TMC-PHX1 Architecture Manual

(TC180G70xx-0102: final design Feb. 26, 1998)

Yasuo Arai

KEK, National High Energy Accelerator Research Organization
1-1 Oho, Tsukuba, Ibaraki 305, Japan
yasuo.arai@kek.jp, http://www-atlas.kek.jp/~araiy/
Tel : +81-298-64-5366
Fax : +81-298-64-2580

Rev 0.7b June. 3, 1998

1. INTRODUCTION 2

2. CIRCUIT DESCRIPTION 4

2.1. Fine Time Measurement.....
2.2. Data Recording.....
2.3. PLL circuit.....
2.4. Encoder.....
2.5. CSR Registers.....
2.5.1. CSR0 : Offset Register.....
2.5.2. CSR1 : Read Pointer Register.....
2.5.3. CSR2 : Write Pointer Register.....
2.5.4. CSR3 : Count Preset Register.....
2.5.5. CSR4 : Event Number control.....
2.5.6. CSR5 : Miscellaneous control.....
2.5.7. CSR6 : Test register.....
2.5.8. CSR7 : Test Data Register.....
2.6. Data Transfer within the chip.....
2.7. Data Readout.....
2.8. Data Format.....
2.8.1. Header.....
2.8.2. Fixed Data Length mode.....
2.8.3. Zero Suppress Mode (ZSUPP=1).....
2.9. Data Output Sequence.....
2.10. Trigger Interface.....
2.11. Chip Test.....
2.11.1. Shift data in/out to/from the synchronization latch.....
2.11.2. Read encoded data.....
2.11.3. Write data to a dual port memory.....
2.11.4. Read data directly from the dual port memory.....
2.11.5. Check the trigger FIFO.....
2.11.6. Check the readout FIFO.....

3. PERFORMANCE.....24

3.1. Time Resolution.....
3.2. Differential and Integral nonlinearity.....2
3.3. PLL Tracking Range and stability.....
3.4. PLL Step Response.....
3.5. Power consumption.....

(This document describes about architecture and functions of TMC-PHX1 chip. Please also read "TMC-PHX1 Electrical Specification" document for electrical and physical specification.)

1. Introduction

TMC-PHX1 is a low-power and high-resolution multi-hit Time-to-Digital Converter LSI produced with Toshiba's 0.5 μm CMOS Sea-of-Gate Technology. The architecture is optimized for the use in the PHENIX experiment at RHIC accelerator in BNL.

The idea of TMC chip was born in 1986 [1], and several kinds of chips [2, 3, 4] and modules [5, 6] were developed. This TMC-PHX1 has most complicated architecture compared with previous chips.

Input signals are digitized at TMC (Time Memory Cell) circuits in (clock period)/32 (0.78 ns @40MHz) time bin. The 32 bits output of the TMC are divided into two 16 bits data, and each 16 bits data are encoded into 5-bit data. The data are stored in a 256 words deep dual port memory. The write and read operation can be performed at the same time and there is no dead time of data conversion. The dual port memories can store data of 6.4 μs period at 40 MHz clock. In addition, there are 128 words readout FIFO's in each channel to derandomize the data readout rate.

TMC-PHX1 chip has implemented 4 channels in a chip. There is also 8 word depth trigger FIFO which enables multiple triggers and overlapped trigger.

To stabilize the internal delay element, PLL circuit is used. User can select 10 - 50 MHz system clock (x1 mode) or 2.5 - 12.5 MHz clock (x4 mode). There are 8 Control and Status Registers (CSR's) which can be read and write through an 8-bit control bus. The package of the chip is 100 pins QFP with 0.8 mm lead pitch.

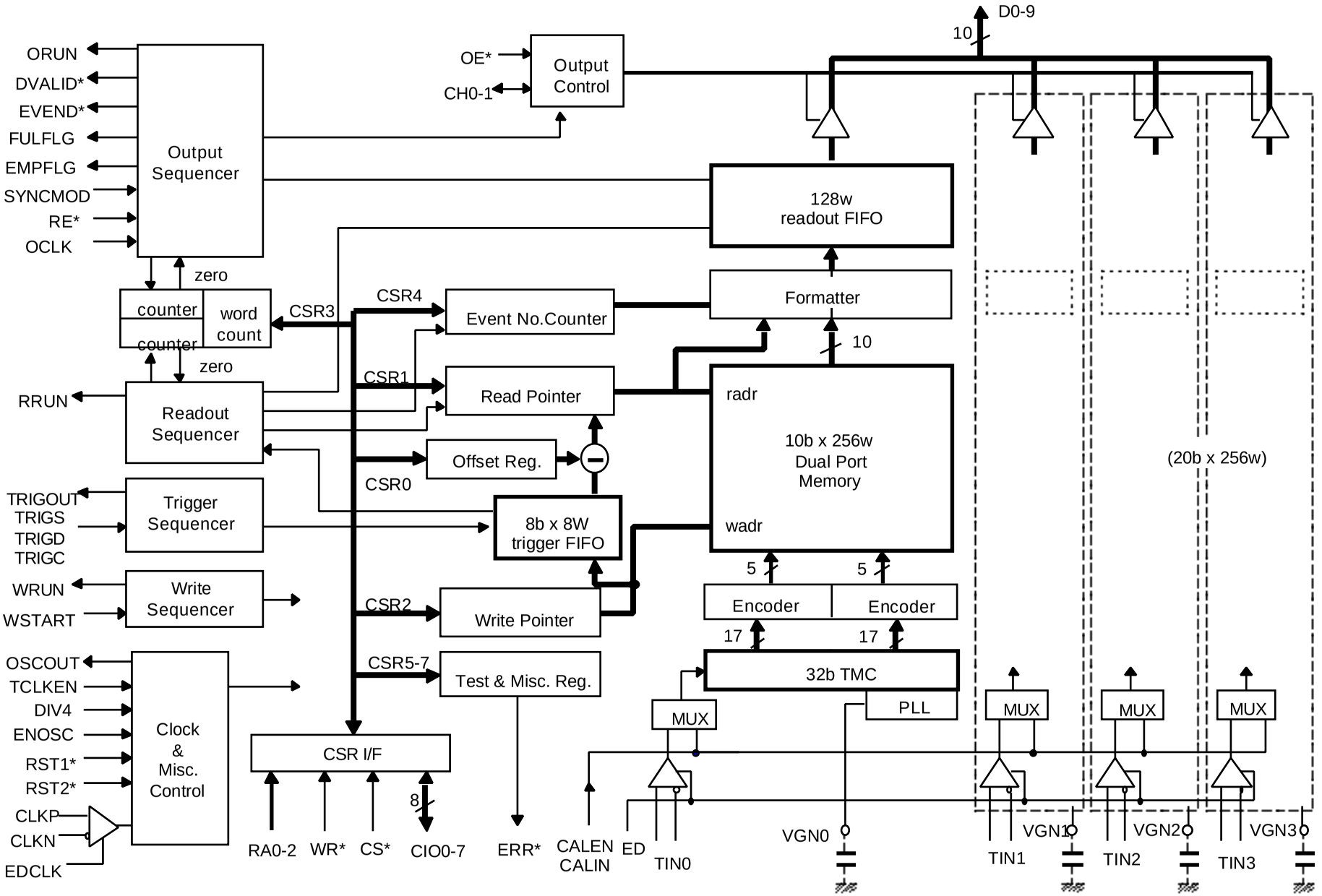

ORUN
DVALID*
EVEND*
FULFLG
EMPFLG
SYNCMOD
RE*
OCLK
Output Sequencer
OE*
CH0-1
Output Control
D0-9
10
128w readout FIFO
Formatter
10
zero
counter
word count
counter
zero
CSR3
CSR4
Event No.Counter
CSR1
Read Pointer
RRUN
Readout Sequencer
Offset Reg.
CSR0
radr
10b x 256w Dual Port Memory
wadr
(20b x 256w)
TRIGOUT
TRIGS
TRIGD
TRIGC
Trigger Sequencer
8b x 8W trigger FIFO
5
5
Encoder
Encoder
WRUN
WSTART
Write Sequencer
CSR2
Write Pointer
17
17
32b TMC
PLL
OSCOUT
TCLKEN
DIV4
ENOSC
RST1*
RST2*
CLKP
CLKN
EDCLK
Clock & Misc. Control
CSR5-7
Test & Misc. Reg.
MUX
MUX
MUX
MUX
CSR I/F
8
RA0-2
WR*
CS*
CIO0-7
ERR*
CALEN
CALIN
ED
TIN0
VGN0
TIN1
VGN1
TIN2
VGN2
TIN3
VGN3

Fig. 1-1 Block diagram of the TMC-PHX1

2. Circuit Description

Fig. 1-1 shows a block diagram of the TMC-PHX1 chip. There are 4 input channels in the chip. Each channel has a TMC/ring oscillator circuit, a PLL circuit, two sets of edge detection encoders, a 256 word dual-port memory (DPM), 128 word readout FIFO (RFIFO). A 20-bit wide dual-port memory is shared by 2 channels as 10 bits each. There are 8 control and status registers (CSR's) and 8 word trigger FIFO in a chip. To receive multiple triggers before data readout, there is a 8 word depth trigger FIFO (TFIFO). An entire event is reconstructed for readout even for an overlapped data which are shared in two triggers.

Input signals for a system clock and time measurement can be either single-ended or differential signals. Table. 1 summarizes the main features of the TMC-PHX1 chip.

Table. 1 TMC-PHX1 MAIN FEATURES

• Least Time Count	0.6 - 3.1 ns/bit (propotional to clock period)
• Time Resolution	RMS = 250 ps (rising edge @40 MHz)
	RMS = 350 ps (falling edge @40 MHz)
• Integral Linearity Error	< 80 ps @40MHz
• Differential Linearity Error	< 60 ps @40MHz
• Stability	< 0.1 LSB (3.0 - 3.6 V. 0 - 70 ˚C)
• System Clock Frequency	10 - 40 MHz (x1 mode)
	2.5 - 10 MHz (x4 mode)
• No. of Channels	4 Channels
• Level 1Buffer	256 words (6.4 μs @40 MHz)
• Double Hit Resolution	<12.5 ns (@40 MHz)
• Trigger FIFO	8 words
• Readout mode	Synchronous or Asynchronous
• Data format	Header (optional) + 5 bit encoding x 2 (rising and falling edge) Zero Suppress mode (optional) Shared data between two triggers are copied.
• Readout FIFO	128 words
• Supply Voltage	3.3 ~ 3.8V
• Process	0.5 μm CMOS Sea-of-Gate
• Power Dissipation	~ ? mW/Channnel
• Package	0.8 mm lead pitch, 100 pin plastic QFP

2.1. Fine Time Measurement

The idea of time memory cell (TMC) was born in 1986[1], and it utilizes a voltage-controlled delay elements which comprise ring oscillator and memories. The ring oscillation is controlled with Phase Locked Loop (PLL) to attain high precision.

The ring oscilltor has 25 ns cycle period for 40 Mhz system clock. To obtaine <1 ns timing resolution, 32 taps are implemented in the oscillator. Fig. 2-1 shows a simplified schematics and its timing diagram of the asymmetric ring oscillator. Fig. 2-1 only shows 8 stages but the actual chip implements 32 stages. The asymmetric ring oscilator was invented to get equally spaced even number (32) of timing signals.

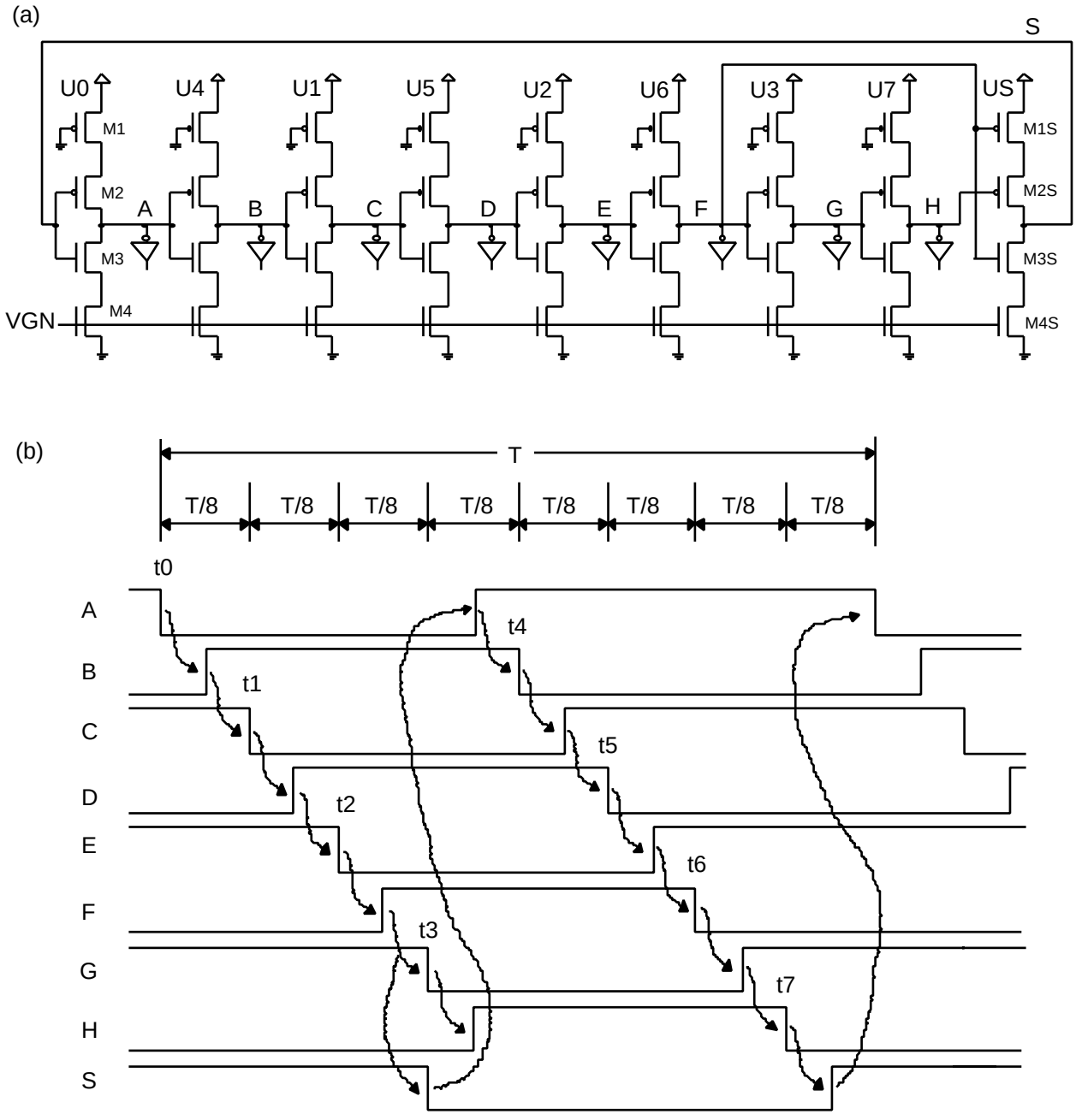


Fig. 2-1 (a) Asymmetric ring oscillator, (b) extracted timing signal.

2.2. Data Recording

By using the fine timing signal derived from the asymmetric ring oscillator, input signal is digitized and stored in memories continuously. Fig. 2-2 shows the data recording scheme of the TMC circuit.

In each cycle, 32 bits plus one additional bit from the next cycle is stored in a synchronization latch. There are two encoders to detect second edge which arrive within a cycle. For each 17 bits, data transition point is encoded into 5 bit data as shown in Table. 2. Then the encoded data are stored in a dual-port memory. The dual port memory works like a psudo FIFO with write and read pointers.

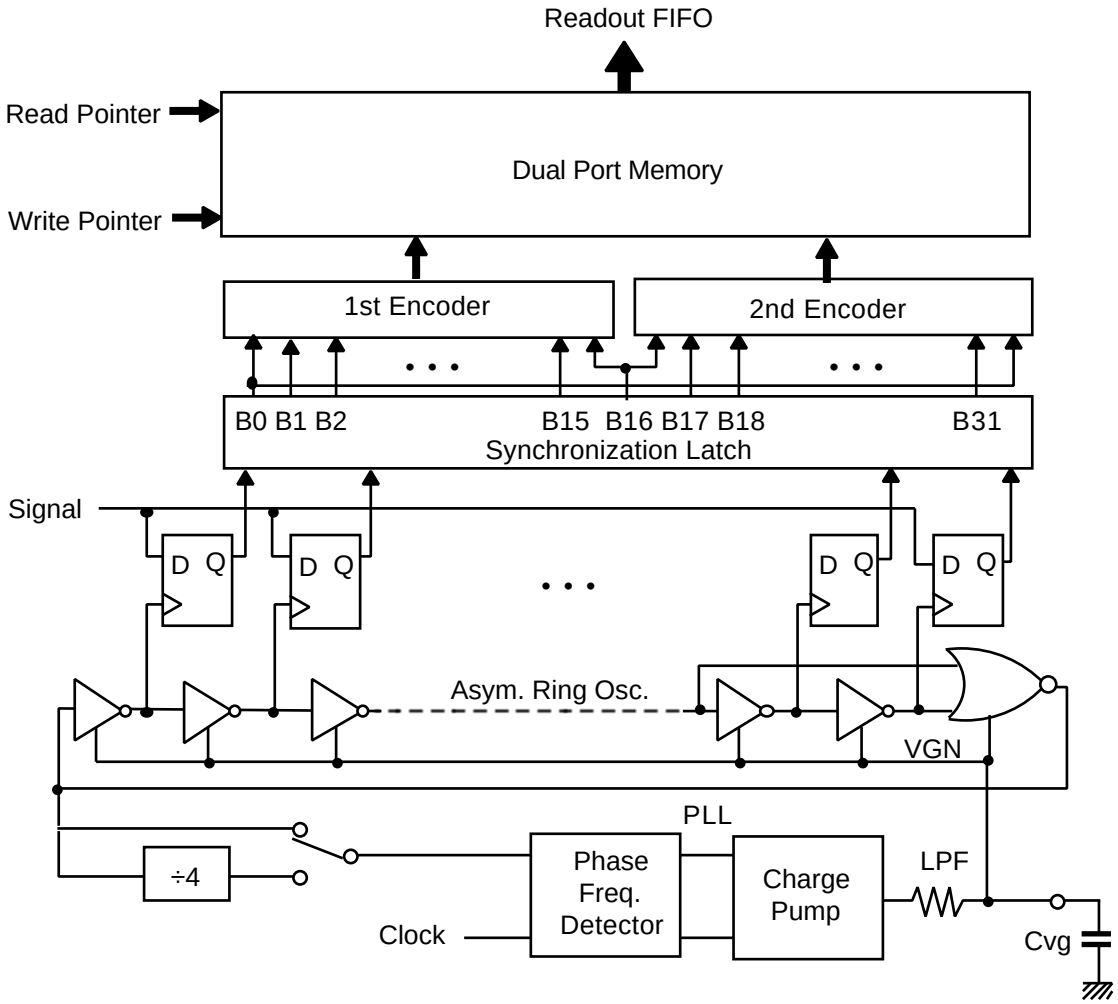


Fig. 2-2 Data recording scheme.

2.3. PLL circuit

The PLL circuit (see Fig. 2-2) comprises a phase frequency detector (PFD), a charge pump, a loop filter (LPF), and a voltage-controlled oscillator (VCO; asymmetric ring oscillator in this case). An external capacitor (Cvg) is required in the loop filter. Optionally a "divide by 4" counter can be

selected after the VCO, thus the frequency of the VCO can be either the same or the “multiplied by 4” of the input frequency.

The propagation delay of the delay elements that determine the oscillation frequency of the VCO is controlled through a control voltage (VGN).

2.4. Encoder

The encoder encode a transition point in 17 bits data to 5 bit code as shown in Table. 2. If there is a 0 to 1 (rising edeg) transitions in the obtained 17 bits, the first transition point is encoded into 16 different codes. Other transitions after the first edge are ignored.

If there is no rising edge and only falling edge exists in the 17 bits, the falling edge position is encoded into 8 different codes. Thus the falling edge time has less resolution. For the wire chamber applications, the falling edge timing is worse than the rising edge timing, so this does not cause problem.

Not to miss the signal transition at the boundary of encoders, 16 bits plus one additional bit is provided to the encoder . Furthermore, "all 0", "all 1", “Header ID”, “End of Data” and “Data Error” conditions are represented by using a spare code.

Table. 2 Data Encoding Scheme

row data bit											Encoded Data	Comment
0	1	2	3	4	5		13	14	15	16		
(16	17	18	19	20	21		29	30	31	next 0)	4 3 2 1 0	
0	0	0	0	0	0	•	0	0	0	0	0 0 0 0 0	all 0
1	1	1	1	1	1	•	1	1	1	1	0 0 0 0 1	all 1
											0 0 0 1 0	Header ID
											0 0 0 1 1	not used
											0 0 1 0 0	not used
											0 0 1 0 1	not used
											0 0 1 1 0	End of Data
											0 0 1 1 1	Data ERROR
1	x	0	0	0	0	•	0	0	0	0	0 1 0 0 0	F edge between bit 0-2
1	1	1	x	0	0	•	0	0	0	0	0 1 0 0 1	F edge between bit 2-4
1	1	1	1	1	x	•	0	0	0	0	0 1 0 1 0	F edge between bit 4-6
1	1	1	1	1	1	•	0	0	0	0	0 1 0 1 1	F edge between bit 6-8
1	1	1	1	1	1	•	0	0	0	0	0 1 1 0 0	F edge between bit 8-10
1	1	1	1	1	1	•	0	0	0	0	0 1 1 0 1	F edge between bit 10-12
1	1	1	1	1	1	•	x	0	0	0	0 1 1 1 0	F edge between bit 12-14
1	1	1	1	1	1	•	1	1	x	0	0 1 1 1 1	F edge between bit 14-16
0	1	x	x	x	x	•	x	x	x	x	1 0 0 0 0	R edge between bit 0-1
0	0	1	x	x	x	•	x	x	x	x	1 0 0 0 1	R edge between bit 1-2
0	0	0	1	x	x	•	x	x	x	x	1 0 0 1 0	R edge between bit 2-3
:	:	:	:	:	:	:	:	:	:	:	:	
0	0	0	0	0	0	•	0	0	1	x	1 1 1 1 0	R edge between bit 14-15
0	0	0	0	0	0	•	0	0	0	1	1 1 1 1 1	R edge between bit 15-16

x = don’t care.

2.5.CSR Registers

There are 8 CSR registers and the bit assignment is shown in Table. 3.

Table. 3 CSR registers bit assignment

bit	7	6	5	4	3	2	1	0
CSR0	Offset Register [OFR(7:0)] (R/W:128)							
CSR1	Read Pointer Register [RPR(7:0)] (R/(W):0)							
CSR2	Write Pointer Register [WPR(7:0)] (R/W:0)							
CSR3	Count Preset Register [CPR(7:0)] (R/W:20)							
CSR4	Event Number Register [ENR(7:0)] (R/W:0)							
CSR5	DPMERR (R/W:0)	OFIFUL (R/W:0)	TFIFUL (R/W:0)	NOHEAD (R/W:0)	ZSUPP (R/W:0)	BYPFIFO (R/W:0)	ENWPUP (R/W:0)	ENTDR (R/W:0)
CSR6	SOUT (R:0)	SIN (R/W:0)	SHIFT (R/W:0)	Chip ID Register [CID(4:0)] (R/W:0)				
CSR7 ENTDR=0	SELH (R/W:0)	TCH1 (R/W:0)	TCH0 (R/W:0)	ENOSCO (R/W:0)	ENCH3 (R/W:1)	ENCH2 (R/W:1)	ENCH1 (R/W:1)	ENCH0 (R/W:1)
ENTDR=1				Test Data Register [TDR(4:0)] (R/W:0)				

(Readable/Writable : Initial value)

2.5.1. CSR0 : Offset Register

- OFR [read/write] : This offset value is used to calculate the read pointer (RPR) value when data transfer from the DPM to RFIFO is necessary. Conceptually, this corresponds to trigger latency. When a trigger signal is received, present write pointer value is stored in the TFIFO. Then the TFIFO value minus this offset value is set to the RPR to read relevant data from the DPM.
  The first data will be read out from the address of 'WPR - OFR +1' (the Header 2 will be 'WPR - OFR' (see section 2.8.1)).

2.5.2. CSR1 : Read Pointer Register

- RPR [read]: Read pointer is a 8-bit up counter which outputs are a addresses of the read port of the Dual Port Memory (DPM). User can read/write this register to test the DPM memory, but in a measurement, this register is automatically set to address to which trigger is occurred.

2.5.3. CSR2 : Write Pointer Register

- WPR [read/write]: Write pointer is a 8-bit up counter which outputs are a addresses of the write port of the Dual Port Memory (DPM). The write pointer is set through this register and read back present value of the write pointer. During measurement, this register is automatically incremented in each cycle.

2.5.4. CSR3 : Count Preset Register

- CPR [read/write]: This register contains number of data (excluding header) to be written to the output FIFO when a trigger signal is received.

2.5.5. CSR4 : Event Number control

- ENR [read/write]: Event Number Register. This 8 bits register keeps a event number. The event number is written to the header part of the data, and incremented by each trigger.

2.5.6. CSR5 : Miscellaneous control.

- ENTDR [read/write] : Enable Test Data Register. Used only in test mode. See section 2.11.
- ENWPUP [read/write] : Enable Write Pointer count up in test mode. See section 2.11.
- BYPFIFO [read/write] : Bypass Output FIFO. Used only in test mode. See section 2.11.
- ZSUPP [read/write] : Zero suppression mode. When this bit is set, only the data which include Rising or Falling edge are transferred to the RFIFO.
- NOHEAD [read/write] : No header insertion mode. When this bit is set, the header is not added to the data.
  (CAUTION: At present, this function does not work properly in the synchronous readout mode.)
- TFIFUL [read/write] : Trigger FIFO full flag. This bit is set when the trigger FIFO becomes full. This bit is not cleared automatically even when a new location become available in the TFIFO. Cleared by RST1* or write ‘0’ to this bit.
- OFIFUL [read/write] : Readout FIFO full flag. This bit is set when the output FIFO becomes full. This bit is not cleared automatically even when a new location become available in the OFIFO. Cleared by RST1* or write ‘0’ to this bit.
- DPMERR [read/write] : DPM buffer overflow flag. This bit is set when the write pointer approaches to the read pointer. The data transfer to the RFIFO will stop by writing data error code to prevent the overwrite to triggered data area.
  This situation may occur only when large CPR is set, very high trigger rate, and slow readout rate.

2.5.7. CSR6 : Test register.

- CID [read/write] : Chip ID. This chip ID is written to the header part of the data.
- SHIFT [read/write] : Enable Shift In/Out. Used only in test mode. See section 2.11.
- SIN [read/write] : Serial Input bit. Used only in test mode. See section 2.11.
- SOUT [read only] : Serial output bit. Used only in test mode. See section 2.11.

2.5.8. CSR7 : Test Data Register.

- ENCH0~3 [read/write] : Enable Channel input. When this bit is cleared, corresponding data channel becomes always zero.

- ENOSCO [read/write] : Enable OSCOUT output. When this bit is set, PLL oscillation output selected by TCH bits is available from OSCOUT. This may useful for checking PLL stability, but it is recommended to set to ‘0’ (default) not to dissipate additional power and generate noises.
- TCH0~1 [read/write]: Test channel select. TCH = [TCH1,TCH0]. Used in selecting PLL oscillation channel and test mode. See section 2.11.
- SELH [read/write]: Select lower or higher part of test data. Used only in test mode. See section 2.11.
- TDR [read/write]: Test Data Register (bit 4 - 0). Used only in test mode. See section 2.11.

2.6.Data Transfer within the chip

Main data transffer scheme is shown in Fig. 2-3, and the sequence is shown below;

- By setting ENOSC=H, the ring oscillator will start to oscillate and input signal is digitized and encoded at every cycle.
- When WSTART signal is asserted, the encoded data are began to be written to the dual port memory which address is pointed by the Write Pointer.
- When trigger signal is received, the present write pointer value is stored in the trigger FIFO.
- When the trigger FIFO is not empty, the read pointer is automatically set to the value of the trigger FIFO data minus offset value stored in a Offset register. This offset corresponds to trigger latency.
- Then header information and data pointed by the read pointer is transfferd to the readout FIFO. Optionally, zero suppression is also done.
- Number of data transfferd from the DPM to the RFIFO is set in the Count Preset register.

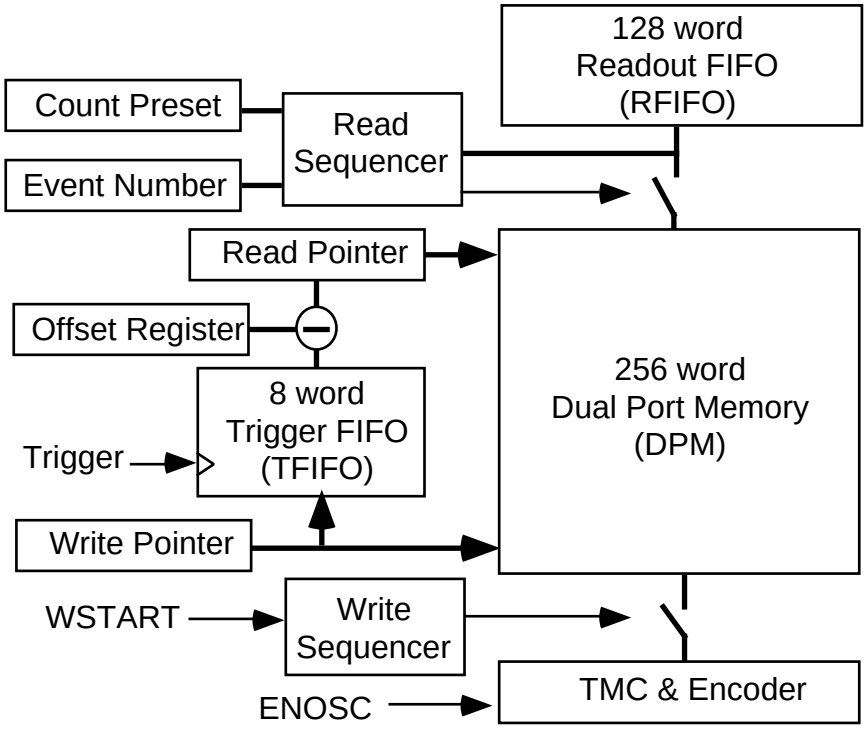


Fig. 2-3. Data trasnsfer scheme.

2.7. Data Readout

The data in the RFIFO are readout through a internal common bus. The output buffer is tri-state buffer and controlled with OE*(output enable*).

There are two modes of readout. One is asynchronous and other is synchronous readout mode. In asynchronous mode (SYNCMOD=L), the readout channel is selected by CH0 and CH1 signals, and the readout timing is controlled with RE* (read enable*) signal.

In synchronous mode (SYNCMOD=H), the readout is proceed with sequencially from channel 0 trough channel 4. The timing is synchronous with OCLK (output clock) and controlled with RE* signal. Channel number of the readout data is available from CH0 and CH1 signals.

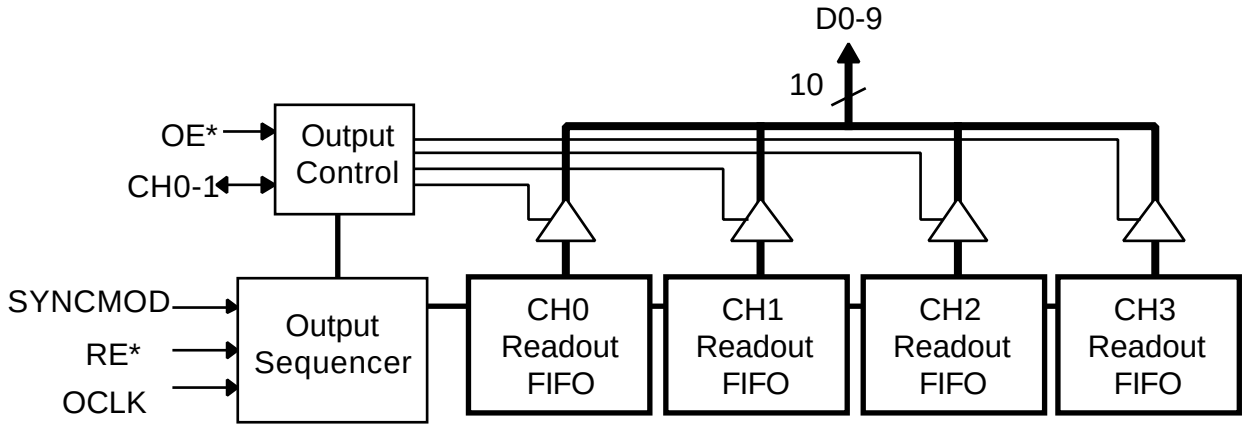


Fig. 2-4. Readout Circuit.

2.8. Data Format

2.8.1. Header

When NOHEAD flag, which is bit 4 of CSR5, is set to ‘0’ (default), header words precede the data. The contents of the header are shown in Table. 4. Upper part of the header 0 is always ‘0 0 0 1 0’ which denotes header (see Table. 2). Chip ID is the value written in CID of CSR6.

Header 1 contains event number which is incremented by 1 for each event and set in CSR4 (ENR). Header 2 contains the read pointer value of the first data minus one (RP - 1), that corresponds to (WPR - OFR). This is only used for debugging purpose. Header 3 shows the data length, that is, content of CSR3 (CPR).

(CAUTION: At present, No Header mode does not work properly in the synchronous readout mode.)

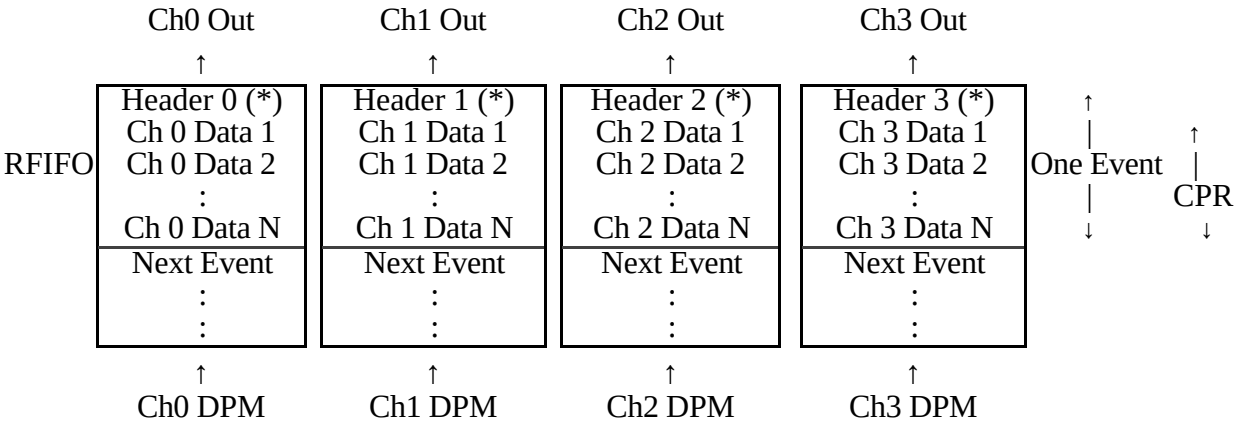
Table. 4. Contents of header words

	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
Header 0	0	0	0	1	0	Chip ID (4:0)				
Header 1	0	1	Event Number (ENR)							
Header 2	1	0	First data address - 1							
Header 3	1	1	Data Length N (=CPR)							

2.8.2. Fixed Data Length mode

In fix data length mode (ZSUPP=0), fixed length of data is transferred from the DPM to the RFIFO. The data format in the RFIFO is shown in Table. 5, and output data sequence in synchronous mode is shown in Table. 6. Each data word contains 1st and 2nd encoder output data as shown in Table. 7.

Table. 5. Data format in the readout FIFO



(*) Inserted when NOHEAD=0.

Table. 6. Output Data sequence in synchronous mode.

CH signal	D9 . . . D0	
0	Header 0 (*)	↑
1	Header 1 (*)	|
2	Header 2 (*)	|
3	Header 3 (*)	|
0	Ch 0 Data 1	|
0	Ch 0 Data 2	One Event
0	:	|
0	Ch 0 Data N	|
1	Ch 1 Data 1	|
1	Ch 1 Data 2	|
1	:	|
1	Ch 1 Data N	|
2	Ch 2 Data 1	|
2	Ch 2 Data 2	|
2	:	|
2	Ch 2 Data N	|
3	Ch 3 Data 1	|
3	Ch 3 Data 2	|
3	:	|
3	Ch 3 Data N	↓

Table. 7. Content of data word

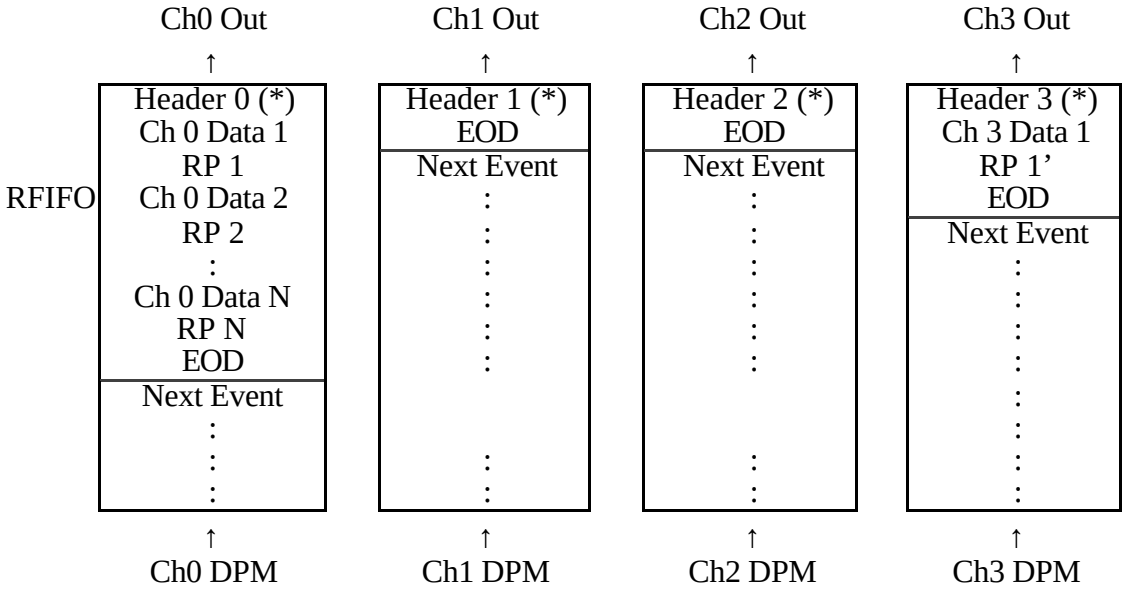
D9 D8 D7 D6 D5	D4 D3 D2 D1 D0
Upper Encoder data (Late Time)	Lower Encoder data (Early Time)

2.8.3. Zero Suppress Mode (ZSUPP=1)

In zero suppression mode (ZSUPP=0), only non-zero data are transferred from the DPM to the RFIFO. Since the encoded data contains the information in a cycle, coarse time value which is derived from the read pointer is added to the data. An example of data format in the RFIFO is shown in Table. 8, and output data sequence in synchronous mode is shown in Table. 9. If you

want to access the data in RFIFO in asynchronous mode, you must check the End of Data (EOD) mark for each channel. Each data word contains read pointer value and encoded data as shown in Table. 10.

Table. 8. An Example of data format in the readout FIFO (Contents depend on data)



(*) Inserted when NOHEAD=0.

Table. 9. Output Data sequence for above example in synchronous mode.

CH signal	D9 D8 D7 D6 D5 D4 D3 D2 D1 D0		
0	Header 0 (*)		^
1	Header 1 (*)		|
2	Header 2 (*)		|
3	Header 3 (*)		|
0	Ch0 Data 1		|
0	0 0 | RP 1		|
0	Ch0 Data 2		|
0	0 0 | RP 2		|
:	:		One Event
:	:		|
0	Ch0 Data N		|
0	0 0 | RP N		|
3	Ch3 Data 1		|
3	1 1 | RP 1		|
0	0 0 1 1 0 | 0 0 1 1 0	End of Data	V

Table. 10. Contents of data words

D9 D8 D7 D6 D5	D4 D3 D2 D1 D0	
Data High (Late Time)	Data Low (Early Time)	Fine Time
Ch No.	RP value	Coarse Time

2.9.Data Output Sequence

In the synchronous mode, data and channel number will appear in output pins controlled with a output sequencer. Simplified state diagrams of the sequencer are shown in Fig. 2-5, and examples of data output sequence are shown in Fig. 2-6.

When “Zero Suppression” is OFF(ZSUPP=0), the data output sequence is very simple. At first, 4 header words are transferred if the NOHEAD =0, then data will be output from channel 0 to channel 3. For each channel, number of data set in the CPR register will be output. In the example of Fig. 2-6 -(a), CPR is set to 2.

When “Zero Suppression” is ON (ZSUPP=1), the data output sequence is a little bit complex and depend on the contents of data. After transferring the header words(NOHEAD=0), channel number is incremented in “NEWCH” cycle, then the contents of data is checked in “CHECK” cycle. If valid data exist, the data and its Read Pointer value (RP) are transfered. Thus DVALID* signal is missing in (i) first cycle of each channel, (ii) first cycle of each data. In the example of Fig. 2-6 -(b), channel 0 has two valid data, channel 3 has one valid data, and channel 1 and 2 has no data.

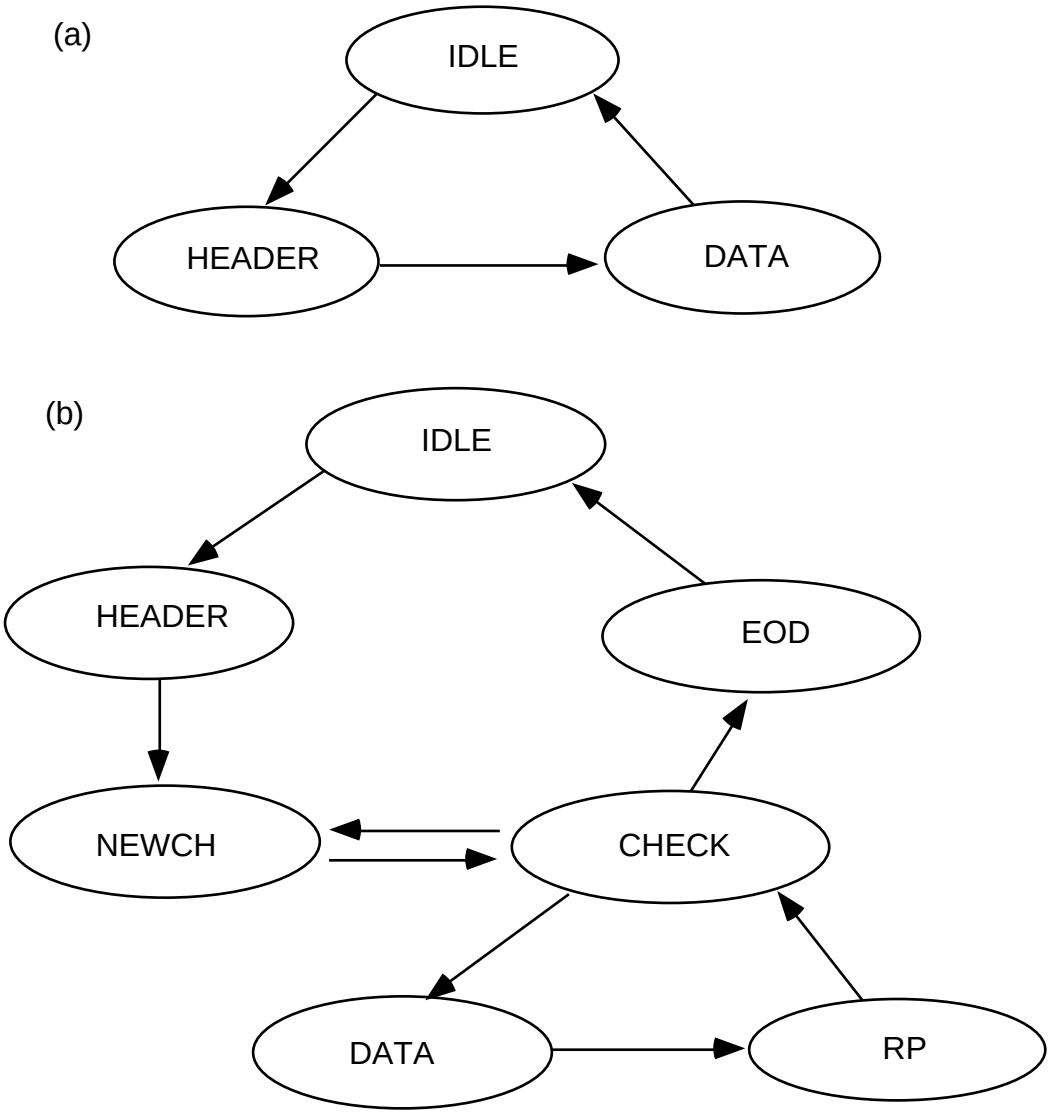


Fig. 2-5 Simplified state diagram of the output sequencer. (a) ZSUPP=0, (b) ZSUPP=1.

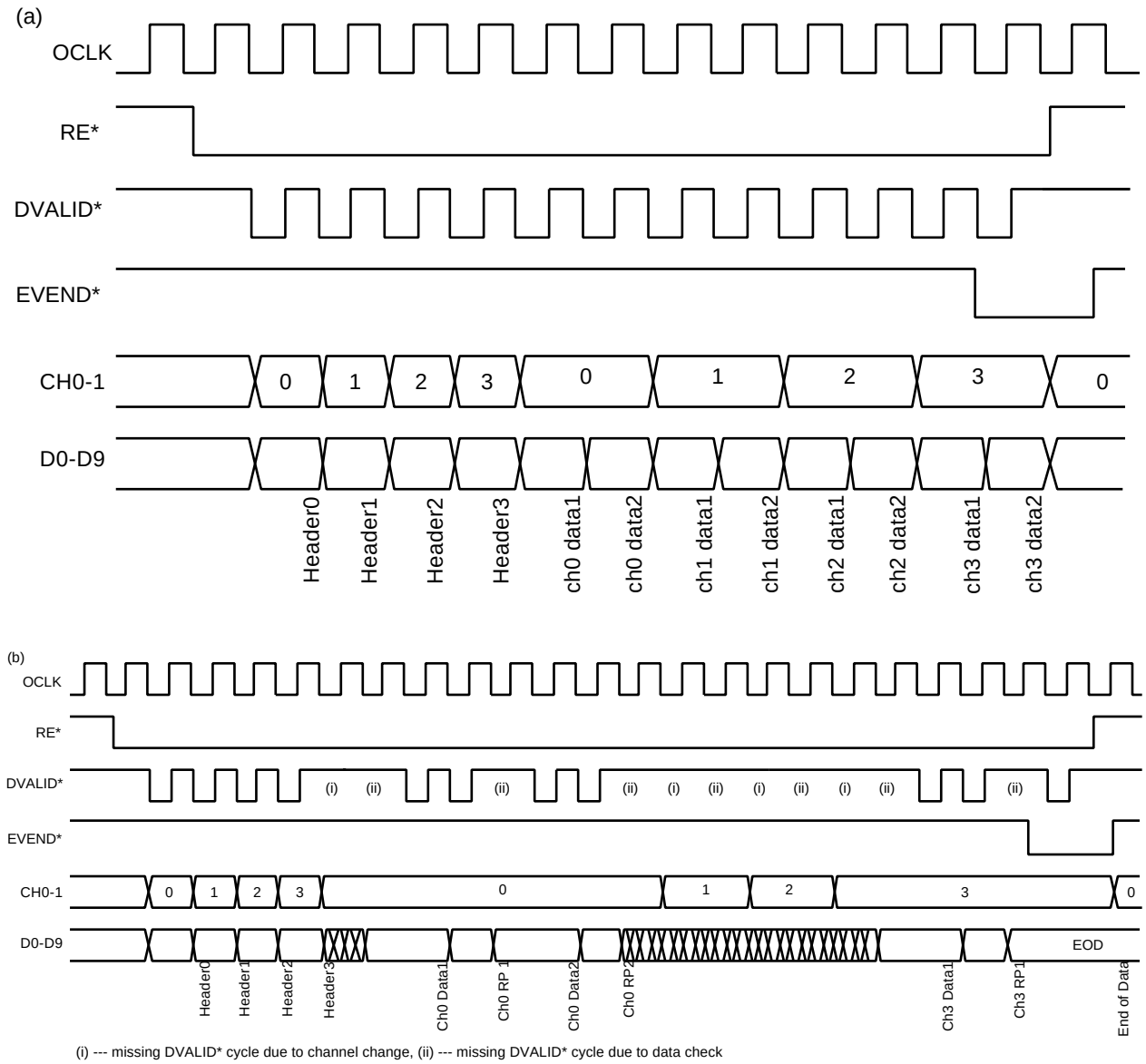


Fig. 2-6 Examples of output data sequence. (a) ZSUPP=0, (b) ZSUPP=1.

2.10.Trigger Interface

Trigger interface is a interface which connect between external trigger system and the TMC circuit. Careful attention must be paid to this connection. A simplified schematic of the interface is shown in Fig. 2-7.

Three input signals of a first flip-flop (FF1) is extracted to input pins. User can configure desired interface by using these pins. Output of the FF1 is synchronized to system clock (~40MHz) with two flip-flop (FF2 and FF3). After internal trigger pulse is created, the FF1 is cleared. The state of the FF1 is available from TRIGOUT pin.

Minimum trigger signal separation is two system clock cycle. Thus only one trigger is

accepted, even there are more than two triggers within two clock cycle,

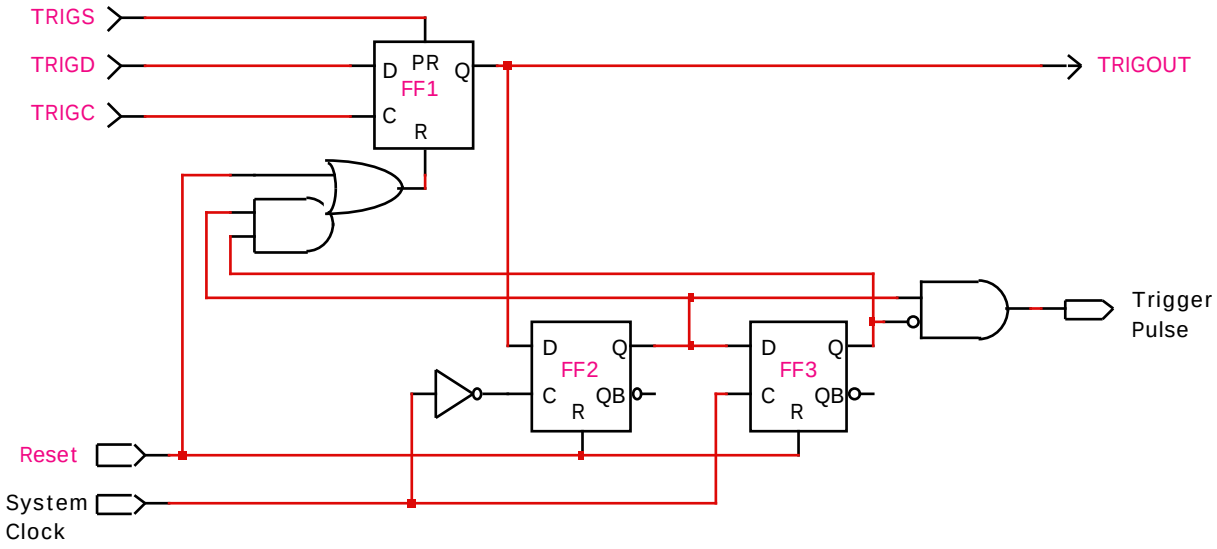


Fig. 2-7 Simplified schematic of the trigger interface. When both PR and R inputs are high level in the FF1, the output Q will be low level.

Three examples of trigger signal connection are shown in Fig. 2-8 and its timing sequence is shown in Fig. 2-9.

(a) pulse trigger : By using “PRESET” input of the FF1, user can use pulse signal. If the width of trigger signal is longer than 2 clock cycle, another trigger signal will be generated.

(b) clock trigger :A trigger pulse is applied to clock input of the FF1. In this case, only one trigger is accept for one trigger pulse regardless of the pulse width.

(c) level trigger : A trigger system clock, which can be different frequency from the system clock, is applied to clock input of the FF1. Then the trigger is controlled with the signal in the “D” input.

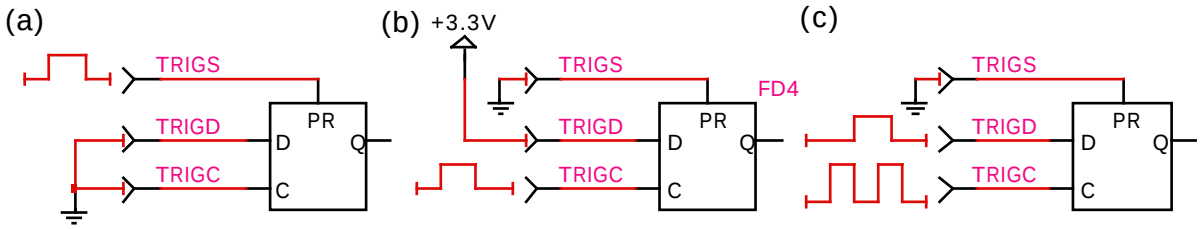


Fig. 2-8. Examples of trigger signal connection. (a) pulse trigger, (b) clock trigger, and (c) level trigger.

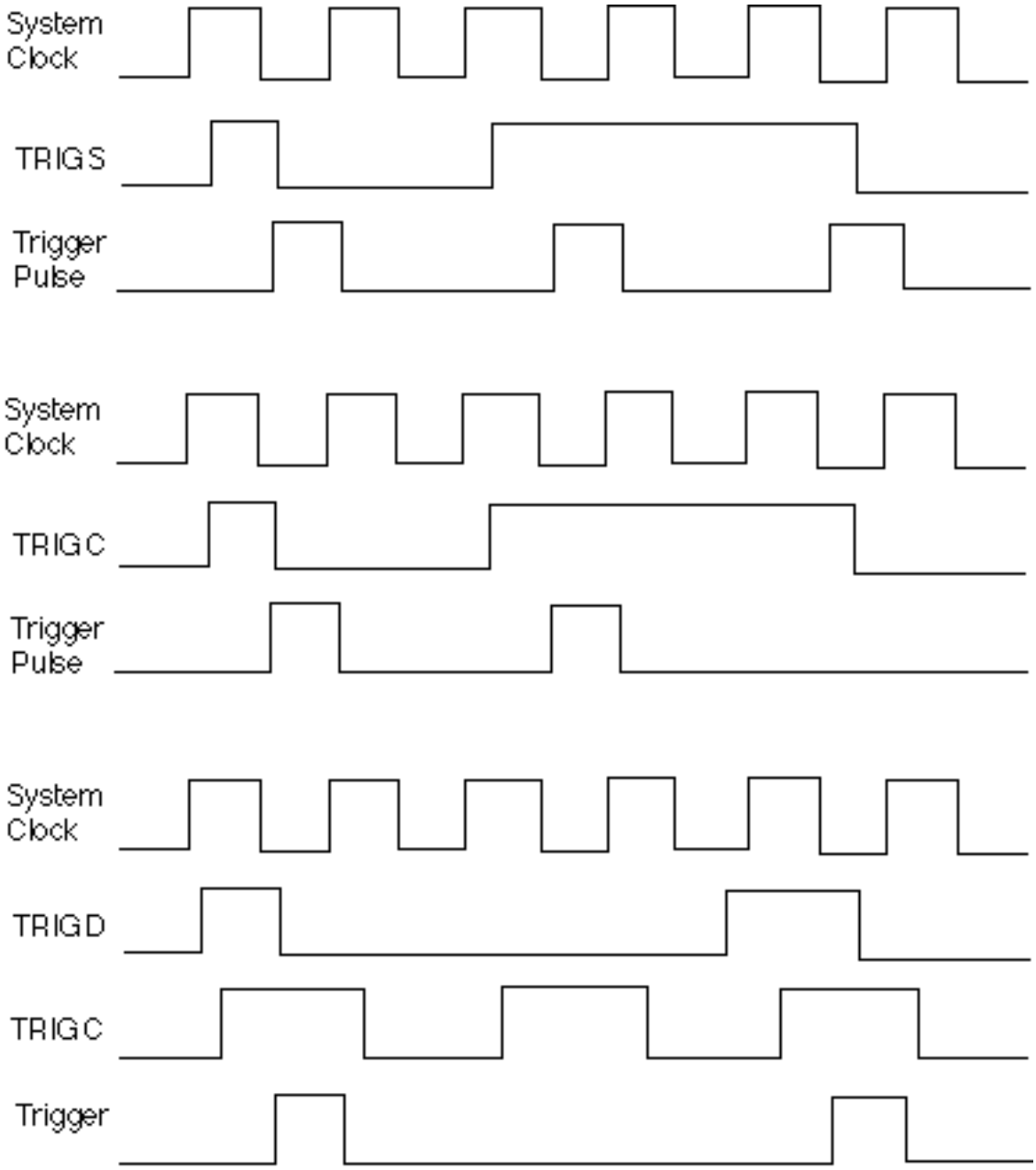


Fig. 2-9. Examples of trigger timing for the connection shown in Fig. 2-8.

2.11. Chip Test

This section describe about the methods for testing logic's, memories etc. of the TMC chip. These methods are mainly used for chip debugging and for LSI test at fabrication. Normally you don't need to care about these, although these method may help some type of system test. Fig. 2-10 shows generic view of the test paths used in these tests.

When ENTDR is set, DPM location selected by TCH and WPR is written TDR in data after the TDR write. At the same time, if BYPFIFO is not set, the content of DPM pointed by RPR is written to RFIFO. If BYPFIFO is set, this writing does not occur.

TDR out is always connected to one of the Encoder output selected by TCH and SELH.

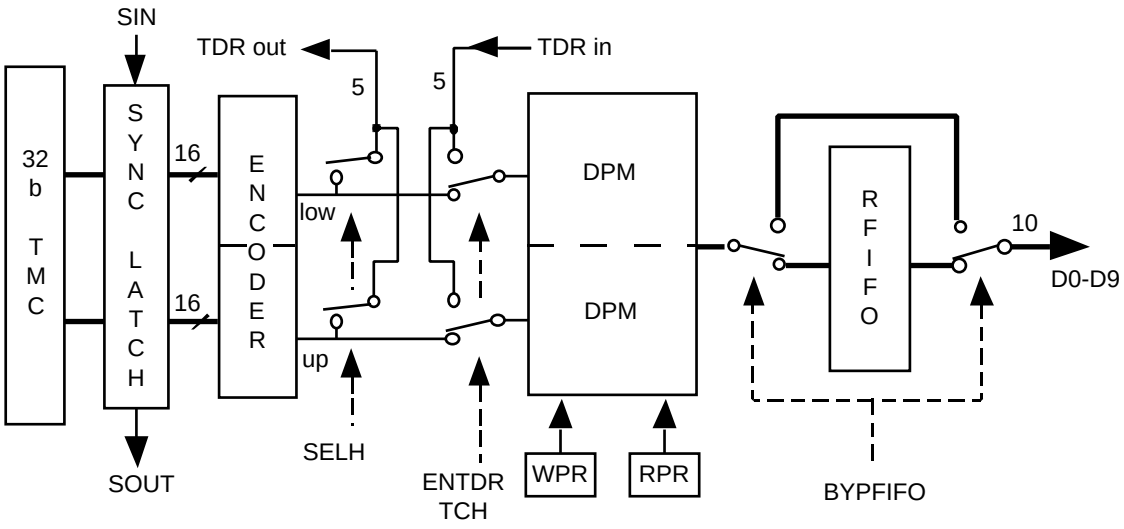


Fig. 2-10. Generic view of test paths of a channel.

Table 2-1 Test path selection. Each item is wriiten to corresponding location.

ENTDR	BYPFIFO	DPM input	RFIFO input	D0-D9
0	x	Encoder	DPM	RFIFO
1	0	Ch A low = TDR in Ch A up = TDR in other Ch = Encoder	DPM	RFIFO
1	1	ditto	(not written)	DPM

(*) TCH = Channel A

Table 2-2 TDR out selection.

SELH	TDR out
0	Channel A Encoder low
1	Channel A Encoder up

(*) TCH = Channel A

2.11.1. Shift data in/out to/from the synchronization latch

You can shift in arbitrary data pattern to the synchronization latch or shift out data pattern stored in the synchronization latch. A simplified schematic diagram is shown in Fig. 2-11. Shift procedures of the latch are as follows;

(i) Set desired channel number to [TCH1,TCH0] of the CSR7.

(ii) write Shift In data to SIN(CSR6[6]) with SHIFT flag (CSR6[5]).

(iii) After writing the CSR6 (negate CS*), SIN is shifted into the first flip/flop (bit 0) of the synchronization latch, and other bit is shifted to higher bit.

You can always read out the value of bit 31 through SOUT (CSR6[7]).

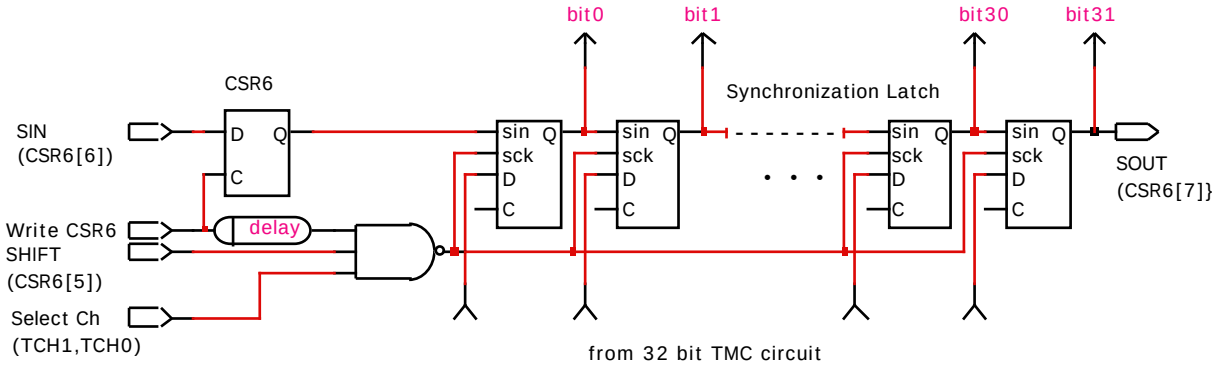


Fig. 2-11 Simplified scheme of the synchronization latch shift circuit.

2.11.2. Read encoded data

You can read encoder output by using a Test Data Register (TDR) as shown in Fig. 2-10. By combining the synchronization latch, you can verify the function of the encoder. To read a specific part of encoder output, you need to set SELH, TCH1, TCH0 bits of CSR7 as shown in Table 2-3, and set ENTDR (CSR5[1]) bit. Then you can read the encoder output through the TDR register (CSR7[4:0]).

Table 2-3. Encoder output selection

TCH1	TCH0	SELH	
0	0	0	Ch 0 lower Encoder
0	0	1	Ch 0 upper Encoder
0	1	0	Ch 1 lower Encoder
0	1	1	Ch 1 upper Encoder
1	0	0	Ch 2 lower Encoder
1	0	1	Ch 2 upper Encoder
1	1	0	Ch 3 lower Encoder
1	1	1	Ch 3 upper Encoder

2.11.3. Write data to a dual port memory

You can write arbitrary data to one channel of the dual port memory (DPM). Lower 5 bit and upper 5 bit of the DPM data will have same value. Since the write signal to the DPM is common to all 4 channels, the contents of other channel's DPM will be normal input value at the write.

DPM channel is selected by TCH1 and TCH0 bits as shown in Table 2-4. The address of the DPM is set through Write Point Register (WPR, CSR2). Write to the DPM is enabled by setting ENTDR bit. At the end of data writing to TDR, the data are transferred to the designated place of the DPM.

Table 2-4. DPM channel selection.

TCH1	TCH0	
0	0	Ch 0 DPM
0	1	Ch 1 DPM
1	0	Ch 2 DPM

1	1	Ch 3 DPM

2.11.4. Read data directly from the dual port memory

Normally the data stored in the DPM are read out through a readout FIFO (RFIFO). Sometime this may cause extra complication in a test. By setting BYPFIFO flag (CSR5[2]), you can skip the readout FIFO as shown in Fig. 2-10.

The address of the DPM is pointed by the Read Pointer Register (RPR, CSR1), and the channel is selected with CH0 and CH1 signals. The output driver for the D0 - D9 signals are controlled with OE* (output enable) signal as usual.

2.11.5. Check the trigger FIFO

There is no direct path to check the trigger FIFO (TFIFO). However, the function of the TFIFO can be checked through other registers. Peripheral registers of the TFIFO is shown in Fig. 2-12. The WPR address (content of TFIFO) is automatically moved from TFIFO to RPR if buffer space is available in RFIFO after subtracting OFR value. Then the RPR is incremented for data readout. Thus the address checking at RPR is rather difficult unless you understand detail operation within the chip. Below I show one example of checking consistency of TFIFO by reading event data.

(i) Write arbitrary number to OFR (CSR0).

(ii) Write arbitrary number to WPR (CSR2).

(iii) Assert Trigger signal (see section 2.10).

(iv) Read out an event and check the header 2 value is equal to ‘WPR - OFR’

Repeat (i) -(iv) in several combinations.

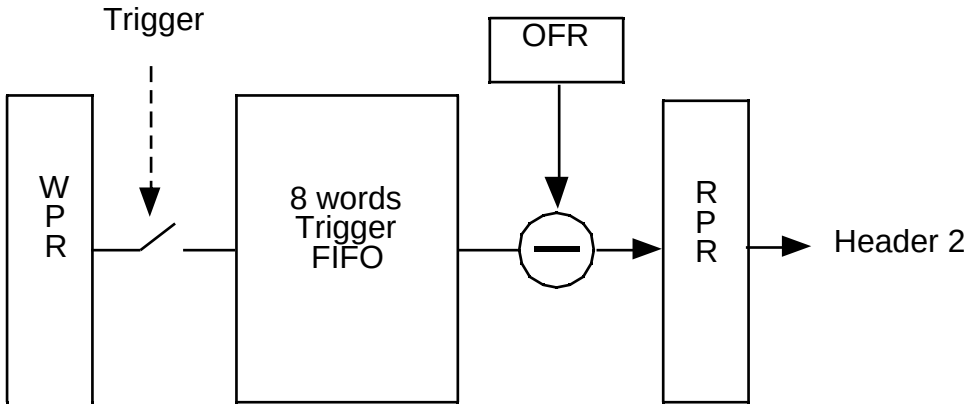


Fig. 2-12 Peripheral circuit of the trigger FIFO.

2.11.6. Check the readout FIFO

Writing arbitrary data to the DPM, and asserting a trigger signal, you can transfer the data to

the RFIFO. By reading out an event, you can check the RFIFO.

3. Performance

3.1. Time Resolution

3.2. Differential and Integral nonlinearity

3.3. PLL Tracking Range and stability

3.4. PLL Step Response

3.5. Power consumption

REFERENCES

[1] Y. Arai and T. Ohsugi; "An Idea of Deadtimeless Readout System by Using Time Memory Cell", Proceedings of the Summer Study on the Physics of the Superconductiong Supercollider, Snowmass, 1986, p.455-457. KEK Preprint 86-64.

[2] Y. Arai and T. Baba; "A CMOS Time to Digital Converter VLSI for High-Energy Physics", Talk at 1988 Symposium on VLSI Circuits, Tokyo, Aug. 22-24 1988, IEEE CAT. No. 88 TH 0227-9 Page 121.

[3] Y. Arai and T. Matsumura and K. Endo; "A CMOS 4 ch x 1 k Time Memory LSI with 1 ns/bit Resolution", IEEE Journal of Solid-State Circuits, Vol.27, No.3, March 1992, p359-364.

[4] Y. Arai and M. Ikeno; “A Time Digitizer CMOS Gate-Array with a 250 ps Time Resolution”, IEEE Journal of Solid-State Circuits, Vol. 31, No. 2, Feb. 1996, P212-220. KEK preprint 95-75.

[5] Y. Arai and M. Ikeno; "A 64-ch Time Memory Cell Module with a DSP and a VME Interface", IEEE Nuclear Science Symposium, San Francisco, Nov. 1993. IEEE Trans. on Nucl. Sci. Vol. 41, No. 4, August 1994, p1187-1191. KEK Preprint 93-151.

[6] H. Shirasu, Y. Arai, M. Ikeno, T. Murata and T. Emura; “A VME 32 ch Pipeline TDC Module with TMC LSIs”, IEEE Nuclear Science Symposium, Oct. 1995. IEEE Trans. on Nucl. Sci. Vol. 43, No. 3 (1996)1799-1803, KEK Preprint 95-143.