AR8031 FAQ

Rev. 1.01

Atheros Communication Inc.

Revision History

Date	Rev	Description
Oct 21, 2010	1.00	first release
Oct 22, 2010	1.01	Update Q&A 1 1.8v/1.5v power-on strapping change to register configurable

1. How to match RGMII interface logic voltage level for 1.5V, 1.8V and 2.5V?

Answer: AR8031 chip supports 3 voltage levels RGMII interface: 1.5V, 1.8V and 2.5V.

AR8031 chip internal power supply structure shows in Figure 1.

VDDH_REG(Pin10) output 2.5V. VDDIO_REG(Pin29) power on default output 1.5V.

If the interface logic level of RGMII is 1.5V, we can use VDDIO_REG(pin27) output as power supply for RGMII logic level. It is reminded that only VDDIO_REG pin is bounding as output pin and with 0.1uF decoupling capacitor. RGMII power input connects this pin on-chip internal.

If the interface logic level of RGMII is 1.8V, we need to write debug register 0x1F[3]=1'b1.

Debug register 0x1F definition

3	sel_1p5_lp8 _pos_reg	Mode	R/W	0=1.5v 1=1.8v
		HW Rst	0	
		SW Rst	Retain	

If the interface logic level of RGMII is 2.5V, we will connect VDDIO_REG(pin27) with VDDH_REG(pin8), then internal VDDIO_REG does not work because the input and out of this power supply module are all 2.5V. And at the same time RGMII power supply input change to 2.5V which is from pin29.

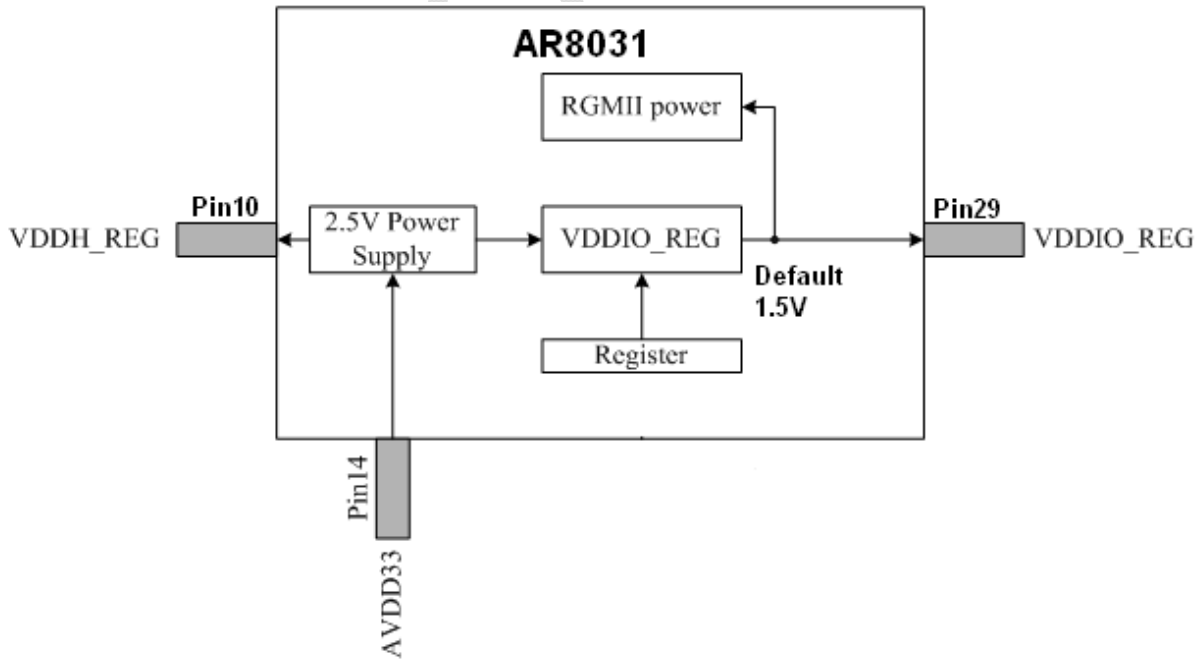


Figure 1. Power Supply Structure

2. Does AR8031 support RGMII 3.3V logic voltage level?

Answer: Yes. AR8031 RGMII interface Tx side can tolerant 3.3V logic signal input. The output signal logic level according to RGMII power supply VDDIO_REG. The detail description please refers to question **1**.

3. How to set external loopback?

Answer:

1. Plug in an external loopback cable (1-3/2-6/4-7/5-8).
2. Config Dbg0xb[15]=0 disable hibernate(power saving mode),
   Dbg0x11[0]=1 enable external loopback,
3. For 1000M Config: Preg0=0x8140 to set1000M and software reset
   For 100M Config Preg0=0xA100 to set100M and software reset
   For 10M Config Preg0=0x8100 to set10M and software reset

Notes:When re-plugs the cable in 1000M mode, need to write Preg0=0x8140 again to make the PHY link.

Need a software or hardware reset to make the PHY out of the loopback mode.

Dbg0xb means debug register whose offset address is 0xb.

Preg0 means basic register 0.

4. How to access the debug register?

Answer:

For example: Configure Dbg0xb[15]=0

1. Write register 0x1d=0xb (0x1d is debug port access address offset register)
2. Read register 0x1e==0xbc40 (data read from debug port 0xb; 0x1e is debug port access data register)
3. Write register 0x1e=0x3c40(set debug port 0xb bit[15]=0)

5. How to access the extended MMD register?

Answer:

AR8031 use MMD3 for PCS and MMD7 for Auto-Negotiation

Two different ways to access:

1. Use the normal SMI frame for read or write

	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

Register definition of AR8031

MMD access control register

Address Offset = 0x0d

Bits	Symbol	Type		Description
15:14	Function	Mode	R/W	00=address 01=data,no post increment 10=data,post increment on reads and writes 11=data,post increment on writes only;
		HW Rst	00	
		SW Rst	Retain	
13:5	Reserved	Mode	RO	
		HW Rst	0	
		SW Rst	0	
4:0	DEVAD	Mode	R/W	Device address
		HW Rst	0	
		SW Rst	Update	

MMD access address data register

Address Offset = 0xE

Bits	Symbol	Type		Description
15:0	Address data	Mode	R/W	If register13.15:14=00, MMD DEVAD's address register. Otherwise, MMD DEVAD's data register as indicated by the contents of its address register
		HW Rst	00	
		SW Rst	Retain	

For example: Write 0x8000 to Register 0 of MMD3

1. Write 0x3 to register 0xD: 0xD=0x0003;(function= address; set the device address)
2. Write 0x0 to register 0xE: 0xE=0x0; (set the register offset address)
3. Write 0x4003 to register 0xD:0xD=0x4003;(function=data; keep the device address)
4. Read register 0xE:0xE==(data from register 0x0 of MMD3)
5. Write 0x8000 to register 0xE :0xE=0x8000(write 0x8000 to register 0x0 of MMD3)

Notes: Read operation please refers to process 1-4

2. Use a special SMI access frame

	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

Notes: Set an address first, then write or read.

6. Does AR8031 have a way of internally adding delay for the RX_CLK or TX_CLK for RGMII interface?

Answer:

For RX_CLK delay:

AR8031 adds internal RX_CLK delay of typical 2ns(fufill min=1.2ns RGMII standard) in RGMII 1000M default mode.

Can be disabled by set Dbg0x0[15]=1'b0; the default value of this bit is 1'b1.

For TX_CLK delay:

AR8031 TX_CLK can add and control internal delay by register.

Can be enabled by set Dbg0x5[8]=1'b1, the default value of this bit is 1'b0.

The delay timing can be configured by register Dbg0xB[6:5] according to demand.

6:5	Gtx_dly_val	Mode	R/W	Select the delay of gtx_clk.
		HW Rst	2'b10	00:0.25ns 01:1.3ns

		SW Rst	Retain	10:2.4ns 11:3.4ns

7. How to make the LEDs work as normal?

Answer:

The LED_1000 and LED_ACT of AR8031 are power-on strapping pins, and also control the LED active status.

When the pin is externally pulled-up, the LED pin will strap the high state and active low.

When the pin is externally pulled-down, the LED pin will strap the low state and active high.

The LED_10_100 active status is also controlled by LED_1000.

So the LED_10_100 and LED_1000 external design should be the same.

Pin symbol	Power-on strapping function	Default status
LED_1000(MPW)	SEL_GPIO_INT Select pin5 work mode: 0:INT, 1:GPIO	With internal Pull-down
LED_1000(A0)	SEL_GPIO_INT Select pin5 work mode: 0:INT, 1:GPIO	With internal Pull-up
LED_10_100(MPW)	Reserved	With internal Pull-down
LED_10_100(A0)	Reserved	With internal Pull-up
LED_ACT (MPW)	Select the core voltage level: 0:1.1V, 1:1.2V (suggest external pull-down with a 10K resistor)	With internal Pull-down
LED_ACT (A0)	PHY address bit[2]	With internal Pull-up

Notes:

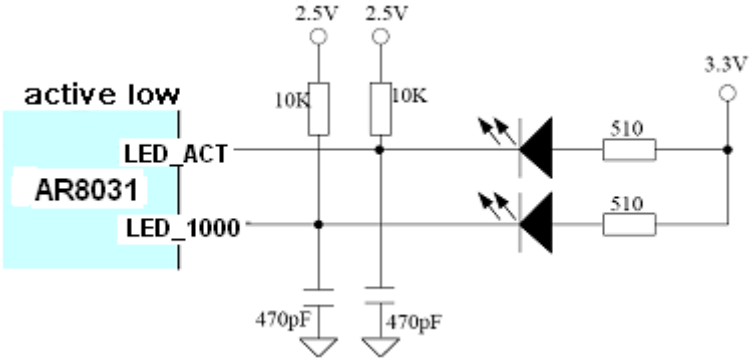
1. MPW edition is sample edition.
2. A0 edition is final mass production edition.
3. In MPW, LED_ACT, LED_10_100 and LED_1000 are all internal PD. In A0 they are all internal PU. If use an external 10k pull-up or pull-down, the internal weak pull-up/pull-down does not take any effect.

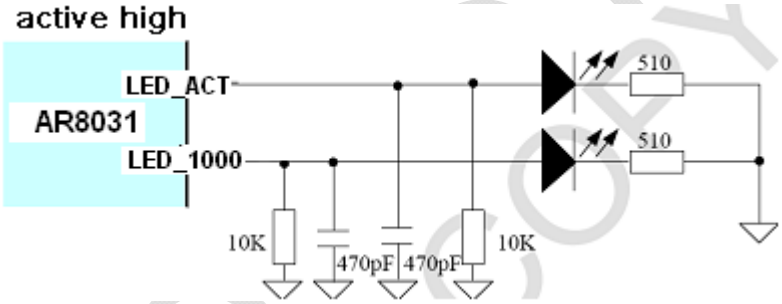
4. In A0 the core voltage is set to 1.1V internal, LED_ACT power-on strapping will be used as PHY address bit[2] which supports 8 AR8031 on the same SMI bus.

For design:

First select the pull-up or pull-down based on power-strapping function.

Then select the LED connect mode.





8. How does AR8031 set register procedure for 802.3 100/1000M template test?

1. 100Base-TX template

a. MDI mode:

(1) Set PHY MDI or MDIX mode: (Based on customer's layout)

=> Step1: PHY register setting offset 0x10=0x0800 (MDI mode)

(2) Set PHY 100Base-TX:

=> Step2: PHY register setting offset 0x00=0xA100

(3) Set power saving disable:

=> Step3: PHY register setting offset 0x1d=0x0029

Step4: PHY register setting offset 0x1e=0x36DC

Step5: PHY register setting offset 0x1d=0x000b

Step6: PHY register setting offset 0x1e=0x3C40

b. MDIX mode:

(1) Set PHY MDI or MDIX mode: (Based on customer's layout)

=> Step1: PHY register setting offset 0x10=0x0820 (MDI mode)

(2) Set PHY 100Base-TX:

=> Step2: PHY register setting offset 0x00=0xA100

(3) Set power saving disable:

=> Step3: PHY register setting offset 0x1d=0x0029

Step4: PHY register setting offset 0x1e=0x36DC

Step5: PHY register setting offset 0x1d=0x000b

Step6: PHY register setting offset 0x1e=0x3C40

2. 1000Base-TX template

a. Test mode 1

(1) Set debugging mode (Bit0) and Link ok (Bit3): (Dreg0x0b=0x0009)

=> Step1: PHY register setting offset 0x1d=0x000b

Step2: PHY register setting offset 0x1e=0x0009

(2) Set PHY 1000Base-T:

=> Step3: PHY register setting offset 0x00=0x8140

(3) Set test mode 1 to test (Template/Peak Volt/Droop)

=> Step4: PHY register setting offset 0x09=0x2200

b. Test mode 2

(1) Set debugging mode (Bit0) and Link ok (Bit3): (Dreg0x0b=0x0009)

=> Step1: PHY register setting offset 0x1d=0x000b

Step2: PHY register setting offset 0x1e=0x0009

(2) Set PHY 1000Base-T:

=> Step3: PHY register setting offset 0x00=0x8140

(3) Set test mode 2 to test (Jitter-Master)

=> Step4: PHY register setting offset 0x09=0x4200

c. Test mode 3

(1) Set debugging mode (Bit0) and Link ok (Bit3): (Dreg0x0b=0x0009)

=> Step1: PHY register setting offset 0x1d=0x000b

Step2: PHY register setting offset 0x1e=0x0009

(2) Set PHY 1000Base-T:

=> Step3: PHY register setting offset 0x00=0x8140

(3) Set test mode 3 to test (Jitter-Master)

=> Step4: PHY register setting offset 0x09=0x4200

d. Test mode 4

(1) Set debugging mode (Bit0) and Link ok (Bit3): (Dreg0x0b=0x0009)

=> Step1: PHY register setting offset 0x1d=0x000b

Step2: PHY register setting offset 0x1e=0x0009

(2) Set PHY 1000Base-T:

=> Step3: PHY register setting offset 0x00=0x8140

(3) Set test mode 4 to test (Distortion/CM Voltage)

=> Step4: PHY register setting offset 0x09=0x8200