# Pemcu Eye Monitor Firmware介绍

Author :Jennychen@zhaoxin.com

Data: 2018/05/15

## 1 PEMCU Eye Monitor 简介

引入Eye Monitor 目的是在于监测每条Lane 上信号质量的好坏，之前介绍过GEN3 EQ, EQ过程中并不能根据接收到的 data判断当前Rx接收信号的误码率（BER），CHX002中希望利用Eye monitor判定Rx接收信号的眼图，从而确定所连接的设备Tx数据的好坏，Eye monitor 本身可以用于GEN1/2/3.

## 2 Eye Monitor 基本原理

计算机生成了可选文字:
基 本 原 理 
VR 
提 供 矩 形 的 两 个 ， 百 
硬 件 对 这 两 卢 之 可 构 砹 
的 形 区 中 包 會 的 所 
有 行 scan 
VRE 
Eye Width 
硬 件 可 测 量 最 大 范 围 
对 于 每 一 个 po 
卜 W 在 指 定 时 间 内 ， 对 此 int 的 
enor countitiiYYÆ ， 
*ßRBER THRESHHOLD 」 3 以 
则 判 定 LEp0'ntÉFall ， 即 标 记 比 
VOInt 为 0 ； 
反 之 ， 则 判 定 此 p 酾 -It 为 P s 
标 记 比 位 
PH 
Height 

### 2.1 Eye monitor implementation process

#### 2.1.1 Mode 1 By Port

Mode 1是汇总所有Lane的scan结果合成一个眼图，即所有Lane 上同一点的BER counter加和后再与threshold比较。

For multiple lane port, this mode set ALL\_PORT to 1, all Lane of the point must be monitor

**SW**

**Step 1:** SW set registers R\_EYE\_MANU\_MUX(Rx2907h)to 0 and SW set EYESCAN\_DONE\_Penn(Rx2901h) to 1 clear this register

**Step 2:** SW set each of the two points at a time, { MCU\_PH\_OS[6:0](Rx2903h), MCU\_EQTNVTH[5:0](Rx2904h)}, { MCU\_PH\_OS\_2[6:0](Rx2905h)，MCU\_EQTNVTH\_2[5:0] (Rx2906h)}, eye scan for each point of the rectangular area covered by the two points

**Step 3:**SW set the register ALL\_PORT (Rx2901h) to 1, Indicates that all lane of the current port needs to be checked

**Step 4:** SW set register EQ\_PORT\_NAME[8:0] (Rx 2901h)、RBER\_THRESHHOLD\_[3:0] (Rx2901h) and MCU\_EYESCANTM\_PEnn[7:0](Rx2900h)

**Step 5:** SW set register MCU\_EYESCANEN\_PEnn(Rx2901h) to 1

**Step 6:** polling this register EYESCAN\_DONE\_PEnn (Rx2901h)

Step 7: When EYESCAN\_DONE\_PEnn is writed to 1, read the BER\_VAL[2047:0] (from Rx2A00h to Rx2AFFh)value which is the EP ERROR count.

**HW**

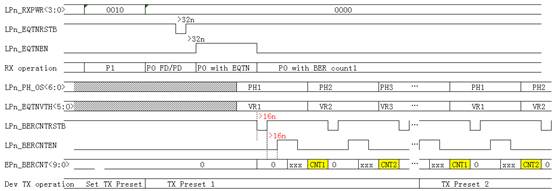
**Step 1:** each time, when SW finish 1.3.1 Step 5, then HW get value LPn\_PH\_OS<6:0> and LPn\_EQTNVTH<5:0>;

**Step 2:**  then reset BER counter(LPn\_BERCNTRSTB is 0), after about 16ns, set LPn\_BERCNTRSTB to 1, after about other 16ns, enable LPn\_BERCNTEN, then BER counter begin coutner bit error

**Step 3:**  wait RX finish BER count, disable LPn\_BERCNTEN

**Step 4:**  at LPn\_BERCNTEN falling edge, the BER count result Epn\_BERCNT<9:0> will send to LPHY; this result will be recorded;

**Step 5:** All lanes of this port should be check, if EP\*\_BERCNT[9:0] value greater than the threshold value of RBER\_THRESHHOLD\_[3:0] (Rx2907h), it means that the point has error, write 0 to the BER\_VAL[2047:0], otherwise it means no error, write 1.



**the result of BER\_VAL[2047:0]**

计算机生成了可选文字:


#### 2.1.2 Mode 2 By Lane

Mode2与Mode1的区别仅在于Step3, By Lane时需要选择Lane number,ALL\_PORT 设置为0， BER\_VAL将报告所有2048个点的scan结果，0为pass，1为fail。

**SW**

**Step 3:** SW coordinates register ALL\_PORT(Rx2901h) to 0, and coordinates register LANE\_SEL[2:0](Rx2901h) to the number which lane needs to be monitor

**HW**

**Step 5:** According the register LANE\_SEL[2:0], the lane will be monitor , for example, if Lane number in register LANE\_SEL[2:0] is 00 and EQ\_PORT\_NAME[8:0] is PE6 9'b0\_0000\_0100 , check EP00\_BERCNT[9:0] value , If greater than the threshold signal value of RBER\_THRESHHOLD\_[3:0] it means that the point has error, in the BER\_VAL[2047:0] write 0, otherwise it means no error, write 1

#### 2.1.3 Mode 3 By Point

By Point是一个debug模式，是对指定Lane的特定point进行测试。不需要设置扫描范围，只需要设置某个点的横纵坐标，SW最后会读出来BER counter值，与阈值比对，决定fail还是pass。

**Step 1:** SW set registers R\_EYE\_MANU\_MUX(Rx2907h)to 1b

**Step 2:** SW set register FRP\*\_BERCNTRSTB(exp. FRP00\_BERCNTRSTB at Rx222h) to 0 and FRP\*\_PH\_OS\_(exp. FRP00\_PH\_OS at Rx222h), and FRP\*\_EQTNVTH\_G\*\_（exp. FRP00\_EQTNVTH\_G1\_ at Rx212h） to the value of the point which will be monitor.

**Step 3:** Wait greater than 16ns and set FRP\*\_BERCNTRSTB to 1.

**Step 4: t**hen wait greater than 16ns and set FRP\*\_BERCNTEN(FRP00\_BERCNTEN at Rx222h) to 1

**Step 5:** Wait some moment， set FRP\*\_BERCNTEN to 0b, read the register EP\*\_BERCNT(ex: EP00\_BERCNT\_ at RX266[9:0]), then set FRP\*\_BERCNTRSTB to 0.

## 3 Eye Monitor Firmware实现过程

### 3.1 Eye Monitor Check Command

首先制定PEMCU 与CPU 之间数据和命令交互协议，如下内容摘自：“PEMCU\_Command\_Spec\_R0.41\_JNY0426.doc”。根据Command\_Spec的PEMCU

Firmware中主要实现两种mode，即Mode1和Mode2。

Mode 1:

For multiple lane port, this mode set ALL\_PORT to 1, all Lane of the point must be monitor.

Mode 2:

For multiple lane port, this mode set ALL\_PORT to 0 and set the lane number with LANE\_SEL[2:0] , which Lane of the point must be monitor

#### 3.1.1 Eye monitor command

CMD type: Eye\_monitor

* BIOS2PEMCU\_scratch registers 0x9DB~9D8h for BIOS to PEMCU in **CMD** cycle:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Bit | 31:16 | 15:8 | 7:4 | 3:0 |
| Description | Reserved | Lane Number | Reserved | CMD |

* **CMD:**

0: Reserved

1: Read

2: Write

3: Tx\_Preset

4: Cursor

5. Eye\_monitor

6~F Reserved

* **Lane Number:**

**2) For mode 2**

0~0x07：Lane0~Lane7

0x08~0xFF : Reserverd

**3) For mode 1**

Reserved

#### 3.1.2 Eye monitor DATA

* BIOS2PEMCU\_scratch registers 0x9D8~9DBh for BIOS to PEMCU in **DATA** cycle:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Bit | 31:24 | 23:16 | 15:8 | 7:0 |
| **Description** | PCIE Port number | \*PHYx | EYE ERROR threshold | Eye Monitor Method |

* **Eye Monitor Method:**

0: Reserved

1: Eye monitor implementation process mode 1: by Port

2: Eye monitor implementation process mode 2: by Lane

3: reserved

4~ 0xF: Reserved

* **EYE ERROR threshold:**

0~ Fh (Standard for pass or fail)

* **PHYx:** (this field only defined for CHA, CHX002 not used)

0: PHYA

1: PHYB

2: PHYC

others: reserved

* **PCIE Port number:**

0：PE0

1：PE1

2：PE2

3：PE3

4：PE4

5：PE5

6：PE6

7：PE7

8 : VPI

9~0xFF Reserved

* The Register for PEMCU 0xF8-0xFB for BIOS to PEMCU in **DATA** cycle

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Bit | 31:24 | 23:16 | 15:8 | 7:0 |
| **Description** | MCU\_EQTNVTH\_2 | MCU\_PH\_OS\_2 | MCU\_EQTNVTH | MCU\_PH\_OS |

* **MCU\_EQTNVTH\_2:** Y value----End VR value

0-0x1F :value range

0x20-0x0FF: reserved

* **MCU\_PH\_OS\_2:** X value-------End PH value

0-0x3F :value range

0x40-0x0FF: reserved

* **MCU\_EQTNVTH:** Y value----Start VR value

0-0x1F :value range

0x20-0x0FF: reserved

* **MCU\_PH\_OS:** X value------- Start PH value

0-0x3F :value range

0x40-0x0FF: reserved

* The Register for PEMCU 0xFC-0xFF for BIOS to PEMCU in **DATA** cycle

|  |  |  |
| --- | --- | --- |
| Bit | 31:8 | 7:0 |
| **Description** | Reserved | Scan Time |

* **Scan Time :**

0 - 1Ah: Reserved

1Bh: BERCNTEN assert 130ns

1Ch: BERCNTEN assert 140ns

... …

FEh: BERCNTEN assert 2400ns

FFh: BERCNTEN assert 2410ns

(270ns - 2550ns for one point, 0.553ms-5.23ms for 2048 points)

* PEMCU2BIOS\_scratch registers for PEMCU to BIOS in **Result** cycle:

**Result:**

0: Reserved

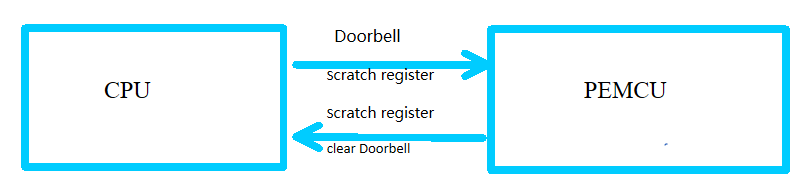
1: Success

2: Failed

3~0xFF: Reserved

|  |  |  |
| --- | --- | --- |
| Bit | 31:8 | 7:0 |
| Description | Reserved | Result |

### 3.2 PEMCU与CPU之间的数据交互

SW UEFI Shell Tool Pemcu Eye Monitor Firmware

计算机生成了可选文字:
数 据 交 互 媒 介 
CPU-S 、 、 
1 ） 、 、 Trite C 、 ID,'DATA Cyce 
PEMCU side 
PEMCU-Firmware 
Scratch register for BIOS № PEMCU 
Rx9D8 
BIOS2PE 、 •ICC DB 
3 ） Process Scratch 
ster information 
Enter ISR 
Exit ISR 
5 ） write 1 to clear 
Doorbell regster 
| 的 Return Result by 
| Pemcu № BIOS scratch regis 
2 ） Trigger INT by write 1 
to doorbell register 
Eye Momtor Tool 
CPU side 
6 ） Sw check doorbell = 0 
7 ） Sw get result information | 
Scratch register for 
Rx9DO 
PEMCU to BIOS 

### 3.3 PEMCU Firmware 执行流程

计算机生成了可选文字:
Pemcu firmware run 
Walte Eye monltor INT 
CMD IS 
Monitor CMD ， 
Process Stepl— 2 
Mode l: •tALL of 
to 
： ALL PORTC,x2901h) 
Process Step3 
“ LANE SEL12111(Rx2901h) 
Process Step4-5 
Step6 pollmg 
SCAN DONE 
6 &Step7 在 Eye TX 
Step7: read the 
BER VAL [ 2 7 〕 0 ] (from 
Rx2A00h to Rx2AFFh 
PEMCU 
： al ， 
It ofBER 
R = 2 00 ． 2AFF 20 ； S 

Step 6 和Step 7在Eye Monitor Tool 中实现，Eye monitor Tool 除了完成上述的step 6 和step 7以外，主要任务是给PEMCU 发命令和数据，根据Pemcu Commen spec的协议发送CMD 和DATA Cycle给PEMCU。Eye Monitor Tool使用方法见Chirs J 文档“Note.docx”。

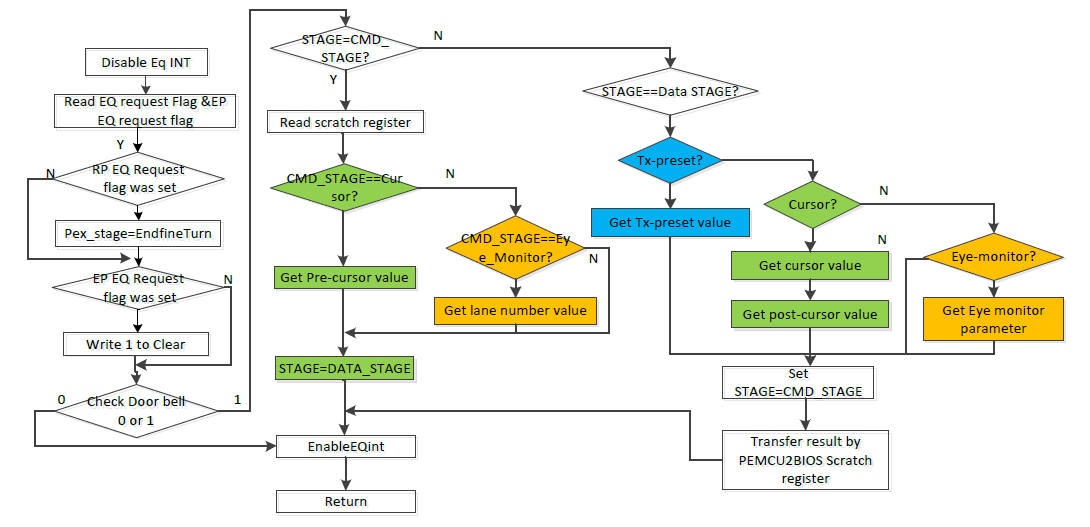
### 3.4 Eye Monitor/EQ ISR

  发生中断之前，EFI Shell Tool或者BIOS需要将Door bell设置为1，并将CMD 写入对应的scratch register中；FW收到中断之后,首先判断：

1）若CMD = Cursor，那么就在CMD\_STAGE和DATA STAGE分别获取pre-cursor，cursor，post-cursor的值。

2）若CMD = Preset ,那么直接跳过CMD\_STAGE,直接到DATA\_STAGE中获取Preset。

3）若CMD =Eye\_Monitor\_CMD, 那么就分CMD\_STAGE和DATA STAGE分别将lane number, eye monitor method, PCIE port number ,THRESHOLD的值获取到。下图为EQ及Eye Monitor 中断服务程序处理流程。



Firmware 中INT执行流程

## 4 Eye monitor Test Tool

Eye Monitor Tool使用方法见Chirs J 文档“Note.docx”。

## 附录1 Scratch register

Offset Address: FB-F8h (PCIEPHYCFG)  
PEMCU Use Register 2 Default Value: 0000h

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | **Attribute** | **HW Property** | **Default** | **Description** | Mnemonic | ChipRev | **PwrDm** | **S** | **P** | **E** |
| 31:0 | RW | RO | 0 | **The Register for PEMCU**  PEMCU will write this register to show some information about PEMCU status according to firmware.  What the 0/1 settings mean will be defined by the firmware.  @((#control\_pemcurw = RW)) | PEMCU\_REG\_2 |  | vcc | x | x | x |

Offset Address: FF-FCh (PCIEPHYCFG)  
PEMCU Use Register 3 Default Value: 0000h

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | **Attribute** | **HW Property** | **Default** | **Description** | Mnemonic | ChipRev | **PwrDm** | **S** | **P** | **E** |
| 31:0 | RW | RO | 0 | The Register for PEMCU  PEMCU will write this register to show some information about PEMCU status according to firmware.  What the 0/1 settings mean will be defined by the firmware.  @((#control\_pemcurw = RW)) | PEMCU\_REG\_3 |  | vcc | x | x | x |

Offset Address: 9DB-9D8h (PCIEPHYCFG)   
EQ Int to MCU Default Value: 0000 0000h

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | **Attribute** | **HW Property** | **Default** | **Description** | Mnemonic | ChipRev | **PwrDm** | **S** | **P** | **E** |
| 31:0 | RW | RO | 0 | **Scratch Register for BIOS to PEMCU**  **BIOS write information here, PEMCU read this information** | BIOS2PEMCU\_scratch |  | vcc | x | x | x |

Offset Address: 9D3-9D0h (PCIEPHYCFG)   
EQ Int to MCU Default Value: 0000 0000h

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | **Attribute** | **HW Property** | **Default** | **Description** | Mnemonic | ChipRev | **PwrDm** | **S** | **P** | **E** |
| 31:0 | RW | RO | 0 | **Scratch Register for PEMCU to BIOS**  **PEMCU write information here, BIOS read this**  **information**  @((#control\_pemcurw = RW)) | PEMCU2BIOS\_scratch |  | vcc | x | x | x |

Offset Address: 9CCh (PCIEPHYCFG)   
EQ Int to MCU Default Value: 00h

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | **Attribute** | **HW Property** | **Default** | **Description** | Mnemonic | ChipRev | **PwrDm** | **S** | **P** | **E** |
| 7:1 | RO | NA | 0 | **Reserved** | Rsv\_9CC |  | vcc | x | x | x |
| 0 | RW1C | RW | 0 | **Reserved**  ((For Internal Reference:  Doorbell Bit for CPU to PEMCU  This bit will be set when CPU wants PEMCU to get some information in register CPU2PEMCU\_scratch.  @((#pemcu\_control = PEMCU\_MAS)) ))  Note:  CPU access this bit is RW.  PEMCU access this bit is RW1C. | BIOS2PEMCU\_DB |  | vcc | x | x | x |

## 附录2 Eye Monitor register

Offset Address: 2900h (PCIEPHYCFG)   
Eye Monitor Default Value: 00h

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | **Attribute** | **HW Property** | **Default** | **Description** | Mnemonic | ChipRev | **PwrDm** | **S** | **P** | **E** |
| 7:0 | RW | RO | 0 | **BERCNTEN Control Timer**  0 ~ 1Ah: Reserved  1Bh: BERCNTEN assert 130ns  1Ch: BERCNTEN assert 140ns  ... …  FEh: BERCNTEN assert 2400ns  FFh: BERCNTEN assert 2410ns  Step by 10ns  @((#control\_pemcurw = RW)) | MCU\_EYESCANTM\_ PEnn\_ |  | vcc | x | x | x |

Offset Address: 2902-2901h (PCIEPHYCFG)   
Eye Monitor 2 Default Value: 0000h

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | **Attribute** | **HW Property** | **Default** | **Description** | Mnemonic | ChipRev | **PwrDm** | **S** | **P** | **E** |
| 15 | RW | RO | 0 | **EYE Monitor Clock Gate Enable Signal**  0: Enable  1: Unable, and also set D0F4\_Rx60[7] (RDYNPM) to 1,the eye monitor clock will be close.  @((#control\_pemcurw = RW)) | REYECLKEN |  | vcc | x | x | x |
| 14 | RW1C | RW | 0 | **EYE SCAN Done, PEMCU can check the result now**  @((#control\_pemcurw = RW))  (( For Internal Reference:  @((control = set\_EYESCAN\_DONE\_PEnn)) )) | EYESCAN\_DONE\_PEnn |  | vcc | x | x | x |
| 13 | RW  ((RWHC)) | RW | 0 | **Eye Scan Enable**  1: Enbale  @((#control\_pemcurw = RW))  (( For Internal Reference:  @((control =MCU\_EYESCANEN\_clr\_PEnn)) )) | MCU\_EYESCANEN\_PEnn |  | vcc | x | x | x |
| 12:4 | RW | RO | 0 | **The Port Doing EQ And EYE SCAN**  VPI 9'b1\_0000\_0000  PE4 9'b0\_0000\_0001  PE5 9'b0\_0000\_0010  PE6 9'b0\_0000\_0100  PE7 9'b0\_0000\_1000  PE0 9'b0\_0001\_0000  PE1 9'b0\_0010\_0000  PE2 9'b0\_0100\_0000  PE3 9'b0\_1000\_0000  @((#control\_pemcurw = RW)) | EQ\_PORT\_NAME\_ |  | vcc | x | x | x |
| 3:1 | RW | RO | 0 | **Available when ALL\_PORT is 0**  **select one lane inside selected port, scan result only associated with this lane**  @((#control\_pemcurw = RW)) | LANE\_SEL\_ |  | vcc | x | x | x |
| 0 | RW | RO | 0 | **Scan result associated with all port or not**  0: only associated with the lane selected by LANE\_SEL\_  1: associated with all ports  @((#control\_pemcurw = RW)) | ALL\_PORT |  | vcc | x | x | x |

Offset Address: 2903h (PCIEPHYCFG)   
Eye Monitor 3 Default Value: 00h

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | **Attribute** | **HW Property** | **Default** | **Description** | Mnemonic | ChipRev | **PwrDm** | **S** | **P** | **E** |
| 7 | RW | RO | 0 | **Reserved** | RSV\_2903 |  | vcc | x | x | x |
| 6:0 | RW | RO | 0 | Scan begin point of PH\_OS  @((#control\_pemcurw = RW)) | MCU\_PH\_OS\_ |  | vcc | x | x | x |

Offset Address: 2904h (PCIEPHYCFG)   
Eye Monitor 4 Default Value: 00h

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | **Attribute** | **HW Property** | **Default** | **Description** | Mnemonic | ChipRev | **PwrDm** | **S** | **P** | **E** |
| 7:6 | RW | RO | 0 | **Reserved** | RSV\_29004 |  | vcc | x | x | x |
| 5:0 | RW | RO | 0 | Scan begin point of EQTNVTH  @((#control\_pemcurw = RW)) | MCU\_EQTNVTH\_ |  | vcc | x | x | x |

Offset Address: 2905h (PCIEPHYCFG)   
Eye Monitor 5 Default Value: 00h

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | **Attribute** | **HW Property** | **Default** | **Description** | Mnemonic | ChipRev | **PwrDm** | **S** | **P** | **E** |
| 7 | RW | RO | 0 | **Reserved** | RSV\_29005 |  | vcc | x | x | x |
| 6:0 | RW | RO | 0 | Scan end point of PH\_OS  @((#control\_pemcurw = RW)) | MCU\_PH\_OS\_2\_ |  | vcc | x | x | x |

Offset Address: 2906h (PCIEPHYCFG)   
Eye Monitor 6 Default Value: 00h

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | **Attribute** | **HW Property** | **Default** | **Description** | Mnemonic | ChipRev | **PwrDm** | **S** | **P** | **E** |
| 7:6 | RW | RO | 0 | **Reserved** | RSV\_29006 |  | vcc | x | x | x |
| 5:0 | RW | RO | 0 | Scan end point of EQTNVTH  @((#control\_pemcurw = RW)) | MCU\_EQTNVTH\_2\_ |  | vcc | x | x | x |

Offset Address: 2907h (PCIEPHYCFG)   
Eye Monitor 7 Default Value: 10h

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | **Attribute** | **HW Property** | **Default** | **Description** | Mnemonic | ChipRev | **PwrDm** | **S** | **P** | **E** |
| 7:5 | RW | RO | 0 | **Reserved** | RSV\_2907 |  | vcc | x | x | x |
| 4 | RW | RO | 1b | **Auto scan or manual scan**  0: auto scan by hardware, result keep at BER\_VAL  1: manual scan, all EPHY port control directly by software.  @((#control\_pemcurw = RW)) | R\_EYE\_MANU\_MUX |  | vcc | x | x | x |
| 3:0 | RW | RO | 0 | **Only BERCNT large than** RBER\_THRESHHOLD **will be consider real error**  @((#control\_pemcurw = RW)) | RBER\_THRESHHOLD\_ |  | vcc | x | x | x |