SONY

CXD1948R

IEEE1394 Link Layer LSI for DVB and DSS

For the availability of this product, please contact the sales office.

Description

The CXD1948R is a Link Layer LSI conforming to the IEEE1394 serial bus standard. During transmission, the MPEG2 transport stream is time stamped, transformed to IEEE1394 format and sent to the IEEE1394 Phy IC.

During reception, the signal from IEEE1394 is kept in the built-in FIFO, synchronized to the time stamp value and output.

This IC utilizes Apple Computer's Fire Wire technology.

Features

- Conforms to IEEE1394 serial bus standard
- Supports DVB and DSS transport streams
- Dedicated ports for asynchronous data/isochronous data
- Isochronous data inserted from asynchronous data port
- Smoothing buffer function
- Large capacity FIFO

Isochronous transmit/receive FIFO	960 × 32 bits
Asynchronous transmit FIFO	30 × 33 bits
Asynchronous receive FIFO	36 × 33 bits
Isochronous Insert Packet Transmit Buffer	47 × 33 bits

- CIP header automatic attachment/detection

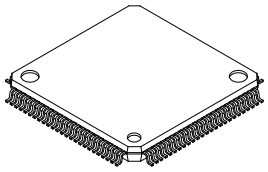
Applications

- Digital interface for D-STB
- Digital interface for D-VHS

Structure

Silicon gate CMOS IC

100 pin LQFP (Plastic)



Absolute Maximum Ratings (Ta = 25°C)

• Supply voltage	V_{DD}	Vss – 0.5 to +4.6	V
• Input voltage	V_I	Vss – 0.5 to V_{DD} + 0.5	V
• Output voltage	V_O	Vss – 0.5 to V_{DD} + 0.5	V
• Operating temperature	Topr	–20 to +75	°C
• Storage temperature	Tstg	–55 to +150	°C

Recommended Operating Conditions

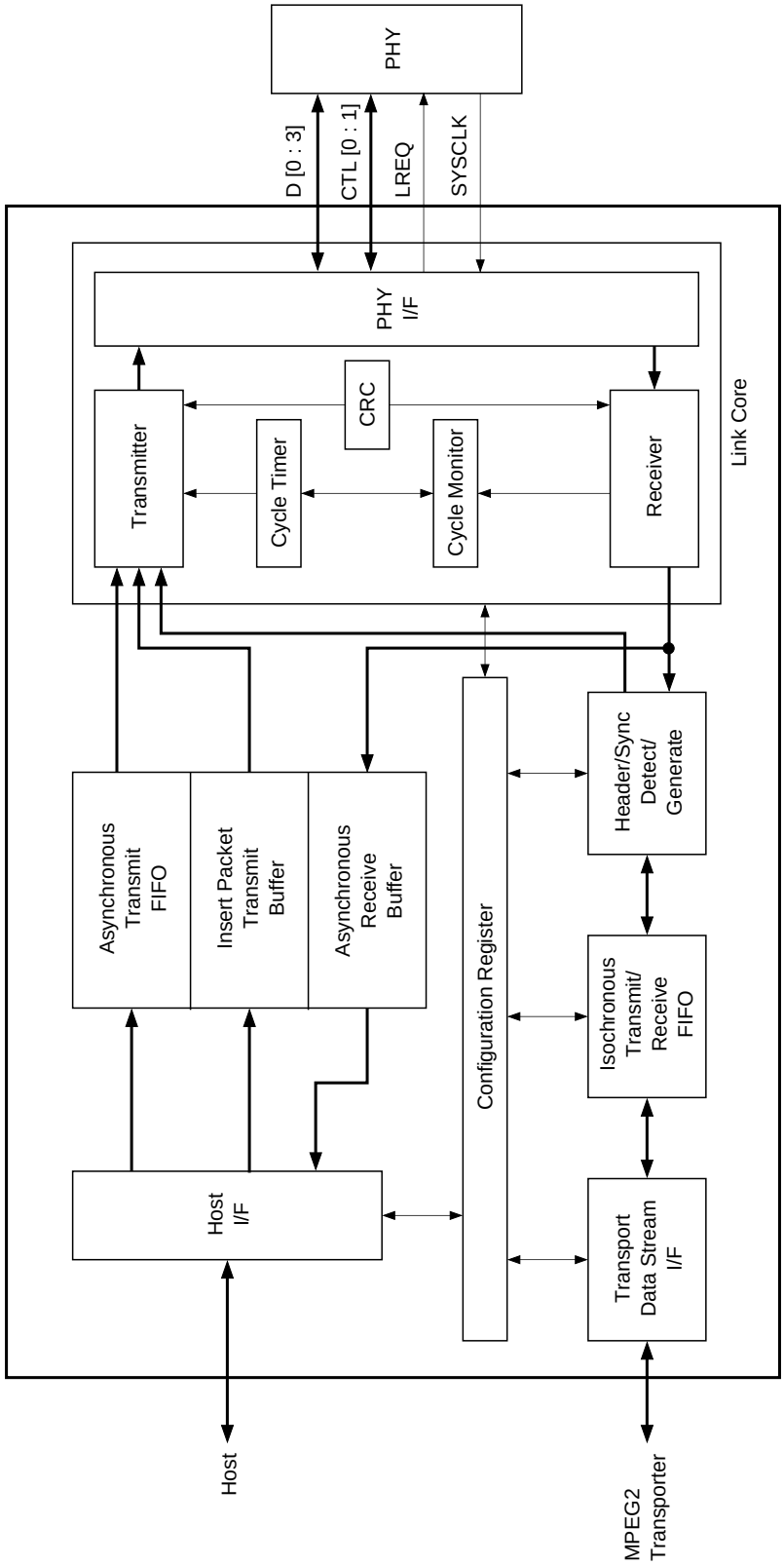
• Supply voltage	V_{DD}	3.0 to 3.6	V
• Operating temperature	Topr	–20 to +75	°C

Fire Wire is a registered trademark of Apple Computer Corporation, USA

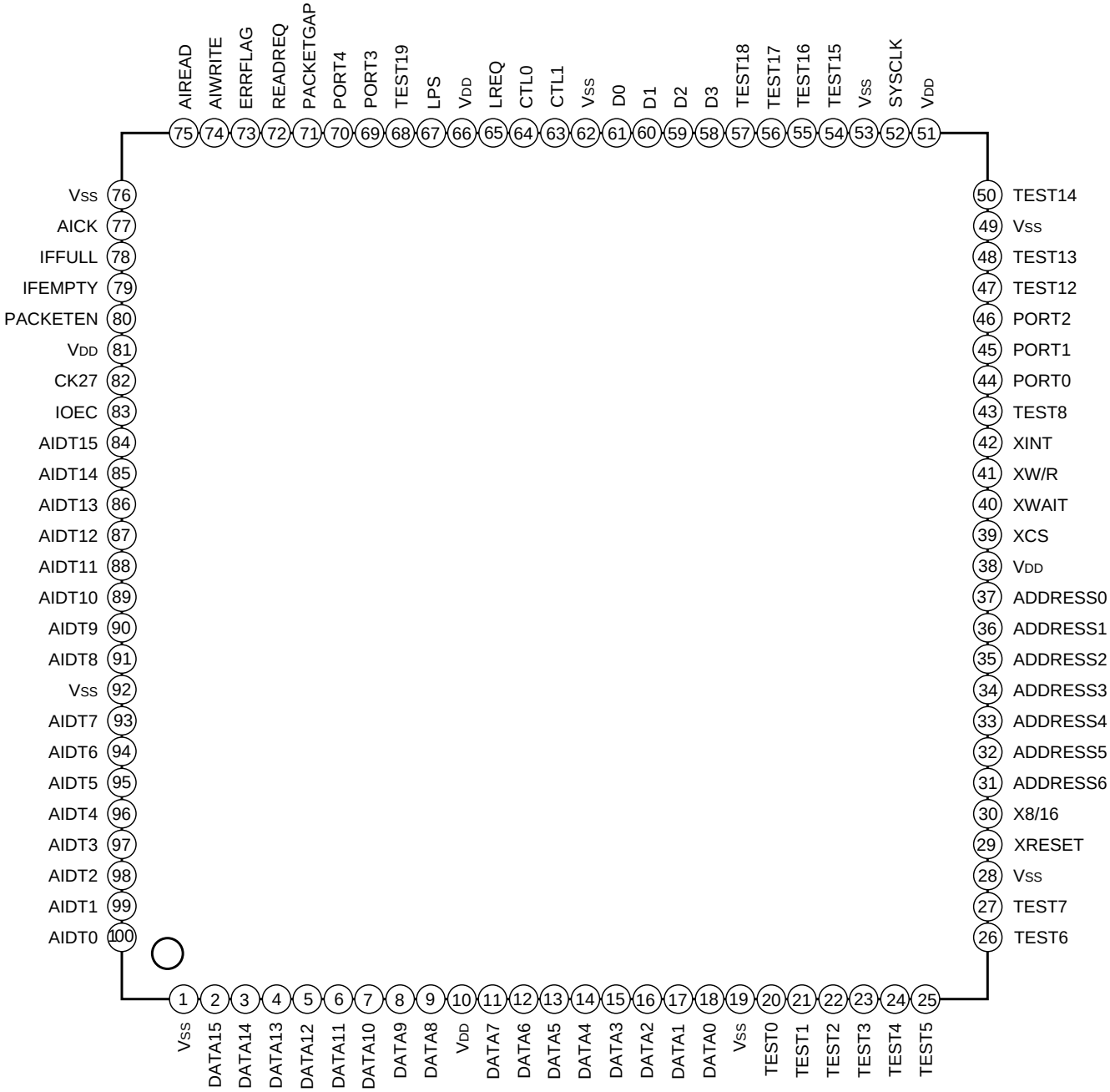
E96844-PS

Contents

1. Block Diagram



2. Pin Configuration



3. Pin Description

Pin No.	Symbol	I/O	Description
1	Vss	—	GND
2	DATA15	I/O	I/O data with host I/F BIT 15
3	DATA14	I/O	I/O data with host I/F BIT 14
4	DATA13	I/O	I/O data with host I/F BIT 13
5	DATA12	I/O	I/O data with host I/F BIT 12
6	DATA11	I/O	I/O data with host I/F BIT 11
7	DATA10	I/O	I/O data with host I/F BIT 10
8	DATA9	I/O	I/O data with host I/F BIT 9
9	DATA8	I/O	I/O data with host I/F BIT 8
10	V_{DD}	—	Power supply
11	DATA7	I/O	I/O data with host I/F BIT 7
12	DATA6	I/O	I/O data with host I/F BIT 6
13	DATA5	I/O	I/O data with host I/F BIT 5
14	DATA4	I/O	I/O data with host I/F BIT 4
15	DATA3	I/O	I/O data with host I/F BIT 3
16	DATA2	I/O	I/O data with host I/F BIT 2
17	DATA1	I/O	I/O data with host I/F BIT 1
18	DATA0	I/O	I/O data with host I/F BIT 0
19	Vss	—	GND
20	TEST0	—	Test pin*
21	TEST1	—	Test pin*
22	TEST2	—	Test pin*
23	TEST3	—	Test pin*
24	TEST4	—	Test pin*
25	TEST5	—	Test pin*
26	TEST6	—	Test pin*
27	TEST7	—	Test pin*
28	Vss	—	GND
29	XRESET	I	Master reset signal 0: Active; 1: Non-active
30	X8/16	I	I/O data with host I/F bus select signal 0: 8 bit; 1: 16 bit
31	ADDRESS6	I	Host I/F address bus BIT 6
32	ADDRESS5	I	Host I/F address bus BIT 5
33	ADDRESS4	I	Host I/F address bus BIT 4
34	ADDRESS3	I	Host I/F address bus BIT 3

* The test pins should be used open.

Pin No.	Symbol	I/O	Description
35	ADDRESS2	I	Host I/F address bus BIT 2
36	ADDRESS1	I	Host I/F address bus BIT 1
37	ADDRESS0	I	Host I/F address bus BIT 0
38	V_{DD}	—	Power supply
39	XCS	I	Host I/F chip select signal 0: Active; 1: Non-active
40	XWAIT	O	Host I/F wait signal 0: Active; 1: Non-active
41	XW/R	I	Host I/F write/read signal 0: Write; 1: Read
42	XINT	O	Host I/F interrupt signal 0: Active; 1: Non-active
43	TEST8	—	Test pin*
44	PORT0	I/O	Parallel I/O port BIT0
45	PORT1	I/O	Parallel I/O port BIT1
46	PORT2	I/O	Parallel I/O port BIT2
47	TEST12	—	Test pin*
48	TEST13	—	Test pin*
49	V_{SS}	—	GND
50	TEST14	—	Test pin*
51	V_{DD}	—	Power supply
52	SYSCLK	I	Phy I/F system clock (49.152MHz)
53	V_{SS}	—	GND
54	TEST15	—	Test pin*
55	TEST16	—	Test pin*
56	TEST17	—	Test pin*
57	TEST18	—	Test pin*
58	D3	I/O	Phy I/F data bus BIT 3
59	D2	I/O	Phy I/F data bus BIT 2
60	D1	I/O	Phy I/F data bus BIT 1
61	D0	I/O	Phy I/F data bus BIT 0
62	V_{SS}	—	GND
63	CTL1	I/O	Phy I/F control bus BIT 1
64	CTL0	I/O	Phy I/F control bus BIT 0
65	LREQ	O	Phy I/F request signal
66	V_{DD}	—	Power supply
67	LPS	O	Phy I/F Link power status signal
68	TEST19	—	Test pin*

* The test pins should be used open.

Pin No.	Symbol	I/O	Description
69	PORT3	I/O	Parallel I/O port BIT3
70	PORT4	I/O	Parallel I/O port BIT4
71	PACKETGAP	I	Timing signal for adding time stamp to inserted isochronous packet 0: Non-active; 1: Active
72	READREQ	O	Received packet synchronization reference signal 0: Non-active; 1: Active
73	ERRFLAG	I/O	Packet error signal 0: Non-active; 1: Active
74	AIWRITE	I	Transport stream I/F data write enable signal 0: Non-active; 1: Active
75	AIREAD	I	Transport stream I/F data read enable signal 0: Non-active; 1: Active
76	V_{SS}	—	GND
77	AICK	I	Transport stream I/F clock
78	IFFULL	O	Isochronous FIFO status output 1: Full
79	IFEMPTY	O	Isochronous FIFO status output 1: Empty
80	PACKETEN	I/O	Transport stream I/F packet enable signal 0: Non-active; 1: Active
81	V_{DD}	—	Power supply
82	CK27	I	Clock input for 27MHz time stamp (open when not in use)
83	IOEC	I	Transport stream I/F data bus control signal 0: Input; 1: Output
84	AIDT15	I/O	Transport stream I/F data bus BIT 15
85	AIDT14	I/O	Transport stream I/F data bus BIT 14
86	AIDT13	I/O	Transport stream I/F data bus BIT 13
87	AIDT12	I/O	Transport stream I/F data bus BIT 12
88	AIDT11	I/O	Transport stream I/F data bus BIT 11
89	AIDT10	I/O	Transport stream I/F data bus BIT 10
90	AIDT9	I/O	Transport stream I/F data bus BIT 9
91	AIDT8	I/O	Transport stream I/F data bus BIT 8
92	V_{SS}	—	GND
93	AIDT7	I/O	Transport stream I/F data bus BIT 7
94	AIDT6	I/O	Transport stream I/F data bus BIT 6
95	AIDT5	I/O	Transport stream I/F data bus BIT 5
96	AIDT4	I/O	Transport stream I/F data bus BIT 4
97	AIDT3	I/O	Transport stream I/F data bus BIT 3
98	AIDT2	I/O	Transport stream I/F data bus BIT 2
99	AIDT1	I/O	Transport stream I/F data bus BIT 1
100	AIDT0	I/O	Transport stream I/F data bus BIT 0

* The test pins should be used open.

4. Electrical Characteristics

4-1. DC Characteristics

(Ta = 25°C, V_{SS} = 0V)

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
Input voltage	V_{IH}	CMOS input cell	0.7V_{DD}			V
Input voltage	V_{IL}	CMOS input cell			0.2V_{DD}	V
Output voltage	V_{OH}	I_{OH} = –4.0mA	V_{DD} – 0.4			V
Output voltage	V_{OL}	I_{OL} = 4.0mA			0.4	V
Input leak current	I_{I1}	Bidirectional pin (input state)	–40		40	µA
Input leak current	I_{I2}	Normal input pin	–10		10	µA
Output leak current	I_{OZ}	Tri-state pin (for high impedance state) V_{IN} = V_{SS} or V_{DD}	–40		40	µA

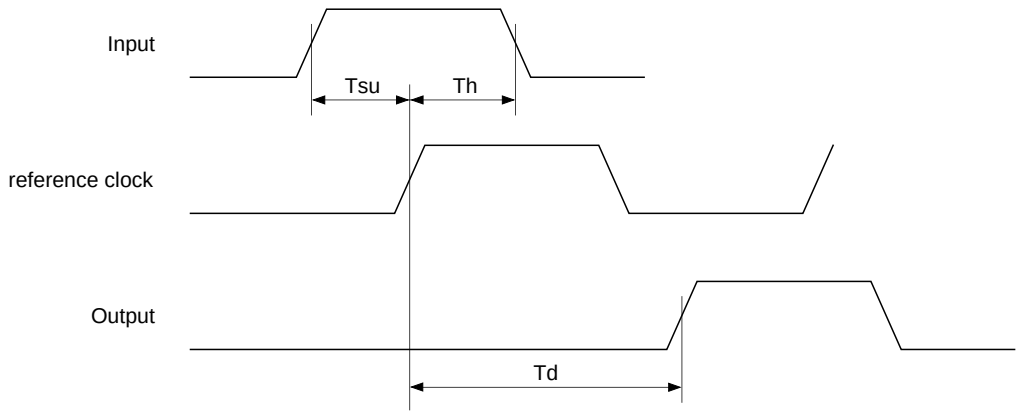
4-2. AC Characteristics

(V_{DD} = 3.0 to 3.6V)

Item	Applicable pin	Symbol	Normal clock	Conditions	Min.	Typ.	Max.	Unit
Input setup	AIDT [15 : 0], ERRFLAG, PACKETEN, IOEC, AIWRITE, AIREAD, PACKETGAP	Tsu_1	AICK AIREAD AIWRITE *1	CL = 10pF			10	ns
Input hold		Th_1					0	ns
Output delay	AIDT [15 : 0], ERRFLAG, READREQ	Td_1			5		40	ns
Input setup	D [3 : 0], CTL [1 : 0]	Tsu_2	SYSCLK	CL = 10pF	5			ns
Input hold		Th_2			2			ns
Output delay	D [3 : 0], CTL [1 : 0], LREQ	Td_2			2		15	ns
Input setup	ADDRESS [6 : 0], DATA [15 : 0], XCS, XWR	Tsu_3	Refer to P.31 ATF/CFR/IPF write timing ATF/CFR read timing					
Input hold		Th_3						
Output delay	DATA [15 : 0], XWAIT, XINT	Td_3						

*1 During asynchronous I/F, the reference clock is AIWRITE (transmit)/AIREAD (receive).

4-3. Timing Definitions



5. Isochronous Communication

The CXD1948R has a function which transmits and receives DVB and DSS transport stream data as isochronous packets.
The communication of transport stream data with the DVB and DSS systems is performed using a dedicated I/O data bus and several control signal pins. (See below.)
Further, it supports a wide variety of application I/F, including 8-bit data and 16-bit data, synchronous and asynchronous.

Name	Width	in/out	Description
AIDT	16	in/out	Transport stream data bus
PACKETEN	1	in/out	Indicates a valid packet (valid: 1; invalid: 0)
ERRFLAG	1	in/out	Indicates that the packet is an error (error: 1; no error: 0)
AIWRITE	1	in	Data write strobe signal
AIREAD	1	in	Data read strobe signal
READREQ	1	out	Packet read request signal
AICK	1	in	Clock for interface (during sync)

5-1. Built-in FIFO

The CXD1948R has a built-in dedicated FIFO for isochronous communication.
The FIFO capacity is 3840 bytes, and it can accumulate 20 packets worth of DVB transport stream.
In order to actually use the built-in FIFO, the bank structure must be set. This is performed by the CFR PacketBanks register.

5-2. Transport Stream Data I/F

5-2-1. Data Bus

The data interface is 8 bit/16 bit, and switching is done by accessing the CFR AIDT16 register. (The default value is 8 bit.)

5-2-2. Transport Packet Size

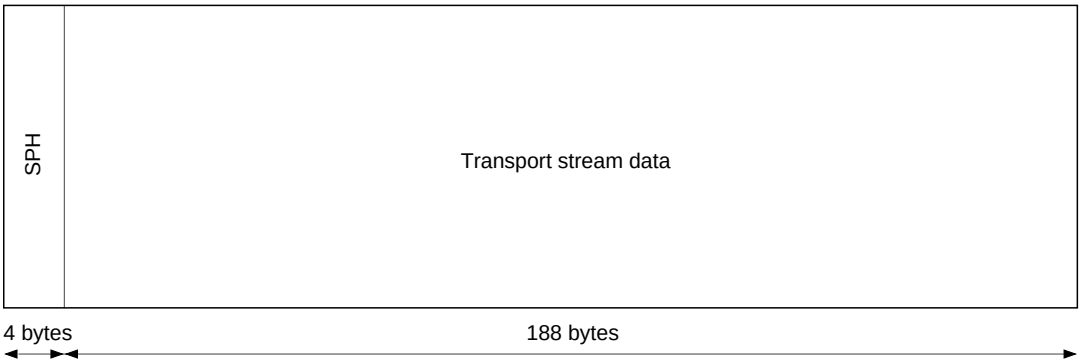
Data communication is done in transport stream packet units.
The data size for one packet can be set optionally by accessing the S_PacketSize register on the CFR.
S_PacketSize expresses the byte size of 1 source packet for isochronous communication.
For DVB specifications, this is 192 (data 188 bytes, source packet header 4 bytes).

5-2-3. Isochronous Additional Data

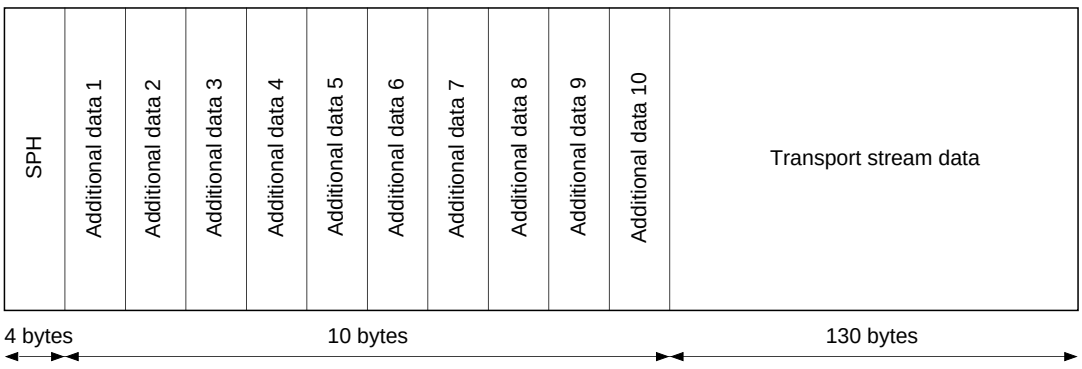
Optional data can be added to the transport stream packet when isochronous communication is performed.
Setting of the additional data is done by setting the CFR AddSize and AddData1 to AddData10 registers.
AddSize is a 4-bit register, and expresses the number of additional bytes for isochronous transmit/receive.
AddData1 to AddData10 are each 8-bit registers, and express the additional data for isochronous transmit/receive.
The default setting is no additional data.
The additional data is inserted between the source packet header and the transport stream data.
The smallest group of data that can be handed under IEEE1394 is 32 bits, so the setting must be such that the total of the transport stream data and additional data is a 32-bit unit.
An example is shown below.

Example of Isochronous Communication Setting

When S_PacketSize = 192 and AddSize = 0 (example of setting for DVB)



When S_PacketSize = 144 and TxAddSize = 10 (example of setting for DSS)



5-2-4. Transmit Interface

The CXD1948R is designed for use not only in transmitting an entire input transport stream, but also in transmitting one program only in the stream. The packet for transmission is selected through PACKETEN input as in the diagram below.

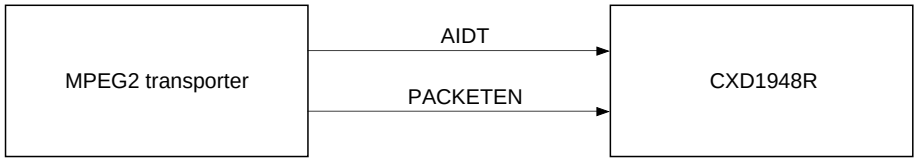
Moreover, constraints are imposed in the packet intervals of streams that can be input to the CXD1948R.

Each time the CXD1948R receives one packet of transport stream data, it adds time stamp data and data from the host I/F, and there must be an interval of 500ns or more for this processing. The timing is illustrated below.

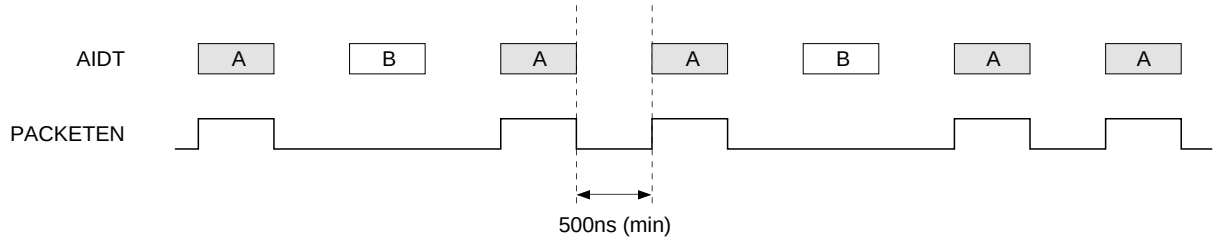
Also, the PACKETEN input signal must input “HIGH” while the first data of the transport stream packet is being sent. The maximum period for PACKETENABLE signal high is the period that the transport stream packet data is valid.

The timing charts for synchronous/asynchronous interface are shown below.

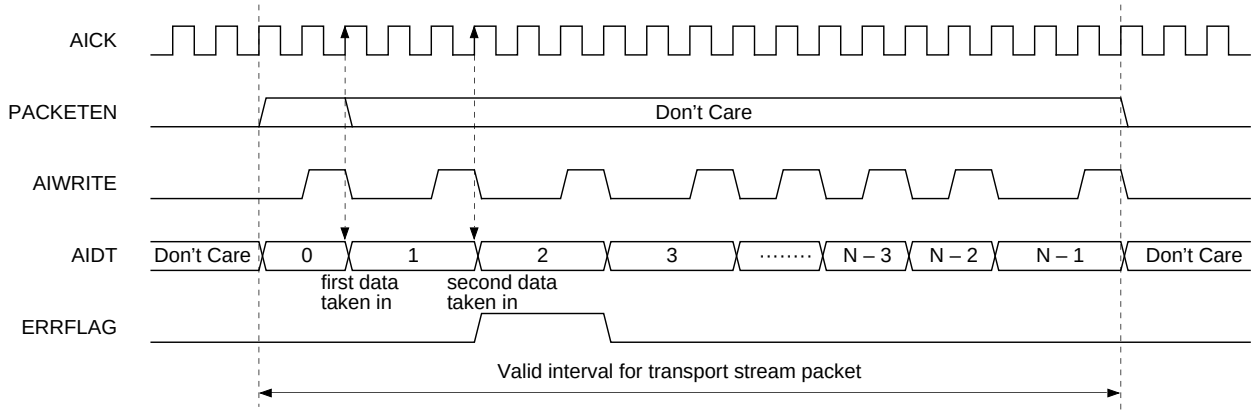
Transmit Interface Limits



Transmission of Program A



Transmit Interface (for sync mode/8 bit/Nbyte)

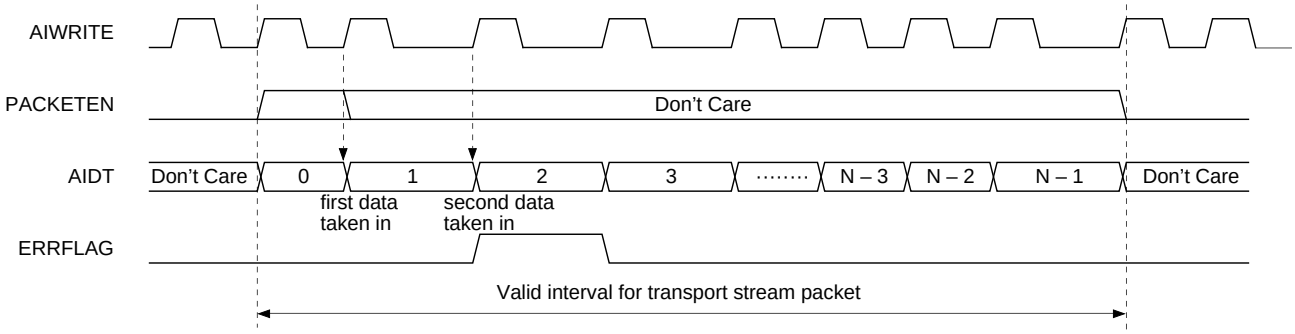


Sync interface mode is obtained by setting the CFR AsyncAI register to 0. (The default is 0.)
The CXD1948R identifies the first data of the transport stream packet by detecting that the PACKETEN signal has gone from low to high.
There is no particular need to input the PACKETEN signal after the first data. The size of the data taken in as valid data is equal to the value set in the CFR S_PacketSize register, decreased by 4 and by the value set in the AddSize register. The timing for taking in of the data internally is done by AICK rise when AIWRITE is high.
The AIWRITE signal is used as the enable signal, so the interval that the AIWRITE signal is high relative to one data must be one AICK clock interval. The interval that the AIWRITE signal is low relative to one data is not specified.
The limits on AICK input frequency are given below.

For 8-bit data input: 40MHz (Max.), 2MHz (Min.)
For 16-bit data input: 20MHz (Max.), 2MHz (Min.)

The ERRFLAG input during transmit can be made valid by setting the CFR ErrBitEnable register to 1. (The default is 0.)
The CXD1948R identifies the subject packet as an error if the ERRFLAG input is high for even one data during transport stream packet valid interval.
The switching timing for ERRFLAG input can be changed in the same way as data switching timing, with AIWRITE signal rise.

Transmit Interface (for asynchronous mode/8 bit/Nbyte)



Asynchronous interface mode is obtained by setting the CFR AsyncAI register to 1. (The default is 0.)
The CXD1948R identifies the first data of the transport stream packet by detecting that the PACKET_EN signal has gone from low to high.
There is no particular need to input the PACKET_EN signal after the first data.
The size of the data taken in as valid data is equal to the value set in the CFR S_PacketSize register, decreased by 4 and by the value set in the AddSize register. The timing for taking in of the data internally is done by AIWRITE rise.
The AIWRITE input is used as the clock, so there must be one AIWRITE input rising edge relative to one data. AIWRITE input duty is not specified, but the AIWRITE input must continue to be input evenly as the clock even outside of the transport stream packet valid interval.
The limits on AIWRITE input frequency are given below.

For 8-bit data input: 20MHz (Max.), 1MHz (Min.)
For 16-bit data input: 10MHz (Max.), 1MHz (Min.)

The ERRFLAG input during transmit can be made valid by setting the CFR ErrBitEnable register to 1. (The default is 0.)
The CXD1948R identifies the subject packet as an error if the ERRFLAG input is high for even one data during transport stream packet valid interval.
The switching timing for ERRFLAG input can be changed in the same way as data switching timing, with AIWRITE signal rise.

5-2-5. Receive Interface

The CXD1948R supports two modes; one in which data is output based on the time stamp value added to the transport stream packet during transmission, and one in which the data is output consecutively as soon as it is ready, without using the time stamp.

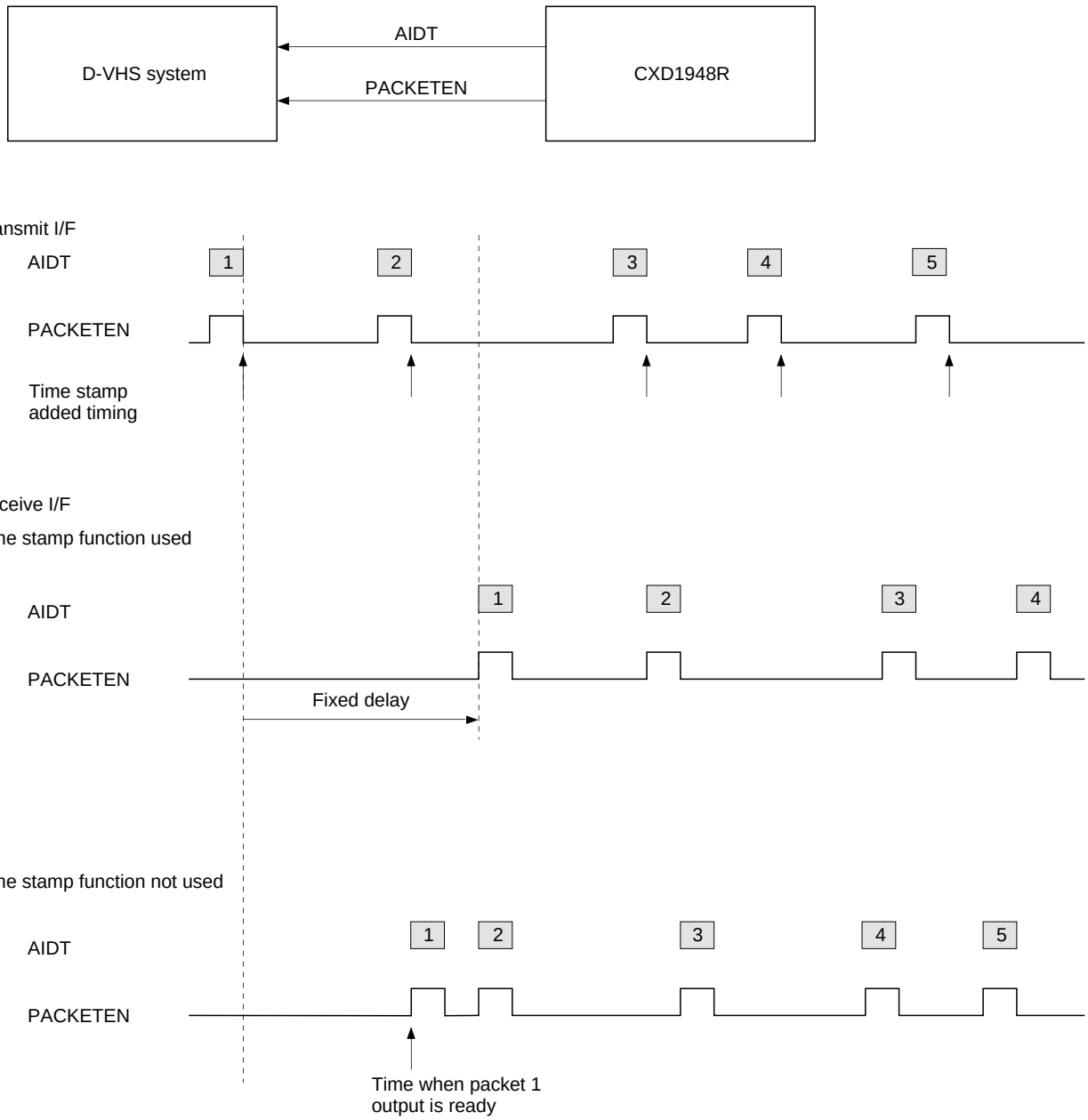
When output is based on the time stamp value, the actual output timing is done using a value which is the time stamp value with a fixed delay added. This is done in order to keep the packet intervals even on the transmit and receive sides.

When the time stamp value is not used for output, data output begins when the packet has completely arrived at the receive side. In this case, the time required for the packet to be sent and arrive completely at the receive side varies for each isochronous cycle, and packet interval on the transmit side can not be guaranteed.

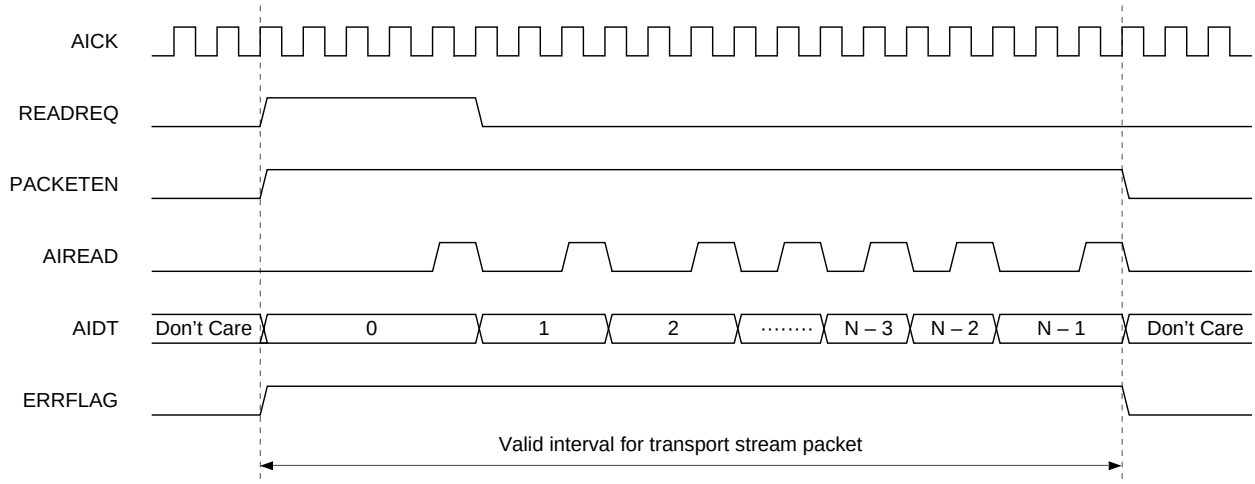
The timing is shown below.

The mode in which the time stamp is not used can be set by setting the CFR TmsDis register to 1. (The default is 0.)

Receive Interface



Receive Interface (for sync mode/8 bit/Nbyte)

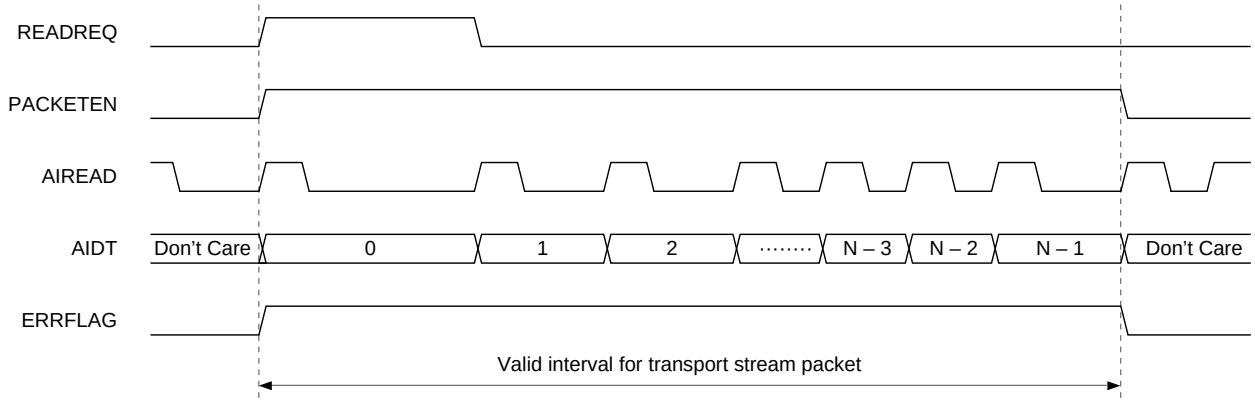


Sync interface mode is obtained by setting the CFR AsyncAI register to 0. (The default is 0.)
The CXD1948R outputs the first data of the transport stream packet by changing the READREQ output signal and PACKETEN output signal from low to high when the packet has been completely received and is ready for output.
The READREQ output signal goes low when the first data is output.
The same as for transmit, the result of subtracting 4 and the value set in the AddSize register from the value set in the CFR S_PacketSize register is used as the size of the valid data output in one packet. The timing is done by the AICK rise when AIREAD is high.
The AIREAD signal is used as the enable signal, so the interval that the AIREAD signal is high relative to one data must be one AICK clock interval. The interval that the AIREAD signal is low relative to one data is not specified.
The limits on AICK input frequency are given below.

For 8-bit data input: 40MHz (Max.), 2MHz (Min.)
For 16-bit data input: 20MHz (Max.), 2MHz (Min.)

The ERRFLAG input during receive can be made valid by setting the CFR ErrBitEnable register to 1. (The default is 0.)
The CXD1948R outputs the ERRFLAG at high during the valid interval if the output transport stream packet is an error.
The switching timing for ERR_FLAG input can be changed in the same way as data switching timing, with AIREAD signal rise.

Receive Interface (for asynchronous mode/8 bit/Nbyte)



Sync interface mode is obtained by setting the CFR AsyncAI register to 0. (The default is 0.)
The CXD1948R outputs the first data by changing the READREQ output signal and PACKETEN output signal from low to high when the packet has been completely received and is ready for output.
The READREQ output signal goes low when the first data is output.
The same as for transmit, the result of subtracting 4 and the value set in the AddSize register from the value set in the CFR S_PacketSize register is used as the size of the valid data output in one packet. The timing is done by the AIREAD signal rise.
The AIREAD signal is used as the clock, so there must be one AIREAD input rising edge relative to one data. AIREAD input duty is not specified, but the AIREAD input must continue to be input evenly as the clock even outside of the transport stream packet valid interval.
The interval that the AIREAD signal is low relative to one data is not specified.
The limits on AICK input frequency are given below.

For 8-bit data input: 20MHz (Max.), 1MHz (Min.)
For 16-bit data input: 10MHz (Max.), 1MHz (Min.)

The ERRFLAG input during receive can be made valid by setting the CFR ErrBitEnable register to 1. (The default is 0.)
The CXD1948R outputs the ERRFLAG at high during the valid interval if the output transport stream packet is an error.
The switching timing for ERR_FLAG input can be changed in the same way as data switching timing, with AIREAD signal rise.

5-2-6. Using the ERRFLAG Pin

The CXD1948R can function as a transport stream data interface using error information.
Concretely, this is done using the ERRFLAG pin.
Like other transport stream data interfaces, the ERRFLAG pin is a bidirectional pin. I/O switching is performed by the CFR IGFMode register.
Also, ERRFLAG control is performed by the CFR ErrBitEn and ErrOutEn registers.
The settings when using ERRFLAG during transmit and receive are given below.

Transmit settings

The ERRFLAG input can be made valid by setting the CFR ErrBitEn register to “1”. (The default is “0”.)
In order to add the error information provided by ERRFLAG to an isochronous packet and transmit this information, the added data must be 4 bytes or more.
Concretely, the CFR AddSize register value must be 4 or larger.
Both of the conditions below must be met to send error information using the ERRFLAG pin during transmit.
(1) The ErrBitEn register is 1.
(2) The AddSize register is 4 or higher.

Receive settings

The ERRFLAG output can be made valid by setting the CFR ErrOutEn register to “1”. (The default is “0”.)
The CXD1948R handles received packet errors in two ways as follows.

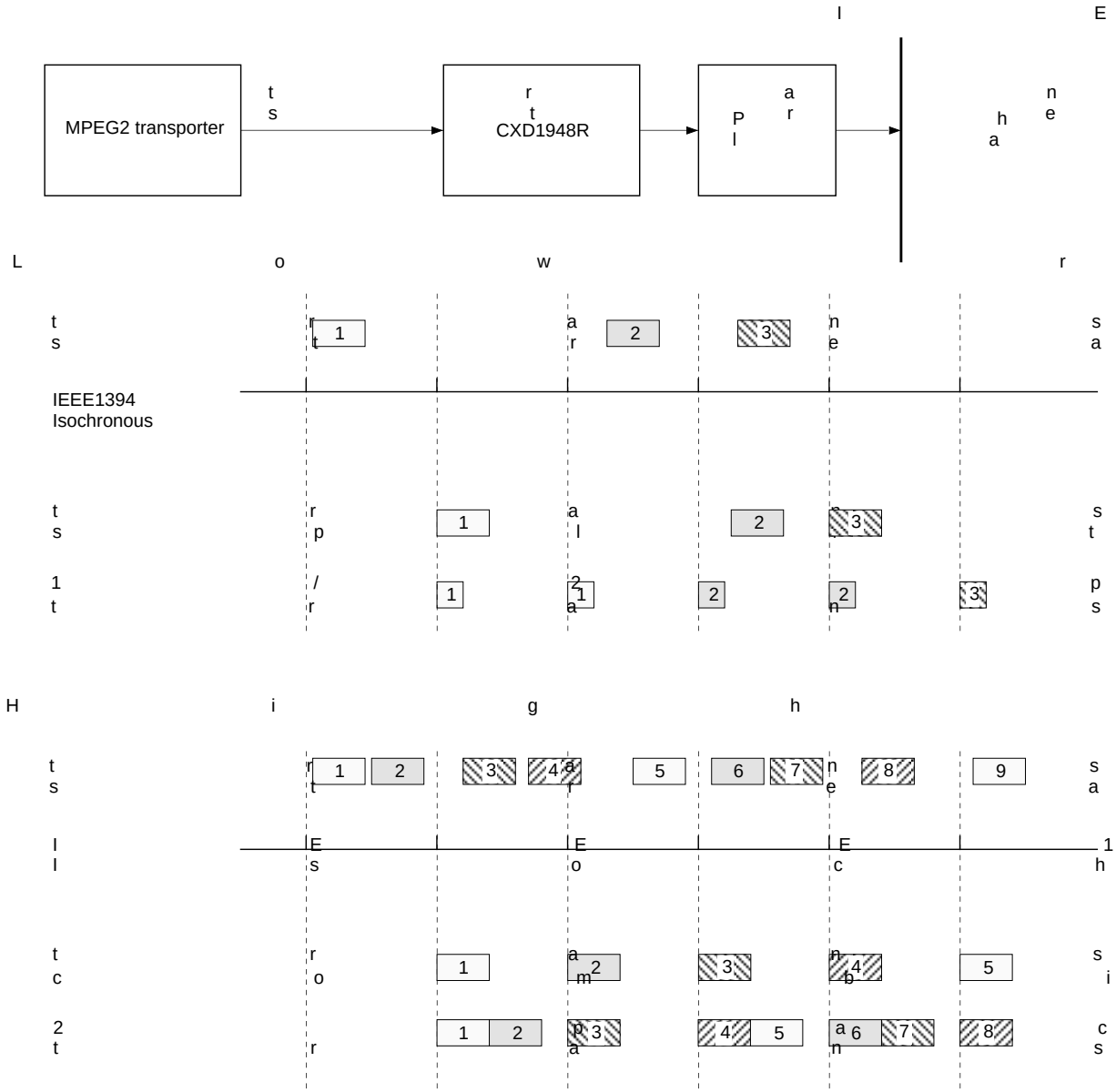
(1) Packets with error information added to the transport stream data
In this case, the error information must be added to the received isochronous packet.
In addition, the CFR ErrBitEn register must be set to 1.

(2) Error packets occurring during isochronous communication
These errors refer to packets received with non-consecutive DBC values due to CRC errors or Late processing during transmit.
In this case, only the CFR ErrOutEn register is set.
Setting this register to “1” inserts an error packet between packets with non-consecutive DBC values.
As a result, one packet of ERRFLAG = 1 data is read by the transport stream data interface.
If the ErrOutEn register is set to 0, this processing is not performed.
Non-consecutive DBC values also result when the receive FIFO overflows, and the processing is the same in these cases as well.

5-3. Transport Stream Packet Split and Combine Functions

The CXD1948R supports split and combine functions in order to use the bus bandwidth effectively.
Concretely, the input transport stream packet can be split or combined according to the input rate and transmitted as an isochronous packet.
The minimum size which can be split on the CXD1948R is one data block.
For DVB specifications, one data block consists of 24 bytes, which means that 1 packet can be divided into eighths for transmission.
The maximum size which can be combined is 15 packets. However, if the number of packets which can be transmitted in that cycle is smaller than the number that can be combined, all the packets present are combined in the transmission FIFO buffer and transmitted.
The number of data blocks to be transmitted in one isochronous cycle is set before hand according to the stream peak rate.
This is done by setting values in the CFR NOSP and NODB registers.
The number of transmit source packets in one isochronous cycle is expressed by NOSP, and the number of transmit data blocks in one isochronous cycle is expressed by NODB.
The default values are NOSP = 0001 and NODB = 000, for transmission of one source packet in one isochronous cycle. Always be sure to set either NOSP or NODB to 0.
If the isochronous transmission parameters are not set such that the transmission rate is greater than the input stream peak rate, the FIFO buffer may fail.

Example of Isochronous Transmit using Packet Split and Combine Functions



5-4. Transport Stream Data Bandwidth

The bandwidths on which transport stream data can be transmitted on the CXD1948R are illustrated below.

DVB

NOSP value	NODB value	No. of transmit data	Transmittable data rate
0000	001	1 data block	1.5Mbps
0000	010	2 data blocks	3.0Mbps
0000	100	4 data blocks	6.0Mbps
0001	000	1 source packet	12.0Mbps
0010	000	2 source packets	24.0Mbps
0011	000	3 source packets	36.0Mbps
0100	000	4 source packets	48.1Mbps
0101	000	5 source packets	60.1Mbps

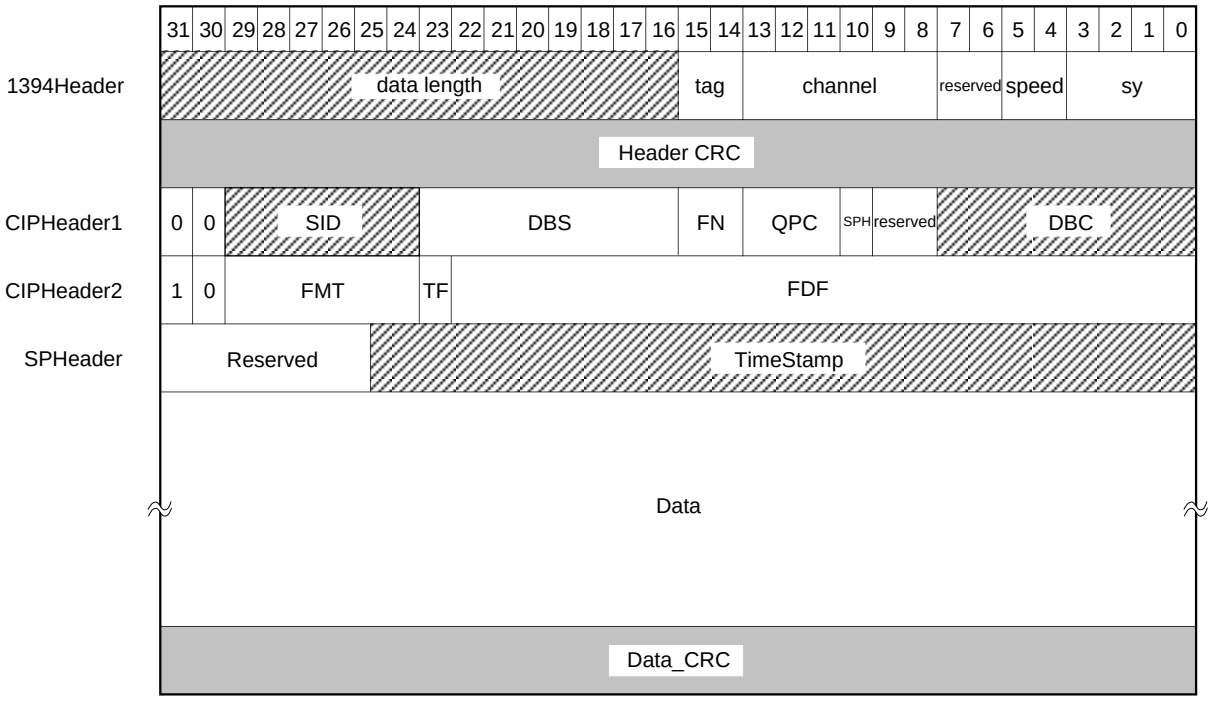
DSS

NOSP value	NODB value	No. of transmit data	Transmittable data rate	Transmittable data rate	
				Addsize = Ah	Addsize = 0h
0000	001	1 data block	2.2Mbps	2.0Mbps	2.2Mbps
0000	010	2 data blocks	4.4Mbps	4.1Mbps	4.4Mbps
0001	000	1 source packet	8.9Mbps	8.3Mbps	8.9Mbps
0010	000	2 source packets	12.0Mbps	16.6Mbps	17.9Mbps
0011	000	3 source packets	26.8Mbps	24.9Mbps	26.8Mbps
0100	000	4 source packets	35.8Mbps	33.2Mbps	35.8Mbps
0101	000	5 source packets	44.8Mbps	41.6Mbps	44.8Mbps
0110	000	6 source packets	53.7Mbps	49.9Mbps	53.7Mbps

5-5. Isochronous Packet Structure

The basic isochronous packet structure supported by the CXD1948R is illustrated below.
On the CXD1948R, CIPHeaders 1 and 2 are automatically attached/detected in conformity to AV protocol.
Also, a source packet header is automatically added/detected on the transport stream packet based on the MPEG Data Transmission for IEEE1394 Digital Interface specifications proposed in DVB-SMI.
When one source packet is transmitted in one isochronous cycle, 1st quadlet is 1394Header, 2nd quadlet is Header_CRC (added at Link Core), 3rd quadlet is CIPHeader1, 4th quadlet is CIPHeader2, 5th quadlet is source packet header and 6th quadlet and after is the data area. The final quadlet is Data_CRC (added at Link Core).
When a dummy packet is transmitted, there are only 1394Header, Header_CRC, CIPHeader1, CIPHeader2 and Data_CRC.

Basic Structure



Note: The diagonally shaded areas (▨) for 1394Header and CIPHeader1 and 2 are attached automatically by the CXD1948R. Other areas are set from the external microcomputer via the host I/F.
In transmission from the CXD1948R to an IEEE 1394PHYIC, the four bits marked "reserved" and "speed" of the 1394Header are automatically replaced by "tcode".

1394Header Fields

Field Name	Description
data_length	Indicates the byte length of data from CIPHeader1.
tag	Type of format for data transferred as isochronous packet. 00: no Header 01: format defined by AV protocol 10: Reserved 11: Reserved
channel	Channel number used by the transmitted packet.
speed	Defines transfer speed. 00: 100Mbit/s 01: 200Mbit/s 10: 400Mbit/s 11: Undefined The CXD1948R supports “00” and “01”.
sy	This field is defined by the application.

Note: The Link Core must be informed of the communication speed in the case of isochronous communication, so [5 : 4] of [7 : 4], where a tCode is inserted, is used as the speed code. This is replaced with tCode (1010) at the Link Core.

CIPHeader1 Fields

Field Name	Description
SID	The CXD1948R Node ID. (0 to 3Fh)
DBS	The number of quadlets transferred in one isochronous packet. For DVB specifications, this is “00000110b” and for DSS specifications, “00001001b”.
FN	Indicates how many data blocks a source packet is divided into.
QPC	The number of quadlets added to an incomplete packet when the source packet is split. Fixed at “0h” on the CXD1948R.
SPH	Expresses whether a source packet header is used or not. Fixed at “1” on the CXD1948R.
DBC	Increased by one for each data block. This is a free run continuity counter. When multiple data blocks are to be sent as a single isochronous packet, indicates the value for the first data block.

CIPHeader2 Fields

Field Name	Description
FMT	Data format ID. This is “100000b” for DVB and “100001b” for DSS specifications.
TF	Indicates whether data is time-shifted. 1 if time-shifted; 0 if not time-shifted.
FDF	Used in application defined by FMT. The value set via the host I/F is input as is on the CXD1948R.

SPHeader fields

Field Name	Description
Reserved	The value set via the host I/F is input as is on the CXD1948R.
TimeStamp	This is the value of the time that the transport stream packet arrived at the CXD1948R plus the fixed delay value.

5-5-1. DVB Format

The DVB format supported by the CXD1948R is described below.

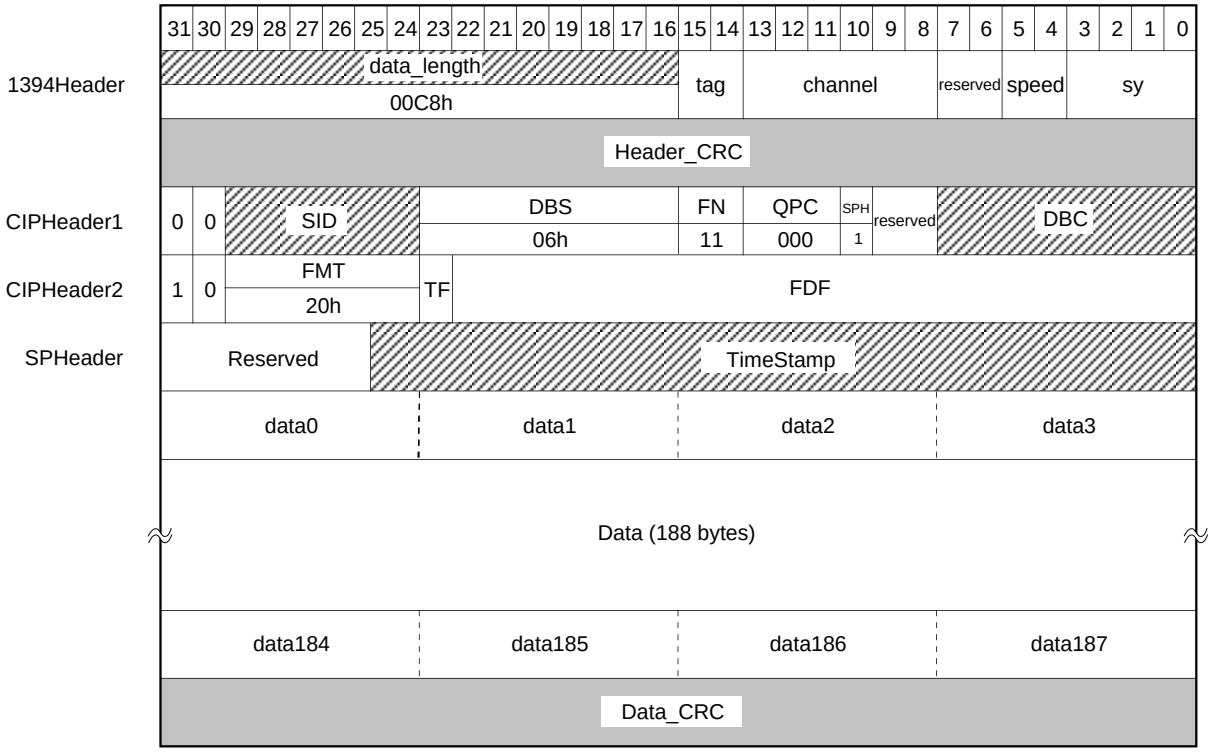
The CFR S_PacketSize and AddSize registers are set as shown below when using the CXD1948R in DVB format.

S_PacketSize: 0C0h
AddSize: 0h

Next, the NOSP and NODB registers are set. These determine how many data are to be transmitted/received in one isochronous cycle.

The structures of isochronous packets for transmit/receive of data consisting of 1 source packet, 4 data blocks or 2 source packets in one isochronous cycle are illustrated below.

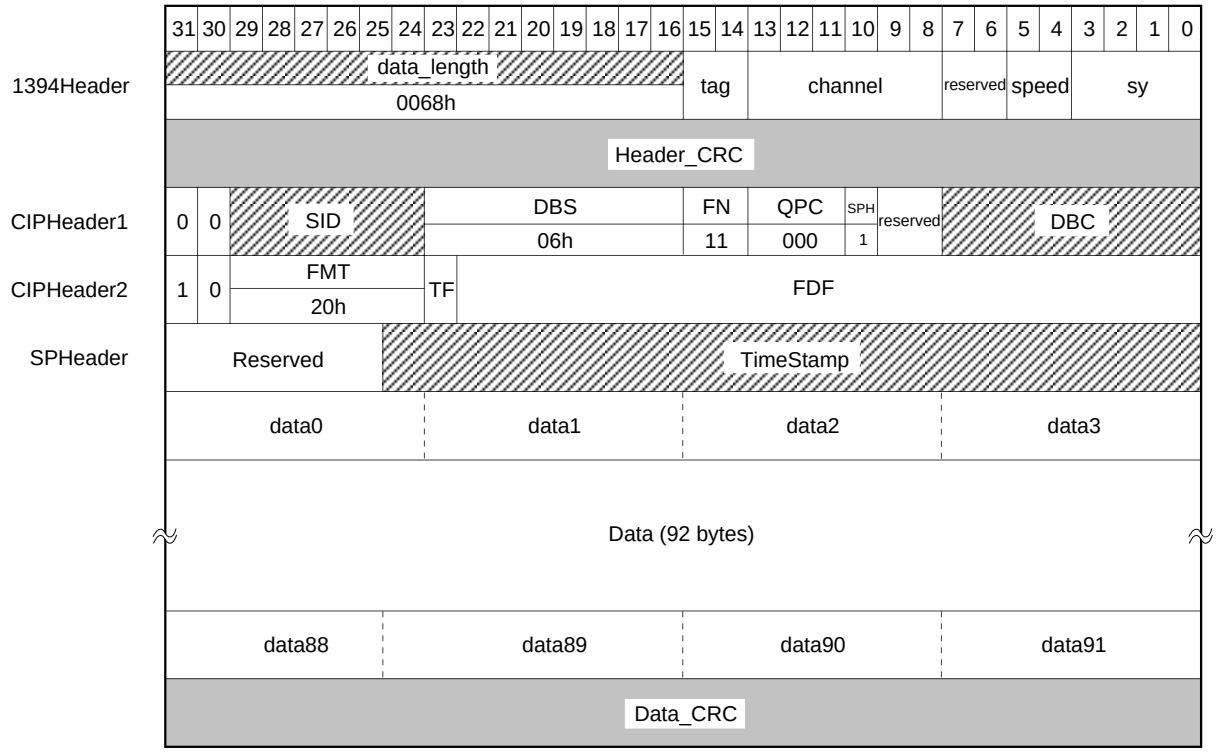
DVB/Isochronous Packet Structure (1 source packet)



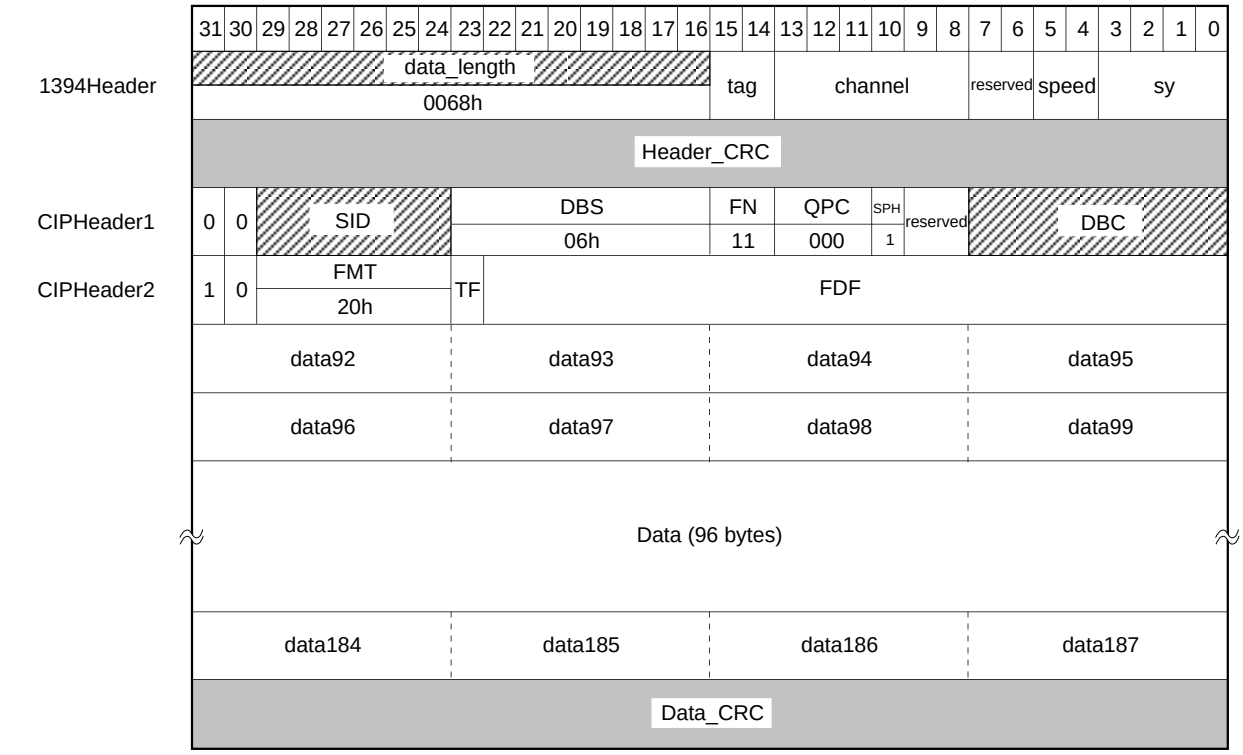
Note: The diagonally shaded areas (▨) for 1394Header and CIPHeader1 and 2 are attached automatically by the CXD1948R. Other areas are set from the external microcomputer via the host I/F.
In transmission from the CXD1948R to an IEEE 1394PHYIC, the four bits marked "reserved" and "speed" of the 1394Header are automatically replaced by "tcode".

DVB/Isochronous Packet Structure (4 data blocks)

(First half data)

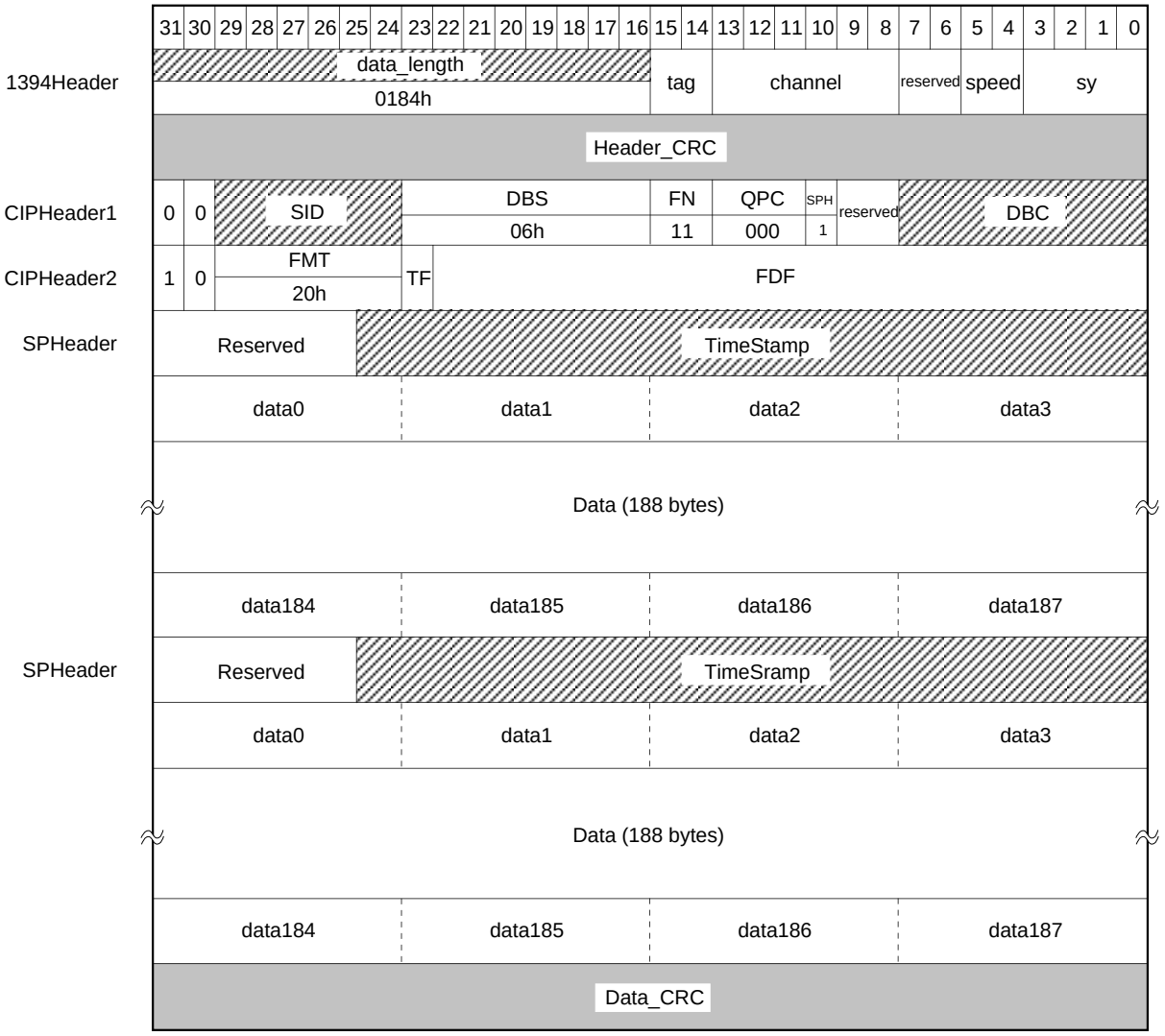


(Second half data)



Note: The diagonally shaded areas (▨) for 1394Header and CIPHeader1 and 2 are attached automatically by the CXD1948R. Other areas are set from the external microcomputer via the host I/F.
In transmission from the CXD1948R to an IEEE 1394PHYIC, the four bits marked "reserved" and "speed" of the 1394Header are automatically replaced by "tcode".

DVB/Isochronous Packet Structure (2 source packets)



Note: The diagonally shaded areas (▨) for 1394Header and CIPHeader1 and 2 are attached automatically by the CXD1948R. Other areas are set from the external microcomputer via the host I/F.
In transmission from the CXD1948R to an IEEE 1394PHYIC, the four bits marked "reserved" and "speed" of the 1394Header are automatically replaced by "tcode".

5-5-2. DSS Format

The DSS format supported by the CXD1948R is described below.

The interface with the system and the CFR setting differ according to whether the CXD1948R is used as DSS-STB or DSS-DVHS system digital I/F, as follows:

A) DSS-STB system digital I/F: S_PacketSize: 090h; AddSize: 0h

B) DSS-DVHS system digital I/F: S_PacketSize: 090h; AddSize: Ah

The isochronous data format on the IEEE1394 bus does not change for A and B. Only the amount of data interfaced with the system changes.

The DSS-STB system interfaces with 130 bytes as one packet of transport stream.

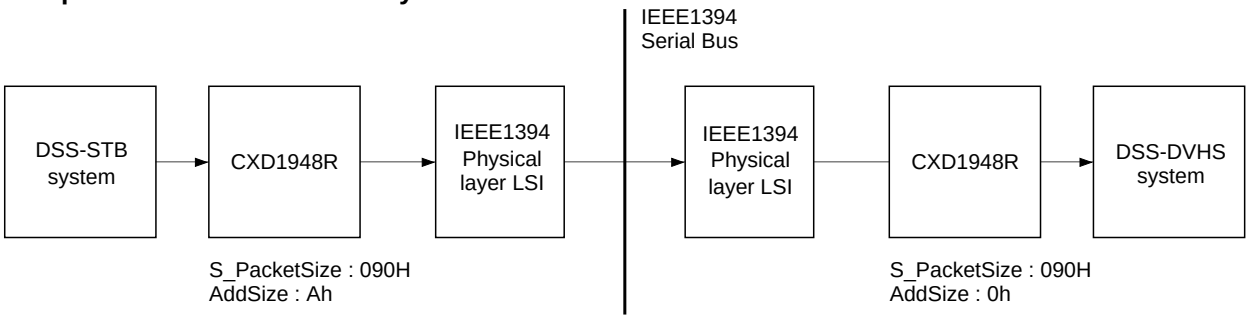
The CXD1948R adds a 4-byte source packet header and 10 bytes of data set on the CFR via the host I/F to the received 130 bytes, then transmits them as an isochronous packet.

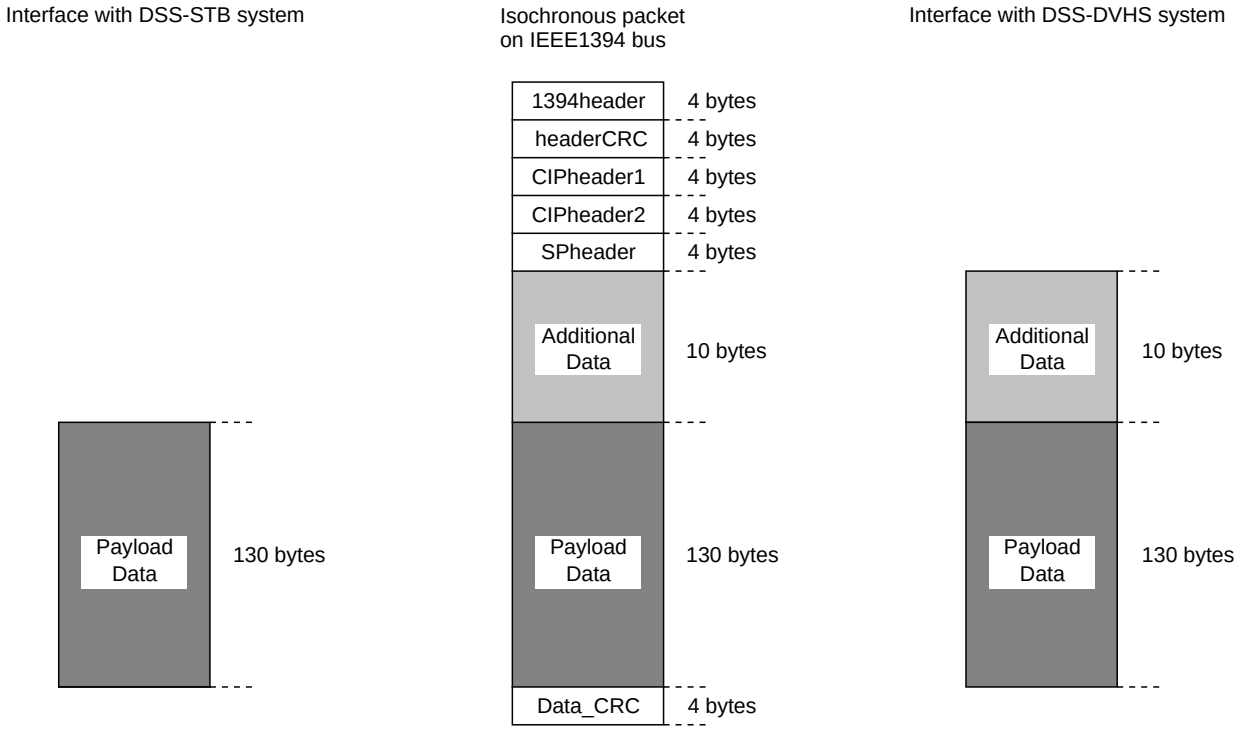
For the DSS-DVHS system, only the source packet header is removed from the received isochronous packet, and 140 bytes of data are output as transport stream data.

The above is for the case of transmission; in reception, 140 bytes of data is input from the DSS-DVHS system, and 130 bytes of data is output to the DSS-STB system.

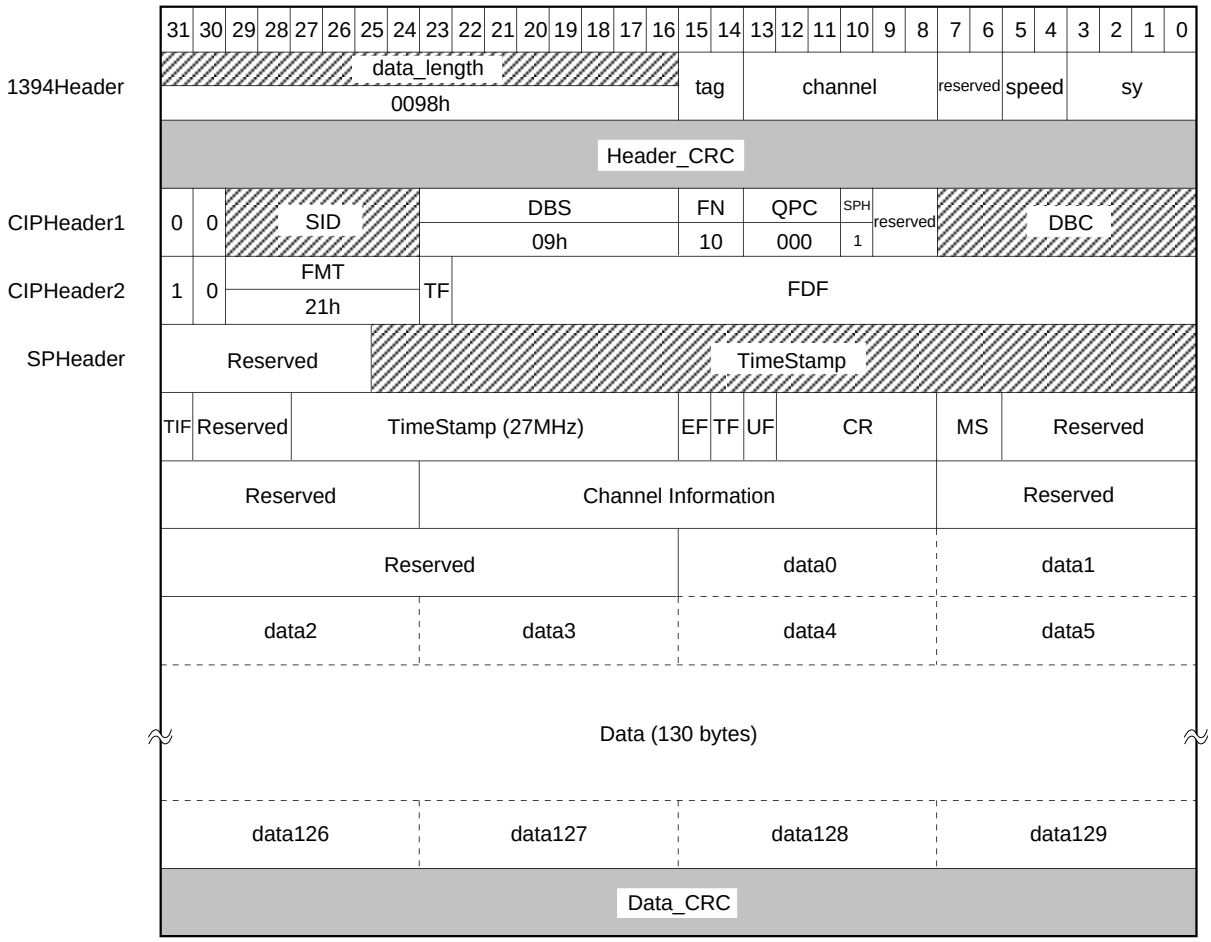
The same as for the DVB format, the amount of data transmitted in one isochronous cycle is set by the NOSP and NODB registers.

Example of Interface with DSS System





DSS/Isochronous Packet Structure (1 source packet)



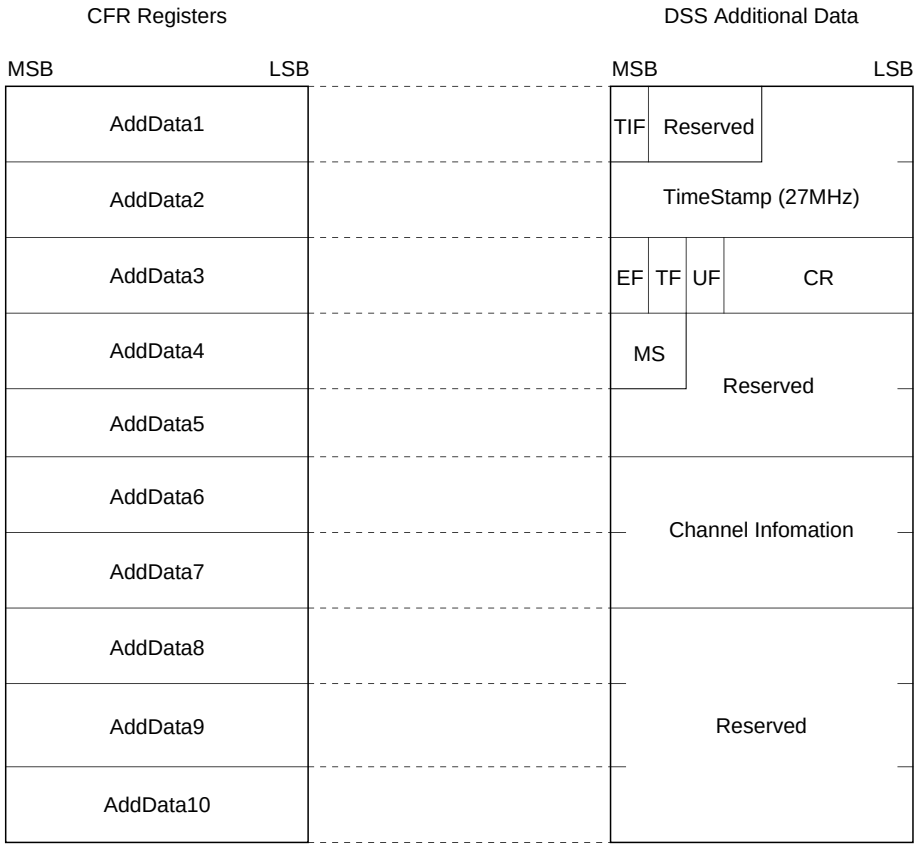
Note: The diagonally shaded areas (▨) for 1394Header and CIPHeader1 and 2 are attached automatically by the CXD1948R. Other areas are set from the external microcomputer via the host I/F.
The white areas (□) indicate additional data.
In transmission from the CXD1948R to an IEEE 1394PHYIC, the four bits marked "reserved" and "speed" of the 1394Header are automatically replaced by "tcode".

Additional Data Fields

Field Name	Description
TIF	Time Stamp Invalid Flag 1: Invalid, 0: valid
TimeStamp	Time stamp value stamped at 27MHz synchronized to MPEG2 transport stream.
EF	Error Flag 1: Error, 0: No Error
TF	Transition Flag 1: Transition (1sec), 0: No Transition
UF	VBV Underflow Flag 1: Underflow, 0: Normal
CR	Copy Right
MS	Multi/Single 00: Single program, 01: 10: 11: Reserved
Channel Information	
Reserved for bit rate	

5-6. Relationship between Additional Data and CFR Registers

The CXD1948R has 10 bytes of data registers for additional data. These are common registers for transmit and receive. Setting is done from the host I/F to add data to the transport stream data.
For receive, the additional data is detected from the isochronous packet and the value is written in.
The relationship between the CFR additional data registers AddData and the additional data in DSS format is described below.



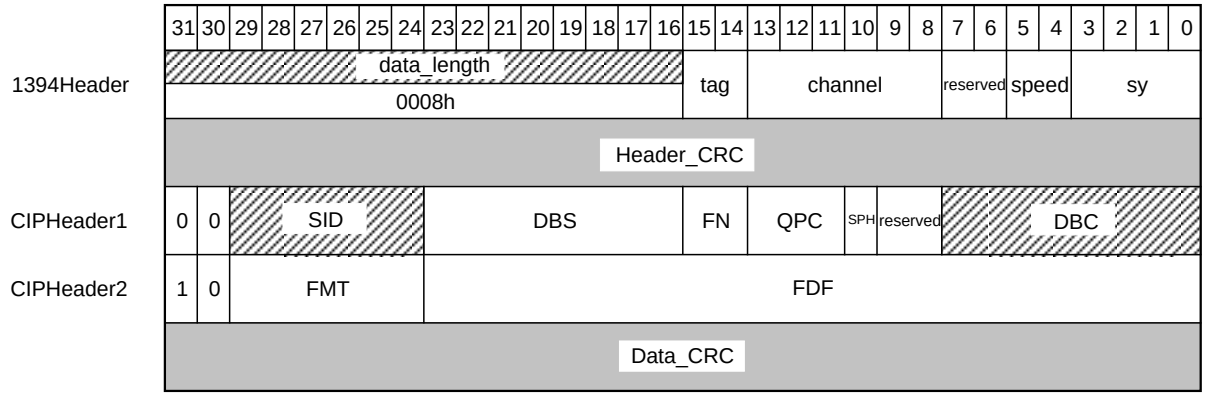
5-7. 27MHz Time Stamp

When a 27MHz time stamp synchronized to the transport stream is added on the CXD1948R, the CFR 27MTS register is set to “1”. Further, a 27MHz clock must be input from the CK27 pin. When the 27MTS register is set to “0”, the 27MHz time stamp field value enters as is to the lowest 4 bits of the AddData1 register and the value of the AddData2 register.

5-8. Dummy Packet Transmission

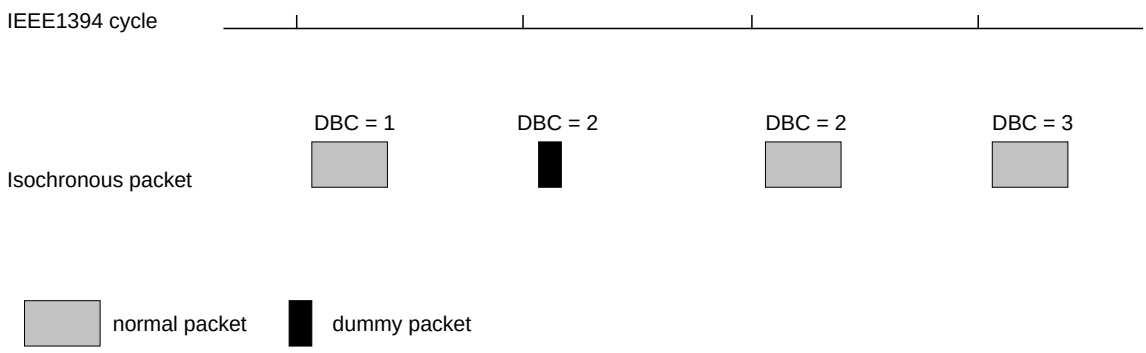
The CXD1948R has a function that automatically transmits a dummy packet when isochronous transmission is muted, or when there is no packet to be transmitted during isochronous transmission.
The dummy packet consists of header information only. The DBC value is the normal packet value transmitted on the next isochronous cycle.

Dummy Packet Structure



Note: The diagonally shaded areas ( ) for 1394Header and CIPHeader1 and 2 are attached automatically by the CXD1948R. Other areas are set from the external microcomputer via the host I/F.
In transmission from the CXD1948R to an IEEE 1394PHYIC, the four bits marked "reserved" and "speed" of the 1394Header are automatically replaced by "tcode".

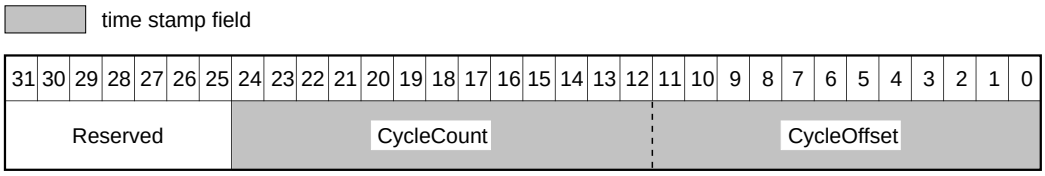
Example of Dummy Packet Transmission (for transmit of 1 data block)



5-9. Time Stamp

The CXD1948R has a function that adds a time stamp to the input transport stream packet and transmits it, detects the time stamp on a received packet, and outputs the transport stream packet based on that value. The time stamp is 25 bits, and is superimposed on the source packet header.

Source Packet Header Structure



Time Stamp Field

Field Name	Description
CycleCount	Time intervals of less than 1 second are expressed in units of 125µs. (0 to 1F3Fh)
CycleOffset	Time intervals of less than 125µs are expressed in units of 24.576MHz clock cycles. (0 to BFF)

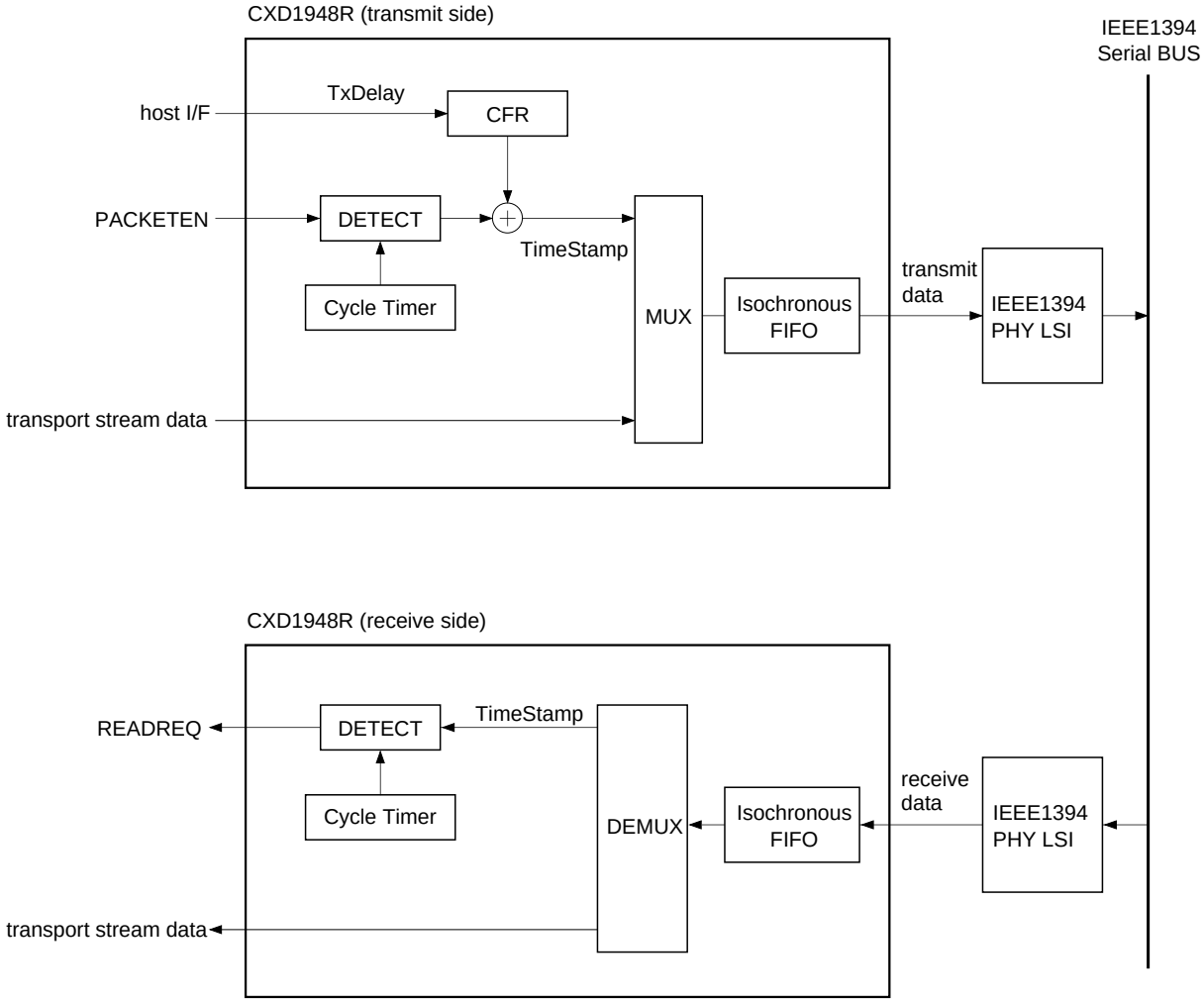
Time Stamp Adding

The CXD1948R takes in the internal cycle timer value when the transport stream packet is completely received. Next it adds the CFR TxDelay register value set from the host I/F to the cycle timer value, and superimposes it on the source packet header as a time stamp.

Time Stamp Detection

After completely receiving the isochronous packet, the CXD1948R detects the superimposed time stamp to the source packet header. It compares the detected time stamp value and the internal cycle timer value, and starts data output from the point where the times match. (Actually output request is performed.)
Also, if the time is already past, it performs data output immediately.

Block Diagram



5-10. Error Processing

When a correct packet can not be received, for example if a CRC error is generated or a packet which violates protocol is received, the CXD1948R automatically performs error processing.
Error processing control is done by setting the CFR ErrOutEn register.
A description of error processing follows.

Processing for Transmit

ErrFlag input becomes valid when the CFR ErrBit Enable register is set to “1”.
The input error packet is processed in the same way as a normal packet.
However, error information is also transmitted, so only the mode when additional data is added to the transport stream data is valid. (The superimposed position of the error information is the MSB of the additional data 3rd byte.)
ErrFlag input becomes invalid when the ErrBitEnable register is set to “0”.

Processing for Receive

ErrFlag output becomes valid when the CFR ErrBit Enable register is set to “1”.
When the received isochronous packet is judged as an error, ErrFlag is made “1” and output.
Because the time stamp cannot be trusted, output is performed immediately.
When the ErrFlagEnable register is set to “0”, only the correctly received isochronous packets are output.

5-11. Late Processing

The CXD1948R automatically performs late processing when an isochronous packet is transmitted.
In late processing, the time stamp superimposed on the source packet header of the packet for transmission is consulted, and if there is the possibility that the receiving side may not receive the entire packet within the timing indicated by the time stamp, the packet transmission is canceled.

The time stamp value is the time when the transport stream packet was received plus TxDelay, so the number of packets for late processing can be reduced by making the TxDelay value larger.
However, the upper limit of the TxDelay value depends on the stream transmission rate and the built-in FIFO buffer capacity.

6. Asynchronous Communication

6-1. Host/IF

The host I/F controls data communication between the external CPU and the CXD1948R ATF/ARF/CFR/IPB*1, respectively.
Communications between the CPU and CXD1948R include:

1) CPU writes data to ATF → asynchronous packet transmit
2) CPU reads data in ARF → asynchronous packet receive
3) CPU writes data to CFR → mode, header data setting
4) CPU reads data in CFR → internal status, header data read-in
5) CXD1948R informs CPU of an interrupt event with an interrupt signal
6) CPU performs insert of isochronous packet from host I/F
7) CPU sets additional information (10 bytes) in isochronous packet

The CXD1948R supports 16-bit and 8-bit host I/F.
The ATF/ARF/CFR built in to the CXD1948R have a 32-bit structure, so all bits can not be accessed with one access. The target address must be accessed two consecutive times for 16 bits and four consecutive times for 8 bits.

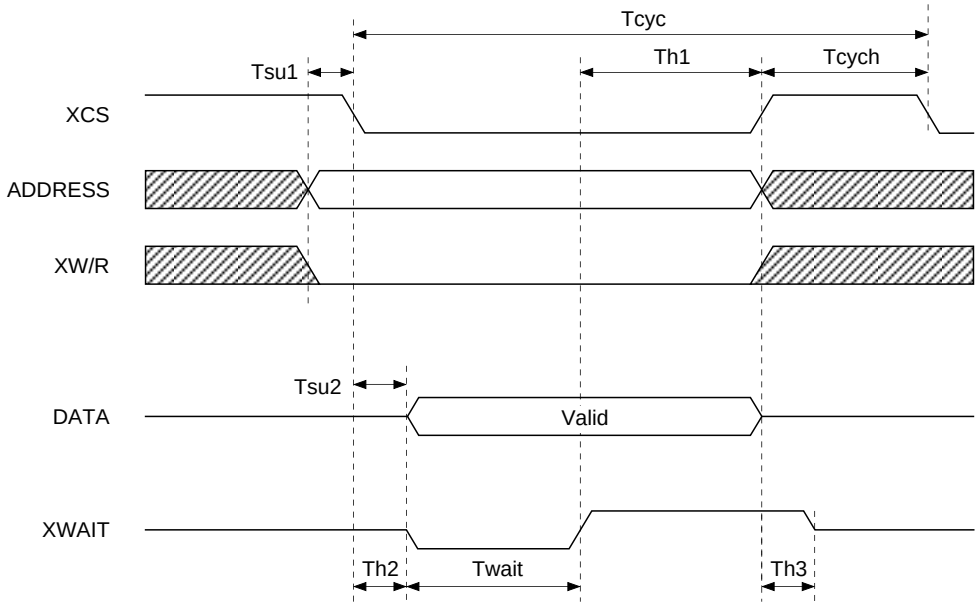
The roles played by the signals communicated between the CXD1948R and the external microcomputer are given bellow.

DATA [15:0]	in/out	Data for writing to or reading from specified address
ADDRESS [7:0]	in	Address for writing or reading data Data destination (CFR or FIFO) and data breakpoint (Write or Confirm) are discriminated according to the address
XCS	in	Access enable from host bus (low active)
XW/R	in	Data read/write enable signal (high: read; low: write)
XWAIT	out	Indicates access (read or write) completed to specified address (low active)
XINT	out	Interrupt signal. Indicates some kind of interrupt when low Type of interrupt and mask specified by CFR
X8/16	in	Host I/F data bus switching H: 16 bits; L: 8 bits

*1 ATF (Asynchronous Transmit Fifo), ARF (Asynchronous Receive Fifo),
CFR (Configuration Register), IPB (Insert Packet transmit Buffer)

Writing Timing to ATF/CFR/IPF

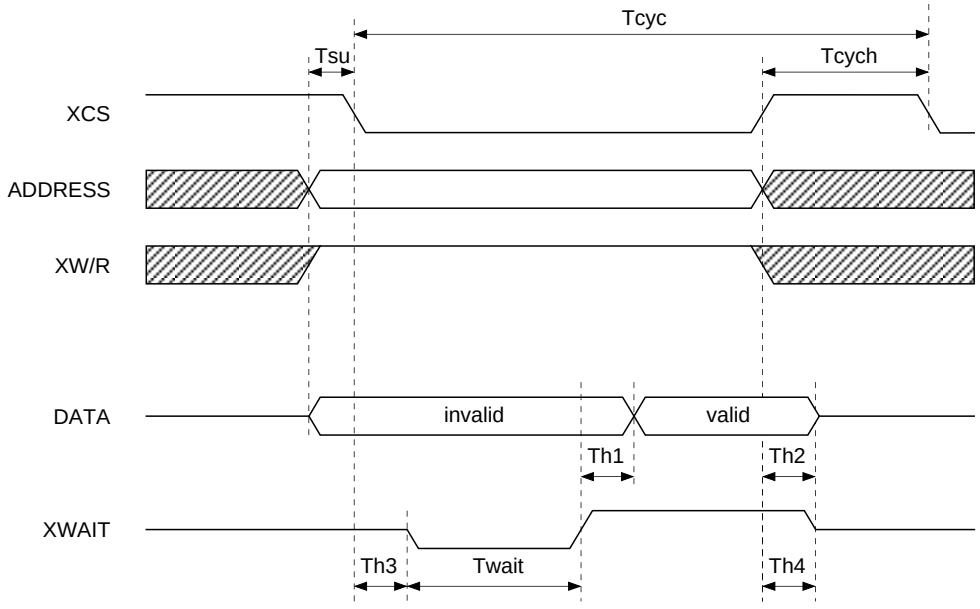
For write, the CXD1948R latches ADDRESS, XW/R and DATA at XCS falling edge. The timing chart is illustrated below.



Tsu1 ≥ 0ns, Tsu2 ≤ 5ns, Th1 ≥ 0ns
Tcyc ≥ 250ns, Tcych ≥ 100ns, Twait ≤ 100ns, Th2 ≤ 5ns, Th3 ≤ 5ns

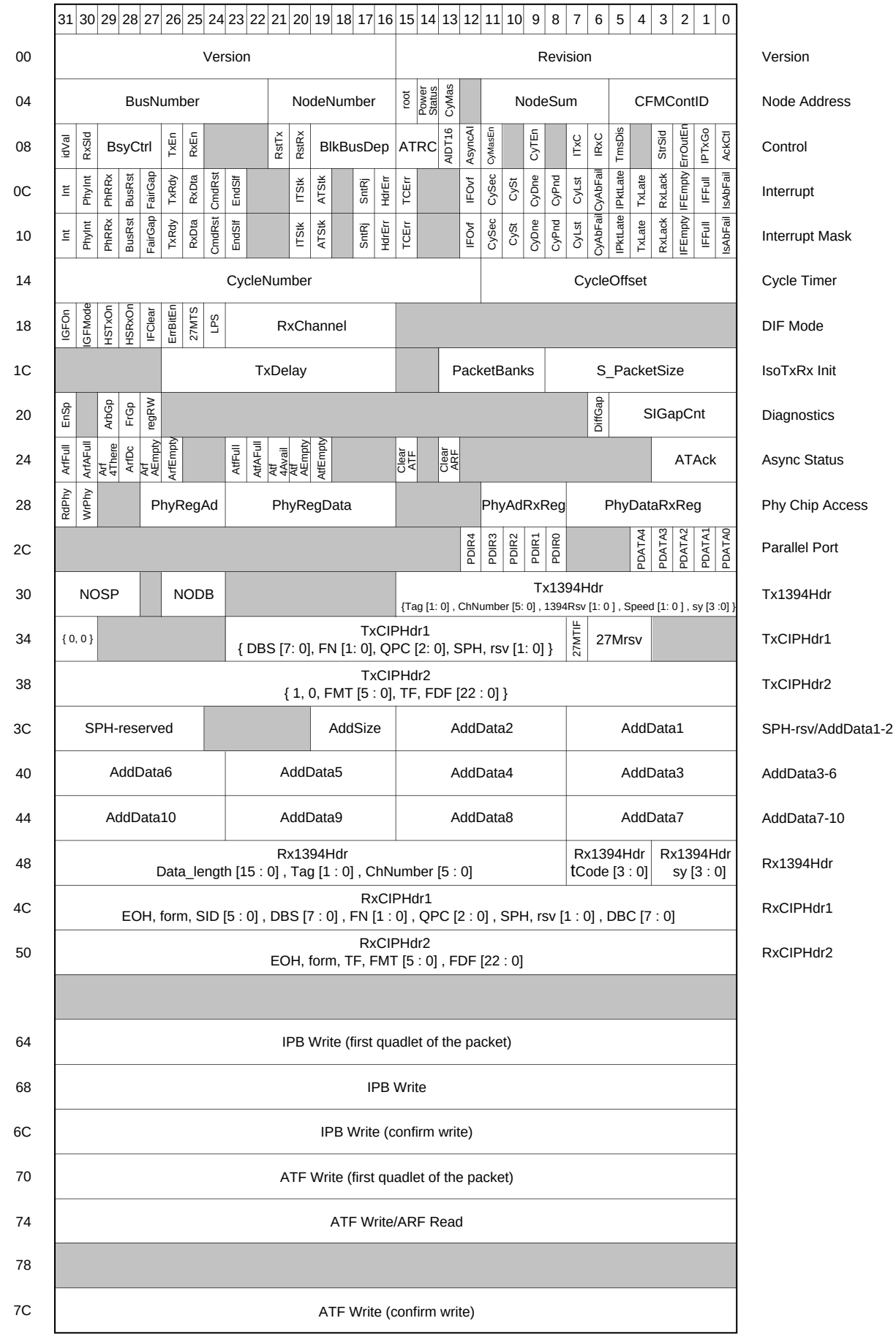
Timing for Data Read from ARF/CFR

For read, the CXD1948R latches ADDRESS and XW/R at XCS falling edge. The timing chart is illustrated below.



Tsu ≥ 0ns, Th1 ≥ 5ns
Tcyc ≥ 400ns, Tcych ≥ 100ns, Twait ≤ 250ns, Th2 ≤ 5ns, Th3 ≤ 5ns, Th4 ≤ 5ns

Configuration Register (CFR) Address Map



Note: The shaded areas ( ) are reserved and can not be used.

6-2. CFR (Configuration Register)

This is a memory space to store the status information and operation mode and packet header information inside the chip. Read/write with the external microcomputer can be performed via the host I/F. (Be sure to set to “0” when writing to the reserved area.)
The address map and register contents are shown below.

Register Description

1) Version/Revision Register

These registers have the CXD1948R version/revision written in them.
The register address is 00h; they are read only, and the initial value is 0000_0001h.

Bit	Name	Function
31 to 16	Version	CXD1948R Version Number
15 to 0	Revision	CXD1948R Revision Number

2) Node Address Register

These registers are used for packet receive/refuse control, and to monitor root/cycle master status and the total number of nodes connected.
The register address is 04h and the initial value is FFFF_0040h.
Only the bus number is read/write, and the other registers are normally read only, but the diagnostic register can be read/write by setting regRW to “1”.

Bit	Name	Function
31 to 22	Bus Number	Bus number of connected bus
21 to 16	Node Number	Node number of this link
15	root	Root/not root for this link 1: root; 0: not root
14	Power Status	Cable power status for this mode
12	CyMst	Whether or not this link is cycle master 1: cycle master; 0: not
11 to 6	NodeSum	Total number of connected nodes. Value is 0 when an error occurs in Self_ID communication.
5 to 0	CFMcontID	The Phy_ID of the contender is inserted here. However, when the node can itself be a contender and it has a larger Phy_ID than this value, then the node itself is the contender.

3) Control Register

These registers perform settings for the CXD1948R basic operations.

The register address is 08h; they are read/write, and the initial value is C400_0A00h.

Bit	Name	Function
31	idVal	Receives packet from the address set in the node address register at “1”. Receives packet at bus number “3Fh” node number “3fh” only at “0”.
30	RxSld	Validates reception of SelfID packet at “1”. Non-valid at “0”. Processing for NodeSum, CFMContID, DiffGap and SIGapCnt is invalid.
29 to 27	BsyCtrl	Controls Busy status of input packet. 000 = Returns busy according to normal busy/retry protocol when necessary. (Always fixed at “000” on the CXD1948R)
26	TxEn	Transmitter does not transmit Arbitration and packet when “0”
25	RxEn	Receiver does not receive packet when “0”
21	RstTx	Sync resets transmitter when set to “1” This bit is cleared automatically
20	RstRx	Sync resets receiver when set to “1” This bit is cleared automatically
19 to 16	BlkBusDep	Access limits on Broadcast packets for the CSR bus-dependent area can be set here. Each bit corresponds to 128 bytes of the CSR bus-dependent area. With the LSB corresponding to the first 128 bytes, and so on until the MSB corresponding to the last 128 bytes, access to any area for which the corresponding bit has been set “1” is invalid.
15, 14	ATRC	The Retry Code can be controlled. The Retry Code for the packet being sent is the logical OR of the value set here and the value stored in ATF.
13	AIDT16	Selects AIDT bus width. 8 bits at “0” and 16 bits at “1”
12	AsyncAI	Transport stream I/F setting. Synchronous at “0” and asynchronous at “1”.
11	CyMasEn	When “1” and the node becomes the ROOT, Cycle Master functions are activated.
9	CyTEn	Validates Cycle Offset increment at “1”
7	ITxC	Transmits isochronous packet with tCode = C at “1” Transmits isochronous packet with tCode = A at “0”
6	IRxC	Receives tCode = A, tCode = C packets as isochronous packet at “1” Receives tCode = A packet as isochronous packet at “0”
0	AckCtl	Controls Ack code which is returned when tCode = 0, 1 (Write request quadlet/block) packets are received. 0: Ack code = 1 (complate), 1: Ack code = 0 (pending)
5	TmsDis	Does not use time stamp for data read during receive at “1” Uses time stamp for data read during receive at “0”
3	StrSid	Takes received SelfID packet in to ARF at “1” Does not take received SelfID packet in to ARF at “0”
2	ErrOutEn	Outputs error packet at ErrFlag High during receive at “1” Does not output error packet during receive at “0”
1	IPTxGo	Transmits insert packet at “1” Automatically changes to “0” after transmit

4) Interrupt and Interrupt-Mask Registers

These registers combine the Interrupt Register, which informs the host I/F of changes in the CXD1948R status, and the Interrupt Mask Register, which masks the Interrupt Register.

The address of the Interrupt Register is 0Ch, and when the regRW bit is “0”, bits other than Int bit are cleared by writing. When the regRW bit is “1” all bits are read/write.

The address of the Interrupt-Mask Register is 10H and it is read/write.

The initial value for both registers is 0000_0000h.

Bit	Name	Function
31	Int	Indicates the OR result of all the interrupt (interrupt mask) bits.
30	PhyInt	Indicates Phy Interrupt received from Phy chip
29	PhRRx	Indicates data received from Phy to Phy register
28	BusRst	Indicates Bus Reset received from Phy
27	FairGap	Indicates FairGap received from Phy
26	TxRdy	Transmitter is able to transmit
25	RxDta	Receiver has received a correct packet
24	CmdRst	Receiver has received a packet addressed to CSR RESET_START register
23	EndSlf	SelfID phase complete
20	ITStk	Transmitter detected wrong data in isochronous FIFO during isochronous transmit
19	ATStk	Transmitter detected wrong data in asynchronous FIFO during asynchronous transmit
17	SntRj	Receiver transmitted Busy Ack for a packet transmitted to this node because receive FIFO is full
16	HdrErr	Receiver detected Header CRC error in the packet transmitted to this node
15	TCErr	Transmitter detected wrong tCode data in transmit FIFO
12	IFOvf	Indicates overflow in isochronous FIFO
11	CySec	Indicates that Cycle Timer register Cycle Number upper 7 bits were incremented. (This is generated almost every second when Cycle Timer is valid.)
10	CySt	Transmitter/Receiver transmitted/received Cycle Start packet
9	CyDne	After transmit or receive of Cycle Start packet, FairGap was detected on the bus. This means that the isochronous cycle is complete.
8	CyPnd	Indicates that Cycle Timer register Cycle Offset is “0”. Stays as is until isochronous cycle is complete.
7	CyLst	When not Cycle Master, indicates that Cycle Timer completed 2 cycles without receiving Cycle Start packet.
6	CyAbFail	Failure of Cycle Start packet transmission Arbitration
5	IPktLate	Late transmit of insert packet
4	TxLate	Late transmit of isochronous packet
3	RxLack	Indicates isochronous packet non-continuous DBC
2	IFEmpty	Isochronous FIFO empty
1	IFFull	Isochronous FIFO full
0	IsAbFail	Failure of isochronous packet transmission Arbitration

5) Cycle Timer Registers

These registers are composed of the 24.576MHz clock cycle Cycle Offset and the 125µs in its host, and the Cycle Master that counts 1 second. The value of all nodes are regulated by the Cycle Master node.
The register address is 14h; it is read/write, and the initial value is 0000_0000h.

Bit	Name	Function
31 to 12	CycleNumber	The upper 7 bits counts seconds (1Hz) and the lower 13 bits count the isochronous cycle (8kHz = 125µs). The values are controlled by Control register Cycle Master and Cycle Timer Enable.
11 to 0	CycleOffset	Counts the system clock (24.576MHz). The Cycle Number is incremented when this counter completes one cycle. The value is controlled by Control register Cycle Master and Cycle Timer Enable.

6) DIF Mode Register

This register performs settings for transmit and receive of isochronous packets, and sets transmitted isochronous packet 1394 header information (specified in the Draft).
The register address is 18h; it is read/write, and the initial value is 0100_0000h.

Bit	Name	Function
31	IGFOn	Isochronous FIFO operation is active at “1”
30	IGFMode	Changes isochronous FIFO operation to transmit at “1”. Receives at “0”
29	HSTxOn	Isochronous transmit side operation is active at “1”. Invalid at “0”
28	HSRxOn	Isochronous receive side operation is active at “1”. Invalid at “0”
27	IFClear	Clears isochronous FIFO at “1”
26	ErrBitEn	Uses 27MHz time stamp at “1”
25	27MTS	Does not use 27MHz time stamp at “0”
24	LPS	LPS pin high at “1” LPS pin low at “0”
23 to 16	RxChannel	Tag/Channel Number for isochronous packet receive

7) IsoTxRx Init

This register performs setting related to isochronous packet transmit/receive time stamp, and packet size and packet bank.
The register address is 1Ch; it is read/write, and the initial value is 0000_0000h.

Bit	Name	Function
26 to 16	TxDelay	Transmission delay time for isochronous packets. The upper 6 bits are added to the lower 6 bits of the Cycle Number, and the lower 5 bits are added to the upper 5 bits of the Cycle Offset to obtain the time stamp.
13 to 9	PacketBanks	Sets number of isochronous FIFO packet bank
8 to 0	S_PacketSize	Sets isochronous packet source packet size The value includes SPH and additional data (byte units)

8) Diagnostic Register

This register controls or monitors the CXD1948R status.

The register address is 20h and the initial value is 0000_0000h.

Only the EnSp bit and regRW bit are read/write; other bits are read/write when the regRW bit is “1” and read only when it is “0”.

Bit	Name	Function
31	EnSp	Receives all packets on the bus regardless of receiver address and format at “1”. Invalid at “0”.
29	ArbGp	Bus is in idle state due to Arbitration Reset Gap
28	FrGp	Bus is in idle state due to Fair Gap
27	regRW	Almost all registers are read/write when set at “1”
6	DiffGap	“1” when there is dispersion in Gap count values in received Self ID
5 to 0	SlGapCnt	This value is entered when all Gap count values in received Self ID are the same. “00h” when bus reset is generated.

9) Asynchronous Transmit and Receive FIFO Status Registers

These registers can monitor and control the ATF/ARF statuses.

The register address is 24h and the initial value is 0428_0000h.

Only the ClearATF bit and ClearARF bit are read/write; other bits are read/write when the regRW bit is “1” and read only when it is “0”.

Bit	Name	Function
31	ARFFull	ARF is full when “1” and receive is not possible
30	ARFAFull	ARF can receive only one more quadlet when “1”
29	ARF4Th	More than 4 quadlets of data are written in ARF when “1”
28	ARFDc	This is the control bit for reading a packet from ARF, and is “1” only when the first and last quadlets of the packet are read.
27	ARFAEmpty	Only one more quadlet of data is written in ARF when “1”
26	ARFEmpty	ARF is empty when “1” and there is no data to be read
23	ATFFull	ATF is full when “1” and write is not possible
22	ATFAFull	ATF can write only one more quadlet when “1”
21	ATF4Avail	More than 4 quadlets of data can be written in ATF when “1”
20	ATFAEmpty	ATF has only one more quadlet of data not transmitted when “1”
19	ATFEmpty	ATF is empty when “1” and there is no data for transmit
15	ClearATF	Sync reset of ATF when “1” (Self Clear)
13	ClearARF	Sync reset of ARF when “1” (Self Clear)
3 to 0	ATAck	Value of received Ack code

10) Phy Chip Access Registers

These registers are used for read/write of the contents of the Phy chip Phy register connected to the CXD1948R.

The register address is 28h and the initial value is 0000_0000h.

Bit	Name	Function
31	RdPhy	The CXD1948R requests read to the address set in PhyRegAd via the Phy I/F when "1"
30	WrPhy	The CXD1948R requests write to the address set in PhyRegAd via the Phy I/F when "1"
27 to 24	PhyRegAd	Sets the read/write address of the connected Phy chip Phy register
23 to 16	PhyRegData	Value of data for write to address specified by PhyRegAd
11 to 8	PhyAdRxReg	Value of read Phy register address during read
7 to 0	PhyDataRxReg	Value of the read Phy register data during read

11) Tx1394Hdr Registers

These registers are used to set the value of 1394Hdr added at the CXD1948R during isochronous transmit, and the number of data blocks transmitted in the isochronous cycle.

The register address is 30h and the initial value is 0000_0000h.

Bit	Name	Function
31 to 28	NOSP	Sets number of source packets transmitted in the isochronous cycle
26 to 24	NODB	Sets number of data blocks transmitted in the isochronous cycle
15 to 0	Tx1394Hdr	Header when transmitting isochronous packet bit15 to bit14: Tag [1 : 0], Isochronous data format tag bit13 to bit8: ChNumber [5 : 0], Isochronous Channel bit7 to bit6: Rsv [1 : 0], Reserved bit5 to bit4: Speed [1 : 0], Speed code bit3 to bit0: Sy [3 : 0], Synchronization Code

12) TxCIPHdr1 Registers

These registers are used to set the value of CIPHdr1 added at the CXD1948R during isochronous transmission and settings related to the 27MHz time stamp.

CIPHdr1 is one of the CIP headers.

The register address is 34h and the initial value is 0000_0070h.

Bit	Name	Function
23 to 8	TxCIPHdr1	Value of CIPHdr1 added during isochronous transmit bit31 to bit30: "00" bit23 to bit16: DBS, one data block size (quadlet) bit15 to bit14: FN [1 : 0], Franction number bit13 to bit11: QPC [2 : 0], Quadlet padding Count bit10: SPH, Source packet Header bit9 to bit8: rsv [1 : 0] reserved
7	27MTIF	27MHz time stamp value invalid at "1" 27MHz time stamp value valid at "0"
6 to 4	27Mrsv	Reserve area when 27MHz time stamp is used

13) TxCIPHdr2 Registers

These registers are used to set the value of CIPHdr2 added at the CXD1948R during isochronous transmission. CIPHdr2 is one of the CIP headers.

The register address is 38h and the initial value is 0000_0000h.

Bit	Name	Function
31 to 16	TxCIPHdr2	Value of CIPHdr2 added during isochronous transmit bit31 to bit30: "10" bit29 to bit24: FMT [5 : 0], Format ID bit23: TF, Time Shift 1: TimeShift 0: no TimeShift bit22 to bit0: Format Dependet Field

14) SPH-reserved/AddData1, 2 Registers

These registers are used for setting the reserve value of the source packet header added/detected by the CXD1948R during isochronous communication, and for setting and storing additional data and setting the number of additional data.

The register address is 3Ch and the initial value is 0000_0000h.

Bit	Name	Function
31 to 25	SPH-reserved	Value of source packet header reserve stored during receive or added during isochronous communication
19 to 16	AddSize	Sets number of data added during isochronous transmission
15 to 8	AddData1	Value of data 1 stored during receive or added during isochronous communication
7 to 0	AddData2	Value of data 2 stored during receive or added during isochronous communication

15) AddData3 to 6 Registers

These registers are used for the data added/detected by the CXD1948R during isochronous communication.

The register address is 40h and the initial value is 0000_0000h.

Bit	Name	Function
31 to 24	AddData3	Value of data 3 stored during receive or added during isochronous communication
23 to 16	AddData4	Value of data 4 stored during receive or added during isochronous communication
15 to 8	AddData5	Value of data 5 stored during receive or added during isochronous communication
7 to 0	AddData6	Value of data 6 stored during receive or added during isochronous communication

16) AddData7 to 10 Registers

These registers are used for the data added/detected by the CXD1948R during isochronous communication.

The register address is 44h and the initial value is 0000_0000h.

Bit	Name	Function
31 to 24	AddData7	Value of data 7 stored during receive or added during isochronous communication
23 to 16	AddData8	Value of data 8 stored during receive or added during isochronous communication
15 to 8	AddData9	Value of data 9 stored during receive or added during isochronous communication
7 to 0	AddData10	Value of data 10 stored during receive or added during isochronous communication

17) Rx1394Hdr Registers

These registers are used to store the value of the 1394Hdr detected by the CXD1948R during isochronous receive.

The register address is 48h and the initial value is 0000_0000h.

Bit	Name	Function
31 to 8 3 to 0	Rx1394Hdr	Value of 1394Hdr detected during isochronous receive bit31 to bit16: Data_length [15 : 0], 1 isochronous packet byte length bit15, bit14: Tag [1 : 0], Isochronous data format tage bit13 to bit8: ChNumber [5 : 0], Isochronous channel Number bit7 to bit4: tCode [3 : 0], Isochronous Transaction code bit3 to bit0: sy [3 : 0] Synchronization Code

18) RxCIPHdr1 Registers

These registers are used to store the value of the CIPHdr1 detected by the CXD1948R during isochronous receive.

The register address is 4Ch and the initial value is 0000_0000h.

Bit	Name	Function
31 to 0	RxCIPHdr1	Value of CIPHdr1 detected during isochronous receive bit31: EOH bit30: form bit29 to bit24: SID, Source [5 : 0] Node ID bit23 to bit16: DBS [7 : 0], Data Block Size bit15, bit14: FN [1 : 0] Fraction bit13 to bit11: QPC [2 : 0], Quadlet Padding Count bit10: SPH, Source Packet Header bit9, bit8: rsv [1 : 0], Reserved bit7 to bit0: DBC [7 : 0], Data Block Counter

19) RxCIPHdr2 Registers

These registers are used to store the value of the CIPHdr2 detected by the CXD1948R during isochronous receive.

The register address is 50h and the initial value is 0000_0000h.

Bit	Name	Function
31 to 0	RxCIPHdr2	Value of CIPHdr2 detected during isochronous receive bit31: EOH bit30: form bit29 to bit24: FMT [5 : 0], Format ID bit23: TF, TimeShift Flag bit22 to bit0: FDF [22 : 0], Format Dependent on Fields

20) IPBWrite (first quadlet of the packet) Registers

The first quadlet of the transmitted insert packet is written in these registers.

The register address is 64h and the initial value is 0000_0000h.

Bit	Name	Function
31 to 0	IPBWrite (first quadlet of the packet)	Writes the first quadlet of the transmitted insert packet.

21) IPBWrite Registers

The second through the next to the last quadlets of the transmitted insert packet are written in these registers.
The register address is 68h and the initial value is 0000_0000h.

Bit	Name	Function
31 to 0	IPBWrite	Writes the second through the next to the last quadlets of the transmitted insert packet.

22) IPBWrite (confirm write) Registers

The last quadlet of the transmitted insert packet is written in these registers.
The register address is 6Ch and the initial value is 0000_0000h.

Bit	Name	Function
31 to 0	IPBWrite (confirm write)	Writes the last quadlet of the transmitted insert packet.

23) ATFWrite (first quadlet of the packet) Registers

The first quadlet of the transmitted asynchronous packet is written in these registers.
The register address is 70h and the initial value is 0000_0000h.

Bit	Name	Function
31 to 0	ATFWrite (first quadlet of the packet)	Writes the first quadlet of the transmitted asynchronous packet.

24) ATFWrite/ARFRead Registers

The second through the next to the last quadlets of the transmitted asynchronous packet are written in these registers. Also, the asynchronous packet read from the ARF during receive is read one quadlet at a time.
The register address is 74h and the initial value is 0000_0000h.

Bit	Name	Function
31 to 0	ATFWrite /ARFRead	Transmit: Writes the second through the next to the last quadlets of the transmitted asynchronous packet. Receive: Writes the asynchronous packet read from ARF one quadlet at a time.

25) ARFWrite (confirm write) Registers

The last quadlet of the transmitted asynchronous packet is written in these registers.
The register address is 7Ch and the initial value is 0000_0000h.

Bit	Name	Function
31 to 0	ATFWrite (confirm write)	Writes the last quadlet of the transmitted asynchronous packet.

26) Parallel Port Registers

These registers are used to set the parallel port I/O direction and data.

The register address is 2Ch and the initial value is 0000_0000h.

Be sure to set all non-pertinent registers to “0”.

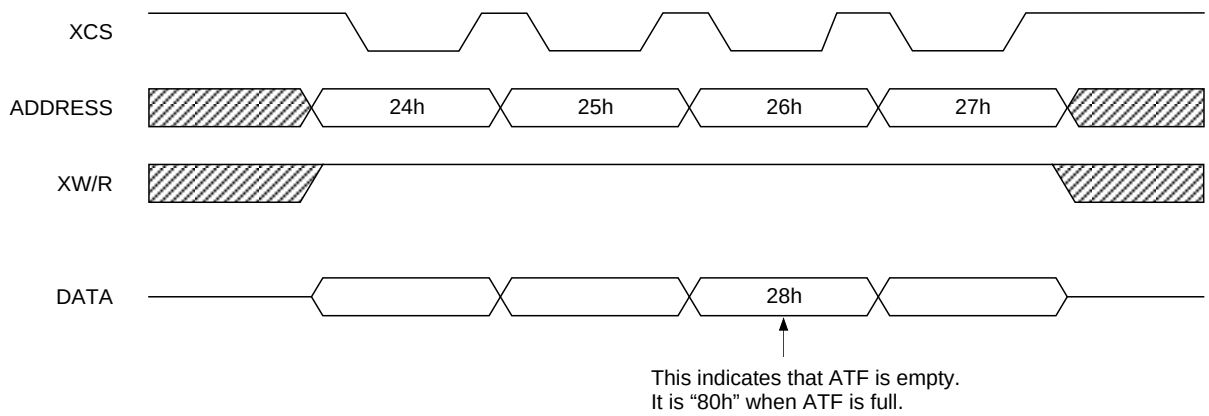
Bit	Name	Function
12	PDIR4	PORT4 I/O switching setting. 0: Input, 1: Output
11	PDIR3	PORT3 I/O switching setting. 0: Input, 1: Output
10	PDIR2	PORT2 I/O switching setting. 0: Input, 1: Output
9	PDIR1	PORT1 I/O switching setting. 0: Input, 1: Output
8	PDIR0	PORT0 I/O switching setting. 0: Input, 1: Output
4	PDATA4	Sets the PORT4 output data when PDIR4 is 0. Inputs the PORT4 input data when PDIR4 is 1.
3	PDATA3	Sets the PORT3 output data when PDIR3 is 0. Inputs the PORT3 input data when PDIR3 is 1.
2	PDATA2	Sets the PORT2 output data when PDIR2 is 0. Inputs the PORT2 input data when PDIR2 is 1.
1	PDATA1	Sets the PORT1 output data when PDIR1 is 0. Inputs the PORT1 input data when PDIR1 is 1.
0	PDATA0	Sets the PORT0 output data when PDIR0 is 0. Inputs the PORT0 input data when PDIR0 is 1.

6-3. Asynchronous Packet Transmission

Packet data is written from the external microcomputer to the ATF inside the CXD1948R in order to transmit an asynchronous packet. At this time the first quadlet of the packet only is written in the CFR ATFWrite (first quadlet of the packet) register (70h). The second through the next to the last quadlets are written in the CFR ATFWrite/ARFRread registers (74h). Then the last quadlet is written in the CFR ATFWrite (confirm write) register (7Ch) and the packet is stored in the ATF.
However, if the ATF is full, write will not actually be performed even when write is executed.
Once the bus is enabled, transmit takes place automatically.

The procedure for transmitting a quadlet write request packet is given here as an example.
(for 8-bit data interface)

(1) Confirming that ATF is not full
The CFR Async Status register (24h to 27h) is read to confirm that the 23rd bit (AtfFull bit) is low. If it is high it means that there are some unsent packets stored and it waits until they are transmitted.

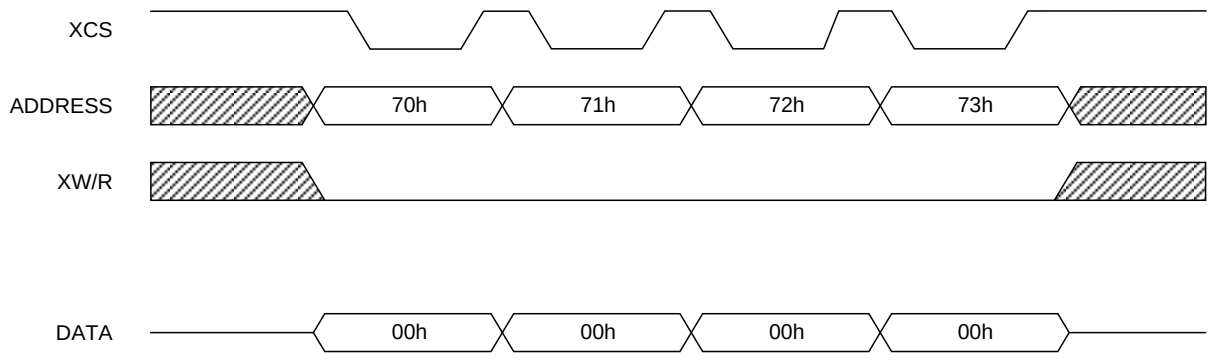


The number of quadlets that can be stored in ATF can be found from the value of the Async Status register bits [23 : 19]. The following six states can be found, so a judgement must be made as to whether write is possible from the number of quadlets in the packet being sent from the external microcomputer.

AtfFull = High:	Can’t write
AtfAFull = High:	only 1 quadlet
All bits low:	2 to 3 quadlet
Atf4Avail = High:	4 to 28 quadlet
Atf4Avail = High, AtfAEmpty = High:	29 quadlet
Atf4Avail = High, AtfEmpty = High:	30 quadlet

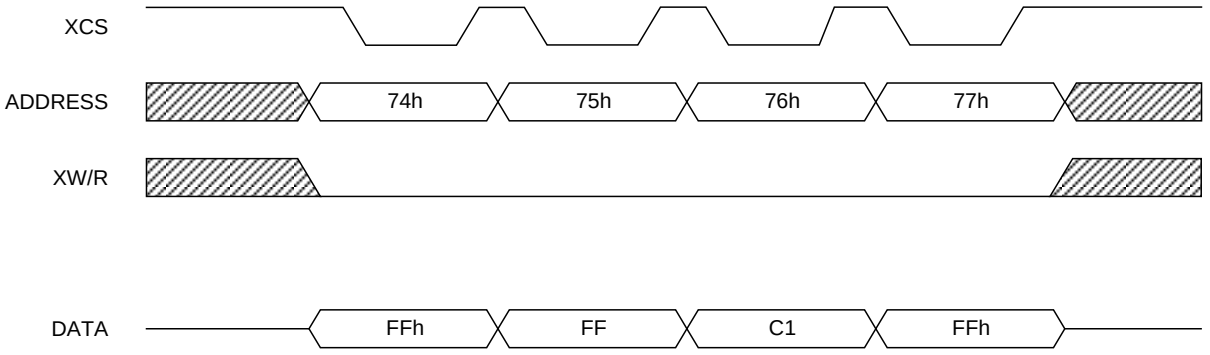
(2) First quadlet of the transmitted packet Write

Let the first quadlet of the quadlet write request packet be “00000000h”.
This is written in the CFR ATFWrite (first quadlet of the packet) register.



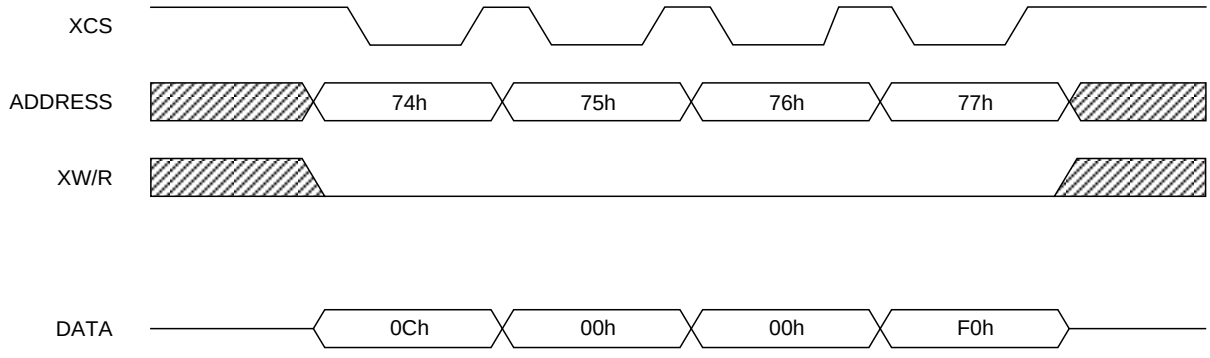
(3) Second quadlet of the transmitted packet Write

Let the second quadlet of the quadlet write request packet be “FFC1FFFFh”.
This is written in the CFR ATFWrite/ARFRead register.



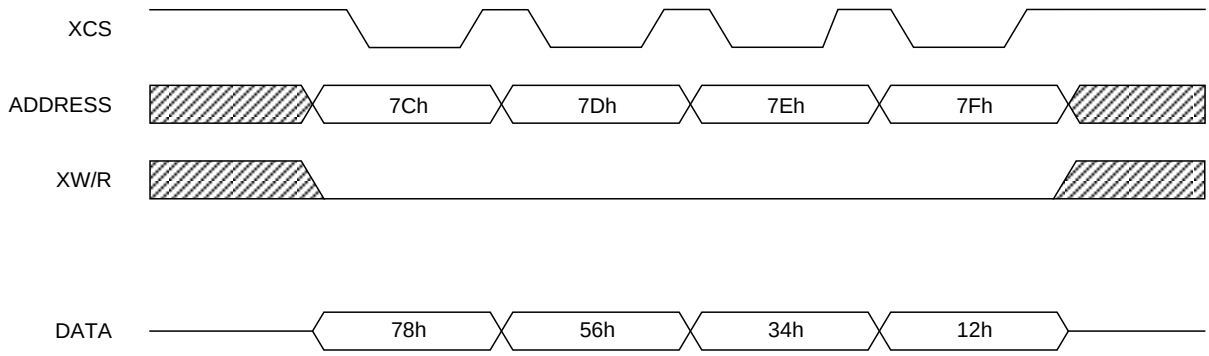
(4) Third quadlet of the transmitted packet Write

Let the third quadlet of the quadlet write request packet be “F000000Ch”.
This is written in the CFR ATFWrite/ARFRead register.



(5) Last quadlet of the transmitted packet Write

Let the last quadlet of the quadlet write request packet be “12345678h”.
This is written in the CFR ATFWrite (confirm write) register.



The quadlet write request packet is stored in the ATF as shown above. When the bus is enabled, the CXD1948R transmits automatically. If transmit does not take place, the CFR interrupt register (0Ch to 0Fh) must be read to confirm if the ATStk bit or TCErr bit is high. If these bits are high, the packet stored in the ATF may not be correct.

ATStk = High: If the first quadlet of the packet was not written in the CFR ATFWrite (first quadlet of the packet) register but was written in the ATFWrite/ARFRead register or the ATFWrite (confirm write) register.

TCErr = High: A value that is not a transaction code able to be transmitted by asynchronous packet is written in the tCode field of the first quadlet of the packet.
The transactions codes that can be transmitted as asynchronous packets are any of (0, 1, 2, 4, 5, 6, 7, 9, B, Eh).

For either of ATStk or TCErr above, the next packet for write will not be transmitted even if it is correct.
At this time “1” must be written in the Async Status register ClearATF bit in order to clear the ATF. Transmit is then enabled when a correct packet is written.

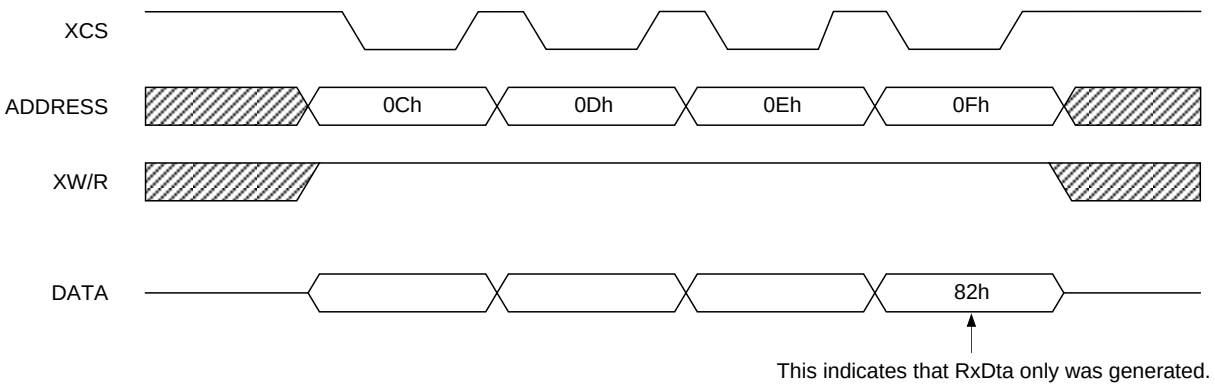
6-4. Asynchronous Packet Reception

Basically, if there is room to write the packet in FIFO and the destination_ID matches, then asynchronous packets are received. Receive is completed when the packet data is read from the ARF inside the CXD1948R by the external microcomputer.

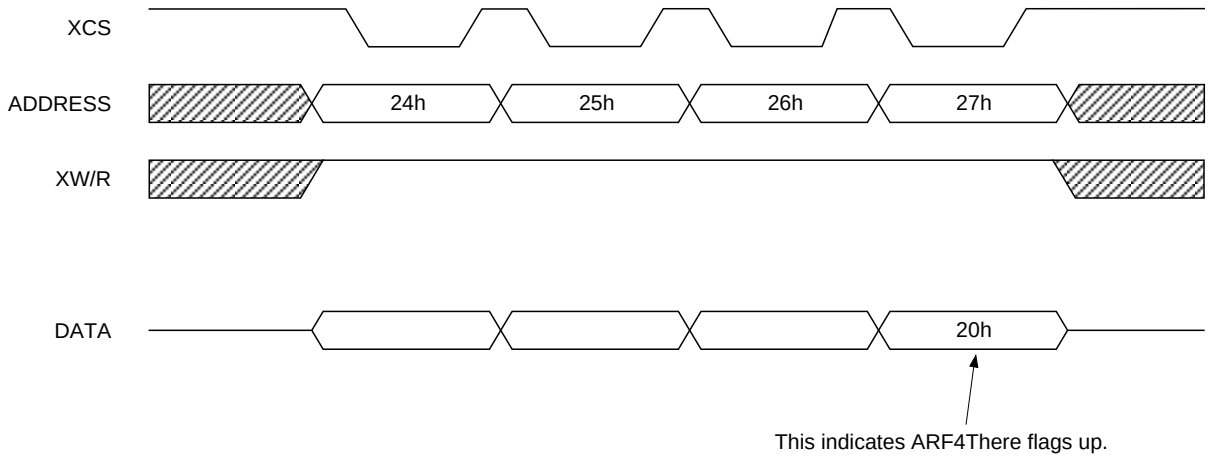
The CXD1948R raises an RxDta flag when a packet is received. (Normally, if the RxDta bit of the CFR Interrupt Mask register (10h to 14h) is set at “1”, XINT goes low when a packet is received and this can be detected.)
Next, the CFR Async Status register (24h to 28h) ArfEmpty bit should be low. This indicates that a correct packet was received.
After this, one quadlet at a time can be read by reading the CFR ATFWrite/ARFRead registers (74h to 78h). Packet receive is completed by repeating this until the ArfEmpty bit goes high.
However, if the ARF status is empty, read will not be done even if Read is executed. In this case, the data read by the microcomputer will be the previously read value.

The procedure for receiving a quadlet write request packet is given here as an example.
(for 8-bit data interface)

(1) Confirming that the packet was received
The CFR Interrupt register (0Ch to 0Fh) is read to confirm that the 25th bit (RxDta bit) is high.



(2) Confirming that the received packet was stored correctly in FIFO
The CFR Async Status register (24h to 27h) is read to confirm that the 26th bit (ArfEmpty bit) is low.
If this bit is high it means that reception may be in progress (all quadlets have not arrived). Read can not be done in this state, so wait and then clear again.

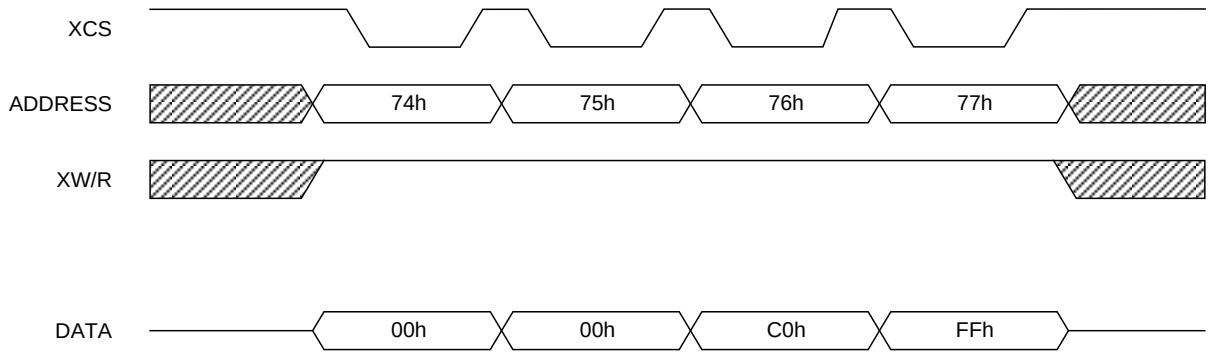


In the above example, the Arf4There are high and ArfEmpty is low.
This means that the packet stored in FIFO has more than 4 quadlets, including the footer (quadlet to which the Link that received the packet is added automatically; speed and ackSent, etc. are written).

ArfDc is “1” only when the read quadlet is a start quadlet or a footer of packet.
The read quadlet can be identified whether it is a start quadlet or a footer by confirming the ArfDc bit of the Async Status register (24h to 27h) after reading the qualet.

(3) First quadlet of the received packet Read

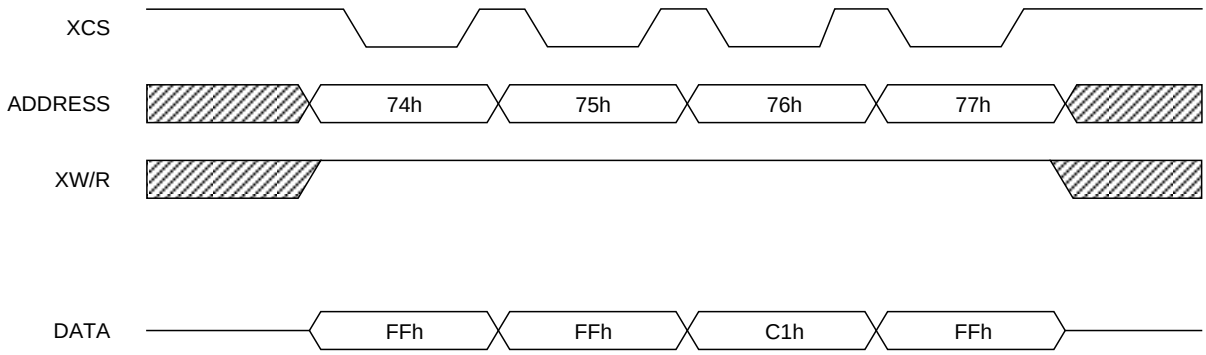
The CFR ATFWrite/ARFRead register is read.



The data read is “FFC00000h”. The read quadlet can be identified as a start quadlet by confirming the ArfDc bit of the Async Status register (24h to 27h) after reading the qualet.

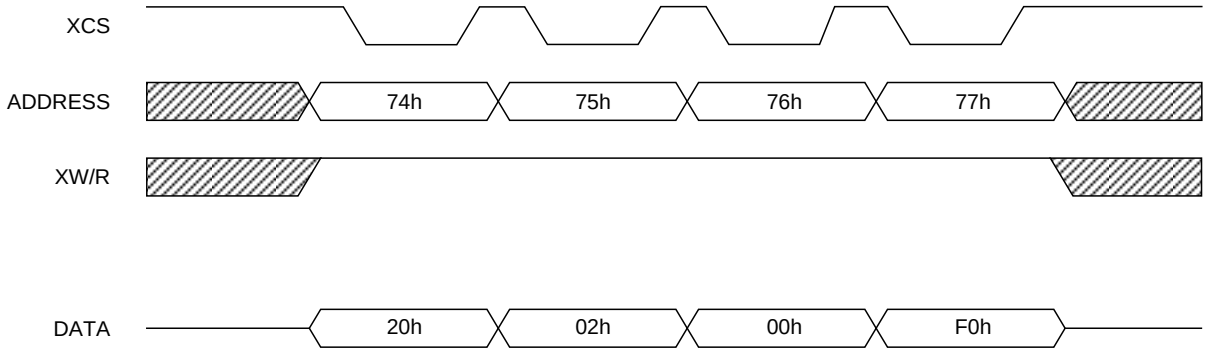
(4) Second quadlet of the received packet Read

The CFR ATFWrite/ARFRead register is read. As in (2) before, when the status is read after this read, the ArfDc bit should be low.



(5) Third quadlet of the received packet Read

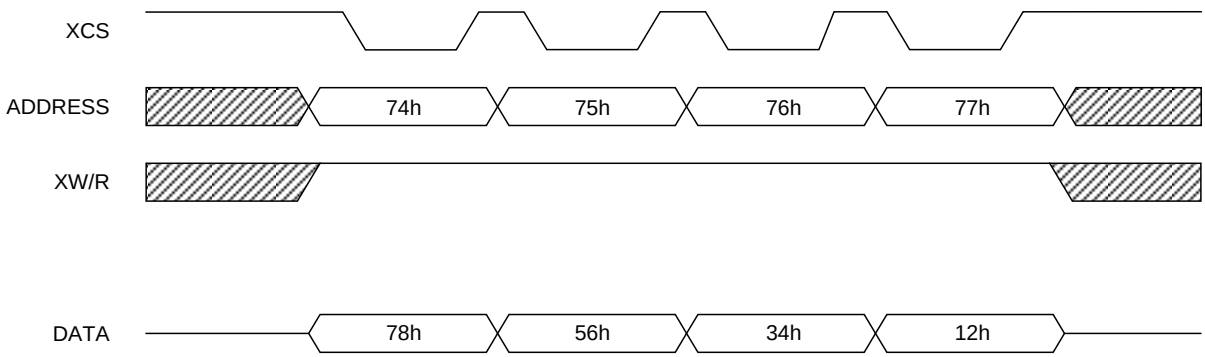
The CFR ATFWrite/ARFRead register is read.



The data read is “F0000220h”.

(6) Fourth quadlet of the received packet Read

The CFR ATFWrite/ARFRead register is read.

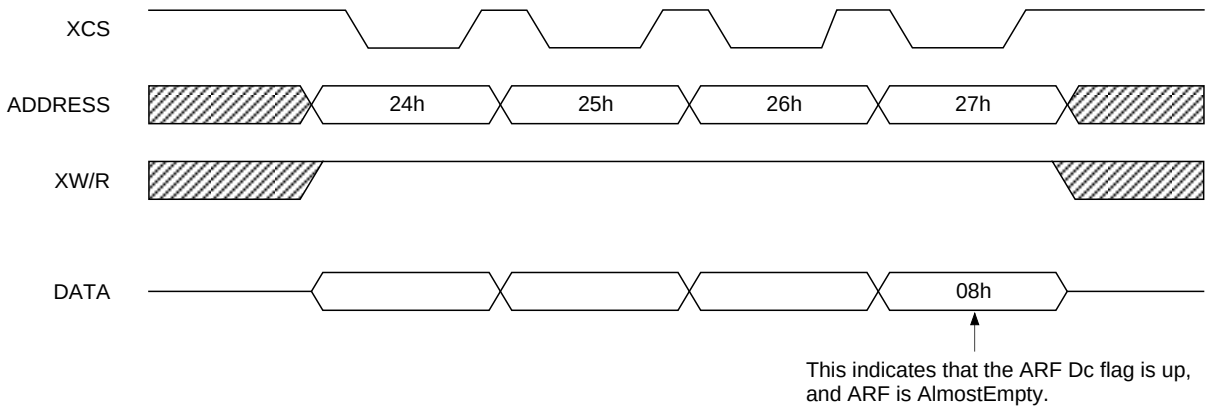


The data read is “12345678h”.

(7) Checking for remaining packets still in FIFO

4 quadlets were read in preceding items (1) to (6). They were read continuously because Arf4There was high. If Arf4There was low, Async Status must be read after 1 quadlet is read, to find out if ArfEmpty is high.

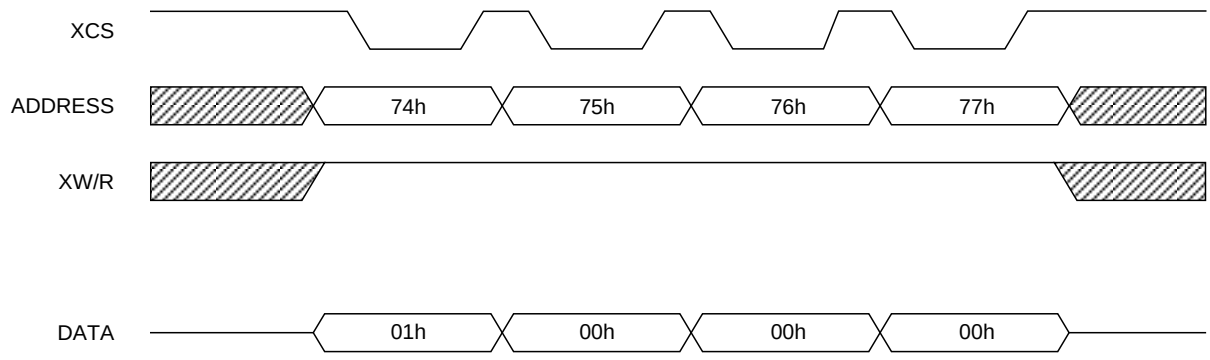
Even if Arf4There is high, as in this case, after the 4th quadlet read must be done while checking ArfEmpty in the same way.



In the above example the Arf4There bit is low, so a maximum of 3 more quadlets can be predicted, but the ArfAEmpty bit is high, so there is only one more quadlet in FIFO. Therefore, the probability is high where the quadlet to be read next is a footer.

(8) Fifth quadlet of the received packet Read

The CFR ATFWrite/ARFRead register is read.



The data read is “00000001h”.

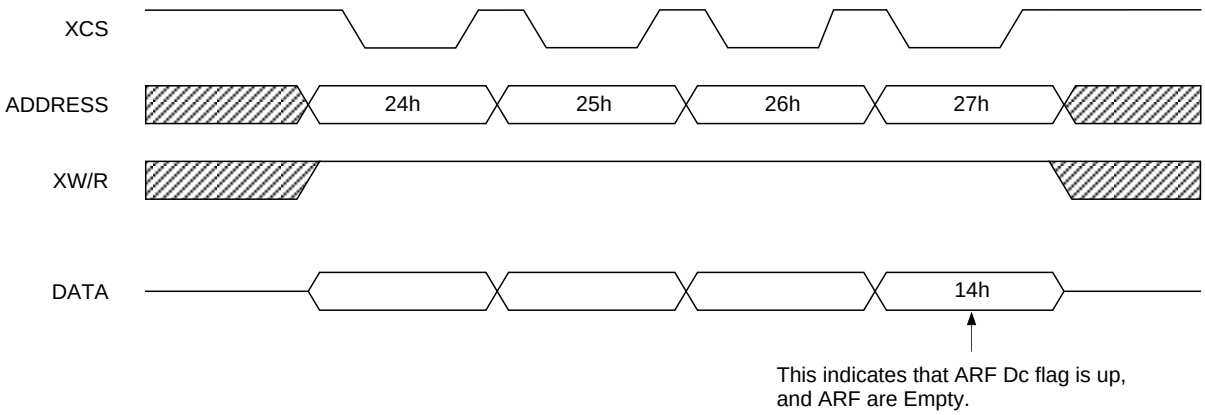
The lower 4 bits of this quadlet are the ackSent field, and this indicates “01h” transmitted as this packet’s Ack_code. This is always written even if the packet is one which does not have Ack_code transmitted, such as a broadcast packet.

If this value is “04h”, the ARF may have become full during receive and quadlets may be missing.

If it is “0Dh”, an error was detected in the data field CRC check of the received packet, or data_length and the actual data length do not match.

(9) Checking for remaining packets still in FIFO

The last quadlet was read in (8) before, so there should be no more packets in FIFO. In addition, it is confirmed whether the quadlet read last is a footer at the same time.



As seen above, it can be confirmed that ARF is Empty and the quadlet read last is a footer.

This completes asynchronous packet reception.

6-5. CXD1948R Data Format

6-5-1. Asynchronous Transmit (Host Bus (CXD1948R)

The following are the four basic formats for asynchronous data during transmit.

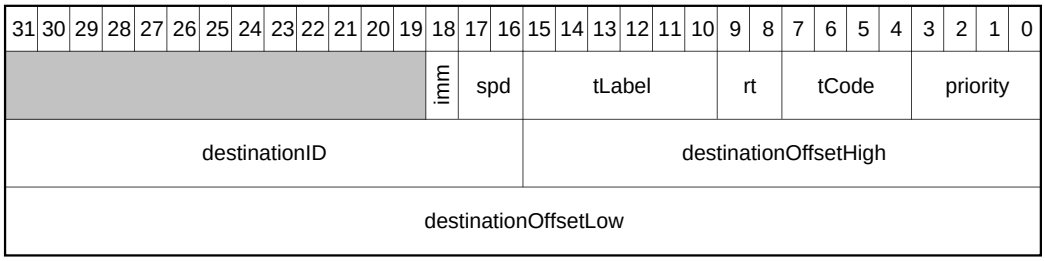
a) No-data Packets (Used for quadlet read requests and all write responses.)
b) Quadlet Packets (Used for quadlet write requests, quadlet read request and block read responses.)
c) Block Packets (Used for lock requests, lock responses, block write requests and block read responses.)
d) Unformatted data

6-5-1-1. No-data Transmit

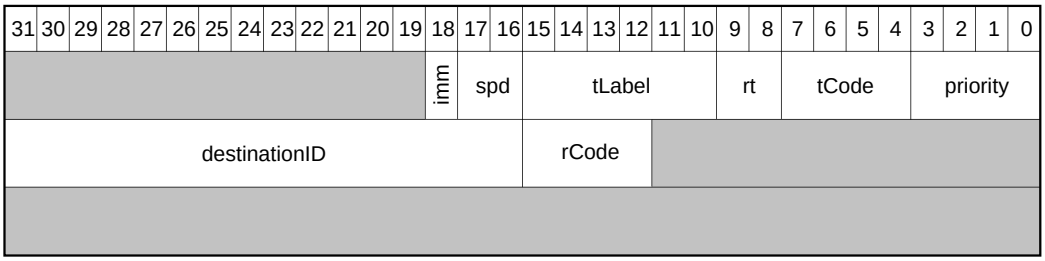
The data format for no-data transmit is shown below.

The 1st quadlet contains packet control information. The 2nd and 3rd quadlets contain 16-bit Destination ID and 48-bit Destination Offset for request, or Response Code for response.

Quadlet Read Request Transmit Format



Write Response Transmit Format



No-data Transmit Fields

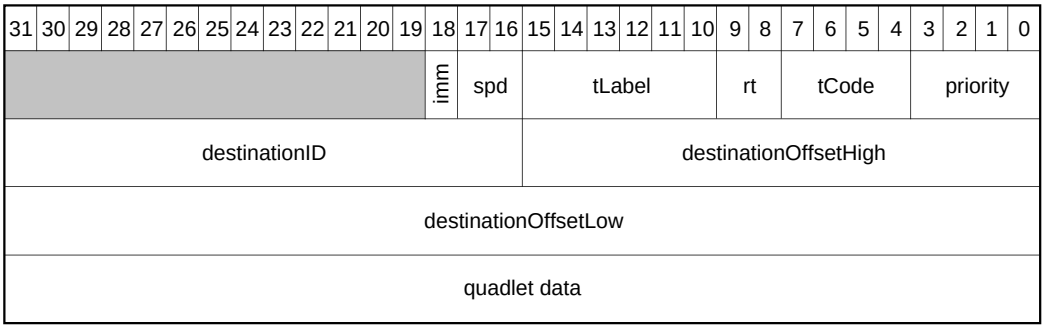
Field Name	Description
imm	Immediately tries to transmit continuously after Acknowledge is sent, if “1” is set (Used for Phy Read and Lock Response.)
spd	Transmit speed 00: 100Mbps; 01: 200Mbps; 10: 400Mbps; 11: Reserved
tLabel	Transaction Label. Used as a pair with response packet relative to request packet.
rt	This packet’s Retry Code 00: New; 01: Retry-X, 10: Retry-A, 11: Retry-B
tCode	This packet’s Transaction Code
priority	This packet’s Priority Level For values other than “0”, the transmitter uses Priority Arbitration relative to this packet
destinationID	Indicates this packet’s Destination bus number in 10 bits and the Node number in 6 bits
destinationOffsetHigh, destinationOffsetLow	Two continuous such areas indicate Destination Node address space address. This address must be in quadlet units.
rCode	Response Code for write response packet

6-5-1-2. Quadlet Transmit

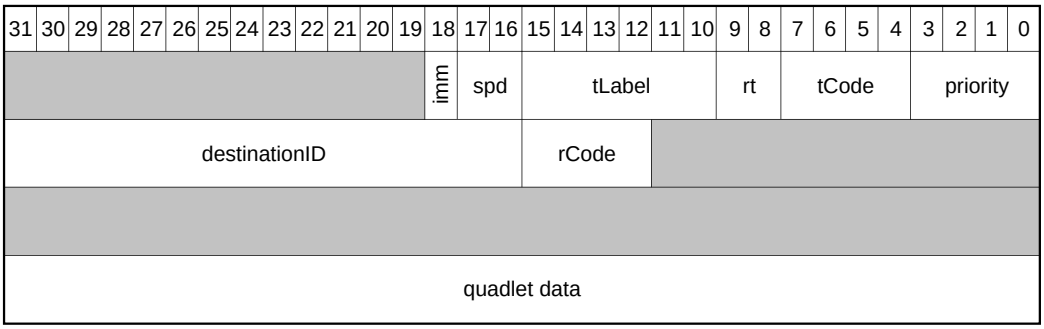
The data format for quadlet transmit is shown below.

The 1st quadlet contains packet control information. The 2nd and 3rd quadlets contain 16-bit Destination ID and 48-bit Destination Offset for request, or Response Code for response. The 4th quadlet is quadlet data for Read responses and Write Quadlet requests. It is Data Length and Reserved for Block Read request.

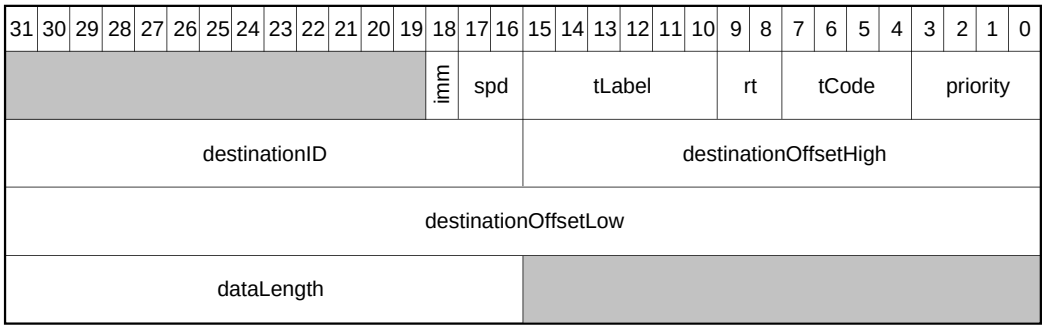
Quadlet Write Request Transmit Format



Quadlet Read Response Transmit Format



Block Read Request Transmit Format



Quadlet Transmit Fields

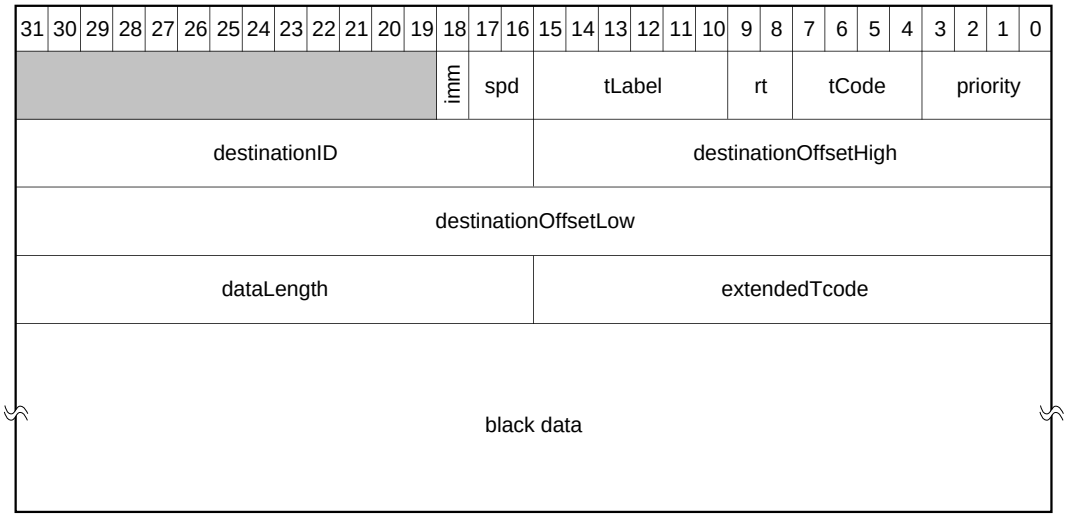
Field Name	Description
imm	Immediately tries to transmit continuously after Acknowledge is sent, if “1” is set (Used for Phy Read and Lock Response.)
spd	Transmit speed 00: 100Mbps; 01: 200Mbps; 10: 400Mbps; 11: Reserved
tLabel	Transaction Label. Used as a pair with response packet relative to request packet.
rt	This packet’s Retry Code 00: New; 01: Retry-X, 10: Retry-A, 11: Retry-B
tCode	This packet’s Transaction Code
priority	This packet’s Priority Level For values other than “0”, the transmitter uses Priority Arbitration relative to this packet
destinationID	Indicates this packet’s Destination bus number in 10 bits and the Node number in 6 bits
destinationOffsetHigh, destinationOffsetLow	Two continuous such areas indicate Destination Node address space address. This address must be in quadlet units.
quadlet data	Writes transmit data for Quadlet Write requests and Quadlet Read response
rCode	Response Code for Quadlet Response packet
dataLength	Writes how many bytes requested for Block Read request

6-5-1-3. Block Transmit

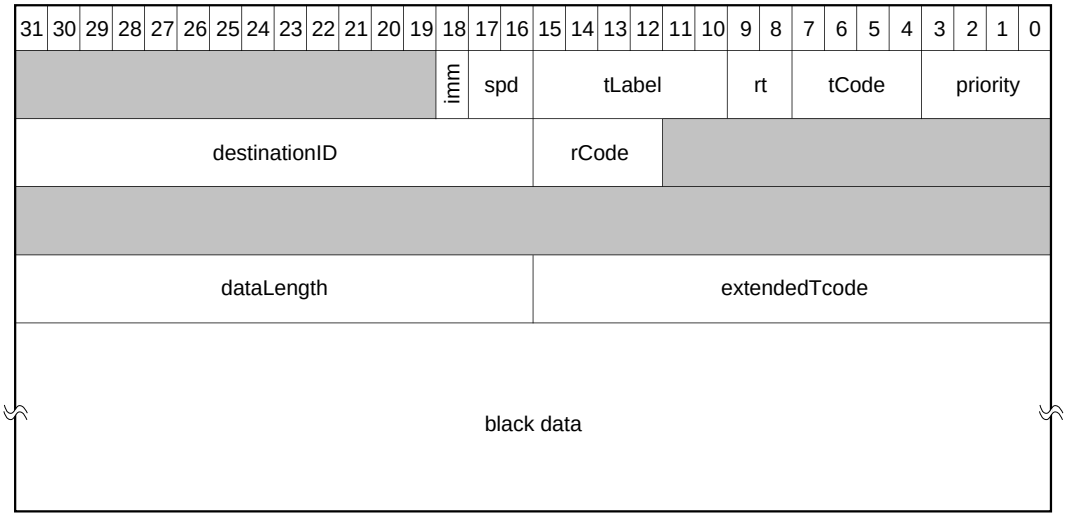
The data format for block transmit is shown below.

The 1st quadlet contains packet control information. The 2nd and 3rd quadlets contain 16-bit Destination ID and 48-bit Destination Offset for request, or Response Code for response. The 4th quadlet contains data length and Extended Transaction Code (all “0” except for Lock Transaction).

Block Transmit Format



Block Read or Lock Transmit Format



Block Transmit Fields

Field Name	Description
imm	Immediately tries to transmit continuously after Acknowledge is sent, if “1” is set (Used for Phy Read and Lock Response.)
spd	Transmit speed 00: 100Mbps; 01: 200Mbps; 10: 400Mbps; 11: Reserved
tLabel	Transaction Label. Used as a pair with response packet relative to request packet.
rt	This packet’s Retry Code. 00: New; 01: Retry-A, 10: Retry-A, 11: Retry-B
tCode	This packet’s Transaction Code.
priority	This packet’s Priority Level. For values other than “0”, the transmitter uses Priority Arbitration relative to this packet.
destinationID	Indicates this packet’s Destination bus number in 10 bits and the Node number in 6 bits.
destinationOffsetHigh, destinationOffsetLow	Two continuous such areas indicate Destination Node address space address. This address must be in quadlet units.
quadlet data	Writes transmit data for Quadlet Write requests and Quadlet Read response.
rCode	Response Code for Quadlet Response packet.
dataLength	Writes how many bytes requested for Block Read request.
extendedtCode	Specifies actual Lock Action performed by this packet data when tCode is a Lock Transaction.
block data	Writes transmitted data. This data is not written in FIFO when dataLength = “0”. The first byte of the block must indicate the upper byte of the first data, regardless of data Destination or Source listing.

6-5-1-4. Unformatted Transmit (Phy Configuration Packet)

The data format for unformatted transmit during Phy Configuration packet transmit is shown below.
The 1st quadlet contains packet control information. The remaining quadlets contain data, and get on the bus and are transmitted regardless of format. There is no CRC attached to the packet data.
Further, there is no CRC attached to the first quadlet.
Logical-inverse is not added at Link Core, so it must be added when transmitting.

Unformatted Transmit Format 1 (PHY configuration Packet)

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="14"></td><td>spd</td><td colspan="8"></td><td>1</td><td colspan="4">tCode
= 1110</td><td colspan="4">priority</td></tr>
<tr><td>00</td><td colspan="6">phy_ID</td><td>R</td><td>T</td><td colspan="6">gap_cnt</td><td colspan="4">0000</td><td colspan="4">0000</td><td colspan="4">0000</td><td colspan="4">0000</td></tr>
<tr><td colspan="32">logical inverse of 2nd quadlet data</td></tr>
</table>

Unformatted Transmit (PHY Configuration Packet) Fields

Field Name	Description
00	Indicates that the transmit packet is a Phy configuration packet.
phy_ID	Sets the force_root bit of the node with this Phy_ID sets to “1”. (Only valid when R is set at “1”.)
R	Sets the force_root bit of the node with this Phy_ID to “1” when set at “1”, and clears other nodes’ force_root bit. The Phy_ID area is ignored when set at “0”.
T	When set to “1”, all nodes use the value of the gap_cnt field to reset their own gap_Count values.
gap_cnt	Indicates values of all node new Phy_CONFIGURATION.gap_Count. This value becomes valid when it is received. (when the T field is “1”)

6-5-1-5. Unformatted Transmit (Link-on Packet)

The data format for unformatted transmit during Link-on packet transmit is shown below.

The 1st quadlet contains packet control information. The remaining quadlets contain data, and are transmitted regardless of format. There is no CRC attached to the packet data. Further, there is no CRC attached to the first quadlet.

Logical-inverse is not added at Link Core, so it must be added when transmitting.

Unformatted Transmit Format 1 (Link-on Packet)

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
														spd									1	tCode = 1110				priority			
01	phy_ID							0000				0000				0000				0000				0000				0000			
logical inverse of 2nd quadlet data																															

Unformatted Transmit (Link-on Packet) Fields

Field Name	Description
01	Indicates that the transmit packet is a Link-on packet.
phy_ID	Indicates the Phy chip ID of this packet's Destination.

6-5-2. Asynchronous Receive (CXD1948R → Host Bus)

The following are the three basic formats for asynchronous data during receive.

a) No-data Packets (Used for quadlet read requests and all write responses.)

b) Quadlet Packets (Used for quadlet write requests, quadlet read request and block read responses.)

c) Block Packets (Used for lock requests, lock responses, block write requests and block read responses.)

The names of receive data areas and their contents are given below.

Async Receive Fields

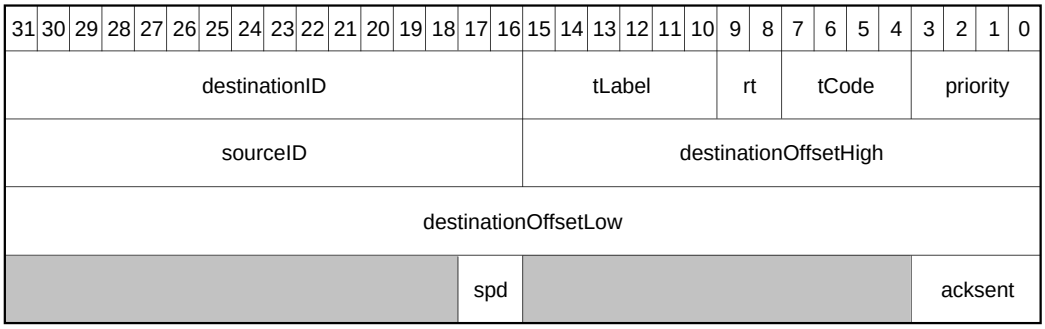
Field Name	Description
destinationID	This node's bus number (all "0" if "local bus") and node number (all "1" if broadcast)
tLabel	Transaction Label. Used as a pair with response packet relative to request packet.
rt	This packet's Retry Code. 00: New; 01: Retry-X, 10: Retry-A, 11: Retry-B
tCode	This packet's Transaction Code
priority	This packet's Priority Level
sourceID	The Node ID of the node that sent this packet
destinationOffsetHigh, destinationOffsetLow	Two continuous such areas indicate Destination Node address space address. This address must be in quadlet units.
rCode	Response Code for Response packet
quadlet data	Received data is written for quadlet write requests and quadlet read response.
dataLength	The number of bytes in received block type's packet data.
extendedtCode	Specifies actual Lock Action performed by this packet data when tCode is a Lock Transaction.
block data	Received data is written. This data is not written in FIFO when dataLength = "0". The first byte of the block must indicate the upper byte of the first data, regardless of data destination or Source listing.
spd	Speed of received packet. 00: 100Mbps; 01: 200Mbps; 10: 400Mbps; 11: Reserved
ackSent	Acknowledge code sent by Link Core relative to this packet is written.

6-5-2-1. No-data Receive

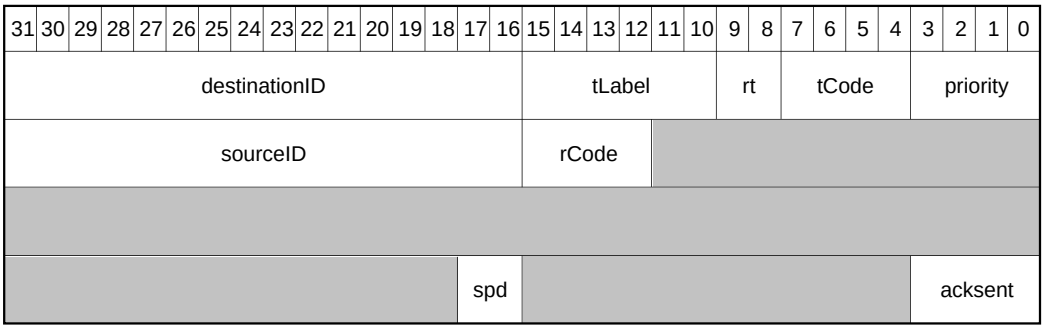
The data format for no-data receive is shown below.

The 1st quadlet contains the destination ID and other packet headers. The 2nd and 3rd quadlets contain 16-bit source ID, and 48-bit destination offset for request and the Response Code for response. The last quadlet contains packet receive status.

Quadlet Read Request Receive Format



Write Response Receive Format



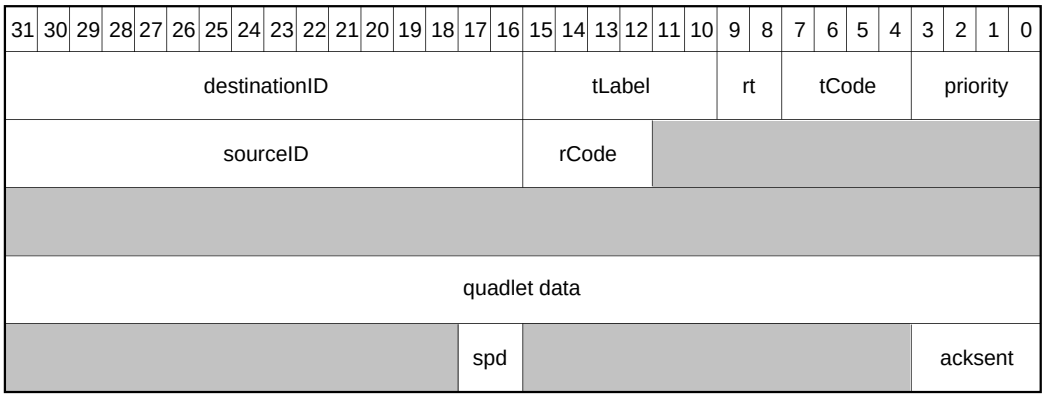
6-5-2-2. Quadlet Receive

The format for Quadlet Receive is shown below.

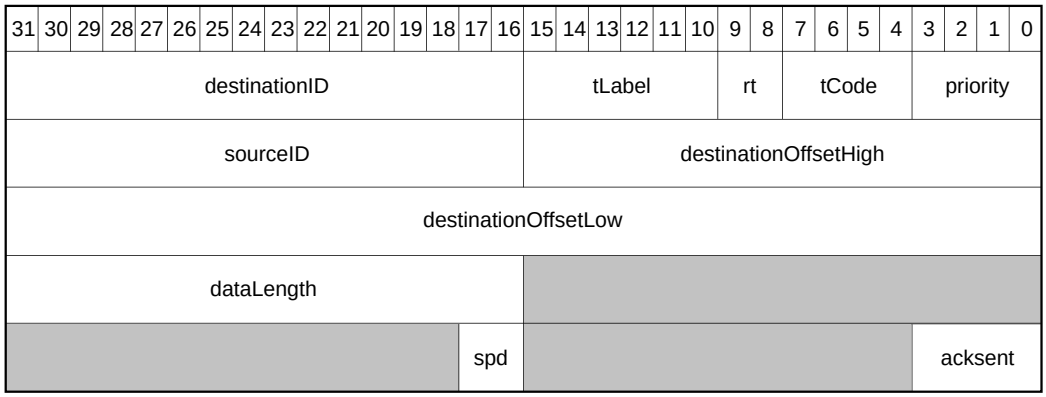
The 1st quadlet contains the destination ID and other packet headers. The 2nd and 3rd quadlets contain 16-bit source ID, and 48-bit destination offset for request and the Response Code for response. The 4th quadlet contains data for Read request and Write Quadlet request, and data length and Reserved for Block Read request. The last quadlet contains packet receive status.

Quadlet Write Request Receive Format

Quadlet Read Response Receive Format



Block Read Request Receive Format

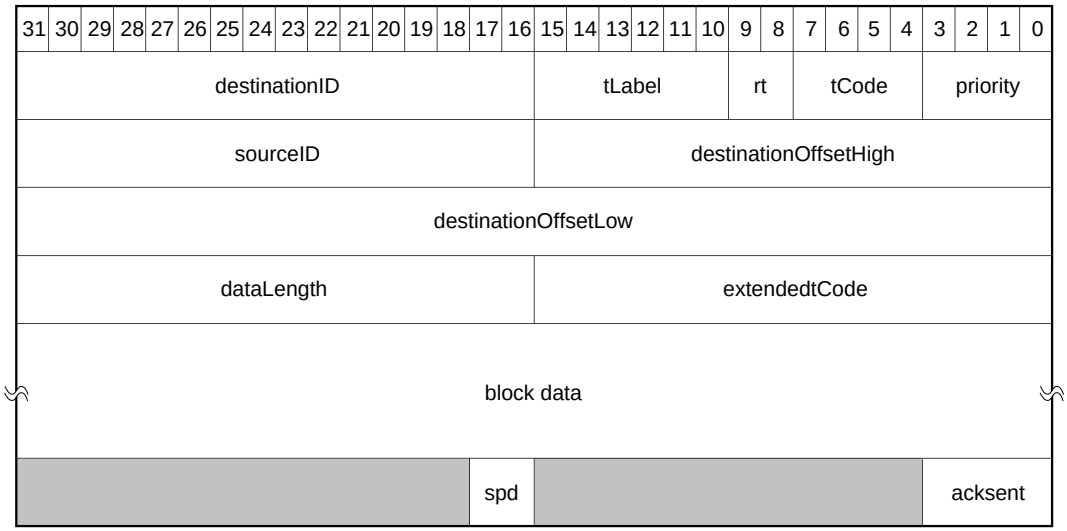


6-5-2-3. Block Receive

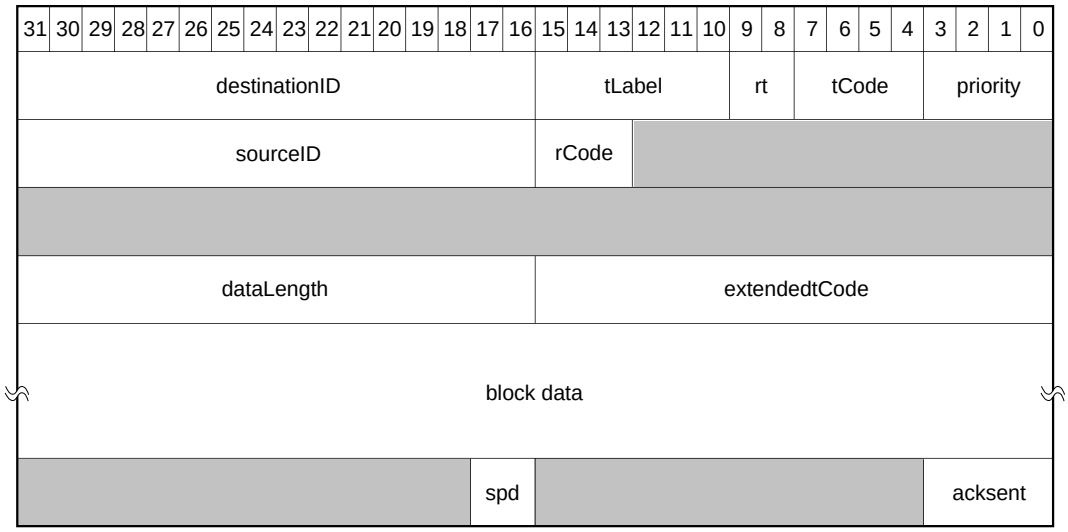
The format for Block Receive is shown below.

The 1st quadlet contains the destination ID and other packet headers. The 2nd and 3rd quadlets contain 16-bit source ID, and 48-bit destination offset for request and the Response Code for response. The 4th quadlet contains data length and Extended Transaction Code (all “0” except for Lock Transaction). This is followed by Block data. The last quadlet contains packet receive status.

Block Receive Format



Block Read or Lock Response Receive Format



6-6. Self-ID Packet Receiving Error Processing

In the Self-ID phase after bus reset on the CXD1948R, if the Self-ID packet could not be received correctly, Self-ID packet receive is stopped immediately and the Node_sum value becomes “0”.

The external microcomputer thus can judge that the Self-ID phase could not be completed correctly.

7. Insert Packet

7-1. Insert Packet Transmission

The CXD1948R has a function where the data set by the host I/F is inserted as isochronous packet and it is transmitted when the transport stream data is being isochronous-transmitted.
The transmit method is the same as for asynchronous packet transmit; it is done by accessing the specified CFR (refer to CFR Address Map) address (64 to 6C).

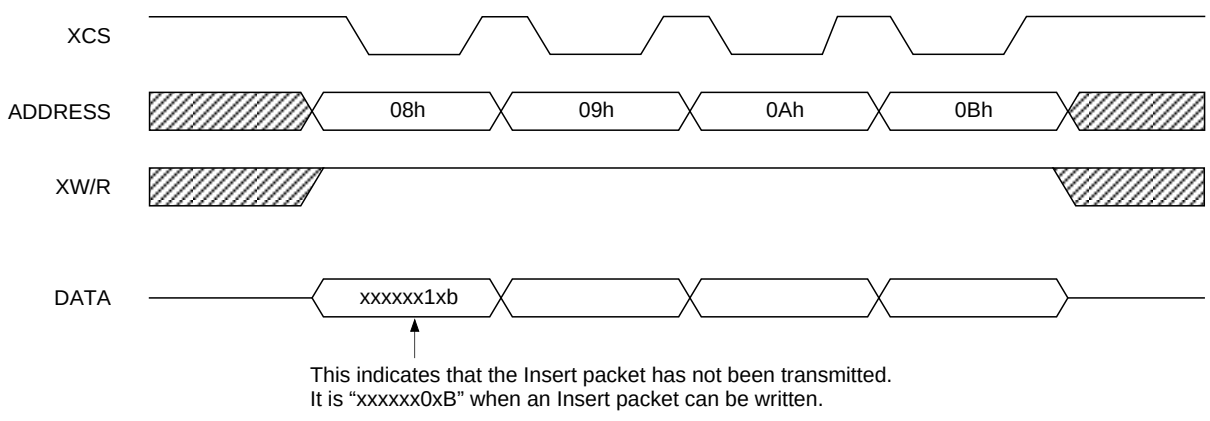
• Insert Packet Transmit
The external writes packet data in the CXD1948R internal IPF in order to perform insert packet transmit. At this time only the first quadlet of the packet is written in the CFR IPBWrite (first quadlet of the packet) register (64h). The second through the next to the last quadlets are written in the CFR IPBWrite register (68h).
Then the last quadlet is written in the IPBWrite (confirm write) register (6Ch) to store the packet in IPB.

There are two types of Insert packet transmission, as described below.
(1) Transmission by writing the last quadlet in IPBWrite (confirm write).
(2) Transmission by setting the CFR IPTxGo register to high.
If IPTxGo is used, the data written earlier in IPB can be transmitted as is, with no changes. The IPTxGo register only becomes “0” when Insert packet transmit is actually performed, or if it is erased from IPB due to transmit Late processing or the like.
The time stamp added to the Insert packet takes the fall timing of the Packet Gap input directly after IPTxGo register value changes from “0” to “1”.
Once the time stamp is added, processing is the same as for transport stream data.

The FIFO capacity for the Insert packet is 188 bytes. Data which exceeds this capacity can not be transmitted. In this case, the data up to 188 bytes is valid.
If the transmitted data is less than 188 bytes, all data bits other than those written are set to “1” and transmitted.
The written data is held after transmit, and to continue sending a packet with the same contents, it is only necessary to set the IPTxGo register to “1” after confirming that it changed to “0”.
Writing to IPB must be done in quadlet units.

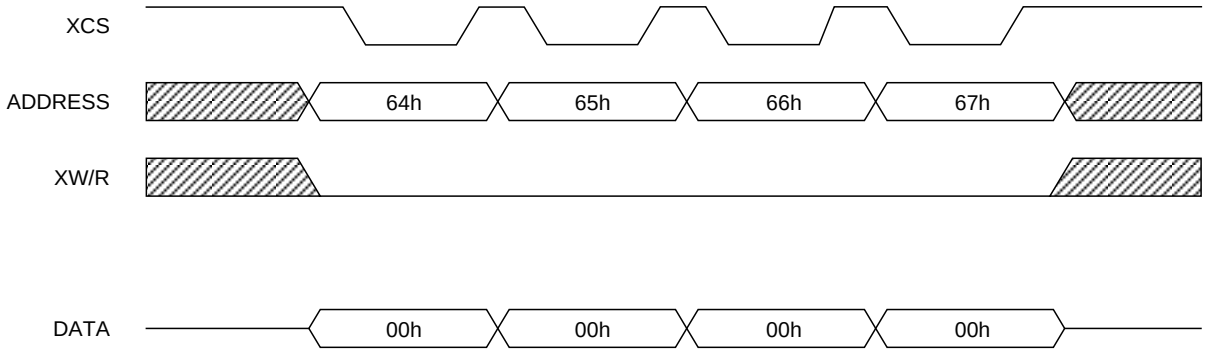
The procedure for transmitting a 4 quadlet Insert packet is described below.
(for 8 bit interface)

(1) Confirming that transmit of the previously written packet is complete
The CFR Control register (08h to 0bh) is read to confirm that its first bit (IPTxGo bit) is low. If it is high, it means that transmit of the previously written packet has not been performed.



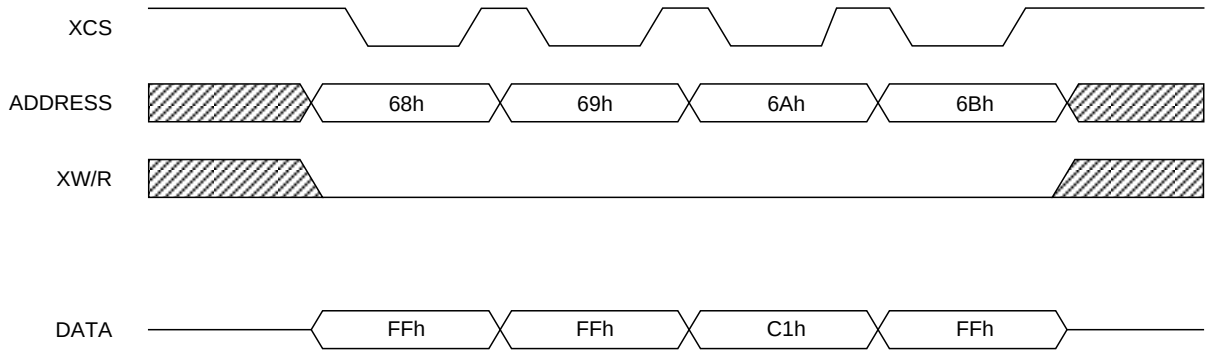
(2) First quadlet of the transmitted packet Write

Let the first quadlet of the quadlet write request packet be “00000000h”.
This is written in the CFR IPBWrite (first quadlet of the packet) register.



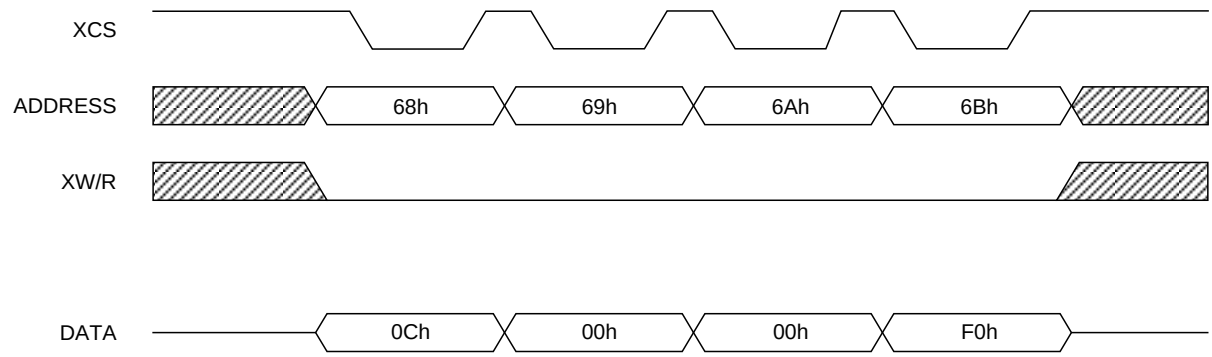
(3) Second quadlet of the transmitted packet Write

Let the second quadlet of the quadlet write request packet be “FFC1FFFFh”.
This is written in the CFR IPBWrite register.



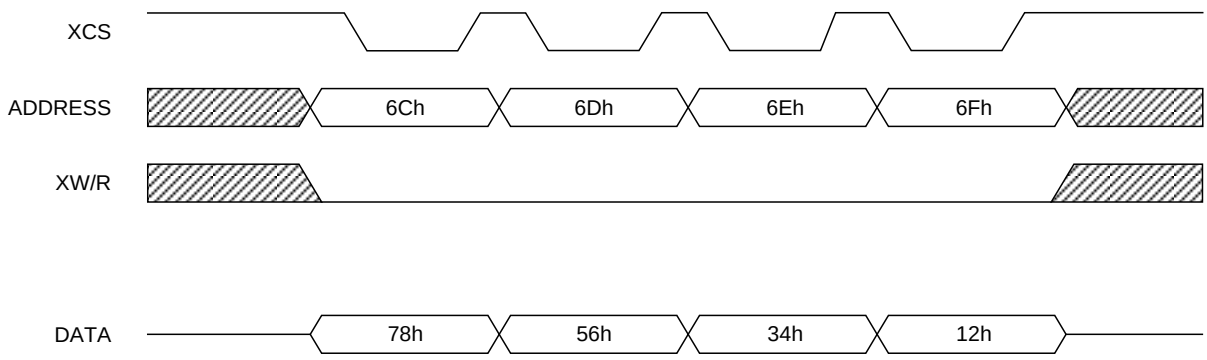
(4) Third quadlet of the transmitted packet Write

Let the third quadlet of the quadlet write request packet be “F000000Ch”.
This is written in the CFR IPBWrite register.



(5) Last quadlet of the transmitted packet Write

Let the last quadlet of the quadlet write request packet be “12345678h”.
This is written in the CFR IPBWrite (confirm write) register.

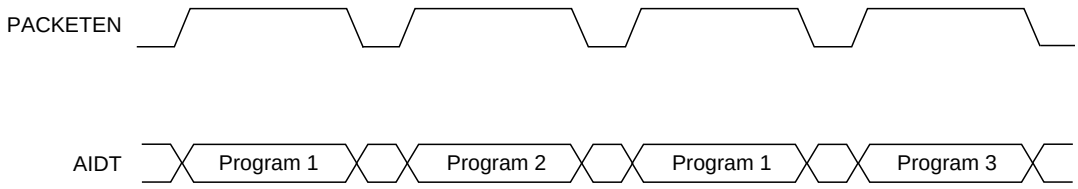


The 4-quadlet data is stored in the IPB as shown above. The CXD1948R transmits the packet with 1394 header and CIP header automatically attached, as with the case for the normal transport stream data, after the time stamp is added to the packet stored in the IPB. If Insert packet transmit does not take place due to Late processing, the CFR interrupt register (0Ch to 0Fh) must be read to confirm that the IpktLate bit is high.
Set the CFR IPTxGo register to high to add the time stamp when transmitting the Insert packet data again without updating its contents.

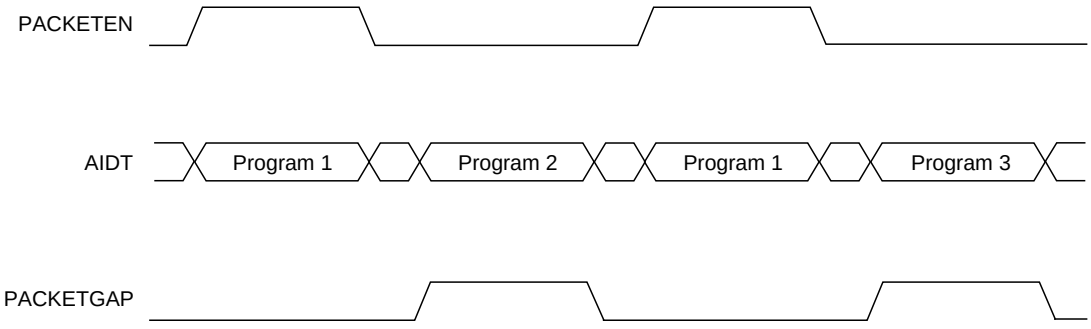
7-2. Adding a Time Stamp to the Insert Packet

A time stamp must be added to the Insert packet in order to process it as an isochronous packet.
The CXD1948R adds the time stamp using PACKETGAP input.
Concretely, the TxDelay register value added at the falling edge of the PACKETGAP input immediately after the Insert packet has been written to the IPB becomes the time stamp value of the Insert packet to be transmitted.
The time stamp must be added to the Insert packet during the interval when there is no transport stream data input.
When transmitting the entire transport stream, there is no gap during which to input PACKETGAP, so the Insert packet cannot be transmitted.
When transmitting the Insert packet, be sure to select a program from the transport stream and then perform transmission.
At this time, PACKETGAP input can add the time stamp to the Insert packet by inputting the PACKETEN of the non-selected programs as is.

When transmitting the entire transport stream



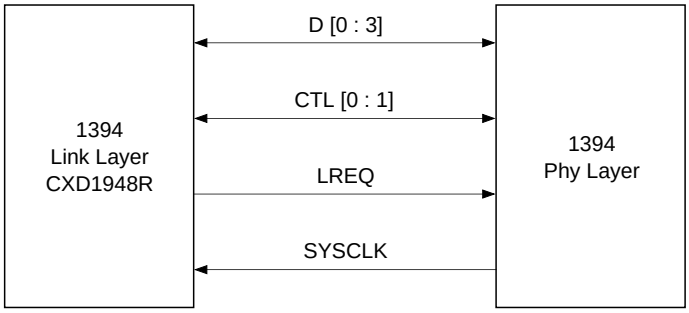
When transmitting only program 1 within the transport stream



8. Link-Phy Communication

8-1. Link-Phy Interface Specifications

The CXD1948R and Phy layer chip communicate using the four signals shown in the block diagram below: D [0 : 3], CTL [0 : 1], LREQ and SYSCLK.



The roles of the signals are as follows.

D [0 : 3]	in/out	Bidirectional data line
CTL [0 : 1]	in/out	Bidirectional control line
LREQ	out	LREQ out Request signal line from CXD1948R to Phy chip. Used for bus access and Phy register read/write requests.
SYSCLK	in	System clock (49.152MHz) supplied from Phy to the CXD1948R.

The types of communication and their contents are described below.

8-2. Communication

There are four types of communication between Phy Link: Request, Status, Transmit and Receive. Except for request, all commands are initialized by the Phy chip.

8-2-1. Bus controlling

CTL [0 : 1] controls communication between Phy and the CXD1948R. The communication contents differ depending on if Phy or the CXD1948R is controlling.

a) Phy controlling

CTL [0 : 1]	Name	Description of Activity
00	Idle	Bus is idle (Default mode)
01	Status	Phy is sending status information to the CXD1948R
10	Receive	Phy is sending a packet to the CXD1948R
11	Transmit	The CXD1948R authorized packet transmit

b) CXD1948R controlling

CTL [0 : 1]	Name	Description of Activity
00	Idle	The CXD1948R completed transmit
01	Hold	The CXD1948R is holding the bus until transmit preparations are complete. Or, the CXD1948R is trying to transmit another packet without arbitration.
10	Transmit	The CXD1948R is transmitting a packet to Phy
11	Reserved	Not used

8-2-2. Request

The CXD1948R always uses serial communication of LREQ to send a request to Phy when a request to the bus or access to the Phy register is required.

There are three types of request: Bus Request, Read Register Request and Write Register Request. The timing chart and contents are illustrated below.

a) Bus Request (length of stream: 7 bits)

Bit	Name	Description
0	Start Bit	Transmit start bit. Always “1”.
1 to 3	Request Type	Indicates type of request. (Refer to Request Type table.)
4 to 5	Request Speed	Indicates request communication speed. (Refer to Request Speed table.)
6	Stop Bit	Last transmit bit. Always “0”.

b) Read Register Request (length of stream: 9 bits)

Bit	Name	Description
0	Start Bit	Transmit start bit. Always “1”.
1 to 3	Request Type	Indicates type of request. (Refer to Request Type table.)
4 to 7	Address	Address for Phy register read.
8	Stop Bit	Last transmit bit. Always “0”.

c) Write Register Request (length of stream: 17 bits)

Bit	Name	Description
0	Start Bit	Transmit start bit. Always “1”.
1 to 3	Request Type	Indicates type of request. (Refer to Request Type table.)
4 to 7	Address	Address for Phy register write.
8 to 15	Data	Write data for Phy register specified by Address.
16	Stop Bit	Last transmit bit. Always “0”.

Request Type

LREQ [1 : 3]	Name	Description
000	ImmReq	Immediate bus acquisition request. To output Ack for an asynchronous packet reception, immediate bus acquisition is requested without arbitration when Idle is detected. Used to transmit Acknowledge.
001	IsoReq	Isochronous request. Requests execution of arbitration. Used for isochronous transmit.
010	PriReq	Priority request. Requests arbitration after Subaction Gap, ignoring Fair protocol. Used for Cycle Master request.
011	FairReq	Fair request Requests execution of arbitration after subaction gap according to fair protocol. Used for Fair transmit.
100	RdReq	Read request. Requests return of register contents according to Status Transfer.
101	WrReq	Write request. Requests write to specified address.
110, 111	Reserved	Reserved.

Request Speed

LREQ [4 : 5]	Data Rate
00	100Mb/s
01	200Mb/s
10	400Mb/s
11	Reserved

8-2-2-1. Bus Request

In order to access Fair or Priority, waits at least 1 clock after the CXD1948R becomes idle to send the request. When the CTL pin is in receive state (CTL [0 : 1] = 10), CXD1948R interprets the request as being refused. It is reissued 1 clock after the next idle state.

In the Cycle Master node, the cycle start message is sent using Priority Request. In order to request sending of isochronous data, the CXD1948R can issue an isochronous request after receiving cycle start. Phy clears the isochronous request only after the bus is acquired successfully.

The CXD1948R must issue ImmReq while it is receiving a packet addressed to itself in order to send Acknowledge. When packet reception is completed, Phy immediately acquires the bus and gives authorization to the CXD1948R. If the header CRC is not erroneous, the CXD1948R returns an Acknowledge signal. However, if the header CRC is erroneous, the CXD1948R immediately releases the bus. The CXD1948R can not use this authorization to send other packets. In order to ensure this operation, the CXD1948R waits 160ns after the completion of packet reception. Then Phy acquires the bus once again, and a CRC error Acknowledge is sent. Then it releases the bus and continues with another request.

Consider a case in which two different nodes confirm that the packet sent is addressed to them (one is correct, one is wrong), and both nodes issue an ImmReq before CRC check. The Phy of both nodes try to capture the bus immediately after packet receive is completed.
At this time, there will be a momentary collision on the local bus. This can be detected by all of the Phy connected to the bus. This collision is not interpreted as Bus Reset, but as high impedance state. After CRC check is completed, the wrong node will withdraw its request and the high impedance state is discontinued. The expected Acknowledge is lost as a side effect of this, but is processed by the host protocol.

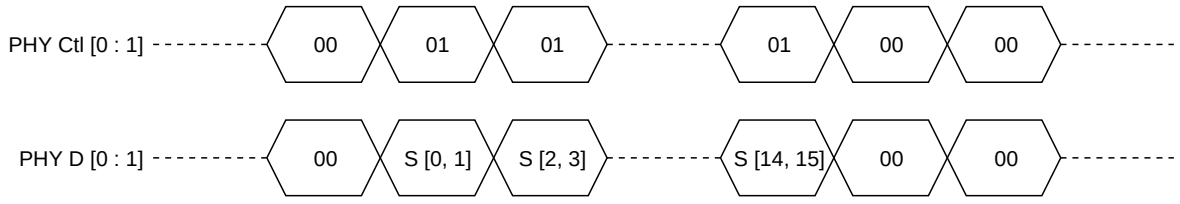
8-2-2-2. Read/Write Request

When the CXD1948R requests reading of a specific register's contents, Phy transmits the register contents to the CXD1948R by Status Transfer. Even if packets are received while Phy is sending status information to the CXD1948R, Phy continues processing until the register contents are transferred.

For a Write Request, Phy loads the data fields into the appropriate register as soon as transmission is completed. The CXD1948R can read/write at any time.

8-2-3. Status

Status transmission is started by Phy when it has some data to transmit to the CXD1948R. Phy begins transmission by simultaneously setting CTL [0 : 1] to "01b" and the first 2 bits of Status information to D [0: 1]. Phy maintains CTL = Status during status transmission.
Phy may finish Status transmission early by setting the CTL value to some other value. This happens if a packet arrives before Status transmit is completed. When such an interruption occurs, Phy repeatedly tries resending until the transfer is successful.
There must be at least one idle cycle in a continuous Status transmission.
Phy normally sends the first four bits of Status to the CXD1948R. These bits are the Status Flags required for the CXD1948R state machine. When transmission of a request containing a Read Request is completed, or when Phy has information to send to the CXD1948R or the transaction layer, Phy sends the first Status packet to the CXD1948R.
The only state in which Phy sends register contents automatically to the CXD1948R is that after completion of Self-Identification, and Physical_ID register contents containing a new node address are transmitted. The transmit timing and bit definitions are illustrated below.



Status bit (Length of stream: 16 bits)

Bit	Name	Description
0	Arbitration Reset Gap	Indicates detection of idle state, for Bus Arbitration Reset Gap Time. This bit is used by the CXD1948R busy/retry state machine.
1	Subaction Gap	Indicates detection of idle state, for Subaction Gap Time. This bit is used by the CXD1948R to detect the end of the isochronous cycle.
2	Bus Reset	Indicates Phy in bus reset state.
3	State Time-out	Indicates that Phy state machine is stopped in a certain state for a long time. Normally used for cable topology loop detection.
4 to 7	Address	Holds the address of the register being read when Phy is trying to send register contents to CXD1948R; for example, when responding to Read via the LREQ pin.
8 to 15	Data	Holds the register being sent to CXD1948R.

8-2-4. Transmit

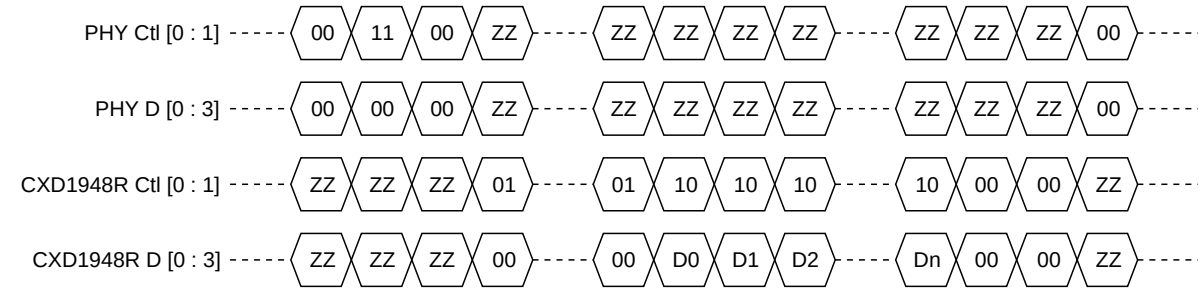
When the CXD1948R requests bus access via the LREQ pin, Phy performs arbitration for bus access. When Phy wins the arbitration, Transmit is sent to the CTL pin for at least 1 SYSCLK cycle, and Idle is then asserted for 1 cycle to give the CXD1948R the bus. After detecting transmit state from Phy, the CXD1948R asserts either Hold or Transmit to the CTL pins to take over interface control. The CXD1948R asserts Hold until the data is ready, in order to keep bus initiative. During this time, Phy asserts Data-on state to the bus. When the packet is ready to transmit , the CXD1948R transmits the first bit of the packet, and at the same time asserts Transmit to the CTL pins. After sending the last bit of the packet, the CXD1948R asserts either Idle or Hold to the CTL pins for 1 cycle. Then it asserts Idle for 1 cycle before these pins become high impedance.

Here, when it is necessary for the CXD1948R to send another packet without releasing the bus, it indicates Hold to Pht. In response to this, Phy asserts Transmit in the same way as before after waiting for the minimum required time. This function is used when multiple isochronous packets are sent in succession to different channels without the CXD1948R releasing the bus, and when a Response packet is sent following Acknowledge. When in this way multiple transmissions are performed in succession, all transmissions are at the same speed. This is because the speed is set at the time of arbitration to acquire the bus, and in successive transmissions intermediate arbitrations are skipped.

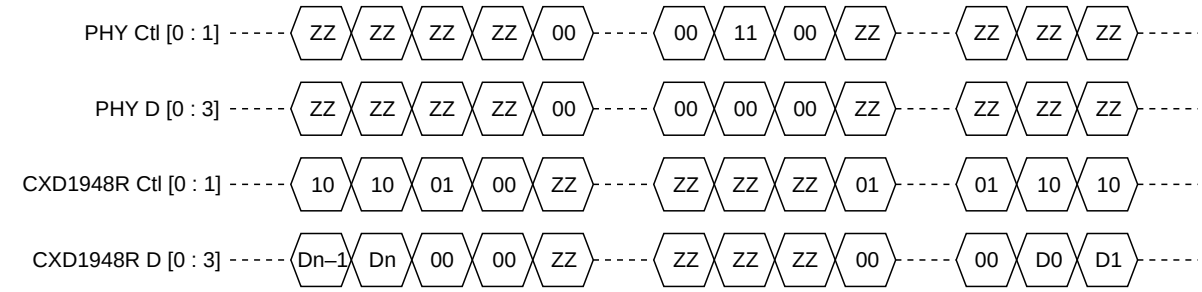
As described above, when the CXD1948R completes sending the last packet on the newest bus initiative, it releases the bus by asserting Idle to the CTL pins for 2 Sclk. When Phy detects Idle from the CXD1948R, it starts to assert Idle to CTL for 1 clock.

The timing chart for transmit is shown below.

Single Packet



Continued Packet

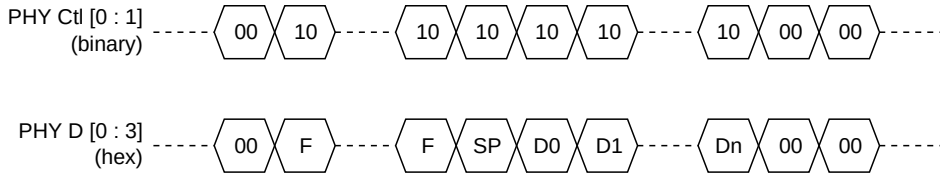


ZZ: High-impedance state, D0 to Dn: packet data

8-2-5. Receive

When data from the bus is received at Phy, it is sent from Phy to the CXD1948R in the following order.
Phy asserts Receive to the CTL pins and “all1” to the D pin. Phy indicates the start of the packet by placing a speed code on the D pin. Next it indicates the contents of the packet, and until transmission of the last symbol in the packet is completed, it holds the CTL pins at Receive. Phy indicates the end of the packet by asserting Idle to the CTL pins. The speed code is specified by Phy-Link protocol, and does not include CRC calculation or other data protect.
Phy can identify if there is data on the bus or not without looking at the packet. This also applies if a packet is being sent at a faster speed than Phy can receive. In this case, the packet is completed by asserting Idle when the Data-on state is completed.
If Phy supports a faster transmission speed than the CXD1948R, the CXD1948R detects the speed code and ignores the packet until it becomes Idle again.

The timing chart for reception is illustrated below.



Note: SP means Speed Code.

Speed codes for receive

D [0 : 3]	Data Rate
00xx	100Mbit/s
0100	200Mbit/s

Note 1: “xx” means that “0” was transmitted, but it is ignored for receive.
Note 2: This LSI supports 100Mbit/s and 200Mbit/s communications.

9. Parallel Input/Output Port

The CXD1948R has a 5-bit I/O port.
The direction of each port and the value during output mode can be set and the value during input mode can be read by accessing the CFR using the host interface.
The status after initializing the CXD1948R is input mode for all ports.

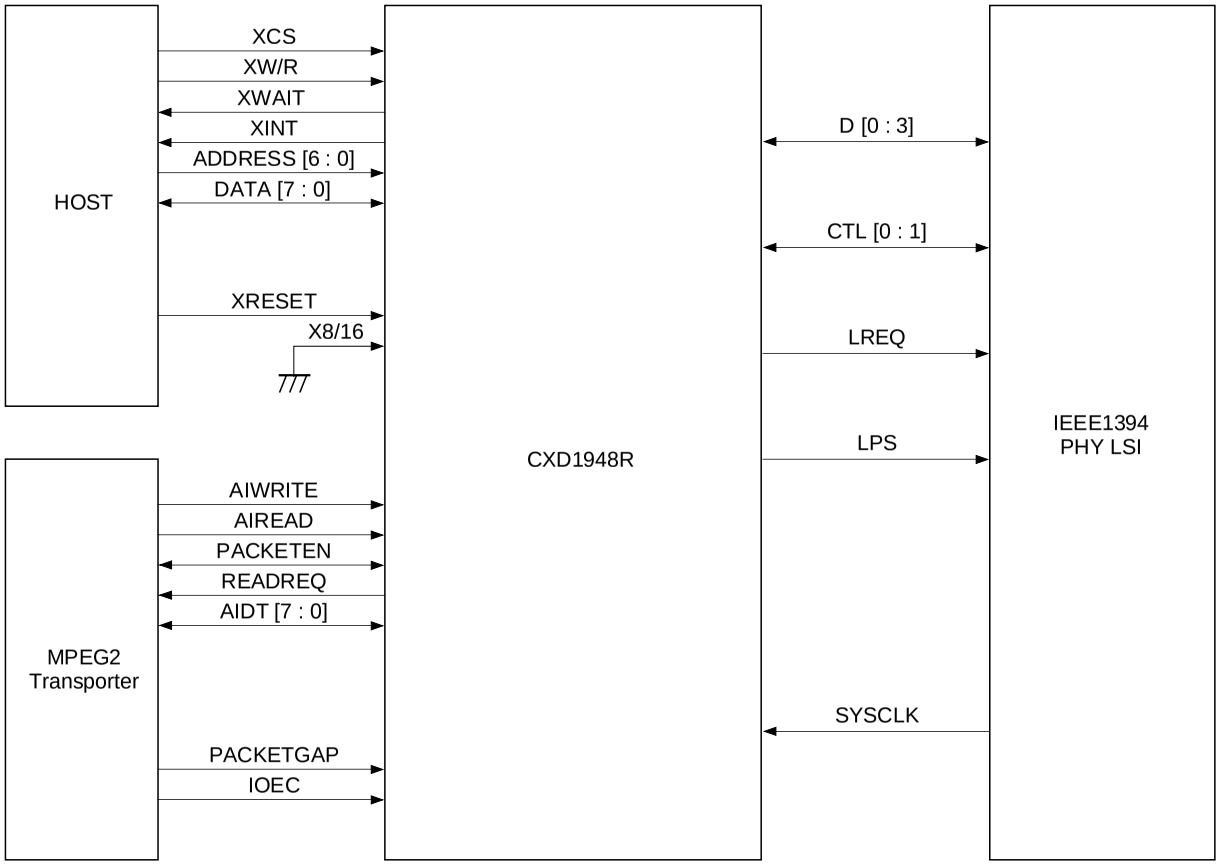
Symbol	Pin No.	CFR register name	
		Direction switching	Value setting (output)/value reading (input)
PORT0	44	PDIR0	PDATA0
PORT1	45	PDIR1	PDATA1
PORT2	46	PDIR2	PDATA2
PORT3	69	PDIR3	PDATA3
PORT4	70	PDIR4	PDATA4

Ports are set to output mode when the corresponding PDIR register is “1”, and to input mode when “0”.
Concretely, when PDIR0 is set to “1”, the value set in the PDATA0 register is output from the PORT0 pin.
If PDIR0 is set to “0”, the value input to the PORT0 pin is loaded to the PDATA0 register.
Each I/O port can be set independently.

10. System Configuration Example

HOST Interface 8bit

Application Interface 8bit/asynchronous/ERRFLAG not used



Annex. Corresponding Table for CFR Access Address And Host Interface I/O Data (8-bit Mode)

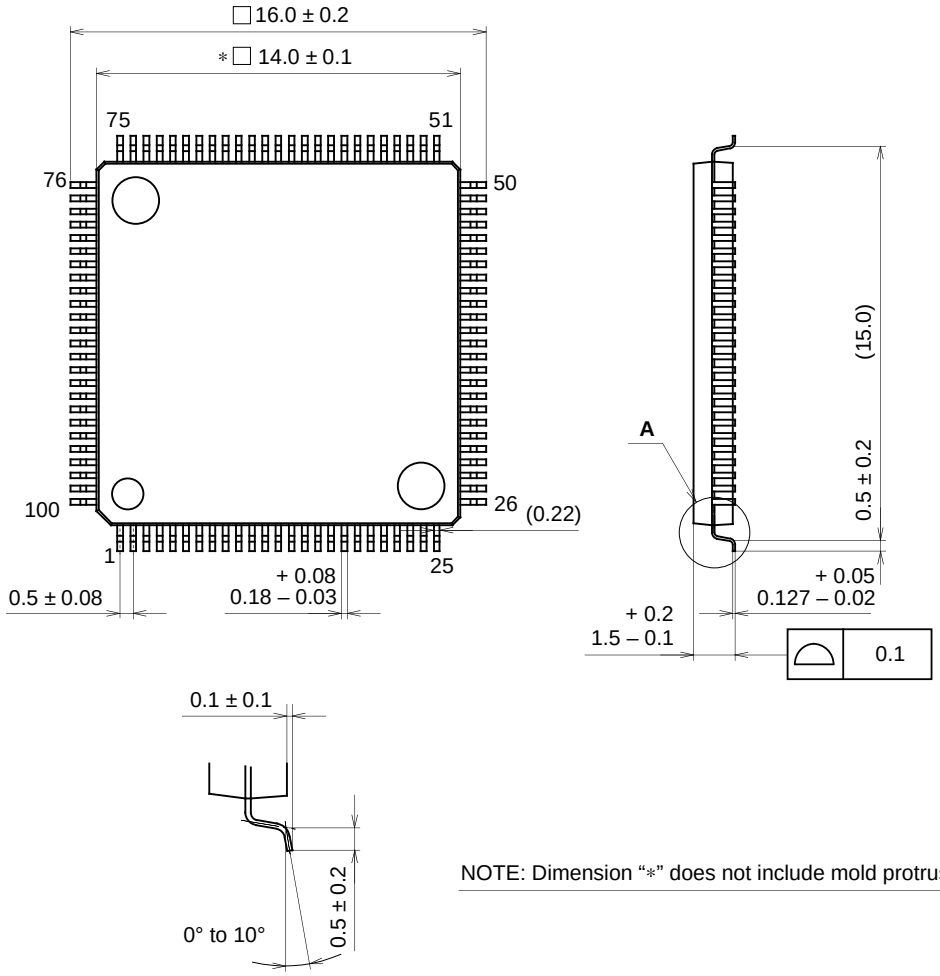
7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	I/O data bit
03h	02h	01h	00h	Version
07h	06h	05h	04h	Node Address
0bh	0ah	09h	08h	Control
0fh	0eh	0dh	0ch	Interrupt
13h	12h	11h	10h	Interrupt Mask
17h	16h	15h	14h	Cycle Timer
1bh	1ah	19h	18h	DIF Mode
1fh	1eh	1dh	1ch	Iso TxRx Init
23h	22h	21h	20h	Diagnostics
27h	26h	25h	24h	Async Status
2bh	2ah	29h	28h	Phy Chip Access
2fh	2eh	2dh	2ch	Parallel Port
33h	32h	31h	30h	Tx1394Hdr
37h	36h	35h	34h	TxCIPHdr1
3bh	3ah	39h	38h	TxCIPHdr2
3fh	3eh	3dh	3ch	SPH-rsv/AddData1 to 2
43h	42h	41h	40h	AddData3 to 6
47h	46h	45h	44h	AddData7 to 10
4bh	4ah	49h	48h	Rx1394Hdr
4fh	4eh	4dh	4ch	RxCIPHdr1
53h	52h	51h	50h	RxCIPHdr2
67h	66h	65h	64h	IPB Write (first)
6bh	6ah	69h	68h	IPB Write
6fh	6eh	6dh	6ch	IPB Write (confirm)
73h	72h	71h	70h	ATF Write (first)
77h	76h	75h	74h	IPB Write ARF Read
7fh	7eh	7dh	7ch	ATF Write (confirm)

Annex. Corresponding Table for CFR Access Address And Host Interface I/O Data (16-bit Mode)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	I/O data bit
02h																00h																Version
06h																04h																Node Address
0ah																08h																Control
0eh																0ch																Interrupt
12h																10h																Interrupt Mask
16h																14h																Cycle Timer
1ah																18h																DIF Mode
1eh																1ch																Iso TxRx Init
22h																20h																Diagnostics
26h																24h																Async Status
2ah																28h																Phy Chip Access
2eh																2ch																Parallel Port
32h																30h																Tx1394Hdr
36h																34h																TxCIPHdr1
3ah																38h																TxCIPHdr2
3eh																3ch																SPH-rsv/AddData1 to 2
42h																40h																AddData3 to 6
46h																44h																AddData7 to 10
4ah																48h																Rx1394Hdr
4eh																4ch																RxCIPHdr1
52h																50h																RxCIPHdr2
66h																64h																IPB Write (first)
6ah																68h																IPB Write
6eh																6ch																IPB Write (confirm)
72h																70h																ATF Write (first)
76h																74h																IPB Write ARF Read
7eh																7ch																ATF Write (confirm)

Package Outline Unit: mm

100PIN LQFP (PLASTIC)



NOTE: Dimension "∗" does not include mold protrusion.

DETAIL **A**

SONY CODE	LQFP-100P-L01
EIAJ CODE	∗QFP100-P-1414-A
JEDEC CODE	————

PACKAGE STRUCTURE

PACKAGE MATERIAL	EPOXY/PHENOL RESIN
LEAD TREATMENT	SOLDER PLATING
LEAD MATERIAL	42 ALLOY
PACKAGE WEIGHT	————