# SH7450 Group, SH7451 Group

## CAN Application Note

R01AN0431EJ0100
Rev. 1.00
Feb. 23, 2012


### Abstract

This document describes the procedures for CAN communication using the SH7450 Group, SH7451 Group.


### Products

SH7450 Group, SH7451 Group

Variables in this document are as follows:
- i: CAN channel number (i = 0 to 4)
- j: Mailbox number (j = 0 to 63)
- k: Mask register number (k = 0 to 9)

When using this application note with other Renesas MCUs, careful evaluation is recommended after making modifications to comply with the alternate MCU.

## Contents

# 1. Initial Settings

The following settings*1 are required for CAN communication:

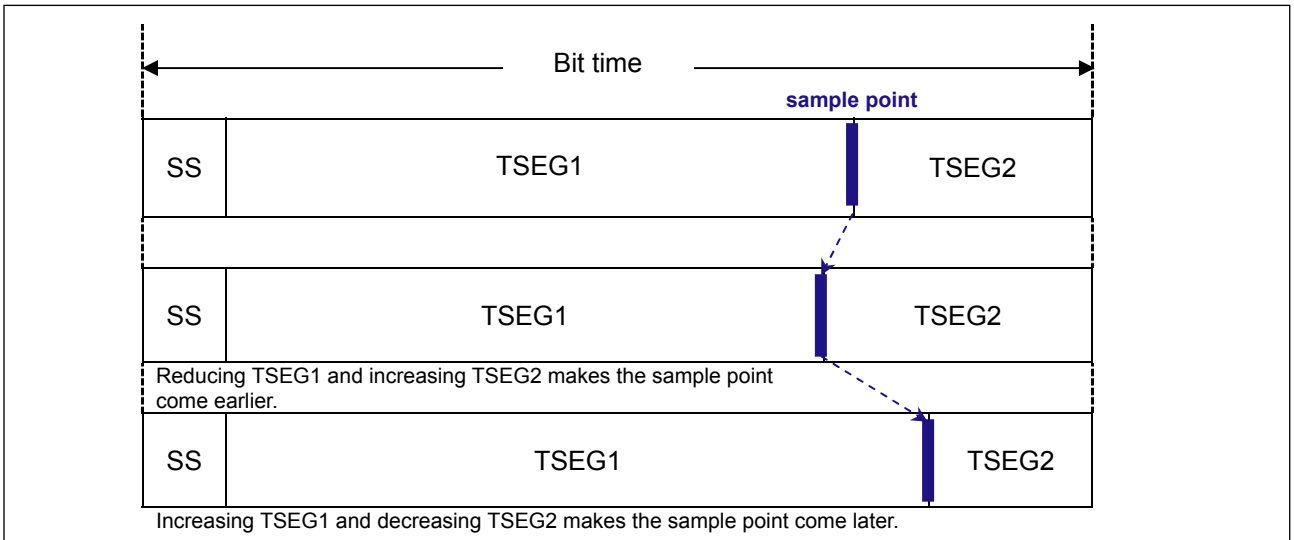
Bit timing configuration (refer to section 1.1 Bit Timing)
Clock setting (refer to section 1.2 Transfer Speed)
Baud rate setting (refer to section 1.2 Transfer Speed)

Note: *1: When using the acceptance filter, refer to section 7.2 Acceptance Filter.

## 1.1 Bit Timing

The bit timing configuration consists of three segments. Figure 1.1 shows the segment configuration and the sample point, table 1.1 lists bit timing configuration examples, and figure 1.2 shows bit timing configuration examples when the sample point is 75%.

Of the three segments, time segment 1 (hereinafter referred to as TSEG1) and time segment 2 (hereinafter referred to as TSEG2) specify the sample point. The sampling timing can be adjusted by changing the values of TSEG1 and TSEG2. The minimum unit in which timing settings are specified is called a Time Quantum (Tq). The Tq is determined by the clock frequency input to the CAN module and the baud rate prescaler division value.




The range of each segment:

- Bit time = 8 Tq to 25 Tq
- SS = 1 Tq
- TSEG1 = 4 Tq to 16 Tq
- TSEG2 = 2 Tq to 8 Tq
- SJW = 1 Tq to 4 Tq

Restrictions of TSEG1 and TSEG2 settings

- TSEG1 > TSEG2 ≥ SJW (but when SJW = 1, TSEG2 ≥ 2)

**Figure 1.1 Bit Time Segment Configuration and Sample Point**

(1) SS: Synchronization Segment
This segment is used for synchronization by monitoring the edge transition from recessive to dominant within the interframe space*1.

(2) TSEG1: Time Segment 1
This segment absorbs the physical delay inherent in the CAN network and compensates for phase error*2 that occurs during resynchronization. The physical delay inherent in the network is equal to two times the sum of the bus delay, the input comparator delay, and the output driver delay.

(3) TSEG2: Time Segment 2
This segment compensates for phase error*2 that occurs during resynchronization.

(4) SJW: Resynchronization Jump Width
This is the maximum width of compensation to correct synchronization misalignment due to phase error*2.

Notes: *1: The interframe space is comprised of intermission, suspend transmission, and bus idle. All nodes can start transmission during the bus idle state.

*2: Phase error refers to synchronization misalignment between nodes, during message transmission or reception, due to factors such as a shift in the oscillator frequency or delay in the signal transfer path.

**Table 1.1 Bit Timing Configuration Examples**

| 1 Bit Time | Setting Values (Tq) | | | Sample Point*1(%) |
|---|---|---|---|---|
| | SS | TSEG1 | TSEG2 | |
| 8 Tq | 1 | 4 | 3 | 62.50 |
| | 1 | 5 | 2 | 75.00 |
| 10 Tq | 1 | 6 | 3 | 70.00 |
| | 1 | 7 | 2 | 80.00 |
| 12 Tq | 1 | 8 | 3 | 75.00 |
| | 1 | 9 | 2 | 83.33 |
| 15 Tq | 1 | 10 | 4 | 73.33 |
| | 1 | 11 | 3 | 80.00 |
| 16 Tq | 1 | 10 | 5 | 68.75 |
| | 1 | 11 | 4 | 75.00 |
| 20 Tq | 1 | 12 | 7 | 65.00 |
| | 1 | 13 | 6 | 70.00 |
| 24 Tq | 1 | 15 | 8 | 66.66 |

Note: *1: The point at which the level of bit time is determined.

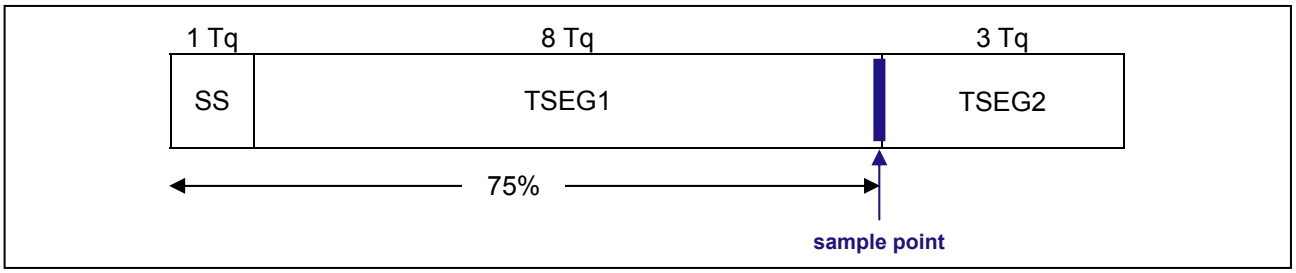



**Figure 1.2 Bit Timing Configuration Examples When the Sample Point is 75% of the Bit Time**

## 1.2 Transfer Speed

The transfer speed is determined by fCAN, the baud rate prescaler division value, and the number of Tq of one bit time. Figure 1.3 is a block diagram of the CAN clock generator. Table 1.2 lists the formula for calculating the primary transfer speeds and actual examples.

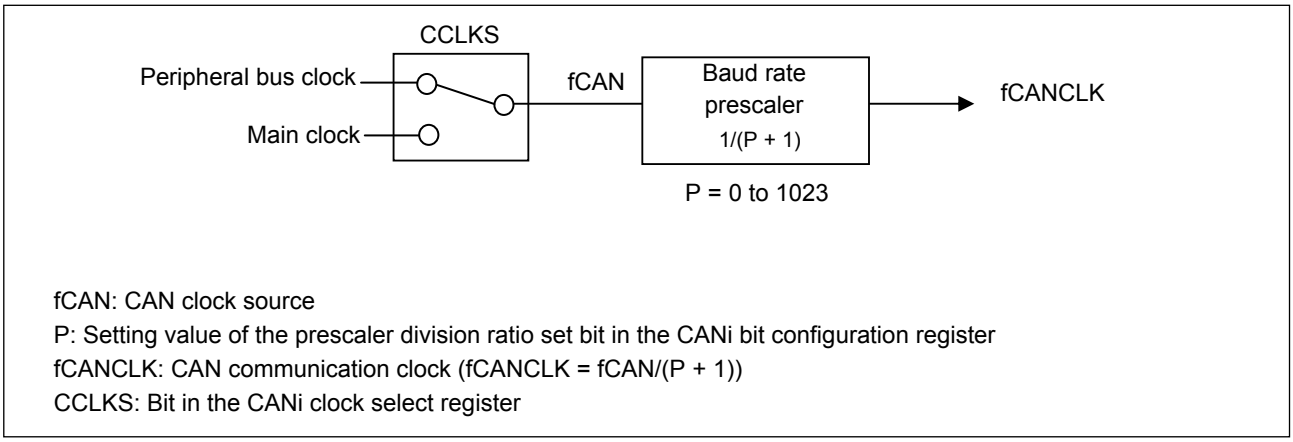



**Figure 1.3 Block Diagram of the CAN Clock Generator**

**Table 1.2 Formula for Calculating Primary Transfer Speeds and Actual Examples**

**Formula for calculating transfer speed**

$$\frac{\text{fCAN}}{\text{Baud rate prescaler division value}^{*1} \times \text{number of Tq of one bit time}}$$

| **fCAN** | **40MHz** | | **32MHz** | | **20MHz** | | **16MHz** | |
|---|---|---|---|---|---|---|---|---|
| **Bit-rate** | **No. of Tq** | **P + 1**[*1] | **No. of Tq** | **P + 1**[*1] | **No. of Tq** | **P + 1**[*1] | **No. of Tq** | **P + 1**[*1] |
| 1 Mbps | 10Tq | 4 | 8Tq | 4 | 10Tq | 2 | 8Tq | 2 |
| | 20Tq | 2 | 16Tq | 2 | 20Tq | 1 | 16Tq | 1 |
| 500 kbps | 10Tq | 8 | 8Tq | 8 | 10Tq | 4 | 8Tq | 4 |
| | 20Tq | 4 | 16Tq | 4 | 20Tq | 2 | 16Tq | 2 |
| 250 kbps | 10Tq | 16 | 8Tq | 16 | 10Tq | 8 | 8Tq | 8 |
| | 20Tq | 8 | 16Tq | 8 | 20Tq | 4 | 16Tq | 4 |
| 83.3 kbps | 8Tq | 60 | 8Tq | 48 | 8Tq | 30 | 8Tq | 24 |
| | 10Tq | 48 | 16Tq | 24 | 10Tq | 24 | 16Tq | 12 |
| | 16Tq | 30 | | | 16Tq | 15 | | |
| | 20Tq | 24 | | | 20Tq | 12 | | |
| 33.3 kbps | 8Tq | 150 | 8Tq | 120 | 8Tq | 75 | 8Tq | 60 |
| | 10Tq | 120 | 10Tq | 96 | 10Tq | 60 | 10Tq | 48 |
| | 20Tq | 60 | 16Tq | 60 | 20Tq | 30 | 16Tq | 30 |
| | | | 20Tq | 48 | | | 20Tq | 24 |

Note: *1: Division value of the baud rate prescaler = P + 1 (P = 0 to 1023)
P: Setting value of the prescaler division ratio set bit in the CANi bit configuration register

## 1.3 CAN Bit Timing and Transfer Speed Settings

Figure 1.4 shows the procedure for setting the CAN bit timing and transfer speed. These settings must be performed during CAN configuration. See section 2.1 for details on the CAN configuration procedure.

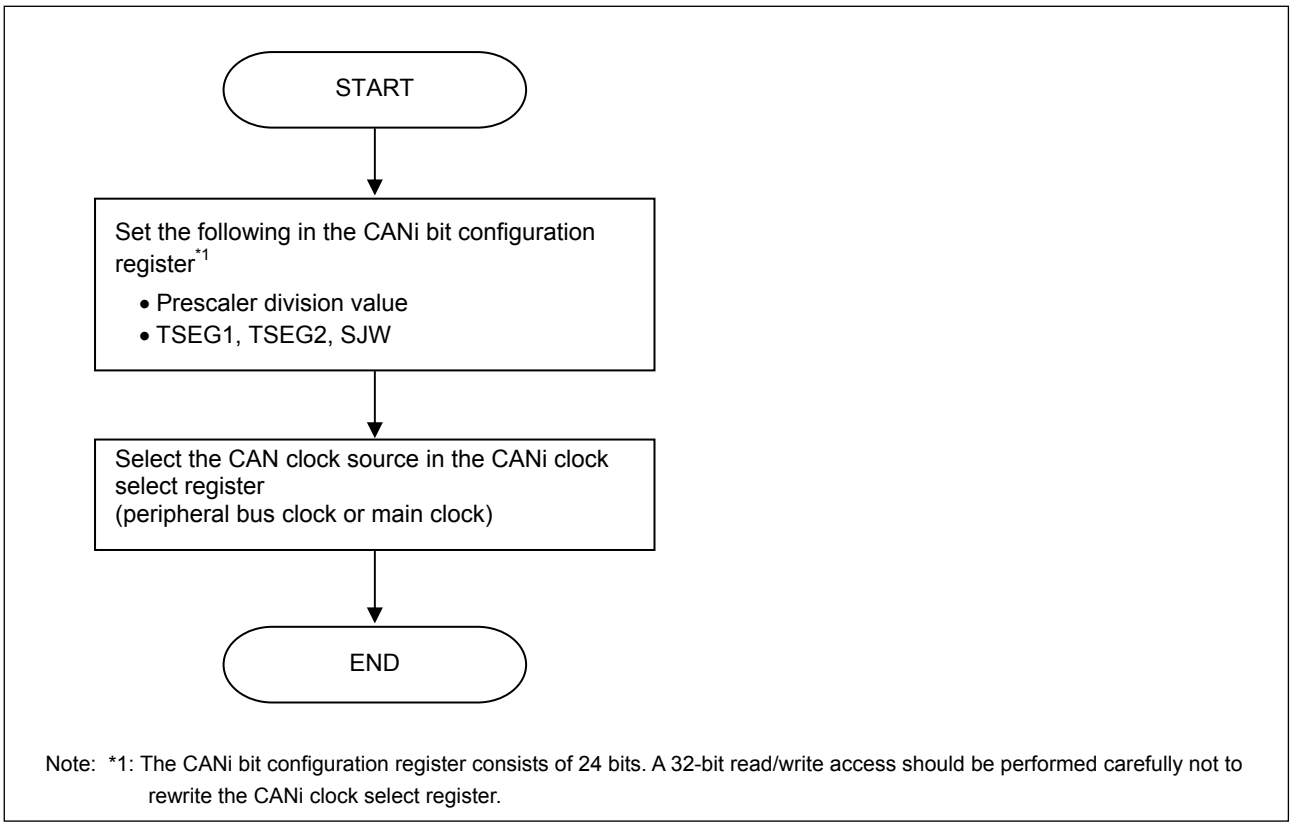



**Figure 1.4 Bit Timing Configuration and Transfer Speed Setting**

## 2. Transmission and Reception of CAN Messages

The three procedures listed below are used to transmit and receive CAN messages. These procedures are used in normal mailbox mode (CAN mailbox mode select bit (MBM) is "0"). For details on FIFO mailbox mode (CAN mailbox mode select bit (MBM) is "1"), see section 3.

(1) CAN Configuration Procedure
During CAN configuration, CAN transfer speed, control mode, acceptance filter, and interrupt settings are made.

(2) Mailbox Configuration Procedure
The mailbox settings for transmission and reception modes are made in the CANi message control register j (CiMCTLj) corresponding to each mailbox. Table 2.1 lists the correspondences between the CANi message control register j (CiMCTLj) settings and the transmission/reception modes.

(3) Data Processing Procedure
The message processing is taken place after message transmission or message reception is completed successfully.

**Table 2.1 Correspondences of CANi Message Control Register j Settings and Transmission/Reception Modes**

| TRMREQ*1 | RECREQ*1 | ONESHOT*1 | Mailbox Transmission/Reception Mode Setting |
|---|---|---|---|
| 0 | 0 | 0 | Mailbox disabled or transmission being aborted |
| 0 | 0 | 1 | Configurable only when transmission or reception from a mailbox (programmed in one-shot mode) is aborted. |
| 0 | 1 | 0 | Configured as a receive mailbox for a data frame or a remote frame. |
| 0 | 1 | 1 | Configured as a one-shot receive mailbox for a data frame or a remote frame. |
| 1 | 0 | 0 | Configured as a transmit mailbox for a data frame or a remote frame. |
| 1 | 0 | 1 | Configured as a one-shot transmit mailbox for a data frame or a remote frame. |
| 1 | 1 | 0 | Do not set. |
| 1 | 1 | 1 | Do not set. |

Note: *1: Bits in CANi message control register j

When a mailbox is configured as a receive mailbox or a one-shot receive mailbox, note the following:

(1) Before a mailbox is configured as a receive mailbox or a one-shot receive mailbox, set the CANi message control register j (CiMCTLj) to "H'00".

(2) A received message is stored into the first mailbox that matches the condition according to the result of receive mode configuration and acceptance filtering. Upon deciding a mailbox which stores the received message, the mailbox with the smaller number has higher priority.

(3) In CAN operation mode, when a CAN module transmits a message whose ID matches with the ID/mask set of a mailbox configured to receive messages, the CAN module never receives the transmitted data. In self-test mode, however, the CAN module may receive its transmitted data. In this case, the CAN module sends an ACK.

When configuring a mailbox as a transmit mailbox or a one-shot transmit mailbox, note the following:

(1) Before a mailbox is configured as a transmit mailbox or one-shot transmit mailbox, ensure that the CANi message control register j (CiMCTLj) is "H'00" and that there is no pending abort process.

## 2.1 CAN Configuration

CAN configuration is comprised of the following three configurations:

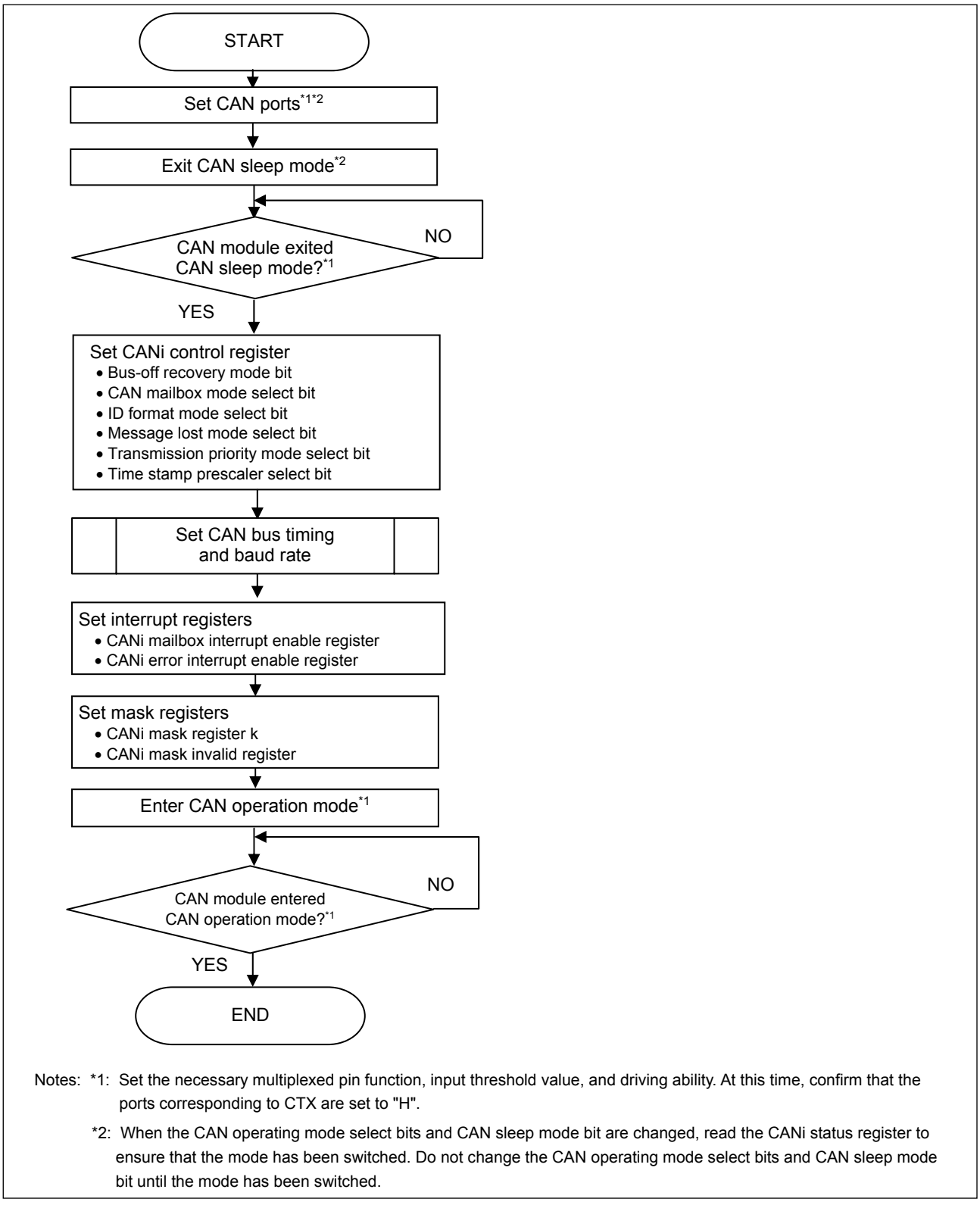
(1) Configuration after a Hardware Reset
This configuration is used after a hardware reset.

(2) Configuration after CAN Reset Mode
This configuration is used after entering CAN reset mode.
The CAN module is reset, so it must be reconfigured.
The configuration for this mode must be used if the transfer speed must be changed.

(3) Configuration after CAN Halt Mode
This configuration is used after entering CAN halt mode.
The CAN module is not reset, so it does not need to be reconfigured.
The configuration for this mode must be used if communication must be stopped temporarily.

### 2.1.1 Configuration after a Hardware Reset

Figure 2.1 shows the CAN configuration procedure after a hardware reset.



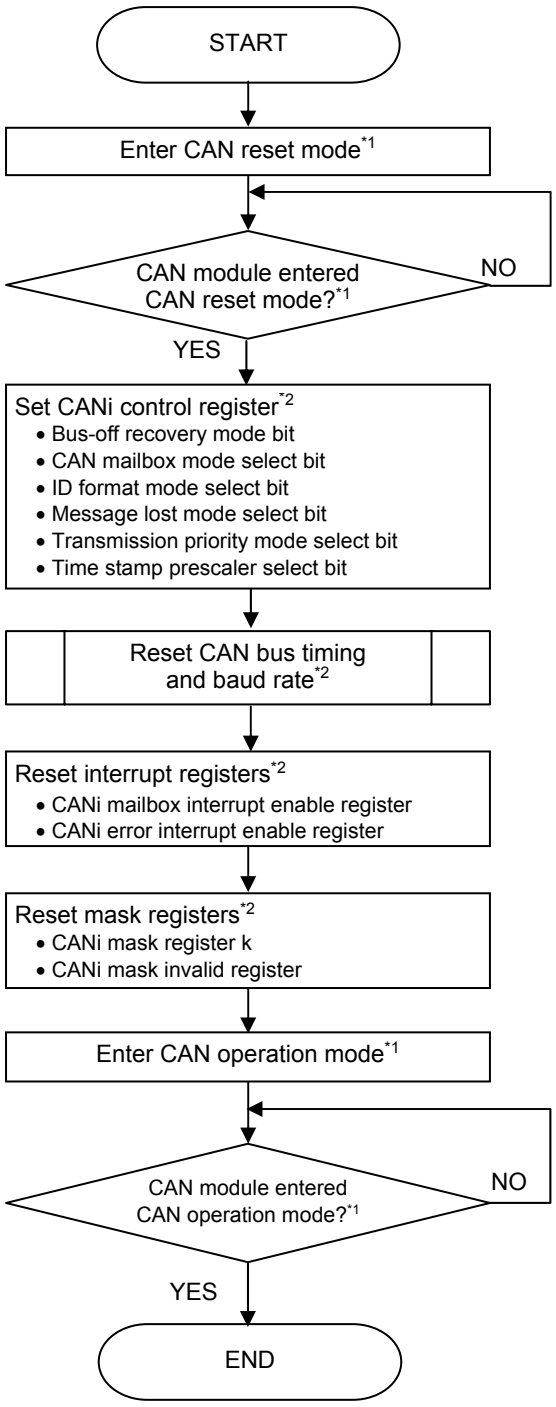

Notes: *1: Set the necessary multiplexed pin function, input threshold value, and driving ability. At this time, confirm that the ports corresponding to CTX are set to "H".

*2: When the CAN operating mode select bits and CAN sleep mode bit are changed, read the CANi status register to ensure that the mode has been switched. Do not change the CAN operating mode select bits and CAN sleep mode bit until the mode has been switched.

**Figure 2.1 CAN Configuration after a Hardware Reset**

### 2.1.2 Configuration after Entering CAN Reset Mode

Figure 2.2 shows the CAN configuration procedure after entering CAN reset mode.



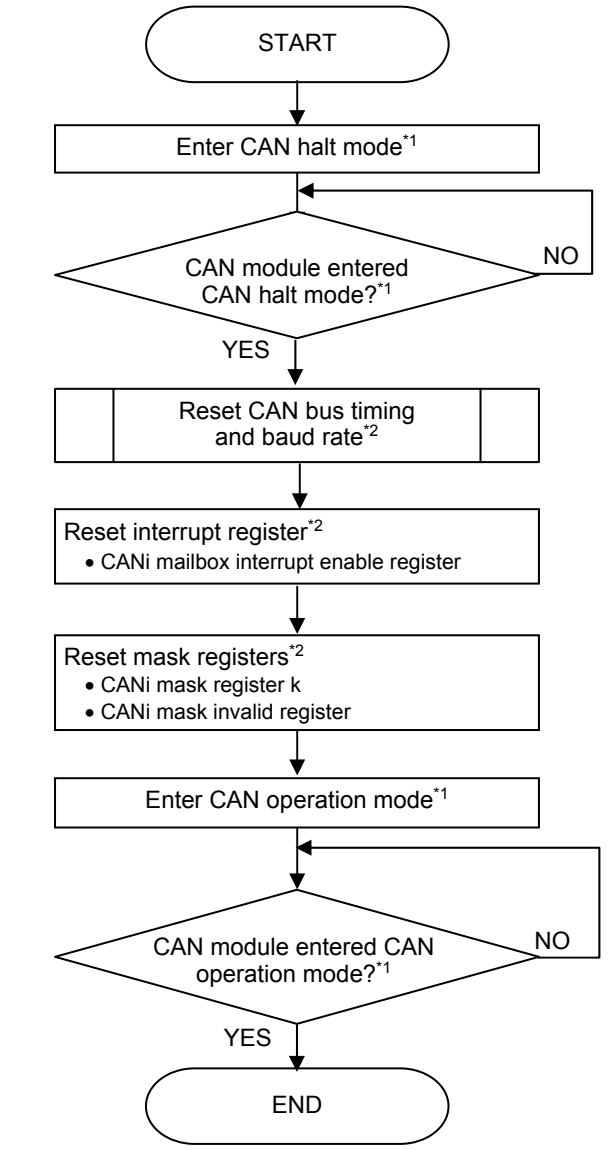

Notes: *1: When the CAN operating mode select bits and CAN sleep mode bit in the CANi control register are changed, read the CANi status register to ensure that the mode has been switched. Do not change the CAN operating mode select bits and CAN sleep mode bit until the mode has been switched.

*2: This step is not always necessary as these registers are not reset after entering CAN reset mode.

**Figure 2.2 CAN Configuration after Entering CAN Reset Mode**

### 2.1.3 Configuration after Entering CAN Halt Mode

Figure 2.3 shows the CAN configuration procedure after entering CAN halt mode.




Notes: *1: When the CAN operating mode select bit is changed, read the CANi status register to ensure that the mode has been switched. Do not change the CAN operating mode select bits until the mode has been switched.

*2: This step is not always necessary as these registers are not reset after entering CAN halt mode.

**Figure 2.3 CAN Configuration after Entering CAN Halt Mode**

## 2.2 Message Transmission

The CAN module incorporates 64 mailboxes for each channel and operates in transmission mode or one-shot transmit mode when transmitting messages. Out of the 64 mailboxes, 32 can be used for transmission in either of the aforementioned modes.

(1) Transmission Mode

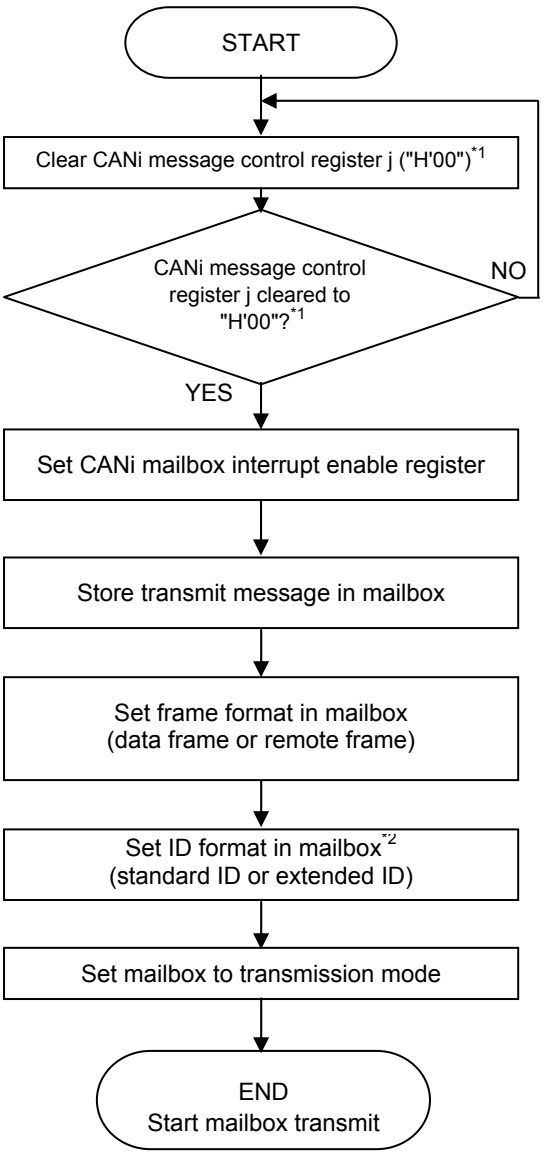
When a mailbox is set to transmission mode, data frames or remote frames set to the mailbox can be transmitted. It is possible to confirm whether or not a transmission is completed successfully by reading the transmission complete flag (SENTDATA) of the corresponding mailbox. The transmission complete flag (SENTDATA) is set to "1" when a transmission is completed successfully. When the CAN module loses arbitration or an error occurs during transmission, the message is retained (the CAN module transmits the message again).

(2) One-Shot Transmission Mode

When a mailbox is set to one-shot transmit mode, data frames or remote frames set to the mailbox can be transmitted. When the one-shot enable bit (ONESHOT) is set to "1", the mailbox transmits a message only once (the CAN module does not transmit the message again if a CAN bus error or CAN bus arbitration lost occurs). It is possible to confirm whether or not a one-shot transmission is completed successfully by reading the transmission complete flag (SENTDATA) or the transmission abort complete flag (TRMABT) of the corresponding mailbox. The transmission complete flag (SENTDATA) is set to "1" when a one-shot transmission is completed successfully. The transmission abort complete flag (TRMABT) is set to "1" when the CAN module loses arbitration or an error occurs during transmission.

### 2.2.1 Transmission Request

Figure 2.4 shows the transmission request procedure. This process should be carried out when there is no transmission or reception request for the corresponding mailbox (CANi message control register j (CiMCTLj) is "H'00" and abort processing is not underway).




Notes: *1: After a transmission is completed successfully, clear CANi message control register j to "H'00" before making new transmit settings.

*2: The ID extension bit is valid only when the ID format mode select bit in the CANi control register is set to "10b" (mixed ID mode). Otherwise, set the ID extension bit to "0".

**Figure 2.4 Transmission Request**

### 2.2.2 Transmission Complete Processing

Figure 2.5 shows the necessary procedure following transmission completion. The necessary procedure is the same regardless of whether an interrupt or polling is used. For details on performing subsequent transmission requests, see section 2.2.1.

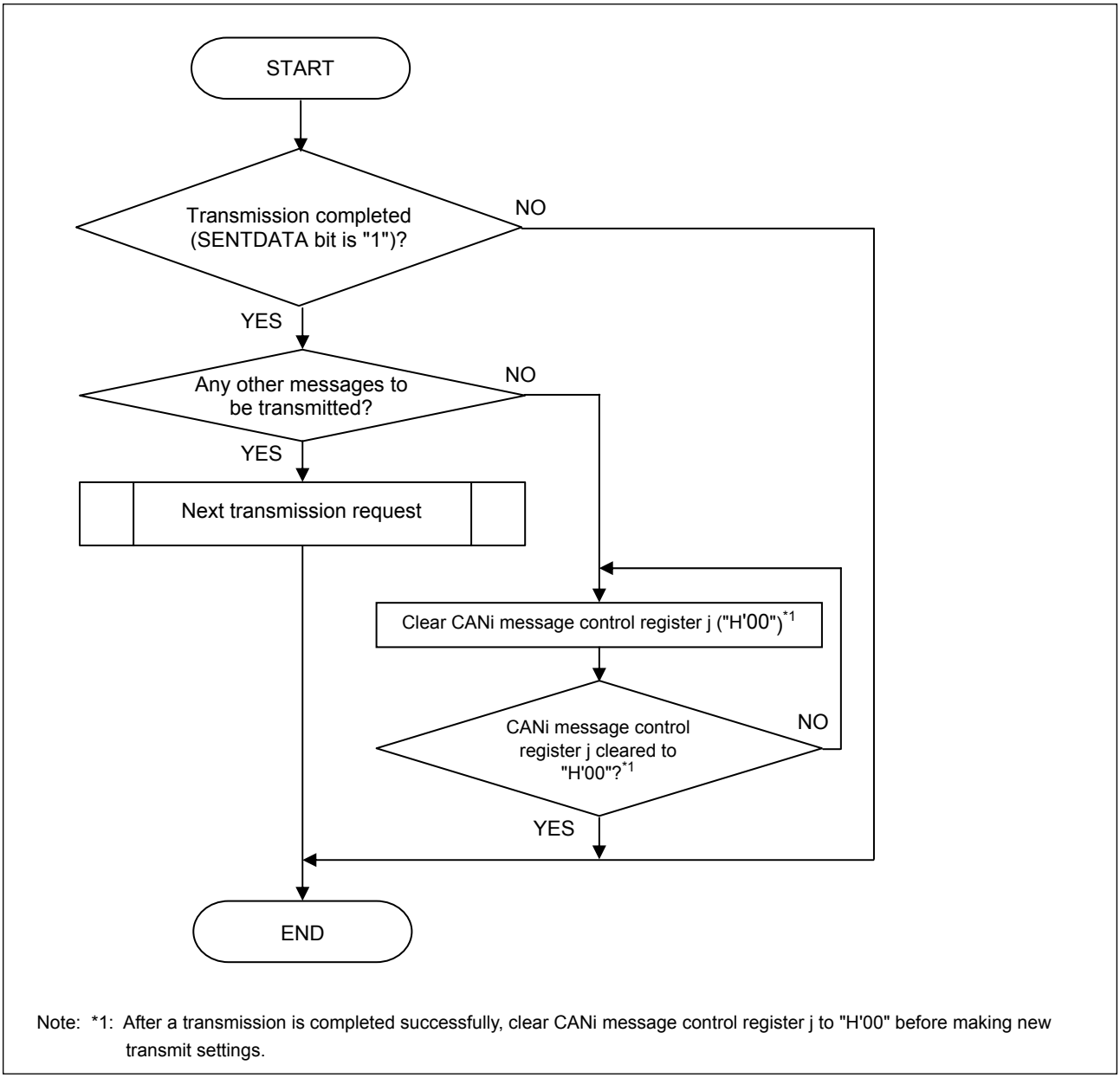
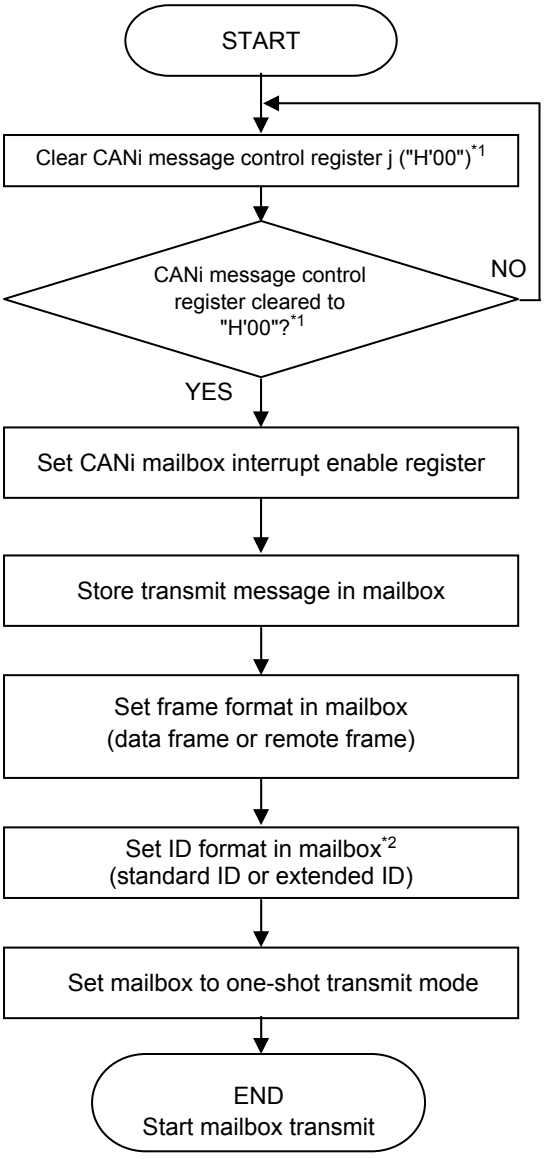



**Figure 2.5 Transmission Complete**

### 2.2.3 One-Shot Transmission Request

When the one-shot enable bit (ONESHOT) is set to "1" in transmission mode, the CAN module transmits only once from the corresponding mailbox. Figure 2.6 shows the one-shot transmission request procedure.

This process should be carried out when there is no transmission or reception request for the corresponding mailbox (CANi message control register j (CiMCTLj) is "H'00" and abort processing is not underway). It is possible to confirm whether or not a one-shot transmission is completed successfully by reading the transmission complete flag (SENTDATA) or the transmission abort complete flag (TRMABT) of the corresponding mailbox. The transmission complete flag (SENTDATA) is set to "1" when a one-shot transmission is completed successfully. The transmission abort complete flag (TRMABT) is set to "1" when the CAN module loses arbitration or an error occurs during transmission.




Notes: *1: After a transmission is completed successfully, clear CANi message control register j to "H'00" before making new transmit settings.

*2: The ID extension bit is valid only when the ID format mode select bit in the CANi control register is set to "10b" (mixed ID mode). Otherwise, set the ID extension bit to "0".

**Figure 2.6 One-Shot Transmission Request**

### 2.2.4 One-Shot Transmission Complete Processing

Figure 2.7 shows the necessary procedure following one-shot transmission completion. This procedure must be performed by polling. If this procedure is performed using an interrupt, when arbitration is lost or transmission is suspended due to an error, a CANi transmission complete interrupt is not generated. For details on performing subsequent one-shot transmission requests for mailboxes set to one-shot transmit mode, see section 2.2.3.

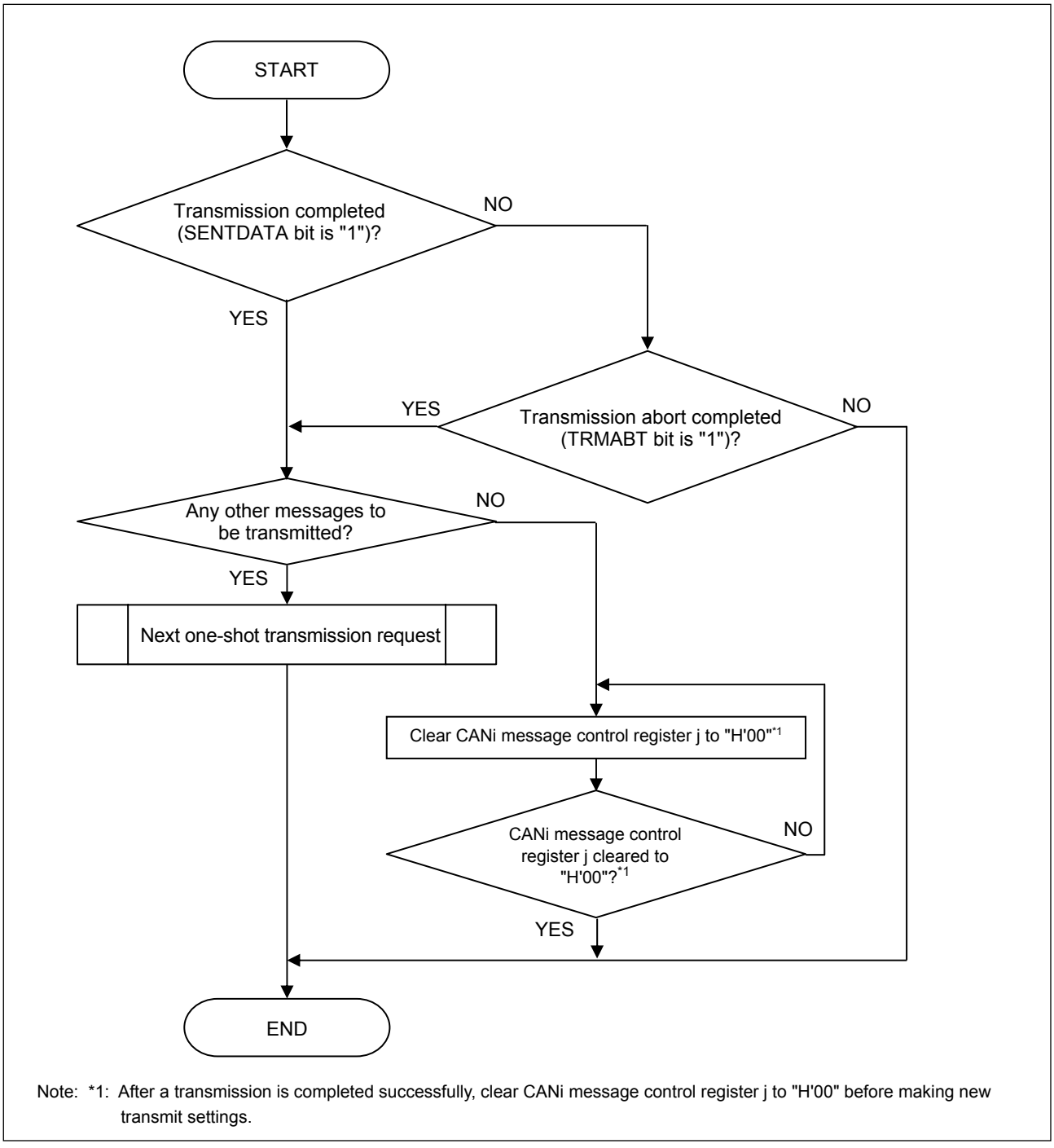



**Figure 2.7 One-Shot Transmission Complete**

### 2.2.5 Transmission Abort

When two or more nodes start transmission simultaneously, the node(s) whose message(s) have lower CAN ID priority lose the arbitration conflict (the message is aborted in one-shot transmission and retained (retransmitted) in transmission). A message can on be successfully transmitted when the CAN module wins arbitration or when transmission is performed when the CAN bus is idle.

The transmission abort function enables discarding of messages that are being retransmitted. It is possible to confirm whether or not transmission abort is completed successfully by reading the transmission complete flag (SENTDATA) or the transmission abort complete flag (TRMABT) of the corresponding mailbox. The transmission complete flag (SENTDATA) is set to "1" when a transmission is completed successfully. The transmission abort complete flag (TRMABT) is set to "1" in the case below.

- Following a transmission abort request, when the transmission abort is completed before starting transmission.
- Following a transmission abort request, when the CAN module detects CAN bus arbitration lost or a CAN bus error.
- In one-shot transmit mode (RECREQ bit is "0", TRMREQ bit is "1", and ONESHOT is "1"), when the CAN module detects CAN bus arbitration lost or a CAN bus error.

The TRMABT bit is not set to "1" when data transmission is completed. In this case, the SENTDATA bit is set to "1". Using the transmission abort function is effective in transmitting a message with a limited time or urgent, higher priority message.

Figure 2.8 shows application examples using the transmission abort function, and figure 2.9 shows the transmission abort procedure.

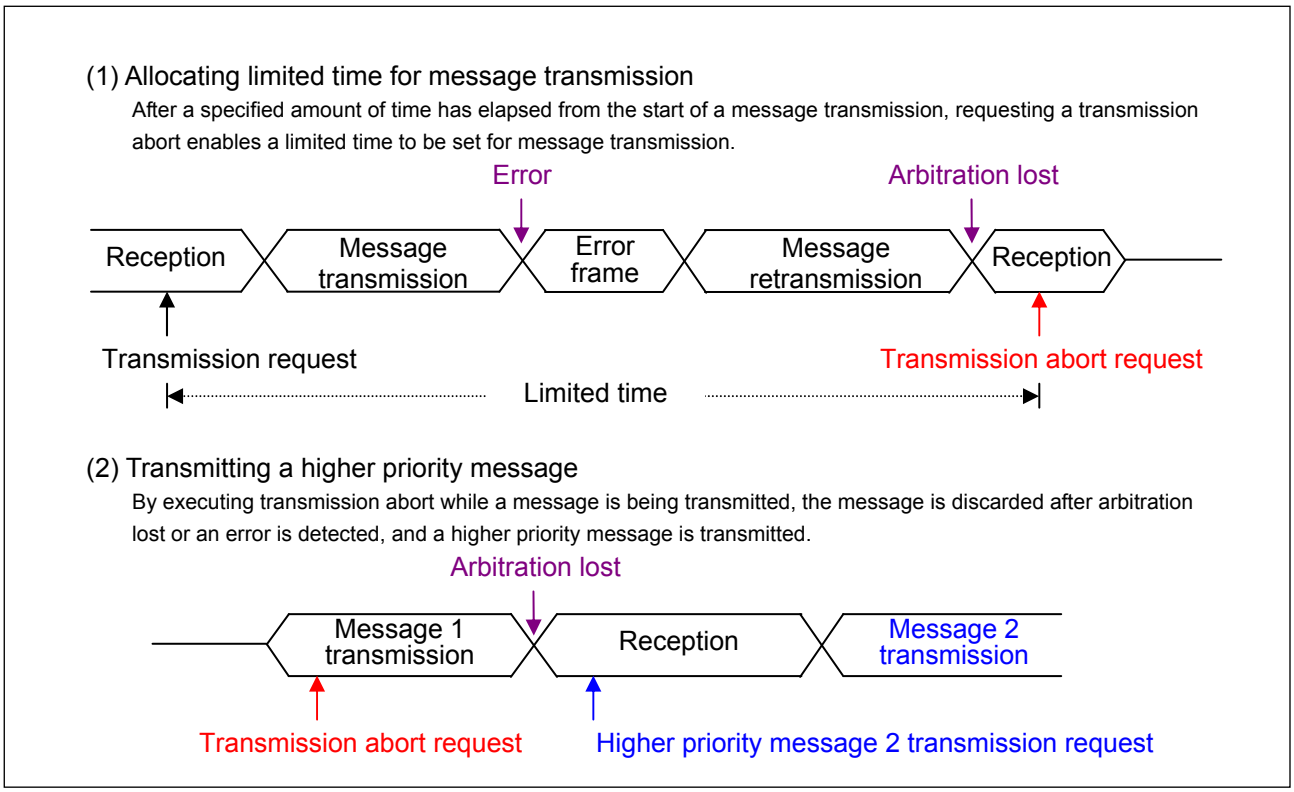
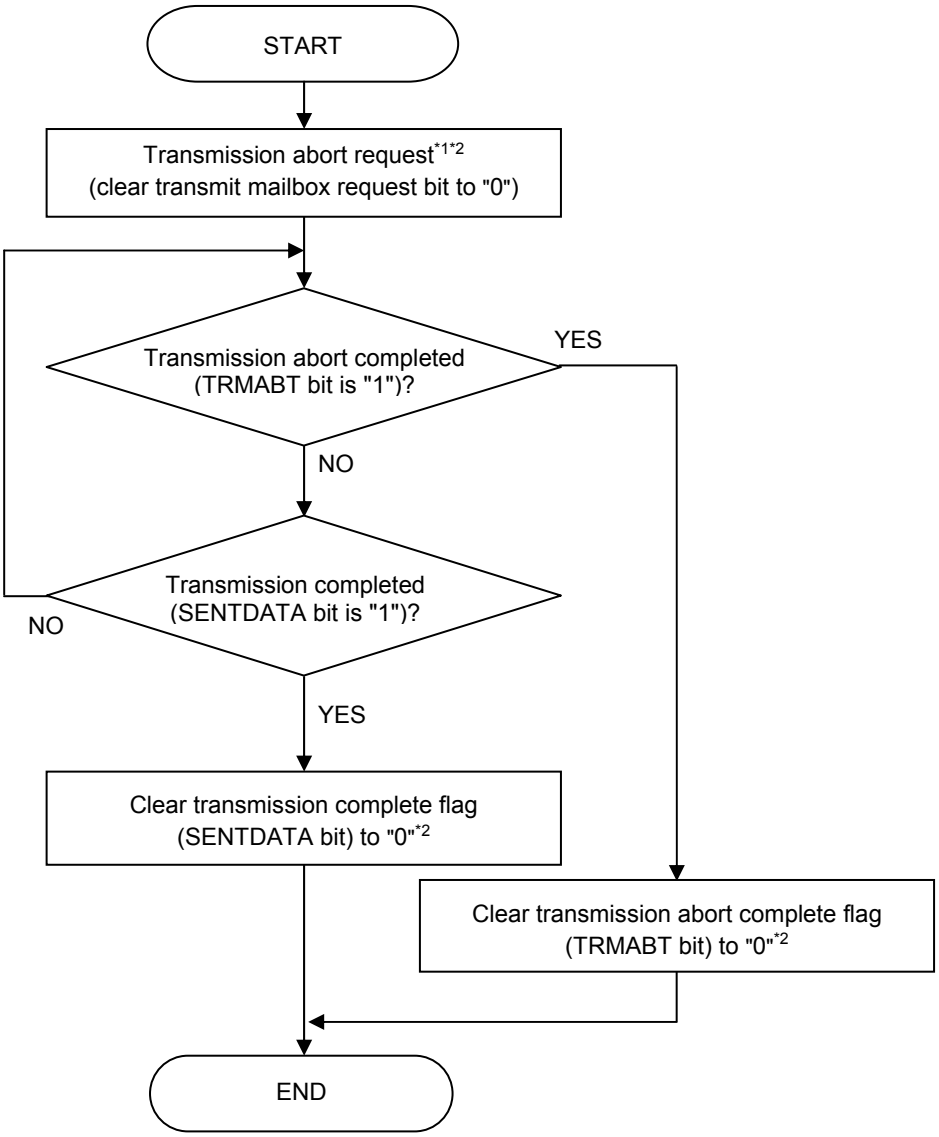



**Figure 2.8 Transmission Abort Function Application Examples**




Notes: *1: Do not change the one-shot setting during this process (if the one-shot enable bit is set to "1", do not clear it to "0").

*2: When writing to the reception complete flag, the transmission complete flag, the message lost flag, the transmission abort complete flag, the receive mailbox request bit, and the transmit mailbox request bit by a program, use the MOV instruction to ensure that only the specified bits are set to "0" and the other bits are set to "1".

**Figure 2.9 Transmission Abort**

## 2.3 Message Reception

The CAN module supports 64 mailboxes for each CAN channel. Receive message are always stored in the lowest-numbered mailbox among the mailboxes set to the matching ID. The receiving message can be selected using an acceptance filter. See section 7 for details on the acceptance filter function. The CAN module operates in reception mode or one-shot receive mode when receiving messages. All 64 mailboxes can be used for receiving messages in reception mode, and 32 mailboxes can be used for receiving messages in one-shot receive mode.

(1) Reception Mode

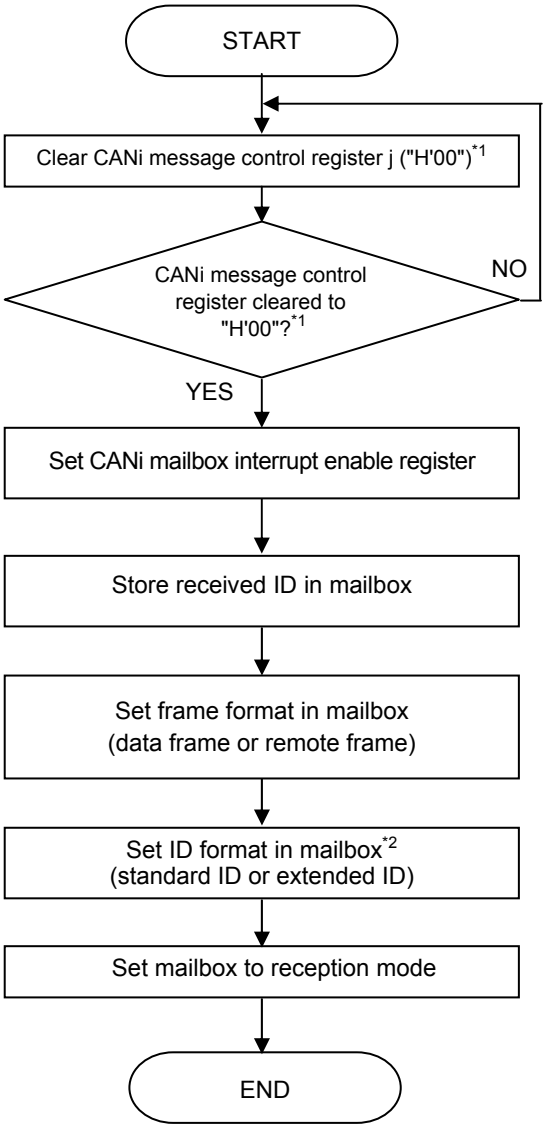
When a mailbox is set to reception mode, data frames or remote frames with the same ID as the mailbox's ID setting (combined with the results of the applicable acceptance filter) can be received. If two or more mailboxes set to reception mode have the same ID, receive messages are always stored in the lowest-numbered mailbox among the mailboxes set to the matching ID. This means it is possible that an overwrite or overrun may occur. (Set the message lost mode select bit (MLM) to select wither overwrite mode or overrun mode.)

(2) One-Shot Receive Mode

When a mailbox is set to one-shot receive mode, data frames or remote frames with the same ID as the mailbox's ID setting (combined with the results of the applicable acceptance filter) can be received. When the one-shot enable bit (ONESHOT) is set to "1", the CAN module receives a message only one time and does not receive any more messages until the receive message is processed by software. If two or more mailboxes set to one-shot receive mode have the same ID, receive messages are stored starting from the lowest-numbered mailbox upward. In other words, the first receive message is stored in the lowest-numbered mailbox. Then, if the first message has not yet been processed, the next receive message is stored in the second-lowest-numbered mailbox.

### 2.3.1 Reception Request

Figure 2.10 shows the reception request procedure. Perform this processing when there is no transmission or reception request for the corresponding mailbox (CANi message control register j (CiMCTLj) is "H'00" and abort processing is not underway).




Notes: *1: After a reception is completed successfully, clear CANi message control register j to "H'00" before making new receive settings.

*2: The ID extension bit is valid only when the ID format mode select bit in the CANi control register is set to "10b" (mixed ID mode). Otherwise, set the ID extension bit to "0".

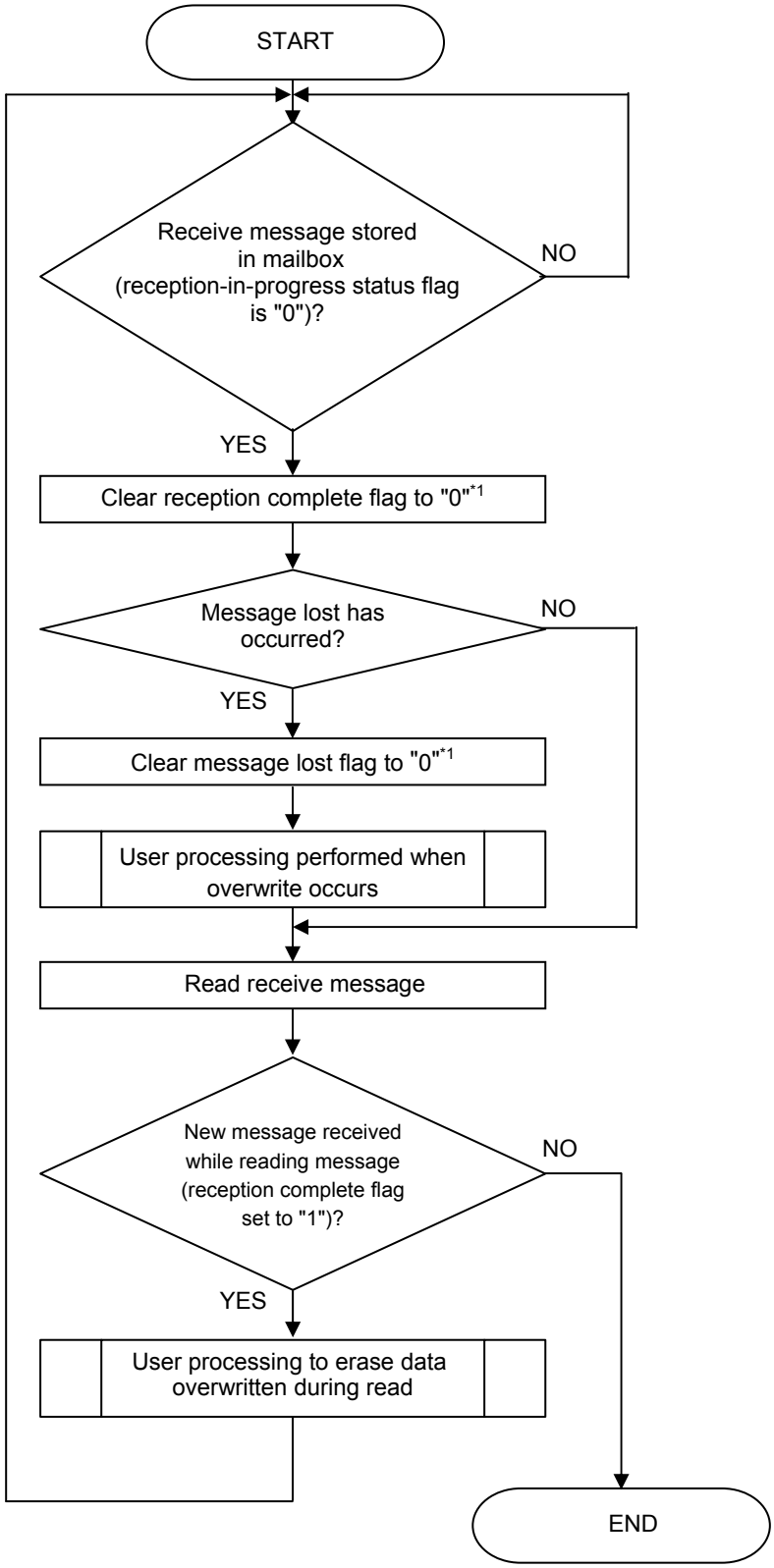
**Figure 2.10 Reception Request**

### 2.3.2 Reception Complete with Mailbox Set to Reception Mode (Overwrite Mode)

Figure 2.11shows the receive message processing procedure when a mailbox is set to overwrite mode (message lost mode select bit (MLM) is "0") and reception mode.

In this mode, before the previously receive message processing is completed by software, if a mailbox receives a new message, the old message in the mailbox is overwritten by the new one. Therefore, it is necessary for the software, after reading the receive message from the mailbox, to confirm that the mailbox was not overwritten while the read operation was in progress. If the mailbox was overwritten, the reception complete flag (NEWDATA) is set to "1". When a mailbox is overwritten while the reception complete flag (NEWDATA) is "1", the corresponding message lost flag (MSGLOST) is set to "1".

When using the receive mailbox search mode, perform the above process after checking to determine the numbers of mailboxes with unprocessed receive messages. For details on the receive mailbox search mode, see section 4.1.1.




Notes: *1: When writing to the reception complete flag, the transmission complete flag, the message lost flag, the transmission abort complete flag, the receive mailbox request bit, and the transmit mailbox request bit by a program, use the MOV instruction to ensure that only the specified bits are set to "0" and the other bits are set to "1".

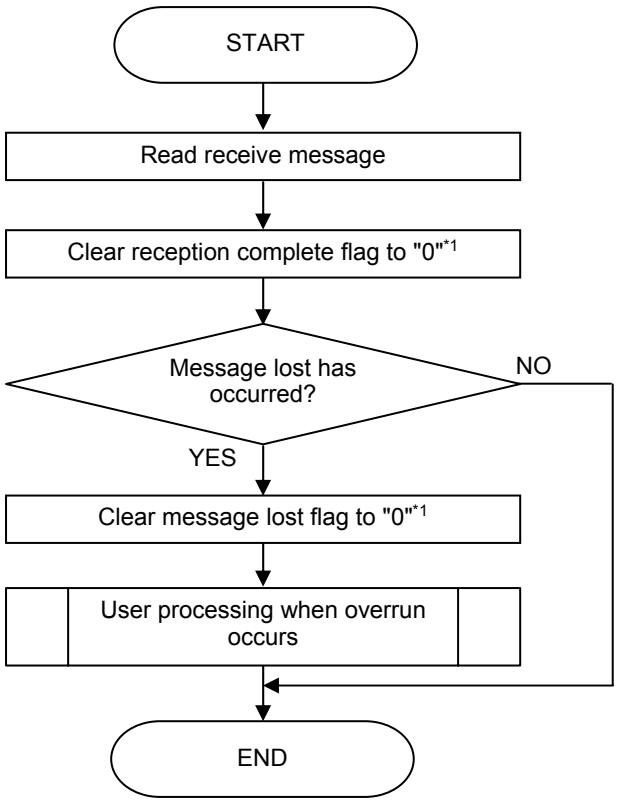
**Figure 2.11 Message Reception Processing with Mailbox Set to Reception Mode (Overwrite Mode)**

### 2.3.3 Reception Complete with Mailbox Set to Reception Mode (Overrun Mode)

Figure 2.12 shows the receive message processing procedure when a mailbox is set to overrun mode (message lost mode select bit (MLM) is "1") and reception mode.

In this mode, before the previously receive message processing is completed by software, if a mailbox receives a new message, the new message is discarded (not stored in the mailbox). In this case, the message lost flag (MSGLOST) corresponding to the mailbox is set to "1" and an overrun interrupt is generated (if the overrun interrupt is enabled).

When using the receive mailbox search mode, perform the above process after checking to determine the number of mailboxes with unprocessed received messages. For details on the receive mailbox search mode, see section 4.1.1.

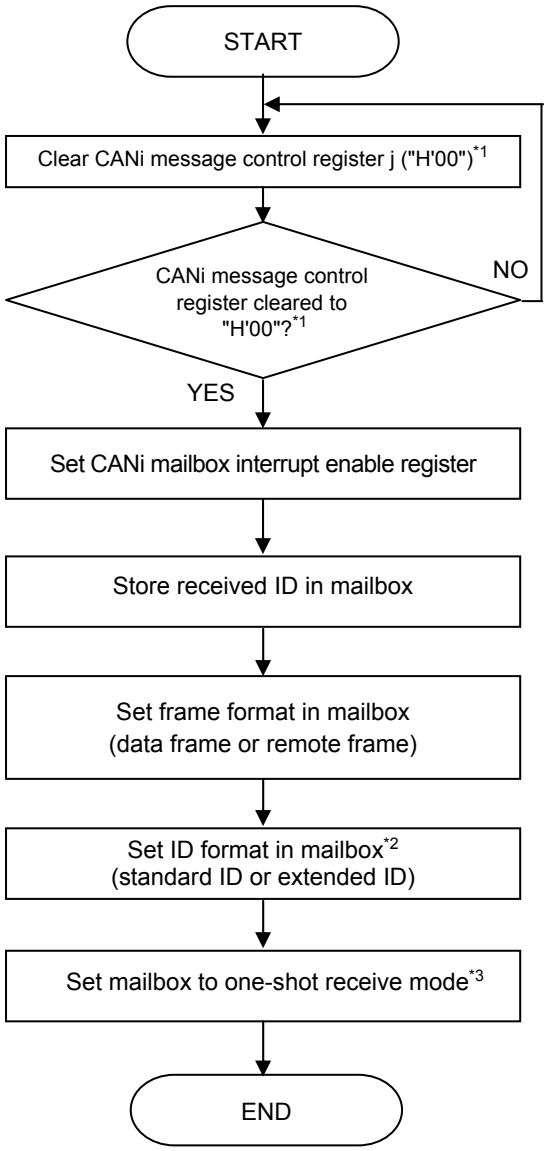



Notes: *1: When writing to the reception complete flag, the transmission complete flag, the message lost flag, the transmission abort complete flag, the receive mailbox request bit, and the transmit mailbox request bit by a program, use the MOV instruction to ensure that only the specified bits are set to "0" and the other bits are set to "1".

**Figure 2.12 Message Reception Processing with Mailbox Set to Reception Mode (Overrun Mode)**

### 2.3.4 One-Shot Reception Request

Figure 2.13 shows the one-shot reception request procedure. Perform this processing when there is no transmission or reception request for the corresponding mailbox (CANi message control register j (CiMCTLj) is "H'00" and abort processing is not underway).




Notes: *1: After a reception is completed successfully, clear CANi message control register j to "H'00" before making new receive settings.

*2: The ID extension bit is valid when the ID format mode select bit in the CANi control register is set to "10b" (mixed ID mode). Otherwise, set the ID extension bit to "0".

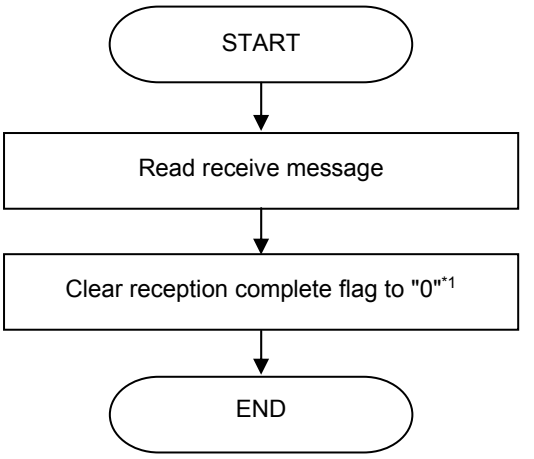
*3: When entering one-shot receive mode, set the one-shot enable bit to "1" at the same time as setting the receive mailbox request bit to "1".

**Figure 2.13 One-Shot Reception Request**

### 2.3.5 Reception Complete with Mailbox Set to One-Shot Receive Mode

Figure 2.14 shows the receive message processing procedure when a mailbox is set to one-shot receive mode. In one-shot receive mode, the setting of the message lost mode select bit (MLM) is irrelevant and a message lost condition is never generated. This is because a mailbox that has received a message receives no further messages until the reception complete flag (NEWDATA) is cleared to "0".

When using the receive mailbox search mode, perform the above process after checking to determine the number of mailboxes with unprocessed receive messages. For details on the receive mailbox search mode, see section 4.1.1.

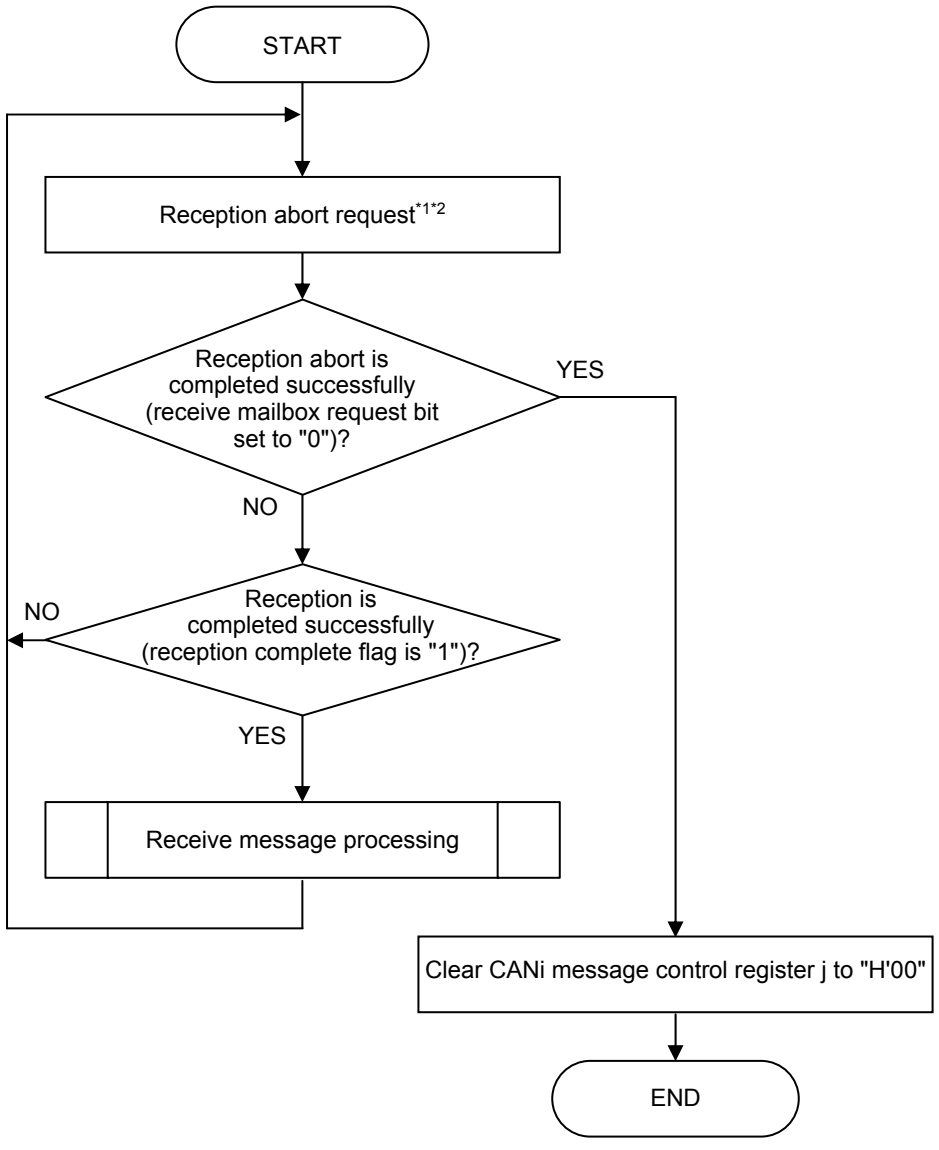



Note: *1: When writing to the reception complete flag, the transmission complete flag, the message lost flag, the transmission abort complete flag, the receive mailbox request bit, and the transmit mailbox request bit by a program, use the MOV instruction to ensure that only the specified bits are set to "0" and the other bits are set to "1".

**Figure 2.14 Message Reception Processing with Mailbox Set to One-Shot Reception Mode**

### 2.3.6 Reception Abort

A reception abort is executed when the receive mailbox request bit (RECREQ), reception complete flag (NEWDATA), and message lost flag (MSGLOST) in the CANi message control register j (CiMCTLj) are cleared to "0" simultaneously. (If one-shot reception is enabled, clear the one-shot enable bit (ONESHOT) to "0" after the abort operation is completed). Figure 2.15 shows the reception abort procedure.




Notes: *1: When writing to the receive mailbox request bit, the reception complete flag, and the message lost flag by a program, use the MOV instruction to ensure that only the specified bits are set to "0" and the other bits are set to "1".

*2: When using one-shot receive mode, after writing "0" to the receive mailbox request bit, confirm that the receive mailbox request bit is "0", and then set the one-shot enable bit (ONESHOT) to "0".

**Figure 2.15 Reception Abort**

# 3. Mailbox Modes

During CAN configuration, normal mailbox mode or FIFO mailbox mode can be selected by setting the CAN mailbox mode select bit (MBM) in the CANi control register (CiCTLR) to 0 and 1, respectively.

(1) Normal Mailbox Mode
When the MBM bit is "0" (normal mailbox mode), mailboxes [0] to [63] are configured as transmit or receive mailboxes.

(2) FIFO Mailbox Mode
When the MBM bit is "1" (FIFO mailbox mode), mailboxes [0] to [55] are configured as transmit or receive mailboxes. Mailboxes [56] to [59] are configured as a transmit FIFO and mailboxes [60] to [63] as a receive FIFO.

## 3.1 Normal Mailbox Mode

Mailboxes [0] to [63] are configured as transmit or receive mailboxes. Figure 3.1 shows the mailbox configuration in normal mailbox mode.

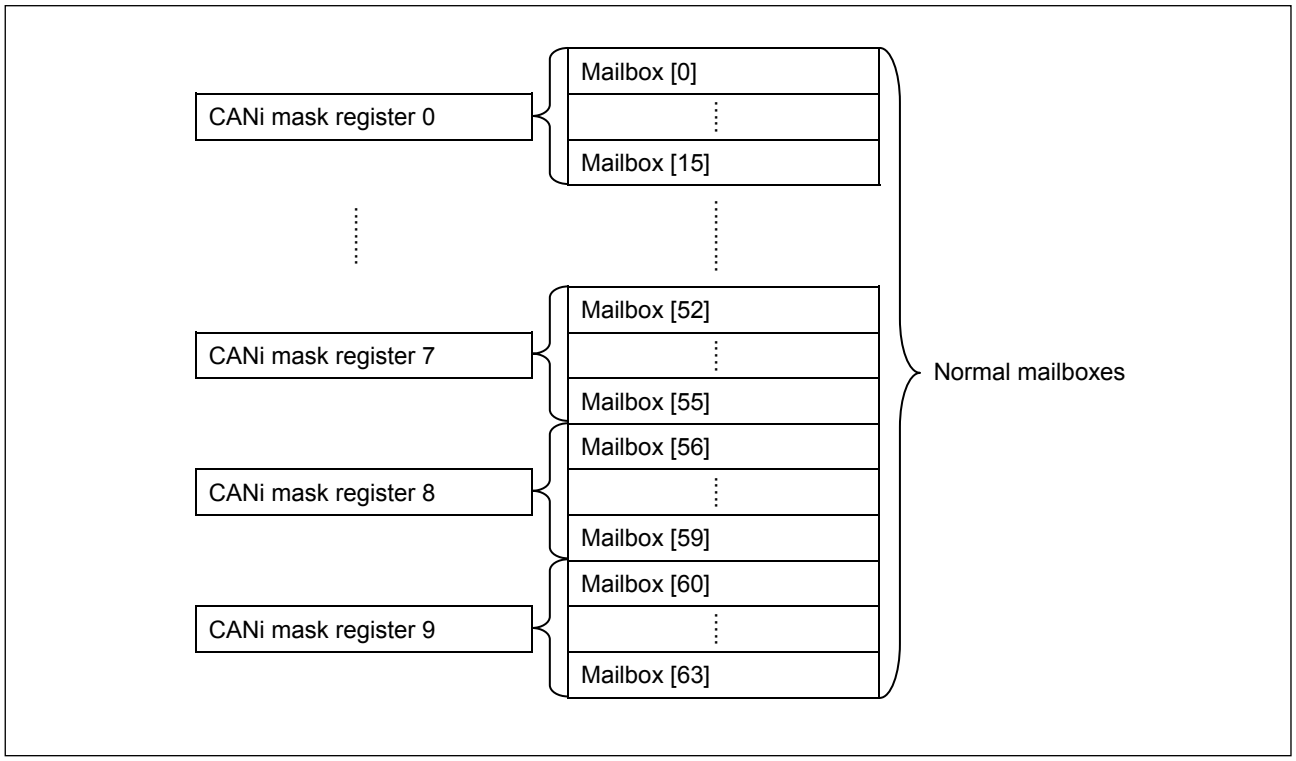



**Figure 3.1 Mailbox Configuration in Normal Mailbox Mode**

## 3.2 FIFO Mailbox Mode

Figure 3.2 shows the mailbox configuration in FIFO mailbox mode.

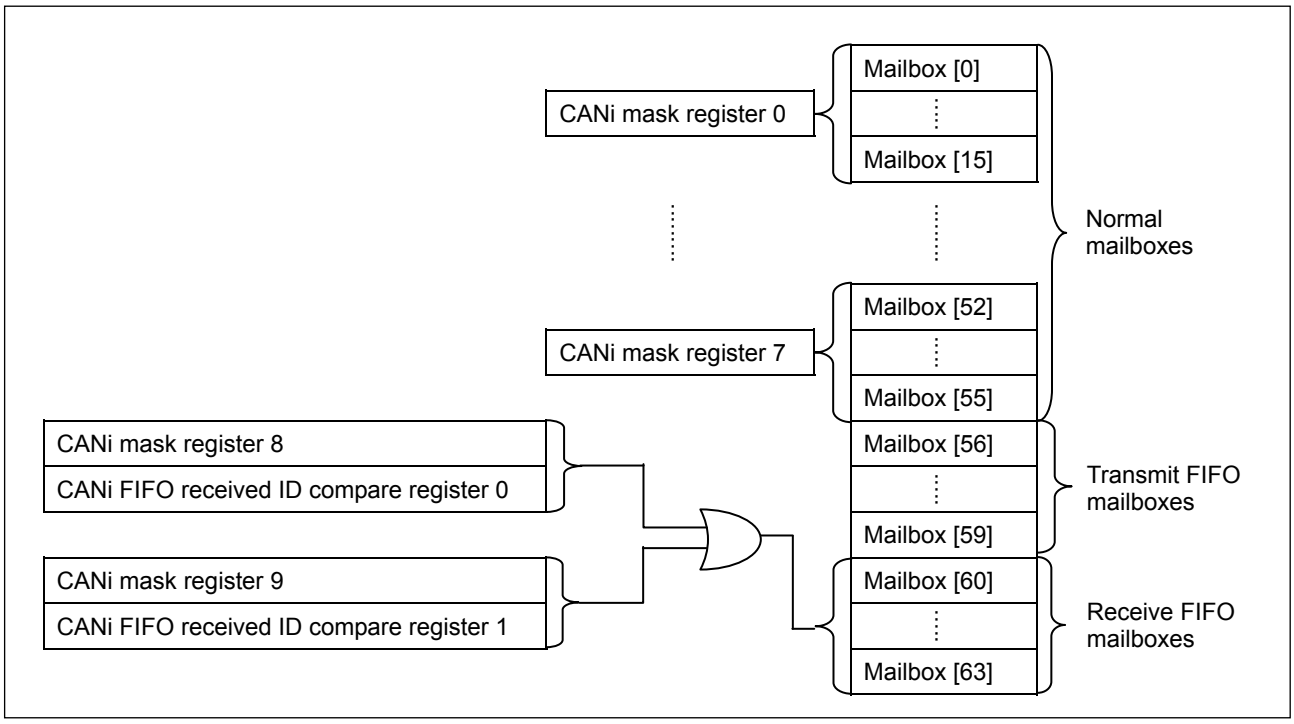



**Figure 3.2 Mailbox Configuration in FIFO Mailbox Mode**

Mailboxes [56] to [59] are set as a transmit FIFO, and mailboxes [60] to [63] are set as a receive FIFO. Write transmit data into mailbox [56], and read receive data from mailbox [60]. Updating the transmit/receive FIFO pointers are enabled by writing "H'FF" to the CANi transmit/receive FIFO pointer control registers (CiTFPCR and CiRFPCR). Mailboxes [0] to [55] are set as normal transmit or receive mailboxes.

The following options can be selected by setting the corresponding bits in the CANi mailbox interrupt enable register (CiMIER1).

- Transmit FIFO and receive FIFO interrupt disabled/enabled
- Transmit FIFO and receive FIFO interrupt generation source

The bits are as follows:

- b24: Transmit FIFO interrupt enable bit (0: Interrupt disabled; 1: Interrupt enabled)
- b25: Transmit FIFO interrupt generation timing control bit (0: Every time transmission is completed; 1: When transmit FIFO becomes empty due to completion of transmission)
- b28: Receive FIFO interrupt enable bit (0: Interrupt disabled; 1: Interrupt enabled)
- b29: Receive FIFO interrupt generation timing control bit (0: Every time reception is completed; 1: When receive FIFO becomes buffer warning by completion of reception[*1])

Note: *1: No interrupt request is generated when the receive FIFO becomes buffer warning from full.

CANi message control register j (CiMCTLj) corresponding to mailboxes [56] to [63] is not used in FIFO mailbox mode. Instead, the CANi transmit FIFO control register (CiTFCR) and CANi receive FIFO control register (CiRFCR) are used.

Setting the transmit FIFO enable bit (TFE) in the CANi transmit FIFO control register (CiTFCR) to "1" causes mailboxes [56] to [59] to function as transmit FIFO. When the transmit FIFO enable bit (TFE) is cleared to "0", mailboxes [56] to [59] do not function as transmit FIFO (transmit FIFO is halted).

When the transmit FIFO enable bit (TFE) is cleared to "0" during transmission from a transmit FIFO, the transmit FIFO is emptied and any unsent messages are lost after the next transmission complete, error, arbitration lost, or transition to CAN halt mode occurs.

The transmit FIFO mailboxes are composed of four stages. When there are no messages in the transmit FIFO, the transmit FIFO empty status bit (TFEST) in the CANi transmit FIFO control register (CiTFCR) is set to "1". When four mailboxes in transmit FIFO contain messages (in other words, if there are four unsent messages), the transmit FIFO full status bit (TFFST) in the CANi transmit FIFO control register (CiTFCR) is set to "1".

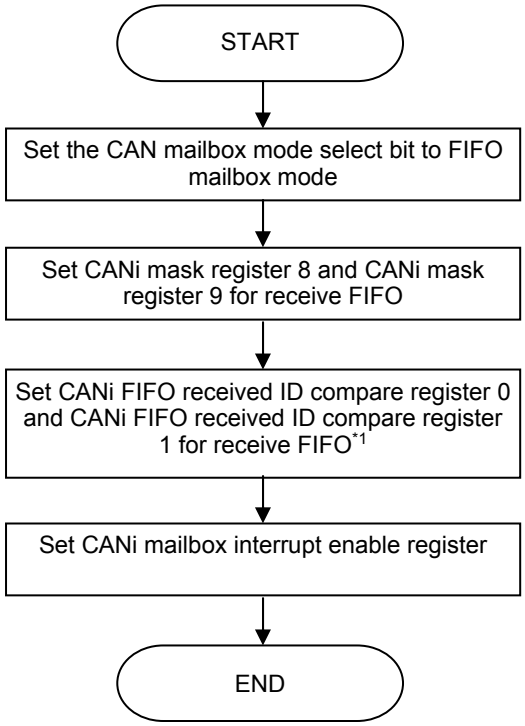
Setting the receive FIFO enable bit (RFE) in the CANi receive FIFO control register (CiRFCR) to "1" causes mailboxes [60] to [63] to function as a receive FIFO. When the receive FIFO enable bit (RFE) is cleared to "0", mailboxes [60] to [63] do not function as a receive FIFO (receive FIFO is halted).

The receive FIFO mailboxes are composed of four stages. When there are no messages in the receive FIFO, the receive FIFO empty status flag (RFEST) in the CANi receive FIFO control register (CiRFCR) is set to "1". When three out of four mailboxes in receive FIFO contain messages, the receive FIFO buffer warning status flag (RFWST) in the CANi receive FIFO control register (CiRFCR) is set to "1". When four mailboxes in the receive FIFO contain messages, the receive FIFO full status flag (RFFST) in the CANi receive FIFO control register (CiRFCR) is set to "1". Furthermore, when a new message is received while the receive FIFO is full, the receive FIFO message lost flag (RFMLF) in the CANi receive FIFO control register (CiRFCR) is set to "1". In this case, the new message is discarded (not stored in the mailbox) if the message lost mode select bit (MLM) in the CANi control register (CiCTLR) is set to overrun mode. If the message lost mode select bit (MLM) is set to overwrite mode, the first receive message among those stored in the receive FIFO is overwritten by the new message (the receive pointers increment automatically).

FIFO mailbox mode uses two mask registers (CiMKR8 and CiMKR9) and two FIFO received ID compare registers CiFIDCR0 and CiFIDCR1. If the ID of a message matches the combined results of one of the two mask registers and one of the two FIFO received ID compare registers, it is stored in the receive FIFO.

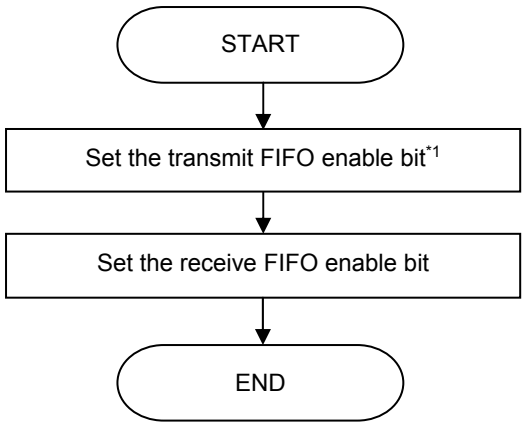
### 3.2.1 FIFO Mailbox Mode Settings

Perform settings for FIFO mailbox mode while the CAN module is in CAN configuration or CAN operation mode. Figure 3.3 shows the FIFO mailbox mode settings while in CAN configuration mode (after a hardware reset, after entering CAN reset mode). Figure 3.4 shows the FIFO mailbox mode settings while in CAN operation mode.




Notes: *1: The ID extension bit is valid only when the ID format mode select bit in the CANi control register is set to "10b" (mixed ID mode). Otherwise, set the ID extension bit to "0".

**Figure 3.3 FIFO Mailbox Mode Setting in CAN Configuration**

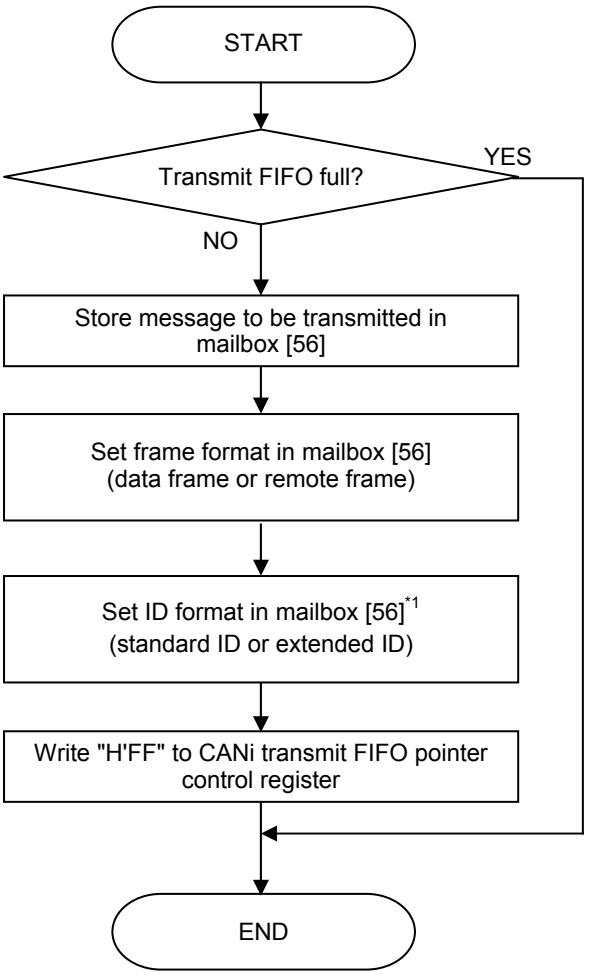



Note: *1: When carrying out this process after disabling the transmit FIFO by clearing the transmit FIFO enable bit to "0", confirm that the transmit FIFO empty status bit is set to "1" before resetting the transmit FIFO enable bit to "1".

**Figure 3.4 FIFO Mailbox Mode Setting in CAN Operation Mode**

### 3.2.2 FIFO Transmission

Figure 3.5 shows the processing procedure for transmitting a message from the FIFO.




Note: *1: The ID extension bit is valid only when the ID format mode select bit in the CANi control register is set to "10b" (mixed ID mode). Otherwise, set the ID extension bit to "0".

**Figure 3.5 Transmission Processing in FIFO Mailbox Mode**

The procedure for aborting FIFO transmission is similar to aborting transmission from a normal mailbox described in section 2.2.5. To abort FIFO transmission, clear the transmit FIFO enable bit (TFE) to "0" instead of the transmit mailbox request bit (TRMREQ). In addition, the transmit FIFO empty status bit (TFEST) is set to "1" instead of the transmission abort complete flag (TRMABT).

### 3.2.3 FIFO Reception (Overwrite Mode)

Figure 3.6 shows the processing procedure for receiving a message by the FIFO using overwrite mode. In overwrite mode, it is necessary to consider the possibility of a message being overwritten while it is being read. If a message is overwritten while it is being read, it cannot be used as a normal message.

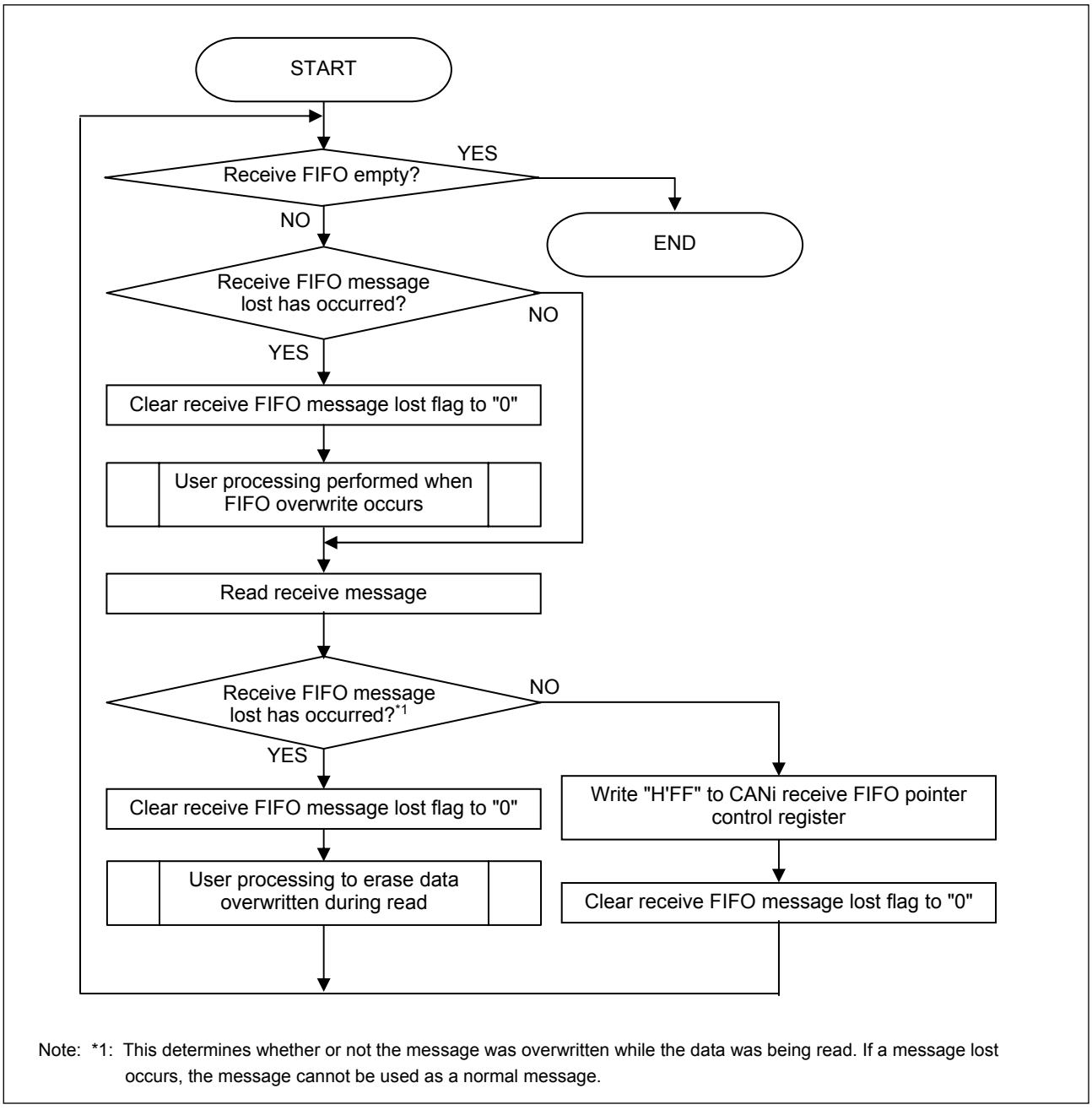



**Figure 3.6 Reception Processing in FIFO Mailbox Mode (Overwrite Mode)**

### 3.2.4 FIFO Reception (Overrun Mode)

Figure 3.7 shows the processing procedure for receiving a message by the FIFO using overrun mode. In overrun mode it is not necessary to consider the possibility of a message being overwritten while it is being read. The read value is not overwritten, even if an overrun occurs.

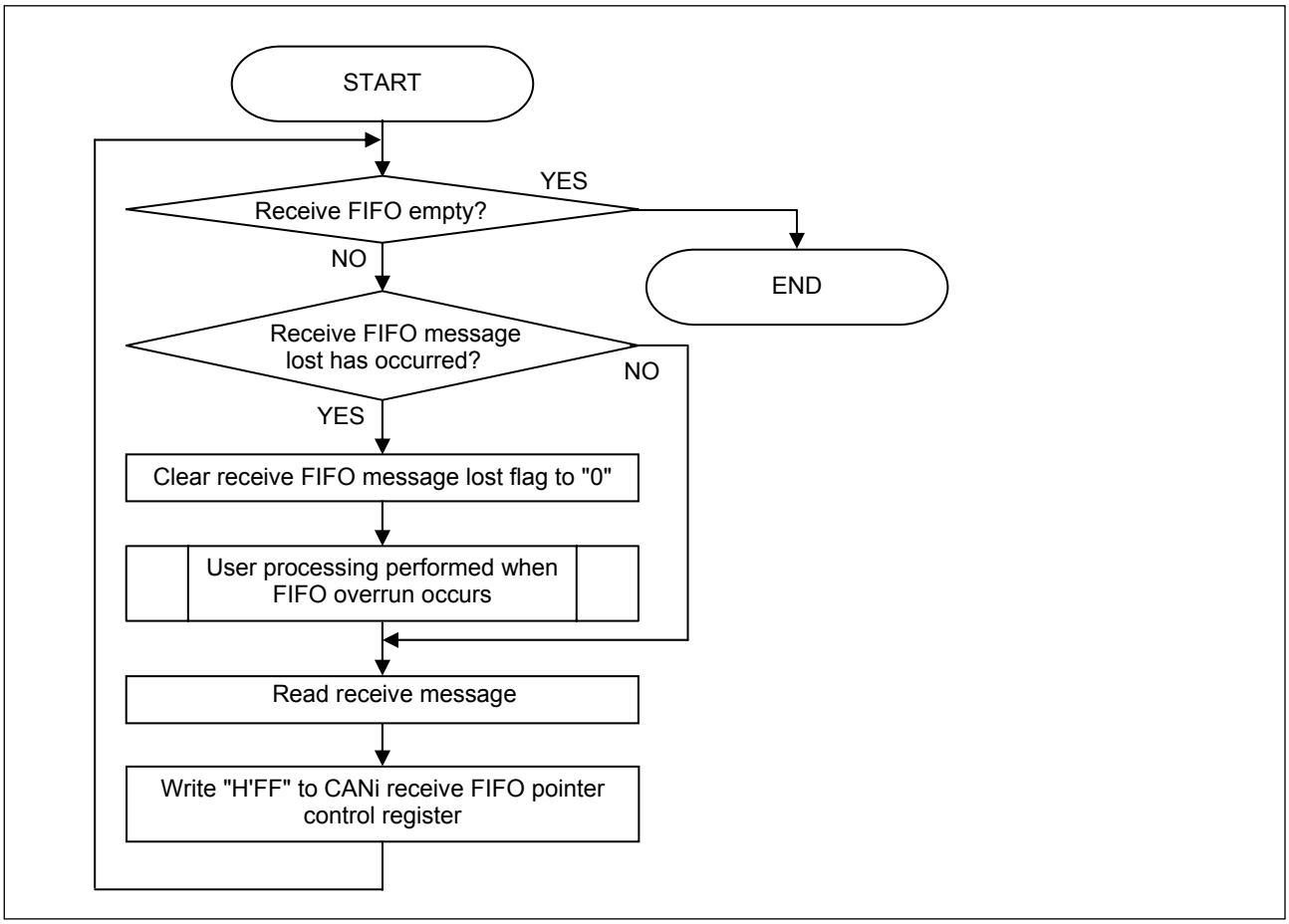



**Figure 3.7 Reception Processing in FIFO Mailbox Mode (Overrun Mode)**

# 4. Mailbox Search Function

Usually it is necessary to search for a mailbox number after each transmission or reception is completed when two or more transmit or receive mailboxes have been set up. This means that the load on the software grows as the number of mailboxes increases. The mailbox search function can be used to reduce the load on the software. The mailbox search mode provides an easy way to search for a mailbox number that has a reception/transmission complete message. The following four modes are used for mailbox searches:

- Receive mailbox search mode
- Transmit mailbox search mode
- Message lost search mode
- Channel search mode

These modes can be used in both normal mailbox mode and FIFO mailbox mode. For polling processing, reading the CANi status register (CiSTR) is recommended before using the mailbox search function. Figure 4.1 shows a CANi status register check example (polling operation).

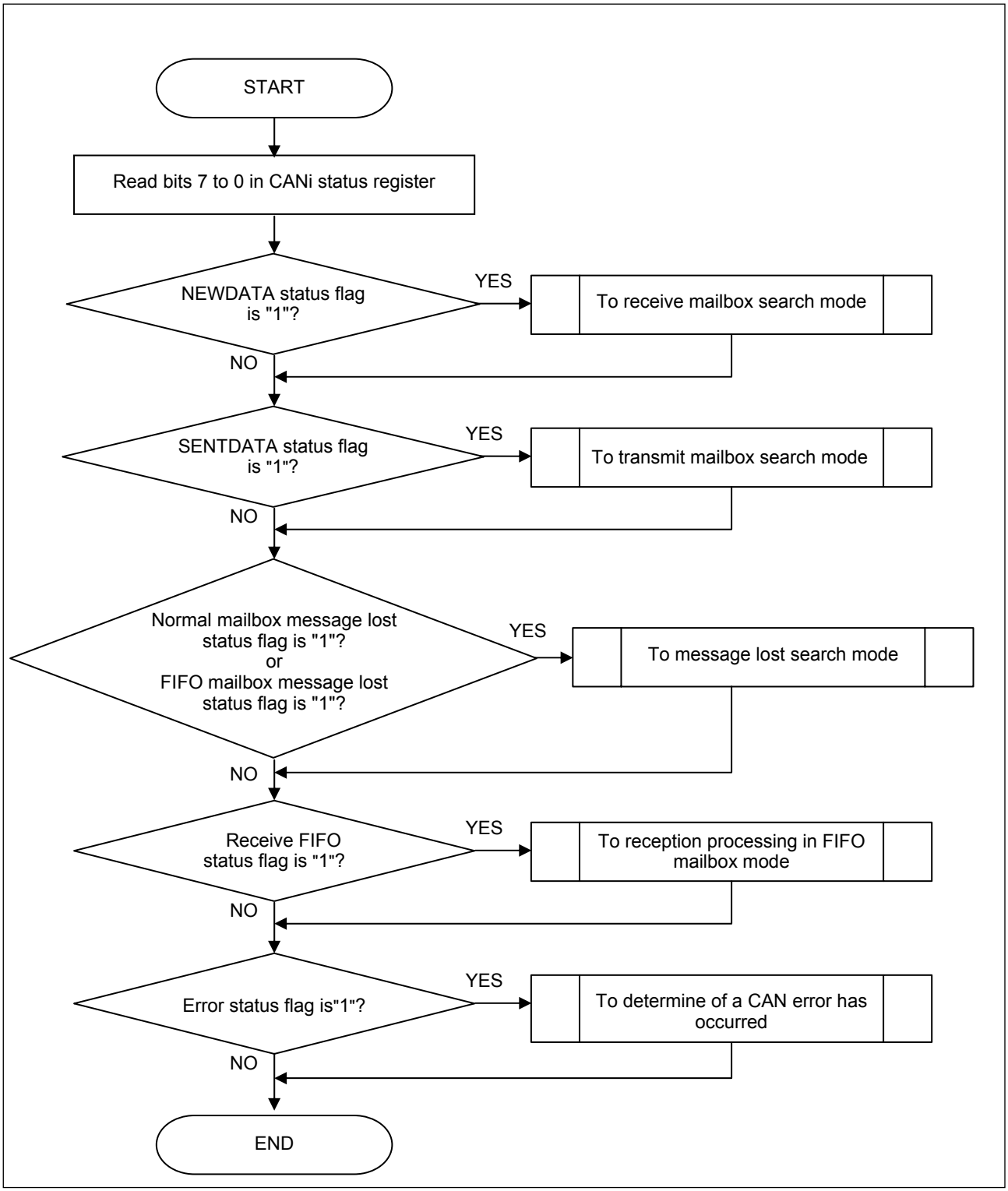



**Figure 4.1 Example of Reading the CANi Status Register (Polling Operation)**

## 4.1 Using the Mailbox Search Function

Table 4.1 lists the setting values of the MBSM bit.

**Table 4.1 Setting Values of Mailbox Search Mode Select Bits**

| CANi Mailbox Search Mode Register (CiMSMR) | | Search Mode |
|---|---|---|
| b1 | b0 | |
| 0 | 0 | Receive mailbox search mode (search for reception complete flag) |
| 0 | 1 | Transmit mailbox search mode (search transmission complete flag) |
| 1 | 0 | Message lost search mode (search message lost flag ) |
| 1 | 1 | Channel search mode |

### 4.1.1 Receive Mailbox Search Mode

This mode searches the lowest mailbox number that has completed reception.

To use this mode, set the MBSM bits to "00b". The reception completed mailbox number can be read from the CANi mailbox search status register (CiMSSR). When two or more mailboxes are in the reception completed state (i.e. when two or more mailboxes have set their reception complete flag (NEWDATA) to "1"), the lowest mailbox number is read.

The reception complete flag (NEWDATA) is cleared to "0" by software as part of the reception completed processing. After that, if there are other mailboxes in the reception completed state, the mailbox search function can be used again to read the next mailbox number. When no other mailboxes are in the reception completed status, the search result status bit (SEST) is set to "1".

In FIFO mailbox mode, mailbox [60] is read for the receive FIFO mailbox when the receive FIFO empty status flag (RFEST) is cleared to "0" (unread message in receive FIFO).

Figure 4.2 shows the procedure for using the receive mailbox search.

For details on reception completed processing, see section 2.3.2, 2.3.3, 2.3.5, 3.2.3, or 3.2.4.

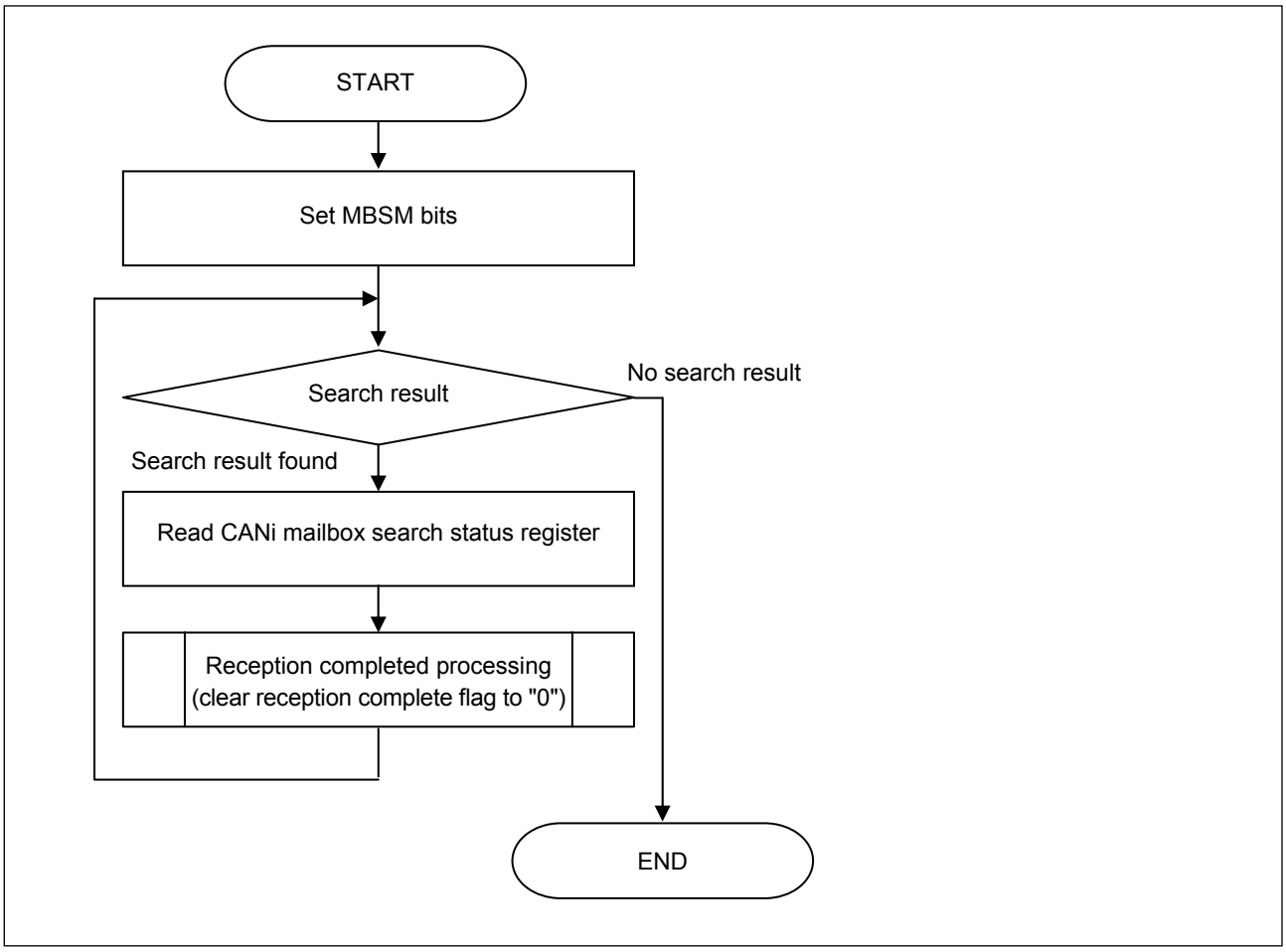



**Figure 4.2 Receive Mailbox Search**

### 4.1.2 Transmit Mailbox Search Mode

This mode searches for the mailbox number that has completed transmission successfully.

To use this mode, set the MBSM bits to "01b". The search result can be read from the CANi mailbox search status register (CiMSSR). When two or more mailboxes are in the transmission completed status (i.e. when two or more mailboxes have set their transmission complete flag (SENTDATA) to "1"), the lowest mailbox number is read.

The transmission complete flag (SENTDATA) is cleared to "0" by software as part of the transmission completed processing. After that, if there are other mailboxes in the transmission completed status, the mailbox search function can be used again to read the next mailbox number. When no other mailboxes are in the transmission completed status, the search result status bit (SEST) is set to "1". In FIFO mailbox mode, transmit FIFO mailboxes are not included in the mailbox search function.

Figure 4.3 shows the procedure for using transmit mailbox search.

For details on continuing to request transmission in normal mailbox mode, see section 2.2.1.

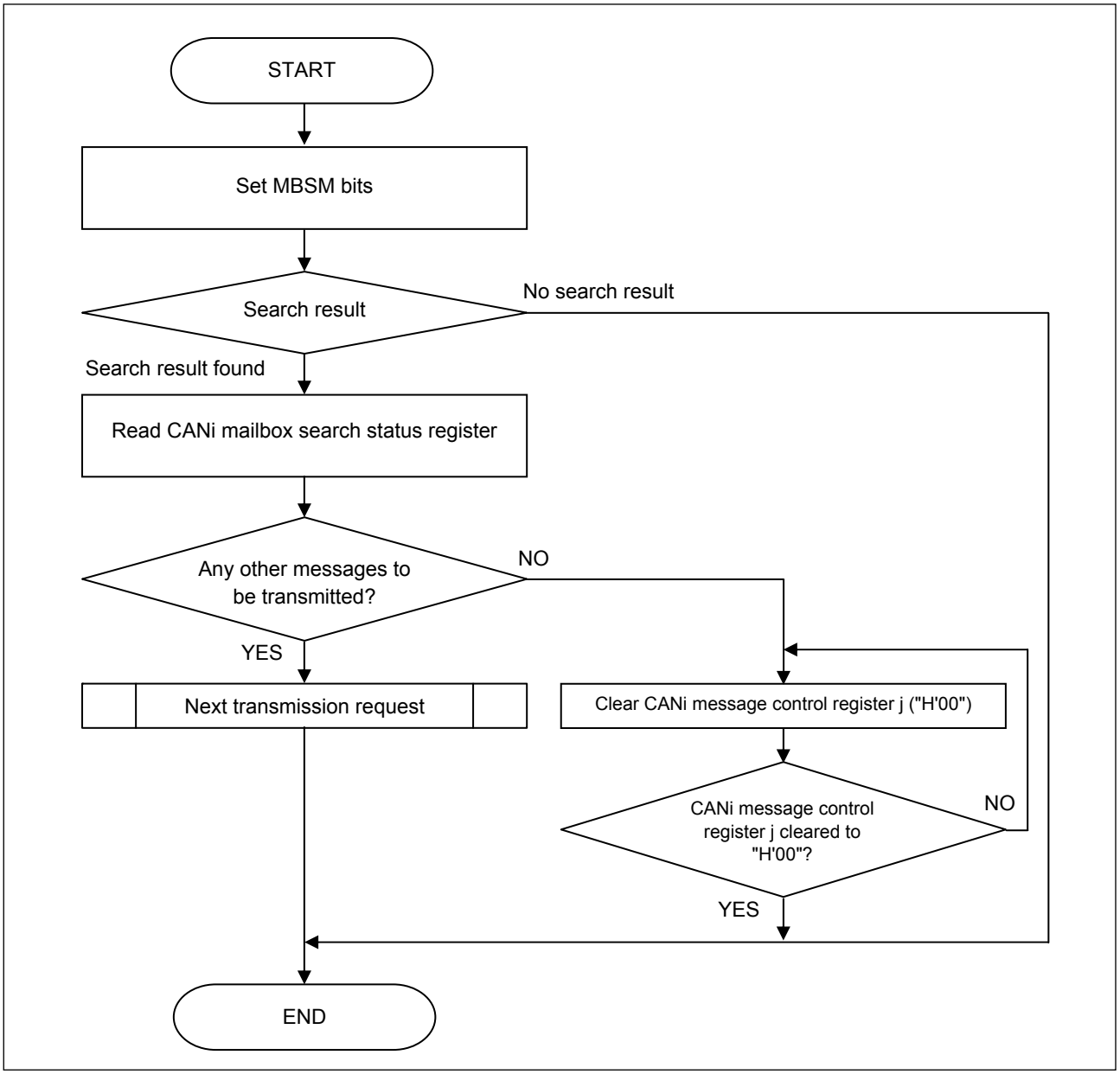



**Figure 4.3 Transmit Mailbox Search**

### 4.1.3 Message Lost Search Mode

This mode searches for the mailbox number that has incurred a message lost.

To use this mode, set the MBSM bits to "10b". The search result can be read from the CANi mailbox search status register (CiMSSR). When two or more mailboxes are in the message lost status (i.e. when two or more mailboxes have set their message lost flag (MSGLOST) or receive FIFO message lost flag (RFMLF) to "1"), the lowest mailbox number is read.

The message lost flag (MSGLOST) is cleared to "0" by software as part of the message lost processing. Then, if there are other mailboxes in the message lost status, the mailbox search function can be used again to read the next mailbox number. When no other mailboxes are in the message lost status, the search result status bit (SEST) is set to "1".

In FIFO mailbox mode, mailbox [60] is read when the receive FIFO message lost flag (RFMLF) is set to "1" (receive FIFO message lost has occurred).

Figure 4.4 shows the procedure for using message lost search.

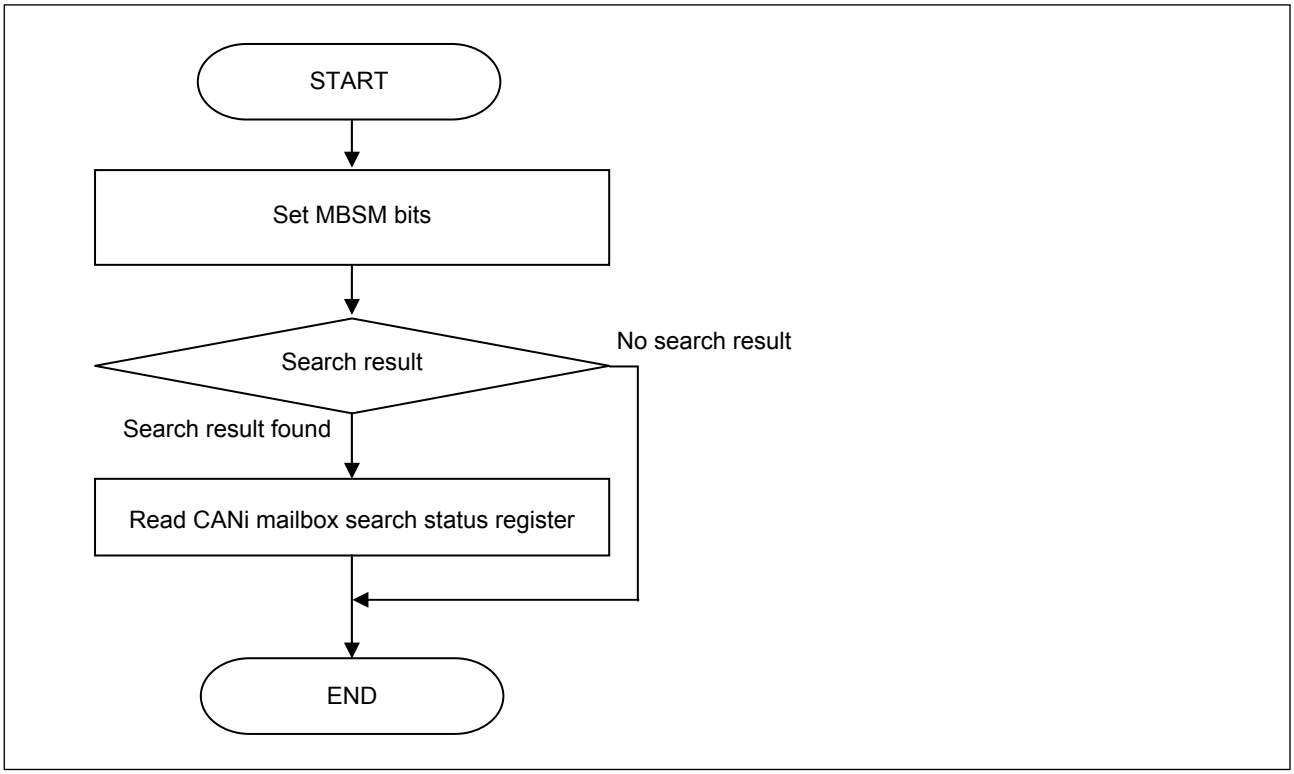



**Figure 4.4 Message Lost Search**

### 4.1.4 Channel Search Mode

The purpose and procedure for using the channel search mode differ from those of the other three modes. This mode does not search for a mailbox number. To use this mode, set the MBSM bits to "11b". Set the channel search value (table value) in the CANi channel search support register (CiCSSR). The encoded value can be read from the CANi mailbox search status register (CiMSSR). When there are two or more channels, the channel numbers are read in order, starting from the lowest.

When the CANi mailbox search status register (CiMSSR) is read, the search result is updated automatically. If there are other channels (bits that become "1"), the next channel number (the bit number) can be read. When there are no other channels (bits that become "1"), the search result status bit (SEST) is set to "1". Figure 4.5 and figure 4.6 show the procedure for using channel search.

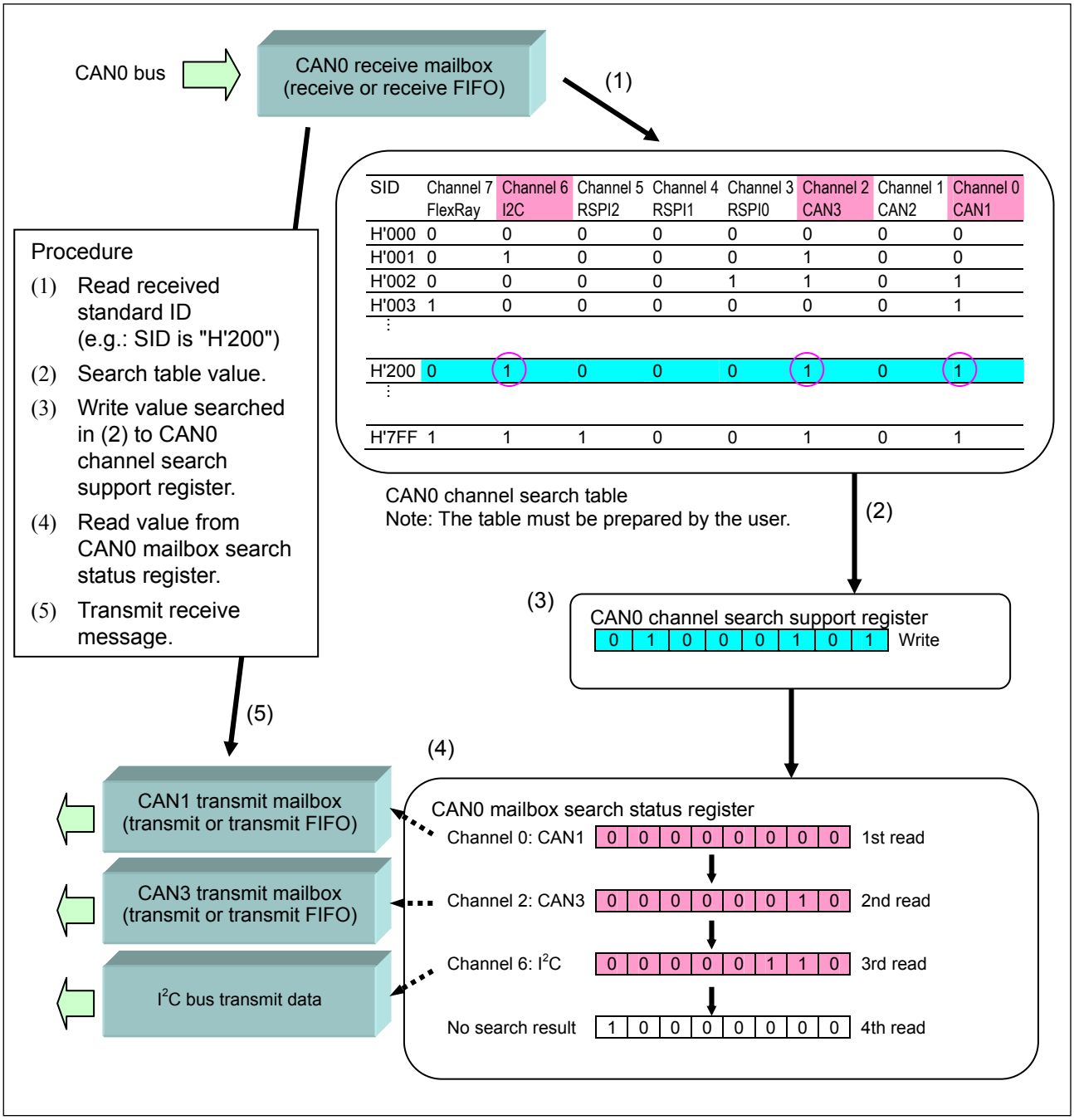



**Figure 4.5 Outline of Channel Search Mode (When CAN0 is Used)**

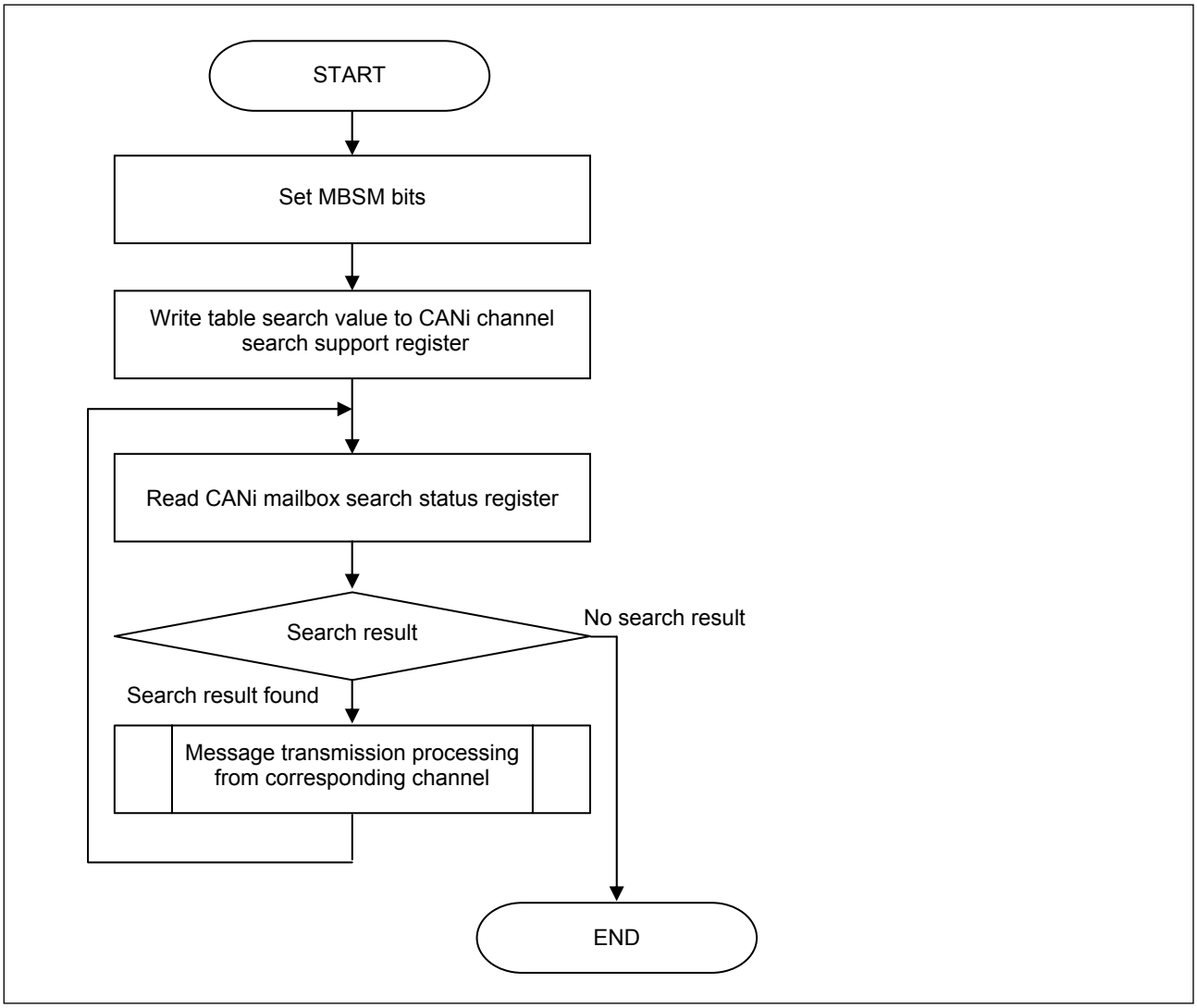



**Figure 4.6 Channel Search**

# 5. CAN Errors

When an error is detected due to a communication frame irregularity during mailbox transmission or reception, the transmit error counter or receive error counter value is incremented, depending on whether the error occurred during transmission or reception. When the transmit error counter or receive error counter value reaches 96 or greater, the error warning detect flag (EWIF) is set to "1". When the transmit error counter or receive error counter value reaches 128 or greater, the CAN status changes from error-active to error-passive, the error passive detect flag (EPIF) is set to "1". When the transmit error counter value reaches 256 or greater, the CAN module enters the bus-off state, and the bus-off entry detect flag (BOEIF) is set to "1".

To use CANi error interrupts, set "1" to the bits in the CANi error interrupt enable register (CiEIER) corresponding to the error interrupts to be used. Whether or not a particular interrupt has occurred can be confirmed by reading the CANi error interrupt factor judge register (CiEIFR). Set the CANi error interrupt enable register (CiEIER) in CAN reset mode. To use CANi error interrupts, set interrupt priority setting register 10, interrupt priority setting register 11, (INT2PRI10 and INT2PRI11), and interrupt mask clear register 1 (INT2MSKCR1) beforehand.

## 5.1 Determining If a CAN Error Has Occurred

**(1) Reading the CANi status register to determine if a CAN error occurred**

Read the error-passive status flag (EPST) and bus-off status flag (BOST) in the CANi status register (CiSTR) to determine if a CAN error occurred. Figure 5.1 shows the procedure for checking for a CANi error by reading the CANi status register (CiSTR).

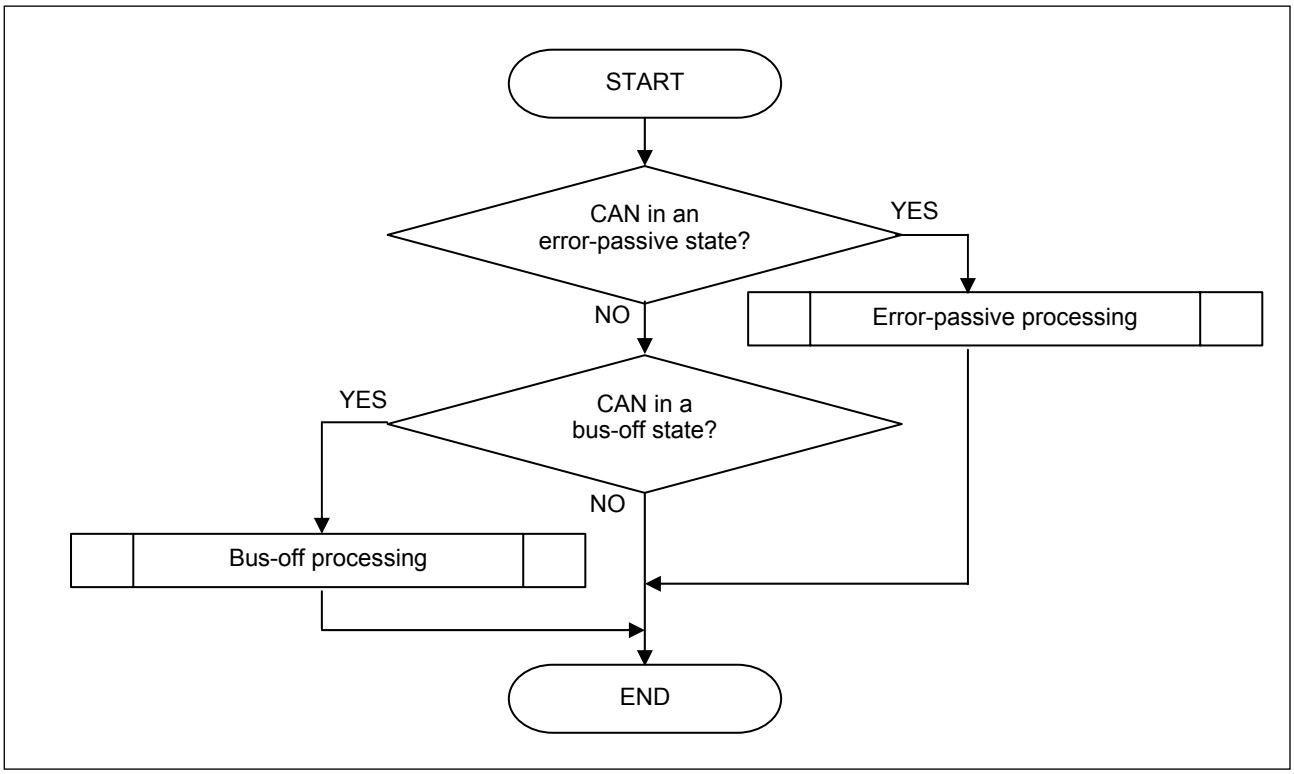
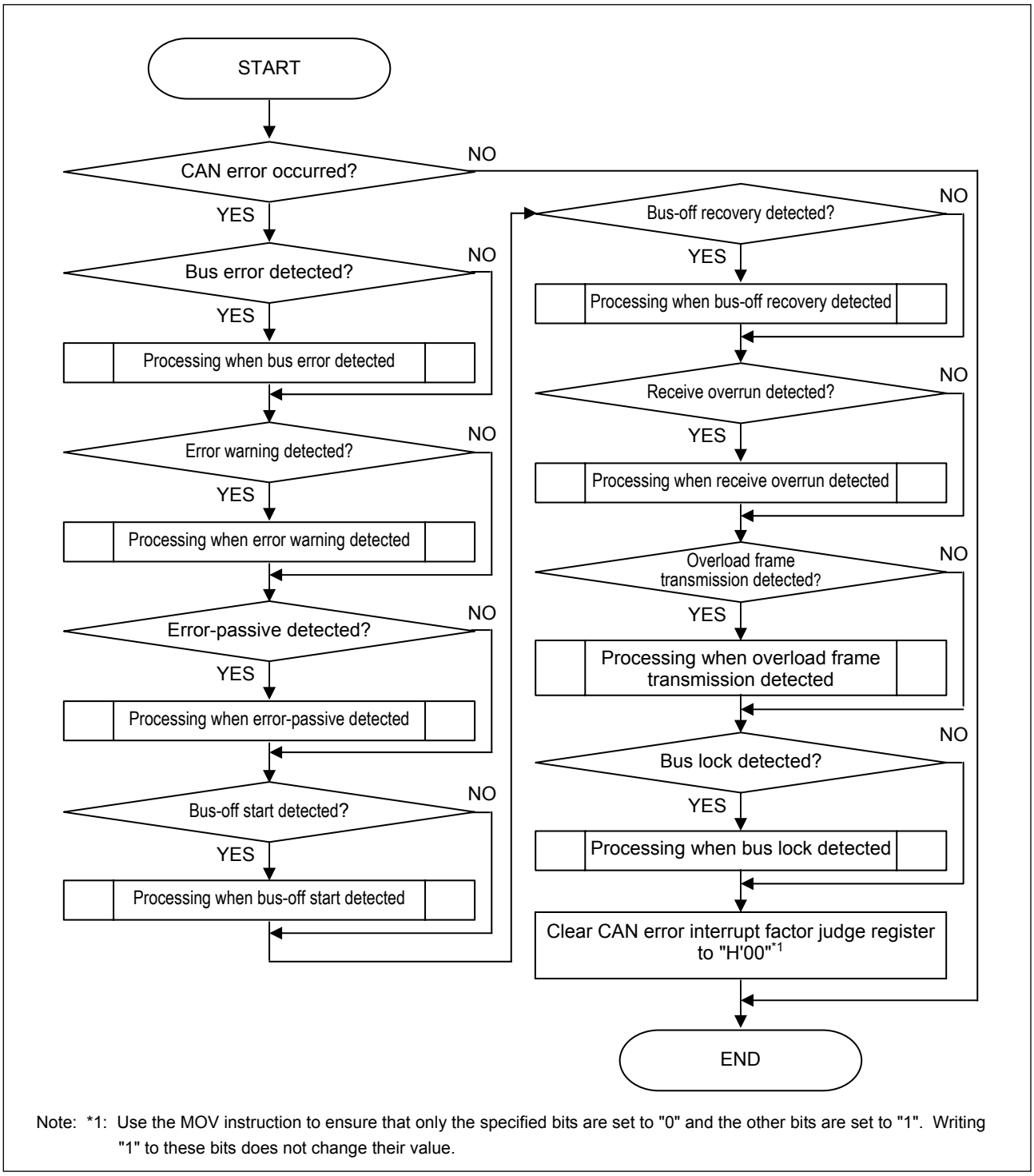



**Figure 5.1 Reading the CANi Status Register to Determine of a CAN Error Has Occurred**

**(2) Reading the CANi error interrupt factor judge register to determine if a CAN error occurred**

CANi error interrupts can be used by setting interrupt priority setting register 10 (INT2PRI10), interrupt priority setting register 11 (INT2PRI11), and interrupt mask clear register 1 (INT2MSKCR1). Read the CANi error interrupt factor judge register (CiEIFR) to determine if a CAN error occurred. Figure 5.2 shows how to read the CANi error interrupt factor judge register (CiEIFR) to determine if a CAN error has occurred. For details on the bus-off recovery, see section 6.




Note: *1: Use the MOV instruction to ensure that only the specified bits are set to "0" and the other bits are set to "1". Writing "1" to these bits does not change their value.

**Figure 5.2 Reading the CANi Error Interrupt Factor Judge Register to Determine if a CAN Error Occurred**

## 6. Bus-Off Recovery Modes

According to the increment and decrement rules of the transmit error counter and receive error counter in the CAN specification (ISO 11898-1), if CAN communication errors occur repeatedly, the CAN module enters the bus-off state and CAN communication cannot be used. The CAN module has five modes for returning from the bus-off state. Table 6.1 lists the description of these modes, their associated registers, and their settings. Figure 6.1 shows transition to and from the bus-off state.

**Table 6.1 Bus-Off Recovery Modes**

| | Name | Description | Bits Used | Bit Setting |
|---|---|---|---|---|
| (1) | Normal mode | After returning from the bus-off state, the CAN module enters the error-active state and CAN communication can be used[*1*2] | BOM[*5] | "00b"[*6] |
| (2) | Forcible return from bus-off | The CAN module enters the error-active state, and CAN communication can be used[*3] | BOM[*5] | "00b"[*6] |
| | | | RBOC[*5] | "1"[*7] |
| (3) | Entry to CAN halt mode automatically at bus-off entry | The CAN module enters CAN halt mode immediately when it reaches the bus-off state[*3] | BOM[*5] | "01b"[*6] |
| (4) | Entry to CAN halt mode automatically at bus-off end | The CAN module enters CAN halt mode after recovering from the bus-off state[*1*2] | BOM[*5] | "10b"[*6] |
| (5) | Entry to CAN halt mode by a program request | When the CAN module is in the bus-off state, it enters CAN halt mode by setting the CAN operating mode select bits to "10b" (CAN halt mode)[*3*4] | BOM[*5] | "11b"[*6] |
| | | | CANM[*5] | "10b"[*8] |

Notes: *1: The CAN module returns from the bus-off state after detecting 11 consecutive recessive bits 128 times.

*2: The bus-off recovery detect flag in the CANi error interrupt factor judge register is set to "1" (bus-off recovery detected).

*3: The bus-off recovery detect flag is not set to "1".

*4: When the CAN operating mode select bit is not set to "10b" (CAN halt mode) while the CAN module is in the bus-off state, operation is identical to (1).

*5: Bit in the CANi control register.

*6: Perform this setting in CAN reset mode.

*7: Perform this setting in the bus-off state. After the forcible return from bus-off bit is set to "1" by a program, it is automatically cleared to "0".

*8: After changing the CAN operating mode select bit, read the CANi status register.

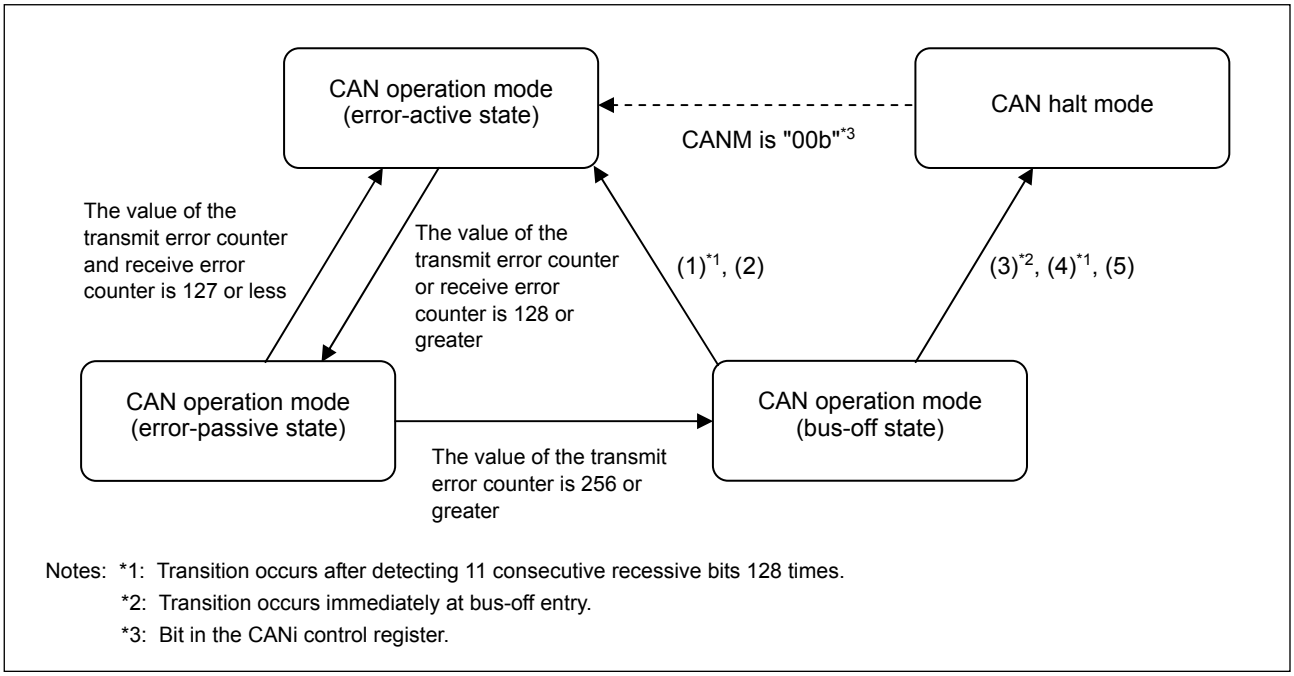



**Figure 6.1 Transition To and From the Bus-Off State**

# 7. Using the Acceptance Filter

An acceptance filter determines whether messages are received or discarded in hardware.

## 7.1 Standard ID and Extended ID

The CAN module transmits and receives both an 11-bit standard ID and a 29-bit extended ID. Figure 7.1 shows bit maps of the standard ID and extended ID.

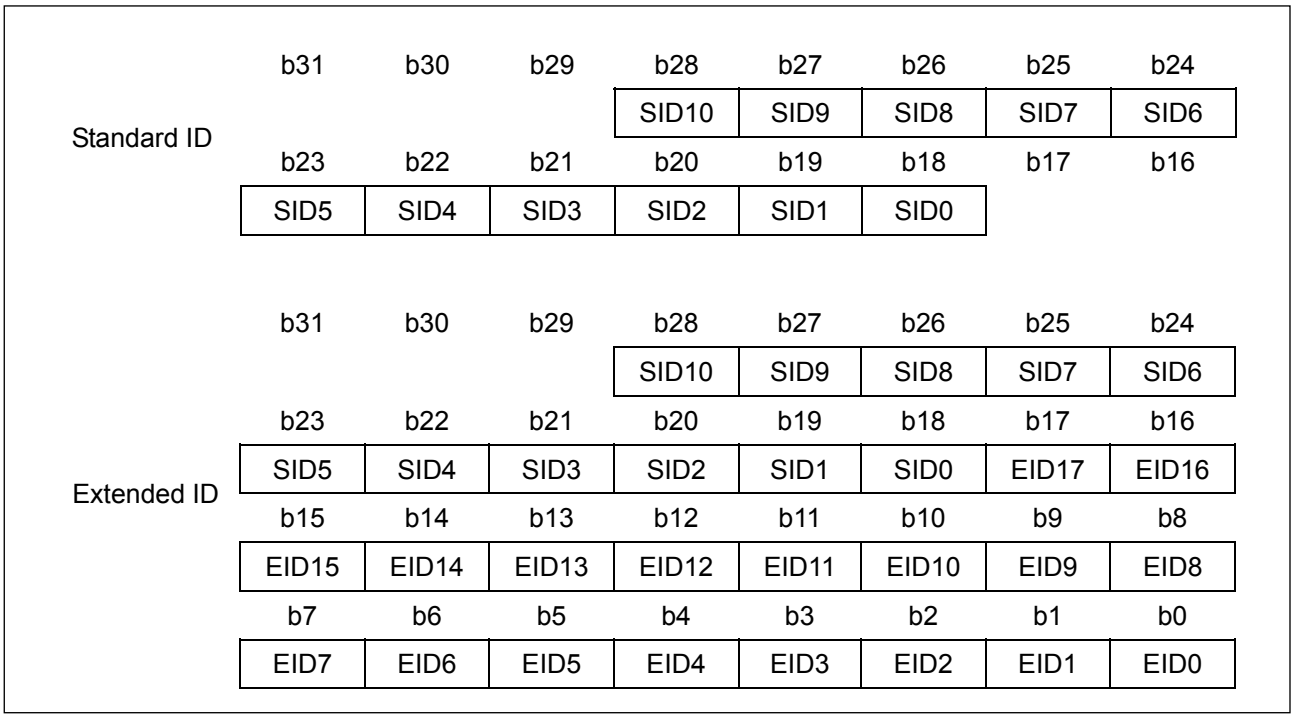


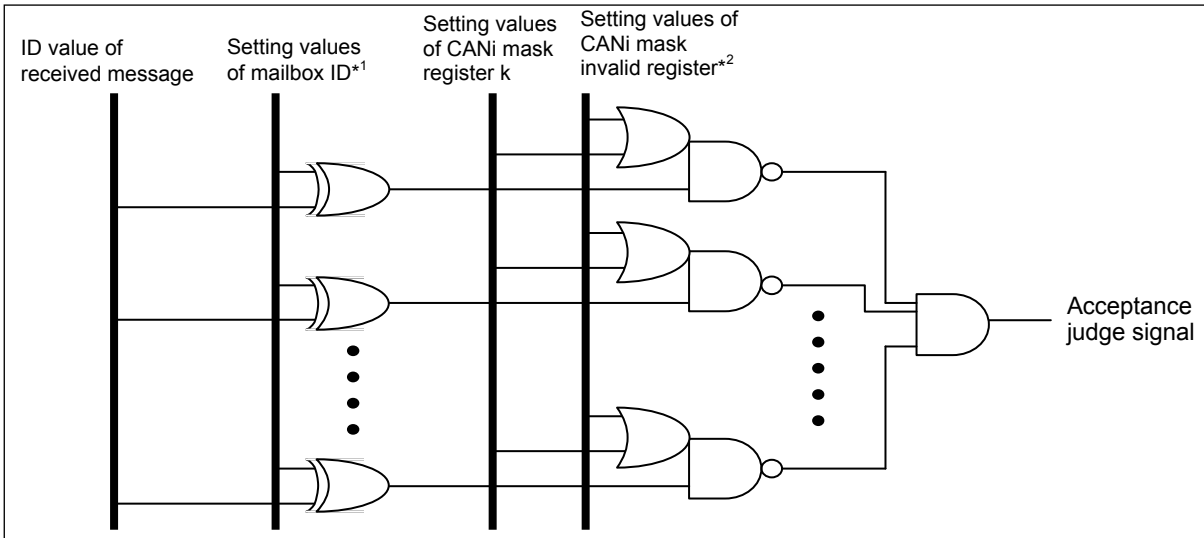

**Figure 7.1 Bit Maps of the Standard ID and Extended ID**

## 7.2 Acceptance Filter

The acceptance filter uses 10 registers (CANi mask register k (CiMKRk)) for filtering. For details on using the acceptance filter in FIFO mailbox mode, refer to section 7.3.

(1) Acceptance Filter Register Configuration

Figure 7.2 shows the ID and mask register configuration, table 7.1 lists the CAN acceptance filter processing, and figure 7.3 shows a bitmap.



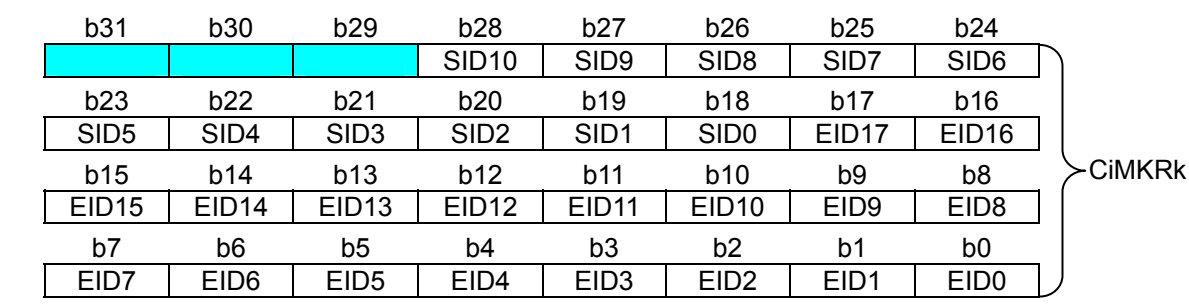

*1: In FIFO mailbox mode, the values set in the CANi FIFO received ID compare registers (CiFIDCR0, CiFIDCR1) are used.
*2: Invalid in FIFO mailboxes.

**Figure 7.2 ID and Mask Register Configuration**

**Table 7.1 CAN Acceptance Filter Processing**

| | |
|---|---|
| **Setting values of mailbox ID** | Set a message ID to receive |
| **Setting values of CANi mask register k** | 0: Corresponding receive message ID bit is not compared<br>1: Corresponding receive message ID bit is compared to corresponding mailbox ID bit*1 |
| **Setting values of CANi mask invalid register** | 0: Mask valid<br>1: Mask invalid |
| **Acceptance judge signal** | 0: Discard message<br>1: Receive message |

Note: *1: The ID extension bit (IDE) and remote transmission request bit (RTR) cannot be masked. A receive RTR is always compared to a mailbox RTR. The ID extension bit (IDE) is compared when the ID format mode select bit (IDFM) is set to mixed ID mode.




i: CAN channel
k: Mask register number

**Figure 7.3 Bitmap**

(2) Acceptance Filter Usage Examples

(a) Usage example 1

Table 7.2 lists the register settings for receiving a standard data frame with ID "H'123" in mailbox [0].

**Table 7.2 Acceptance Filter Usage Example 1**

| | | IDE*1, RTR, SID10 to SID6 | SID5 to SID0, EID17 to EID16 | EID15 to EID8 | EID7 to EID0 |
|---|---|---|---|---|---|
| Mailbox [0] | | 00-00100 | 100011XX | XXXXXXXX | XXXXXXXX |
| Mask register | CiMKR0 | - --11111 | 111111XX | XXXXXXXX | XXXXXXXX |
| Received message | ID "H"123 | 00-00100 | 100011- - | - - - - - - - - | - - - - - - - - |

Note: *1: The ID extension bit (IDE) is valid when ID format mode select bit (IDFM) in CANi control register is set to mixed ID mode. Otherwise, set the ID extension bit to "0".

(b) Usage example 2

Table 7.3 lists the register settings for receiving a standard remote frame with ID "H'123" in mailbox [0].

**Table 7.3 Acceptance Filter Usage Example 2**

| | | IDE*1, RTR, SID10 to SID6 | SID5 to SID0, EID17 to EID16 | EID15 to EID8 | EID7 to EID0 |
|---|---|---|---|---|---|
| Mailbox [0] | | 01-00100 | 100011XX | XXXXXXXX | XXXXXXXX |
| Mask register | CiMKR0 | - --11111 | 111111XX | XXXXXXXX | XXXXXXXX |
| Received message | ID "H"123 | 01-00100 | 100011- - | - - - - - - - - | - - - - - - - - |

Note: *1: The ID extension bit (IDE) is valid when ID format mode select bit (IDFM) in CANi control register is set to mixed ID mode. Otherwise, set the ID extension bit to "0".

(c) Usage example 3

Table 7.4 lists the register settings for receiving two standard data frames with IDs "H'122" and "H'123" in mailbox [0].

**Table 7.4 Acceptance Filter Usage Example 3**

| | | IDE*1, RTR, SID10 to SID6 | SID5 to SID0, EID17 to EID16 | EID15 to EID8 | EID7 to EID0 |
|---|---|---|---|---|---|
| Mailbox [0] | | 00-00100 | 100011XX | XXXXXXXX | XXXXXXXX |
| Mask register | CiMKR0 | - --11111 | 111110XX | XXXXXXXX | XXXXXXXX |
| Received message | ID "H'122" | 00-00100 | 100010- - | - - - - - - - - | - - - - - - - - |
| | ID "H'123" | 00-00100 | 100011- - | - - - - - - - - | - - - - - - - - |

Note: *1: The ID extension bit (IDE) is valid when ID format mode select bit (IDFM) in CANi control register is set to mixed ID mode. Otherwise, set the ID extension bit to "0".

(d) Usage example 4

Table 7.5 lists the register settings for receiving an extended data frame with ID "H'12345678" in mailbox [0].

**Table 7.5 Acceptance Filter Usage Example 4**

| | | IDE*1, RTR, SID10 to SID6 | SID5 to SID0, EID17 to EID16 | EID15 to EID8 | EID7 to EID0 |
|---|---|---|---|---|---|
| Mailbox [0] | | 10-10010 | 00110100 | 01010110 | 01111000 |
| Mask register | CiMKR0 | - --11111 | 11111111 | 11111111 | 11111111 |
| Received message | ID "H'12345678" | 10-10010 | 00110100 | 01010110 | 01111000 |

Note: *1: The ID extension bit (IDE) is valid when ID format mode select bit (IDFM) in CANi control register is set to mixed ID mode. Otherwise, set the ID extension bit to "0".

(e) Usage example 5

Table 7.6 lists the register settings for receiving an extended remote frame with ID "H'12345678" in mailbox [0].

**Table 7.6 Acceptance Filter Usage Example 5**

| | | IDE*1, RTR, SID10 to SID6 | SID5 to SID0, EID17 to EID16 | EID15 to EID8 | EID7 to EID0 |
|---|---|---|---|---|---|
| Mailbox[0] | | 11-10010 | 00110100 | 01010110 | 01111000 |
| Mask register | CiMKR0 | - --11111 | 11111111 | 11111111 | 11111111 |
| Received message | ID "H'12345678" | 11-10010 | 00110100 | 01010110 | 01111000 |

Note: *1: The ID extension bit (IDE) is valid when ID format mode select bit (IDFM) in CANi control register is set to mixed ID mode. Otherwise, set the ID extension bit to "0".

## 7.3 Acceptance Filter for the Receive FIFO

This acceptance filter is used with FIFO mailbox mode. Two acceptance masks can be applied to the receive FIFO. This extends the range of IDs that can be received by the receive FIFO. This acceptance filter mode uses the two mask registers (CiMKR8 and CiMKR9) and two FIFO receive ID compare registers (CiFIDCR0 and CiFIDCR1). In this acceptance filter mode, the IDs of received messages are compared to the CANi FIFO receive ID compare registers (CiFIDCR0 and CiFIDCR1) instead of each mailbox ID. Figure 7.4 shows the ID and mask register configuration.

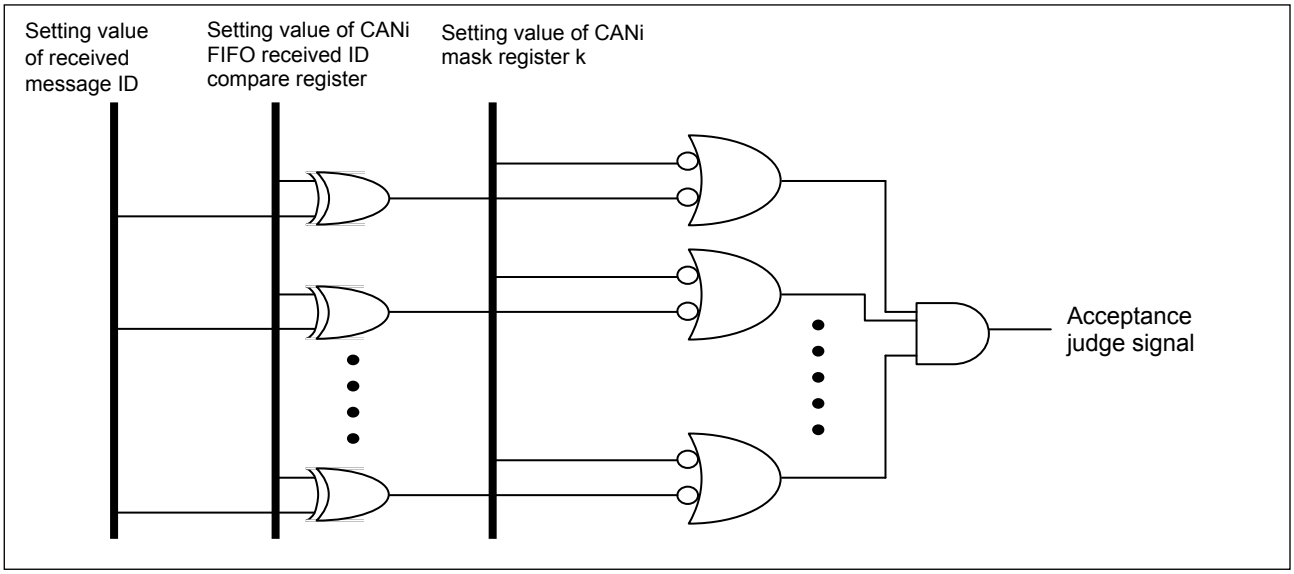



**Figure 7.4 FIFOi Receive ID Compare Register and CANi Mask Register k Configuration**

**Table 7.7 CAN FIFO Acceptance Filter Processing**

| | |
|---|---|
| **CANi FIFO Received ID Compare Register** | Set a message ID to receive |
| **CANi mask register k** | 0: Corresponding receive message ID bit is not compared<br>1: Corresponding receive message ID bit is compared to corresponding ID bit in CAN FIFO receive ID compare register*1 |
| **Acceptance judge signal** | 0: Discard message<br>1: Receive message |

Note: *1: The ID extension bit (IDE) and remote transmission request bit (RTR) cannot be masked. A receive RTR is always compared to a RTR in the FIFO received ID compare register. The ID extension bit (IDE) is compared when the ID format mode select bit (IDFM) is set to mixed ID mode.

## 7.4 Acceptance Filter Support Unit

The acceptance filter support unit can be used for data table (8 bits × 256) search. In the data table, all standard IDs created by the user are set to be valid/invalid in bit units. First, register the IDs to be received in the data table. Next, store a reception completed ID in the CANi acceptance filter support register (CiAFSR), read the received ID that is decoded from the CANi acceptance filter support register (CiAFSR), and search the data table. The acceptance filter support unit can only be used with standard frame IDs. Using the acceptance filter support unit is effective in the following situations:

- When the ID to receive cannot be masked by the acceptance filter.
  (e.g.: IDs to receive "H'078", "H'087", and "H'111")
- When there are too many IDs to receive and software filtering would take an excessive amount of time

### 7.4.1 Using the Acceptance Filter Support Unit

This example shows how to use the acceptance filter support unit when receiving IDs "H'000", "H'00D", "H'6F3", "H'6F4", and "H'6FF".

(1) Data Table Settings

Create a data table in the ROM or RAM for registering IDs to be received. The data table can be mapped to any addresses. In the data table, the vertical axis is the value of the upper 8 bits (SID10 to SID3) of the ID to be received, and the horizontal axis is the value, which is decoded the lower 3 bits (SID2 to SID0) into 8 bits. Set bits that correspond to receive IDs to "1"; set other bits to "0".

(2) Writing to the CANi Acceptance Filter Support Register (CiAFSR)

When CANi receives a message, the receive ID is written is written to the CANi acceptance filter support register (CiAFSR).

(3) Reading from the CANi Acceptance Filter Support Register (CiAFSR)

When reading the CANi acceptance filter support register (CiAFSR), read the value of the upper 8 bits of the receive ID (SID10 to SID3), and read the value of the 8 bits which is decoded from the lower 3 bits of the receive ID (SID2 to SID0).

(4) Determining Validity of Received IDs

Using the values read from the CANi acceptance filter support register (CiAFSR) in step (3), search the data table created in step (1) to determine whether the message is valid or invalid.

Figure 7.5 shows the data table configuration and figure 7.6 shows the states when writing to and reading from the CANi acceptance filter support register (CiAFSR).

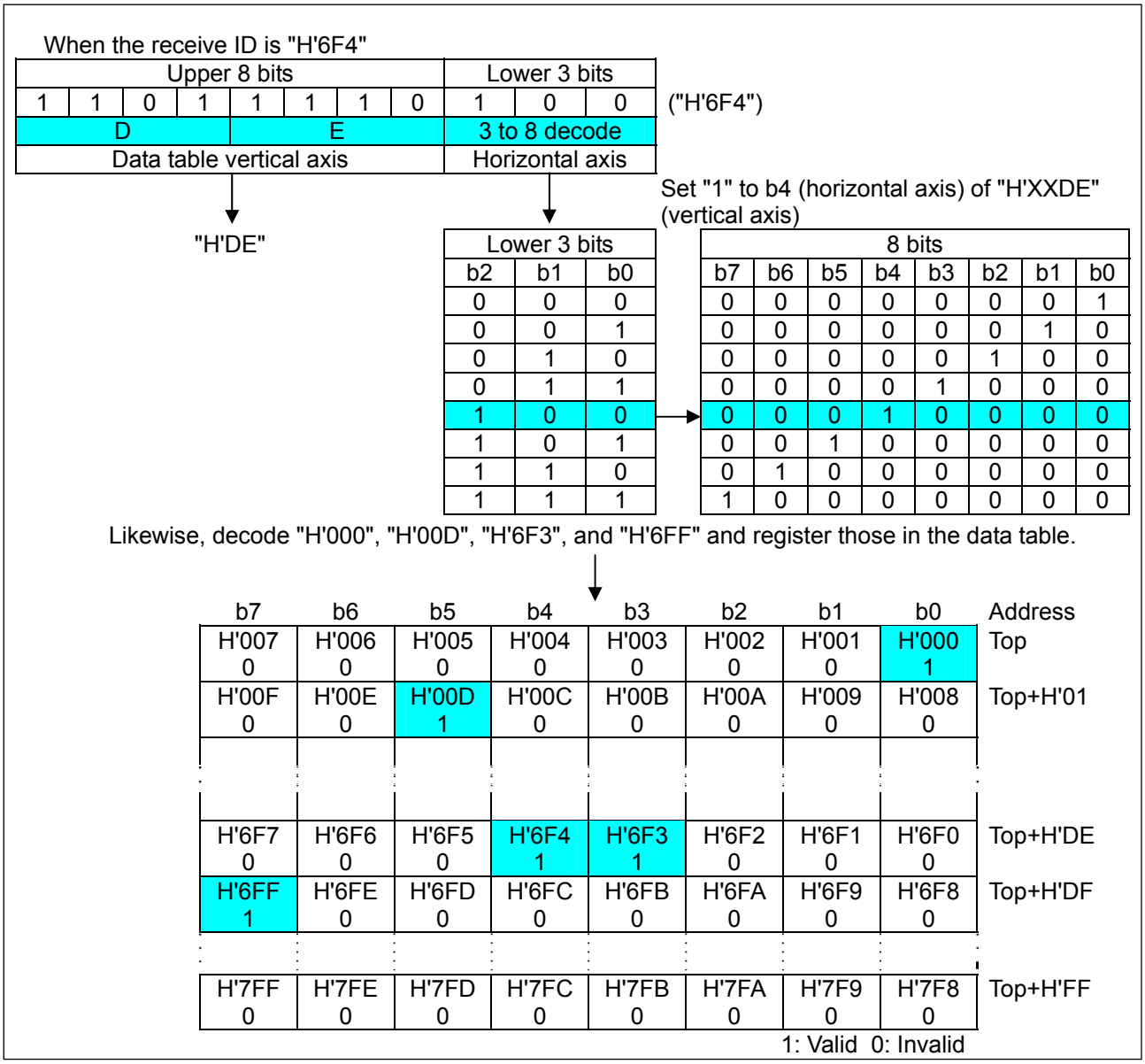



**Figure 7.5 Data Table Configuration (Example of Data Table Creation)**

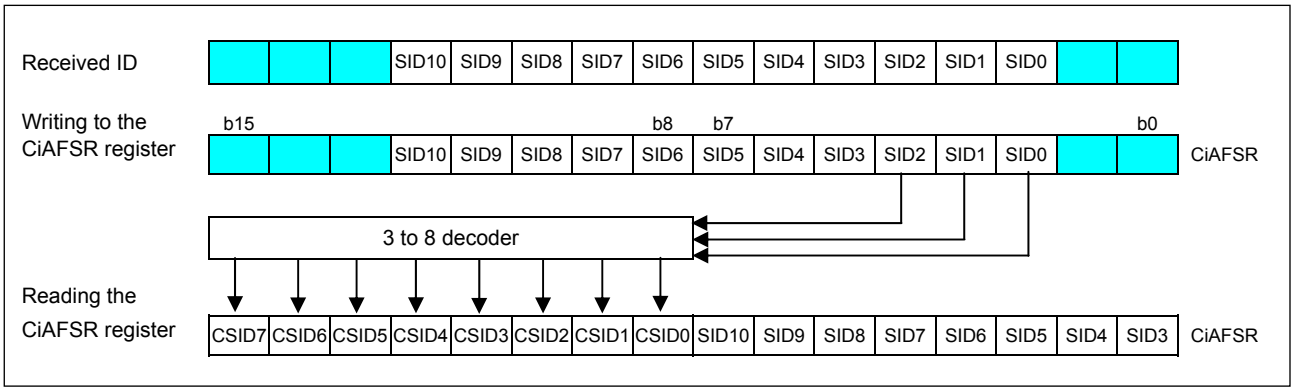
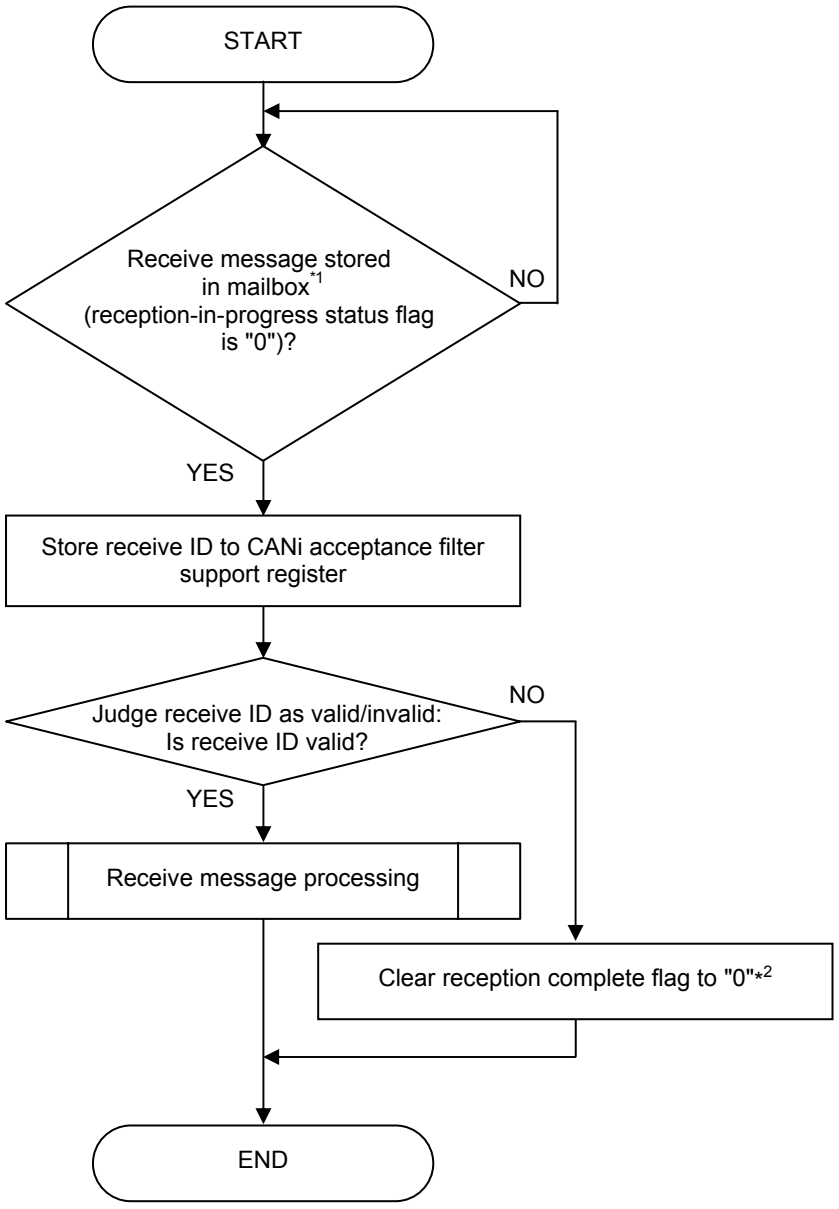



**Figure 7.6 States When Writing to and Reading from the CANi Acceptance Filter Support Register**

Figure 7.7 shows the acceptance filter support unit operating procedures.




Notes: *1: In overrun mode or one-shot receive mode, this processing is not required.

*2: When writing to the reception complete flag, transmission complete flag, message lost flag, transmission abort complete flag, receive mailbox request bit, and the transmit mailbox request bit by a program, use the MOV instruction to ensure that only the specified bits are set to "0" and the other bits are set to "1".

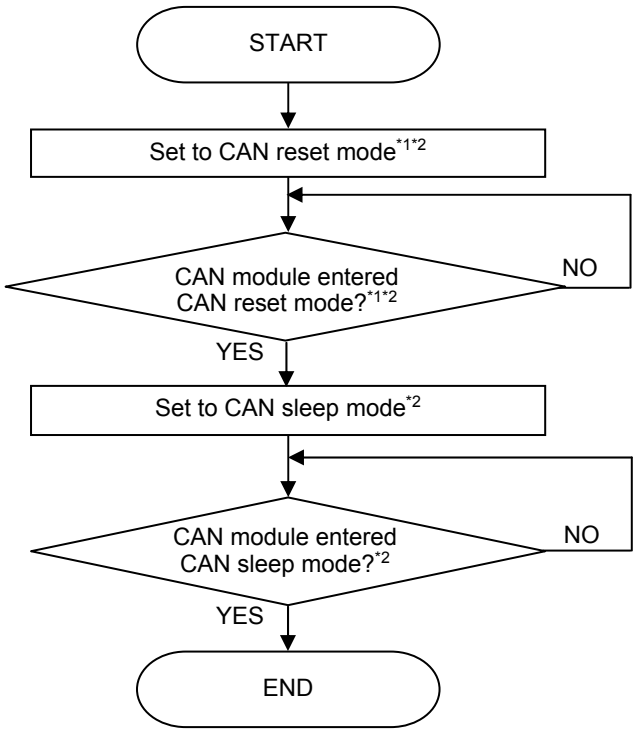
**Figure 7.7 Acceptance Filter Support Unit Operation**

# 8. CAN Sleep Operation and CAN Wakeup Operation

## 8.1 CAN Sleep Operation

When the CAN module is in CAN sleep mode, no clock is supplied to it. Hence, the CAN module does not operate at all. When the CAN module is not in use, setting it to CAN sleep mode is recommended to reduce current consumption. Before entering CAN sleep mode, switch the CAN module to CAN reset mode or CAN halt mode.

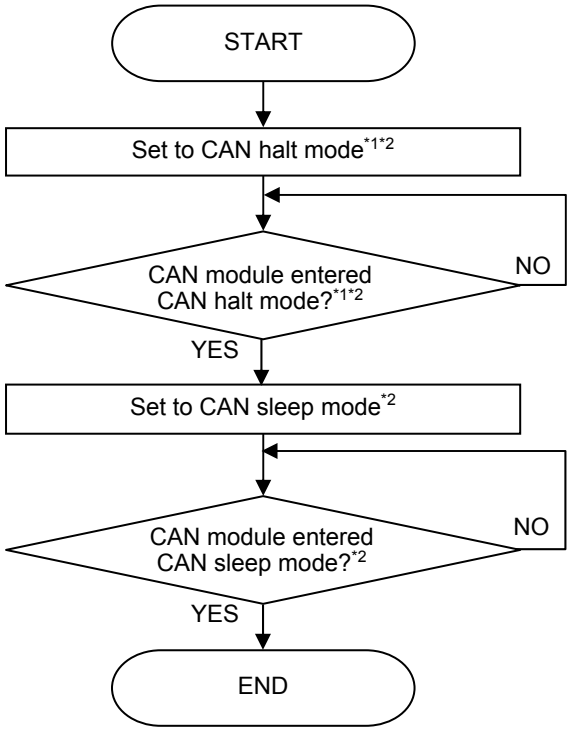
Figure 8.1 shows the procedure for switching the CAN module to CAN reset mode and then to CAN sleep mode, and figure 8.2 shows the procedure for switching the CAN module to CAN halt mode and then to CAN sleep mode.




Notes: *1: When the CAN module is transmitting, even if it enters CAN reset mode, it does not enter CAN reset mode until the transmit operation is completed.

*2: When the CAN operating mode select bit and CAN sleep mode bit are changed, read the CANi status register to ensure that the mode has been switched.

**Figure 8.1 Switching the CAN Module to CAN Reset Mode and then to CAN Sleep Mode**




Notes: *1: When the CAN module is transmitting or receiving, even if it enters CAN halt mode, it does not enter CAN halt mode until transmit or receive operation is completed.

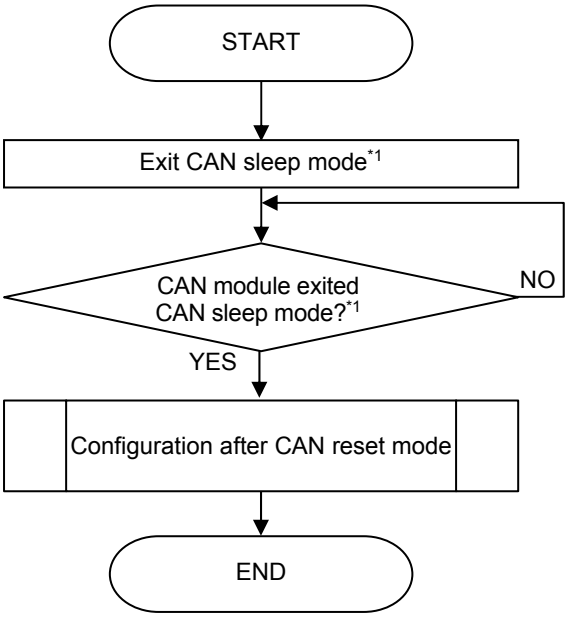
*2: When the CAN operating mode select bit and CAN sleep mode bit are changed, read the CANi status register to ensure that the mode has been switched.

**Figure 8.2 Switching the CAN Module to CAN Halt Mode and then to CAN Sleep Mode**

## 8.2 CAN Wakeup Operation

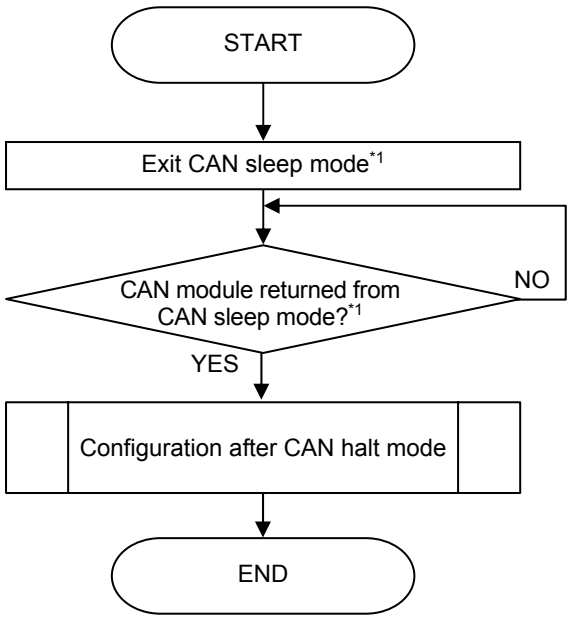
When the CAN module is in CAN sleep mode, exit CAN sleep mode by clearing CAN sleep mode bit in the CANi control register (CiCTLR) to "0". After exiting CAN sleep mode, the CAN module re-enters the mode (CAN reset mode or CAN halt mode) it was in before entering CAN sleep mode.

Figure 8.3 shows the CAN wakeup procedure when the CAN module was switched to CAN sleep mode from CAN reset mode, and figure 8.4 shows the CAN wakeup procedure when the CAN module was switched to CAN sleep mode from CAN halt mode.




Note: *1: When the CAN sleep mode bit is changed read the CANi status register to ensure that the mode has been switched.

**Figure 8.3 CAN Wakeup in CAN Reset Mode**




Note: *1: When the CAN sleep mode bit is changed, read the CANi status register to ensure that the mode has been switched.

**Figure 8.4 CAN Wakeup in CAN Halt Mode**

# 9. Test Modes

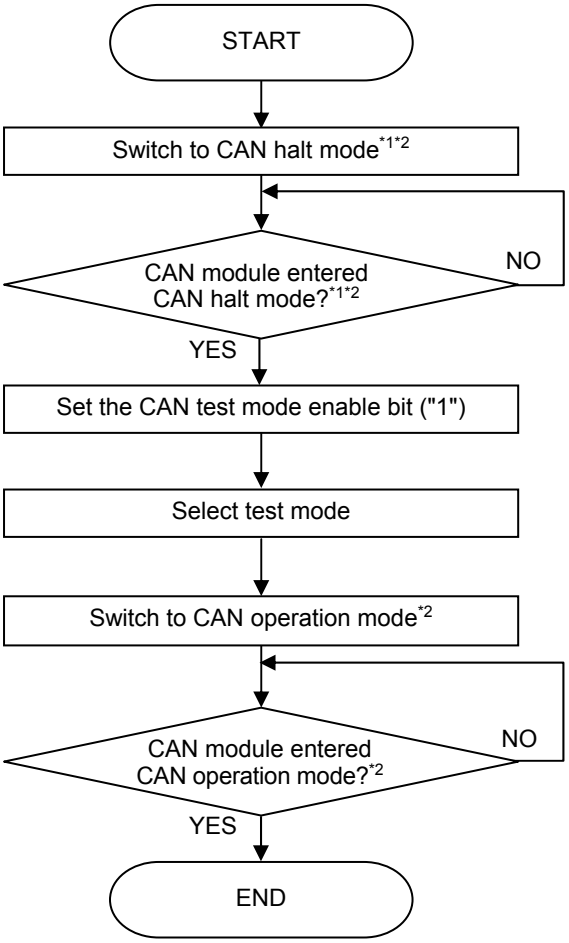
The following three test modes are provided for evaluation by the user:

- Listen-only mode
- Self-test mode 0 (external loop back)
- Self-test mode 1 (internal loop back)

Select a test mode when the CAN module is in CAN halt mode.

## 9.1 Test Mode Setting

Figure 9.1 shows the procedure for selecting a test mode.




Notes: *1: When the CAN module is transmitting or receiving, even if it enters CAN halt mode, it does not enter CAN halt mode until the transmit operation or receive operation is completed.

*2: When the CAN operating mode select bit is changed, read the CANi status register to ensure that the mode has been switched.

**Figure 9.1 Test Mode Setting**

## 9.2 Listen-Only Mode

The CAN specification (ISO 11898-1) recommends an optional bus monitoring mode. In listen-only mode, the CAN node is able to receive valid data frames and valid remote frames, send only recessive bits on the CAN bus, and the protocol controller is not required to send the ACK bit, overload flag, or active error flag. Listen-only mode can be used for baud rate detection. Do not request transmission from any mailboxes in listen-only mode. Set the CAN test mode select bits (TSTM) to "01b" to select listen-only mode. Figure 9.2 shows listen-only mode selected.

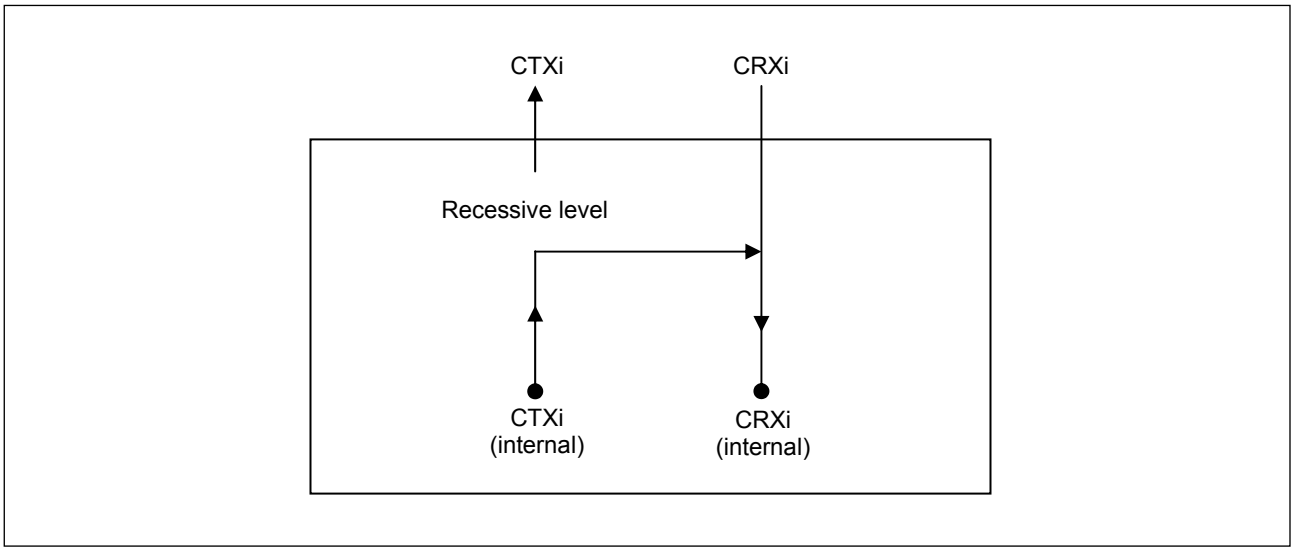



**Figure 9.2 Connection when Listen-Only Mode is Selected**

## 9.3 Self-Test Mode 0 (External Loop Back)

Self-test mode 0 is provided for CAN transceiver tests. In this mode the protocol controller treats its own transmitted messages as messages received via the CAN transceiver and stores them into the receive mailbox. To be independent from external stimulation, the protocol controller generates the ACK bit. Connect the CTXi and CRXi pins to the CAN transceiver. Set the CAN test mode select bits (TSTM) to "10b" to select self-test mode 0. Figure 9.3 shows self-test mode 0 selected.

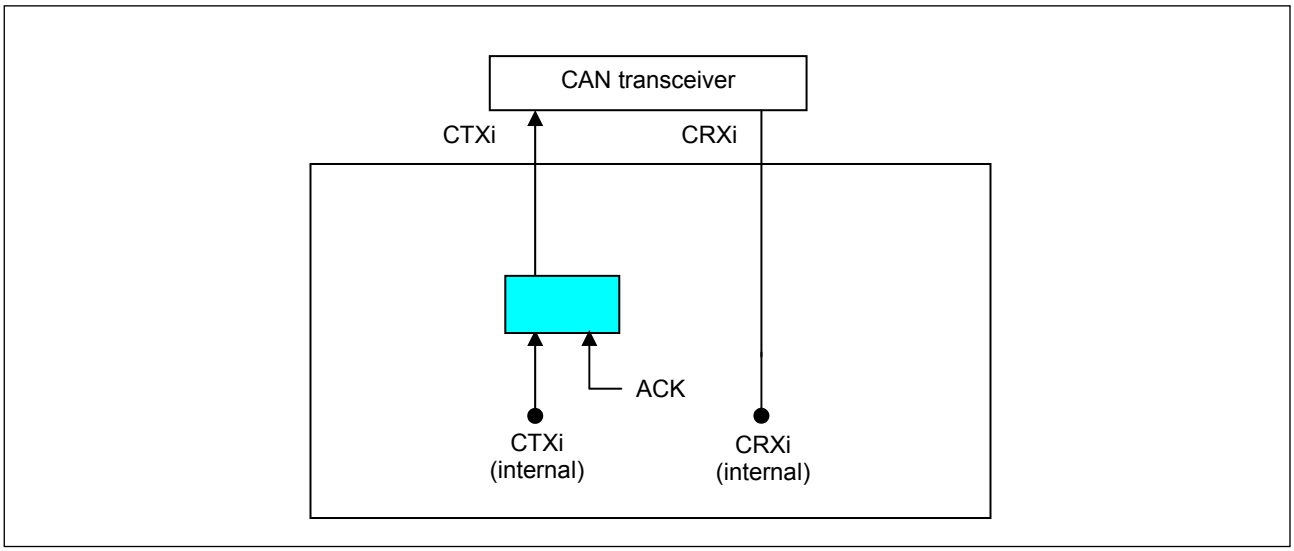



**Figure 9.3 Connection when Self-Test Mode 0 is Selected**

## 9.4 Self-Test Mode 1 (Internal Loop Back)

Self-test mode 1 is provided for self-test functions. In this mode, the protocol controller treats its transmitted messages as received messages and stores them into the receive mailbox. To be independent from external stimulation, the protocol controller generates the ACK bit. In self-test mode 1, the protocol controller performs an internal feedback from the internal CTXi pin to the internal CRXi pin. The input value of the external CRXi pin is ignored. The external CTXi pin outputs only recessive bits. The CTXi and CRXi pins do not need to be connected to the CAN bus or any external device. Set the CAN test mode select bits (TSTM) to "11b" to select self-test mode 1. Figure 9.4 shows self-test mode 1 selected.

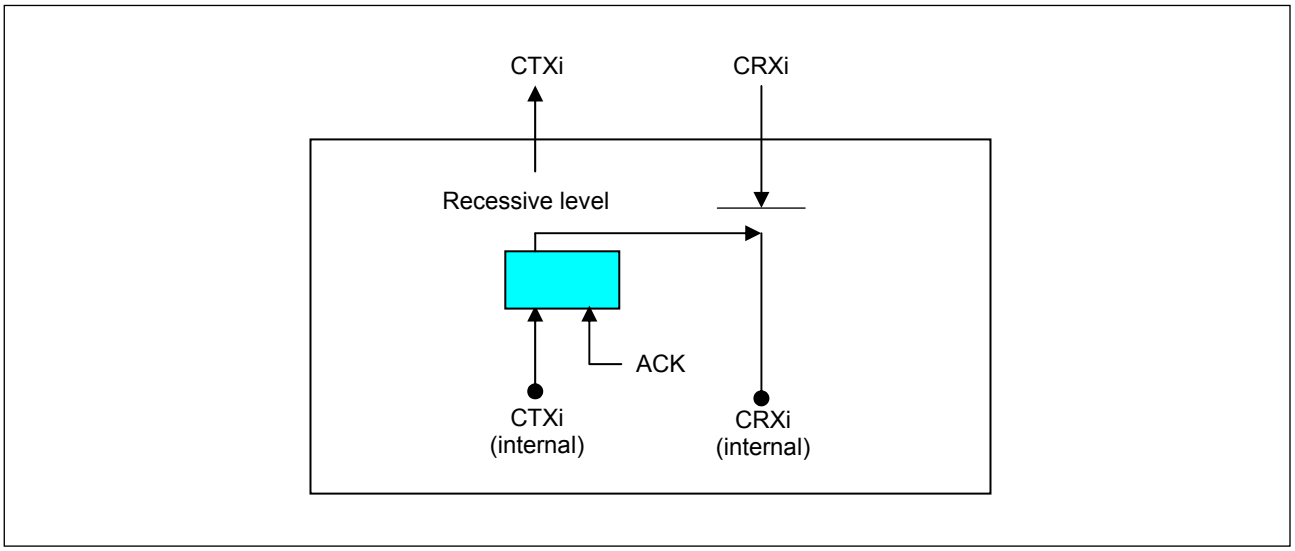


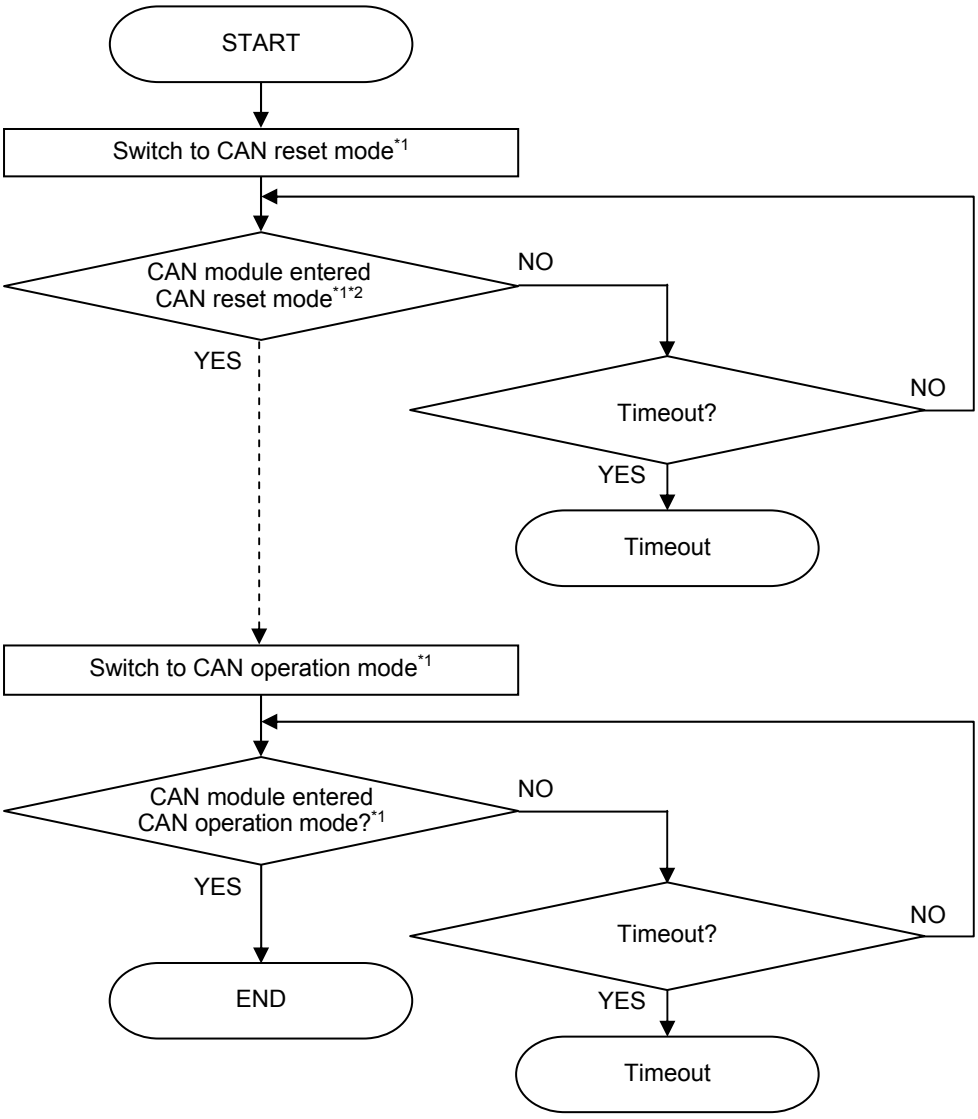

**Figure 9.4 Connection when Self-Test Mode 1 is Selected**

# 10. Notes on the Processing Flow

## 10.1 Infinite Loops

Since the description is simplified, there are infinite loops in the processing flow. When creating the user program, set up a time limit in each loop and exit the loop when the time limit has passed. Figure 10.1 shows a processing example of a loop with the time limit.

Configuration after CAN reset mode




Notes: *1: When the CAN operating mode select bit is changed, read the CANi status register to ensure that the mode has been switched. Do not change the CAN operating mode select bit until the mode has been switched.

*2: When the CAN module is switched to CAN reset mode during a transmit operation, the time limit for the transition to CAN reset mode during transmission or reception is up to about 160 bits (maximum number of bits in extended data frame: 130 bits + maximum number of stuff bits: 30 bits ) per frame to complete a transmission and reception.

Example: Time necessary to transmit and receive 1 frame (160 bits) at 500 kbps is 320 µs. After the time limit of 320 µs has elapsed, exit the loop.

**Figure 10.1 Processing Example of Loop with Time Limit**

## Website and Support

Renesas Electronics Website

http://www.renesas.com/

Inquiries

http://www.renesas.com/inquiry

| Revision History | SH7450 Group, SH7451 Group<br>CAN Application Note |
|---|---|

| Rev. | Date | Description | |
|---|---|---|---|
| | | Page | Summary |
| 1.00 | Feb. 23, 2012 | — | First edition issued |
| | | | |

## General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this manual, refer to the relevant sections of the manual. If the descriptions under General Precautions in the Handling of MPU/MCU Products and in the body of the manual differ from each other, the description in the body of the manual takes precedence.

1. Handling of Unused Pins
   Handle unused pins in accord with the directions given under Handling of Unused Pins in the manual.
   The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.
2. Processing at Power-on
   The state of the product is undefined at the moment when power is supplied.
   The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.
   In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed.
   In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.
3. Prohibition of Access to Reserved Addresses
   Access to reserved addresses is prohibited.
   The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.
4. Clock Signals
   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized.
   When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.
5. Differences between Products
   Before changing from one product to another, i.e. to one with a different type number, confirm that the change will not lead to problems.
   The characteristics of MPU/MCU in the same group but having different type numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different type numbers, implement a system-evaluation test for each of the products.

# 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.

RENESAS

## SALES OFFICES

Renesas Electronics Corporation

http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.


**Renesas Electronics America Inc.**
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

**Renesas Electronics Canada Limited**
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

**Renesas Electronics Europe Limited**
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

**Renesas Electronics Europe GmbH**
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

**Renesas Electronics (China) Co., Ltd.**
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

**Renesas Electronics (Shanghai) Co., Ltd.**
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

**Renesas Electronics Hong Kong Limited**
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

**Renesas Electronics Taiwan Co., Ltd.**
13F, No. 363, Fu Shing North Road, Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

**Renesas Electronics Singapore Pte. Ltd.**
1 harbourFront Avenue, #06-10, keppel Bay Tower, Singapore 098632
Tel: +65-6213-0200, Fax: +65-6278-8001

**Renesas Electronics Malaysia Sdn.Bhd.**
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

**Renesas Electronics Korea Co., Ltd.**
11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141





Colophon 1.1