# Design specification of Peripheral readout for MOST2

 Version 2.0

Modification record:

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Modification | Date | Note |
| 1 | Creation | 2019-2-22 | Xiaomin WEI |
| 2 | I/O location | 2019-3-04 | Ying ZhangTianya Wu |
| 3 | Modify pixel mode as a normal modeReduce the required signal linesModify spi interface | 2019-4-22 | Xiaomin WEI |
| 4 | I/O location of FEI3 version | 2019-4-22 | Taken from Wei’s email by Xiaomin  |

# 1. Introduction

This design aims for the readout of pixel array in MOST2.

The CPS is designed for CEPC. The hit density and the chip data rate are calculated in Table.1. In this estimation, the chip area is 3.2768cm2 (1024\*512 pixel array, 25um pixel pitch), and the cluster size is 3 pixels. Each hit pixel is recoded with 32 bits (See Fig.3, Timestamp: 8 bits, Address: 19 bits). The trigger latency is supposed 3~6 us, and the average trigger rate is **50 kHz**.

Table. 1 The hit density of CEPC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameter | Unit | Higgs | W | Z |
| Bunch spacing | **ns** | **680** | **210** | **25** |
| Hit density | **hits/bunch/cm2** | **2.5** | **2.5** | **0.2** |
| hits/bunch | 8.2 | 8.2 | 0.66 |
| pixels/bunch | 25 | 25 | 2 |
| Hit pixel rate | MHz/cm2 | 11 | 36 | 24 |
| MHz/chip | 36 | 120 | 80 |
| Chip data rate(triggerless) | Gbps  | 1.15 | 3.84 | 2.56 |
| MHz/32bit | 36 | 120  | 80 |
| Chip data rate(trigger, no error) | Mbps（Average） | 40 | 40 | 3.2 |
| MHz/32bit | 1.25 | 1.25 | 0.1 |
| Chip data rate(trigger, 7LSB error) | Mbps （Average） | 40 | 40 | 25.6 |
| MHz/32bit | 1.25 | 1.25 | 0.8 |
| Designed data rate (trigger) | MHz/32bit | **5** | **5** | **5** |
| Designed data rate(triggerless) | MHz/32bit | **160** | **160** | **160** |

# 2. Features

• Pixel array 512\*1024

• Pixel pitch 25 μm

• Support trigger and triggerless mode

• Data compression

 Suggested in triggerless mode

• Column dead time 500 us

• A modified SPI interface for chip in series

# 3. Interface Discription



Fig.3.1 the interface between peripheral logic and related blocks (此图需要修改)

Table.2 Interface Signal Description

|  |  |  |
| --- | --- | --- |
| **Signal Name** | **Type** | **Descriptions** |
| **Common inout pins to PAD** |
| TEST[1:0] | I | 00: Device in Normal operation mode.01: Device in Scan test mode.10: Device in Memory BIST mode.11: invalid. |
| CLK\_40MHZ | I | Input system clock, 40MHz |
| XRSTN | I | External reset. Asynchronous reset, low active.  |
| STAR\_I (STAR\_I/SCAN\_EN) | I | When TEST[1:0] = 00, a positive pulse for generating SPI slaver select signal. See section 4.1.When TEST[1:0] = 01, Scan shift enable pin.When TEST[1:0] = 10 or 11, invalid. |
| STAR\_OUT | O | When TEST[1:0] = 00, a positive pulse for generating SPI slaver disable signal. See section 4.1.When TEST[1:0] = 10 or 11, invalid. |
| SPI\_MOSI(SPI\_MOSI/SI0 /TEST\_H)  | I | When TEST[1:0] = 00: SPI slaver data input.When TEST[1:0] = 01: Scan chain 0 shift data input pin.When TEST[1:0] = 10: Memory BIST test\_h input.When TEST[1:0] = 11, invalid. |
| SPI\_CLK(SPI\_CLK/TEST\_CLK) | I | When TEST[1:0] = 00: SPI clock input. When TEST[1:0] = 01: Scan test clockWhen TEST[1:0] = 10: Memory BIST clock input.When TEST[1:0] = 11, invalid. |
| SPI\_MISO(SPI\_MISO /SO0/TEST\_DONE ) | O | When TEST[1:0] = 00: SPI slaver data output.When TEST[1:0] = 01: Scan chain 0 shift data output pin.When TEST[1:0] = 10: Memory BIST test result TEST\_DONE output pin.When TEST[1:0] = 11, invalid. |
| SPI\_MISO\_ OEN | O | When TEST[1:0]=00: SPI\_MISO pin GPIO control. When TEST[1:0] =01,10, or 11, invalid. |
| INT(INT/ SO1/FAIL\_H) | O | When TEST[1:0] = 00: An internal abnormal interrupt output pin. When INT=1, please read register0 to check memory full flag.When TEST[1:0] = 01: Scan chain 1 shift data output pin.When TEST[1:0] =10: Memory BIST test result FAIL\_H output pin.When TEST[1:0] = 11, invalid. |
| TRIGGER(TRIGGER/SI1) | I | When TEST[1:0]=00: Trigger input pin. Fig.2. The average frequency is 50KHz. The actual frequency depends on the happen of true event. In the trigger mode, the data fit the TRIGGER arriving time are read out, and the other data will be disposed. It is disable in the triggerless mode. See Fig.4 for the data transmission.When TEST[1:0] = 01: Scan chain 1 shift data input pin.When TEST[1:0]=10,11: invalid. |
| SI2 / ApulseSI3 / Dpulse | I | When TEST[1:0] = 01:Scan chain 2 shift data input pin.Scan chain 3 shift data input pin.When TEST[1:0] = 00:APLUSE (pixel\_analog\_calibration\_pulse) A test pin for analog pixel. It should be a positive pulse. Pulse width 1us. Connect to pin Apulse.DPULSE (pixel\_digital\_test\_pulse) A test pin for digital pixel. It should be a positive pulse. Pulse width 1us. Connect to pin Dpulse.When TEST[1:0]=10,11: invalid. |
| SO2,SO3 | O | Scan chain 2 shift data output pin.Scan chain 3 shift data output pin. |
| **Power on reset** |
| POR\_RSTN | I | Power on reset.  |
| **PLL** |
| PSET | O | Logic is equal to ~(XRSTN&POR\_RSTN). |
| CLK\_160MHZ(CK\_OUT) | I | Used for data output from FIFO2 in triggerless mode.TRIGN=0, 5 MHz, invalid in peripheral logic.TRIGN=1, NSEL=0, 160 MHzTRIGN=1, NSEL=1, 80 MHz |
| ~~CLK\_40MHZ~~ | ~~I~~ | ~~40MHz, the main clock in peripheral logic.~~ |
| LVDS\_DATA [31:0](D[31:0]) | O | Data output to LVDS transmitter. LVDS\_DATA [31:0] should be received on the rising edge of LVDS\_REF\_CLK.See Fig.4.3.1 for the data format and Fig. 4.3.2 for the timging sequence. |
| LVDS\_DATA\_OPC | O | LVDS\_DATA [30:0] data odd parity check bit. It is generated before writing into FIFO2. Did not output in the present version. |
| LVDS\_REF\_CLK(SCK) | O | LVDS PLL reference clock.In trigger mode, the output frequency is defined by DOFREQ[1:0], the minimum output frequency is 2MHz and the default frequency is 5 MHz. In triggerless mode, the output frequency is the same with CLK\_160MHZ.LVDS\_DATA [31:0] should be received on the rising edge of LVDS\_REF\_CLK.See Fig.4.3.1 for the data format and Fig. 4.3.2 for the timging sequence.Note: The LVDS PLL should support a 2MHz~160MHz reference input. |
| **Pixel array** |
| READi (i=0,1,…,511) | O | Hit address read signal. Reset the hit pixel when high. |
| FASTORi (i=0,1,…,511) | I | OR results of all the pixel digital output in a double column.  |
| ADDRi[9:0] (i=0,1,…,511) | I | The address of the hit pixel in a double column. |
| G\_rsti (i=0,1,…,511) | O | Equal to (Xrstn&por\_rstn), Low active. |
| Load\_ci (i=0,1,…,511) | O | Latch enable for pixel analog mask. See section 4.5. |
| Load\_mi (i=0,1,…,511) | O | Latch enable for pixel digital mask. See section 4.5. |
| Apulsei(i=0,1,…,511) | O | pixel\_analog\_calibration\_pulse A test pin for analog pixel. It should be a positive pulse. Pulse width 1us. Connect to pin Apulse. |
| Dpulse(i=0,1,…,511) | O | pixel\_digital\_test\_pulse A test pin for digital pixel. It should be a positive pulse. Pulse width 1us. Connect to pin Dpulse. |
| Con\_datai (i=0,1,…,511) | O | The input of the pixel mask shift chain for each double column. See section 4.5. |
| Con\_clki (i=0,1,…,511) | O | Clock for the pixel mask shift chain for each double column. It is the same for all the 512 double columns. See section 4.5. |
| Output of Configuration registers |
| xreserved2[15:0], xreserved3[15:0], xreserved4[15:0], xreserved5[15:0],Xreserved6[15:0],Xreserved7[15:0], Xreserved8[15:0], Xreserved9[15:0], Xreserved10[15:0], Xreserved11[15:0],Xreserved12[15:0],Xreserved13[15:0], | O | See section 5. Control and Status Registers |

# 4. Interface Timing Discription

## 4.1 Modified SPI slaver Interface

## 4.1.1 Modified SPI slavers Cascade Connection



Fig.4.1.1 Modified SPI Slaver Cascade Connection

To realize multi-chip expansion, a Modified SPI interface is designed in this chip. The multi\_chip connection is show in Fig.4.1.1. The CLK, generated by master controller, is a standard SPI clock for multi Modified SPI slaver synchronization. The Mater Out Slaver In (MOSI) is the data bus driven by Master controller. The Mater In Slaver Out (MISO) is the data bus driven by slavers, the operation mode of MOSI and MISO is the same as standard SPI interface. The difference of the Modified SPI and the standard SPI is the chip selection. In our Modified SPI interface, the Modified SPI slaver is selected at the raising edge of a start pulse (STAR\_I ), and de-selected by rising edge of next chip start pulse (the raising edge of current slaver STAR\_OUT) .



Fig.4.1.2 Operation of multi Modified SPI slavers cascade connection

 The operation of multi Modified SPI slavers cascade connection is shown in Fig.4.1.2. When 1.8V VDD power on, the internal POR or external RSTN will reset START\_OUT of all chips. Therefore, all the Modified SPI slavers are deselected, and all MISO output Hiz if OEN is used. When master controller sends a start pulse (START\_PULSE0) to chip01, the chip01 is selected. The master controller and the chip 01 communicates point-to-point. The data driven by master on data bus MOSI will only be received by chip 01. The MISO of chip 01 output normally, therefore the data bus MISO controlled by chip01. The master sends command packets one by one. When the master controller wants to configurate the chip02, the master controller should set SPI\_D =1 (control registers 00001) in chip01. When SPI\_D of chip01 is set as 1, the chip 01 will assert a pulse internally on START\_OUT port of chip 01 which connects with STAR\_I of chip 02. As a result, the chip01 operation is disabled and the chip02 operation is enabled by setting SPI\_D of chip 01.



Fig.4.1.3 tri-state GPO Control

 If MISO is used as bus connection, a tri-state MISO general purpose output (GPO) as shown in 4.1.3 or general purpose input/output (GPIO) IO circuits is necessary. In the digital\_top module, SPI\_MISO\_OEN is the tri-state control of the IO circuits. When SPI\_MISO\_OEN=1，the IO circuits output two state of SPI\_MISO; When SPI\_MISO\_OEN=0，the IO circuits output Hiz.

## 4.1.2 Modified SPI slavers timing



Fig.4.1.4 Modified SPI slaver operation timing

The Modified SPI slaver latches input data (MOSI) on rising edge of clock (CLK). Data out (MISO) is changed during falling edge of the clock.

The Modified SPI interface communication relies on packets of 16 bits. The start pulse rising edge will reset internal CSB, which is SPI slaver select signal. When the select signal is activated (internal\_CSB falling edge), the SPI interface starts to sample the MOSI pin on each rising edge of CLK. The internal\_CSB will stay low for the whole packet transmission period, except for a SPI\_D bit write 1 command, a SPI\_D bit write 1 command will set internal\_CSB to 1 after current packet transferred completed.



Fig.4.1.5 internal\_CSB generate circuits

With internal\_CSB =0, the first bit that is taken into account has to be a logic 1. All incoming bits with a logic 0 arriving prior the logic 1 are ignored. The first bit at logic 1 is part of the first packet (first bit of 16 bits packet) and acts as synchronization bit. Once it is synchronized, the Modified SPI interface keeps tracking the packet positions using a clock counter. Each group of 16 clock pulses defines a new packet. The following packets are not re-synchronized and solely relied on the initial alignment. The master can verify that the synchronization is still valid by checking that each reply on MISO starts with a pattern of three-bit 1 followed by three-bit 0. If the master and slaver lose synchronization, a new start pulse should be send for interface re-synchronization.

The internal\_CSB will be set to 1 after the SPI\_D bit write to 1 or chip reset or start pulse input , all Modified SPI interface logic and registers will be reset when the internal\_CSB is set to 1. The control registers are kept unchanged as long as the power supply is maintained. The interface reset is released as soon as internal\_CSB is set to 0.

The Modified SPI interface is used to write or read the internal control or status registers. The W/R bit in 16-bit packet is used to define a write operation or a read operation (“0” write, “1” read ) of current packet. The A4~0 defines the address of the target register to be operated on. DI7~0 is the 8 bits data to be write in control register during write operation, and the DO7~0 is the 8 bits data returned to SPI master during read operation

## 4.2 Tigger



Fig.4.2.1 Timing of input signal TRIGGER.

The average cycle of TRIGGER is 20us. The pulse width of TRIGGER is larger than a clock cycle (25ns). TRIGGER is synchronized to Trigger\_syn with the positive edge of clk. The TRIGGER arriving time are recoded at the positive edge of Trigger\_syn.

## 4.3 LVDS Transmitter



Fig.4.3.1 The output data format.

The output data format is defined as shown in Fig.4.3.1. LVDS\_DATA[31] is the data available flag. 1:data valid. 0 : data invalid. LVDS\_DATA [30:23] are 8 bits time stamp. LVDS\_DATA [22:14] are the double column address. LVDS\_DATA [13:4] are the hit pixel address in a double column. LVDS\_DATA [3:0] is the data compression pattern.

Fig.4.3.2 Data output Timing.

The data output timing is shown in Fig.4.3.2 The output data LVDS\_DATA are sending at the falling edge of LVDS\_REF\_CLK. In the triggerless mode, the output frequency is 160MHz. In the trigger mode, the external trigger start signal (trigger) is synchronized (Trigger\_syn) by CLK\_40MHz. The output data frequency is set by DOFREQ[1:0].

## 4.4 Pixel Array

Fig.4.4.1 Timing of double column readout.

Readout timing of double column is shown in Fig.4.4.1.FASTOR is set when any pixel in the double column is hit. Since the hit pixel will be reset after readout, FASTOR changes into zero when the last hit address is read out. TDFOR should be less than one clock cycle 25ns. FASTOR is synchronized as Fastor\_syn in peripheral logic. READ is active when FASTOR\_sync is “1”. The cycle of READ is 50ns which corresponds the maximal delay of the addressing encoding (TDA) is 20 ns. For the peripheral logic, the hit timestamp (Time\_stamp[7:0]) is recoded at the positive edge of Fastor\_sync. The address of hit pixel (ADDR[9:0]) is taken at the clock positive edge when ADDR[9:0] are available.

Note: all the delays are referred to READ at the end of column. The delay of READ in the pixel array should be included.

## 4.5 pixel array mask funciton

## 4.5.1 circuit discription

Fig.4.5.1 (a) mask function in pixel level



Fig.4.5.1 (b) block diagram pixel mask function



Fig.4.5.1(c) timing sequence of pixel mask function

## 4.5.2 flow for software operation

The generation of con\_clk, load\_c/load\_m are realized by writing control registers 00110-00111. When the register index is 00110, the input data will be connect to the col\_mask registers at the end of column. The input DI7~DI0 corresponds to con\_data504~con\_data511, …, and con\_data0~con\_data7. When the register index is 00111, con\_clk is generated definitely and loadc/loadm is generated according the data in the register 00111. The setting flow for a pixel array of 1024\*512 is shown in Fig 4.5.2(a). The timing sequence of con\_clk (2 spi\_clk cycles), load\_c/load\_m (4 spi\_clk cycles) is shown in Fig 4.5.2 (b).



Fig.4.5.2 (a) Setting flow for pixel mask function



Fig.4.5.2 (b)Timing sequence of con\_clk (2 spi\_clk cycles), load\_c/load\_m (4 spi\_clk cycles)

# 5. Control and Status Registers

|  |  |  |
| --- | --- | --- |
| Operation Type(W/R) | Address | DB7~0 |
| DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DB0 |
| R | 00000 | FIFO1\_FUL | FIFO2\_FUL | FIFOT\_FUL |  |  |  |  |  |
| W/R | 00001 | TRIGN/TMOD | CPRN | DOFREQ[1:0] |  | CTM | SPI\_D |  |
| W/R | 00010 | TRIGGER\_LATENCY [7:0] |
| W/R | 00011 | TRIGGER\_UNCERTAIN [2:0] |  |
| W/R | 00100 | ADR[9:2] |
| W/R | 00101 | ADR[1:0] | FVL[2:0] | API | F/V | OCTE |
| W | 00110 | Pixelmask\_data[7:0] |
| W | 00111 | Loadc\_e | Loadm\_e |  |  |  |  |  |  |
| W/R | 01000 | Resrved2 [7:0] >>>>>>DAC\_REG [7:0] Default: 0001 0001 |
| W/R | 01001 | Resrved2 [15:8] >>>>>>DAC\_REG [15:8] Default:0110 0000 |
| W/R | 01010 | Resrved3 [7:0] >>>>>>DAC\_REG [23:16] Default: 1000 0000 |
| W/R | 01011 | Resrved3 [15:8] >>>>>>DAC\_REG [31:24] Default: 0000 0001 |
| W/R | 01100 | Resrved4 [7:0] >>>>>>DAC\_REG [39:32] Default:0000 0110 |
| W/R | 01101 | Resrved4 [15:8] >>>>>>DAC\_REG [47:40] Default: 1110 0000 |
| W/R | 01110 | Resrved5 [7:0] >>>>>>DAC\_REG [55:48] Default: 0001 1111 |
| W/R | 01111 | Resrved5 [15:8] >>>>>>DAC\_REG [63:56] Default: 1000 0000 |
| W/R | 10000 | Resrved6 [7:0] >>>>>>DAC\_REG [71:64] Default:0111 1111 |
| W/R | 10001 | Resrved6 [15:8] >>>>>>DAC\_REG [79:72] Default:0000 0000 |
| W/R | 10010 | Resrved7 [7:0] >>>>>>DAC\_REG [87:80] Default:1111 1110 |
| W/R | 10011 | Resrved7 [15:8] >>>>>>DAC\_REG [95:88] Default:0000 0001 |
| W/R | 10100 | Resrved8 [7:0] >>>>>>DAC\_REG [103:96] Default:1111 1000 |
| W/R | 10101 | Resrved8 [15:8] >>>>>>DAC\_REG [111:104] Default: 0000 0111 |
| W/R | 10110 Resrved9 [7:0] | BSEL | ISEL1 | ISEL0 | CKESEL | NSEL | DSEL |  |  |
| W/R | 10111 | Resrved9 [15:8] |
| W/R | 11000 | Resrved10 [7:0] |
| W/R | 11001 | Resrved10 [15:8] |
| W/R | 11010 | Resrved11 [7:0] |
| W/R | 11011 | Resrved11 [15:8] |
| W/R | 11100 | Resrved12 [7:0] |
| W/R | 11101 | Resrved12 [15:8] |
| W/R | 11110 | Resrved13 [7:0] |
| W/R | 11111 | Resrved13 [15:8] |

**Register 00000:** Read only register.

FIFO1\_FUL: When any one of 512 double column FIFOs （FIFO1） is full, FIFO1\_FUL will be set automatically.

FIFO2\_FUL: When any one of 4 chip FIFOs（FIFO2） is full. FIFO2\_FUL will be set automatically. FIFO2 is used in both trigger and triggerless modes. In triggerless mode, the number of FIFO2 should be discussed in the final version for 512 double column readout. There is only one FIFO2 (256\*32bit) in the present version for 96 double column readout.

FIFOT\_FUL: When trigger FIFO is full. FIFOT\_FUL will be set automatically. Trigger FIFO is only active in trigger mode. In triggerless mode, FIFOT\_FUL is 0.

**Register 00001:**

TRIGN:

0: tigger mode,

1: triggerless mode

CPRN:

1: by pass the data compression block

0: data compression for the addresses of four adjacent pixels in a double column. CIM should be set probably according the order of the addresses from the pixel array.

1. CTM=1, the received address is in ascending order.

 The first pixel address in a double column is DOUT[14:5]. DOUT[4],DOUT[3],DOUT[2] and DOUT[1] note whether the address of DOUT[14:5]+4, DOUT[14:5]+3, DOUT[14:5]+2, DOUT[14:5]+1 is ‘1’, respectively. For example, DOUT[14:5]=19’d0, DOUT[4:1]=3’b0001, then the pixel digital outputs of addresses 19’d0,19’d1,19’d2,19’d3, 19’d4are 1,1,0, 0 and 0.

1. CTM=0, the received address is in descending order.

 The first pixel address in a double column is DOUT[14:5]. DOUT[4],DOUT[3],DOUT[2] and DOUT[1] note whether the address of DOUT[14:5]-4, DOUT[14:5]-3, DOUT[14:5]-2, DOUT[14:5]-1 is ‘1’, respectively. For example, DOUT[14:5]=19’d4, DOUT[4:1]=3’b0001, then the pixel digital outputs of addresses 19’d4,19’d3,19’d2,19’d1, 19’d0are 1,1,0, 0 and 0.

CTM: pixel array address from double column in descending or ascending order. Active when CPRN=0.

 0: descending (the received addresses are 3,2,1,0…)

 1: ascending (the received addresses are 0,1,2,3…)

SPI\_D: Start pulse control bit. See 4.1 Modified SPI Slaver Interface.

0: Dis-assert start pulse;

1: Assert a start pulse.

DOFREQ[1:0] :Set the data output (DOUT[31:0]) frequency in trigger mode.

00: 2MHz

01: 5MHz,

10: 10MHz,

11: 20MHz

**Register 00010, Register 00011: Setting the time window in trigger mode.**

Note:  TRIGGER\_LATENCY + TRIGGER\_UNCERTAIN <=  256 – (fifo1\_deepth 15). In the verision for first MPW，please ensure TRIGGER\_LATENCY + TRIGGER\_UNCERTAIN <241. Otherwise, some data may be lost in trigger mode.

TRIGGER\_LATENCY [7:0]: Trigger latency: 0-6us

00000000: 0ns

00000001: 25ns

 …

11101111: 5.975us (TRIGGER\_UNCERTAIN must be 000, 001 or 010. Matched timestamp range: present time-5.950us ~ present time -5.975us)

11110000: 6us (TRIGGER\_UNCERTAIN must be 000 or 001. Matched timestamp rang: present time-5.975us ~ present time -6us)

11110001~11111111: invalid

TRIGGER\_UNCERTAIN [2:0] :Negative trigger error:

000: 0ns

001:25ns

010:50ns (default)

011:75ns

100:100ns

101:125ns

110:150ns

111:175ns

If the trigger uncertain is negative, please adjust the trigger latency and provide negative trigger error. See Fig.7.



Fig.5.1 Example of setting trigger latency and trigger uncertain.

 Suppose the trigger signal comes at 6us, and users wants to acquire the hits from 2.925us to 3.075us, then we should set TRIGGER\_LATENCY as 8’d 123 (123\*25ns=3.075us), and TRIGGER\_UNCERTAIN as 3’b110. (0.15us).

**Register 00100~00101:**

On chip channel test control. The function is not included in the present version.

OCTE: On chip channel test enable. 1: enable , 0: disable (default).

F/V: Fastor or Valid waveform select. 0: fastor waveform, 1: Valid waveform.

FVL[2:0]: Fastor or Valid waveform length. Unit in read counts

API: Line-by-line or interlaced address generating selecting. 0: Line-by-line, 1: interlaced

ADR[9:0]: The on chip channel test controller start address.

**Register 00110~00111: setting for load\_C, load\_m**

**Register 01000~10101: DAC control and setting. Please double click the document to view the details.**

****

**Register 10110: PLL&LVDS setting, Refer the presentation of Xiaoting Li.**

BSEL: Loop bandwidth configuration

ISEL1,ISEL0: charging current configuration

CKESEL: clock delay selection

NSEL: Division selection. NSEL=0 for 2.56 GHz; NSEL=1 for 1.28 GHz.

DSEL: Input data selection. DSEL=0 for pixel array data; DSEL=1 for PRBS data.

TMOD : It should be setting in Register 00001.

**Register 10111~11111:**

Reserved registers for others.

# 6. I/O location of pixel array

  For double column FE-I3, top cell is Pixel\_DBCol\_2c, (0,0) is supposed at bottom left of the layout. M4 layer is used with wide of 0.28um and space of 0.84 um.

     LOAD\_M:  15.34

     D\_TEST: 17.02

     CON\_DATA: 18.7

 READ: 19.54

 FASTOR: 20.38

 ADDR<0>: 21.22

 ADDR<1>: 22.06

 ADDR<2>: 22.9

 ADDR<3>: 23.74

 ADDR<4>: 24.58

 ADDR<5>: 25.42

 ADDR<6>: 26.26

 ADDR<7>: 27.1

 ADDR<8>: 27.94

 ADDR<9>: 28.78

 APLSE: 29.62

 CON\_CLK: 30.46

 G\_RST: 32.14

 LOAD\_C:  33.82