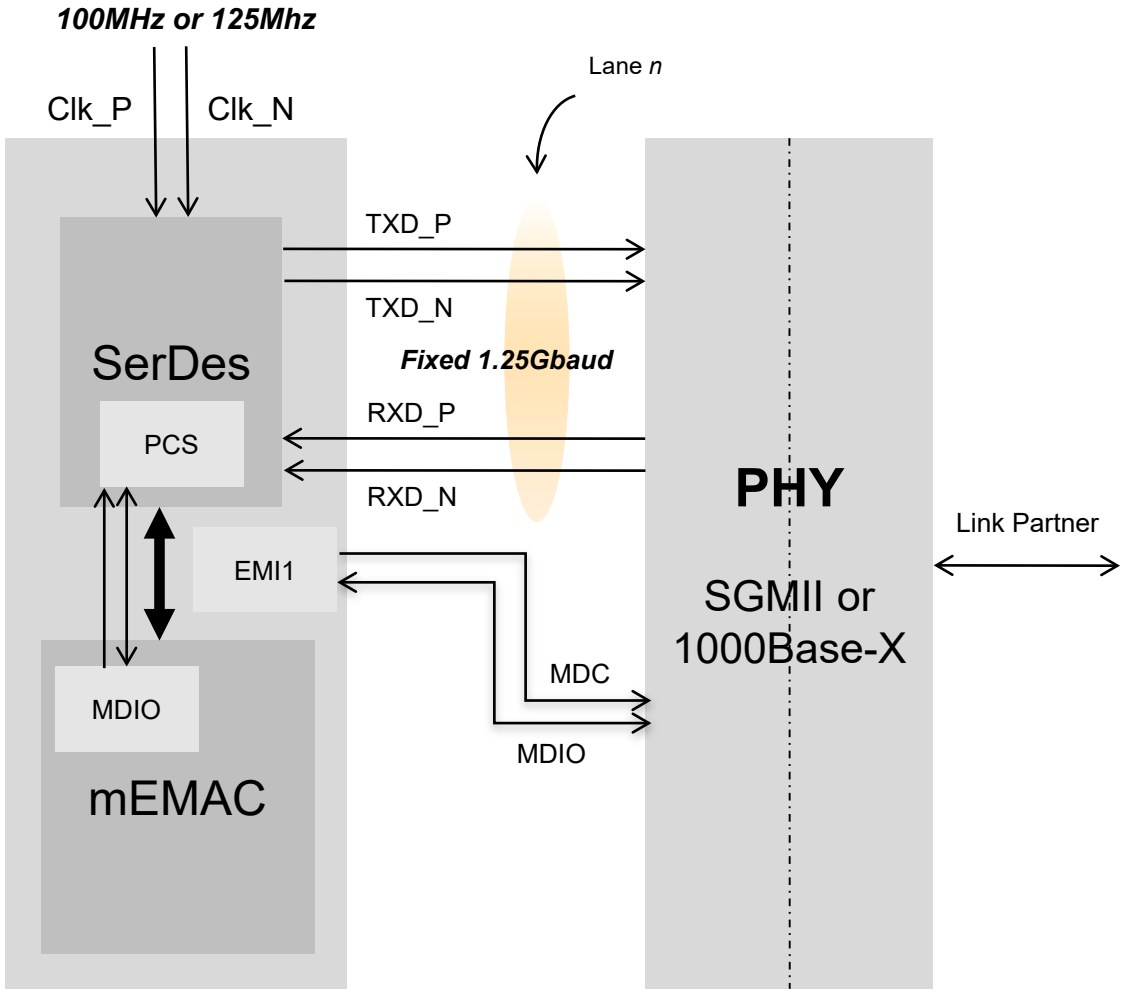
TIP: Some SoCs have in band link status/control for the RGMII interface

Agenda

- Overview
- Supported Parallel and Serial Data Interfaces
- **PHY and PHYless Considerations**
- Management Bus and Controller
- RCW and Signal Integrity Considerations
- Documentation : SoC and DPAAx Reference Manuals
- U-Boot command line check for link status
- Summary / Conclusion

mEMAC SGMII PHY Example

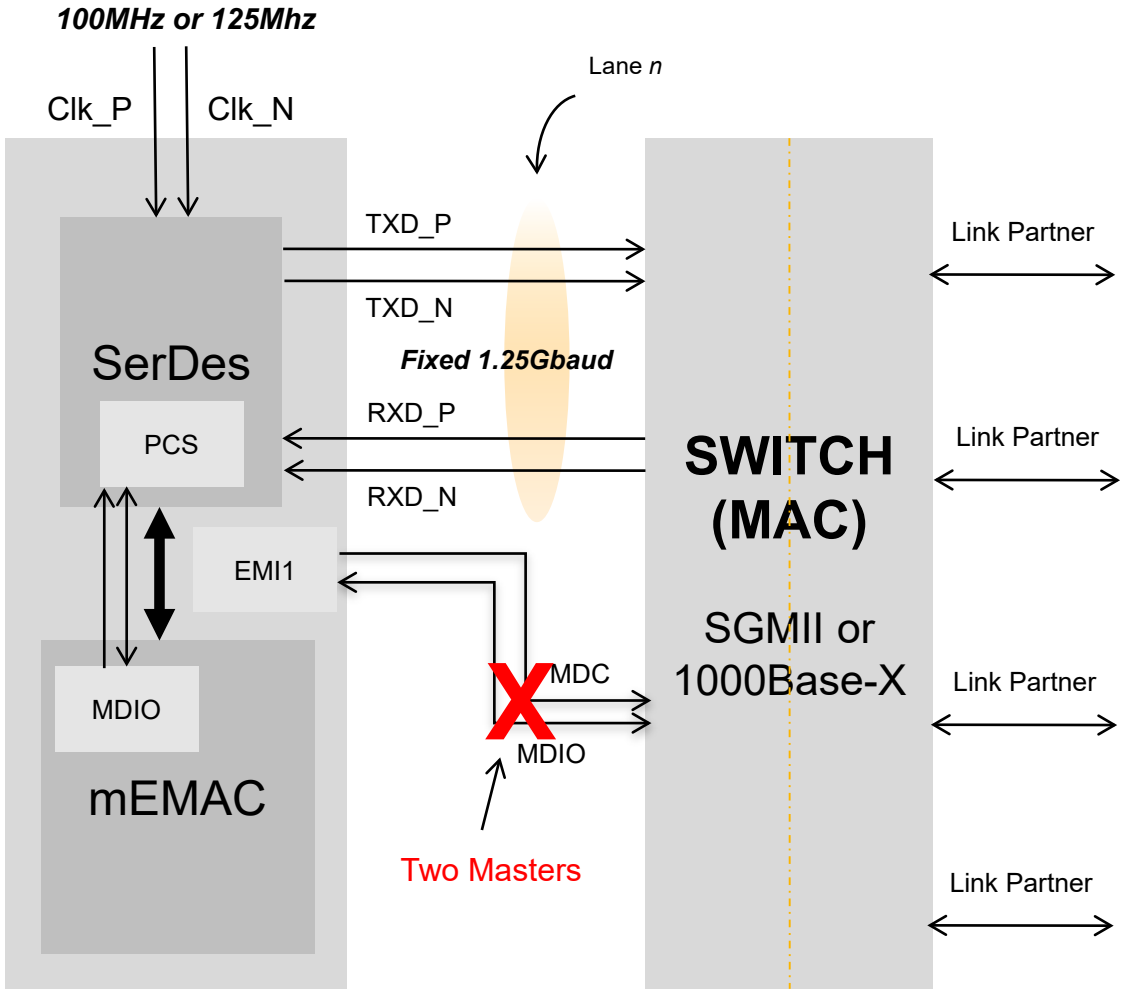
- SerDes block is used (differential signaling)
- 10/100/1000 Mbps supported with rate adaptation logic for SGMII
- 1000Base-X is only 1 Gbps
- SGMII PCS has Speed and Duplex settings and other settings
- Advertisement registers may have two formats: 1000Base-X or SGMII. Software determines which is used.
- Internal MDIO bus per MAC



TIP: Reception of valid symbols means SGMII_SR[Link_Stat] gets set

mEMAC SGMII PHYless (MAC-to-MAC) Example

- SerDes block is used (differential signaling)
- 10/100/1000 Mbps supported with rate adaptation logic for SGMII
- 1000Base-X is only 1 Gbps
- SGMII PCS has Speed and Duplex settings and other settings
- Advertisement registers may have two formats: 1000Base-X or SGMII. Software determines which is used.
- Internal MDIO bus per MAC
- May Lose MII management because no slave device. Configuration and status info can be placed in system memory for access or maybe SPI or I2C (depends on connected device)



TIP: Reception of valid symbols means SGMII_SR[Link_Stat] gets set

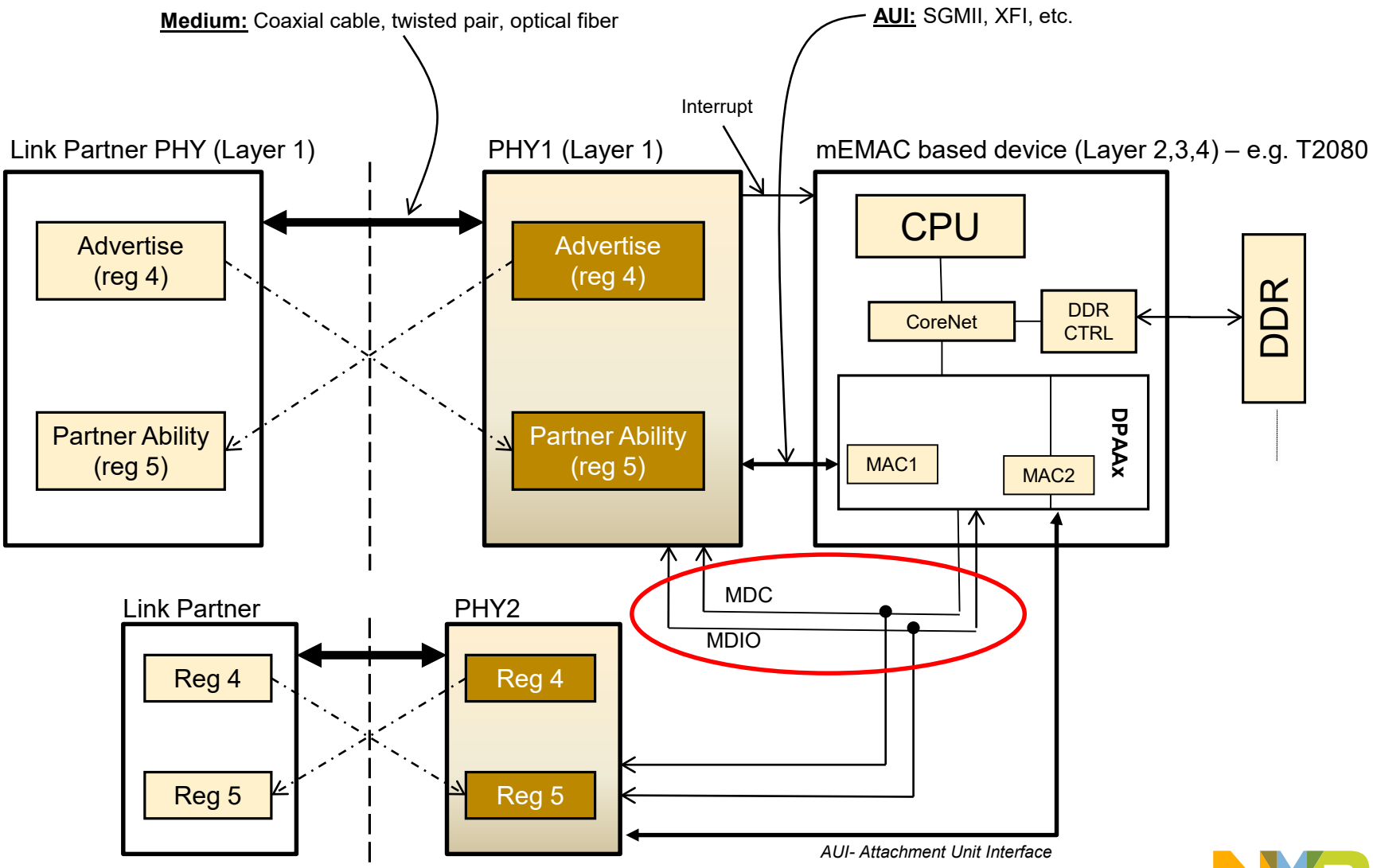
Additional notes

- XFI
  - SerDes Lane operates at 10.3125Gbaud
  - May need a re-timer in the path
  - Needs 156.25MHz SerDes Reference Clock
  - Internal MDIO uses Clause 45
  - Link Status in 0x3.0x20
- QSGMII
  - SerDes Lane operates at 5Gbaud
  - Time Division with four MACs
  - Internal MDIO uses Clause 22
  - MDIO registers by port number

Agenda

- Overview
- Supported Parallel and Serial Data Interfaces
- PHY and PHYless Considerations
- **Management Bus and Controller**
- RCW and Signal Integrity Considerations
- Documentation : SoC and DPAAx Reference Manuals
- U-Boot command line check for link status
- Summary / Conclusion

MII Management Bus (MDC & MDIO – two wire bus)



MII Management Controllers –
(*Internal* and External)

Clause 22 (*TBI* / *PCS* / EMI1)

- Typically 3.3v, 2.5v or 1.8v
- PHY addr & REG addr
- eTSEC
  - Internal: TBI per MAC
  - External: Normally 1st MAC pins are routed externally
- DPAA1-FMAN-dTSEC/mEMAC
  - Internal: TBI or PCS per MAC
  - External: Dedicated (not MAC dependent)
- DPAA2-WRIOP-mEMAC
  - Internal: PCS per MAC
  - External: Dedicated (not MAC dependent)

Clause 45 (*PCS* / EMI2)

- Typically 1.2v or 1.8v
- PORT addr, DEV addr, & REG addr
- DPAA1-FMAN-10GEC/mEMAC
  - Internal: PCS per MAC
  - External: Dedicated (not MAC dependent)
- DPAA2-WRIOP-mEMAC
  - Internal: PCS per MAC
  - External: Dedicated (not MAC dependent)

MII Management –

Supported Serial Interfaces with Internal MII Mgmt Registers (1 of 2)

- **SGMII TBI (*DPAA Reference Manual*)**
  - Clause 22
  - MAC: eTSEC/dTSEC
  - TBIPA defines the PHY address for Clause 22 accesses
  - MIIMCOM, MIIMADD, MIIMCON, MIIMSTAT used to perform read/writes
  - e.g., TBICON is at register address 0x11

- **SGMII PCS**
  - Clause 22; Clause 45 for 1000Base-KX mode (device address = 0x03)
  - MAC: mEMAC
  - SGMIInCR1[MDEV_PORT] defines the PHY address for Clause 22 accesses and the PORT address for Clause 45 accesses. (***SoC Reference Manual***)
  - MDIO_CTL & MDIO_DATA used to perform reads/writes (***DPAA Reference Manual***)
  - e.g., MDIO_SGMII_SR is at register address 0x1; MDIO_SGMII_SR is at register 0x8001 for 1000Base-KX

MII Management –

Supported Serial Interfaces with Internal MII Mgmt Registers (2 of 2)

- **QSGMII PCS**
  - Clause 22
  - QSGMIInCR1[MDEV_PORT] defines the internal PHY address (upper 3 bits of MDIO_CTL[PHY addr]) and the QSGMII port address (lower 2 bits of MDIO_CTL[PHY addr]) for Clause 22 accesses to each of the four QSGMII ports. (***SoC Reference Manual***)
  - MDIO_CTL & MDIO_DATA used to perform reads/writes (***DPAA Reference Manual***)
  - e.g., QSGMII_SR is at register address 0x1 (ports 0,1,2,3)

- **XFI PCS**
  - Clause 45 (including 10GBASE-KR)
  - XFInCR1[MDEV_PORT] defines the PORT address for Clause 45 accesses. (***SoC Reference Manual***)
  - MDIO_CTL & MDIO_DATA used to perform reads/writes (***DPAA Reference Manual***)
  - e.g., MDIO_XFI_PCS_SR1 is at register address 0x1 (device address = 0x3); MDIO_XFI_10GKR_PMD_SR is at register address 0x97 (device address =0x1)

Memory Mapped MDIO Registers – MDIO Configuration Register (MDIO_CFG)

Offset 0x030 Access: Mixed

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R	BSY	CMP	CIM	—					NEG[1]	EHOLD	—					
W		w1c														
Resets: Internal	All zeros															
External	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

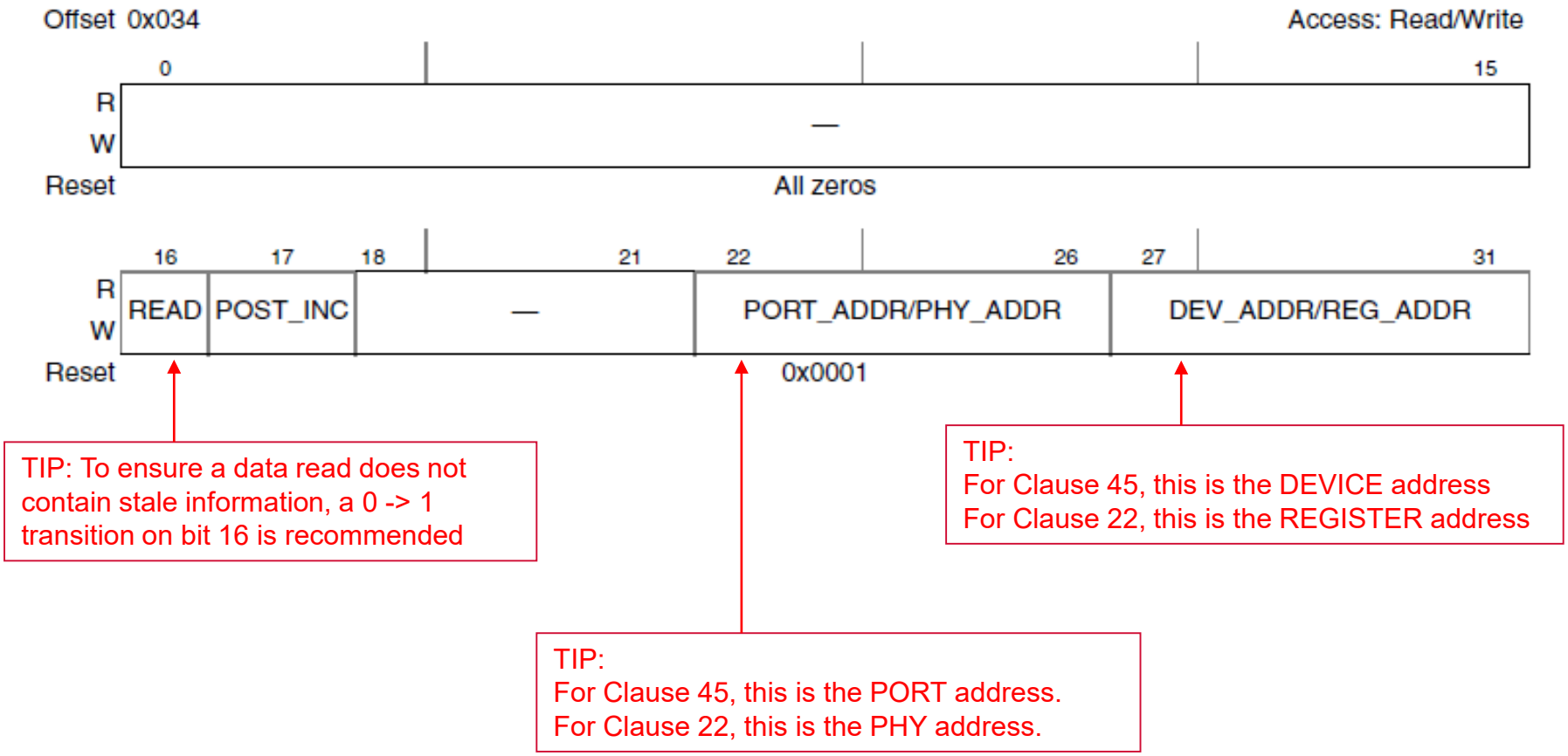
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
R	MDIO_CLK_DIV[2]									ENC45	PRE	MDIO_HOLD			MDIO_ RD_ER	—
W																
Resets: Internal	0	0	0	1	0	1	0	0	0	1	0	0	1	0	0	0
External	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0

Figure 6-93. MDIO_CFG Register Definition

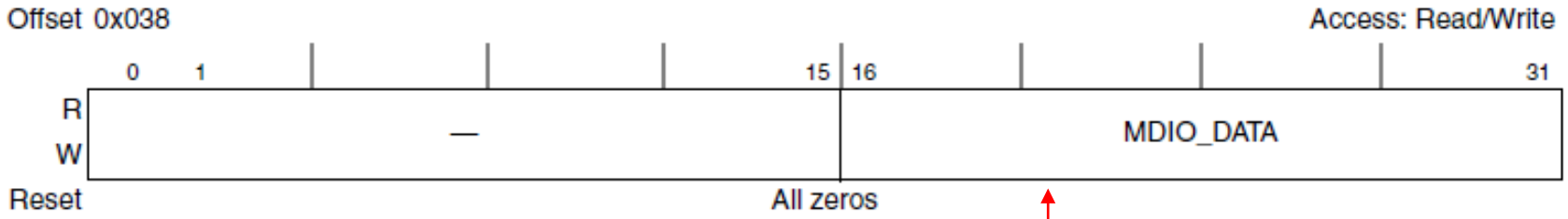
1 Reset value of NEG for external MDIOs is 1

2 Reset value of MDIO_CLK_DIV for external MDIOs is 9'b0

Memory Mapped MDIO Registers – MDIO Control Register (MDIO_CTL)

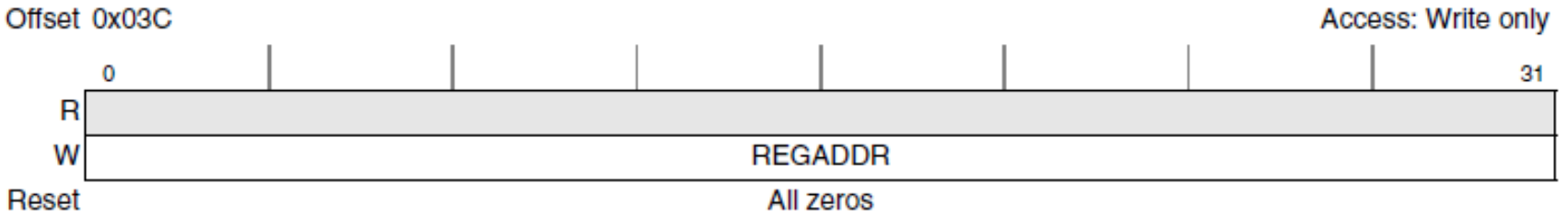


Memory Mapped MDIO Registers – MDIO Data Register (MDIO_DATA)



TIP:
For reads, this contains the results
For writes, the info to be programmed is sent

Memory Mapped MDIO Registers –
MDIO Register Address Register (MDIO_ADDR)

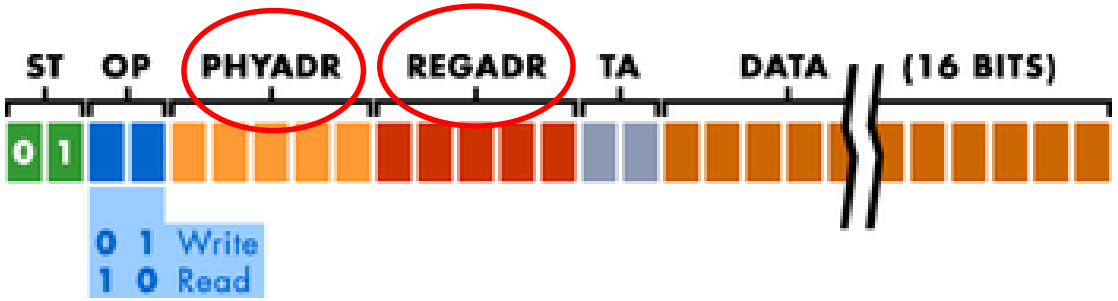


TIP: FOR CLAUSE 45 ONLY

MII Management - Clause 22

Clause 22 defines the MDIO communication basic frame format which is composed of the following elements:

The frame format only allows a 5-bit number for both the PHY address and the register address, which limits the number of MMDs that the STA can interface. Additionally, Clause 22 MDIO only supports 5V tolerant devices and does not have a low voltage option.



ST	2 bits	Start of Frame (01 for Clause 22)
OP	2 bits	OP Code
PHYADR	5 bits	PHY Address
REGADR	5 bits	Register Address
TA	2 bits	Turnaround time to change bus ownership from STA to MMD if required
DATA	16 bits	Data Driven by STA during write Driven by MMD during read

MII Management - Clause 22 Internal PCS Register Set

31.7.8.1 MDIO_SGMII Memory Map

Base address: 0h

Offset	Register	Width (In bits)	Access	Reset value
0h	SGMII Control (SGMII_CR)	16	RW	1140h
1h	SGMII Status (SGMII_SR)	16	RO	0009h
2h	SGMII PHY Identifier Upper (SGMII_PHY_ID_H)	16	RO	0083h
3h	SGMII PHY Identifier Lower (SGMII_PHY_ID_L)	16	RO	E400h
4h	SGMII Device Ability for 1000Base-X (SGMII_DEV_ABIL_1KBX)	16	RW	01A0h
4h	SGMII Device Ability for SGMII (SGMII_DEV_ABIL_SGMII)	16	RW	01A0h

Table continues on the next page...

MII Management – SGMII PCS Status (register 0x1) - *MDIO_SGMII_SR*

Address: 0h base + 1h offset = 1h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read	100BASET4	100BASEX_FD	100BASEX_HD	10MBPS_FD	10MBPS_HD	100BASET2_FD	100BASET2_HD	EXT_STAT	Reserved		AN_COMP	REM_FAULT	AN_ABIL	LINK_STAT	JAB_DET	EXT_CAP
Write																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1

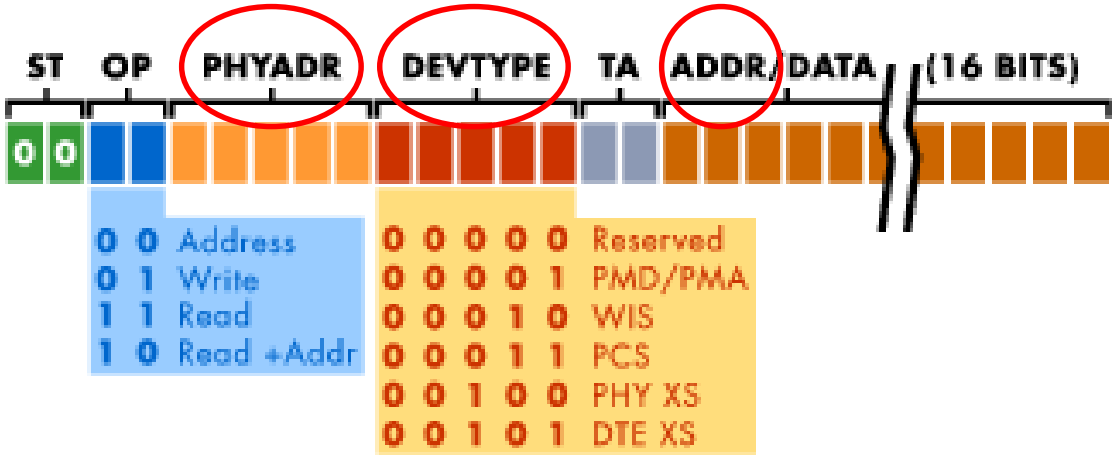
MII Management – SGMII PCS Interface Mode (register 0x14) - *MDIO_SGMII_IF_MODE*

Address: 0h base + 14h offset = 14h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read	Reserved											SGMII_DUPLEX	SGMII_SPEED		USE_SGMII_AN	SGMII_EN
Write																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MII Management - Clause 45

In order to address the deficiencies of Clause 22, Clause 45 was added to the 802.3 specification. Clause 45 added support for low voltage devices down to 1.2V and extended the frame format to provide access to many more devices and registers. Some of the elements of the extended frame are similar to the basic data frame:



ST	2 bits	Start of Frame (00 for Clause 45)
OP	2 bits	OP Code
PHYADR	5 bits	PHY Address
DEVTYPE	5 bits	Device Type
TA	2 bits	Turnaround time to change bus ownership from STA to MMD if required
ADDR/DATA	16 bits	Address or Data Driven by STA for address Driven by STA during write Driven by MMD during read Driven by MMD during read-increment-address

MII Management - Clause 45 Internal PCS Register Set

19.5.2 XFI PCS MDIO Memory Map/Register Definition

The XFI PCS register space is selected when the associated XFInCR1[MDEV_PORT] matches the Ethernet MAC port address (MDIO_CTL[PORT_ADDR]) and the device address (MDIO_CTL[DEV_ADDR]) is 03h.

MDIO memory map

Offset address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
0	XFI PCS Control 1 (MDIO_XFI_PCS_CR1)	16	R/W	2000h	19.5.2.1/ 1323
1	XFI PCS Status 1 (MDIO_XFI_PCS_SR1)	16	R	0002h	19.5.2.2/ 1324
2	XFI PCS Device Identifier Upper (MDIO_XFI_PCS_DEV_ID_H)	16	R	0083h	19.5.2.3/ 1326
3	XFI PCS Device Identifier Lower (MDIO_XFI_PCS_DEV_ID_L)	16	R	E400h	19.5.2.4/ 1326
4	XFI PCS Speed Ability (MDIO_XFI_PCS_SPEED_ABIL)	16	R	0001h	19.5.2.5/ 1327
5	XFI PCS Devices In Package 0 (MDIO_XFI_PCS_DEV_PRES0)	16	R	008Ah	19.5.2.6/ 1328
6	XFI PCS Devices in Package 1 (MDIO_XFI_PCS_DEV_PRES1)	16	R	0000h	19.5.2.7/ 1329
7	XFI 10G PCS Control 2 (MDIO_XFI_PCS_CR2)	16	R	000Bh	19.5.2.8/ 1330
8	XFI 10G PCS Status 2 (MDIO_XFI_PCS_SR2)	16	R	8001h	19.5.2.9/ 1330
E	XFI PCS Package Identifier Upper (MDIO_XFI_PCS_PKG_ID_H)	16	R	0083h	19.5.2.10/ 1331

MII Management – PCS link Status (register 0x3.0x20)
XFI 10GBASE-R PCS Status 1 (MDIO_XFI_PCS_10GR_SR1)

Address: 0h base + 20h offset = 20h

Bit	15	14	13	12	11	10	9	8
Read	Reserved			RX_LNK_STAT	Reserved			
Write								
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read	Reserved				PRBS9_ABIL	PRBS31_ABIL	PCS_HI_BER	PCS_BLK_LK
Write								
Reset	0	0	0	0	0	0	0	0

TIP: Register 0x3.0x1 also has link status information

Agenda

- Overview
- Supported Parallel and Serial Data Interfaces
- PHY and PHYless Considerations
- Management Bus and Controller
- **RCW and Signal Integrity Considerations**
- Documentation : SoC and DPAAx Reference Manuals
- U-Boot command line check for link status
- Summary / Conclusion

T2080 Example: Reset Config Word (RCW) - serial

Table 19-1. SerDes Lanes Assignments and Multiplexing (continued)

SERDES1										SERDES2									
SRDS_PRTCL_S1	A	B	C	D	E	F	G	H	Per Lane PLL Mapping	SRDS_PRTCL_S2	A	B	C	D	E	F	G	H	Per Lane PLL Mapping
									2222										2222
5F	Higig9				PCIe4	SG4	SG5	SG6	1111 2222	1F[1]	*PCIe1*				PCIe2				1111 2222
65	*Higig9*				PCIe4	SG4	SG5	SG6	1111 2222	1F[1]	*PCIe1*				PCIe2				1111 2222
6B	XFI9	XFI10	XFI1	XFI2	PCIe4	SG4	SG5	SG6	1111 2222	1F[1]	*PCIe1*				PCIe2				1111 2222
6C	XFI9	XFI10	SG1	SG2	PCIe4				1122 2222	16	PCIe1				PCIe2		SATA1	SATA2	1111 1122
6D	XFI9	XFI10	SG1	SG2	PCIe4				1122 2222	1F[1]	*PCIe1*				PCIe2				1111 2222

T2080 Example: Reset Config Word (RCW) - parallel

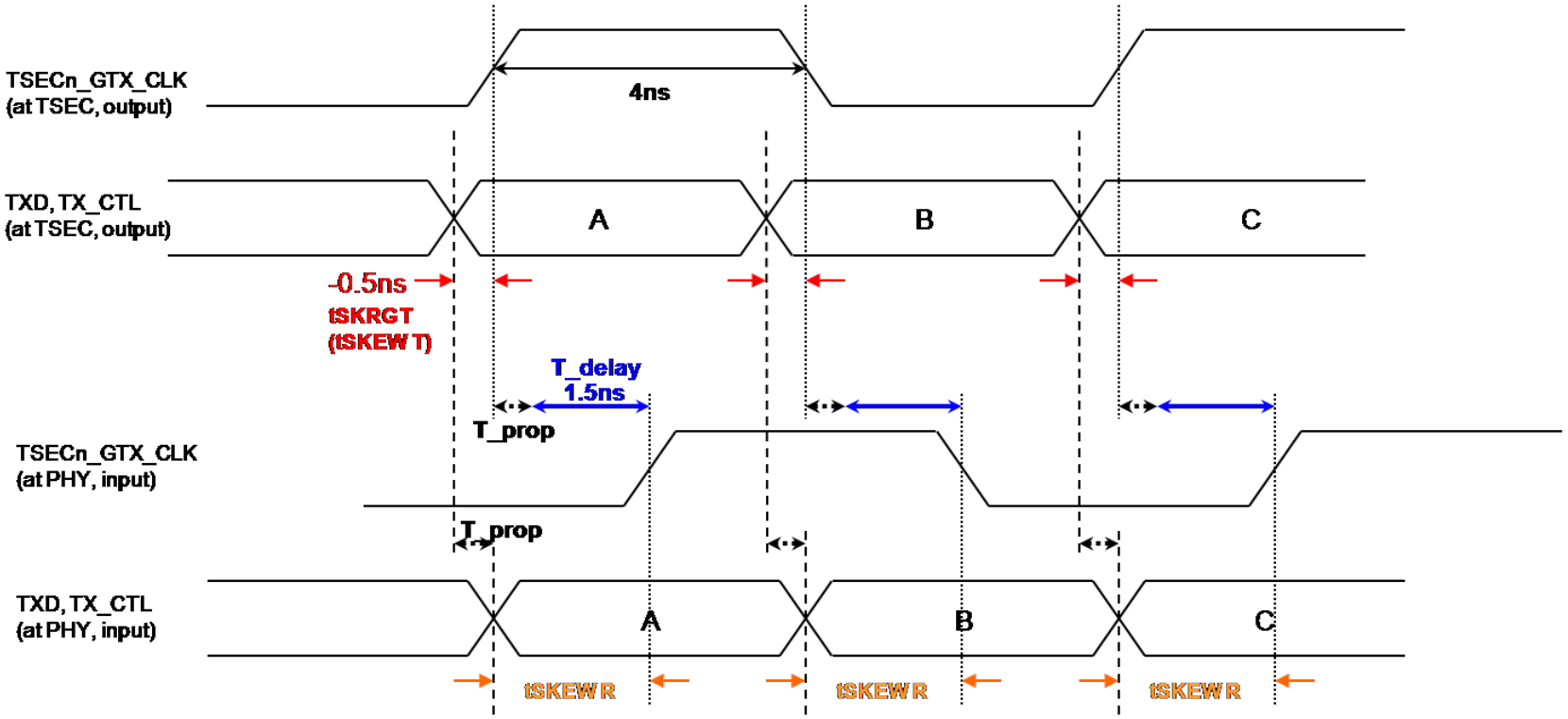
3.4.10 Ethernet controller 1 and GPIO3 signal multiplexing

The Ethernet controller 1 (EC1) RGMII interface shares signals with GPIO3. The functionality of these signals is determined by the EC1 field in the reset configuration word (RCW[EC1]). When RGMII is enabled for a particular MAC, it always takes precendence over SerDes-based interfaces.

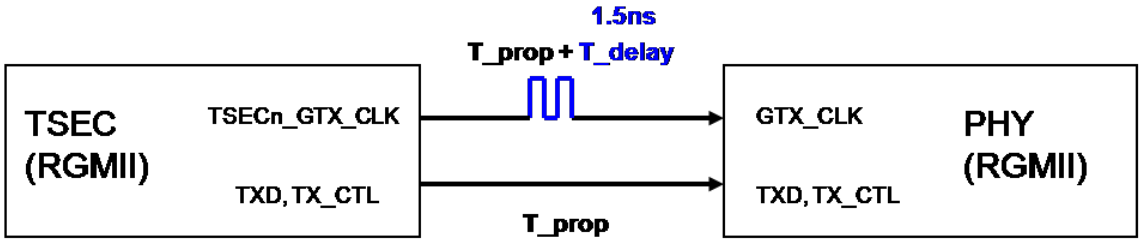
Table 3-12. EC1 RGMII signal configuration

Signal name	Signal function	RCW[EC1]
EC1_TXD[3:0]	EC1_TXD[3:0]	00
	GPIO3[11:14]	10
EC1_GTX_CLK	EC1_GTX_CLK (FM_MAC3 RGMII)	00
	GPIO3[16]	10
EC1_GTX_CLK125	EC1_GTX_CLK125 (FM_MAC3 RGMII)	00
	GPIO3[17]	10
EC1_RX_CLK	EC1_RX_CLK	00
	GPIO3[23]	10
EC1_RXD[3:0]	EC1_RXD[3:0]	00
	GPIO3[18:21]	10
EC1_TX_CTL	EC1_TX_CTL (FMan-MAC3 RGMII)	00
	GPIO3[15]	10
EC1_RX_CTL	EC1_RX_CTL (FMan-MAC3 RGMII)	00
	GPIO3[22]	10

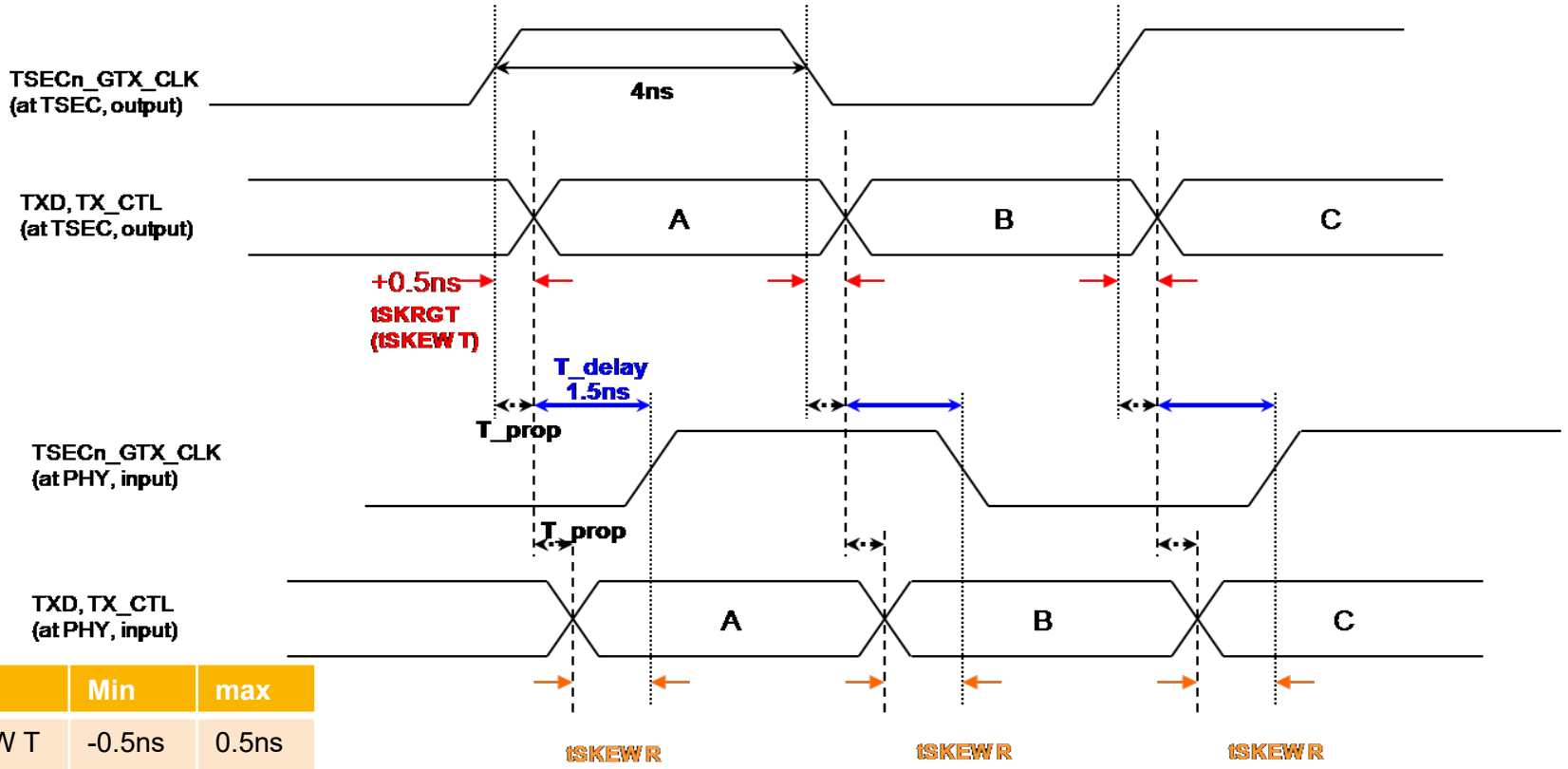
RGMII - Tx Data leads the GTX_CLK (-0.5ns)



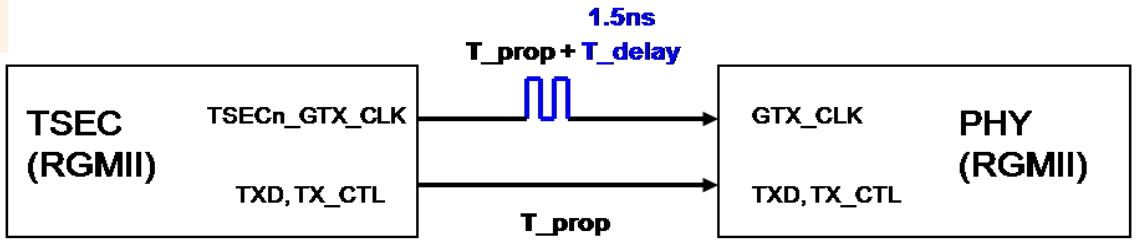
Spec	Min	max
tSKEW T	-0.5ns	0.5ns
tSKEW R	1ns	2.6ns



RGMII - Tx Data lags GTX_CLK (+0.5ns)

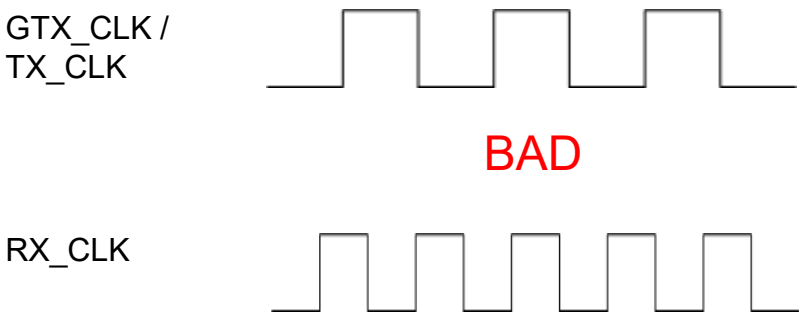


Spec	Min	max
tSKEW T	-0.5ns	0.5ns
tSKEW R	1ns	2.6ns

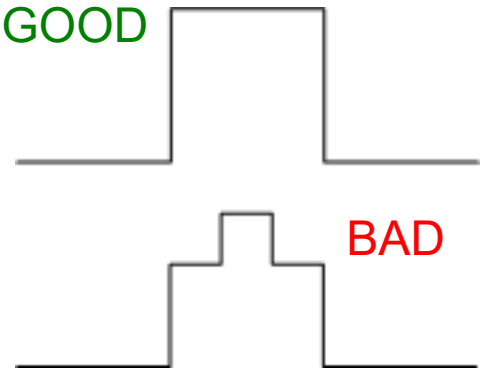


Parallel Interface Clock Integrity

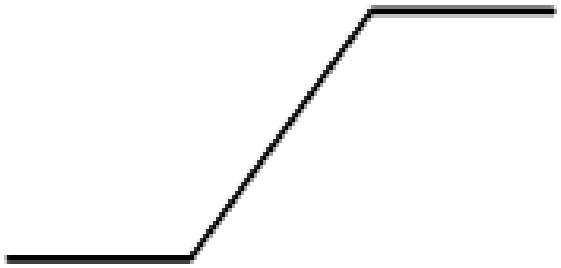
- FREQ Greater than +/- 300 PPM



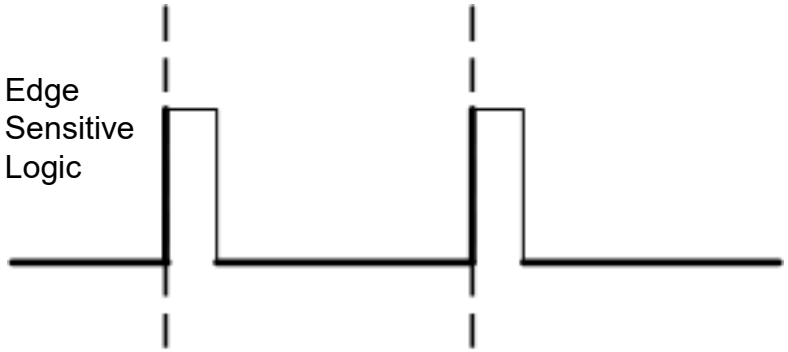
- Monotonic Edges



- Rise/Fall Meets Spec



- Really Duty Cycle Agnostic



Parallel Interface Clock Integrity (cont.)

- When both Rx &TX are used, a frequency requirement of +/- 300ppm between Rx and TX clock sources must be met.
- For example…
  - At 125MHz and 300ppm… this allows for an error of 37.5 KHz (125*300) between the clocks.
  - To ensure we have margin… let's assume we go with 125MHz and 200ppm (tighter accuracy). This allows for an error of 25KHz between the clocks (125*200)
  - Oscillator 1 – 25 MHz +/- 25 ppm => +/- 625 Hz error from the oscillator. GTX clock would have error magnified by PHY PLL (x5)… giving +/-3125 Hz error
  - Oscillator 2 – 25 MHz +/- 25 ppm => +/- 625 Hz error from the oscillator. RX_CLK2 would have error magnified by PHY PLL (x5)… giving +/-3125 Hz error
  - Worst case frequency error of the two clocks would be 3125+3125=6250 Hz worth of error. This scenario would meet the ppm requirement.

RGMII In-Band Link Status/Control Registers – Interface Mode Register (IF_MODE)

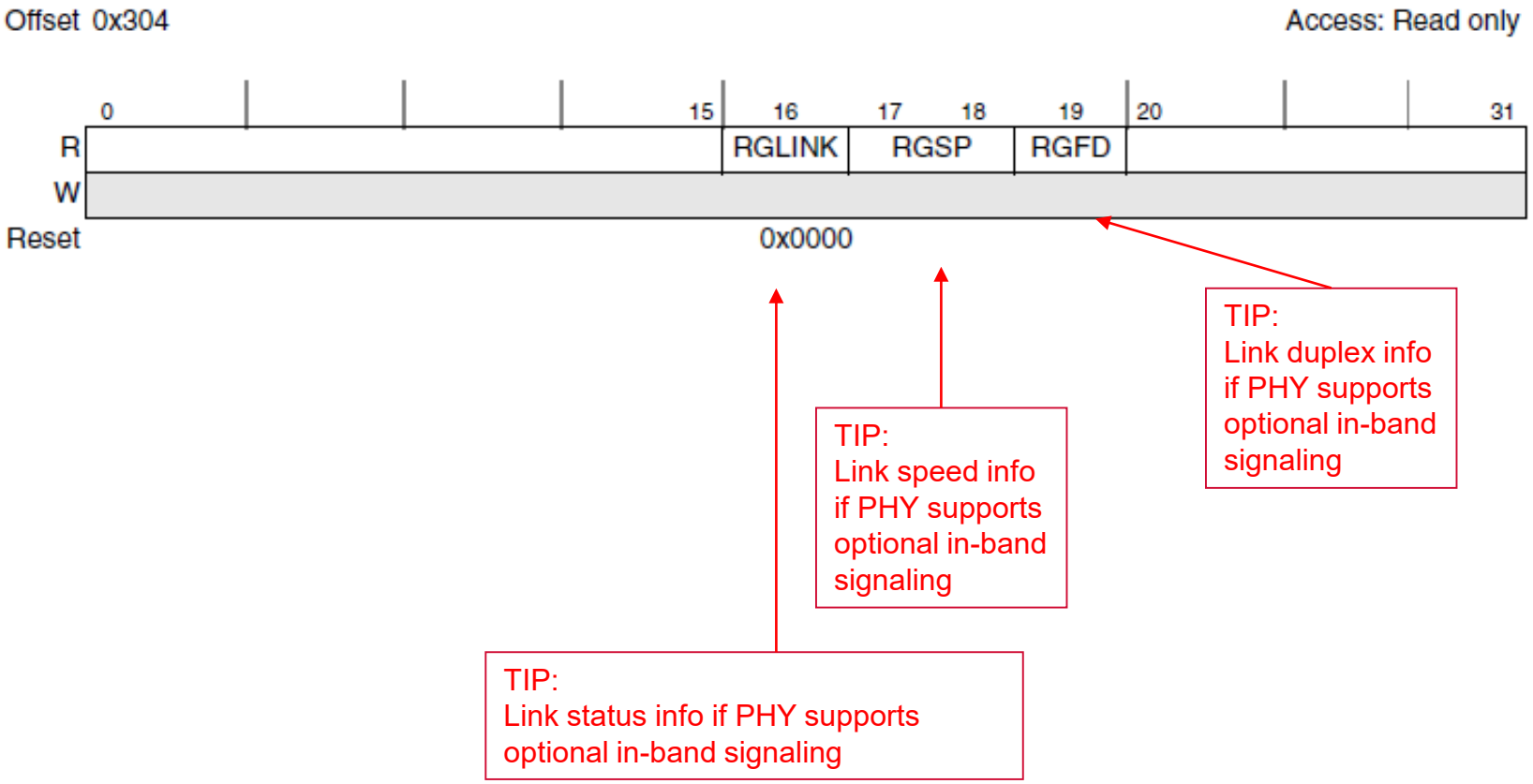
Offset 0x300 Access: Read/Write

	0	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
R W	—		ENA	SSP		SFD	—		MSG	HG	—	HD	RLP	—		RG	IFMODE	
Reset	0x0000		0	0	0	1	0	0	0	0	0	0	0	0	0	0	x	x

TIP:
Speed setting when not automatically determined by ENA=1

TIP:
Must be 0b10 for RGMII mode

RGMII In-Band Link Status/Control Registers – Interface Status Register (IF_STATUS)



Serial Interface Recommendations

Table 23. SerDes 1 pin termination checklist (continued)

Signal Name	I/O type	Used	Not Used	Completed
SD1_PLL1_TPA	O	Do not connect. These pins should be left floating.		
SD1_PLL1_TPD	O	Do not connect. These pins should be left floating.		
SD1_PLL2_TPA	O	Do not connect. These pins should be left floating.		
SD1_PLL2_TPD	O	Do not connect. These pins should be left floating.		
SD1_REF_CLK[1:2]_P	I	Ensure clocks are driven correctly.	If the SerDes *n* lanes are not used in the system, connect to S*n*GND, where *n* corresponds to the unused SerDes lanes.	
SD1_REF_CLK[1:2]_N	I	Ensure clocks are driven correctly.	If the SerDes *n* lanes are not used in the system, connect to S*n*GND, where *n* corresponds to the unused SerDes lanes.	
SD1_RX[0:7]_P	I	Ensure these pins are terminated correctly.	If the SerDes interface is entirely or partly unused, the unused pins must be connected to S*n*GND.	
SD1_RX[0:7]_N	I	Ensure these pins are terminated correctly.	If the SerDes interface is entirely or partly unused, the unused pins must be connected to S*n*GND.	
SD1_TX[0:7]_P	O	Ensure these pins are terminated correctly.	If the SerDes interface is entirely or partly unused, the unused pins must be left unconnected.	
SD1_TX[0:7]_N	O	Ensure these pins are terminated correctly.	If the SerDes interface is entirely or partly unused, the unused pins must be left unconnected.	

Agenda

- Overview
- Supported Parallel and Serial Data Interfaces
- PHY and PHYless Considerations
- Management Bus and Controller
- RCW and Signal Integrity Considerations
- **Documentation : SoC and DPAAx Reference Manuals**
- U-Boot command line check for link status
- Summary / Conclusion

SoC and DPAAx Reference Manuals

SoC RM for PCS info

MDIO register spaces

MDIO memory map

Offset address (hex)	Register name	Width (in bits)	Access	Reset value	Section/ page
0	SGMII Control (MDIO_SGMII_CR)	16	R/W	1140h	19.5.8.1/ 1337
1	SGMII Status (MDIO_SGMII_SR)	16	R	0009h	19.5.8.2/ 1339
2	SGMII PHY Identifier Upper (MDIO_SGMII_PHY_ID_H)	16	R	0083h	19.5.8.3/ 1340
3	SGMII PHY Identifier Lower (MDIO_SGMII_PHY_ID_L)	16	R	E400h	19.5.8.4/ 1341

DPAA RM for MAC info

0xF_0000–0xF_07FF	2 KB	EMAC9(10/2.5/1Gbps) (mEMAC 9 with HiGig[4])	6.4.2/65-5
0xF_0800–0xF_0FFF	2 KB	MACsec on EMAC9	6.4.2/65-5
0xF_1000–0xF_1FFF	4 KB	MDIO9 for EMAC9	6.4.2/65-5
0xF_2000–0xF_27FF	2 KB	FMan_v3: EMAC10 (10/2.5/1Gbps) (mEMAC 10 with HiGig[4])	6.4.2/65-5
0xF_2800–0xF_2FFF	2 KB	MACsec on EMAC10	6.4.2/65-5
0xF_3000–0xF_3FFF	4 KB	FMan_v3: MDIO10 for EMAC10	6.4.2/65-5
0xF_4000–0xF_BFFF	48 KB	Reserved	
0xF_C000–0xF_CFFF	4 Kbytes	FMan_v3: Dedicated MDIO1[4]	
0xF_D000–0xF_DFFF	4 Kbytes	FMan_v3: Dedicated MDIO2	
0xF_E000–0xF_EFFF	4 KB	IEEE 1588	7.4.1/75-2
0xF_F000–0xF_FFFF	4 KB	Reserved	

Agenda

- Overview
- Supported Parallel and Serial Data Interfaces
- PHY and PHYless Considerations
- Management Bus and Controller
- RCW and Signal Integrity Considerations
- Documentation : SoC and DPAAx Reference Manuals
- **U-Boot command line check for link status**
- Summary / Conclusion

T2080 U-boot Link Status Example – SRDS1_PRTCL = 0x6C

MAC9 - Lane A - XFI PCS

```
=> #Display MDEV_PORT (SerDes1_XFIACR1)
=> md fe0ea6c4
fe0ea6c4: 00000000

=> #put Lane A in SerDes loopback
=> md fe0ea83c 1
fe0ea83c: 00000000
=> mw.l fe0ea83c 10000000
=> md fe0ea83c 1
fe0ea83c: 18000000

=> #check XFI PCS Link Status - register 0x3.0x20
=> #confirm clause 45
=> md fe4f1030 1
fe4f1030: 40001448
```

XFI PCS (cont.)

```
=> #mii mgmt. cycles
=> mw.l fe4f1034 0003
=> mw.l fe4f103c 20
=> mw.l fe4f1034 8003
=> mw.l fe4f1034 8003
=> #display result of two reads to 0x3.0x20
=> md fe4f1038 1
fe4f1038: 00001001
=> # perform ping to self if supported
=> setenv ethact FM1@TGEC1
=> ping $ipaddr
Using FM1@TGEC1 device
host 10.81.55.72 is alive
```

T2080 U-boot Link Status Example – SRDS1_PRTCL = 0x6C

MAC1 – Lane C - SGMII PCS

```
=> #Display MDEV_PORT (SerDes1_SGMIIHCR1)
=> md fe0ea674
fe0ea674: 00000008bf

=> #put Lane C in SerDes loopback
=> md fe0ea8bc 1
fe0ea8bc: 04000000
=> mw.l fe0ea8bc 10000000
=> md fe0ea8bc 1
fe0ea8bc: 18000000
=> #check SGMII PCS Link Status - register 0x1
=> #MDIO_SGMII_SR
=> #confirm clause 22
=> md fe4e1030 1
fe4e1030: 40001408
```

SGMII PCS (cont.)

```
=> #mii mgmt
=> mw.l fe4e1034 0001
=> mw.l fe4e1034 8001
=> mw.l fe4e1034 8001

=> #display result of two reads to 0x1
=> md fe4f1038 1
fe4e1038: 0000002d

=> # perform ping to self if supported
=> setenv ethact FM1@DTSEC1
=> ping $ipaddr
```

Examples: LS1043ARDB - Using SERDES1 Protocol: 5205 (0x1455) [internal]

- **QSGMII PCS-P0 = FM1.MAC1**
- QSGMII PCS-P1 = FM1.MAC2
- QSGMII PCS-P2 = FM1.MAC5
- QSGMII PCS-P3 = FM1.MAC6

Check PCS Link status (register 1)
FMAN offset = 0x0_01A0_0000
MDIO-1 offset in FMAN = 0xE_1000
(little endian)

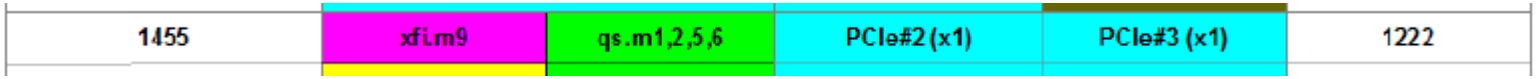
=> mm 01ae1034
01ae1034: 00000000 ? 01800000
01ae1038: 29000000 ? x
=> mm 01ae1034
01ae1034: 01000000 ? 01800000
*01ae1038: **2d**000000 ? x*

- **XFI PCS = FM1.MAC9**

Check PCS Link Status (register 0x3.0x20)
FMAN offset = 0x0_01A0_0000
MDIO-9 offset in FMAN = 0xF_1000
(little endian)

=> mm 01af103c
*01af103c: 40200000 ? **20**000000*
01af1040: 00000000 ? x
=> mm 01af1034
*01af1034: 03000000 ? **0380**0000*
*01af1038: 01**1**00000 ? x*

Examples: LS1043ARDB - Using SERDES1 Protocol: 5205 (0x1455)



- Check what PHYs are recognized by the external MDIO controllers from the u-boot prompt

⇒*mdio list*

FSL_MDIO0:

1 - RealTek RTL8211F <--> FM1@DTSEC3 **RGMII, MAC3*

2 - RealTek RTL8211F <--> FM1@DTSEC4

4 - Vitesse VSC8514 <--> FM1@DTSEC1 **QSGMII, Port 0, MAC1, lane B*

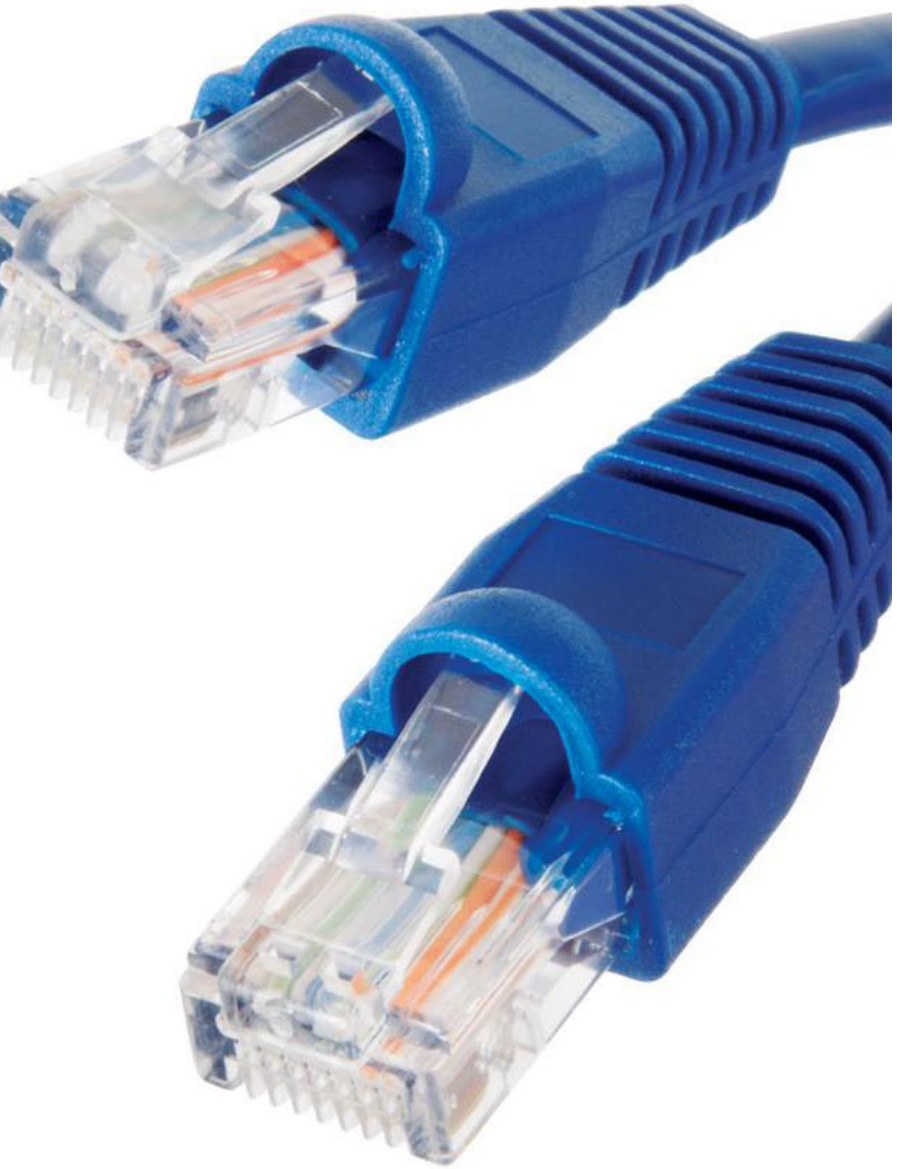
5 - Vitesse VSC8514 <--> FM1@DTSEC2

6 - Vitesse VSC8514 <--> FM1@DTSEC5

7 - Vitesse VSC8514 <--> FM1@DTSEC6

FM_TGEC_MDIO:

1 - Aquantia AQR105 <--> FM1@TGEC1 **AQRate PHY, XFI, MAC9, lane A *



Additional notes

- External PHY accesses
  - Use u-boot command ‘mdio list’ to show external interfaces
  - ‘mdio’ usage to check external PHY registers
    - Clause 45
      - mdio read FM1@TGEC1 0x3.0x20
    - Clause 22
      - mdio read FM1@DTSEC1 1

Examples: LS1043ARDB - Using SERDES1 Protocol: 5205 (0x1455) [external]

- FSL_MDIO0 equates to EMI1, Clause 22 bus-RGMII PHY

=> mdio read FM1@DTSEC3 1
Reading from bus FSL_MDIO0
PHY at address 1:
1 - 0x79ad

=> mdio read 1 1
1 is not a known ethernet
Reading from bus FSL_MDIO0
PHY at address 1:
1 - 0x79ad

- FM_TGEC_MDIO equates to EMI2, Clause 45 bus-AQ PHY

=> mdio read FM1@TGEC1 0x1e.0x20
Reading from bus FM_TGEC_MDIO
PHY at address 1:
30.32 - 0x200

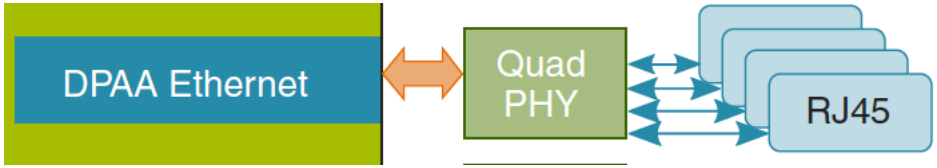
=> mdio read FM1@TGEC1 0x1e.0xc885
Reading from bus FM_TGEC_MDIO
PHY at address 1:
30.51333 - 0xb9

Examples: LS1043ARDB - Using SERDES1 Protocol: 5205 (0x1455) [external]

- FSL_MDIO0 equates to EMI1, Clause 22 bus-QSGMII PHY

=> mdio read FM1@DTSEC1 1
Reading from bus FSL_MDIO0
PHY at address 4:
1 - 0x796d

=> mdio read 4 1
4 is not a known ethernet
Reading from bus FSL_MDIO0
PHY at address 4:
1 - 0x796d



Agenda

- Overview
- Supported Parallel and Serial Data Interfaces
- PHY and PHYless Considerations
- Management Bus and Controller
- RCW and Signal Integrity Considerations
- Documentation : SoC and DPAAx Reference Manuals
- U-Boot command line check for link status
- **Summary / Conclusion**

Summary / Conclusion

- mEMAC based devices added link info registers for RGMII interface
  - e.g., See IF_MODE and IF_STATUS in T2080 DPAA RM
- Consult Datasheet/Bring-Up Guide/Design Checklist to confirm if pins are open-drain
- Be mindful of endianness between Power Arch and ARM based SoCs
- Consult Bring-Up Guide for unused pin termination recommendations
- Some connected devices (e.g., SFP modules) also support the I2C bus in addition to (or in lieu of) the MDIO bus
- Internal PCS Link Status should follow the live link (reception of valid symbols should set link status bit)
  - e.g., unplug cable or power down lane ***should*** clear link bit

SECURE CONNECTIONS
FOR A SMARTER WORLD

Terminology

- MAC – Media Access Controller
- PCS – Physical Coding Sublayer
- SFP – Small Form/Factor Pluggable
- PHY – Physical Transceiver
- mEMAC – Multi-rate Ethernet MAC
- eTSEC – Enhanced Three Speed Ethernet Controller
- dTSEC – DataPath Three Speed Ethernet Controller
- 10GEC - 10Gbps Ethernet Controller
- PPFE – Packet Processing Forwarding Engine