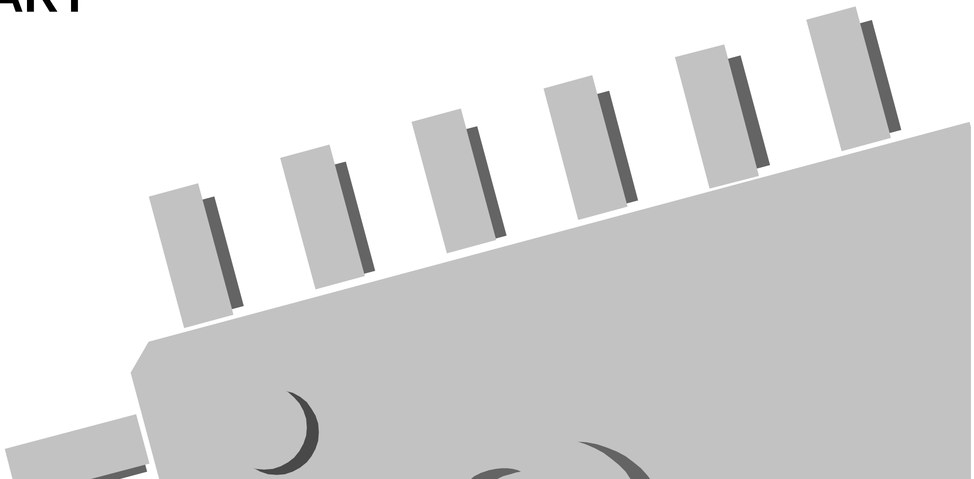
FUJIFILM
I&I – Imaging & Information

LINK (IEEE 1394)
MD8413
Application Note

PRELIMINARY



FUJIFILM MICRODEVICES CO., LTD. Ver1.1

MEMO

history

Legend:

Data LSB, MSB	: Right for LSB, left for MSB
Description of negative logical signal	: Signal name with # at its end
Numerical descriptions	: Binary digit ****b or **** Decimal digit **** Hexadecimal digit ****h or 0x****
Table descriptions	: In the tables of registers, etc., the hatched parts without names denote the ineffective areas and no changes take place in operation in these areas.
Terminology	: Byte Data with 8-bit width Word Data with 16-bit width Quadlet Data with 32-bit width Octat Data with 64-bit width

Associated materials

IEEE1394 Draft 8.0v1 Standard for a High Performance Serial Bus

IEEE Std 1212-1991 Command and Status Register architecture

Contents

Tables and Figures

1. Introduction

The MD8413 is a controller for the link layer of the high-speed serial bus, designed in accordance with the IEEE draft standard, IEEE P1394 Draft 7.1. It is suitable for use as a device to be incorporated in peripheral/terminal equipment, with a variety of functions necessary for the link layer and also those which can relieve the system load in the case of isochronous transfer.

The MD8413 has the following functions:

*Packing for transmission and unpacking for reception in accordance with the IEEE P1394 Draft 8.2
*Cycle master support.
*Parity generation and error detection by the 32-bit CRC.
*Detection of dropped cycle start message.
*Direct with the PHY chip (MD8402) and interface by AC coupling.
*Control of the transfer frequency during isochronous transfer.
*Automatic insertion of a header in an isochronous packet during transmission, and automatic separation of the header during reception.
*Full support of the outbound retry sequence.
*Exclusive data bus for isochronous transmission/reception.
*Support of control signals from PHY (MD8402) to the LPS (Link Power Status).
*Support of bus time register.
*Support of S100 and S200.

Regarding the isochronous system in particular, it is possible to establish an external cycle timer as the cycle timer value can be output to the isochronous bus. However, the output of cycle timer values is distinguished from other isochronous data by virtue of the CT signal.

In this application note, the method of MD8413 usage is explained in conjunction with each function. Regarding information about registers, etc., however, please refer to the user's manual.

Descriptions are given here in regard to malfunctions of the MD8413 for its present versions.

2. Example of applications

When realizing MD8413 and MD8402 applications, the processing blocks are separated to those for the hardware and those for the software. The hardware is mainly in charge of isochronous transmission and reception, and the software is in charge of asynchronous data transfer.

For asynchronous data transfer, it is necessary to provide for the various processing features, such as node controller, isochronous resource manager, FCP, and so on.

In the case of IEEE1394, it is necessary to use an ID, which is unique to each node (Unique ID). Since the MD8402/MD8413 does not support these processing features, such processing need be performed at the node controller of the firmware installed in the MPU.

2-1. Block Diagram

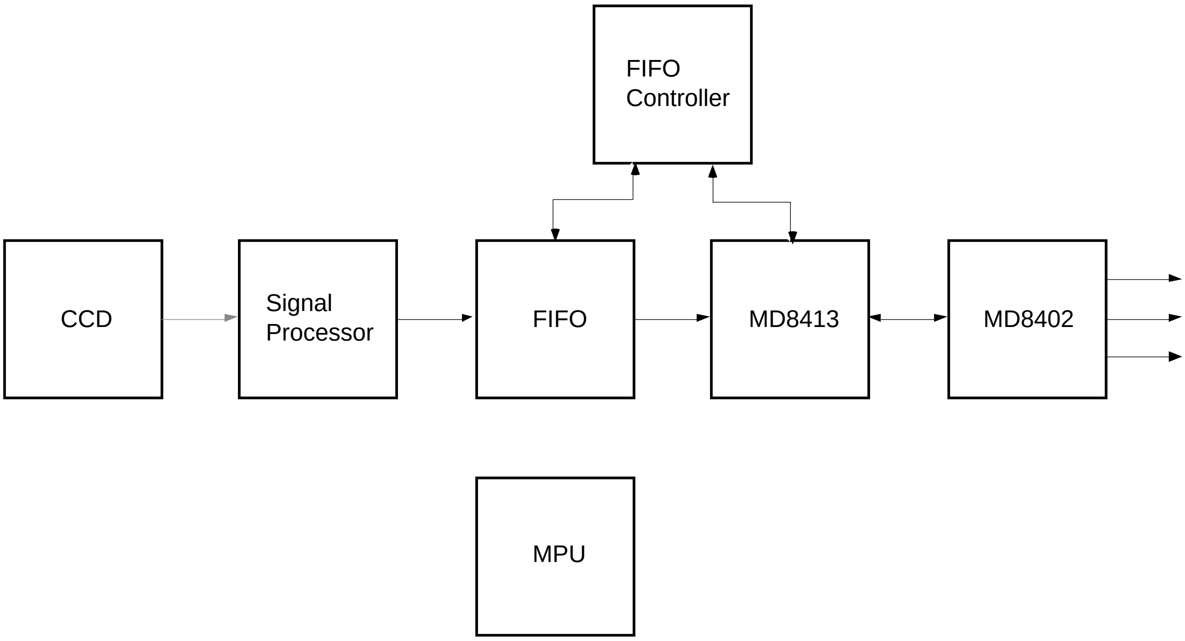


Figure 2-1-1 Example of Digital Camera

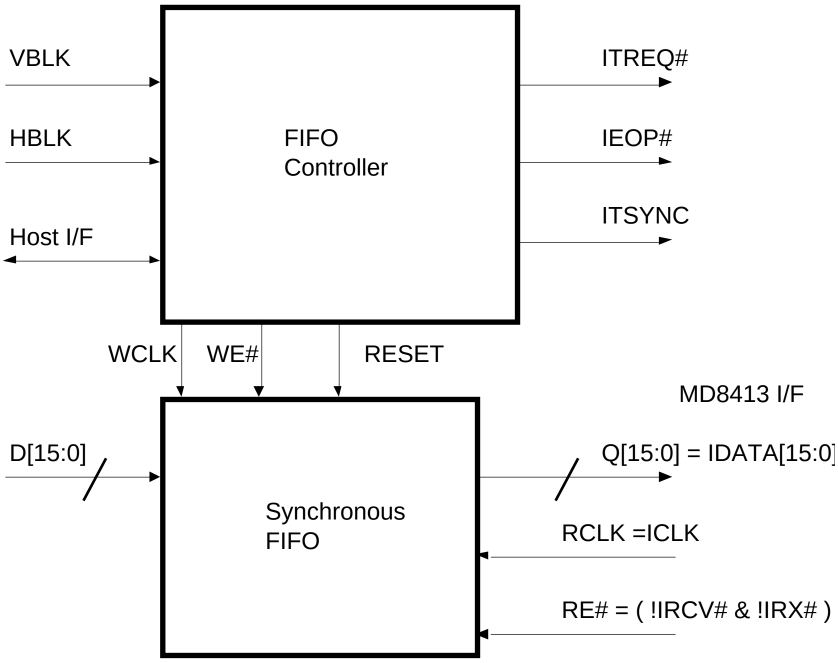


Figure 2-1-2 Example of Digital Camera (Part of FIFO controller)

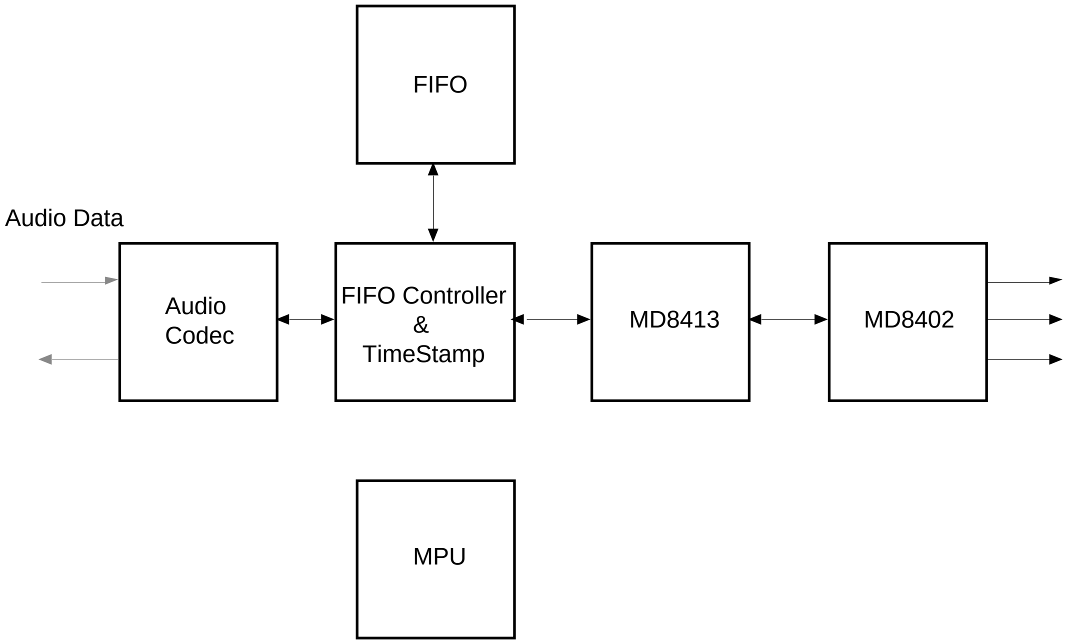


Figure 2-1-3 Example of audio data

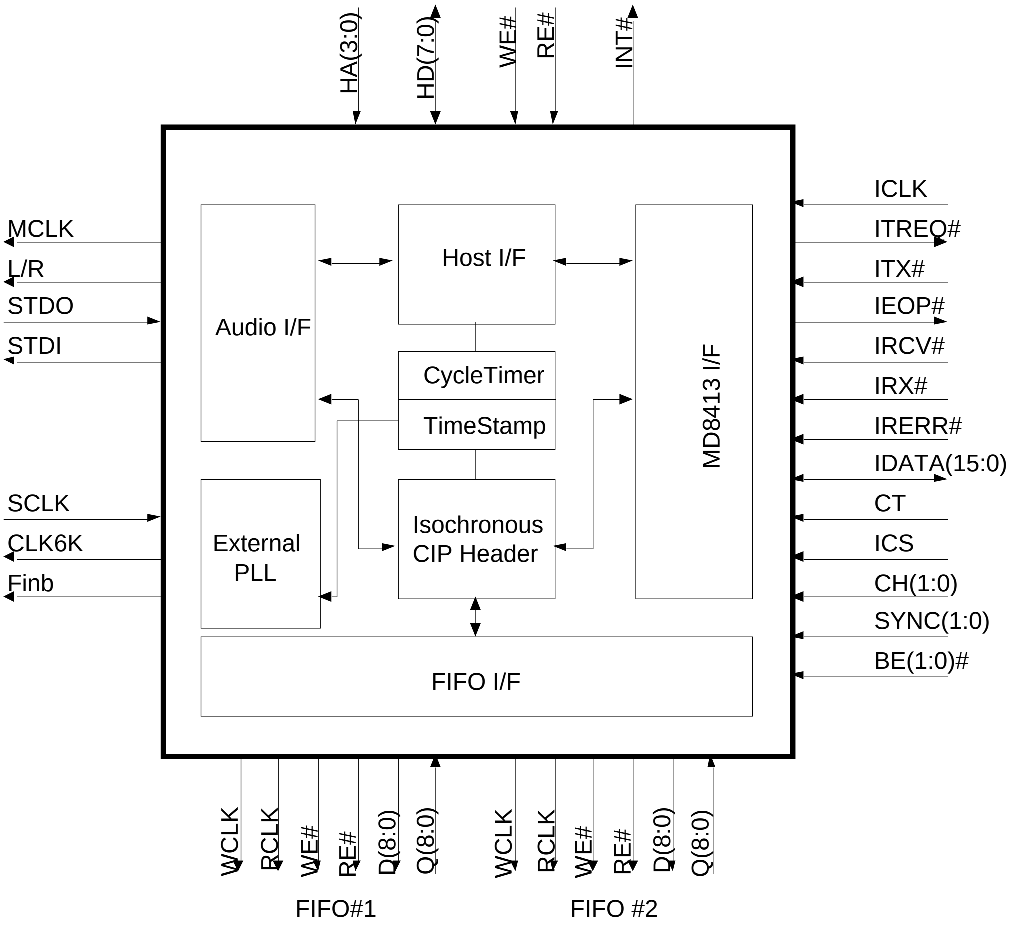


Figure 2-1-4 Example of audio data (Part of FIFO controller & Timestamp)

3. Register using procedures

3-1 Chip version

This register shows the chip version and revision numbers. It is useful for the future control of IEEE 1394 LINK chips in the software.

Bit 15~0 Version : IC chip's version number (R- initial value: 0002h)
Version number of the MD8413 is shown. For the MD8413, "0002h" is always read out.

Bit 15~0 Revision : IC chip's revision number (R- initial value: 0000h)
Revision number of the MD8413 is shown. The value begins with 0 and increases each time revision is finished.

3-2 Control register access

Access to the register employs an SRAM-like asynchronous bus as shown in Fig. 5-1-1.

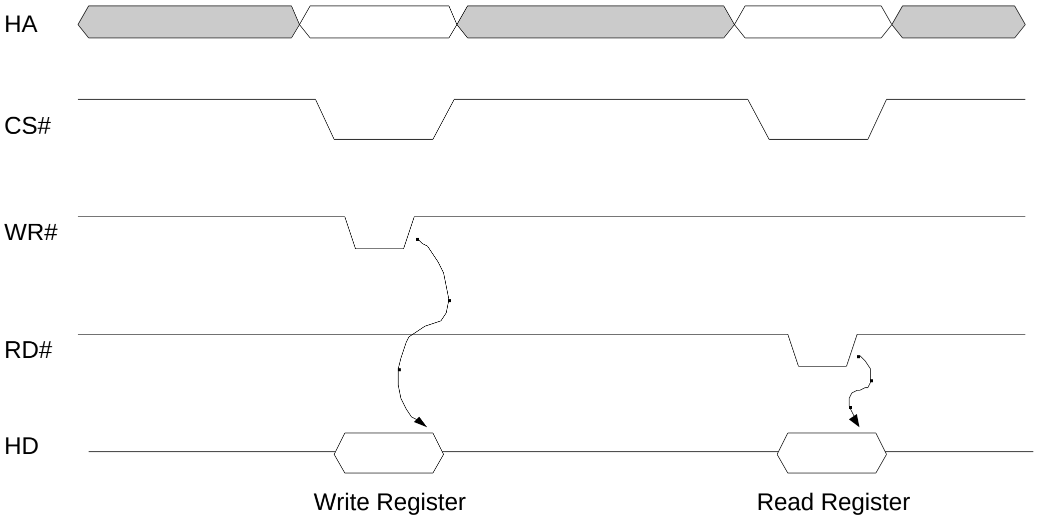


Figure 3-2-1 Host access timing

4. Host I/F

4-1 Data bus width

The effective bits for access from the host are in the fixed 16-bit width.

After the power supply has been turned on, it is necessary for the control register to make processing in the following procedures:

1. The LinkOn bit is set up, in order to enable communication with the PHY chip. Before the setting of the LinkOn bit, no access is valid to any destination other than the control register.

After the above setting has been made, the environment is ready to start communication with the PHY chip. Since then setting is made for various control registers. Except for the buffer control register (ATF.ARF), various registers can be accessed with 16 bits with fixed width.

4-2 DMA

The MD8413 supports the DMA transfer functions during data transfer with the send/receive buffer. The mode of the DMA being supported is only if the DMA service request signal (DREQ) is of the level sense. As a buffer being the object of DMA transfer, only one buffer can be selected by the SelectDreq bit. Whether the DREQ signal is made effective or not is controlled by the DreqEn bit. When DreqEN="1" : the DREQ signal being valid, the assert/negate conditions for the DREQ signal are shown in Table 5-1-3. In the case of DreqEN="0", the DREQ signal is always in the state of negate.

When transferring the transmission data, size must be set up and executed for the data that are being transferred to the DMAC. When the condition is then made for DREQEn="1", a DREQ request is issued for the DMAC, and therefore DMA transfer is executed.

When the receiving data are transferred to the host side, the data length is read out and the resultant value is set in the DMAC so that this DMAC can accomplish the instruction. When DreqEn is set at "1", a DREQ request is issued for the DMAC, thus leading to the execution of DMA transfer.

SelectDreq bit	Forwarding buffer	DREQ assert condition	DREQ negate condition
0b	ATF	When ATF buffer is not full	When ATF buffer is full (ATFFull='1')
1b	ARF	When the data remains in ARF buffer	When ARF buffer is vacant (ARFEmpty='1')

Table 4-2-1 Assert/negate condition of DREQ signal

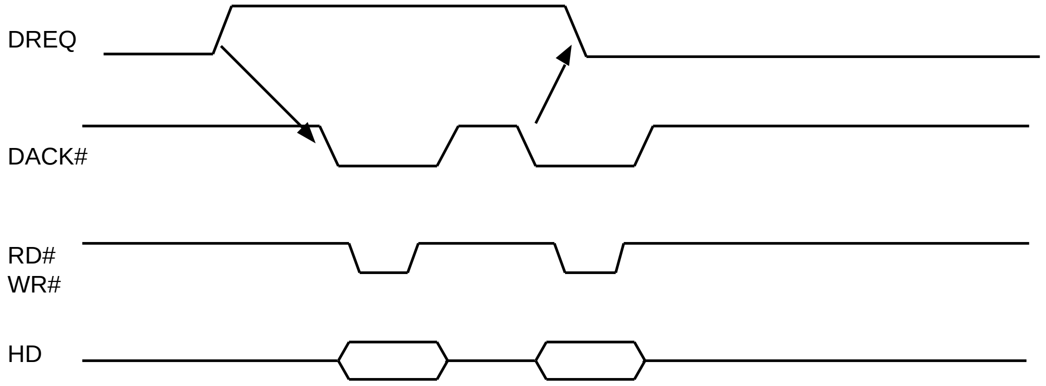


Figure 4-2-1 Timing of DMA transfer

The following is the additional explanation of negate timing for the DMA service request signal (DREQ). As shown in Table 5-1-3, the negate condition is held when the internal buffer is full. When writing is attempted to make this buffer full (by entering WR# input), the DREQ signal is negated with a timing as shown in Fig. 5-1-2.

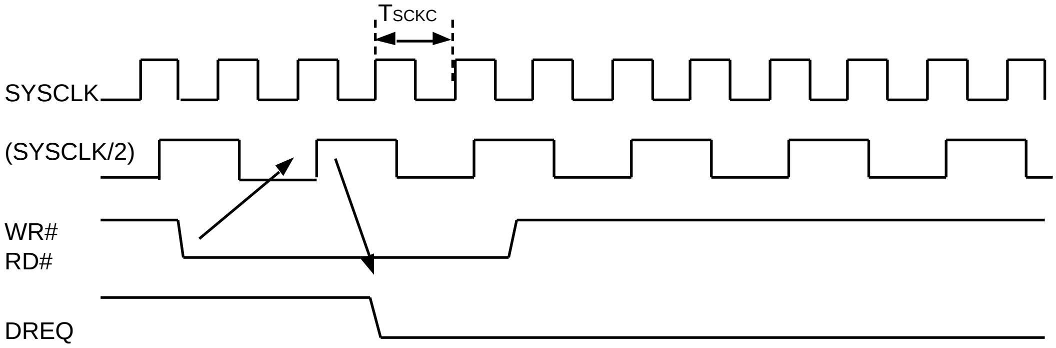
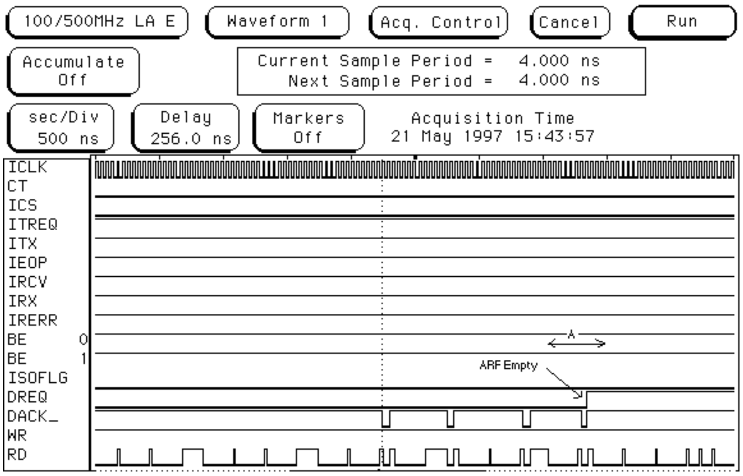


Figure 4-2-2 DREQ negate timing

Actual DMA operation is shown in Figs. 4-2-3 and 4-2-4.



Accumulate Off

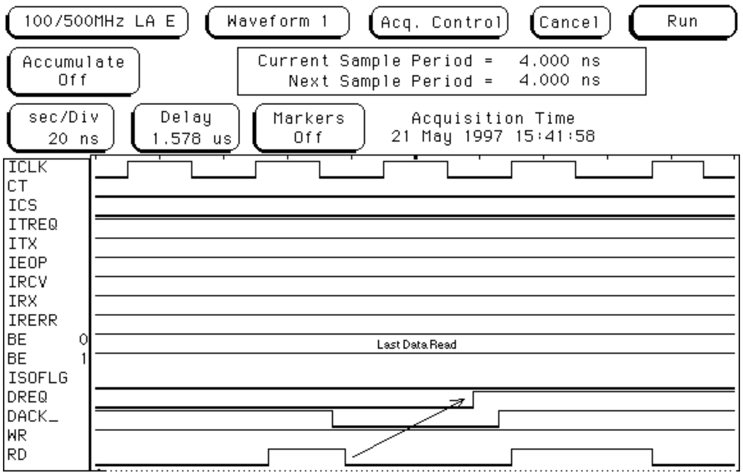
Current Sample Period = 4.000 ns
Next Sample Period = 4.000 ns

sec/Div 500 ns

Delay 256.0 ns

Markers Off

Acquisition Time
21 May 1997 15:43:57



100/500MHz LA E

Waveform 1

Acq. Control

Cancel

Run

Accumulate Off

Current Sample Period = 4.000 ns
Next Sample Period = 4.000 ns

sec/Div 20 ns

Delay 1.578 us

Markers Off

Acquisition Time
21 May 1997 15:41:58

Figure 4-2-4 DREQ Negate Block of DMA Transfer (Read-out): Part A Enlarged

4-3 Interrupt

The MD8413 has the two INT# terminals to be used for sending an announcement to the host, regarding the cause of interrupt defined at the interrupt register and the interrupt mask register. The signals from the INT# terminals are at active low, and asserted in the form that OR is removed from the interrupt factor not masked at the interrupt mask register. By writing '1' in all bits in the interrupt register, each bit is cleared and the INT# terminal is negated. INT1# and INT2# perform the same operation, and contents of the interrupt registers can be assigned to INT1# or INT2# by the use of Interrupt Mask1,2.

For example, the interrupt factors of BusReset and PHYInt, PhyRegRecvd can be assigned to INT1#, and the interrupt factors of asynchronous transfer can be assigned to INT2#. By this assignment, interrupt routines can be arranged by the MPU and it is possible to change the order of preference or such factors.

5. Transmission

In the MD8413, packet transmission is effected by either of the two methods, asynchronous or isochronous transmission. Both transmission methods are enabled by writing a request in FIFO.

5-1 Asynchronous transmission

Procedures for asynchronous transmission are described below. In asynchronous transmission, operation is furthered in a pair of transactions, transmission and response (acknowledge). For more details, refer to the specifications of IEEE1394. In this section, a brief explanation is given in regard to Procedure 1 for transmission.

5-1-1 Transmission transactions

Transactions are roughly divided into the following two categories: (1) generation of a packet for transmission and execution of transmission (data packet request) and (2) confirmation of the transmission result to examine how the receiving party has finished reception.

* Unified Transactions

* SplitTransactions

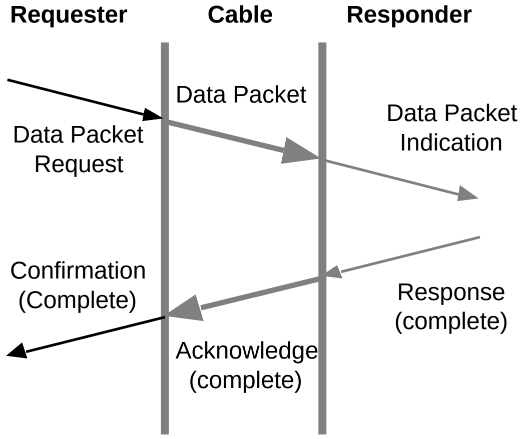


Figure 5-1-1 Unified transaction

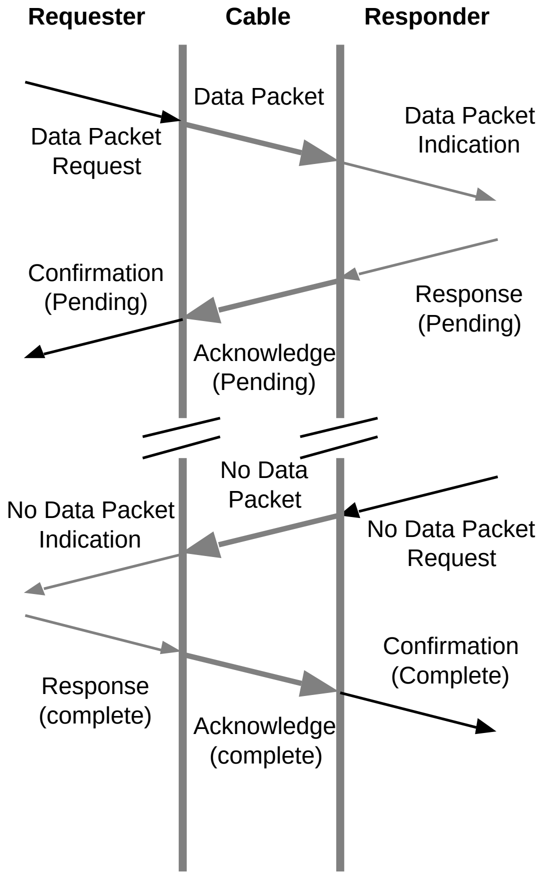
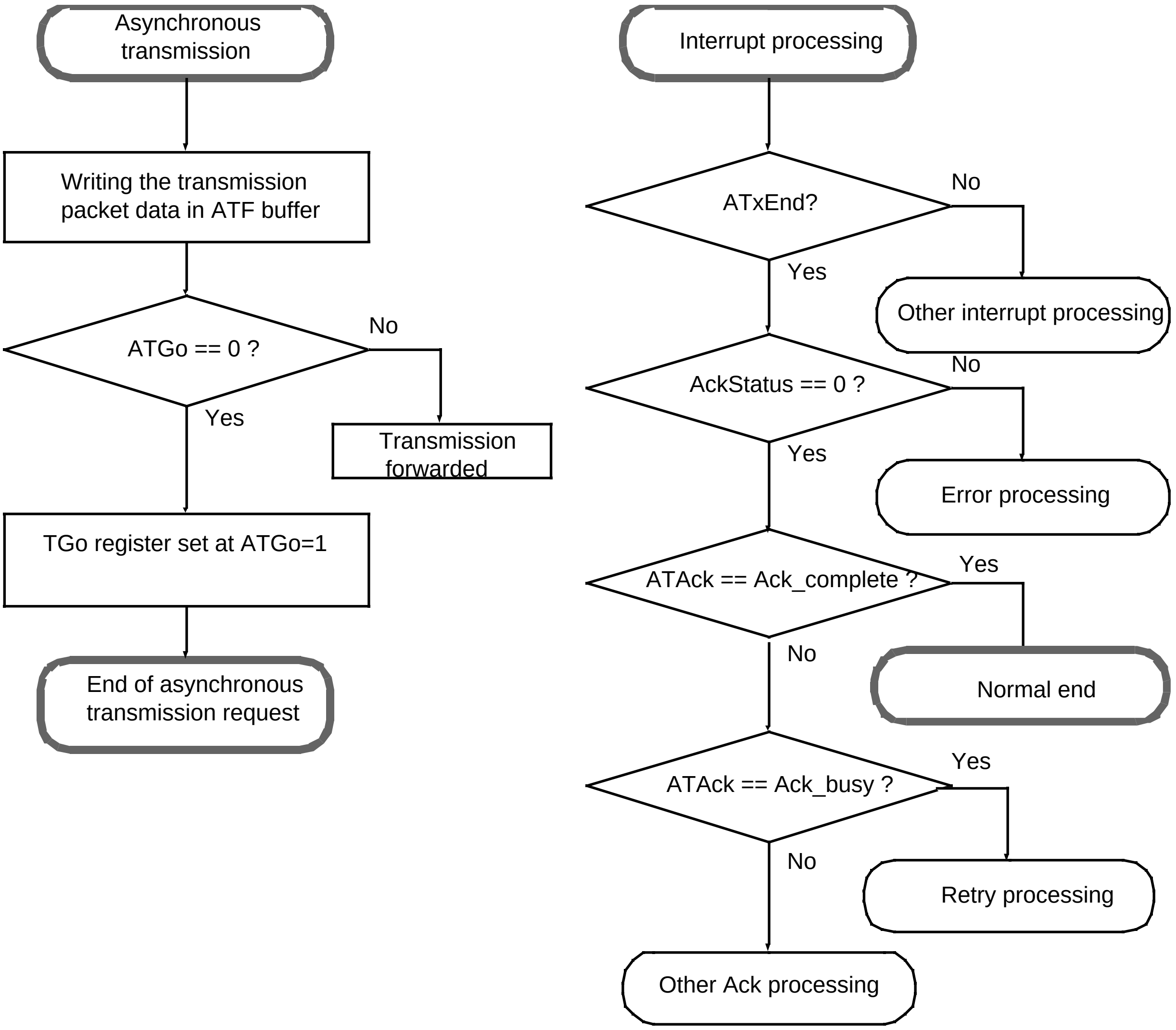


Figure 5-1-2 Split transaction

The transmission transactions can be explained in the following flow chart:



As an example of transmission, the case of Write Quadlet Request is shown below.

<Conditions>

1.	Node of sending party	: NodeID=0
2.	Node of receiving party	: NodeID=1 (Local Bus)
3.	Transmission address	: 0xFFC1 0000 1000 0000
4.	Transmission data	: 0xAAAA 5555
5.	tLabel	: 0x0

<Transactions>

1. The following 4 quadlet data are written in the ATF.
   0x0000 0000
   0xFFC1 0000
   0x1000 0000
   0xAAAA 5555
2. Confirming that the ATGo bit is "0" in the Tgo register.
3. "1" is written in the ATGo bit.

4. Packet transmission is started after the above steps.

5. When transmission is finished and Ack code is returned, the ATxEnd bit is set in the interrupt register.

6. Confirming that the ATxEnd bit has been set in the interrupt register (interrupt or polling).

7. The AckStatus bit is read in the diagnostic status register. If it is found to be "00", the received Ack code (ATAck bit) is processed.

8. The ATAck bit is read in the diagnostic status register. If it is found to be "Ack_complete", this means the successful completion of normal transmission, and therefore the transactions are finished.

9. In cases other than Ack_complete, transactions are conducted to cope with the corresponding Ack code. In the case of Ack_Busy, for example, retry processing is furthered.

5-1-2 Retry control

Retry is a trial of sending the same data again when the receiving party cannot receive the transmitted data for some reasons (no storage in FIFO on the receiving side). When transmission is attempted as shown below, but Busy is returned with the Ack code, the same data are transmitted again.

The MD8413 has the function of processing the retry on the transmission side (outbound), using hardware. When the sending data are stored in the ATF buffer and the returned Ack code shows a busy condition, the contents in the ATF buffer are not cleared and transmission is attempted again. The maximum number of retries can be set in the register. If the number of retries has exceeded the maximum preset value and the condition is still busy, then this busy information is sent to the host side.

The retries specified in IEEE1394 are covered by the two protocols; the dual phase retry protocol to be controlled by RetryA/B and the single phase retry protocol controlled by RetryX. The MD8413 covers both protocols. Use of these retry protocols is defined by setting at the receiving party. In the case of the MD8413 also, the Busy Status bit in the packet control register can be used to specify which protocol should be selected.

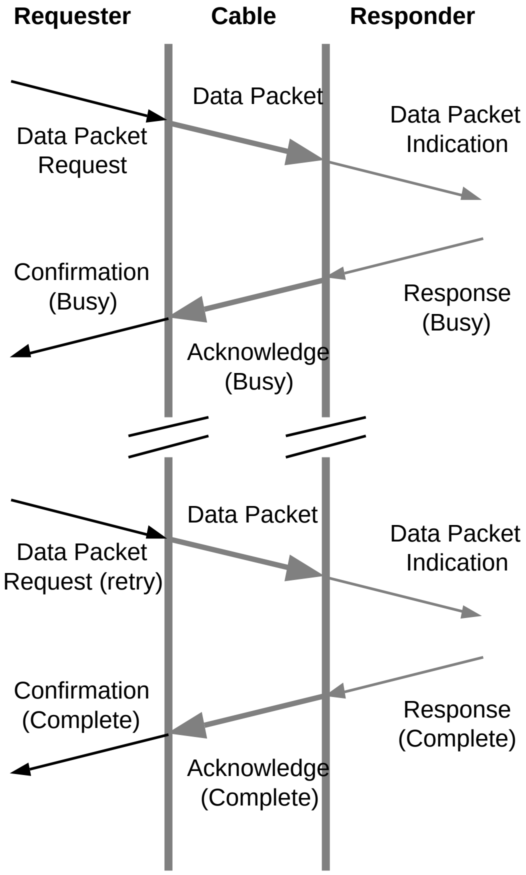


Figure 5-1-5 Retry transaction

The retry flow chart is shown below in case when the maximum number of retries is only one. Fig. 5-1-6 shows that Packet1 is transmitted, but Ack_busy is returned from the destination, and that the MD8413 makes retry transmission by reusing the packet data stored in the ATF buffer. In the second transmission, Ack_complete is returned to indicate normal reception and therefore this transmission is completed. For Packet3, however, Ack_busy is returned even after the second retry, which means the attainment of the maximum retry number and interrupt takes place. This interrupt is the same ATxEnd as for normal transmission, but it can be used for ATAck to identify if it is normal transmission or not.

Figure 5-1-6 Retry transaction

5-2 Isochronous transmission

5-2-1 CycleStart packet transmission

CycleStart packet transmission is carried out under the following conditions:

1) The root bit is already set. (Automatically set shortly after BusReset)

2) The CycleMasterEn bit is set.

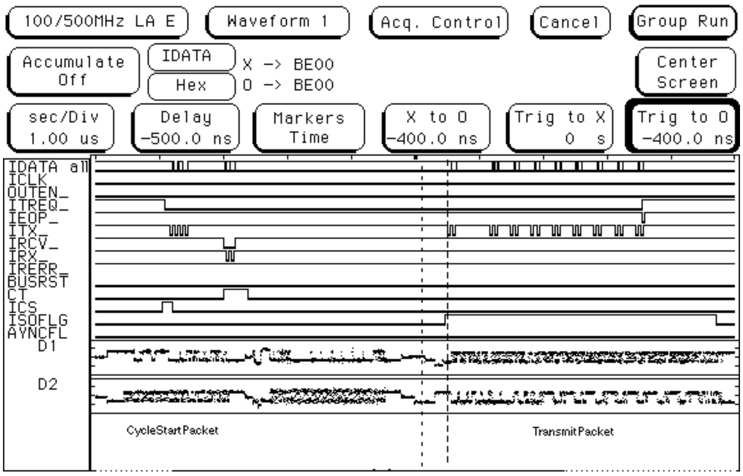
3) The CycleTimerEn bit is set.

When all the above conditions have been satisfied, CycleStartPacket is transmitted.

5-2-2 Normal mode transmission transactions

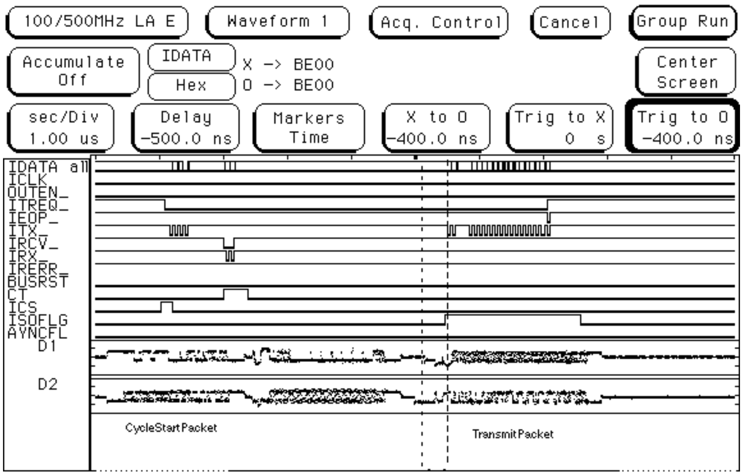
Fig. 5-2-1 shows the waveforms observed during transmission at 100Mbps. Fig. 5-2-2 shows the waveforms observed when the same data are transmitted at 200Mbps. The ITREQ signal has already been asserted shortly after the rising of the ICS signal (after 1-ICLK). The ITREQ signal is also negated before the final data. Under such conditions, data of another channel are flowing in the bus before transmission.

When the ITREQ signal is asserted after the ICS signal, data for 2 quadlets (4 times) are read out.



Accumulate Off
IDATA Hex
X -> BE00
O -> BE00
sec/Div 1.00 us
Delay -500.0 ns
Markers Time
X to O -400.0 ns
Trig to X 0 s
Trig to O -400.0 ns

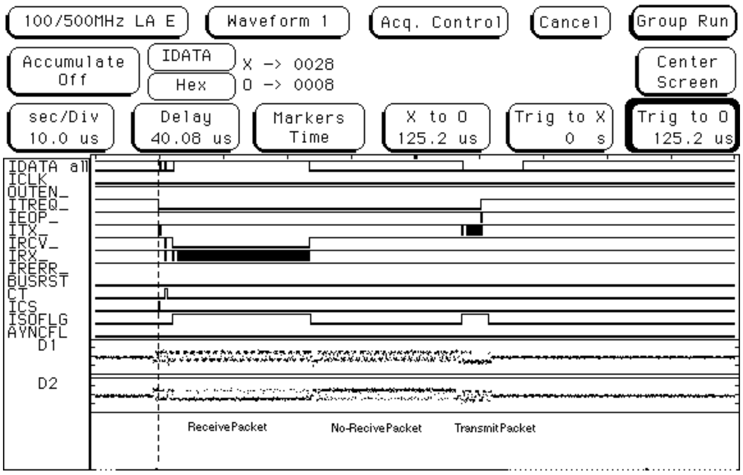
Figure 5-2-1 Isochronous transmission (100MBPS)



Accumulate Off
IDATA Hex
X -> BE00
O -> BE00
sec/Div 1.00 us
Delay -500.0 ns
Markers Time
X to O -400.0 ns
Trig to X 0 s
Trig to O -400.0 ns

Figure 5-2-2 Isochronous transmission (200MBPS)

Fig. 5-2-3 shows the waveforms to be observed when there is packet reception before transmission.

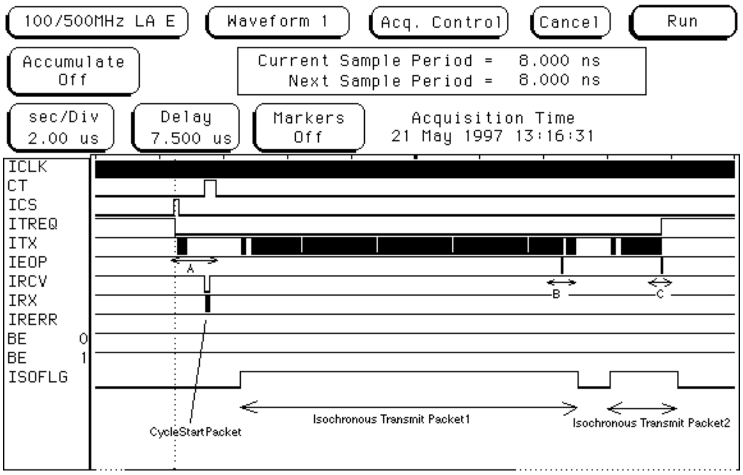


Accumulate Off
IDATA Hex
X -> 0028
O -> 0008
sec/Div 10.0 us
Delay 40.08 us
Markers Time
X to O 125.2 us
Trig to X 0 s
Trig to O 125.2 us

Figure 5-2-3 Isochronous Transmission (when there is a reception packet)

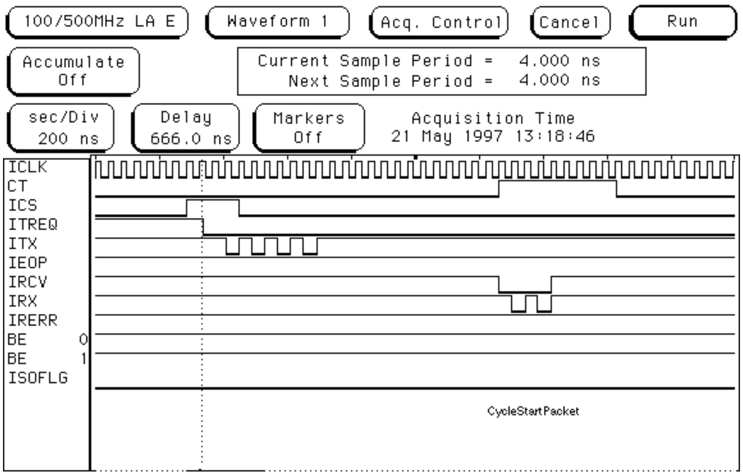
Fig. 5-2-4 shows the waveforms observed during 2-channel transmission. Usually, in the case of transmission for one channel only, the ITREQ signal is negated before the transmission of the last data in the packet. In this state, it is necessary for the ITREQ signal to assume the negate condition when the IEOP signal is asserted.

In the case of 2-channel transmission, however, the ITREQ signal is kept asserted at the time of the last data of one channel (timing when the IEOP signal is asserted). Then, a transmission data request (ITX signal asserted) is generated from the MD8413. At that time, as shown in Fig. 5-2-6, preliminary reading of the next packet takes place as soon as the packet of the first channel has been finished. If the ITREQ signal is kept negated till the last data of the last channel packet, transmission is finished with its isochronous cycle. (Refer to Fig. 5-2-7.)



Accumulate Off
Current Sample Period = 8.000 ns
Next Sample Period = 8.000 ns
sec/Div 2.00 us
Delay 7.500 us
Markers Off
Acquisition Time
21 May 1997 13:16:31

Figure 5-2-4 Part A Enlarged for Isochronous Transmission (Multi-Channel)



100/500MHz LA E | Waveform 1 | Acq. Control | Cancel | Run

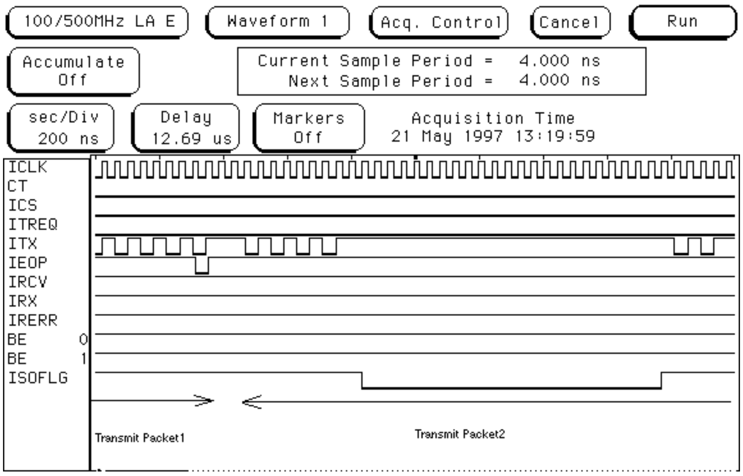
Accumulate Off

Current Sample Period = 4.000 ns
Next Sample Period = 4.000 ns

sec/Div 200 ns | Delay 666.0 ns | Markers Off

Acquisition Time 21 May 1997 13:18:46

Figure 5-2-5 Part A Enlarged for Isochronous Transmission (Multi-Channel)



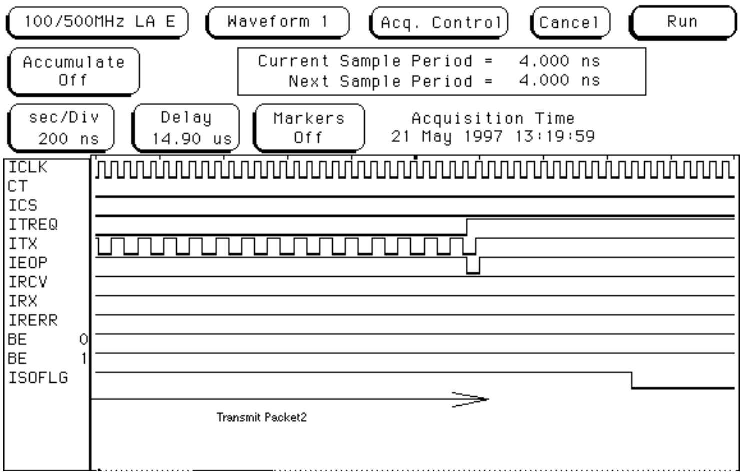
100/500MHz LA E | Waveform 1 | Acq. Control | Cancel | Run

Accumulate Off

Current Sample Period = 4.000 ns
Next Sample Period = 4.000 ns

sec/Div 200 ns | Delay 12.69 us | Markers Off

Acquisition Time 21 May 1997 13:19:59



100/500MHz LA E | Waveform 1 | Acq. Control | Cancel | Run

Accumulate Off

Current Sample Period = 4.000 ns
Next Sample Period = 4.000 ns

sec/Div 200 ns | Delay 14.90 us | Markers Off

Acquisition Time 21 May 1997 13:19:59

Figure 5-2-7 Part C Enlarged for Isochronous Transmission (Multi-Channel)

5-2-3 Auto-mode transmission transactions

In the auto-transmission mode, the data input from the isochronous bus has the data from which an isochronous header has been excluded. The isochronous header is set in the register specified below. For transmission, the preset isochronous header is added again to the data.

Isochronous Transmit Configuration Register 1

7	6	5	4	3	2	1	0
▨	▨	▨	▨	▨	▨	Speed	
15	14	13	12	11	10	9	8
Tag		Channel					

Bit 1 ~ 0 Speed: Speed bit (RW - initial value: 0b)

0 = Transmission at 100Mbps

1 = Transmission at 200Mbps

In the transmission mode, a transmitting transfer speed is set up on the cable.

Bit 13 ~8 Channel: Channel bit (RW - initial value: 00h)

The channel is specified for the transmitting isochronous packet. The channel number specified here is inserted in the packet header for transmission. The setting range is 0 to 63.

Bit 15 ~ 14 Tag: Tag bit (RW - initial value: 00h)

A tag is specified for the isochronous bit. The setting range is 0 to 3.

Isochronous Transmit Configuration Register 2

7	6	5	4	3	2	1	0
StopSync				▨	▨	▨	SyncEn
15	14	13	12	11	10	9	8
Sync				StartSync			

Bit 0 SyncEn: Sync Enable bit (RW - initial value: 0b)

0 = Contents of the Sync register are reflected at any time in the Sync area of the packet header, regardless of the ITSYNC signal.

1 = The Sync field value is defined in the packet header to be transmitted according to the ITSYNC signal. Contents of StartSync are put in the transmission packet shortly after ITSYNC active. Since then, the Sync data are entered and StopSync is entered in the transmission packet shortly after ITSTART non-active. When ITSYNC is disabled, the following conditions are assumed according to the setting value of ITZERO:

ITZERO=0 : No transmission is effected.

ITZERO=1 : Sync field is entered at any time.

Bit 7 ~ 4 StopSync: Stop Sync bit (RW - initial value: 00h)

When the ITSTART terminal is made to be non-active with SyncEn="1b", this value is written in the Sync field of the subsequent packet.

Bit 11 ~ 8 StartSync: Start Sync bit (RW - initial value: 00h)

When the ITSTART terminal is made to be active with SyncEn="1b", this value is written in the Sync field of the packet to be sent subsequently.

Bit 15 ~ 12 Sync: Sync bit (RW - initial value: 00h)

With SyncEn="1b" at the ITSTART terminal, this value is written in the packet other than the StartSync packet/StopSync packet specified.

Isochronous Transmit Configuration Register 3

7	6	5	4	3	2	1	0
ITLength							
15	14	13	12	11	10	9	8
ITLength							

Bit 15 ~ 0 ITLength: Isochronous Transmit Length bit (RW - initial value: 0h)

Length of the transmitting packet (in the byte unit) is set up in this register. This value is used for the header block during transmission.

5-2-3 Sync field control during auto-mode transmission

The value to be entered in the Sync field of the isochronous header can be controlled from the outside. The signal used for this control is the ITSYNC signal.

To control the Sync field of the isochronous header with this ITSYNC, it is necessary to set the SyncEn bit of the register shown below.

Isochronous Transmit Configuration Register 2

7	6	5	4	3	2	1	0
StopSync				▨	▨	▨	SyncEn
15	14	13	12	11	10	9	8
Sync				StartSync			

Bit 0 SyncEn: Sync Enable bit (RW - initial value: 0b)

0 = Contents of the Sync register are reflected at any time in the Sync area of the packet header, regardless of the ITSYNC signal.

1 = The Sync field value is defined in the packet header to be transmitted according to the ITSYNC signal. Contents of StartSync are put in the transmission packet shortly after ITSYNC active. Since then, the Sync data are entered and StopSync is entered in the transmission packet shortly after ITSTART non-active. When ITSYNC is disabled, the following conditions are assumed according to the setting value of ITZERO:
ITZERO=0 : No transmission is effected.
ITZERO=1 : Sync field is entered at any time.

Bit 7 ~ 4 StopSync: Stop Sync bit (RW - initial value: 00h)

When the ITSTART terminal is made to be non-active with SyncEn="1b", this value is written in the Sync field of the subsequent packet.

Bit 11 ~ 8 StartSync: Start Sync bit (RW - initial value: 00h)

When the ITSTART terminal is made to be active with SyncEn="1b", this value is written in the Sync field of the packet to be sent subsequently.

Bit 15 ~ 12 Sync: Sync bit (RW - initial value: 00h)

With SyncEn="1b" at the ITSTART terminal, this value is written in the packet other than the StartSync packet/StopSync packet specified.

Regarding the detailed timing of the ITSYNC signal, the ITSYNC signal is sampled at the time point of "2" in the ICS signal as shown in Fig. 5-2-3-1. According to this information, the sync field value of the isochronous cycle is defined as shown in Fig. 5-2-3-2.

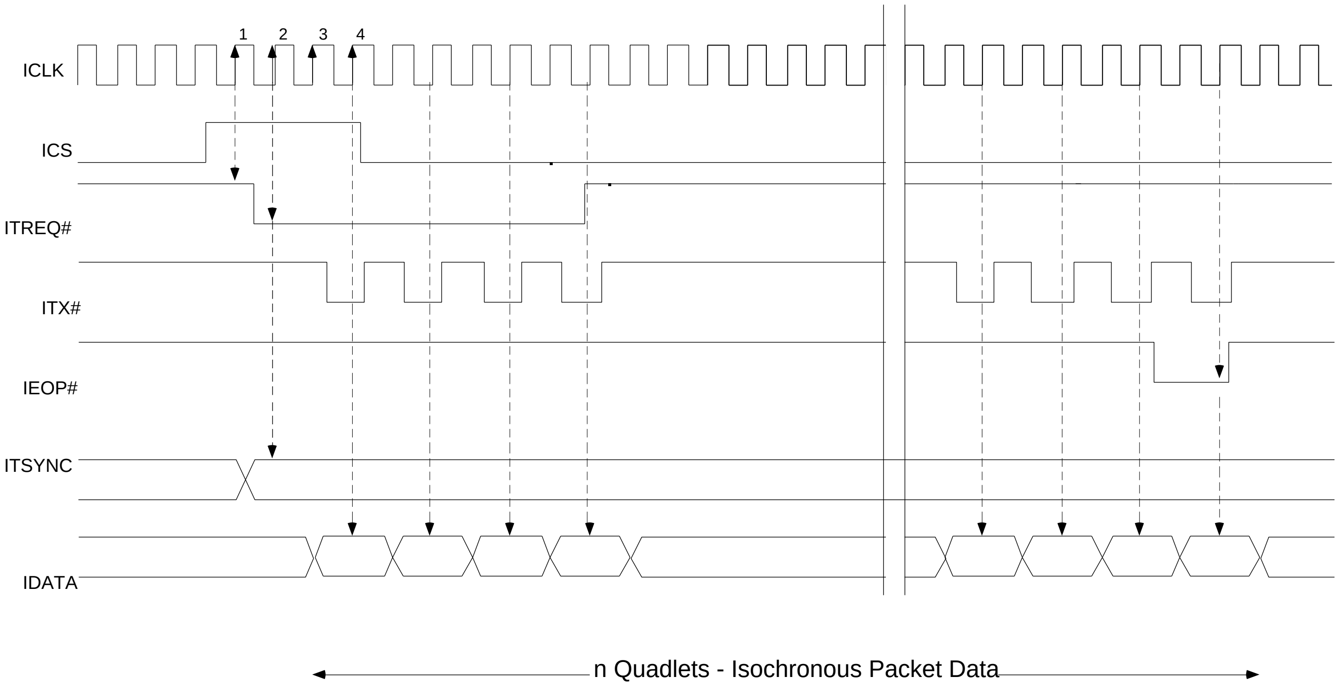


Figure 5-2-3-1 ITSYNC timing

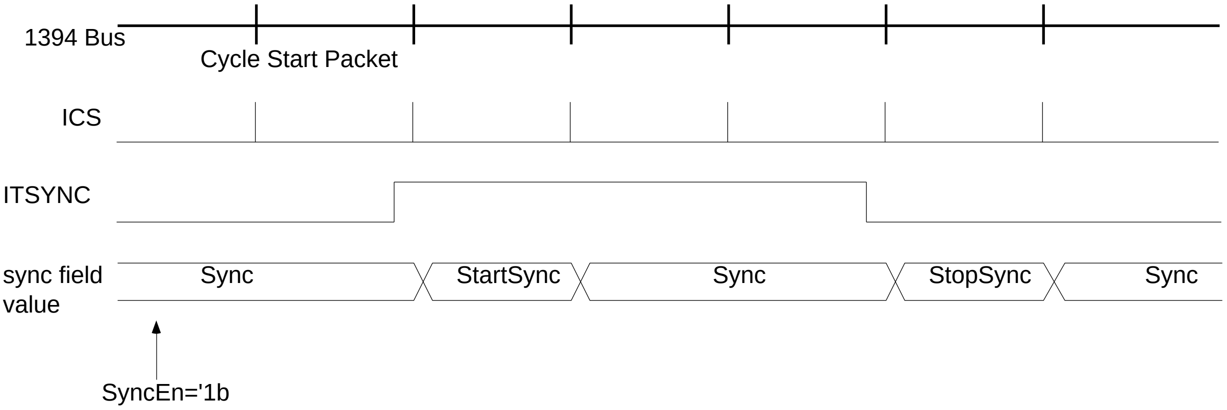


Figure 5-2-3-2 Relationship between ITSYNC signal and sync field

6. Reception

6-1 Asynchronous reception

6-1-1 Reception transactions

When an asynchronous or broadcast packet is received at your node number, reception interrupt (ARxEnd) takes place.

The table below shows the status of interrupt indicating the receiving condition and how the status information is presented.

Status	When normal reception fails due to FIFO full in the middle of reception		Normal reception accomplished
	When ARF vacancy rate is 4 Quadlets or below at the reception start	When ARF vacancy rate is 5 Quadlets or below at the reception start	
ARxEnd interrupt	0	0	1
ARFRej interrupt	1	1	0
ARF status	Same status as that before reception. Nothing written.	Same status as that before reception. Nothing written.	All stored.

When normal reception is accomplished, the data stored in the ARF register can subsequently be read out by sending out a read-out request to this ARF register.

As an example of reception, reception of Write Quadlet Request is shown below.

<Conditions>

1. Node of sending party	: NodeID=0
2. Node of receiving party	: NodeID=1 (Local Bus)
3. Reception address	: 0xFFC1 0000 1000 0000
4. Reception data	: 0xAAAA 5555
5. tLabel	: 0x0
6. Speed	: 100Mbps

<Transactions>

1. ARxEnd interrupt takes place. This ARxEnd interrupt is cleared.
2. When the ARStatus bit is written and Ack_complete is confirmed, this means that normal reception has been accomplished.
3. Since normal reception is finished, contents of the ARF register are read out. The first 1Quadlet is read out. The following contents are read out:
   0x0000 0000
4. The tCode portion is identified as "00" and this is regarded as a Write Quadlet request. Therefore, components of 3 Quadlets are read from the ARF register in a size inclusive of data components. The following contents are read out:

0xFFC1 0000

0x1000 0000

0xAAAA 5555

5. Lastly, data follow showing the status of reception. These data are also read out of the ARF register. The data shown below are read out. In this case, Ack_complete becomes the status at 100Mbps.
   0x0000 0001 at 100Mbps
6. Following the above steps, the receiving data can be read out.
   When a block packet is received, the data length of the packet is stored in the 4th quadlet. Therefore, components of these data are read from the ARF register.

6-1-2 Ack control

In the case of reception, the MD8413 returns an Ack code without exception. The Ack code to be returned generally comes in the following three types:

1. Ack codes in case of normal reception (Ack_complete, Ack_pending)
2. Ack codes in case when the packet storing area is not located in the ARF (Ack_busyX, Ack_busyA, Ack_BusyB)
3. Ack codes in case when packet data contain errors (Ack_data_error, Ack_type_error)

The Ack codes to be controlled by the MD8413 are Ack_complete, Ack_pending, and Ack_busy, as specified below.

Packet	WritePending = '0b'			WritePending = '1b'		
	ack_complete	ack_pending	ack_busy	ack_complete	ack_pending	ack_busy
Write Request	O	-	O	-	O	O
Read Request	-	O	O	-	O	O
Write Response	O	-	O	O	-	O
Read Response	O	-	O	O	-	O
Lock Request	-	O	O	-	O	O
Lock Response	O	-	O	O	-	O

Among these, Ack_Busy can be used for the selection of BusyX or BusyA/B. The difference between Ack_busyX and Ack_busyA/B is whether a retry is effected in the single phase (Ack_busyX) or in the dual phase (Ack_busyA/B). Refer to the draft of IEEE1394 regarding the retry protocols for the single and dual phases.

Setting values for the Busy Control bit:

000 =A busy acknowledge signal is returned according to the dual phase retry protocol, only if the internal Async receiving buffer has no vacant area for a receiving packet.

001 = An acknowledge signal is returned in the BusyA status, only if the internal Async receiving buffer has no vacant area for a receiving packet.

010 =An acknowledge signal is returned in the BusyB status, only if the internal Async receiving buffer has no vacant area for a receiving packet.

011 =An acknowledge signal is returned in the BusyX status, only if the internal Async receiving buffer has no

vacant area for a receiving packet.

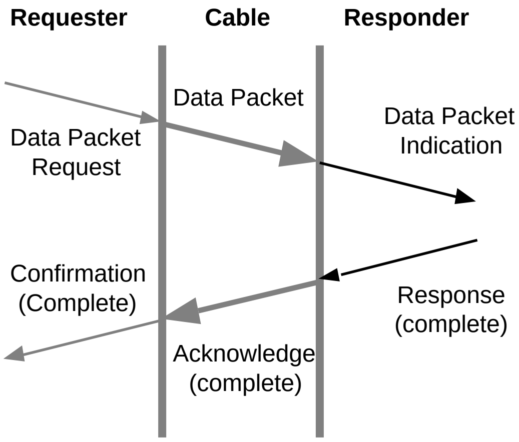
100 =A busy acknowledge signal is returned to all the received packets according to the dual phase retry protocol, regardless of whether the internal Async receiving buffer has a vacant area for a receiving packet.

101 =A busy acknowledge signal is returned to all the received packets in the BusyA status, regardless of whether the internal Async receiving buffer has a vacant area for a receiving packet.

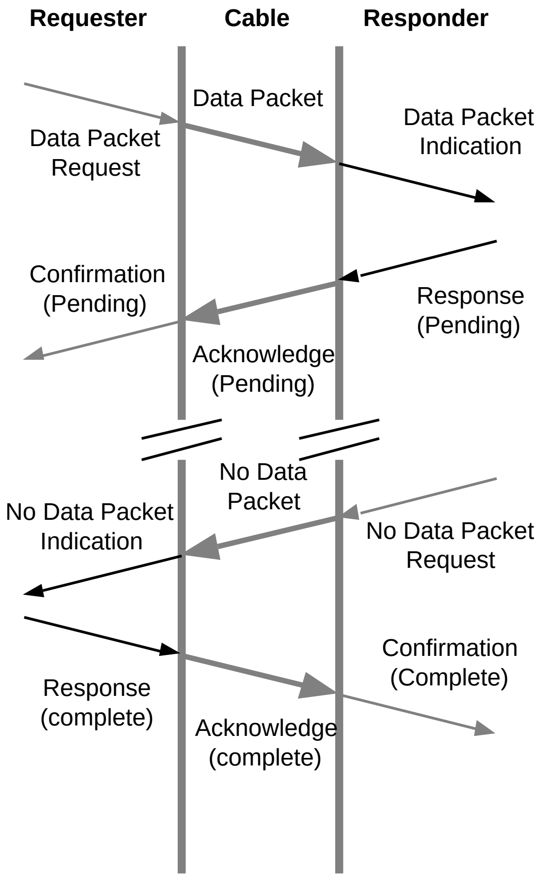
110 =A busy acknowledge signal is returned to all the received packets in the BusyB status, regardless of whether the internal Async receiving buffer has a vacant area for a receiving packet.

111 =A busy acknowledge signal is returned to all the received packets in the BusyX status, regardless of whether the internal Async receiving buffer has a vacant area for a receiving packet.

In this case, the difference between Ack_complete and Ack_pending results in the following communication transactions as described in Paragraph 5-1-1:



In case of Ack_complete

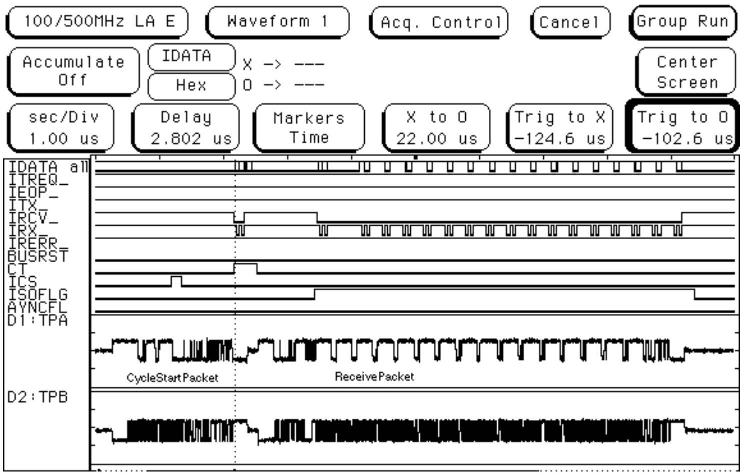


In case of Ack_pending

6-2 Isochronous reception

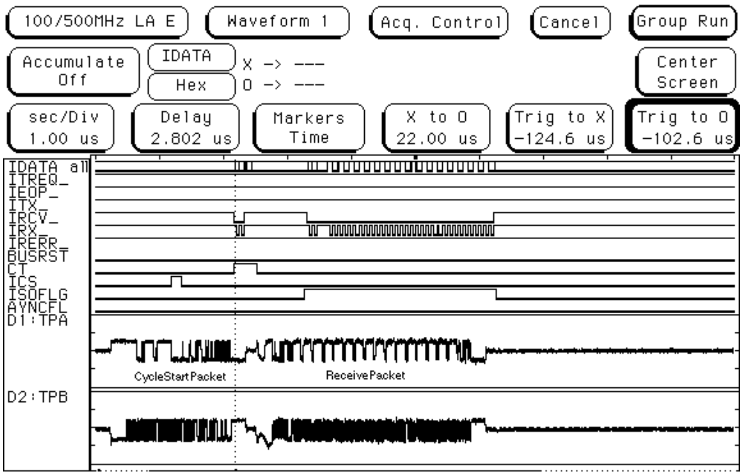
6-2-1 Receiving transactions in normal mode

When IsoMode is "00", reception is enabled only in the receiving channel that has been set for Isochronous Receive Configuration 1-4. When IsoMode is "01", reception is effected for all isochronous packets. Waveforms observed at that time are shown in Figs. 6-2-1 and 6-2-2.



Accumulate Off
IDATA Hex
X -> ---
O -> ---
sec/Div 1.00 us
Delay 2.802 us
Markers Time
X to O 22.00 us
Trig to X -124.6 us
Trig to O -102.6 us

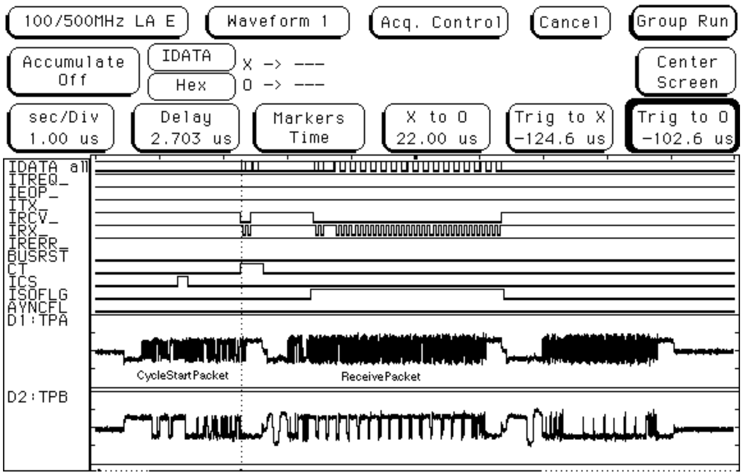
Figure 6-2-1 Isochronous reception (100MBPS)



Accumulate Off
IDATA Hex
X -> ---
O -> ---
sec/Div 1.00 us
Delay 2.802 us
Markers Time
X to O 22.00 us
Trig to X -124.6 us
Trig to O -102.6 us

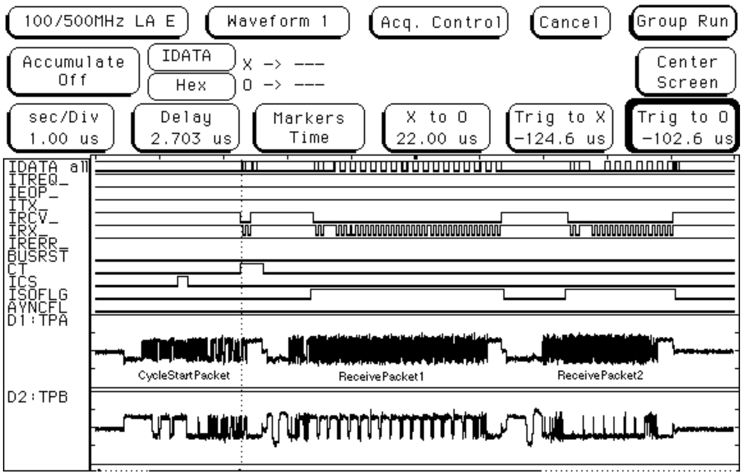
Figure 6-2-2 Isochronous reception (200MBPS)

Figs. 6-2-3 and 6-2-4 show the waveforms in the IsochronousBus when IsoMode is changed for the data in the cable. Fig. 6-2-3 shows the case when the first packet only is made to be received for Isochronous Receive Configuration1 with IsoMode= "00". Fig. 6-2-4 shows the waveforms observed when setting is made for IsoMode="01". Fig. 6-2-5 shows enlarged waveforms continuously received.



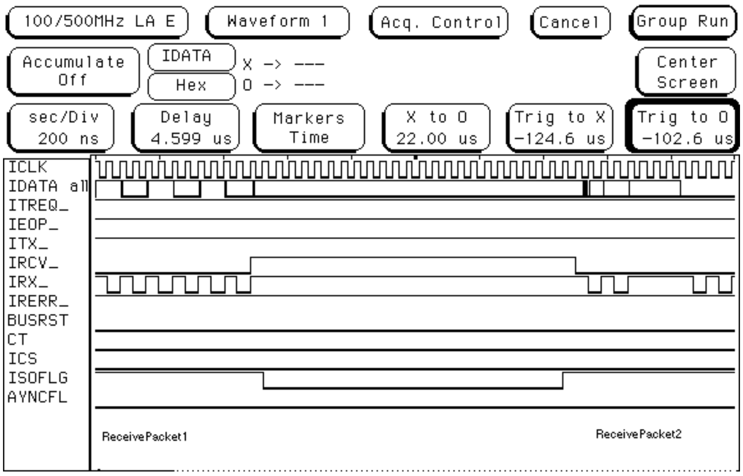
Accumulate Off
IDATA Hex
X -> ---
O -> ---
sec/Div 1.00 us
Delay 2.703 us
X to O 22.00 us
Trig to X -124.6 us
Trig to O -102.6 us
IDATA all

Figure 6-2-3 Isochronous reception (200MBPS)



Accumulate Off
IDATA Hex
X -> ---
O -> ---
sec/Div 1.00 us
Delay 2.703 us
X to O 22.00 us
Trig to X -124.6 us
Trig to O -102.6 us
IDATA all

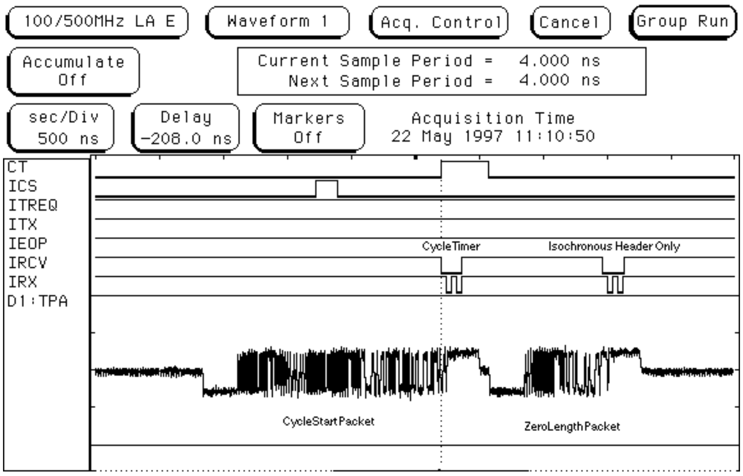
Figure 6-2-4 Isochronous 2ch reception (200MBPS)



Accumulate Off
IDATA Hex
X -> ---
O -> ---
sec/Div 200 ns
Delay 4.599 us
X to O 22.00 us
Trig to X -124.6 us
Trig to O -102.6 us
IDATA all

Figure 6-2-5 Enlarged Waveforms for Isochronous 2ch Reception (200Mbps)

Fig. 6-2-6 shows the waveforms when an isochronous packet of Length=0 is received under the condition of IsoMode= "00" or "01" (normal mode). When a packet of Length=0 is received, the IRCV signal is asserted and the IRX signal is also asserted for the amount of 1 quadlet (twice). In this case, only the header part of the isochronous packet is output at IDATA [15:0].



Accumulate Off

Current Sample Period = 4.000 ns
Next Sample Period = 4.000 ns

sec/Div 500 ns

Delay -208.0 ns

Markers Off

Acquisition Time
22 May 1997 11:10:50

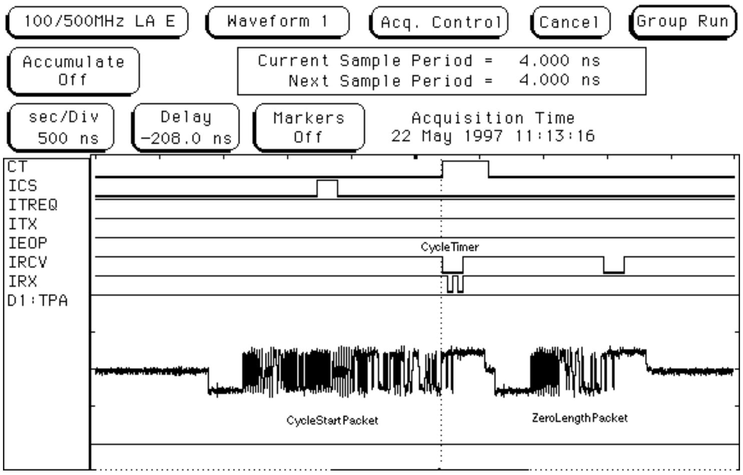
Figure 6-2-6 Reception of Isochronous ZiroLengthPacket (Normal mode)

6-2-2 Receiving transactions in AUTO mode

In the AUTO mode, a maximum of 4 channels can be received by setting the receiving conditions in the registers specified below. Since the isochronous header is removed by the MD8413 itself in this mode, an actual payload block only is output from the isochronous bus.

Channel out of the data (channel, tag, length) stored in the isochronous header is output from an exclusive pin (CH) of the MD8413. Length can be defined from the frequency of IRX# assertion while IRCV# is asserted.

Fig. 6-2-1 shows the waveforms observed when an isochronous packet of Length=0 is received under the condition of IsoMode="10" or "11" (AUTO mode). When a packet of Length=0 is received, the IRCV signal is asserted. In the AUTO mode, however, no output is generated from IDATA as the isochronous packet header is separated inside the MD8413.



Accumulate Off

Current Sample Period = 4.000 ns
Next Sample Period = 4.000 ns

sec/Div 500 ns

Delay -208.0 ns

Markers Off

Acquisition Time 22 May 1997 11:13:16

Figure 6-2-1 Reception of Isochronous Zero Length Packet (Auto mode)

6-2-3 Reception control by Sync field

TBD

7. Phy register access

The internal register of Phy stores a variety of information such as node number, root data, GapCount, etc., determined after the completion of SelfD phase. After bus reset, the node number must be acquired first of all. At that time, data are read out by the use of the register for gaining access to the Phy.

Transactions for accessing to the connected Phy are explained below.

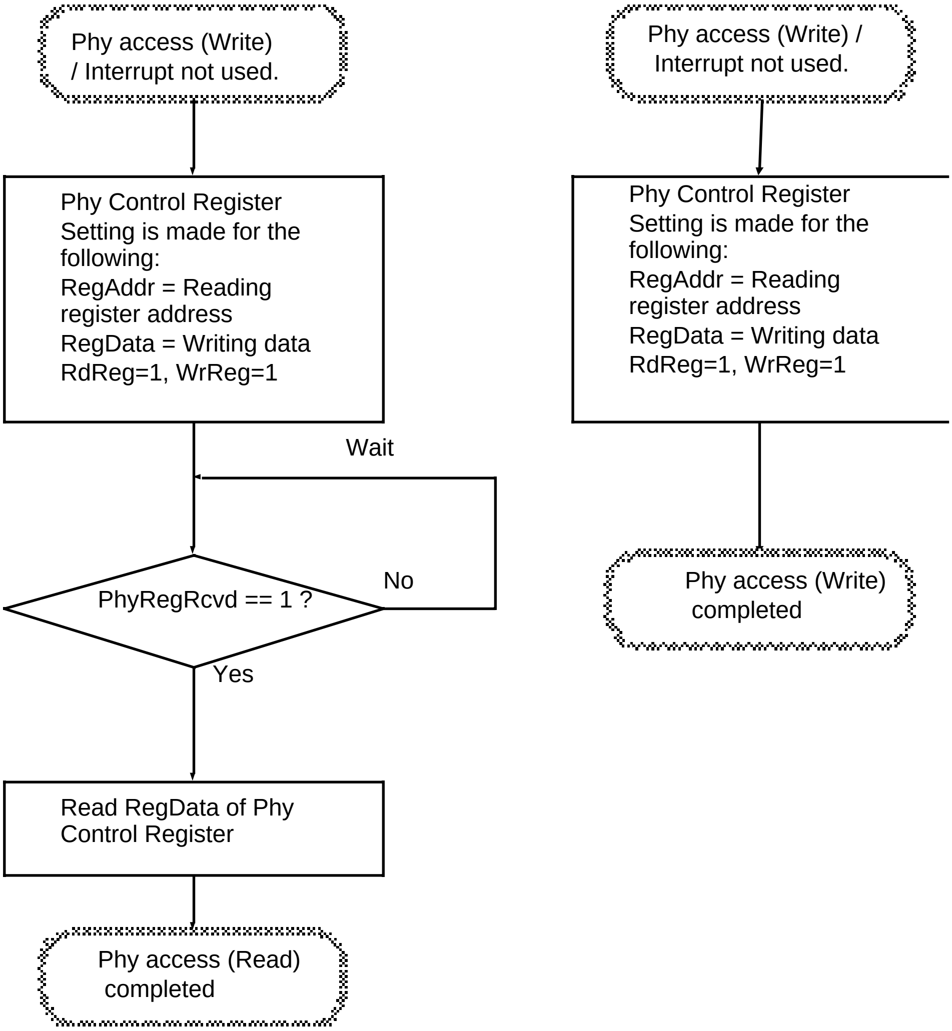

Phy access (Write) / Interrupt not used.
Phy Control Register Setting is made for the following: RegAddr = Reading register address RegData = Writing data RdReg=1, WrReg=1
Wait
PhyRegRcvd == 1 ?
No
Yes
Read RegData of Phy Control Register
Phy access (Read) completed
Phy access (Write) / Interrupt not used.
Phy Control Register Setting is made for the following: RegAddr = Reading register address RegData = Writing data RdReg=1, WrReg=1
Phy access (Write) completed

Figure 7-1 Method of Phy access

7-1 Reading request

It may be necessary to send out a read-out request to the Phy. In such a case, own node number and root data may be acquired after bus reset.

As shown in Fig. 7-1, the internal register address is set in RegAddr for the Phy from which data are being read out. When RdReg is set for "1", it is possible to issue a read-out request for this Phy. At that time, the RegData block is disabled.

When writing is attempted in Phy control register in this setting, a read-out request command is issued toward the Phy and data is output from the Phy to the MD8413. Then the MD8413 sets up the PhyRegRcvd of the interrupt register at "1".

After reception of this information at the host, the Phy control register begins to be read out and the resultant read-out value is stored in the RegData block.

7-2 Writing request

When requesting the Phy for writing, GapCount may be set up.

As shown in Fig. 7-1, the internal register address is set in RegAddr for the Phy in which data are being written and the writing data are set in RegData. When WrReg is set for "1", it is possible to issue a write-in request for this Phy.

8 LinkOn

8-1 LinkOn packet transmission

The LinkOn signal output is given from the connected PHY (MD8402). Since the output signals are not handled by the MD8413, such handling must be carried out on the system side. Transactions for LinkOn packet transmission toward another PHY are described below.

When the data shown below are stored in the ATF and transmitted, it is possible to send a LinkOn packet to the PHY with the corresponding node number. Contents of the transmission packet are as follows:

•0x000000e0	Identification data for control packet transmission
•0x41000000	LinkOn for Phy with NodeID=1
•0xbeffffff	Logical reversal data for the above data

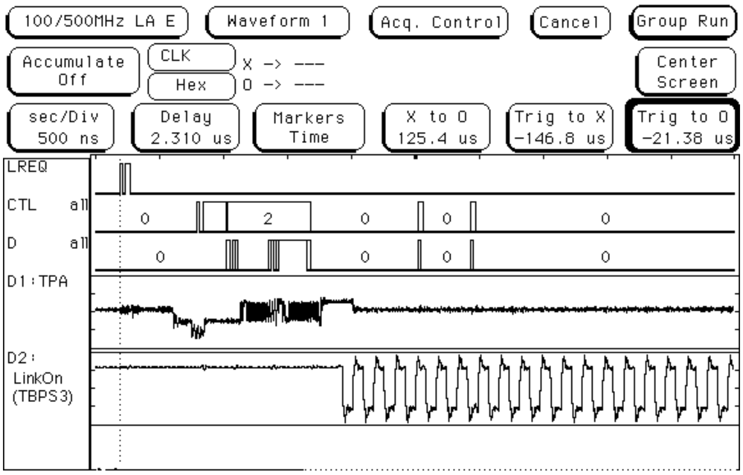
The above-mentioned first quadlet is for the identification data of the control packet. The second and third quadlets are actually used as the LinkOn packets. Refer to IEEE1394-1955 regarding the format of LinkOn packet.

8-2 LinkOn packet reception

As shown in Fig. 8-2-1, the received LinkOn packet is generated from the LinkOn signal of the MD8413, terminal name TBPS3, based on the clock signal. The output conditions for the LinkOn signal are as follows:

(1) LinkPowerStatus (EPSTAT)='L'

Therefore, when the MD8413 is in normal operation, LPS of the MD8402 is maintained at 'H' and therefore the LinkOn signal output is not generated at that time.



Accumulate Off

CLK X -> ---

Hex O -> ---

sec/Div 500 ns

Delay 2.310 us

Markers Time

X to O 125.4 us

Trig to X -146.8 us

Trig to O -21.38 us

Figure 8-2-1 Received LinkOn Packet and LinkOn (TBPS3) Signal

9 PHY Configuration Packet

9-1 PHY configuration packet transmission

The Phy control packet can be transmitted by storing and transmitting the following data in the ATF:

Contents of transmission packet:

•0x000000e0	Identification data for control packet transmission
•0x01410000	Writing GapCount=1 for Phy with NodeID=1
•0xfebeffff	Logical reversal data for the above data

As a result of transmission, the gap count of the Phy with NodeID=1 is modified. The first quadlet is for the identification data of the control packet. The second and third quadlets are actually used as the PHY configuration packets. Refer to IEEE1394-1995 regarding the format of PHY configuration packet.

10. LPS (Link Power Status)

The LPS signal output is generated from the LINK chip and sent to the connected PHY chip. With this signal, the PHY chip can know whether the LINK chip is active or non-active.

The LPS signal is controlled by the LPSOn bit in the control register. In addition, the contents of this output can differ according to the status of the DIRECT terminal that determines the condition of PHY-LINK connection (direct or isolated).

Usually the LPS terminal is connected to the LPS terminal of the PHY chip. However, when the LPSOn bit is "0" in the control register, the PHY is informed that the LINK chip is non-active. Thus the SCLK signal output (49.152MHz) supplied from the PHY is suspended.

The relationship among DIRECT terminals, LPSOn bits, and LPS terminal output is shown below.

DIRECT pin	LPSOnbit	LPS pin Output
High	0	Low
High	1	High
Low	0	Low
Low	1	2MHz

Receiving the LPS data, the PHY chip stores information in the link_active field of the Self_IDPacket, regarding whether its own node is LINK-active or non-active. Then it transmits this information to another node.

The Self_IDPacket format is shown below.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1110b				0	0	0	0

Table10-1 SelfID Packet format(frst quadlet)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	phy_ID						0	L	gap_cnt						sp		del		C	pwr			p0		p1		p2		i	m
logical inverse of first quadlet																															

Table10-2 SelfID Packet format(SelfID Packet #0)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	phy_ID							1	n		rsv		pa		pb		pc		pd		pe		pf		pg		ph		r	m
logical inverse of first quadlet																															

Table10-3 SelfID Packet format(SelfID Packet #1, #2, & #3)

	n	pa	pb	pc	pd	pe	pf	pg	ph
pkt #1	0	p3	p4	p5	p6	p7	p8	p9	p10
pkt #2	1	p11	p12	p13	p14	p15	p16	p17	p18
pkt #3	2	p19	p20	p21	p22	p23	p24	p25	p26

Table10-5 SelfID Packet format(pn)

phy_ID : physical_ID field

Node ID of the PHY chip of this node

L : link_active field

0 = LINK is non-active.

1 = This node has an active LINK and transaction layer.

gap_cnt : gap_count field

Present value is stored in regard to the PHY_CONFIGURATION gap_count field of this node.

sp : PHY_SPEED field

00 = 98.304Mbps

01 = 98.304 and 196.608Mbps

10 = 98.304, 196.608 and 393.216Mbps

11 = Reserved

Available and possible speed is stored.

del : PHY_DELAY field

00 = below 144ns (~14/BASE_RATE)

01~11 = Rreserved

Delay time of the repeater in the worst case is stored.

C : CONTENDER field. CMC/LINKON terminal setting.

When this field is set and the link_active field is also set, such a condition indicates that this node has become a bus or isochronous resource manager.

pwr: : POWER_CLASS field. PC[2:0] terminal setting.

000 =The node does not require any power supply.

001 =The node has its own power supply, capable of supplying a minimum of 15W.

010 =The node has its own power supply, capable of supplying a minimum of 30W.

011 =The node has its own power supply, capable of supplying a minimum of 45W.

100 =The node is fed power from the cable, and consumes a maximum of 1W.

101 =The node is fed power from the cable, and consumes a maximum of 1W.

In addition, it consumes up to 2W to enable a link or a layer in higher hierarchy.

110 =The node is fed power from the cable, and consumes a maximum of 1W.

In addition, it consumes up to 5W to enable a link or a layer in higher hierarchy.

111 =The node is fed power from the cable, and consumes a maximum of 1W.

In addition, it consumes up to 9W to enable a link or a layer in higher hierarchy.

p0 ... p26 : NORT,child[NPORT],connected[NPORT] field

11 =Connected to the child node.

10 =Connected to the parent node.

01 =Not connected to another PHY.

11 =This PHY is not offered.

The port status is shown.

i : initiated_reset field

When setting is made, this node issues a bus reset for this time.

m : more_packets field

When setting is made, there is another SelfID packet following for this node.

n : Extended field

Extension SelfID packet sequence number (any value between 0 to 2).

r,rsv : reserved field

Reserved.

11. PHY_LINK interface

This section provides a description about some examples of PHY-LINK interface connections, based on the examples for the PHY chip MD8402 and the LINK chip MD8413.

11-1 Example of MD8413-MD8402 interface connections (DC)

Shown below is an example of recommended connections for the PHY-LINK interface of the LINK chip MD8411 that employs the PHY chip MD8401. The MD8411 supports the PHY-LINK interface in DC connections.

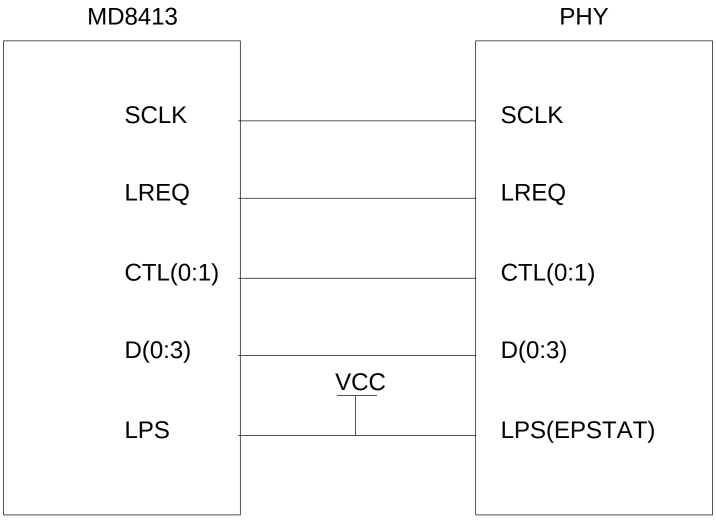


Figure 10-1-2 Connection diagram for the MD8413 and PHY chip (DC connection)

11-2 Example of MD8413-MD8402 interface connections (AC)

Shown below is an example of recommended connections for the PHY-LINK interface of the LINK chip MD8413 that employs the PHY chip MD8402. The MD8413 supports the PHY-LINK interface in AC and DC connections.

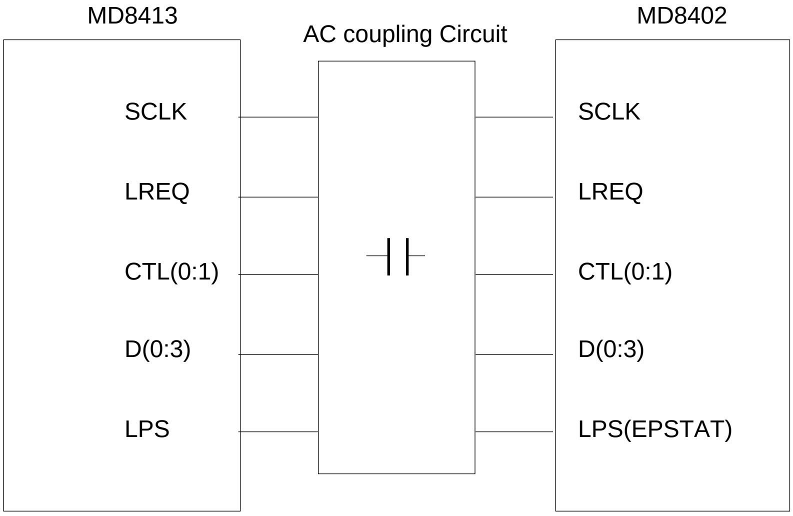
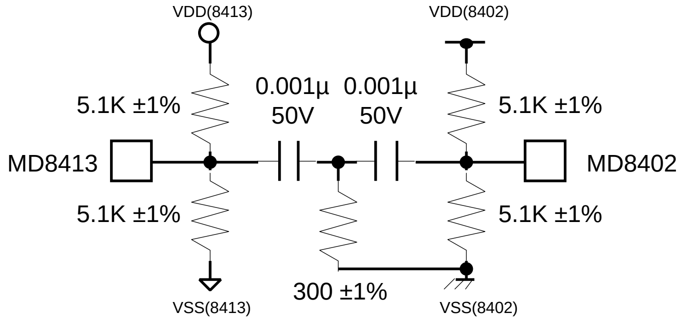


Figure 10-1-3 Connection diagram for the MD8413 and MD8402 chip (DC connection)

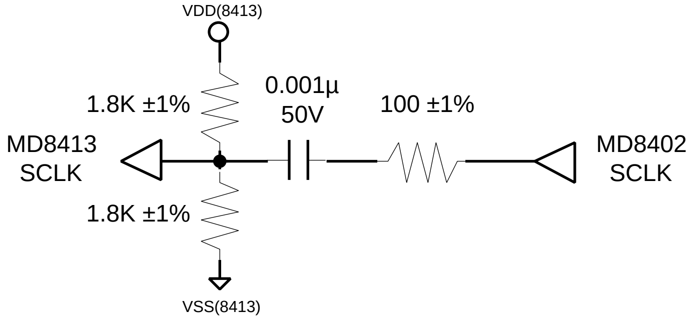
11-2-1 Isolation barrier circuit diagram

Diagrams of isolation barrier circuits are shown below, applicable when the PHY-LINK interface is in AC connections for the PHY chip MD8402 and the LINK chip MD8413.

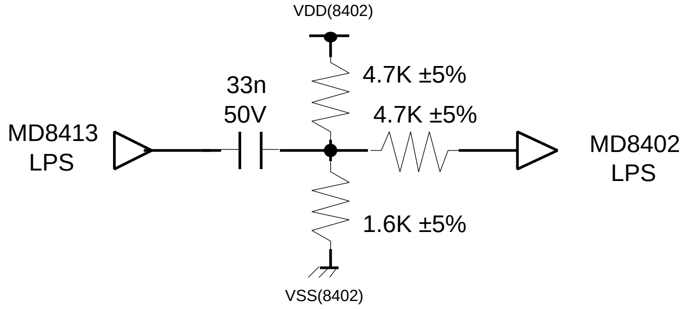
1)LREQ,CTL(1:0),D(3:0)



2)SCLK



3)LPS



12. Method of external CycleTimer structuring

The MD8413 has a function of sending out its CycleTimer value from the exclusive isochronous bus, at a timing for the transmission or reception of a CycleStart packet. When this function is used, it is possible to establish the same CycleTimer as that on the bus of IEEE1394, according to the external logic. Fig. 12-1 shows a block diagram of CycleTimer constructed with the external logic.

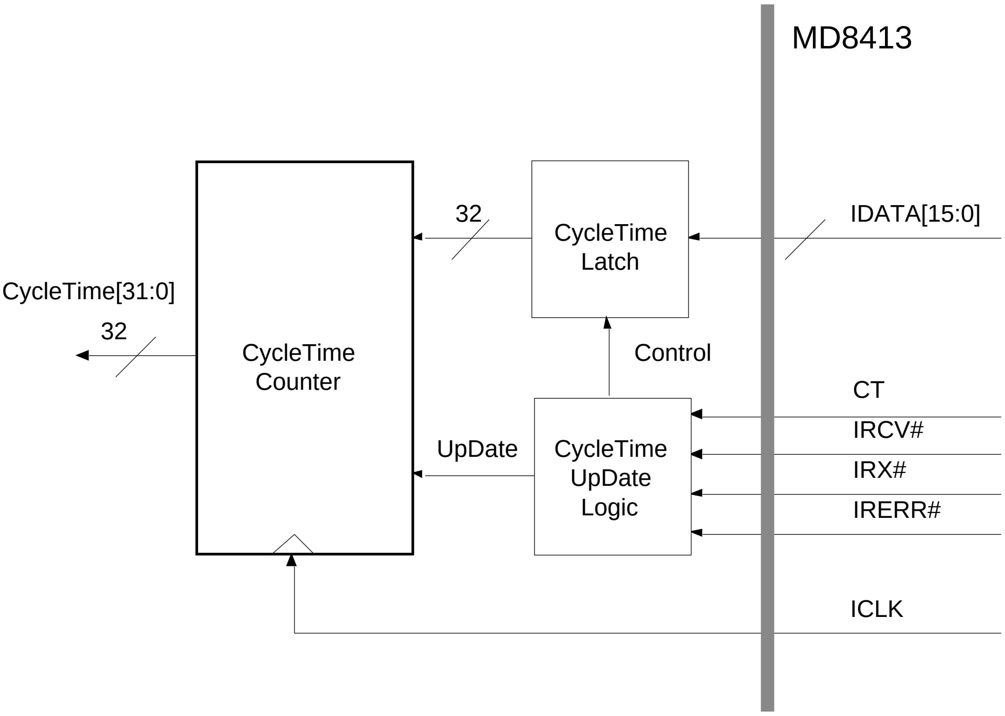


Figure 12-1 Cycle Timer

When the above block is installed in an external circuit, it is then possible to get the same CycleTime [31:0] as the CycleTime value possessed by the connected MD8413.

In an application where TimeStamp transaction is needed, it is possible to obtain a TimeStamp value from this CycleTimer and generate a required signal based on the condition of coincidence between the received TimeStamp value and this CycleTimer.

13. MD8413 deficiencies and notes for usage

The current version mentioned in this section denotes the MD8413 Version No. 2, Revision No. 0.

13-1 DualPhase retry deficiency

For retry operation of the dual phase on OutBound side, there must be an environment where a CycleStart packet is transmitted (CycleMaster) or received. In other words, a CycleStart packet need be present on the bus. Otherwise, retry operation will be repeated endlessly.

Generally, isochronous transfer is regarded as optional in the IEEE1394 Standard. Therefore, if asynchronous transfer only is intended, dual-phase retry cannot be effected on the OutBound side because of this deficiency.

If a bus reset occurs and the route is changed, transmission of a CycleStart packet is suspended on the bus. After the lapse of 625mS, the isochronous resource manager or the full-bus manager begins to function and gives an order to transmit a CycleStart packet to the route. Thus the CycleStart packet flows in the bus. In other words, for the duration of 625mS after BusReset, there is no CycleStart packet. If a dual-phase retry occurs in this duration, retry operation is kept continued for 625mS.

This problem can be solved by disabling the phase retry in the state that no CycleStart packet is flowing.

13-2 Deficiency in isochronous Sync control reception

Due to the Sync field value during isochronous reception, there is no operation of a logic which controls the start or end of reception. According to the specification, reception is started with a packet having a Sync field that coincides with the StartSync, by setting the SyncEn bit of the isochronous receive configuration register. This reception continues till the packet having a Sync field that coincides with the StopSync. When reception is started under the condition that isochronous data are already flowing on the bus, all packets are disregarded until the reception of a packet having a Sync field that coincides with the first StartSync.

Because of such a deficiency, it is impossible to control whether reception should be effected with Sync. However, normal operation is effected with SyncEn=0.

13-3 Deficiency in the RegRcvd bit of the Phy control register (0x1c)

In the case of read request toward the Phy, the RegRcvd bit may be used as a bit intended to indicate that the data from the Phy have been stored in the RegData. In some cases, however, this bit is not set up. Accordingly, there can be an infinite loop if a logic has been arranged to wait for the setting by polling this RegRcvd bit only.

As a measure to solve this problem, provisions should be made to utilize the PhyRegRcvd bit of the interrupt register (0x56) that has a similar function. This bit should be used for polling or interrupt transaction.

13-4 Deficiency in the bus timer register (0x2c, 0x2e)

This register offers the same operation of the BusTime register specified by IEEE1394-1995. At present, however, it offers an unstable value. When constructing a BusTime register in accordance with IEEE1394-1995, utilize the CycleSecond bit that causes an interrupt every second in the interrupt register. Using this timing, construct a BusTime register on the microcomputer side.

Note

Contents of this manual may be modified without notice.

Partial or overall reproduction and reprinting of this manual shall be prohibited without permission of the company.

Owing this manual shall not fall out permission for the use of industrial and copy rights reserved by the company and Fuji Photo Film Co., Ltd. The company shall not be responsible for industrial and copy rights of the third party.

The company shall not be responsible for any damage caused from this manual and from this chip.

Prior to use, exchange of invoice specifications is required.

Information available from:

FUJIFILM MICRODEVICES Co., Ltd.
*Head Office
1-6, Matsusakadaira, Taiwa-cho
Kurokawa-gun, MIYAGI, 981-34 JAPAN
TEL: 81-22-347-1128 FAX: 81-22-347-1136
*Tokyo Office
Fukumasu Bldg, 6F
1-20 Ageba-cho, Shinjuku-ku
Tokyo, 106 JAPAN
TEL: 81-3-3269-2141 FAX: 81-3-5229-7381