Specification of Ethernet MDIO

Version 0 Revision 1

Oct. 10, 2018 (July 7, 2018)

Future Design Systems, Inc.
www.future-ds.com / contact@future-ds.com

Abstract

This document describes specifications Ethernet MDIO (Management Data Input/output).

Table of Contents

1 General description

Figure 1: Ethernet MDIO block

2 IO ports and internal structure

2.1 IO ports

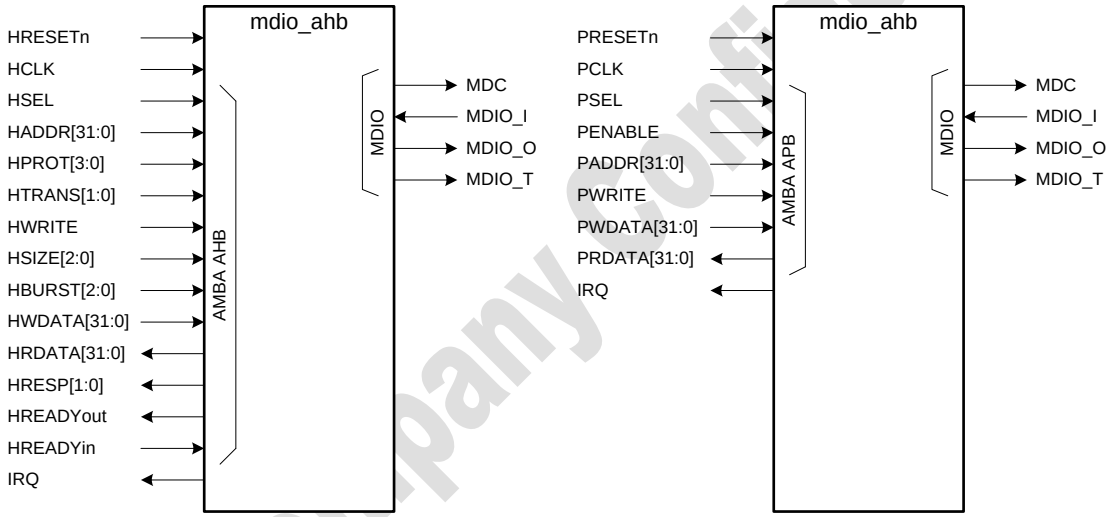

Figure 2: Ports

The table below summarizes the ports of MCB Gigabit Ethernet MAC.

Table 1: Ports

group	port	width	direction	description
common	HRESETn	1	Input	Asynchronous reset
	HCLK	1	Input	clock
	IRQ	1	Output	Interrupt
AHB slave port	HSEL	1	Input	
	HADDR	32	Input	
	HPROT	4	Input	
	HTRANS	2	Input	
	HWRITE	1	Input	
	HSIZE	3	Input	
	HBURST	3	Input	
	HWDATA	32	Input	

	HRDATA	32	Output	
	HRESP	2	Output	
	HREADYin	1	Input	
	HREADYout	1	Output	
Etc	MDC	1	Output	MDIO clock
	MDIO_I	1	Input	MDIO data input
	MDIO_O	1	Output	MDIO data output
	MDIO_T	1	Output	MDIO data output enable when low

2.2 Parameters and macros

2.2.1 Macros and parameters

The following parameters and macros are used in the design.

Table 2: Macros

Macro name	Default value	Meaning
MDIO_VERSION	32'h2018_1011	Defines the current version of RTL. It is used by API in order to check compatibility.

Table 3: Parameters

Macro name	Default value	Meaning
P_CLK_FREQ	50_000_000	Main clock (PCLK or HCLK) frequency to calculate
P_MDC_FREQ	2_000_000	Default frequency of MDC. ● Clock division number for MDC equation ● N = (Freq of CLK)/((Freq of MDC)*2)-1 ● Note that MDC can be 2.5Mhz at most.

3 Control and Status Registers

The address field indicates a relative address in hexadecimal. All register except CONTROL and STATUS should not be altered while CONTROL.TX_EN/RX_EN is '1'.

Table 4: Ethernet MDIO Controller CSR

Name	Address offset	Bit#		description
NAME0	+000h		RO	Name character 0 ~ 3
NAME1	+004h		RO	Name character 4 ~7
NAME2	+008h		RO	Name character 8 ~ 11
NAME3	+00Ch		RO	Name character 12 ~ 15
COMP0	+010h		RO	Company name 0 ~ 3

COMP1	+014h		RO	Company name 4 ~ 7
COMP2	+018h		RO	Company name 8 ~ 11
COMP3	+01Ch		RO	Company name 12 ~ 15
VERSION	+020h		RO	Version (0x2011_0707)
RESERVED	+024h			Reserved
	+028h			Reserved
	+02Ch			Reserved
MDIO_CONTROL	+030h		RW	(default: 0x0000_0031)
			31	EN_MDIO: enable when 1
			30:18	Reserved
			17	IE_MDIO_RD_DONE - Enables interrupt on completion of MDIO RD
			16	IE_MDIO_WR_DONE - Enables interrupt on completion of MDIO WR
			15:0	CLKDIV_MDIO: Clock division (MDC frequency) = (HCLK frequency)/(2x(value+1))
MDIO_STATUS	+034h		RW	(default: 0x0000_0000)
			31:18	reserved
			17	IP_MDIO_RD_DONE - It is set when MDIO RD completes and ‘IE_MDIO_RD_DONE is ‘1’. - It is cleared by writing ‘1’.
			16	IP_MDIO_WR_DONE - It is set when MDIO WR completes and ‘IE_MDIO_WR_DONE’ is ‘1’. - It is cleared by writing ‘1’.
			15:3	reserved
			2	Status when DONE is ‘1’. (read-only) - 0: OK - 1: Error
			1	Done when ‘1’ (read-only) It will be ‘0’ when ‘BUSY’ is ‘0’ and MDIO_WR/RD_CMD is written. And then it will be ‘1’ on completion.
			0	Busy state when ‘1’. (read-only) It will be ‘0’ on completion.
MDIO_WR_CMD	+038h		RW	Writing to this starts MDIO write. - It is recommended that check ‘MDIO_STATUS.busy’ is ‘0’. - Msb-bit is driven first at MDIO pin.
			31:28	2’b11 - fixed
			29:28	Start (2’b01) - fixed
			27:26	Opcode (2’b01) – fixed
			25:21	Physical address
			20:16	MII register address
			15:0	data to be written

MDIO_RD_CMD	+03Ch		RW	Writing to this starts MDIO read. - It is recommended that check 'MDIO_STATUS.busy' is '0'. - Msb-bit is driven first at MDIO pin.
			31:28	2'b11 – fixed
			29:28	Start (2'b01) – fixed
			27:26	Opcode (2'b10) -fixed
			25:21	Physical address
			20:16	MII register address
			15:0	Valid data when 'MDIO_STATUS.DONE' is '1'.
Reserved	+140h		RW	

4 MDIO interface

Each MDIO transaction should be followed by 32-bit preamble (all in '1') and consists of two parts, which are address phase and data phase. The address phase is divided further into ST (start), OP (opcode), PHYAD (PHY address), and REGAD (register address). Before data phase, 2-bit TA (turnaround) is required.

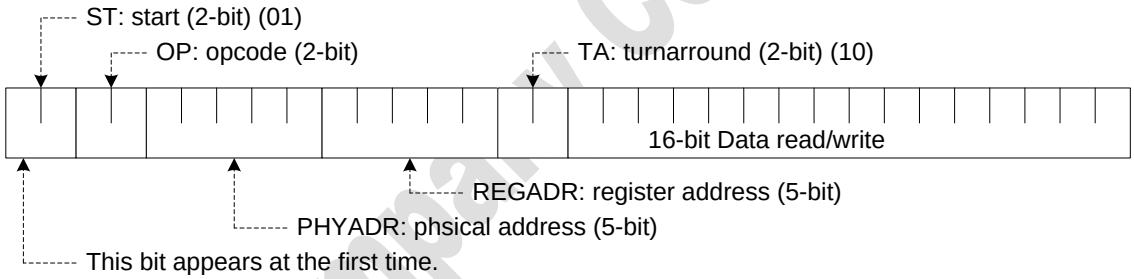

Figure 3: MDIO frame format

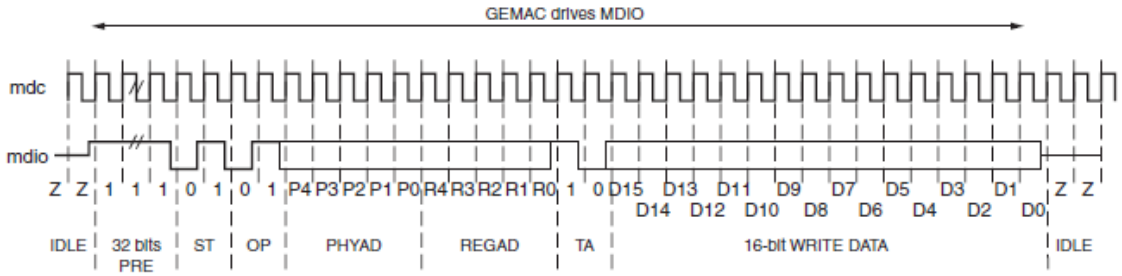

Figure 4: MDIO write timing diagram

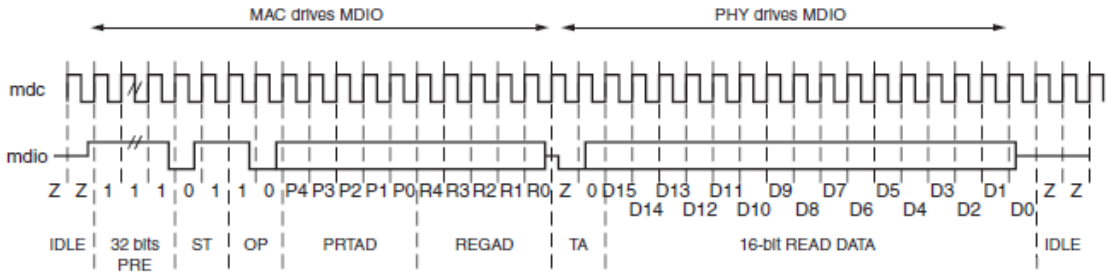

Figure 5: MDIO read timing diagram

Since 5-bit is allocated to PHYAD, up to 32 different MDIO devices can exist on the same MDIO bus. Since 5-bit is allocated to REGAD, there can be up to 32 registers in each MDIO device. The frequency of MDC can be up to 2.5Mhz[1], i.e., 400ns. The actual frequency is determined by the following equation.

f(MDC) = f(HCLK) / (2*(1+(Value of MDC_CONTROL.CLK_DIV)))

4.1 How to control MDIO through CSR

Write

MDIO_CONTROL.EN_MDIO <== 1
MDIO_STATUS.BUSY == 0
MDIO_WR_CMD <= {PHY_ADDR, REG_ADDR, DATA}
MDIO_STATUS_DONE == 1

Read

MDIO_CONTROL.EN_MDIO <== 1
MDIO_STATUS.BUSY == 0
MDIO_RD_CMD <= {PHY_ADDR, REG_ADDR}
MDIO_STATUS_DONE == 1
DATA <= MDIO_RD_CMD.DATA

5 Control API

It checks version and returns 0 when OK. Otherwise, returns 1.

It clears all CSR.

[1] It depends on real PHY chip.

6 Testing setup and scenario

7 Trouble shooting

8 Wish list

9 References

[1] ARM, AMBA Specification (Rev 2.0), ARM IHI 0011A, 1999.

[2] IEEE, IEEE Std. 802.3-2008, Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) access method and Physical Layer specifications, Section 2.

10 Revision history

- 2018.09.15: Document started by Ando Ki (adki@dynalith.com)