

To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: <http://www.renesas.com>

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (<http://www.renesas.com>)

Send any inquiries to <http://www.renesas.com/inquiry>.

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: “Standard”, “High Quality”, and “Specific”. The recommended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as “Specific” without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as “Specific” or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is “Standard” unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.
 - “Standard”: Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.
 - “High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.
 - “Specific”: Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

IEBus™ (Inter Equipment Bus™) PROTOCOL CONTROL LSI

DESCRIPTION

The μPD6708 is a peripheral LSI for microcontrollers that controls the protocol of the IEBus.

This LSI processes the protocol of the IEBus. Because it is provided with a transmit/receive buffer, the microcontroller can concentrate on the application processing of the IEBus. Because the μPD6708 also contains an IEBus driver/receiver, it can be directly connected to the bus.

FEATURES

- Protocol control of IEBus
 - Multi-master system
 - Broadcast communication function (communication between one unit and multiple units)
 - Choice of three modes with different transmission speeds
- On-chip IEBus driver/receiver
- Transmit/receive buffer
 - Transmit: 4-byte FIFO
 - Receive: 20-byte FIFO
- Interface with microcontroller
 - Three-line serial I/O (\overline{SCK} , SO, SI pins)
 - Transfer with MSB first
- Oscillation frequency (fx): 12 MHz, 12.58 MHz
 - In modes 0 and 1: $\pm 1.5\%$
 - In mode 2: $\pm 0.5\%$
- Supply voltage: $V_{DD} = 5\text{ V} \pm 10\%$

	At 12 MHz	At 12.58 MHz
Mode 0	Approx. 3.9 Kbps	Approx. 4.1 Kbps
Mode 1	Approx. 17 Kbps	Approx. 18 Kbps
Mode 2	Approx. 26 Kbps	Approx. 27 Kbps

ORDERING INFORMATION

Part Number	Package
μPD6708CX	16-pin plastic DIP (300 mil)
μPD6708GS	16-pin plastic SOP (300 mil)

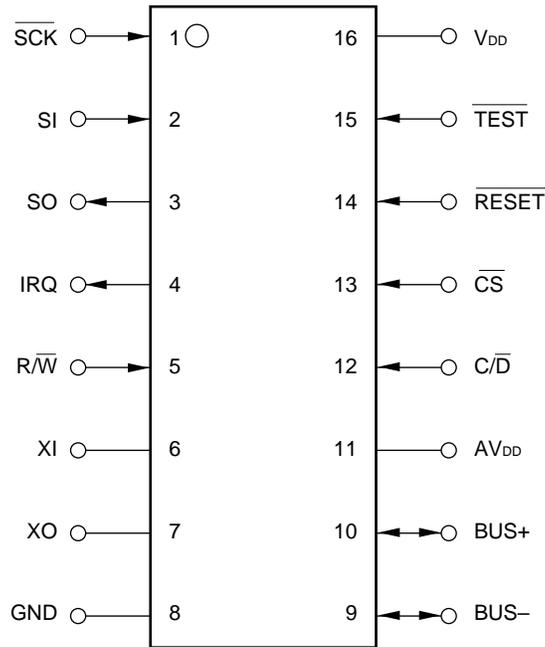
APPLICATION FIELD

Fields where a small-scale digital data transfer system is required between equipment, such as automobile electronic systems and industrial equipment

The information in this document is subject to change without notice.

PIN CONFIGURATION (TOP VIEW)

- 16 pin plastic DIP (300 mil)
μPD6708CX
- 16 pin plastic SOP (300 mil)
μPD6708GS



- $\overline{\text{SCK}}$: Serial clock input
- SI : Serial data input
- SO : Serial data output
- IRQ : Interrupt request output
- $\text{R}/\overline{\text{W}}$: Read/write switchover input
- XI, XO : System clock
- GND : Ground
- BUS-, BUS+ : IEBus input/output
- AV_{DD} : IEBus analog power supply (connected to V_{DD} pin)
- $\text{C}/\overline{\text{D}}$: Command/data switchover input
- $\overline{\text{CS}}$: Chip select input
- $\overline{\text{RESET}}$: Reset input
- $\overline{\text{TEST}}$: Test input (connected to V_{DD} pin)
- V_{DD} : Positive power supply

CONTENTS

1. PIN FUNCTIONS 5

 1.1 List of Pin Functions 5

2. IEBus OPERATION 6

 2.1 Operation Overview 6

 2.2 IEBus Communication Protocol 7

 2.2.1 Bus mastership determination (arbitration) 8

 2.2.2 Communication modes 8

 2.2.3 Communication address 9

 2.2.4 Broadcast communication 9

 2.3 Transfer Protocol 10

 2.4 Transfer Data (Contents of Data Field) 16

 2.5 Bit Format 19

3. INTERNAL CONFIGURATION 20

 3.1 Data Link Layer Controller 21

 3.2 Physical Layer Controller 21

 3.3 IEBus Driver/Receiver 21

 3.4 Host Interface 21

4. INTERFACING WITH HOST CONTROLLER 22

 4.1 Accessible Buffers and Registers from Host Controller 22

 4.1.1 Write data buffer (WDB) 22

 4.1.2 Read data buffer (RDB) 22

 4.1.3 Command register (CMR) 22

 4.1.4 Status register (STR) 23

 4.2 Host Interface Modes 23

 4.2.1 Switching through pin control 24

 4.2.2 Switching through software control 26

 4.3 Reset Mode 28

5. COMMUNICATION CONTROL COMMANDS 30

 5.1 Overview of Communication Control Commands 30

 5.2 Communication Control Command Functions 31

 5.2.1 INIT command (command code: 0000) 31

 5.2.2 SETSA command (command code: 0001) 32

 5.2.3 MREQ1 command (command code: 0010) 33

 5.2.4 MREQ2 command (command code: 0011) 34

 5.2.5 ABORT command (command code: 0100) 34

 5.2.6 SETSD command (command code: 0101) 35

 5.2.7 GETSTA command (command code: 0110) 36

 5.2.8 SETREV command (command code: 0111) 37

6. RETURN CODES 38

 6.1 Return Codes in Master/Slave Data Transmission 38

 6.2 Return Codes in Master Reception 38

 6.3 Return Codes in Slave Reception 39

 6.4 Return Codes in Broadcast Reception 39

 6.5 Return Codes Generation Intervals 40

7. COMMUNICATING WITH HOST CONTROLLER 43

 7.1 Master Transmission 43

 7.1.1 Master transmission by MREQ1 command 43

 7.1.2 Master transmission by MREQ2 command 44

7.2 Slave Transmission 44

 7.2.1 Data transmission 44

 7.2.2 Transmitting slave status address and lock address 45

7.3 Master Reception 45

7.4 Slave Reception 46

7.5 Broadcast Reception 47

8. EXAMPLE OF HOST CONTROLLER PROCESSING FLOW 48

 8.1 Main Routine 48

 8.2 Interrupt Service Routine 49

 8.3 Processing Routine 50

 8.3.1 μPD6708 initialization routine 50

 8.3.2 Communication control command processing routine 51

 8.3.3 Master transmission processing routine 57

 8.3.4 Slave data transmission processing routine 58

 8.3.5 Master reception processing routine 59

9. ELECTRICAL SPECIFICATIONS 63

10. PACKAGE DRAWINGS 67

11. RECOMMENDED SOLDERING CONDITIONS 69

APPENDIX MAJOR DIFFERENCES BETWEEN μPD6708 AND μPD72042A, μPD72042B 70

1. PIN FUNCTIONS

1.1 List of Pin Functions

Pin No.	Pin Name	Input/Output	Function	I/O Format	At Reset
1	$\overline{\text{SCK}}$	Input	Input for serial clock used to interface with microcontroller.	CMOS input	Input
2	SI	Input	Input for serial data used to interface with microcontroller.	CMOS input	Input
3	SO	Output	Output for serial data used to interface with microcontroller.	CMOS output	High level
4	IRQ	Output	Output used by interrupt request signals generated by communication and command execution results. Used as operation start request signal to microcontroller. The interrupt request signal is output for 8 μ s or longer at high level.	CMOS output	Low level
5	$\overline{\text{R/W}}$	Input	Input for switching serial interface read/write mode. When high, it is in the read mode. When low, it is in the write mode. When this pin is low and $\overline{\text{C/D}}$ pin high, the read and write modes can be switched by commands input from the serial interface.	CMOS input	Input
6 7	XI XO	—	Connection pins for system clock resonator. Use a 12- or 12.58-MHz crystal, or ceramic resonator. Frequency precision depends on the communication mode used. Mode 0 : ± 1.5 % Mode 1 : ± 1.5 % Mode 2 : ± 0.5 %	—	(Oscillation continues)
8	GND	—	Ground	—	—
9 10	BUS– BUS+	Input/output	Input/output for IEBus.	—	High impedance
11	AV_{DD}	—	IEBus driver/receiver analog power supply. Connect to V_{DD} .	—	—
12	$\overline{\text{C/D}}$	Input	Input used to switch between processing data input to the serial interface as commands or data. When set to high, data is processed as commands; when low, data is processed as data. When this pin is high and $\overline{\text{R/W}}$ pin low, the read and write modes can be switched by commands input from the serial interface.	CMOS input	Input
13	$\overline{\text{CS}}$	Input	Chip select input. When low, serial interface input is enabled. When high, serial clock ($\overline{\text{SCK}}$) input is disabled, SO pin becomes high impedance, and the serial clock counter is reset. The status of $\overline{\text{CS}}$ pin is not affected by IEBus transmit and receive operations.	CMOS input	Input
14	$\overline{\text{RESET}}$	Input	System reset signal input pin. Low input effects a reset. Always input the low signal for 6 μ s or longer after turning on the power.	CMOS input	Input
15	$\overline{\text{TEST}}$	Input	Always connect this pin to the V_{DD} .	CMOS input	—
16	V_{DD}	—	Positive power supply input. Apply a voltage of 5 V ± 10 %.	—	—

2. IEBus OPERATION

2.1 Operation Overview

- ★ The μ PD6708 is an IEBus interface CMOS LSI device.
The IEBus is a bus for a small-scale digital data transfer system designed to transfer data between electronic devices.
The μ PD6708 is connected to a microcontroller incorporated in electronic equipment with a serial interface ($\overline{\text{SCK}}$, SO, SI pins). The data and commands required to transfer data with the host controller (microcontroller) are set via this serial interface.
When the host controller transmits data to the μ PD6708 via the serial interface, signals are output from the BUS pins (BUS+ and BUS-). Data received from the BUS pins can be read by the host controller via the serial interface.

2.2 IEBus Communication Protocol



An overview of the IEBus is as follows.

- Communication system: Half-duplex asynchronous communication
- Multi-master system
All the units connected to the IEBus can transfer data to the other units.
- Broadcast communication function (communication between one unit and multiple units)
Group broadcast communication: Broadcast communication with group units
General broadcast communication: Broadcast communication with all units.
- Three modes with different transfer speeds selectable.

	$f_x = 12 \text{ MHz}$	$f_x = 12.58 \text{ MHz}$	Maximum Number of Transfer Bytes (bytes/frame)
Mode 0	Approx. 3.9 Kbps	Approx. 4.1 Kbps	16
Mode 1	Approx. 17 Kbps	Approx. 18 Kbps	32
Mode 2	Approx. 26 Kbps	Approx. 27 Kbps	128

- Access control: CSMA/CD (Carrier Sense Multiple Access with Collision Detection)
The priority order for bus occupancy is as follows.

<1> Broadcast communication takes precedence over ordinary communication (i. e., communication between one unit and another).

<2> The lowest master address has the highest priority.

- Communication scale
Number of units: MAX. 50
Cable length: MAX. 150 m (with twisted-pair cable <Resistance: 0.1 Ω/m or less>)
Load capacity: MAX. 8000 pF <between BUS- and BUS+>, $f_x = 12 \text{ MHz}$
MAX. 7100 pF <between BUS- and BUS+>, $f_x = 12.58 \text{ MHz}$
Terminating resistor: 120 Ω

2.2.1 Bus mastership determination (arbitration)

★ When a unit connected to the IEBus controls another unit, it performs an operation to occupy the bus. This operation is called arbitration.

Arbitration is to select one unit, and if several units begin to transmit data simultaneously, gives permission to occupy the bus to that one unit.

So that one unit is granted the permission to occupy the bus as a result of the arbitration, the following priority conditions are determined.

Remark The units not given permission through arbitration are automatically allowed to get into retransfer mode (number of retransfer times for the μPD6708: 3).

(1) Priority according to type of communication

Broadcast communication (between a single and multiple units) takes precedence over ordinary communication (between single units).

(2) Priority according to master address

If the communication devices are of the same type, the unit with the lowest master address has the highest priority.

Example The master address comprises 12 bits, and unit 000H has the highest priority while unit FFFH has the lowest priority.

2.2.2 Communication modes

The IEBus is provided with three communication modes with different transfer speeds. The transfer speed and maximum number of transfer bytes in a single communication frame in each communication mode are shown in Table 2-1.

Table 2-1. Transfer Speed and Maximum Number of Transfer Bytes in Each Communication Mode

Communication Mode	Maximum Number of Transfer Bytes (bytes/frame)	Actual Transfer Speed ^{Note 1} (Kbps)	
		f _x = 12 MHz ^{Note 2}	f _x = 12.58 MHz ^{Note 2}
0	16	Approx. 3.9	Approx. 4.1
1	32	Approx. 17	Approx. 18
2	128	Approx. 26	Approx. 27

- Notes**
1. Actual transfer speed when the maximum number of bytes is transferred
 2. Oscillation frequency when the μPD6708 is used

- Cautions**
1. A communication mode is selected for each unit connected to the IEBus before communication is performed. If the communication mode of the master unit is not the same as that of the unit with which the master unit is to communicate (slave unit), communication cannot be performed correctly.
 2. If the oscillation frequency of one unit is f_x = 12 MHz and that of the other unit is f_x = 12.58 MHz, communication cannot be performed correctly even if the communication mode is the same. Make sure that the oscillation frequencies of the two units to communicate are the same.

2.2.3 Communication address

With the IEBus, a 12-bit communication address is assigned to each unit. The communication address is made up as follows. ★

Higher 4 bits: Group number (number which identifies the group to which the unit belongs)

Lower 4 bits: Unit number (number which identifies a unit within a group)

2.2.4 Broadcast communication

In ordinary communication, there is only one master unit and one slave unit, and transmission or reception is performed on an one-to-one basis. In broadcast communication, however, there are a number of slave units and the master unit performs transmission with these slave units. Because there are several slave units, no acknowledge signals is returned from the slave units during communication.

Whether broadcast communication or ordinary communication is performed is specified by the broadcast bit (for the broadcast bit, see 2.3 (1) <2> “Broadcast bit”).

There are two kinds of broadcast communication, as follows.

(1) Group broadcast communication

Broadcast communication is performed to the units in a group whose group numbers are the same as that specified by the higher 4 bits of the communication address.

(2) General broadcast communication

Broadcast communication is performed to all units irrespective of their group numbers.

Group broadcast communication or general broadcast communication is identified by the value of a slave address (for the slave address, see 2.3 (3) “Slave address field”).

2.3 Transfer Protocol

The IEBus transfer signal format is shown in Figure 2-1.

Data is transferred as a series of signals called a communication frame. The number of data that can be transferred in one communication frame and the transfer speed differ depending on the communication mode.

Figure 2-1. Transfer Signal Format

(f_x = at 12 MHz)

Field Name	Header		Master Address Field		Slave Address Field			Control Field			Message Length Field			Data Field					
Number of Bits	1	1	12	1	12	1	1	4	1	1	8	1	1	8	1	1	8	1	1
	Start Bit	Broad-cast Bit	Master Address	P	Slave Address	P	A	Control Bit	P	A	Message Length Bit	P	A	Data Bit	P	A	Data Bit	P	A
Transfer Time																			
Mode 0	Approx. 7330 μs												Approx. 1590 × N μs						
Mode 1	Approx. 2090 μs												Approx. 410 × N μs						
Mode 2	Approx. 1590 μs												Approx. 300 × N μs						

- P: Parity bit (1 bit)
- A: Acknowledge bit (1 bit)
When A = 0: ACK
When A = 1: NAK
- N: Number of data bytes

Remark In broadcast communication, the value of the acknowledge bit is ignored.

(1) Header

A header comprises a start bit and a broadcast bit, as described below.

<1> Start bit

The start bit is a signal which tells the other units that data transmission will start.

The unit which is about to start transmitting data will output the low signal (the start bit) for a specified time, and then outputs the broadcast bit.

If another unit is already outputting a start bit before one unit outputs a start bit, the unit will not output the start bit. It will wait until the another unit completely outputs the start bit, and then outputs the broadcast bit.

The units other than the one that has started transmission detect this start bit and enters the reception state.

<2> Broadcast bit

The broadcast bit distinguishes between broadcast communication and ordinary communication.

When this bit is '0', it indicates broadcast communication; when it is '1', it indicates ordinary communication. There are two types of broadcast communication: group broadcast and general broadcast. These types are identified by the value of the slave address (for the slave address, see **(3) "Slave address field"**).

In broadcast communication, there are a number of slave units. Therefore, the acknowledge bit is not returned in the fields described in (2) below and onward.

If two or more units start to transmit a communication frame simultaneously, broadcast communication takes precedence over ordinary communication, and wins in the arbitration.

(2) Master address field

The master address field is used to transmit the unit address of the master unit (master address) to the other units. ★

The master address field consists of master address bits and a parity bit.

The master address comprises 12 bits and is output from the MSB.

If two or more units start transmitting the broadcast bit of the same value simultaneously, the arbitration decision is made by the master address field.

The master address field compares the data the master has output with the data on the bus each time the master transmits 1 bit of data. If the master address output by the master unit is different from the data on the bus, the master unit assumes that it has lost in arbitration, stops transmission, and enters the reception state.

Because the IEBus has a wired-AND configuration, the unit having the lowest master address of the units participating in the arbitration (arbitration masters) wins in the arbitration. Ultimately, only one unit remains in the transmission state as the master unit after outputting a 12-bit master address.

This master unit then outputs a parity bit ^{Note}, makes the other units confirm the master address, and then outputs the slave address field.

Note Even parity is used. When the number of the bits that are '1' in the master address is odd, the parity bit is '1'.

(3) Slave address field

The slave address field is used to transmit the address (slave address) of a unit (slave unit) with which the master wishes to communicate.

The slave address field consists of slave address bits, a parity bit, and an acknowledge bit.

The slave address comprises 12 bits and is output from the MSB. After the 12-bit slave address is transmitted, the parity bit is output to prevent the slave address from being received incorrectly. Next, the master unit looks for the acknowledge signal (bit) from the slave unit to confirm that the slave unit exists on the bus. When the master unit detects the acknowledge signal, it starts outputting the control field. In the case of broadcast communication, however, the master unit outputs the control field without waiting for the acknowledge bit.

A slave unit outputs the acknowledge signal if it has detected that its slave address coincides with that selected by the master and that the parities of both the master and slave addresses are even. If the parity is odd, the slave unit assumes that the master or slave address has not been correctly received, and does not output the acknowledge signal. In this case, the master unit enters the standby (monitor) state and communication ceases.

In the case of broadcast communication, the slave address is used to distinguish between group broadcast and general broadcast as follows:

Slave address = FFFH: General broadcast communication

Slave address ≠ FFFH: Group broadcast communication

Remark In the case of group broadcast communication, the group number is the value of higher 4 bits of the slave address.

(4) Control field

★ The control field indicates the type of data and the transfer direction of the subsequent data field.

The control field consists of 4 control bits, a parity bit, and an acknowledge bit.

The control bits are output from the MSB.

A parity bit is output after the control bits. When the parity is even and the slave can execute the function requested by the master unit, the slave unit outputs an acknowledge signal, and then outputs the next message length field. If the slave unit cannot execute the function requested by the master unit even if the parity is even, or if the parity is odd, the slave unit does not output the acknowledge signal but returns to the standby (monitor) state.

After the master unit has confirmed the acknowledge signal, it starts outputting the next message length field.

If the master unit is cannot confirm the acknowledge signal, it enters the standby state and stops communication. In the case of broadcast communication, however, the master unit starts outputting the message length field without confirming the acknowledge signal.

For the functions of the control bits, see Table 2-3.

(5) Message length field

The message length field is used to specify the number of communication data bytes.

The message length field comprises 8 message length bits, a parity bit and, an acknowledge bit.

The message length bits are output from the MSB. The message length bits indicate the number of communication data bytes as shown in Table 2-2.

Table 2-2. Meaning of Message Length Bits

Message Length Bits (hex)	Number of Transmission Data Bytes
01H	1 byte
02H	2 bytes
:	:
:	:
FFH	255 bytes
00H	256 bytes

Remark In the communication mode, if the number of bytes exceeding the maximum number of transfer bytes per frame is set, two or more frames are communicated. In this case, the message length bits indicate the number of remaining communication data bytes during the second communication and onward.

The operation of this field differs depending on whether the master transmits (bit 3 of control bits is 1) or receives (bit 3 of control bits is 0) data.

<1> When master transmits data

The message length bits and parity bit are output by the master unit. The slave unit outputs the acknowledge signal and then the next data field if it detects that the parity is even. The slave unit does not output the acknowledge signal in the case of broadcast communication.

If the parity is odd, the slave unit assumes that the message length bits have not been received correctly, and returns to the standby (monitor) state without outputting the acknowledge signal. In this case, the master unit also returns to the standby state, and communication ceases.

<2> When master receives data

The message length bits and parity bit are output by the slave unit. The master unit outputs the acknowledge signal if it detects that the parity bit is even. ★

If the parity is odd, the master unit assumes that the message length bits have not been received correctly, and returns to the standby state without outputting the acknowledge signal. In this case, the slave unit also returns to the standby state, and communication ceases.

(6) Data field

The data field is used to transmit/receive data to/from the slave units.

The master unit uses the data field to transmit data to and receive data from the slave units.

The data field consists of 8 data bits, a parity bit, and an acknowledge bit.

The data bits are output from the MSB.

Following the data bits, the parity bit and acknowledge bit are output from the master unit and the slave unit, respectively.

Broadcast communication is performed when only the master unit transmits data. At this time, the acknowledge signal is ignored.

The operation differs depending on whether the master performs transmission or reception, as follows.

<1> When master transmits data

When the master unit writes data to the slave unit, the master unit transmits data bits and a parity bit to the slave unit. The slave unit receives the data bits and parity bit. If the parity is even and the receive buffer is empty, the slave unit outputs the acknowledge signal. If the parity is odd and the receive buffer is not empty, the slave unit denies acknowledgment of the corresponding data and does not output the acknowledge signal.

If no acknowledge signal is output from the slave unit, the master unit transmits the same data again. The master unit continues this operation until it detects the acknowledge signal from the slave unit or the data reaches the maximum number of transfer bytes.

If the parity is even and the acknowledge signal has been output from the slave unit, and if the master unit has more data to transmit and the maximum number of transfer bytes is not exceeded, the master unit will transmit the next data.

In the case of broadcast communication, the slave unit does not output the acknowledge signal, and the master unit transfers data on a byte-by-byte basis.

<2> When master receives data

When the master unit reads data from the slave unit, the master unit outputs synchronization signals corresponding to all the read bits.

The slave unit outputs the contents of the data and parity bits onto the bus in accordance with the synchronization signals from the master unit.

The master unit reads the data and parity bit output by the slave unit, and checks the parity.

If the parity is odd or the receive buffer is not empty, the master unit denies acknowledgement of that data and does not output the acknowledge signal. If the data is within the maximum number of transfer bytes that can be transmitted in one frame, the master unit repeatedly reads the same data.

If the parity is even and the receive buffer is empty, the master unit acknowledges the data and transmits back the acknowledge signal. If the data is within the maximum number of bytes that can be transmitted in one frame, the master unit reads the next data. ★

(7) Parity bits

Parity bits are used to check that there is no error in the transfer data.

A parity bit is added to the master address bits, slave address bits, control bits, message length bits, and data bits.

Even parity is used. If the number of the bits that are '1' bits in data is odd, the parity bit is '1', and if the number of the bits that are '1' bits is even, the parity bit is '0'.

(8) Acknowledge bits

In ordinary communication (between two units), an acknowledge bit is added to the following places to confirm that data has been acknowledged correctly.

- At the end of the slave address field.
- At the end of the control field.
- At the end of the message length field.
- At the end of a data field.

The definition of the acknowledge bit is as follows.

- '0': Indicates that transfer data has been acknowledged (ACK).
- '1': Indicates that transfer data has not been acknowledged (NAK).

Note that the value of the acknowledge bit is ignored in broadcast communication.

<1> Acknowledge bit at the end of the slave field

When any of the following conditions is met, the acknowledge bit at the end of the slave field is NAK, and communication is discontinued.

- If the parity of the master address bits or slave address bits is incorrect.
- If a timing error (error in bit format) occurs.
- If the slave unit does not exist.

<2> Acknowledge bit at the end of the control field

When any of the following conditions is met, the acknowledge bit at the end of the control field is NAK, and communication is discontinued.

- If the parity of the control bits is incorrect.
- ★ If bit 3 of the control bits is '1' (write operation) when the slave receive buffer ^{Note} is not empty.
- If the control bits indicate read operation (3H or 7H) when the slave transmit buffer ^{Note} is empty.
- If 3H, 6H, 7H, AH, BH, EH, or FH of control bits is requested from a unit other than the unit which set the lock when a lock has been set.
- If the control bits indicate lock address read (4H) when a lock has not been set.
- If a timing error occurs.
- If the control bits are undefined.

Note See 2.4 (1) "Reading slave status (SSR) (control bit: 0H, 6H)".

<3> Acknowledge bit at the end of a message length field

When either of the following conditions is met, the acknowledge bit at the end of the message length field is NAK, and communication is discontinued.

- If the parity of the message length bits is incorrect.
- If a timing error occurs.

<4> Acknowledge bit at the end of a data field

When any of the following conditions is met, the acknowledge bit at the end of a data field is NAK, and communication is discontinued.

- If the parity of the data bits is incorrect ^{Note}.
- If a timing error occurred in or after the previous acknowledge bit transmission.
- If the receive buffer is full and cannot accept any more data ^{Note}.

Note In this case, if the number of transfer bytes is within the maximum number of bytes which can be transmitted, the transmitting side re-executes transmission of that data field.

2.4 Transfer Data (Contents of Data Field)

The contents of the data field are data specified by the control bits.

Table 2-3. Functions of Control Bits

★

	Bit 3 ^{Note 1}	Bit 2	Bit 1	Bit 0	Function ^{Note 2}
0H	0	0	0	0	Reads slave status (SSR)
1H	0	0	0	1	Undefined
2H	0	0	1	0	Undefined
3H	0	0	1	1	Reads and locks data
4H	0	1	0	0	Reads lock address (lower 8 bits)
5H	0	1	0	1	Reads lock address (higher 4 bits)
6H	0	1	1	0	Reads and unlocks slave status (SSR)
7H	0	1	1	1	Reads data
8H	1	0	0	0	Undefined
9H	1	0	0	1	Undefined
AH	1	0	1	0	Writes and locks command
BH	1	0	1	1	Writes and locks data
CH	1	1	0	0	Undefined
DH	1	1	0	1	Undefined
EH	1	1	1	0	Writes command
FH	1	1	1	1	Writes data

Notes 1. Depending on the value of bit 3 (MSB), the transfer direction of the message length bits of the subsequent message field and data field differs.

When bit 3 is “1”, data are transferred from the master unit to the slave unit.

When bit 3 is “0”, data are transferred from the slave unit to the master unit.

2. 3H, 6H, AH, and BH are control bits that specify locking or unlocking.

If any of undefined values 1H, 2H, 8H, 9H, CH, or DH is transmitted, no acknowledge bit is returned.

A unit locked by the master unit rejects acknowledging the control bits and does not output the acknowledge bit if the control bits received from the master unit which requested locking is in any other state than that shown in Table 2-4.

Table 2-4. Control Field Corresponding to Locked Slave Unit

	Bit 3	Bit 2	Bit 1	Bit 0	Function
0H	0	0	0	0	Reads slave status
4H	0	1	0	0	Reads lock address (lower 8 bits)
5H	0	1	0	1	Reads lock address (higher 4 bits)

(1) Reading slave status (SSR) (control bit: 0H, 6H)

The master unit can learn the reason why the slave unit has not returned the acknowledge bit (ACK) by reading the slave status.

The slave status is determined by the results of the last communication performed by the slave unit.

All the slave units can provide slave status information.

The meanings of the slave status are shown in Table 2-5.

Figure 2-2. Bit Configuration of Slave Status (SSR)

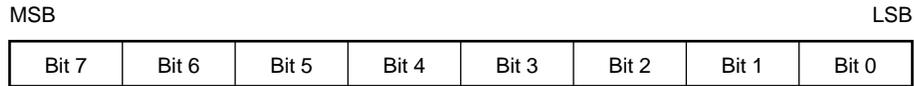


Table 2-5. Meanings of Slave Status

Bit	Value	Meaning	
Bit 0 <small>Note 1</small>	0	Slave transmit buffer empty	
	1	Slave transmit buffer is not empty.	
Bit 1 <small>Note 2</small>	0	Slave receive buffer empty	
	1	Slave receive buffer is not empty.	
Bit 2	0	Unit is not locked.	
	1	Unit is locked.	
Bit 3	0	Fixed to '0'	
Bit 4 <small>Note 3</small>	0	Slave transmission ends	
	1	Slave transmission enabled	
Bit 5	0	Fixed to '0'	
Bit 7	00	Mode 0	Indicates the highest mode the unit supports <small>Note 4</small> .
Bit 6	01	Mode 1	
	10	Mode 2	
	11	For future expansion	

- Notes**
1. The slave transmit buffer is the buffer accessed during data read processing (control bits: 3H, 7H).
With the μPD6708, this buffer corresponds to the write data buffer (WDB) when the SETSD command is valid (see 5.2.6 “SETSD command”).
 2. The slave receive buffer is the buffer accessed during data write processing (control bits: 8H, AH, BH, EH, FH).
With the μPD6708, this buffer corresponds to the read data buffer (RDB).
 3. The value of bit 4 can be selected by INIT command (see 5.2.1 “INIT command”).
 4. Because the μPD6708 can support mode 2, bits 7 and 6 are fixed at '10'.

(2) Transferring data command (control bit: read (3H, 7H), write (AH, BH, EH, FH))

During data read (3H, 7H), the data in the data buffer of the slave unit are read to the master unit.

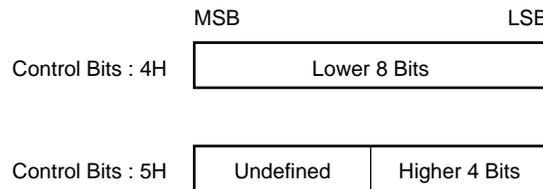
During data write (BH, FH) or during command write (AH, EH), the data the slave unit has received are processed according to the operation convention.

- Remarks**
1. The user can voluntarily select data and command as his system requires.
 2. Control bits 3H, AH, and BH may be locked depending on the communication condition and status.

(3) Reading lock address (control bits: 4H, 5H)

★ When the lock address is read processing (4H, 5H), the address (12 bits) of the master unit that has issued the lock instruction is read in 1-byte units, as shown below.

Figure 2-3. Lock Address Configuration



(4) Locking and unlocking (locking (3H, AH, BH), unlocking (6H))

The lock function is used to transfer a message over two or more frames.

A locked unit receives data only from the unit that has locked the unit.

Locking and unlocking are performed as described below.

<1> Locking

After the transmission/reception of the acknowledge bit '0' of the message length field by the control bits (3H, AH, BH) which specify the lock has ended, if the communication frame is completed without completing the transmission or reception of the number of data bytes specified by the message length bits, the slave unit is locked by the master unit. At this time, the bit (bit 2) relating to the locking of the byte which indicates the slave status is set to '1'.

<2> Unlocking

After completion of transmission or reception of data in one frame by the number of data bytes specified by the message length bits with control bits (3H, AH, or BH) specifying locking or control bits (6H) specifying unlocking, the slave unit is unlocked by the master unit. At this time, the bit (bit 2) relating to the locking of the byte which indicates the slave status is reset to '0'.

Locking and unlocking are not performed in the case of broadcast communication.

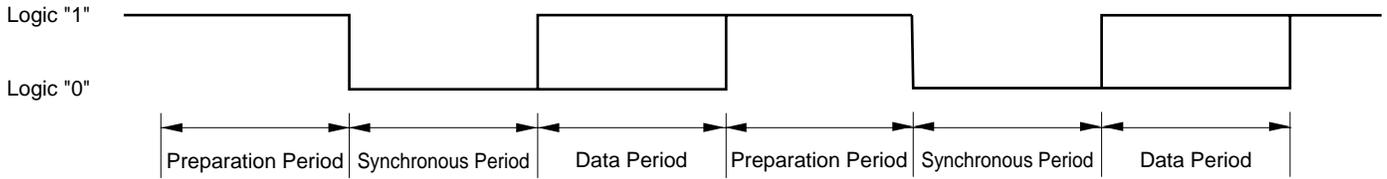
Caution To unlock the unit specified to be unlocked by the unit itself, the INIT command (see 5.2.1 "INIT command") must be executed with the μPD6708 (Whether a unit is locked or not can be checked by using the GETSA command (see 5.2.7 "GETSA command").

2.5 Bit Format

The IEBus communication frame bit format (concept) is shown in Figure 2-4.

★

Figure 2-4. IEBus Bit Format (Concept)



Logic "1": Potential difference between bus lines (BUS+ pin and BUS- pin) is 20 mV or lower (low level).
 Logic "0": Potential difference between bus lines (BUS+ pin and BUS- pin) is 120 mV or higher (high level).

- Preparation period: The first or subsequent low-level (logic "1") period
- Synchronous period: The next high-level (logic "0") period
- Data period: The period that expresses the bit value (logic "1": low level; logic "0": high level)

The synchronous period and data period have approximately the same length.

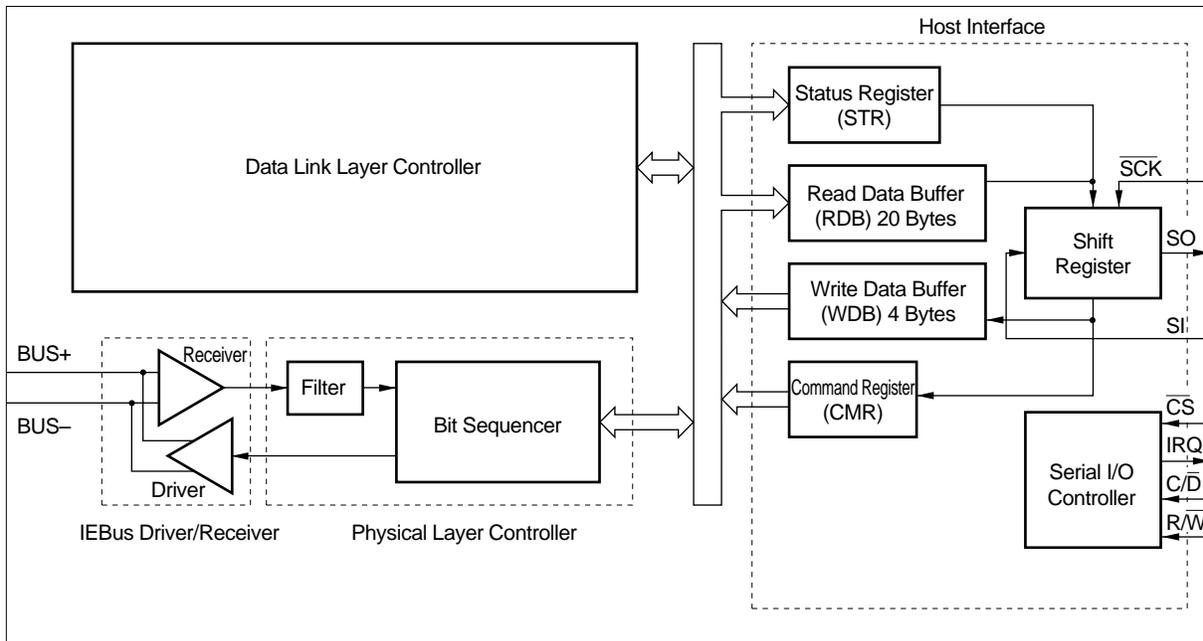
The IEBus uses bit-by-bit synchronization. The specifications for the total bit time and the periods allocated to the bits depend on the type of transfer bit, and on whether the unit is the master unit or the slave unit.

3. INTERNAL CONFIGURATION

The μPD6708 is composed of the following four blocks.

- (1) Data link layer controller
- (2) Physical layer controller
- (3) IEBus driver/receiver
- (4) Host interface

Figure 3-1. μPD6708 Internal Blocks



3.1 Data Link Layer Controller

The data link layer controller performs processing of the IEBus protocol data link layer (frame composition and resolution, communication error detection, etc.), execution of communication control commands set by the host controller, and generate a return code that informs the host controller of the communication status.

3.2 Physical Layer Controller

The physical controller performs generation and resolution of bit timing and also converts the signals between the bus lines through the driver/receiver.

3.3 IEBus Driver/Receiver

The driver/receiver performs conversion between the logic signals within the μPD6708 and the IEBus signals. The IEBus signals and their relationship to the logic statuses are shown in Table 3-1.

Table 3-1. Relationship between IEBus Signals and Logical Statuses

Logical Status	IEBus Signals
0	$(\text{BUS+}) - (\text{BUS-}) \geq 120\text{mV}$
1	$(\text{BUS+}) - (\text{BUS-}) \leq 20\text{mV}$

3.4 Host Interface

The host Interface is a block which controls the transmission and reception of data to and from the host controller. It accepts communication control commands, passes on return codes, and forwards transmit data.

The forwarding of transmit data takes place through the FIFO buffers, 4 bytes of write data buffer (WDB) and 20 bytes of read data buffer (RDB). It also absorbs the differences between IEBus transmission speed and the transmission speed on the serial interface between the μPD6708 and the host controller.

4. INTERFACING WITH HOST CONTROLLER

This chapter will explain the interfacing that occurs between the μPD6708 and the host controller.

4.1 Accessible Buffers and Registers from Host Controller

The host controller, which controls the μPD6708, can access the write data (WDB), the read data buffer (RDB), the command register (CMR), and the status register (STR) within the μPD6708.

4.1.1 Write data buffer (WDB)

WDB is a 4-byte FIFO buffer in which the host controller transmit data and the parameters of the communication control commands are written.

4.1.2 Read data buffer (RDB)

RDB is a 20-byte FIFO buffer which stores the receive data acknowledged by the data link layer controller in the μPD6708. The host controller reads the μPD6708 receive data from RDB.

4.1.3 Command register (CMR)

CMR is an 8-bit register used to write control commands for the μPD6708.

As shown in Table 4-1, the host controller sets the reset mode and the host interface mode in higher 4 bits and sets the communication control command code in lower 4 bits.

Table 4-1. Contents of Command Register

Bit	Value	Meaning	
Bit 7	1	Entering the reset mode	
	0	Exiting the reset mode	
Bit 6	1	Data of lower 4 bits of CMR is valid.	
	0	Data of lower 4 bits of CMR is not valid.	
Bit 5	00	Change of mode through pin control	Switches the host interface mode
Bit 4	01	Data write mode	
	10	Data read mode	
	11	Status read mode	
Bit 3 to Bit 0		Set the communication control command codes	

4.1.4 Status register (STR)

STR is an 8-bit register used to determine the status of the μPD6708.

The statuses of WDB and RDB and the status of interrupts can be read from higher 4 bits. The return code, which indicates the result of the communication, can be read from lower 4 bits.

Table 4-2. Contents of Status Register

Bit	Value	Meaning	Description
Bit 7	1	WDB is full	Indicates whether data can be written to WDB
	0	WDB is not full	
Bit 6	1	RDB is empty	Indicates whether data can be read from RDB
	0	RDB is not empty	
Bit 5	1	WDB is empty	Indicates whether data is in WDB
	0	WDB is not empty	
Bit 4	1	Interrupt requested	Indicates whether interrupt servicing is being requested (Bit 4 of the status register is reset by STR by the host controller)
	0	Interrupt not requested	
Bit 3 to Bit 0		Return code	Return code will be read

4.2 Host Interface Modes

The host controller can access WDB, RDB, CMR, and STR within the μPD6708 via the serial interface (\overline{SCK} , SI, SO). There are four modes for accessing the serial interface, as shown in the Table 4-3.

There are two method for switching among these four host interface modes: by using $\overline{C/D}$ pin and $\overline{R/W}$ pin, and by writing data to CMR (software control).

Table 4-3. Host Interface Mode

Mode	Operation
Data write mode	Data input to SI pin is written to WDB from MSB at the rising edge of the serial clock input to \overline{SCK} pin. Data setting is completed at the eighth serial clock cycle.
Data read mode	RDB data is output from MSB to SO pin at the falling edge of the serial clock input to \overline{SCK} pin. A data read is completed by inputting eight serial clock cycles. Data at SI pin is ignored.
Command write mode	Data input to SI pin is written to CMR from MSB at rising edge of the serial clock input to \overline{SCK} pin. Data setting is completed at the eight serial clock cycle.
Status read mode	STR data is output from MSB to SO pin at the falling edge of the serial clock input to \overline{SCK} pin. A data read is completed by inputting eight serial clock cycles. Data at SI pin is ignored.

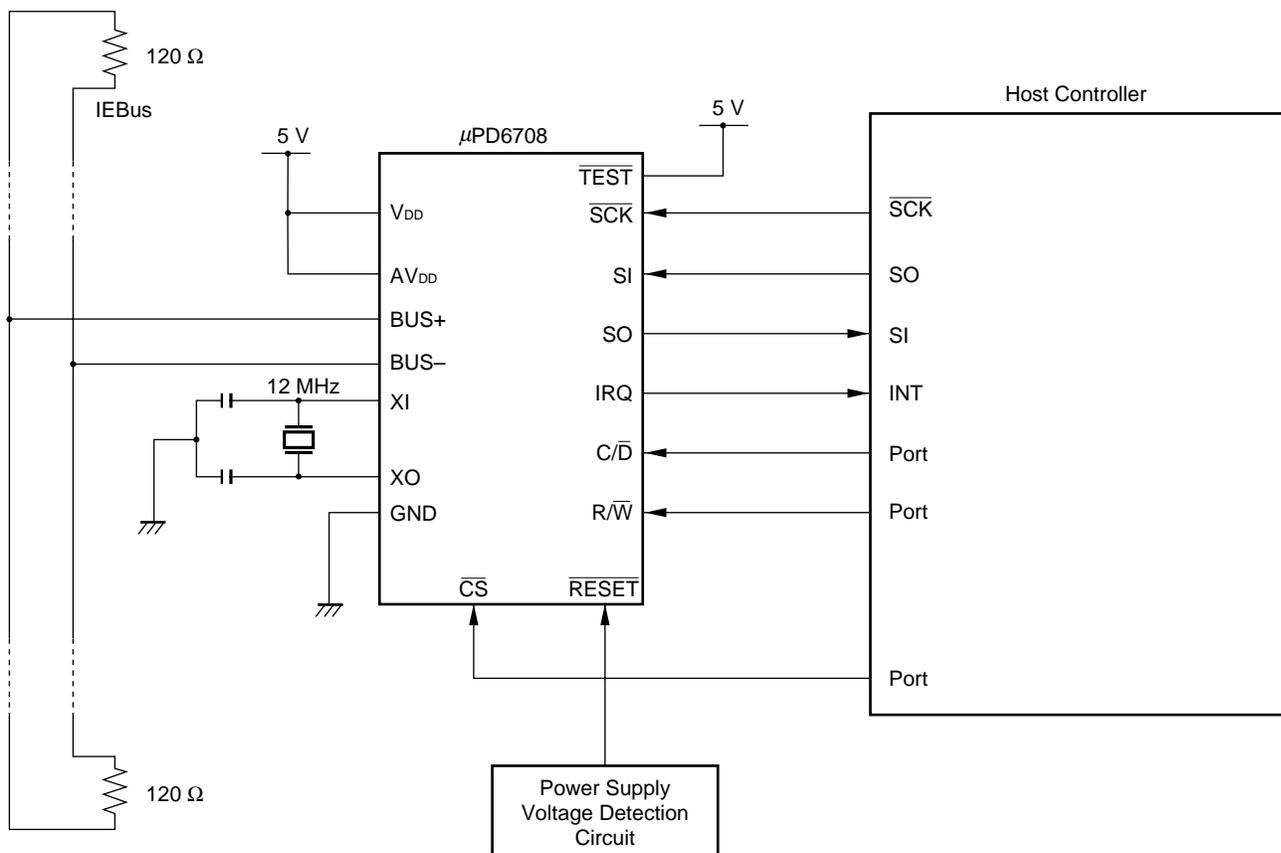
4.2.1 Switching through pin control

With bits 5 and 4 of CMR both '0', the host interface mode can be switched by setting the C/D pin and R/W pin to the values shown in Table 4-4.

Table 4-4. Switching Host Interface Mode by Pin Control

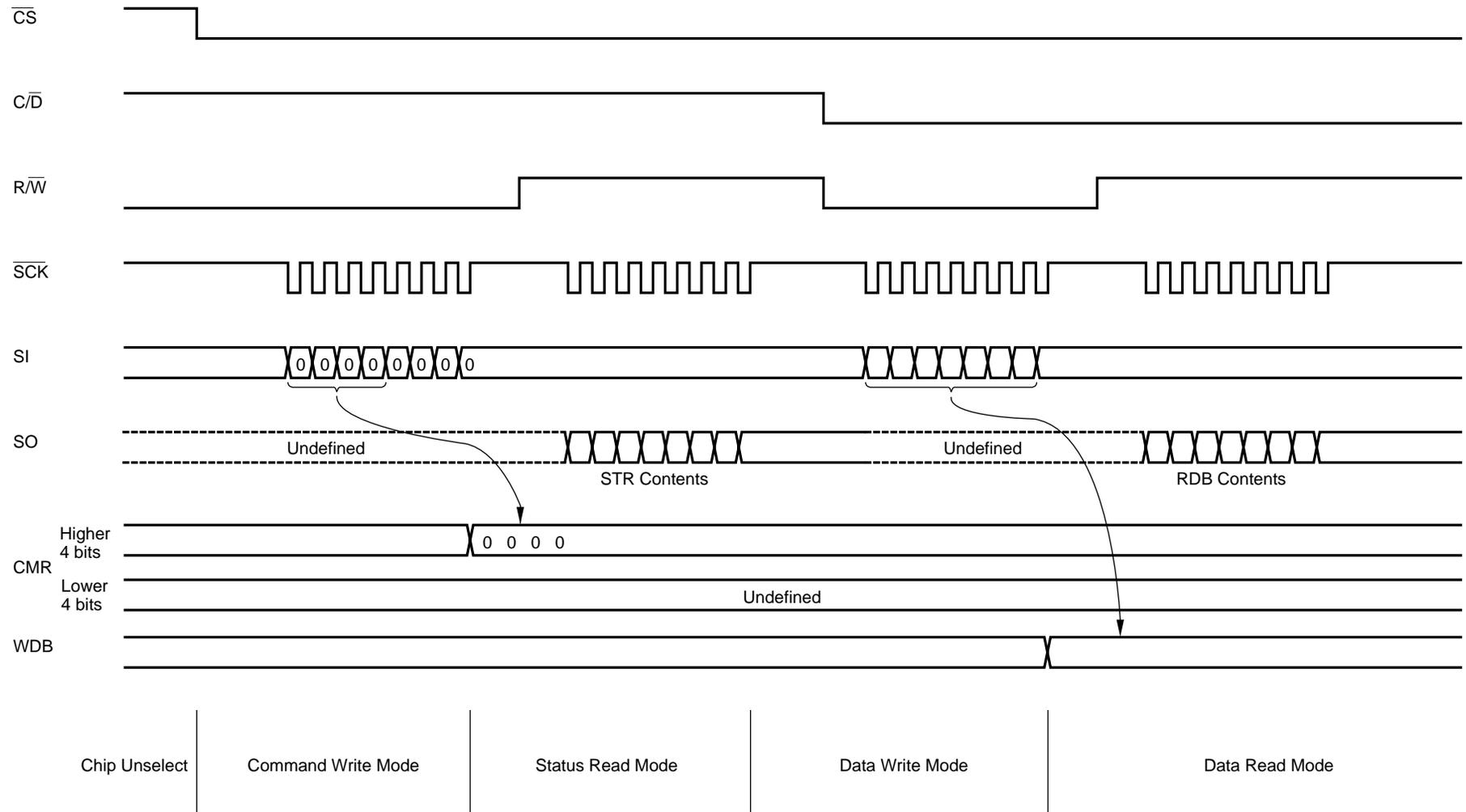
C/D	R/W	Host Interface Mode
0	0	Data write mode
0	1	Data read mode
1	0	Command write mode
1	1	Status read mode

Figure 4-1. Example of Host Controller Connections by Pin Control



Caution If the power supply voltage moves out of the 5 V ±5 % range, the RESET pin must be driven low for 6 μs or more in order to reset the μPD6708.

Figure 4-2. Host Interface Timing by Pin Control



4.2.2 Switching through software control

With the C/\overline{D} pin at the high level and the R/\overline{W} pin at the low level, the host interface mode can be switched from the host controller by setting bits 5 and 4 of CMR to the values shown in Table 4-5.

Table 4-5. Switching Host Interface Mode by Software Control

Bit 5	Bit 4	Host Interface Mode
0	0	Mode switching by pin control
0	1	Data write mode
1	0	Data read mode
1	1	Status read mode

After one byte of data has been forwarded, the host interface mode will become the command write mode, which is controlled by the C/\overline{D} and R/\overline{W} pins.

Figure 4-3. Example of Host Controller Connections by Software Control

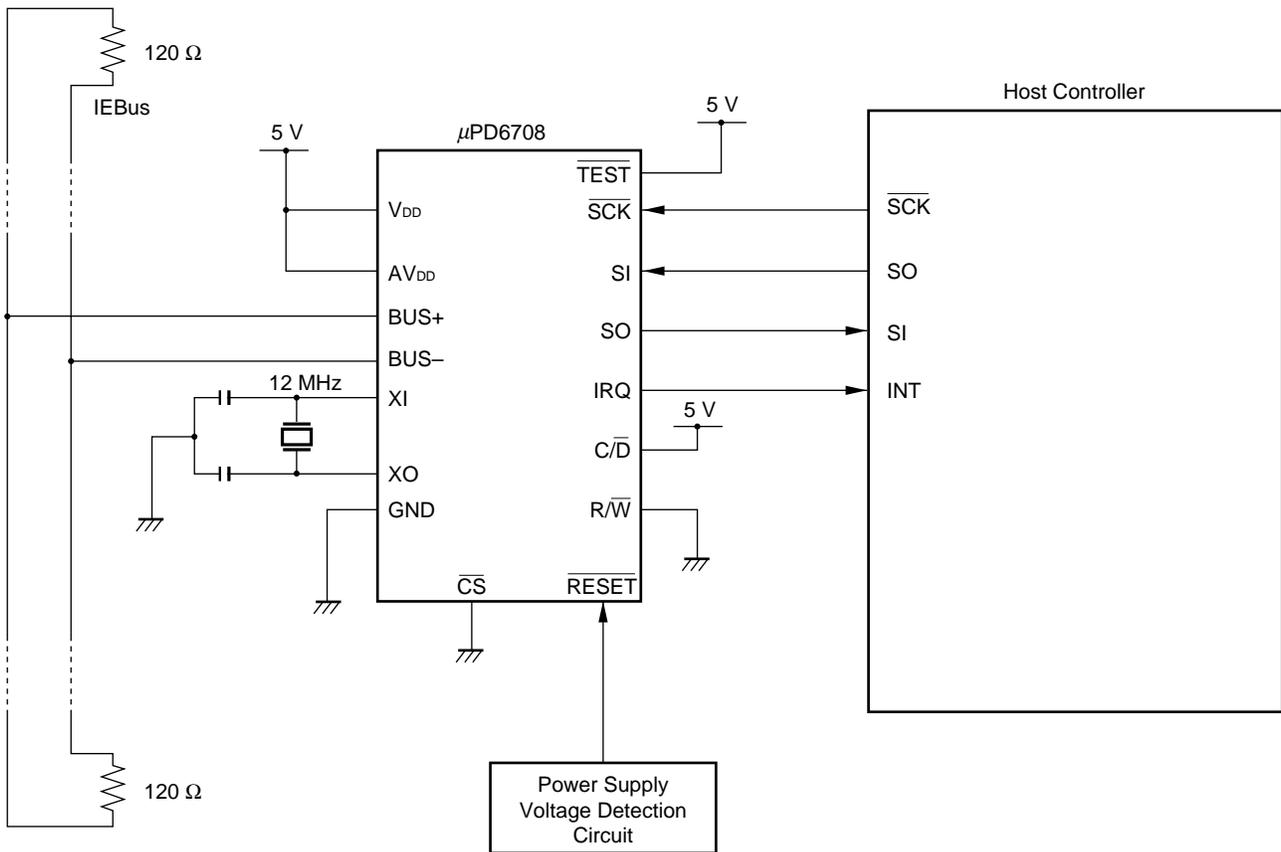
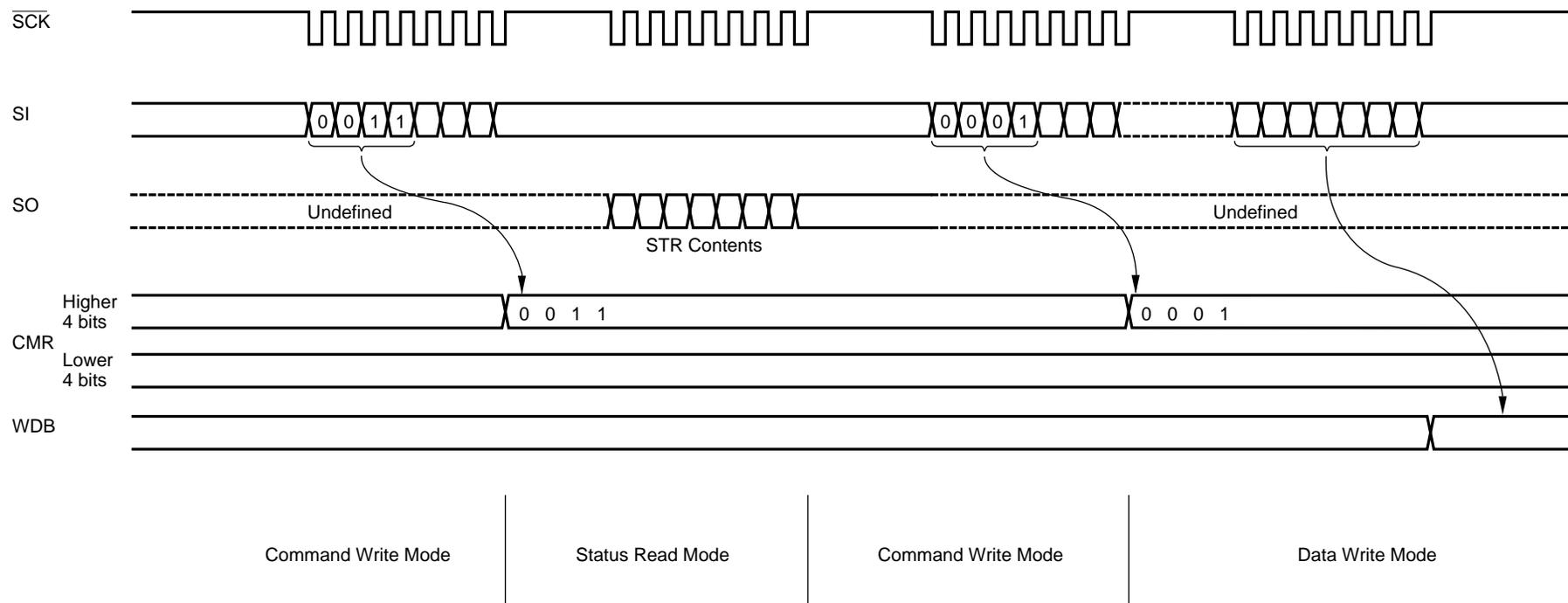


Figure 4-4. Host Interface Timing by Software Control



Remark Connect $\overline{C/D}$ and $\overline{R/W}$ pins to V_{DD} and GND, respectively.

4.3 Reset Mode

When the $\overline{\text{RESET}}$ pin is driven low, the μPD6708 enters the reset mode. To release the reset mode, the $\overline{\text{RESET}}$ pin must be driven high and a reset release command input.

There are two methods of resetting the μPD6708, as follows.

(1) Resetting with $\overline{\text{RESET}}$ pin

If the $\overline{\text{RESET}}$ pin is driven low, the μPD6708 will enter the reset mode.

To exit the reset mode, drive the $\overline{\text{RESET}}$ pin high and set bit 7 of CMR to '0'.

(2) Resetting by software

If bit 7 of CMR is set to '1' with the $\overline{\text{RESET}}$ pin fixed high, the μPD6708 will enter the reset mode.

To exit the reset mode, set bit 7 to '0'.

When powering on, the $\overline{\text{RESET}}$ pin must be driven low to execute a reset.

The μPD6708 will be in the following condition directly after leaving the reset mode.

<1> IEBus slave status is initialized.

Table 4-6. Slave Status Values after Leaving Reset Mode

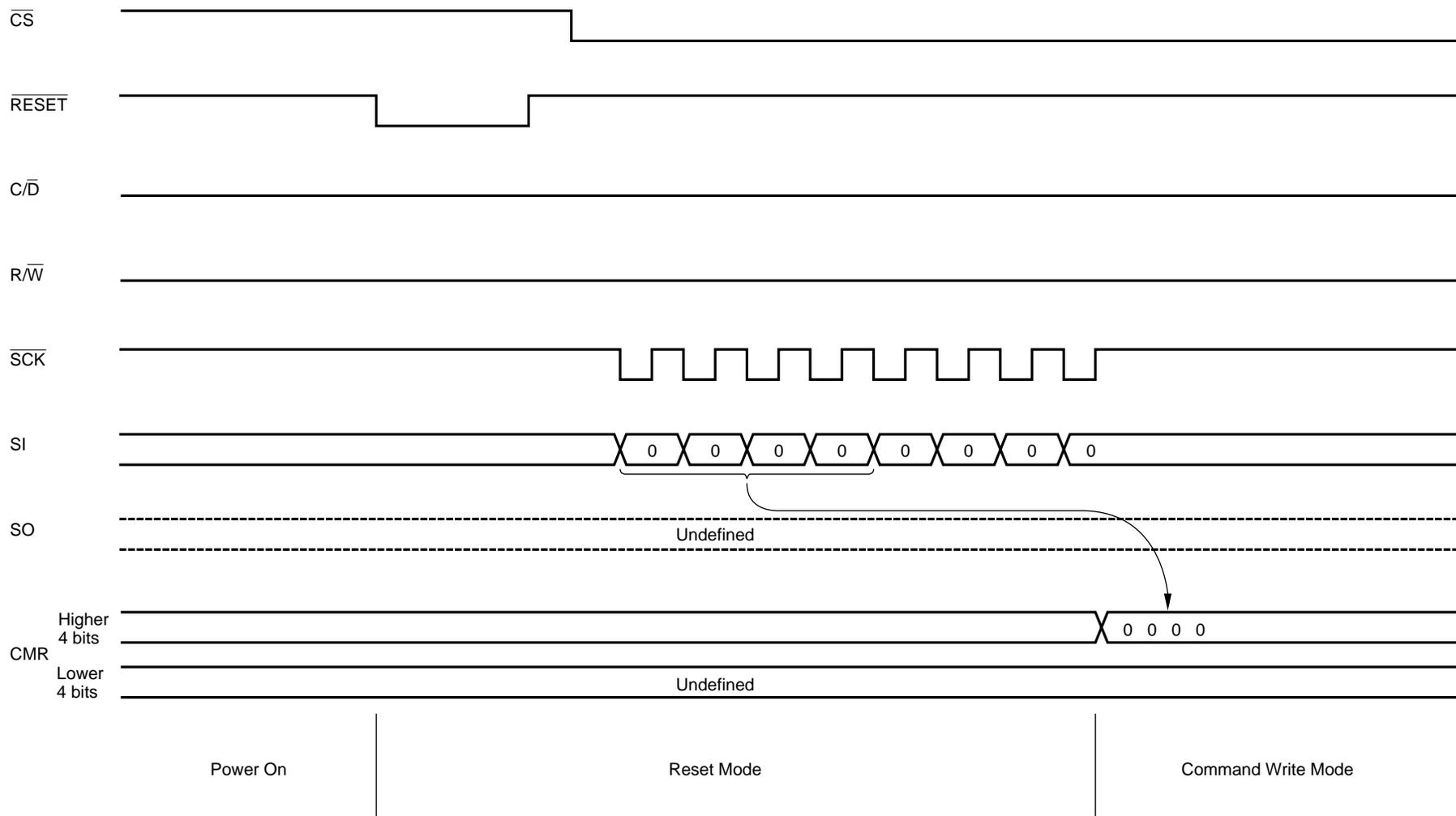
Bit	Value	Meaning
Bit 7	1	Up to mode 2 is supported
Bit 6	0	
Bit 5	0	Always '0'
Bit 4	0	The slave transmission section has stopped.
Bit 3	0	Always '0'
Bit 2	0	Unit is not locked.
Bit 1	0	Slave receive buffer is empty.
Bit 0	0	Slave transmit buffer is empty.

<2> WDB and RDB are empty.

<3> Reception is disabled.

Slave reception and broadcast reception are not acknowledged.

Figure 4-5. Example of $\overline{\text{RESET}}$ Control on Powering on



5. COMMUNICATION CONTROL COMMANDS

The operation conditions of the μPD6708 can be controlled by giving it a command from the host controller.

After a communication using the appropriate procedure (see 8.3.2 “Communication control command processing routine”), it is executed in a period in which communication is not being performed (standby state).

5.1 Overview of Communication Control Commands

Table 5-1. Overview of Communication Control Commands

Command Name	Description
INIT (Initialize)	Sets local address and initializes.
SETSA (Set slave address)	Sets the unit to communicate with.
MREQ1 (Master request 1)	Communicates as a master unit.
MREQ2 (Master request 2)	Continues in previous condition as the master unit and communicates.
ABORT (Abort)	Aborts communications.
SETSD (Set slave data)	Sets data for slave transmission.
GETSTA (Get status)	Reads communication status.
SETREV (Set receive)	Sets reception disabled state/enabled state.

(1) Write command

The command codes and command parameters for the write commands are shown in Table 5-2.

Table 5-2. Command Codes and Command Parameters of Write Commands

Command Name	Command Code (Lower 4 Bits of CMR)	Command Parameters (WDB)				
		First Byte		Second Byte	Third Byte	Fourth Byte
INIT	0000	Unit address		Condition setting code		
SETSA	0001	Slave address		0000		
MREQ1	0010	Broad-cast bits	Control bits	Number of master transmit data bytes ^{Note}	Master transmit data (first byte) ^{Note}	Master transmit data (second byte) ^{Note}
MREQ2	0011					
ABORT	0100					
SETSD	0101	Number of slave transmit data bytes		Slave transmit data (first byte)	Slave transmit data (second byte)	Slave transmit data (third byte)
SETREV	0111	Reception status code				

MSB LSB

Note Only set when transmitting.

Caution Note that even if the host controller makes a mistake in setting the number of command parameter bytes, an error message will not be returned by the μPD6708, and command processing will be performed as though it were a correctly set command.

(2) Read command

The command code of the read command is shown in Table 5-3.

Table 5-3. Command Code of Read Command

Command Name	Command Code (Lower 4 Bits of CMR)	Data Placed in RDB after Command Execution				
		First Byte		Second Byte		
GETSTA	0110	Lock status	Address of locked unit (12 bits)			
	MSB	LSB	MSB	LSB	MSB	LSB

Remark With a read command, the command execution result is placed in RDB, and therefore it is performed in the reception disabled state.

5.2 Communication Control Command Functions

5.2.1 INIT command (command code: 0000)

(1) Functions

<1> Unit address setting

This command sets the unit address (12 bits), The unit address will be used as the master address when a unit is communicating as the master unit, and as the slave address when a unit is communicating as the slave.

<2> Condition setting

- The status of bit 4 of IEBus slave status is set.
Slave transmission block operation enabled, stopped (bit 4)
(Setting of use/non-use of the function that transmits data to the master unit)
- The communication mode to be used is set.

Table 5-4. Condition Setting Method

Condition Setting Code		Condition Setting Contents
Bits 3 and 2	00	Communication performed in mode 0
	01	Communication performed in mode 1.
	10	Communication performed in mode 2.
	11	Undefined
Bit 1	0	Fixed at '0'
Bit 0	0	Slave transmission block stopped
	1	Slave transmission block operational

The local-station address and condition setting contents set by INIT command retain their set values unless power is turned off or reset mode is entered (see 4.3 “Reset Mode”).

<3> Slave status initialization

The slave status is initialized as shown in Table 5-5.

Table 5-5. Slave Status after Execution of INIT Command

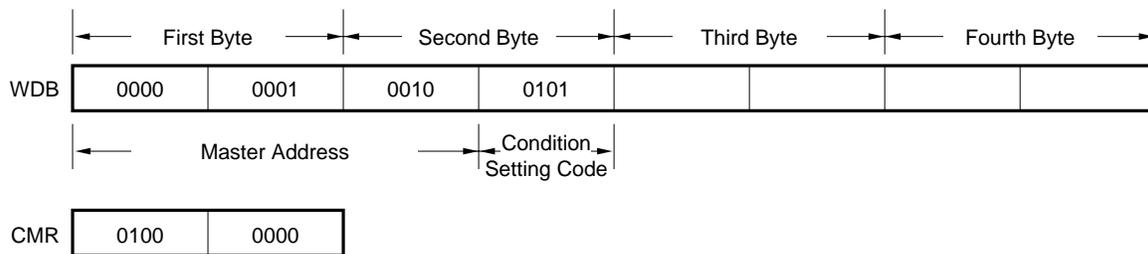
Bit	Value	Meaning
Bit 2	0	Unit is not locked.
Bit 0	0	Slave transmit buffer is empty.

<4> Slave transmission and broadcast reception are enabled.

<5> After the 2-byte command parameter (master address and condition setting code) have been read from the write data buffer (WDB), WDB is cleared.

(2) Example

When INIT command specifies that the master address is '012H' and the condition setting are 'communication in mode 1' and 'slave data transmission section operable', the contents of WDB and CMR are as shown below.



5.2.2 SETSA command (command code: 0001)

(1) Functions

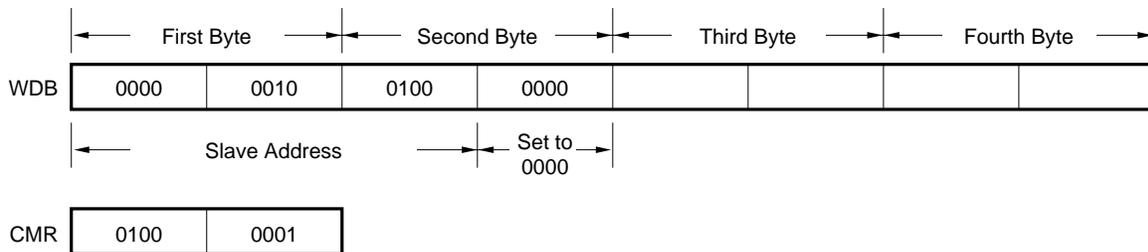
<1> Slave address (12 bits) setting

The value for the slave address set by the SETSA command remains unchanged until the power is turned off or the reset mode is entered.

<2> This command clears WDB after reading the 2-byte command parameter (slave address) from WDB.

(2) Example

When SETSA command sets the slave address as '024H', the contents of WDB and CMR are as shown below.



5.2.3 MREQ1 command (command code: 0010)

(1) Functions

This command executes a master communication (transmission or reception). After execution of the command, the unit begins communication as the master unit. As long as it does not lose in arbitration, the master unit will communicate with the slave unit which has the slave address specified by SETSA command.

- <1> Selected broadcast communication or ordinary communication
 Broadcast communication selection : 0H (broadcast bit '0' output)
 Ordinary communication selection : 8H (broadcast bit '1' output)
- <2> Sets the control bits (4 bits)
- <3> Sets the number of transmit data bytes (8 bits) (transmission only)

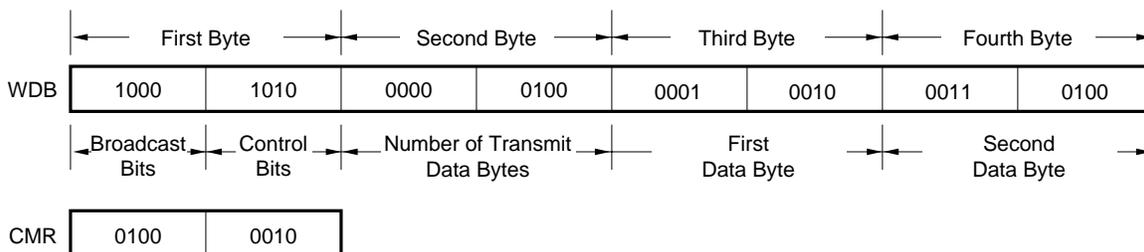
Table 5-6. Number of Transmit Data Bytes Setting

Number of Transmit Data Bytes	Command Parameter
1 byte	1H
2 bytes	2H
:	:
:	:
255 bytes	FFH
256 bytes	00H

- <4> Sets the transmit data (transmission only)

(2) Example

When the MREQ1 command is used to select 'ordinary communication', set the control bit to 'AH' (command write and lock), the number of transmit data bytes to 4, and the transmit data to 12H, 34H, 56H, and 78H, the contents of WDB and CMR are shown below.



Caution Transmit data 56H and 78H should be set when the above command parameters have been read and WDB is empty.

5.2.4 MREQ2 command (command code: 0011)

(1) Functions

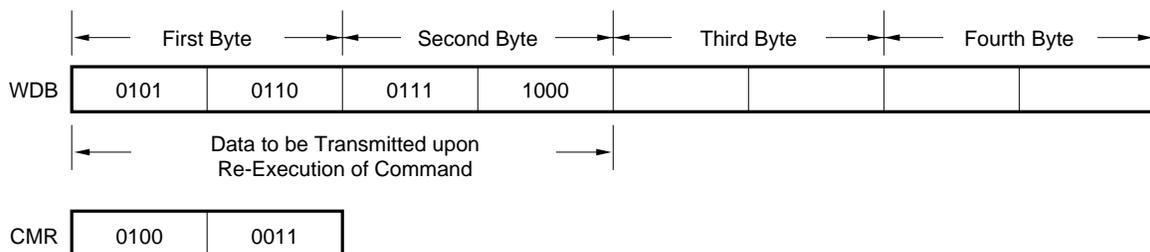
This command re-executes a master communication (transmission or reception). If master transmission or reception stops midway, the master communication is re-executed from the stopped condition.

(2) Command execution conditions

If a communication control command other than an MREQ2 command is executed after the master communication ends midway, the MREQ1 command may not re-execute the communication correctly from the communication interrupted condition.

(3) Example

When re-execution is performed by the MREQ2 command when communication has been interrupted due to generation of a timing error after transmission of two bytes (12H and 34H) in mode 1, as in the MREQ1 command example, the contents of WDB and CMR are as shown below.



The previously set MREQ1 command values are used for the broadcast bits, control bits and number of transmit data bytes.

- Cautions**
1. A master communication performed by execution of the MREQ1 and MREQ2 commands is performed in only one frame. However, if the unit loses in arbitration, the frame is automatically reset up twice (three times in total).
 2. INIT command must be executed before setting the MREQ1 or MREQ2 command. If MREQ1 or MREQ2 is set before execution of INIT command, master communication will not be performed.

5.2.5 ABORT command (command code: 0100)

(1) Functions

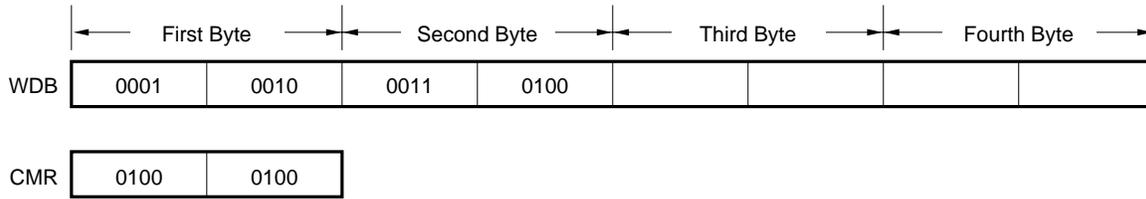
This command aborts master communications and slave unit data transmissions.

- <1> It clears the data placed in WDB.
- <2> It cancels the slave transmit data (SETSD command).

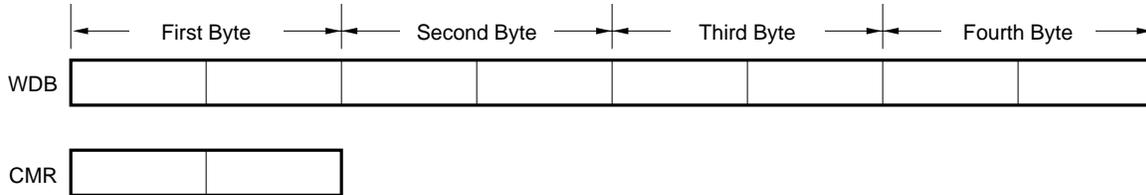
(2) Example

When the master unit begins communication as in the MREQ1 command example, a communication error is generated and the two bytes of transmit data (12H and 34H) remaining in WDB are canceled by ABORT command, the contents of CMR are as shown below.

[Before execution of ABORT command]



[After execution of ABORT command]



The data placed in WDB is cleared.

5.2.6 SETSD command (command code: 0101)

(1) Functions

This command specifies the data transmitted to the master unit when a 'data read and lock' (control bits: 3H) or a 'data read' (control bits: 7H) is received from the master unit.

<1> Sets the number transmit data bytes (8 bits)

Table 5-7. Number of Transmit Data Bytes

Number of Transmit Data Bytes	Command Parameter
1 byte	1H
2 bytes	2H
:	:
:	:
64 bytes	40H

<2> Sets the transmit data

(2) Validity of SETSD command

When SETSD command is executed, it remains valid until one of the following cases arises.

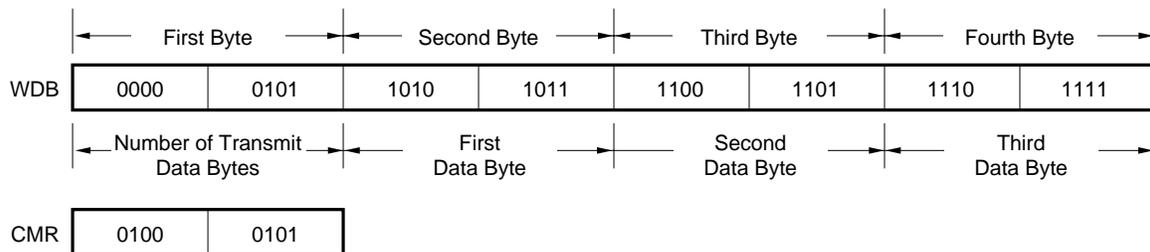
- 'Data read and lock' (control bits: 3H) or 'data read' (control bits: 7H) is received from the master unit.
- ABORT command is executed.
- Power is turned off, or the reset mode is entered.

When the SETSD command is valid, WDB functions as the slave transmit buffer.

Caution The SETSD command can be executed even if the unit is placed in the slave transmission selection halted state by INIT command.

(3) Example

When the SETSD command is used to set the number of transmit data bytes to 5, and the transmit data to ABH, CDH, EFH, 14H, and 25H, the contents of WDB and CMR are as shown below.



Caution Transmit data 14H and 25H should be set when the above command parameters have been read and WDB is empty.

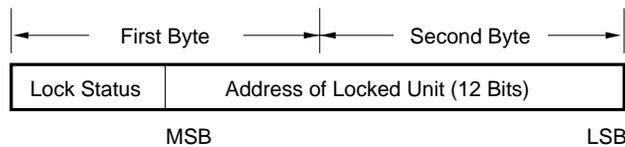
5.2.7 GETSTA command (command code: 0110)

The GETSTA command is used by a unit to check whether it is locked by another unit.

(1) Functions

- <1> Reads the lock status which indicates whether or not this unit is locked by another unit. '1J' is placed in RDB if the unit is locked, and '0H' if not locked.
- <2> The address (12 bits) of a locked unit is placed in RDB. This data is meaningless when the unit is not locked.

After execution of GETSTA command, the data placed in RDB is as follows.



(2) Command setting conditions

The reception disabled state must be set and RDB emptied before setting the GETSTA command.

Caution With the IEBus, the lock function is provided to enable communication to run over a number of frames. However, if a locked unit goes down without being unlocked, the locked unit is unable to receive any further data. To avoid this situation, in a system which uses the lock function it is necessary to execute the GETSTA command periodically to monitor the lock status (a unit lock is released by executing INIT command).

5.2.8 SETREV command (command code: 0111)**(1) Functions****<1> Set reception enabled/disabled status**

- When reception status code is 00H : Set to reception disabled status.
In the reception disabled status, bit 1 of the slave status is '1', the slave receive buffer becomes virtually empty and no longer exists, and slave reception and broadcast reception are no longer performed.
- When reception status code is 01H : Set to reception enabled status.
When the prescribed conditions are met, slave reception and broadcast reception are performed.

The reception enabled status is also set when a command other than SETREV is executed.

<2> This command clears WDB after reading the 1-byte command parameter (reception status code) from WDB.

6. RETURN CODES

The μPD6708 sets the communication status as a return code in lower 4 bits of the status register (STR) and requests an interrupt (IRQ output). As a result of the interrupt request from the μPD6708, the host controller can ascertain the communication result by reading the return code in the status read mode.

6.1 Return Codes in Master/Slave Data Transmission

Table 6-1 shows the return codes placed in the status register when a unit has executed the MREQ1 or MREQ2 command and becomes the master unit (including broadcast communication), and when the SETSD command is executed and the slave unit transmits data.

Table 6-1. Return Codes in Master/Slave Transmission

Return Code Name	Code	Description
Transmission start	0000	Indicates that master/slave transmission will start. The point of generation differs between master transmission and slave transmission. <1> Master transmission Set when the master address field ends and the unit wins as the master unit. <2> Slave transmission Set when control bits (3H, 7H) which request data transmission are received from the master unit.
Transmission normal termination	0010	Indicates that transmission of the number of data bytes specified by the message length bits has ended within one frame.
Termination during transmission	0011	Indicates that the communication has ended without completion of transmission of the number of data bytes specified by the message length bits within one frame. In master transmission, termination during transmission is not flagged if the unit loses once in arbitration, and transmission is attempted up to three times.

6.2 Return Codes in Master Reception

Table 6-2 shows the return codes placed in STR when a unit has executed the MREQ1 or MREQ2 command and becomes the master unit, and receives data, a status or lock address from a slave unit.

Table 6-2. Return Codes in Master Reception

Return Code Name	Code	Description
Master reception start	0100	This return code is generated when the master unit correctly receives the message length code from the slave unit, informing the host controller of the start of master reception.
Master receive data full	0101	Each time 20 bytes (RDB capacity) of master receive data is received, if RDB is full, this return code makes a request to the host controller for a read of receive data from RDB.
Master reception normal termination	0110	Indicates that reception of the number of data bytes specified by the message length bits has ended within one frame.
Termination during master reception	0111	Indicates that the communication has ended without completion of transmission of the number of data bytes specified by the message length bits within one frame. Termination during master reception is not flagged if the unit loses once in arbitration, and reception is attempted up to three times.

6.3 Return Codes in Slave Reception

Table 6-3 shows the return codes placed in STR when data or a command is received from the master unit.

Table 6-3. Return Codes in Slave Reception

Return Code Name	Code	Description
Slave reception start	1000	This return code is generated when the slave unit correctly receives the message length codes from the master unit, informing the host controller of the start of slave reception.
Slave receive data full	1001	Each time 20 bytes (RDB capacity) of slave receive data is received, if RDB is full, this return code makes a request to the host controller for a read of receive data from RDB.
Slave reception normal termination	1010	Indicates that reception of the number of data bytes specified by the message length bits has ended within one frame.
Termination during slave reception	1011	Indicates that the communication has ended without completion of transmission of the number of data bytes specified by the message length bits within one frame.

6.4 Return Codes in Broadcast Reception

Table 6-4 shows the return codes placed in STR when data or a command is received from the master unit in broadcast communication.

Table 6-4. Return Codes in Broadcast Reception

Return Code Name	Code	Description
Broadcast reception start	1100	This return code is generated when the slave unit correctly receives the message length codes from the master unit, informing the host controller of the start of slave reception.
Broadcast receive data full	1101	Each time 20 bytes (RDB capacity) of slave receive data is received, if RDB is full, this return code makes a request to the host controller for a read of receive data from RDB.
Broadcast reception normal termination	1110	Indicates that reception of the number of data bytes specified by the message length bits has ended within one frame.
Termination during broadcast reception	1111	Indicates that the communication has ended without completion of transmission of the number of data bytes specified by the message length bits within one frame.

6.5 Return Codes Generation Intervals

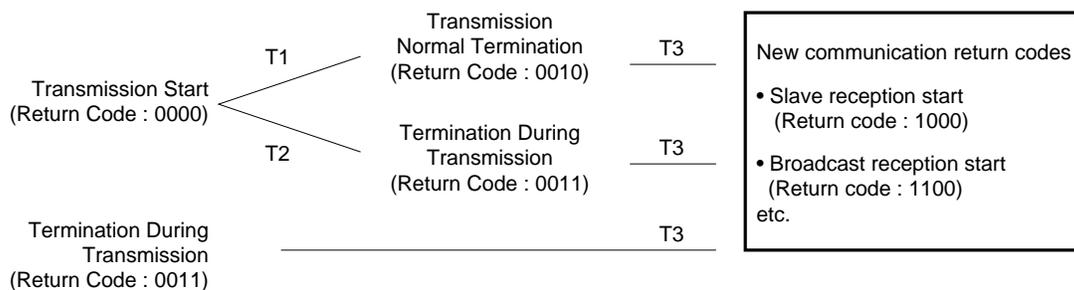
This section describes the generation order and the minimum generation interval for return codes generated each time communication is performed,

Each time a new return code is generated, it is placed in STR without regard to STR read. For this reason, the host controller must take account of the minimum return code generation interval in controlling the μPD6708.

(1) Master transmission

After execution of the MREQ1 or MREQ2 command, the order of generation of master transmission return codes is as shown in Figure 6-1.

Figure 6-1. Return Code Generation Order in Master Transmission



The minimum generation intervals for return codes in master transmission are shown below.

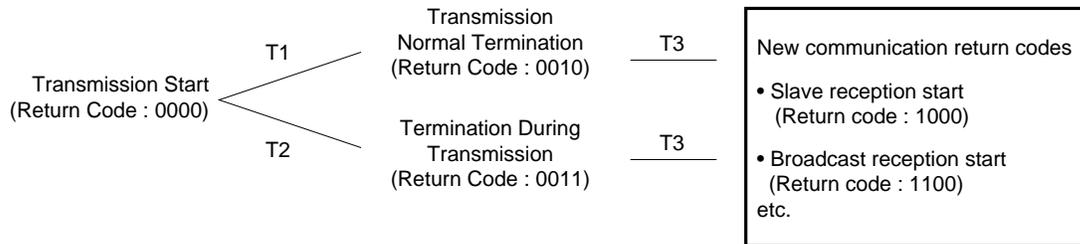
Table 6-5. Minimum Generation Intervals for Return Codes in Master Transmission (μs)

Time	Mode 0	Mode 1	Mode 2
T1	Approx. 6325	Approx. 1605	Approx. 1160
T2	Approx. 10	Approx. 10	Approx. 10
T3	Approx. 7290	Approx. 2050	Approx. 1550

(2) Slave transmission

After execution of the SETSD command, the order of generation of slave transmission return codes is as shown in Figure 6-2.

Figure 6-2. Return Code Generation Order in Slave Transmission



The minimum generation intervals for return codes in slave transmission are shown below.

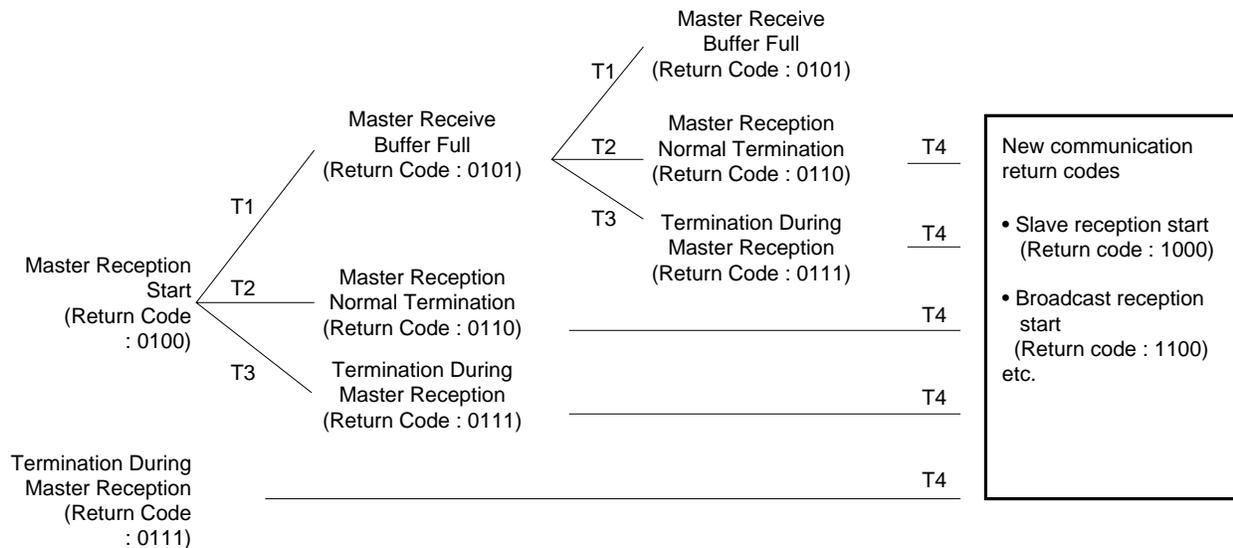
Table 6-6. Minimum Generation Intervals for Return Codes in Slave Transmission (μs)

Time	Mode 0	Mode 1	Mode 2
T1	Approx. 1580	Approx. 400	Approx. 290
T2	Approx. 10	Approx. 10	Approx. 10
T3	Approx. 7290	Approx. 2050	Approx. 1550

(3) Master reception

After execution of the MREQ1 or MREQ2 command, the order of generation of master reception returns codes is as shown in Figure 6-3.

Figure 6-3. Return Code Generation Order in Master Reception



The minimum generation intervals for return codes in master transmission are shown below.

Table 6-7. Minimum Generation Intervals for Return Codes in Master Reception (μs)

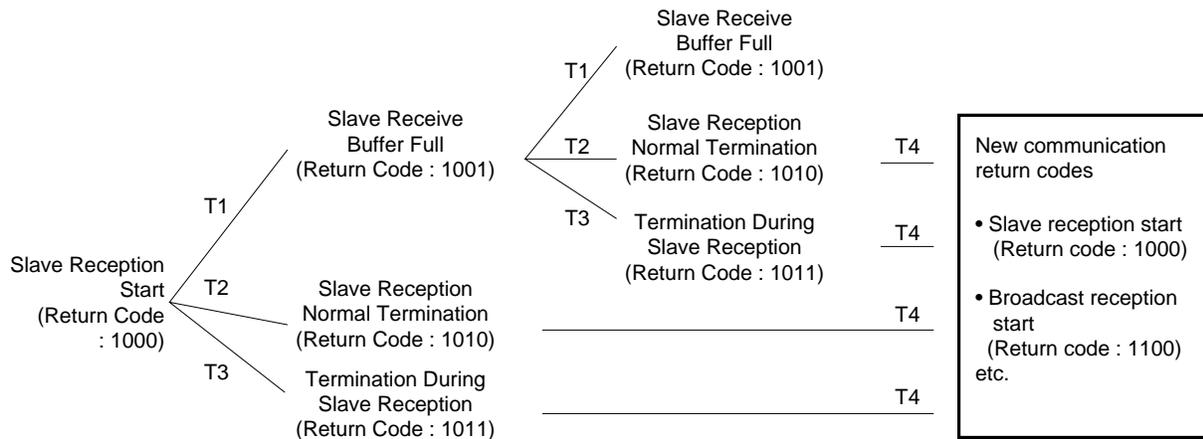
Time	Mode 0	Mode 1	Mode 2
T1	Note	Approx. 8030	Approx. 5800
T2	Approx. 1580	Approx. 400	Approx. 290
T3	Approx. 10	Approx. 10	Approx. 10
T4	Approx. 7290	Approx. 2050	Approx. 1550

Note The mode 0 master receive data consists of up to 19 bytes. Therefore, the receive buffer (20 bytes) does not become full, and a return code is not generated.

(4) Slave reception

The order of generation of slave reception return codes is as shown in Figure 6-4.

Figure 6-4. Return Code Generation Order in Slave Reception



The minimum generation intervals for return codes in master transmission are shown below.

Table 6-8. Minimum Generation Intervals for Return Codes in Slave Reception (μs)

Time	Mode 0	Mode 1	Mode 2
T1	Note	Approx. 8030	Approx. 5800
T2	Approx. 1580	Approx. 400	Approx. 290
T3	Approx. 10	Approx. 10	Approx. 10
T4	Approx. 7290	Approx. 2050	Approx. 1550

Note The mode 0 master receive data consists of up to 19 bytes. Therefore, the receive buffer (20 bytes) does not become full, and a return code is not generated.

(5) Broadcast reception

The return code generation order and minimum generation intervals in the case of broadcast reception are the same as for **(4) Slave reception** as shown above.

7. COMMUNICATING WITH HOST CONTROLLER

This section explains the flow of data between the μPD6708 and the host controller via the serial interface (\overline{SCK} , SO, SI pins) during communications.

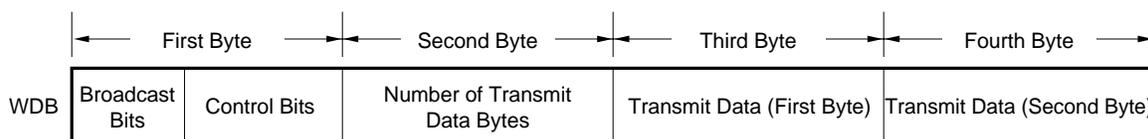
7.1 Master Transmission

Master transmission is the communication data exchange which takes place when a unit becomes a master unit by specifying AH, BH, and FH as control bits and executing the MREQ1 or MREQ2 command, and then transmitting data and commands to slave units.

7.1.1 Master transmission by MREQ1 command

- (1) The control bits, number of transmit data bytes and transmit data are placed in the write WDB as command parameters as shown in Figure 7-1, and the MREQ1 command (command code: 2H) is executed.
- (2) When a unit wins in arbitration as the master unit (the end of the master address field), the transmission start return code (0H) is placed in the status register (STR), and an interrupt request is generated for the host controller. At this time, the host controller places the third and following bytes of transmit data in WDB.

Figure 7-1. Data Exchange During Master Transmission (Contents of WDB)



- (3) If the number of data or command bytes specified by “number of transmit data bytes” are transmitted correctly, a “transmission end” return code (2H) will be placed in STR and an interrupt request will be generated.
- (4) If an error occurs during transmission and the data communication is halted, a “termination during transmission” return code (3H) will be placed in STR and an interrupt request will be generated.

The timing at which the μPD6708 reads transmit data from WDB is shown below.

Table 7-1. Timing for Reading Transmit Data from WDB
(The minimum time after transmission start return code (0H) is set in STR)

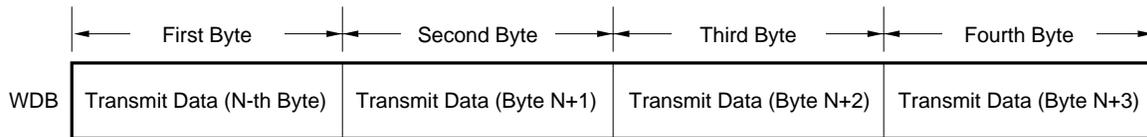
Transmit Data	Timing for Reading Transmit Data from WDB (μs)		
	Mode 0	Mode 1	Mode 2
Transmit data (first byte)	Approx. 4745	Approx. 1205	Approx. 870
Transmit data (second byte)	Approx. 6325	Approx. 1605	Approx. 1160
:	:	:	:
Transmit data (N-th byte)	Approx. 3165+1580 × N	Approx. 805+400 × N	Approx. 580+290 × N

7.1.2 Master transmission by MREQ2 command

If the MREQ1 command is executed and a communication error occurs during the transmission of data or a command, with the result that not all the data is transmitted, the remaining data can be transmitted by executing the MREQ2 command.

- (1) The remaining data is placed in WDB as command parameters as shown in Figure 7-2, and the MREQ2 command (command code: 3H) is executed.

Figure 7-2. Data Exchange During Master Transmission (Contents of Write Data Buffer)



The remaining operations are the same as 7.1.1 “Master transmission by MREQ1 command”.

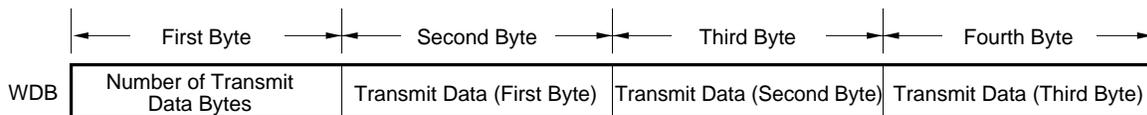
7.2 Slave Transmission

7.2.1 Data transmission

When the slave unit receives control bit 3H or 7H from the master unit, it transmits data as follows.

- (1) It places the number of transmit data bytes and transmit data in WDB as command parameters as shown in Figure 7-3, and executes the SETSD command (command code: 5H).
- (2) When the unit receives control bit 3H or 7H from the master unit, the “transmission start” return code (0H) is placed in STR, and an interrupt request is generated. At this time, the host controller places the fourth and following bytes of transmit data in WDB.

Figure 7-3. Data Exchange During Slave Transmission (Contents of WDB)



- (3) If the number of data or command bytes specified by “number of transmit data bytes” is transmitted correctly, a “transmission end” return code (2H) will be placed in STR and an interrupt request will be generated.
- (4) If an error occurs during transmission and the data communication is halted, a “termination during transmission” return code (3H) will be placed in STR and an interrupt request will be generated.

The timing at which the μPD6708 reads transmit data from WDB is shown below.

Table 7-2. Timing for Reading Transmit Data from WDB
 (The minimum time after transmission start return code (0H) is set)

Transmit Data	Timing for Reading Transmit Data from WDB (μs)		
	Mode 0	Mode 1	Mode 2
Transmit data (first byte)	Approx. 1580	Approx. 400	Approx. 290
Transmit data (second byte)	Approx. 3160	Approx. 800	Approx. 580
⋮	⋮	⋮	⋮
Transmit data (N-th byte)	Approx. 1580 × N	Approx. 400 × N	Approx. 290 × N

7.2.2 Transmitting slave status address and lock address

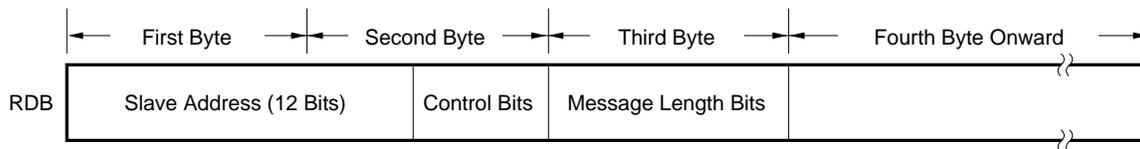
When the μPD6708 receives 0H, 4H, 5H, and 6H as control bits from the master unit, the slave status and lock address are generated automatically and sent to the master unit. As a result, there is no necessity for the host controller to be involved in the transmission of the slave status and lock address.

7.3 Master Reception

When a unit becomes a master unit by setting 0H, 3H, 4H, 5H, 6H, and 7H as control bits and executing the MREQ1 or MREQ2 command, and receives data, slave status and lock address from the slave unit, the following will occur.

- (1) When the master unit returns an acknowledge in the message length field, the slave address, control bits and message length bits are placed in the read data buffer (RDB) by the MREQ1 command as shown in Figure 7-4, the “master reception start” return code (6H) is placed in STR, and an interrupt request is generated.

Figure 7-4. Data Exchange During Master Reception (Contents of RDB)



- (2) Each time one byte of receive data is received, it is placed in RDB.
- (3) Each time 20 bytes (RDB capacity) of receive data are received, if RDB is full, a “master receive buffer full” return code (5H) is placed in STR, and an interrupt request is generated.
- (4) After one frame of data is placed in RDB, a “master reception normal termination” return code (6H) is placed in STR and an interrupt request is generated.
- (5) If a communication error occurs during reception and communication stops without receiving all of the data transmitted from the slave unit, a “termination during master reception” return code (7H) is placed in STR and an interrupt request is generated.

The areas where μPD6708 places receive data, etc., in RDB are shown below.

Table 7-3. Placing Receive Data in RDB

Parameter/ Communication Data	Areas where RDB is read	Time (μs) ^{Note}		
		Mode 0	Mode 1	Mode 2
Slave address	When ACK bit is transmitted in message length field	0	0	0
Control bits				
Message length bits				
Receive data (first byte)	When first ACK bit is transmitted in data field	Approx. 1580	Approx. 400	Approx. 290
Receive data (second byte)	When second ACK bit is transmitted in data field	Approx. 3160	Approx. 800	Approx. 580
Receive data (N-th byte)	When N-th ACK bit is transmitted in data field	Approx. 1580 × N	Approx. 400 × N	Approx. 290 × N

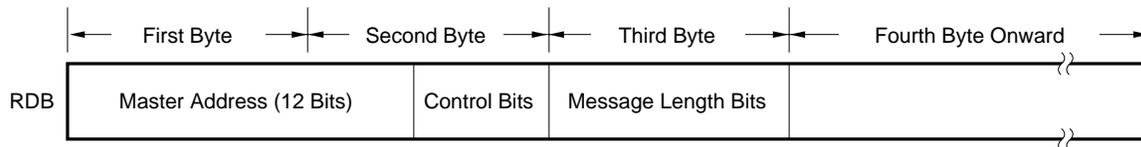
Note Minimum time after the “master reception start” return code is placed in STR and an interrupt request is generated.

7.4 Slave Reception

If a slave unit receives AH, BH, EH, or FH as control bits from the master unit and receive data or a command, the following will occur.

- (1) When the slave unit returns an acknowledge in the message length field, the master address, control bits and message length bits are placed in RDB as shown in Figure 7-5, a “slave reception start” return code (8H) is placed in STR, and an interrupt request is generated.

Figure 7-5. Data Exchange During Slave Reception (Contents of RDB)



- (2) Each time one byte of receive data is received, it is placed in RDB.
- (3) Each time 20 bytes (RDB capacity) of receive data are received, if RDB is full, a “slave receive buffer full” return code (9H) is placed in STR, and an interrupt request is generated.
- (4) After the final data of one frame is placed in RDB, a “slave reception normal termination” return code (AH) is placed in STR and an interrupt request is generated.
- (5) If a communication error occurs during reception and communication stops without receiving all of the data transmitted from the master unit, a “termination during slave reception” return code (BH) is placed in STR and an interrupt request is generated.

The areas where the μPD6708 places receive data, etc., in RDB are shown below.

Table 7-4. Placing Receive Data in RDB

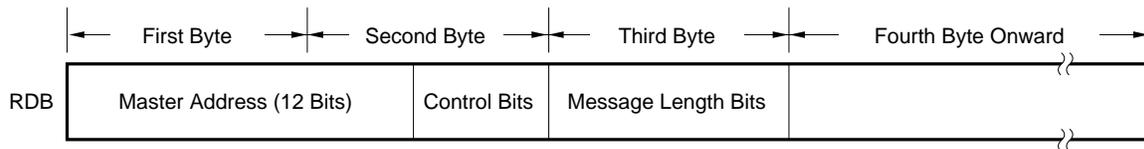
Parameter/ Communication Data	Areas where RDB is read	Time (μs) ^{Note}		
		Mode 0	Mode 1	Mode 2
Master address	When ACK bit is received in message length field	0	0	0
Control bits				
Message length bits				
Receive data (first byte)	When first ACK bit is received in data field	Approx. 1580	Approx. 400	Approx. 290
Receive data (second byte)	When second ACK bit is received in data field	Approx. 3160	Approx. 800	Approx. 580
Receive data (N-th byte)	When N-th ACK bit is received in data field	Approx. 1580 × N	Approx. 400 × N	Approx. 290 × N

Note Minimum time after the “slave reception start” return code is placed in STR and an interrupt request is generated.

7.5 Broadcast Reception

- (1) When a broadcast reception unit receives the message length field, the master address, control bits and message length bits are placed in RDB as shown in Figure 7-6, a “broadcast reception start” return code (CH) is placed in STR, and an interrupt request is generated.

Figure 7-6. Data Exchange During Broadcast Reception (Contents of RDB)



- (2) Each time one byte of receive data is received, it is placed in RDB.
- (3) Each time 20 bytes (RDB capacity) of receive data are received, if RDB is full, a “slave receive buffer full” return code (DH) is placed in STR, and an interrupt request is generated.
- (4) After the final data of one frame is placed in RDB, a “broadcast reception normal termination” return code (EH) is placed in STR and an interrupt request is generated.
- (5) If a communication error occurs during reception and communication stops without receiving all of the data transmitted from the master unit, a “termination during broadcast reception” return code (FH) is placed in STR and an interrupt request is generated.

The areas where the μPD6708 places receive data, etc., in RDB are the same as those shown in Table 7-4.

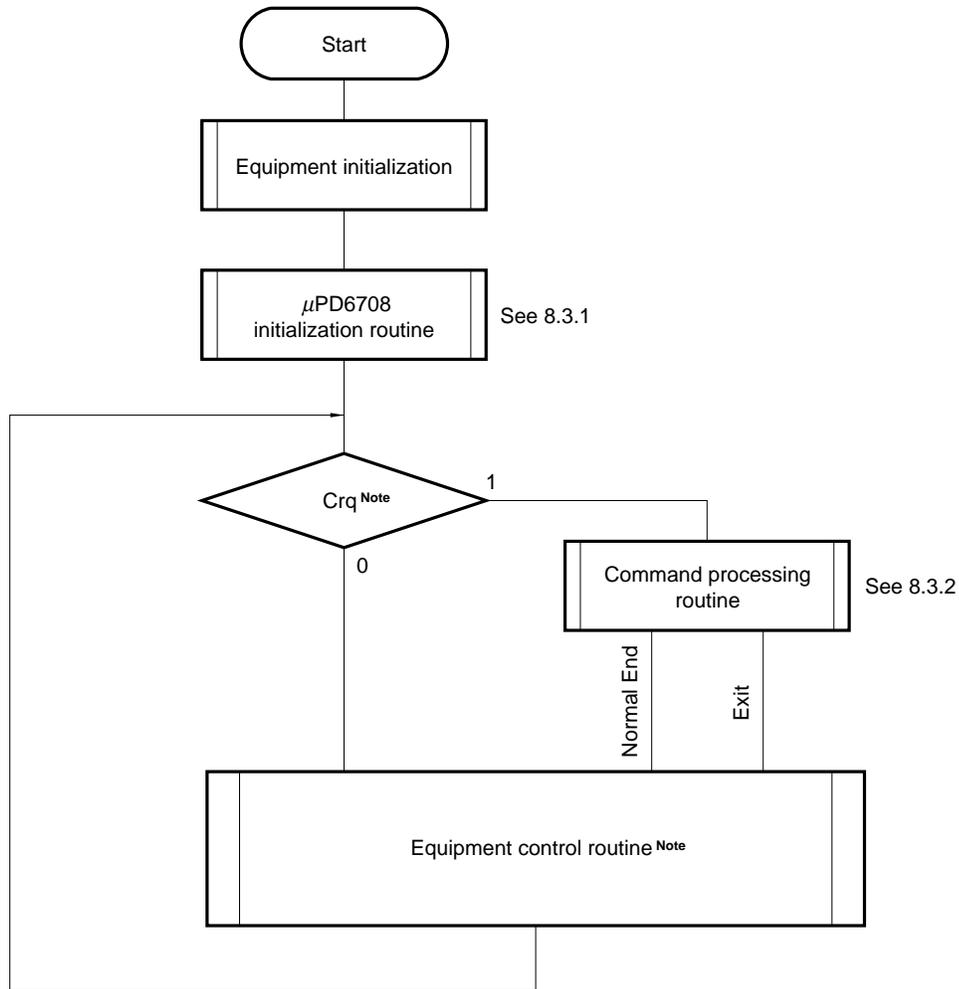
8. EXAMPLE OF HOST CONTROLLER PROCESSING FLOW

This chapter presents an example of the host controller processing flow for controlling the μPD6708 via the serial interface (SCK, SO, and SI pins).

The host controller processing flow comprises the following routines.

- Main routine
- Interrupt service routine

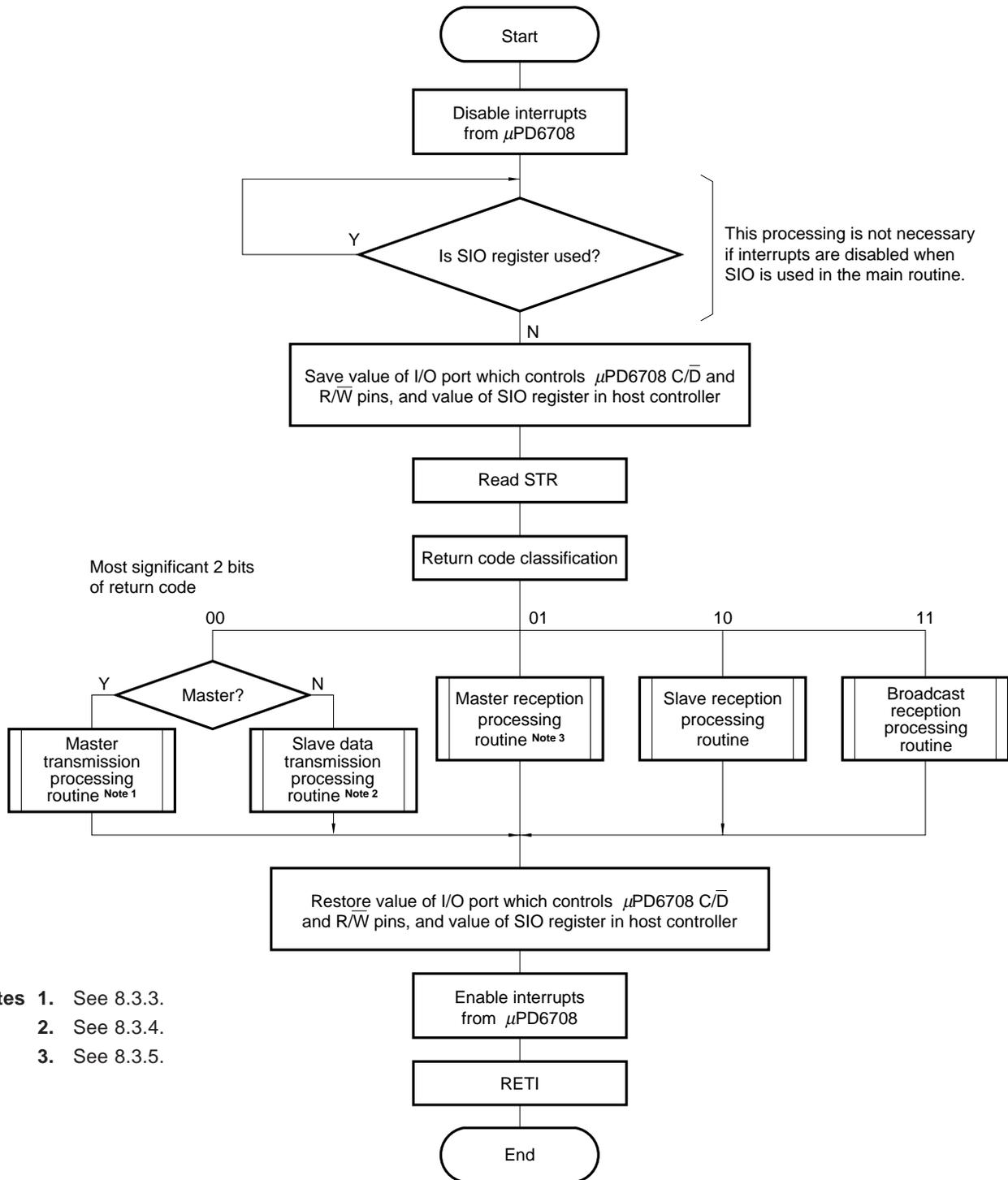
8.1 Main Routine



Note The equipment control routine is a routine which performs host controller application processing. When controlling the μPD6708, set the Crq flag when there is a command request.

8.2 Interrupt Service Routine

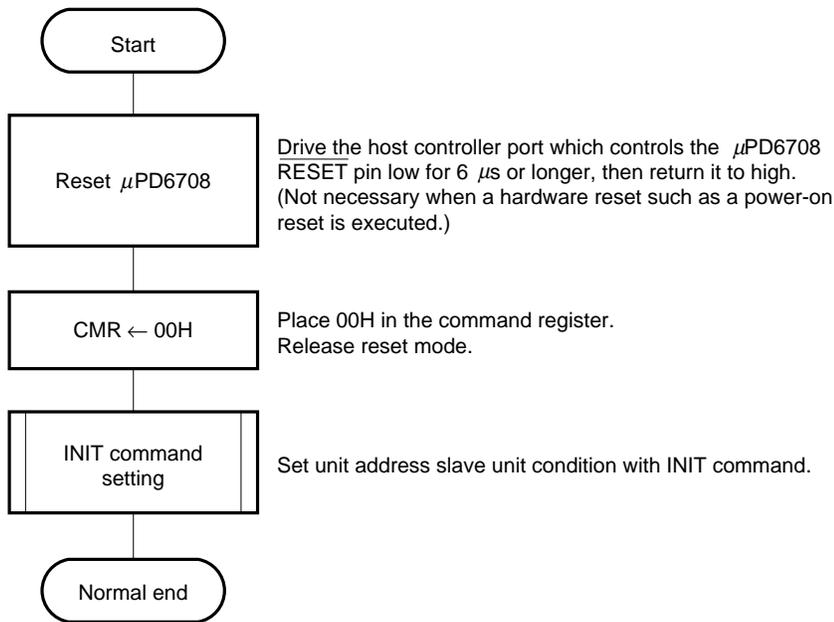
This routine is used when an interrupt request is generated from the μPD6708 (when IRQ pin becomes high). This routine reads the contents of the return code, performs flag setting, and reads receive data.



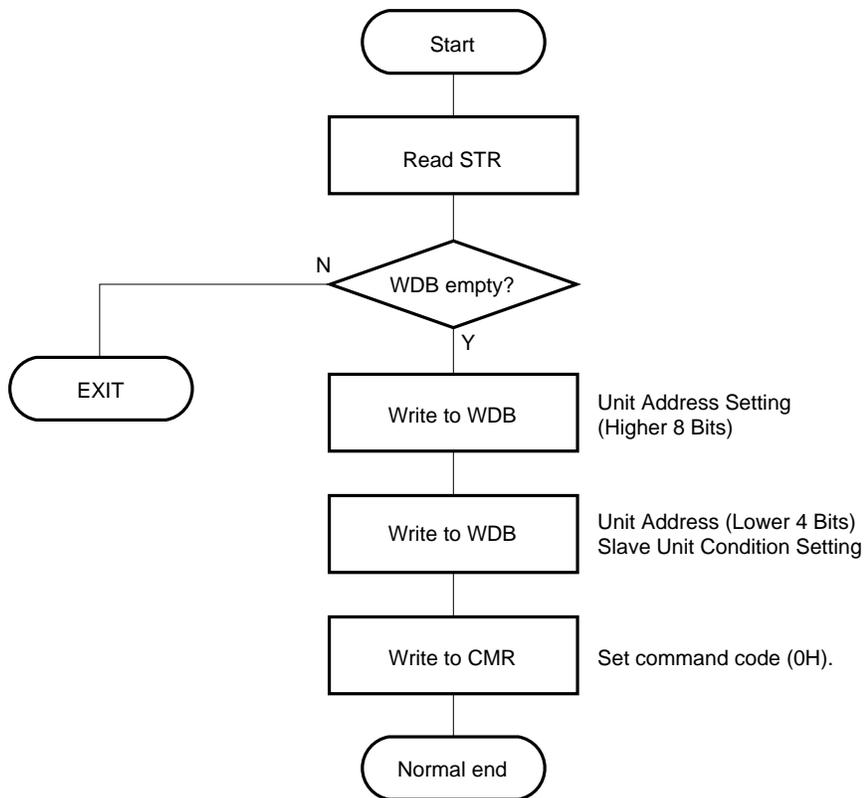
- Notes**
- 1. See 8.3.3.
 - 2. See 8.3.4.
 - 3. See 8.3.5.

8.3 Processing Routine

8.3.1 μPD6708 initialization routine



• INIT command setting flow

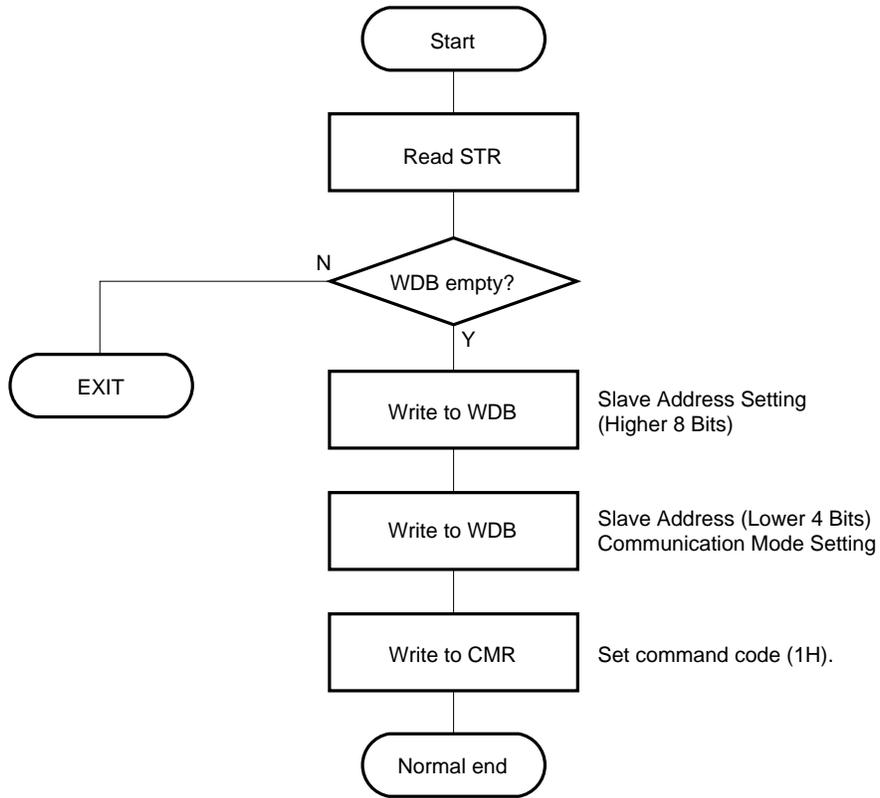


8.3.2 Communication control command processing routine

This processing routine sets the μPD6708 communication control commands SETSA, MREQ1, MREQ2, ABORT, GETSA, and SETREV.

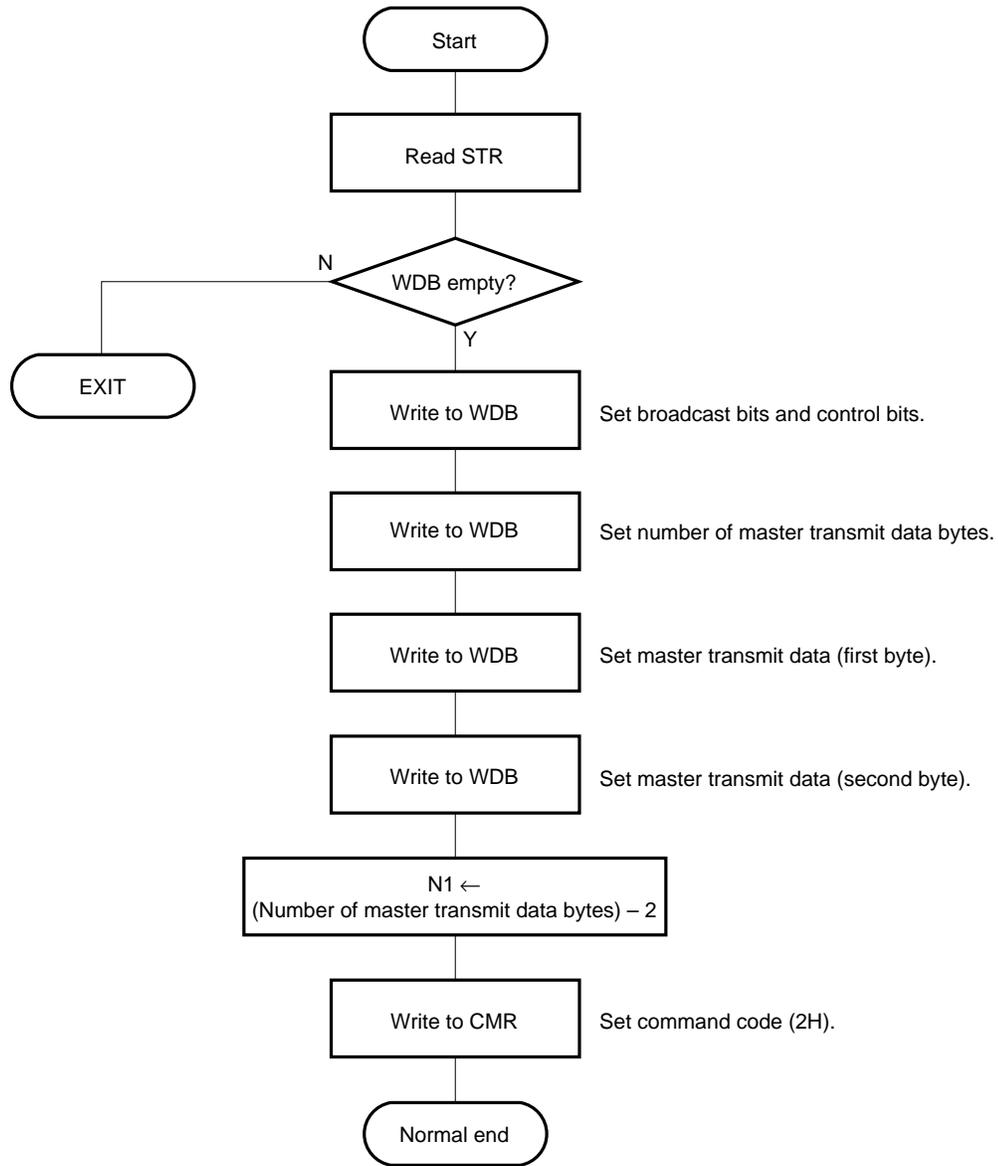
Processing starts when there is a command request during equipment control (Crq = 1)

(1) SETSA command



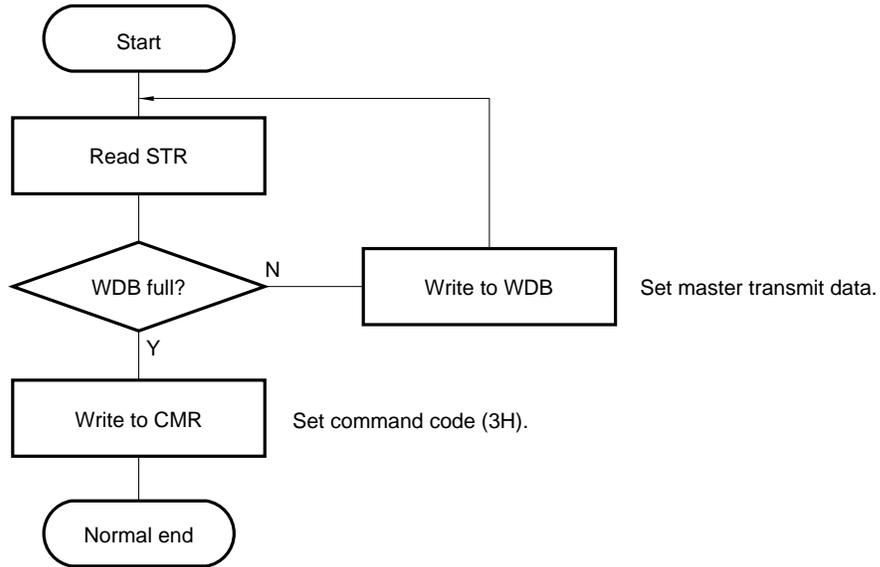
(2) MREQ1 command

When MSB of the control bits is 1 (master transmission)



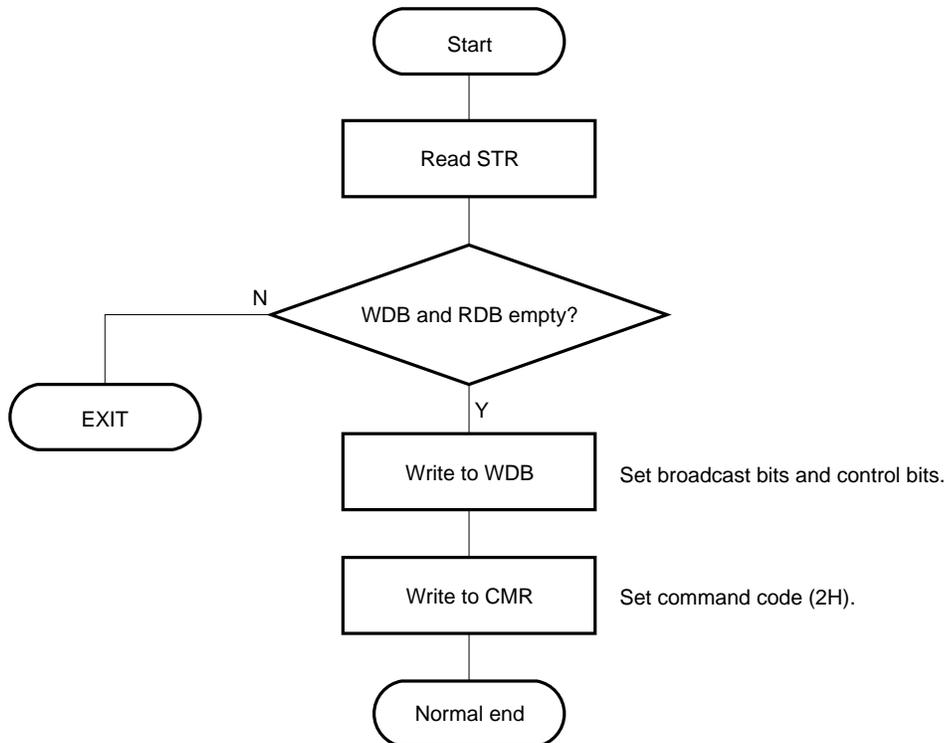
(3) MREQ2 command

After master transmission ends midway



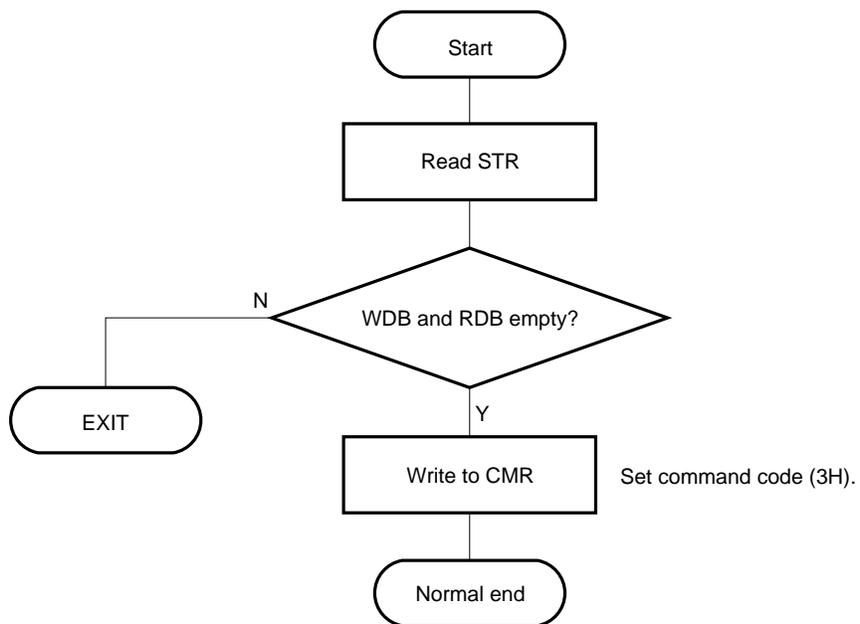
(4) MREQ1 command

When MSB of the control bits is 0 (master reception)

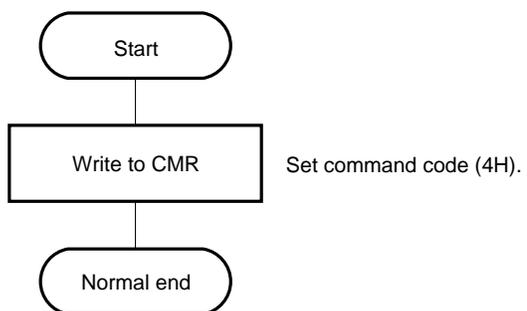


(5) MREQ2 command

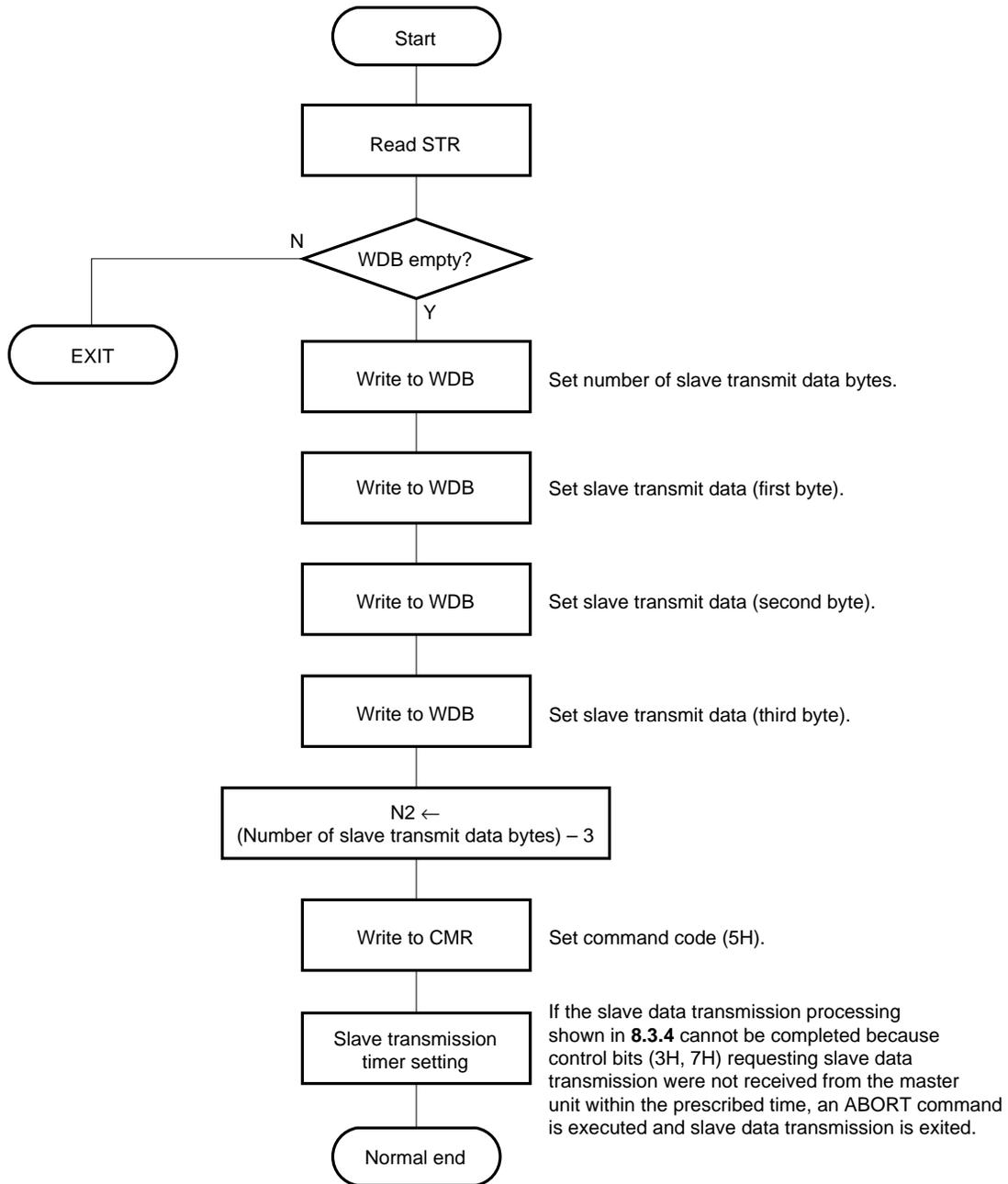
When master reception ends midway



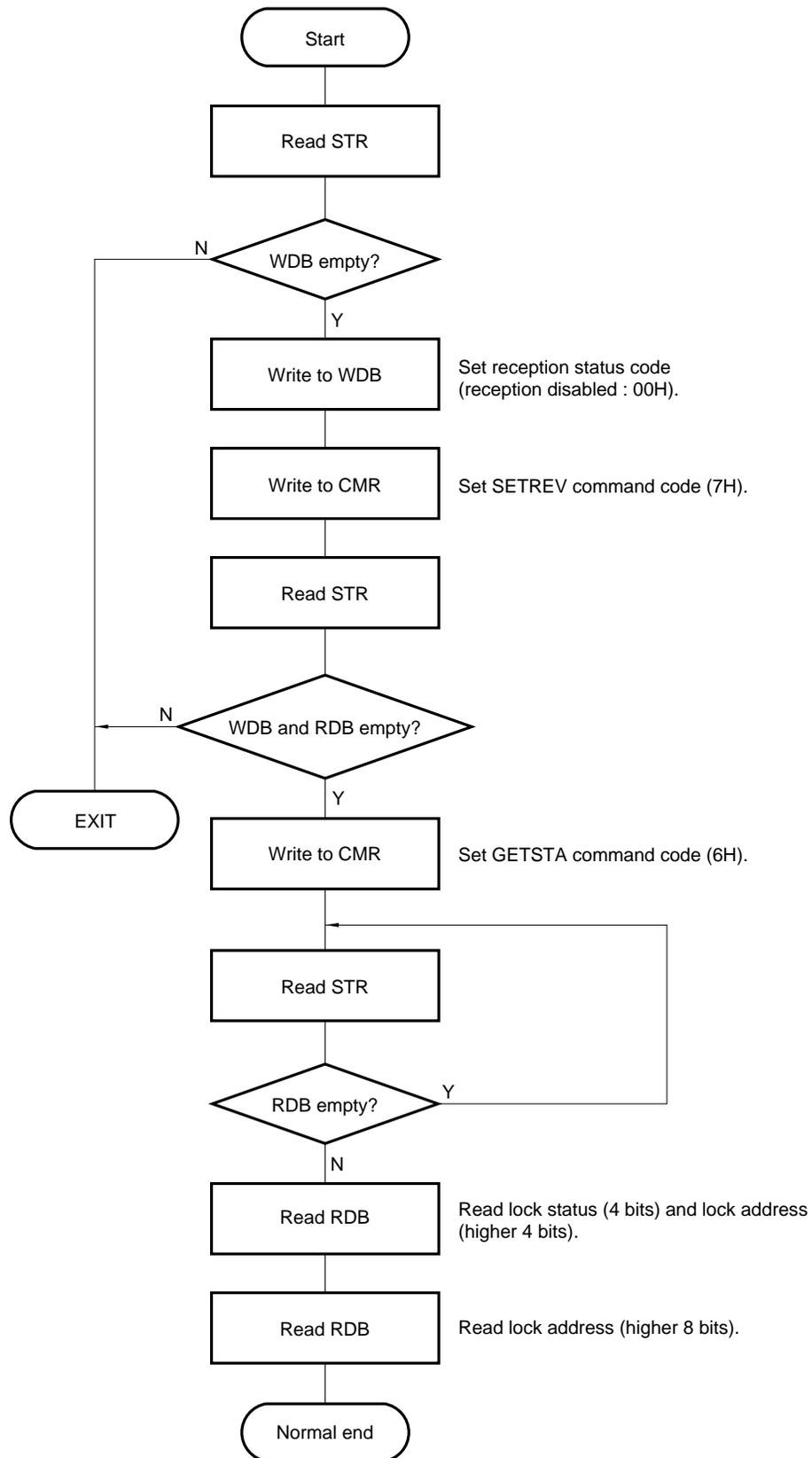
(6) ABORT command



(7) SETSD command



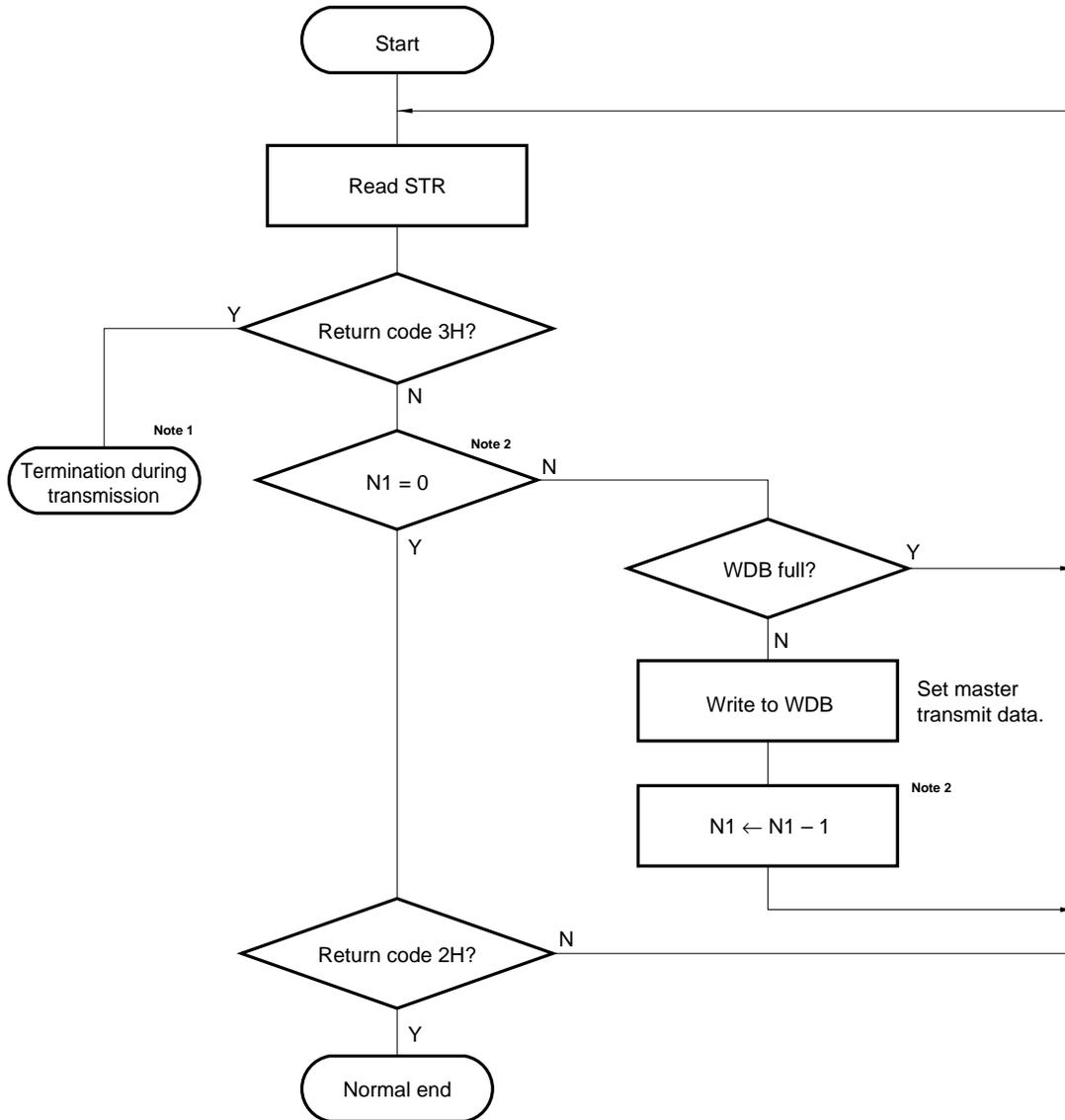
(8) GETSTA command



8.3.3 Master transmission processing routine

This processing routine is used when a unit becomes the master unit after executing the MREQ1 or MREQ2 command and transmits data and commands to a slave unit.

This routine is an interrupt service routine, and is executed when the return code (higher 2 bits = 00) is read in master transmission.

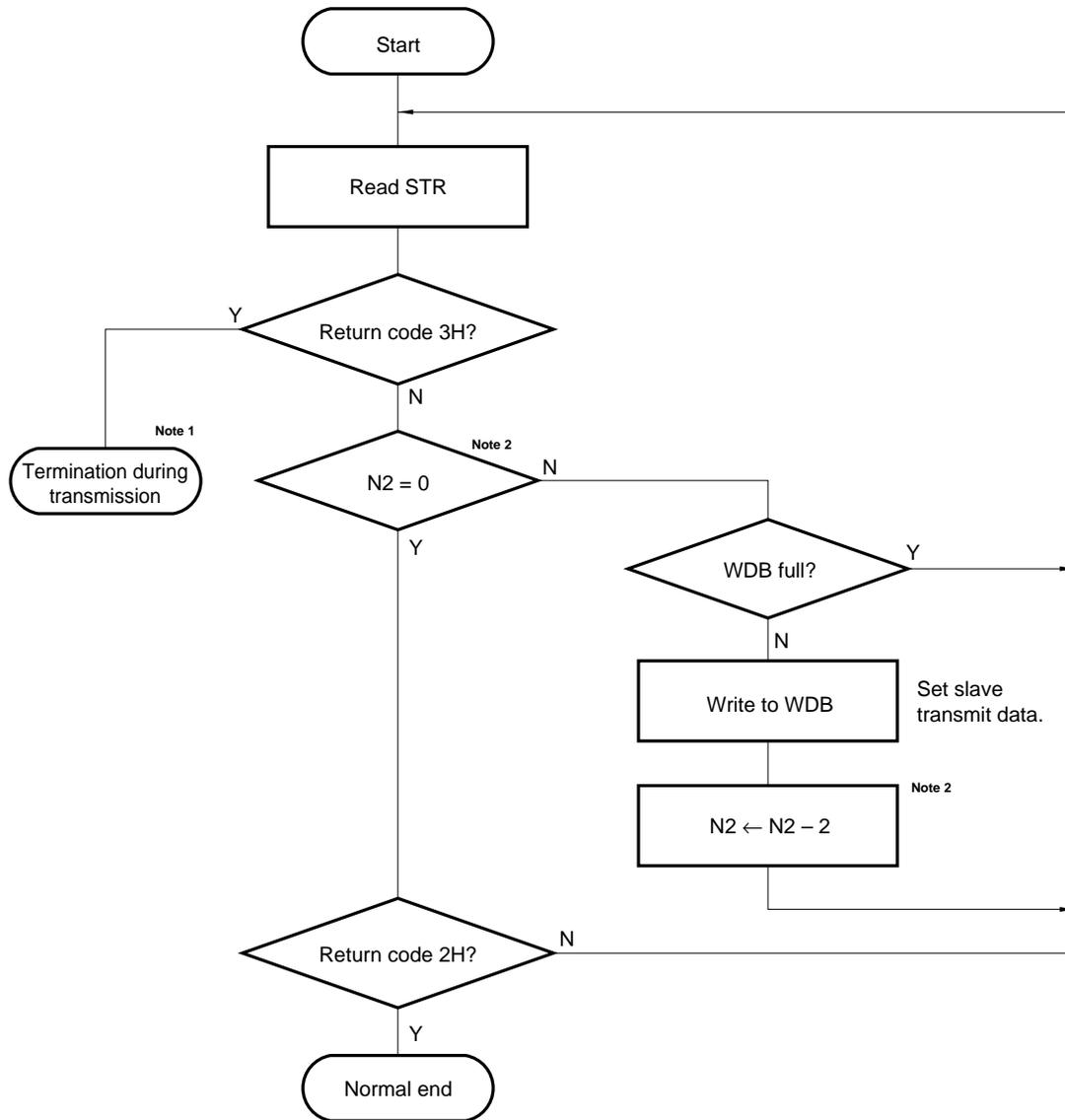


- Notes**
1. To transmit the remaining data, execute the MREQ2 command (see 8.3.2 (3)).
To abort master transmission, execute the ABORT command (see 8.3.2 (6)).
 2. N1: Number of master transmit data bytes.

8.3.4 Slave data transmission processing routine

This processing routine is used when a slave unit transmits data to the master unit after setting the SETSD command.

This routine is an interrupt service routine, and is executed when the return code (higher 2 bits = 00) is read in slave data transmission.



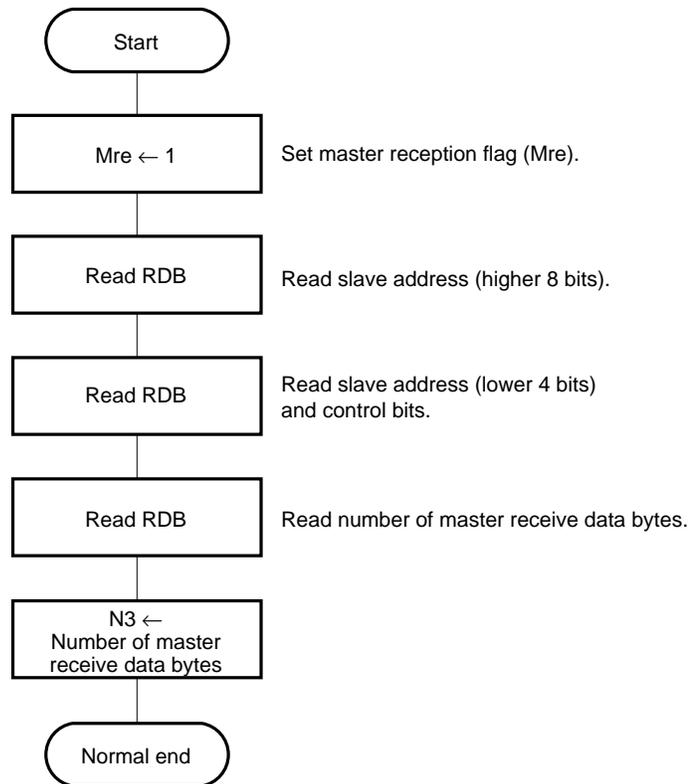
- Notes**
1. Untransmitted data in WDB is cleared by executing the ABORT command (see 8.3.2 (6)).
 2. N2: Number of slave transmit data bytes

8.3.5 Master reception processing routine

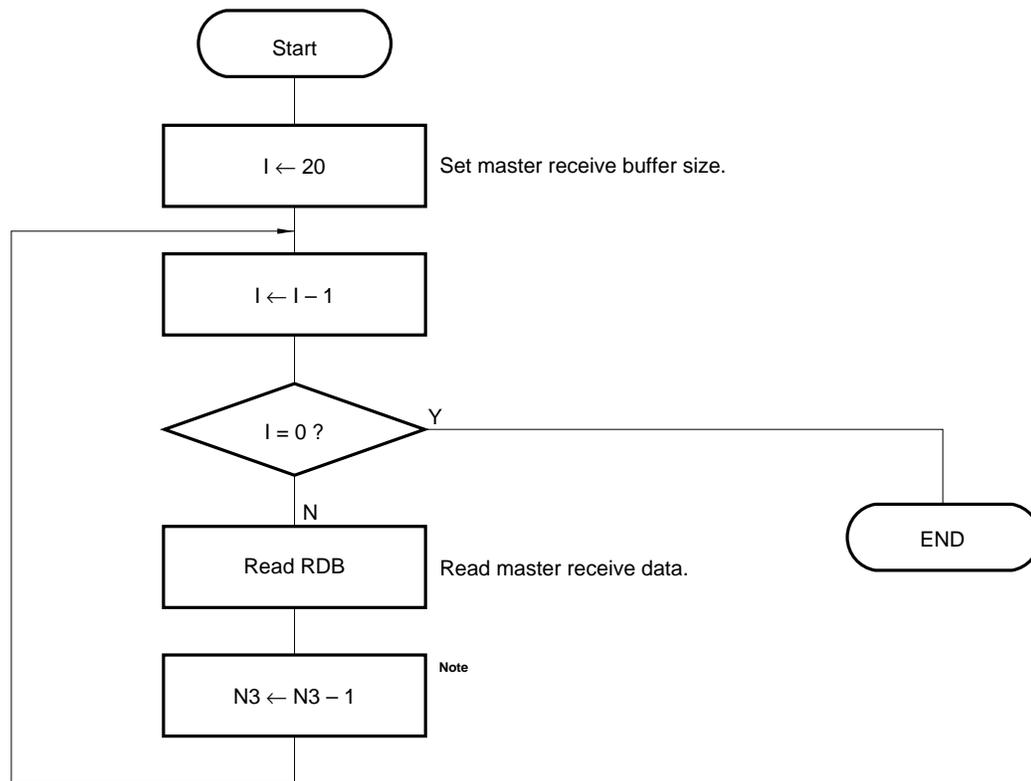
This processing routine is used when the master unit receive data, slave status and lock address after execution of the MREQ1 or MREQ2 command.

One of four different routines is executed depending on the contents of the return code.

(1) Processing in case of master reception start return code (4H)

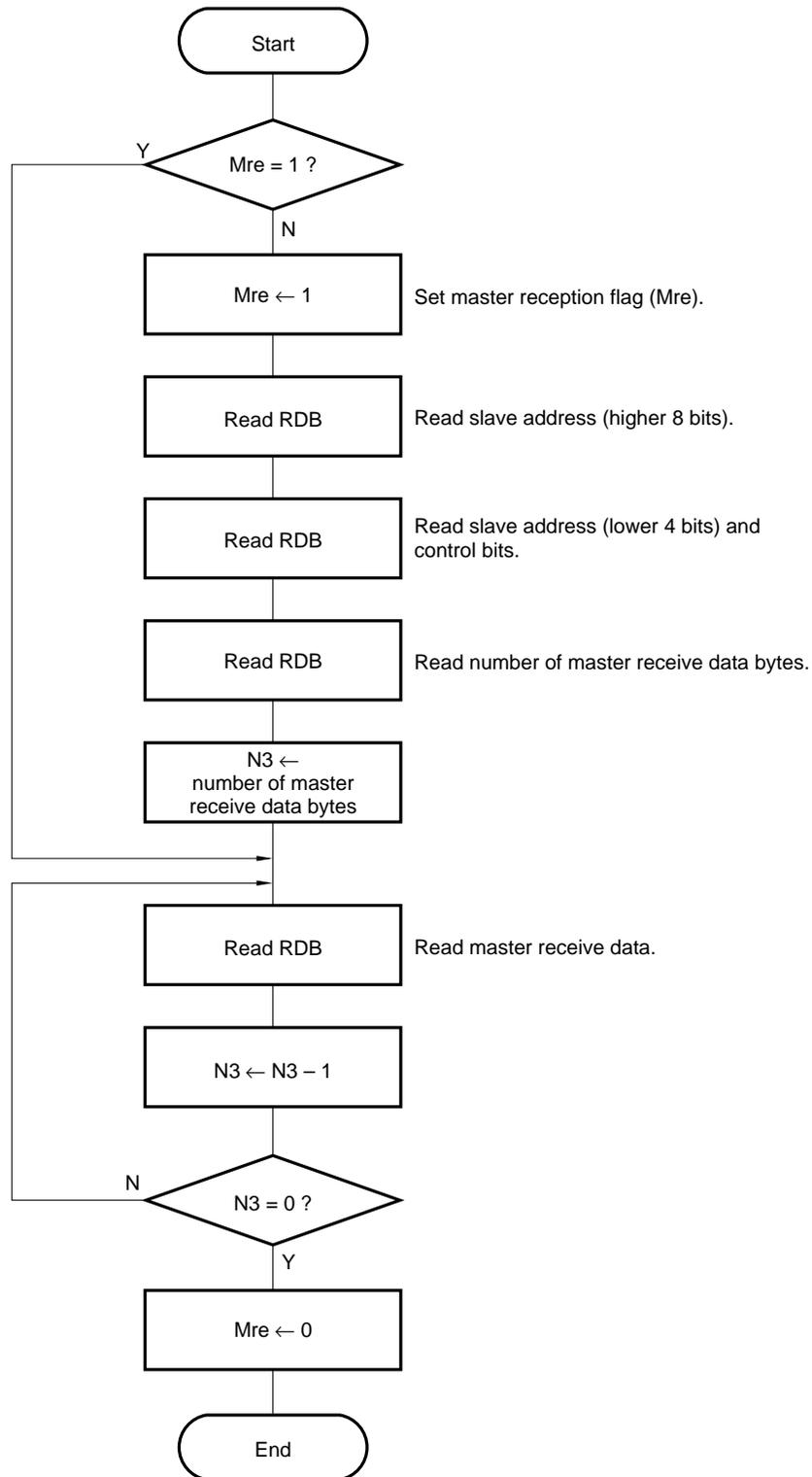


(2) Processing in case of master receive buffer full return code (5H)

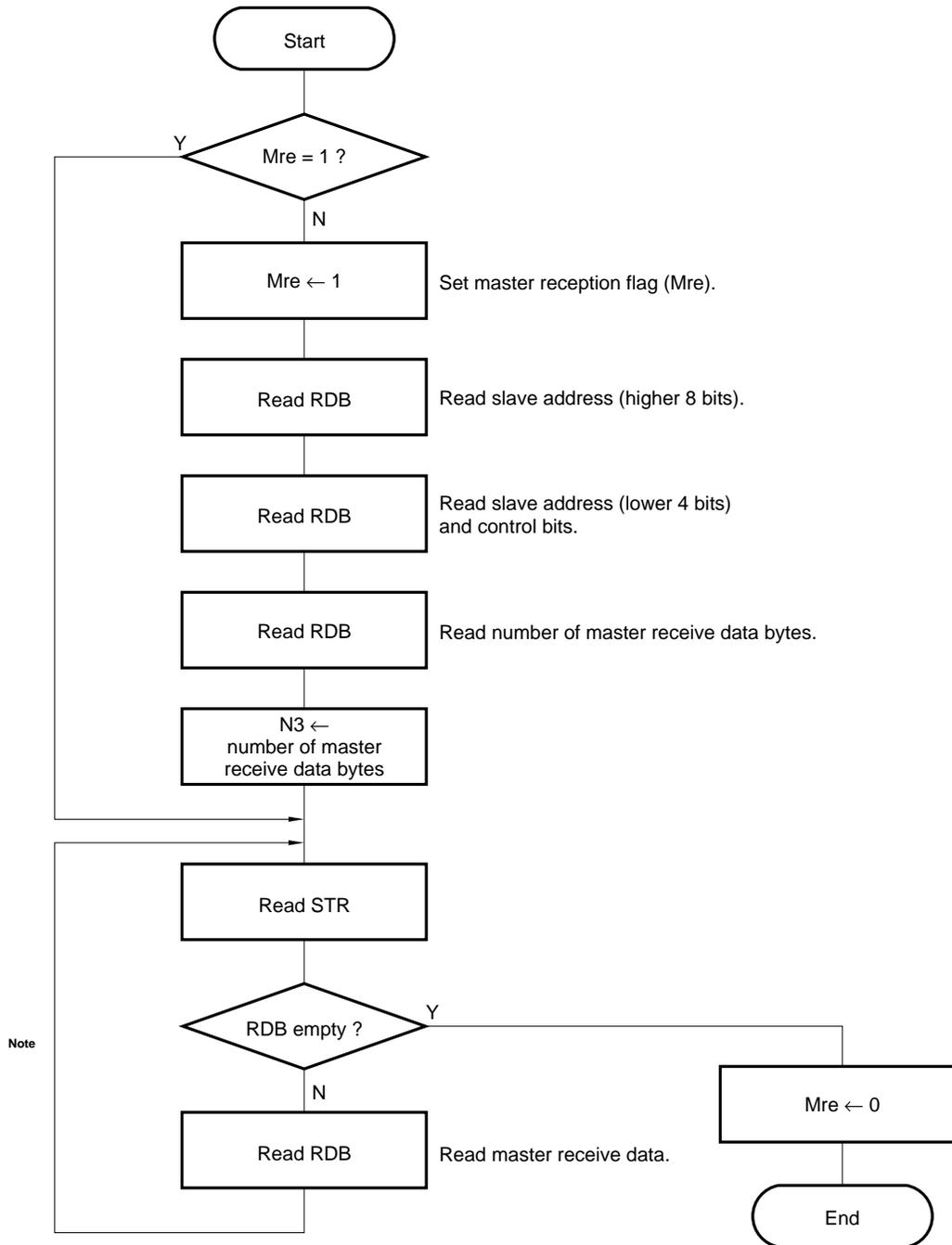


Note N3: Number of master receive data bytes

(3) Processing in case of master reception normal end return code (6H)



(4) Processing in case of end during master reception return code (7H)



Note To distinguish data from receive data (slave reception, broadcast reception) newly input in the next communication frame, after reading RDB (inputting one serial clock pulse to \overline{SCK} pin), STR should be read (inputting one serial clock pulse to \overline{SCK} pin) within the time shown below.

- When the next frame is mode 2 : Approx. 290 μs
- When the next frame is mode 1 : Approx. 400 μs
- When the next frame is mode 0 : Approx. 1580 μs

However, the above time limits for reading STR do not apply if this processing routine is executed in the reception disabled state after executing the SETREV command.

The flowcharts for slave reception processing and broadcast reception processing are the same as for master reception processing, and are therefore omitted here.

9. ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings (T_A = 25 °C)

PARAMETER	SYMBOL	TEST CONDITIONS	RATINGS	UNIT
Power supply	V _{DD} , AV _{DD}	V _{DD} - AV _{DD} < 0.5 V	-0.5 to +7.0	V
Logic input voltage	V _I	The pin without BUS+ and BUS-	-0.5 to V _{DD} + 0.3	V
Logic output voltage	V _O		-0.5 to V _{DD} + 0.3	V
Bus input voltage	V _{BI}	BUS+ and BUS-	-0.5 to + 6.0	V
Bus output voltage	V _{BO}		-0.5 to + 6.0	V
Operating ambient temperature	T _A		-40 to + 85	°C
Storage temperature	T _{stg}		-65 to + 150	°C

DC Characteristics (T_A = -40 to +85 °C, V_{DD} = AV_{DD} = 5 V ±10 %)

PARAMETER	SYMBOL	TEST CONDITIONS	MIN.	TYP.	MAX.	UNIT
Input voltage high	V _{IH}		0.8V _{DD}		V _{DD}	V
Input voltage low	V _{IL}		0		0.2V _{DD}	V
Output voltage high	V _{OH}	I _{OH} = -400 μA	0.7V _{DD}			V
Output voltage low	V _{OL}	I _{OL} = 2.5 mA			0.4	V
Input leakage current high	I _{LIH}	V _I = V _{DD}			10	μA
Input leakage current low	I _{LIL}	V _I = 0 V			-10	μA
Output leakage current high	I _{LOH}	V _O = V _{DD}			10	μA
Output leakage current low	I _{LOL}	V _O = 0 V			-10	μA
Supply current	I _{DD1}	Carrier sense		3.5	10	mA
	I _{DD2}	Reset mode		1.2	3	mA

Capacitance Characteristics (T_A = 25 °C, V_{DD} = AV_{DD} = 0 V)

PARAMETER	SYMBOL	TEST CONDITIONS	MIN.	TYP.	MAX.	UNIT
Input capacitance	C _I	f _c = 1 MHz,			15	pF
Input/output capacitance	C _{IO}	unmeasured pins returned to 0 V			15	pF

Recommended Ceramic Resonator (12 MHz)

Manufacturer	Product Name	External Capacities [pF]	
		C1	C2
Murata Mfg.	CST12.0MT <small>Note 1</small>	—	—
	CSA12.00MX241	30	30
Kyocera Corp.	KBR-12.0M <small>Note 2</small>	33	33

Recommended Crystal Resonator (12 MHz)

Manufacturer	Product Name	External Capacities [pF]	
		C1	C2
Kinseki, Ltd	HC-49/U	22	22

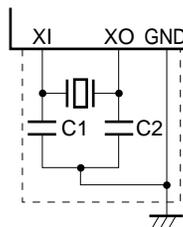
★ **Recommended Ceramic Resonator (12.58 MHz)**

Manufacturer	Product Name	External Capacities [pF]	
		C1	C2
Murata Mfg.	CSTCS12.5MTA 904 <small>Note 1</small>	—	—
TDK	FCR12.58M2S <small>Note 1</small>	33	33

- Notes**
1. Can only be used when communication mode 0 or 1 is used (frequency accuracy: ±1.5 %).
 2. This is a custom product, and therefore the manufacturer should be contacted directly.

★ **Caution** The oscillation circuit constants and oscillation voltage range shown above indicate the condition under which oscillation is stable and do not guarantee the oscillation frequency accuracy. If a high oscillation frequency accuracy is required from the actual circuit, the resonator mounted to the actual circuit must be adjusted. For details, directly consult the manufacturer of the resonator.

★ **External Circuit**



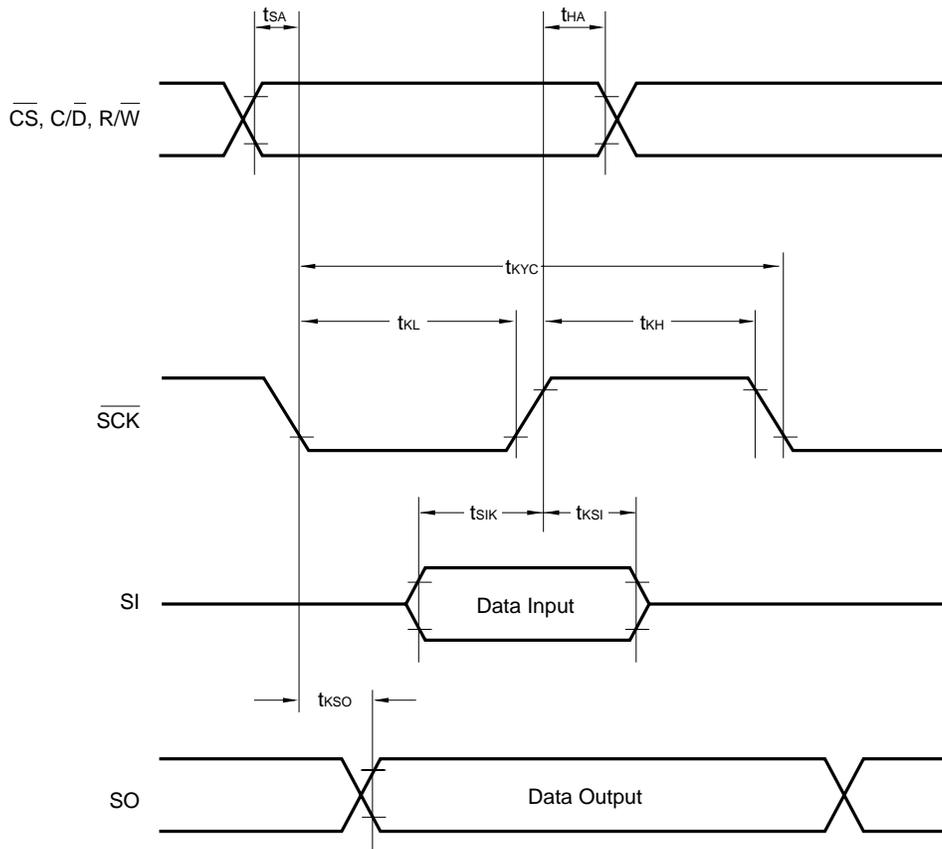
Caution Wire the dotted portion in the above figure as follows to prevent adverse influences of wiring capacitance when using the system clock oscillation circuit.

- Keep the wiring length as short as possible.
- Do not cross the wiring with any other signal lines.
- Do not place the wiring in the vicinity of a line through which a high alternating current flows.
- Always keep the ground point of the capacitor of the oscillation circuit at the same potential as Vss.
- Do not ground the wiring to a ground pattern through which a high current flows.
- Do not extract signals from the oscillation circuit.

AC Characteristics (T_A = -40 to +85 °C, V_{DD} = AV_{DD} = 5 V ±10 %)

PARAMETER	SYMBOL	TEST CONDITIONS	MIN.	TYP.	MAX.	UNIT	
System clock		f _x = 12 MHz	Using communication mode 2	11.94	12.00	12.06	MHz
			Using communication mode 0,1	11.82	12.00	12.18	MHz
		f _x = 12.58 MHz	Using communication mode 2	12.52	12.58	12.64	MHz
			Using communication mode 0,1	12.40	12.58	12.76	MHz
$\overline{\text{SCK}}$ cycle time	t _{KCY}		0.8			μs	
$\overline{\text{SCK}}$ high-level width	t _{KH}		0.4			μs	
$\overline{\text{SCK}}$ low-level width	t _{KL}		0.4			μs	
SI setup time (to $\overline{\text{SCK}}\uparrow$)	t _{SIK}		100			ns	
SI hold time (from $\overline{\text{SCK}}\uparrow$)	t _{KSI}		400			ns	
SO output delay time (from $\overline{\text{SCK}}\uparrow$)	t _{KSO}				300	ns	
$\overline{\text{CS}}$, C/ $\overline{\text{D}}$, R/ $\overline{\text{W}}$ setup time (to $\overline{\text{SCK}}\uparrow$)	t _{SA}		0			ns	
$\overline{\text{CS}}$, C/ $\overline{\text{D}}$, R/ $\overline{\text{W}}$ hold time (from $\overline{\text{SCK}}\uparrow$)	t _{HA}		400			ns	
$\overline{\text{CS}}$ high-level width			400			ns	
IRQ output high-level width			8		11	μs	
$\overline{\text{RESET}}$ low-level width			6			μs	
Oscillation stabilization time	t _{OS}	f _x = 12 MHz	20			ms	

Serial Transfer Timing



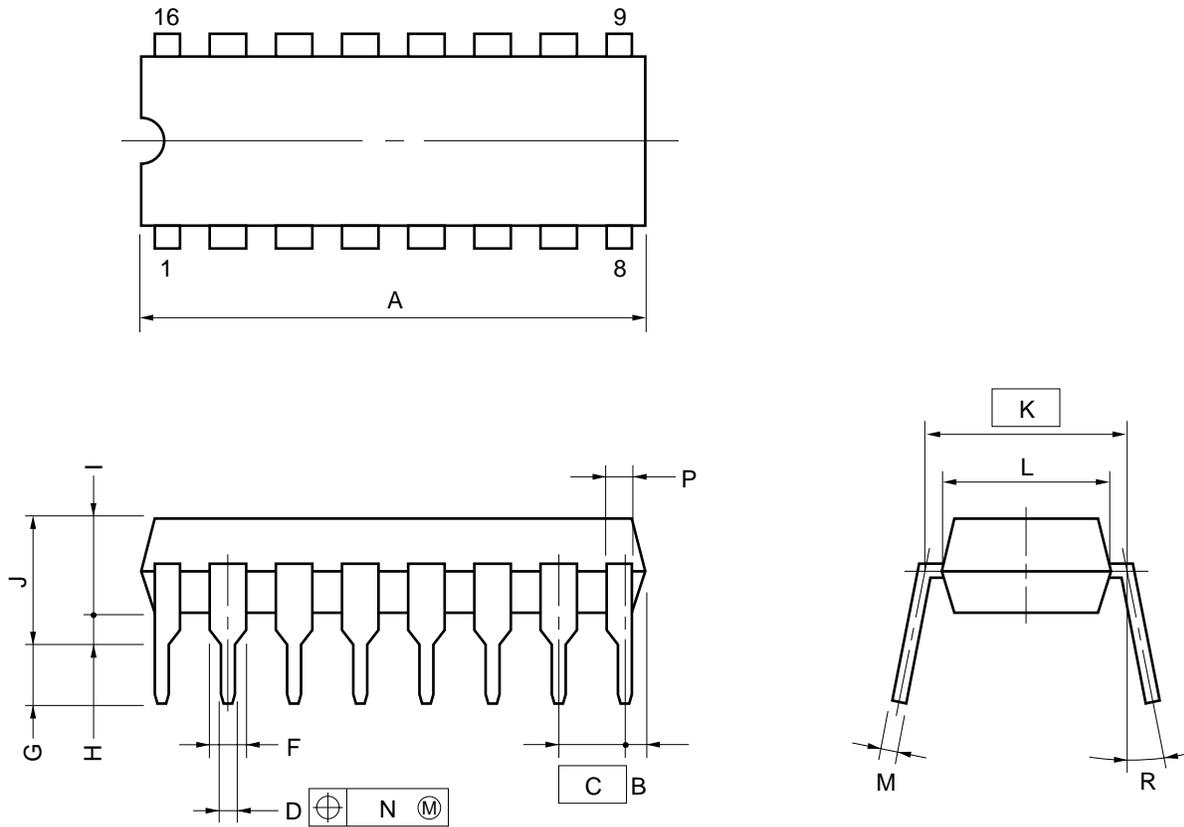
IEBus Driver/Receiver Characteristics (T_A = -40 to +85 °C, V_{DD} = AV_{DD} = 5 V ±10 %)

PARAMETER	SYMBOL	TEST CONDITIONS	MIN.	TYP.	MAX.	UNIT
Output current high	I _{OH}	R _L = 60 Ω ±5 %	- 2.73		- 6.22	mA
Output current low	I _{OL}				1.0	μA
In-phase output voltage	V _{OCOM}	When high and low	$\frac{V_{DD}}{2} - 0.25$	$\frac{V_{DD}}{2}$	$\frac{V_{DD}}{2} + 0.25$	V
Input voltage high	V _{IH}		120			mV
Input voltage low	V _{IL}				20.0	mV
Input hysteresis voltage	V _{IHYS}			25		mV
In-phase input voltage high	V _{IHCOM}		1.00		V _{DD} - 1.0	V
In-phase input voltage low	V _{ILCOM}		0		V _{DD}	V
Driver output resistance	R _O	Between BUS + and BUS -	100			kΩ
Driver output capacitance	C _O	Between BUS + and BUS -, Between BUS + and GND,			25	pF
Receiver input capacitance	C _I	Between BUS - and GND			25	pF

10. PACKAGE DRAWINGS

16 PIN PLASTIC DIP (300 mil)

★



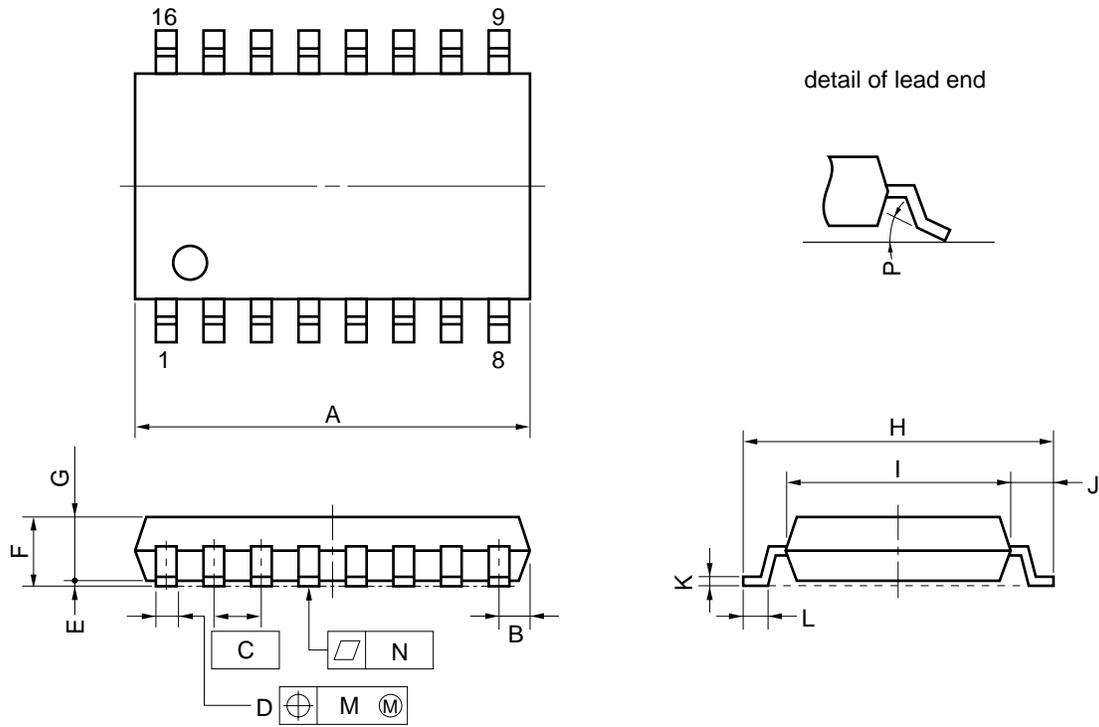
NOTES

- 1) Each lead centerline is located within 0.25 mm (0.01 inch) of its true position (T.P.) at maximum material condition.
- 2) Item "K" to center of leads when formed parallel.

ITEM	MILLIMETERS	INCHES
A	20.32 MAX.	0.800 MAX.
B	1.27 MAX.	0.050 MAX.
C	2.54 (T.P.)	0.100 (T.P.)
D	0.50±0.10	0.020 ^{+0.004} _{-0.005}
F	1.1 MIN.	0.043 MIN.
G	3.5±0.3	0.138±0.012
H	0.51 MIN.	0.020 MIN.
I	4.31 MAX.	0.170 MAX.
J	5.08 MAX.	0.200 MAX.
K	7.62 (T.P.)	0.300 (T.P.)
L	6.5	0.256
M	0.25 ^{+0.10} _{-0.05}	0.010 ^{+0.004} _{-0.003}
N	0.25	0.01
P	1.1 MIN.	0.043 MIN.
R	0~15°	0~15°

P16C-100-300B-1

16 PIN PLASTIC SOP (300 mil)



NOTE

Each lead centerline is located within 0.12 mm (0.005 inch) of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS	INCHES
A	10.46 MAX.	0.412 MAX.
B	0.78 MAX.	0.031 MAX.
C	1.27 (T.P.)	0.050 (T.P.)
D	0.40 ^{+0.10} _{-0.05}	0.016 ^{+0.004} _{-0.003}
E	0.1±0.1	0.004±0.004
F	1.8 MAX.	0.071 MAX.
G	1.55	0.061
H	7.7±0.3	0.303±0.012
I	5.6	0.220
J	1.1	0.043
K	0.20 ^{+0.10} _{-0.05}	0.008 ^{+0.004} _{-0.002}
L	0.6±0.2	0.024 ^{+0.008} _{-0.009}
M	0.12	0.005
N	0.10	0.004
P	3° ^{+7°} _{-3°}	3° ^{+7°} _{-3°}

P16GM-50-300B-4

11. RECOMMENDED SOLDERING CONDITIONS

This product should be soldered and mounted under the conditions recommended in the table below.

For details of recommended soldering conditions, refer to the information document **Semiconductor Device Mounting Technology Manual (IEI-1207)**.

For soldering methods and conditions other than those recommended, please contact NEC representative.

Table 11-1. Surface Mount Type Soldering Conditions

μPD6708GS : 16-pin plastic SOP (300 mil)

Soldering Method	Soldering Conditions	Symbol
Infrared ray reflow	Package peak temperature: 230 °C, Reflow time: 30 seconds or less (at 210 °C or higher), Number of reflow processes: 1	IR30-00-1
VPS reflow	Package peak temperature: 215 °C, Reflow time: 40 seconds or less (at 200 °C or higher), Number of reflow processes: 1	VP15-00-1
Partial heating	Pin temperature: 300 °C or below, Flow time: 3 seconds or less (per side of device)	—

Caution Use of more than one soldering method should be avoided (except for partial heating).

Table 11-2. Hole-Through Type Soldering Conditions

μPD6708CX : 16-pin plastic DIP (300mil)

Soldering Method	Soldering Conditions
Wave soldering (pin only)	Solder temperature: 260 °C or below, Flow time: 10 seconds or less
Partial heating	Pin temperature: 300 °C or below, Flow time: 3 seconds or less (per pin)

★

Caution Wave soldering is used on the pin only, and care must be taken to prevent solder from coming into direct contact with the package body.

★ APPENDIX MAJOR DIFFERENCES BETWEEN μPD6708 AND μPD72042A, μPD72042B

Part Number		μPD6708	μPD72042A	μPD72042B
Parameter				
Oscillation frequency (f _x)		12 MHz	6 MHz	
Supply voltage (V _{DD})		5 V ±10 %		
Operating ambient temperature (T _A)		-40 to +85 °C		
IEBus	Communication mode	Modes 0, 1, and 2		
	Driver/receiver	Provided		
	External protection resistor	Not necessary	Necessary (connect 180 Ω in series to BOS+ and BOS-)	
	Transmit buffer	4 Bytes	33 Bytes	
	Receive buffer	20 Bytes	40 Bytes	
Interfacing with microcontroller ^{Note}		Serial interface (3-wire)	Serial interface (3-wire/2-wire)	
		MSB first	MSB first	LSB first
Package		16-pin SOP (300 mil)	16-pin SOP (375 mil)	
		16-pin DIP (300 mil)		

Note Setting commands/data and related pins differs between the μPD6708 and μPD72042A, μPD72042B.

NOTES FOR CMOS DEVICES

① PRECAUTION AGAINST ESD FOR SEMICONDUCTORS

Note: Strong electric field, when exposed to a MOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop generation of static electricity as much as possible, and quickly dissipate it once, when it has occurred. Environmental control must be adequate. When it is dry, humidifier should be used. It is recommended to avoid using insulators that easily build static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work bench and floor should be grounded. The operator should be grounded using wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions need to be taken for PW boards with semiconductor devices on it.

② HANDLING OF UNUSED INPUT PINS FOR CMOS

Note: No connection for CMOS device inputs can be cause of malfunction. If no connection is provided to the input pins, it is possible that an internal input level may be generated due to noise, etc., hence causing malfunction. CMOS device behave differently than Bipolar or NMOS devices. Input levels of CMOS devices must be fixed high or low by using a pull-up or pull-down circuitry. Each unused pin should be connected to V_{DD} or GND with a resistor, if it is considered to have a possibility of being an output pin. All handling related to the unused pins must be judged device by device and related specifications governing the devices.

③ STATUS BEFORE INITIALIZATION OF MOS DEVICES

Note: Power-on does not necessarily define initial status of MOS device. Production process of MOS does not define the initial operation status of the device. Immediately after the power source is turned ON, the devices with reset function have not yet been initialized. Hence, power-on does not guarantee out-pin levels, I/O settings or contents of registers. Device is not initialized until the reset signal is received. Reset operation must be executed immediately after power-on for devices having reset function.

IEBus and Inter Equipment Bus are trademarks of NEC Corporation.

The export of this product from Japan is prohibited without governmental license. To export or re-export this product from a country other than Japan may also be prohibited without a license from that country. Please call an NEC sales representative.

No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Corporation. NEC Corporation assumes no responsibility for any errors which may appear in this document.

NEC Corporation does not assume any liability for infringement of patents, copyrights or other intellectual property rights of third parties by or arising from use of a device described herein or any other liability arising from use of such device. No license, either express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC Corporation or others.

While NEC Corporation has been making continuous effort to enhance the reliability of its semiconductor devices, the possibility of defects cannot be eliminated entirely. To minimize risks of damage or injury to persons or property arising from a defect in an NEC semiconductor device, customer must incorporate sufficient safety measures in its design, such as redundancy, fire-containment, and anti-failure features.

NEC devices are classified into the following three quality grades:

“Standard”, “Special”, and “Specific”. The Specific quality grade applies only to devices developed based on a customer designated “quality assurance program” for a specific application. The recommended applications of a device depend on its quality grade, as indicated below. Customers must check the quality grade of each device before using it in a particular application.

Standard: Computers, office equipment, communications equipment, test and measurement equipment, audio and visual equipment, home electronic appliances, machine tools, personal electronic equipment and industrial robots

Special: Transportation equipment (automobiles, trains, ships, etc.), traffic control systems, anti-disaster systems, anti-crime systems, safety equipment and medical equipment (not specifically designed for life support)

Specific: Aircrafts, aerospace equipment, submersible repeaters, nuclear reactor control systems, life support systems or medical equipment for life support, etc.

The quality grade of NEC devices in “Standard” unless otherwise specified in NEC's Data Sheets or Data Books. If customers intend to use NEC devices for applications other than those specified for Standard quality grade, they should contact NEC Sales Representative in advance.

Anti-radioactive design is not implemented in this product.