

RL78/F13, F14, F15

LIN マスタ通信設定手順（導入編）

要旨

本アプリケーションノートでは、RL78/F13, F14, F15 の LIN/UART モジュール（以下、RLIN3）の LIN 通信機能を、マスタ・モードで使用了した場合の設定手順を説明します。

使用条件によっては、本アプリケーションノートで説明する例と動作が異なる場合があります。実装後は十分な評価を実施してください。また、RLIN3 の機能やクロック、割り込み機能の詳細は必ずご使用する製品のユーザーズマニュアルで確認してください。

対象デバイス

- RL78/F13, F14, F15

目次

1. LIN マスタ通信仕様	3
1.1 LIN マスタ通信使用条件.....	3
1.2 LIN ネットワーク接続例.....	4
1.3 使用する SFR	5
2. LIN マスタ通信概略	7
2.1 LIN マスタ通信概略手順.....	7
2.2 LIN フレームのタイミング (フレーム・セパレート・モード).....	9
2.3 LIN マスタ通信状態と SFR	10
3. LIN マスタ通信ドライバ	11
3.1 関数一覧	11
3.2 ドライバ API 関数 [Public].....	13
3.2.1 LIN マスタ・モジュール初期化.....	13
3.2.2 LIN モジュール初期化解除	14
3.2.3 LIN マスタ・ヘッダ送信.....	15
3.2.4 スリープ遷移前処理	16
3.2.5 LIN マスタ・ウェイクアップ信号送信	17
3.3 ドライバ内部関数 [Private].....	18
3.3.1 LIN0 ポート初期化	18
3.3.2 LIN0 割り込み初期化.....	19
3.3.3 LIN リセット・モード遷移	20
3.3.4 LIN オペレーション・モード遷移	21
3.3.5 LIN ウェイクアップ・モード遷移	22
3.4 割り込み処理	23
3.4.1 LIN マスタ送信完了割り込み処理	23
3.4.2 LIN マスタ受信完了割り込み処理	25
3.4.3 LIN マスタ・ステータス・エラー割り込み処理	26
3.4.4 LIN マスタ・ウェイクアップ割り込み処理.....	27
3.5 ユーザ通知関数 [API].....	28
3.5.1 エラー通知関数 API.....	28
3.5.2 ヘッダ送信完了通知関数 API	29
3.5.3 レスポンス送信完了通知関数 API	30
3.5.4 レスポンス受信完了通知関数 API	31
3.5.5 ウェイクアップ信号送信完了通知関数 API	32
3.5.6 ウェイクアップ信号受信通知関数 API.....	33
4. LIN / UART モジュール (RLIN3) モード遷移.....	34
5. 参考資料.....	35
6. 付録.....	36
改訂記録	37

1. LIN マスタ通信仕様

1.1 LIN マスタ通信使用条件

本アプリケーションノートで説明する LIN マスタ通信の使用条件を以下に示します。

表 1.1 LIN Master Mode Usage Conditions

Item		Description
CPU/peripheral hardware clock frequency (f_{CLK})		32 MHz
LIN clock source (f_{LIN})		f_{CLK}
LTXD0		P13
LRXD0		P14
Baud rate		19200 bps
Mode		LIN master mode
Noise filter		Enabled
LIN interrupts		INTLIN0TRM, INTLIN0RVC, INTLIN0STA, INTLIN0WUP
Checksum type		Classic checksum or Enhanced checksum (User selected)
Error detection	Bit error	Enabled
	Physical bus error	Enabled
	Timeout error	Enabled
	Framing error	Enabled
	Checksum error	Enabled
	Response preparation error	Enabled
Number of bytes in frame's data field		0 to 8 bytes
Response space		0 Tbit
Inter-byte space		1 Tbit

1.2 LIN ネットワーク接続例

LIN マスタ - スレーブ間のネットワーク接続例を以下に示します。

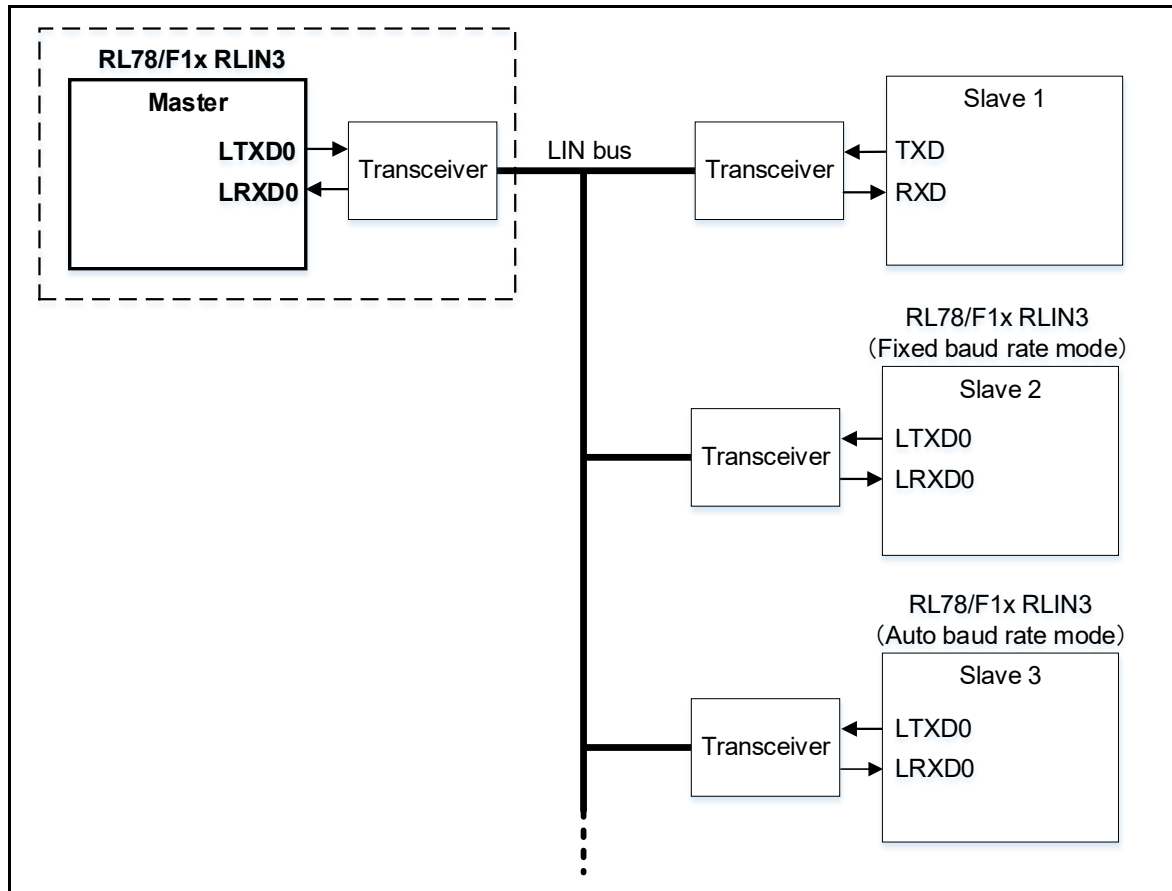


図 1.1 LIN Master - Slaves Network Connection (example)

1.3 使用する SFR

LIN マスタ通信で使用する SFR の設定を以下に示します。

表 1.2 SFRs Used for LIN Communication in Master Mode (1/2)

Register		Setting Value	Conditions
Name	Symbol		
Peripheral enable register 2	PER2	LIN0EN = 0 or 1	0: Stop LIN0 clock supply, 1: LIN0 clock supply
LIN clock select register	LINCKSEL	10H	LIN0MCK=0 (LIN0 communication clock = f_{CLK}) LIN0MCKE=1 (LIN0 communication clock supply)
LIN channel select register	LCHSEL	00H	LSEL0=0 (Enable LIN0 register access)
LIN wake-up baud rate select register 0	LWBR0	01H	LWBR0=1 (Wake-up signal: LIN2.x specification) LPRS[2:0]=000B (LIN0 prescaler clock: 1/1) NSPB[3:0]=0000B (Bit sampling: 16 sampling)
LIN/UART baud rate prescaler 0 register 0	LBRP00	67H	$f_a = 19200\text{bps}$, $f_b = 9600\text{bps}$, $f_c = 2400\text{bps}$
LIN/UART baud rate prescaler 1 register 0	LBRP01	5FH	$f_d = 10417\text{bps}$
LIN self-test control register 0	LSTC0	00H	LSTM=0 (LIN self-test mode is disabled)
LIN/UART mode register 0	LMD0	10H	LMD[1:0]=00B (LIN master mode) LCKS[1:0]=00B (LIN system clock: f_a (19200bps)) LIOS=1 (Enable LIN0 transmission/reception/status interrupt) LRDNFS=0 (LRXD0 noise filter enabled)
LIN/UART break field configuration register 0	LBFC0	00H	BLT[3:0]=0000B (Transmission break-low width: 13 Tbit) BDT[1:0]=00B (Transmission break delimiter width: 1 Tbit)
LIN/UART space configuration register 0	LSC0	10H	IBSH[2:0]=000B (Inter-byte space (header) / Response space: 0 Tbit) IBS[1:0]=01B (Inter-byte space: 1 Tbit)
LIN wake-up configuration register 0	LWUP0	40H	WUTL[3:0]=0100B (Wake-up transmission low width: 5 Tbit)
LIN interrupt enable register 0	LIE0	0FH	FTCIE=1 (Frame / Wake-up transmission completion interrupt: Enabled) FRCIE=1 (Frame / Wake-up reception completion interrupt: Enabled) ERRIE=1 (Error detection interrupt: Enabled) SHIE=1 (Header transmission completion interrupt: Enabled)
LIN/UART error detection enable register 0	LEDE0	8FH	BERE=1 (Bit error detection: Enabled) PBERE=1 (Physical bus error detection: Enabled) FTERE=1 (Response timeout error detection: Enabled) FERE=1 (Framing error detection: Enabled) LTES=1 (Timeout selection bit: Response timeout)
LIN/UART control register 0	LCUC0	00H, 01H, 03H	LCUC0=00H (LIN reset mode transition) LCUC0=01H (LIN wake-up mode transition) LCUC0=03H (LIN operation mode transition)
LIN/UART transmission control register 0	LTRC0	01H, 02H	LTRC0=01H (Start Header transmission / Wake-up transmission) LTRC0=02H (Start response transmission/reception)
LIN/UART mode status register 0	LMST0	Read (00H, 01H, 03H)	LMST0=00H (LIN reset mode status) LMST0=01H (LIN wake-up mode status) LMST0=03H (LIN operation mode status)
LIN/UART status register 0	LST0	Read 00H (Status clear)	Read status flag Bit 0: FTC (1: Frame / Wake-up transmission completion) Bit 1: FRC (1: Response reception completion) Bit 3: ERR (1: Status error detection) Bit 6: D1RC (1: Data[1] reception completion) Bit 7: HTRC (1: Header transmission completion)
LIN/UART error status register 0	LEST0	Read 00H (Error clear)	Read error status flag Bit 0: BER (1: Bit error detection) Bit 1: PBER (1: Physical bus error detection) Bit 2: FTER (1: Response timeout error detection) Bit 3: FER (1: Framing error detection) Bit 5: CSER (1: Checksum error detection) Bit 7: RPER (1: Response preparation error detection)
LIN/UART data field configuration register 0	LDFC0	xxH	RFDL[3:0]: Response data length (0 to 8) RFT: Response direction (0: Reception, 1: Transmission) CSM: Checksum type (0: Classic checksum, 1: Enhanced checksum) FSM=1 (Select frame separate mode) LSS=0 (Next transmission/reception data group is the last one)

表 1.2 SFRs Used for LIN Communication in Master Mode (2/2)

Register		Setting Value	Conditions
Name	Symbol		
LIN/UART ID buffer register 0	LIDB0	xxH	Set transmission ID and parity bit
LIN checksum buffer register 0	LCBR0	xxH	Read reception checksum ^{Note}
LIN/UART data buffer [1-8] registers	LDB01 - LDB08	xxH	Set transmission data[n], or read reception data[n]

Note: LCBRO register is not used in this application note. Check for checksum error detection.

表 1.3 SFRs Used for LIN Communication in Master Mode (Port related registers)

Register		Setting Value	Conditions
Name	Symbol		
Input switch control register	ISC	ISC2=1	LRXD0 input pin interrupt enabled
Peripheral I/O redirection register 4	PIOR4	PIOR44=0	LRXD0: P14, LTXD0: P13 selected
Port register 1	P1	P13=1	Port P1.3 output latch = High
Port mode register 1	PM1	PM13=0, PM14=1	Port P1.3: output direction, Port P1.4: input direction
Port output mode register 1	POM1	POM13=0	Port P1.3: CMOS output
Port input mode register 1	PIM1	PIM14=0	Port P1.4: CMOS input
Port input threshold control register 1	PITHL1	PITHL14=0	Port P1.4: Schmitt1 input type selected
Pull-up resistor option register 1	PU1	PU14=0	Port P1.4: Internal pull-up disabled

表 1.4 SFRs Used for LIN Communication in Master Mode (Interrupt related registers)

Register		Setting Value	Conditions
Name	Symbol		
External interrupt rising edge enable register 1	EGP1	EGP11=0	LRXD0: Falling edge detection enabled
External interrupt falling edge enable register 1	EGN1	EGN11=1	
Interrupt request flag register 0H	IF0H	LIN0TRMIF=0	Clear the LIN0 transmission interrupt request flag
Interrupt request flag register 1L	IF1L	LIN0RVCIF=0	Clear the LIN0 reception interrupt request flag
		LIN0STAIF=0	Clear the LIN0 error status interrupt request flag
Interrupt request flag register 2L	IF2L	LIN0WUPIF=0	Clear the LRXD0 falling edge detection interrupt request flag
Interrupt mask flag register 0H	MK0H	LIN0TRMMK=0,1	LIN0 transmission interrupt mask flag (0: Interrupt enabled, 1: Interrupt mask)
Interrupt mask flag register 1L	MK1L	LIN0RVCMK=0,1	LIN0 reception interrupt mask flag (0: Interrupt enabled, 1: Interrupt mask)
		LIN0STAMK=0,1	LIN0 status interrupt mask flag (0: Interrupt enabled, 1: Interrupt mask)
Interrupt mask flag register 2L	MK2L	LIN0WUPMK=0,1	LRXD0 falling edge detection interrupt mask flag (0: Interrupt enabled, 1: Interrupt mask)
Priority specification flag registers 10H/00H	PR10H, PR00H	LIN0TRMPR=01B	LIN0 transmission interrupt priority level flag 0, 1 (01B: Level-1)
Priority specification flag registers 11L/01L	PR11L, PR01L	LIN0RVCPR=01B	LIN0 reception interrupt priority level flag 0, 1 (01B: Level-1)
		LIN0STAPR=01B	LIN0 status interrupt priority level flag 0, 1 (01B: Level-1)
Priority specification flag registers 12L/02L	PR12L, PR02L	LIN0WUPPR=01B	LRXD0 falling edge detection interrupt priority level flag 0, 1 (01B: Level-1)

2. LIN マスタ通信概略

2.1 LIN マスタ通信概略手順

LIN マスタ通信処理例の概略手順を以下に示します。

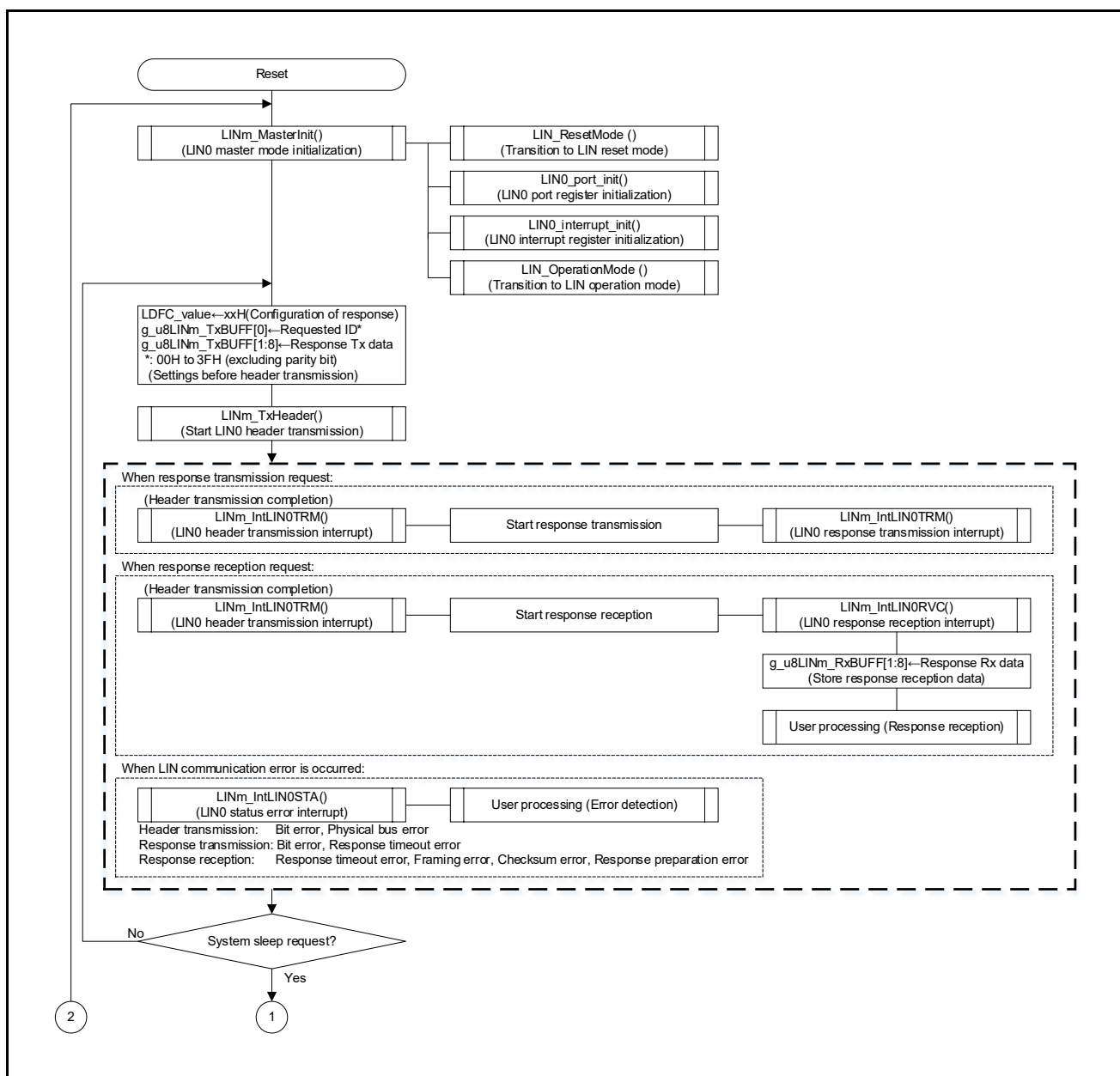


図 2.1 LIN Communication Process in Master Mode (1/2)

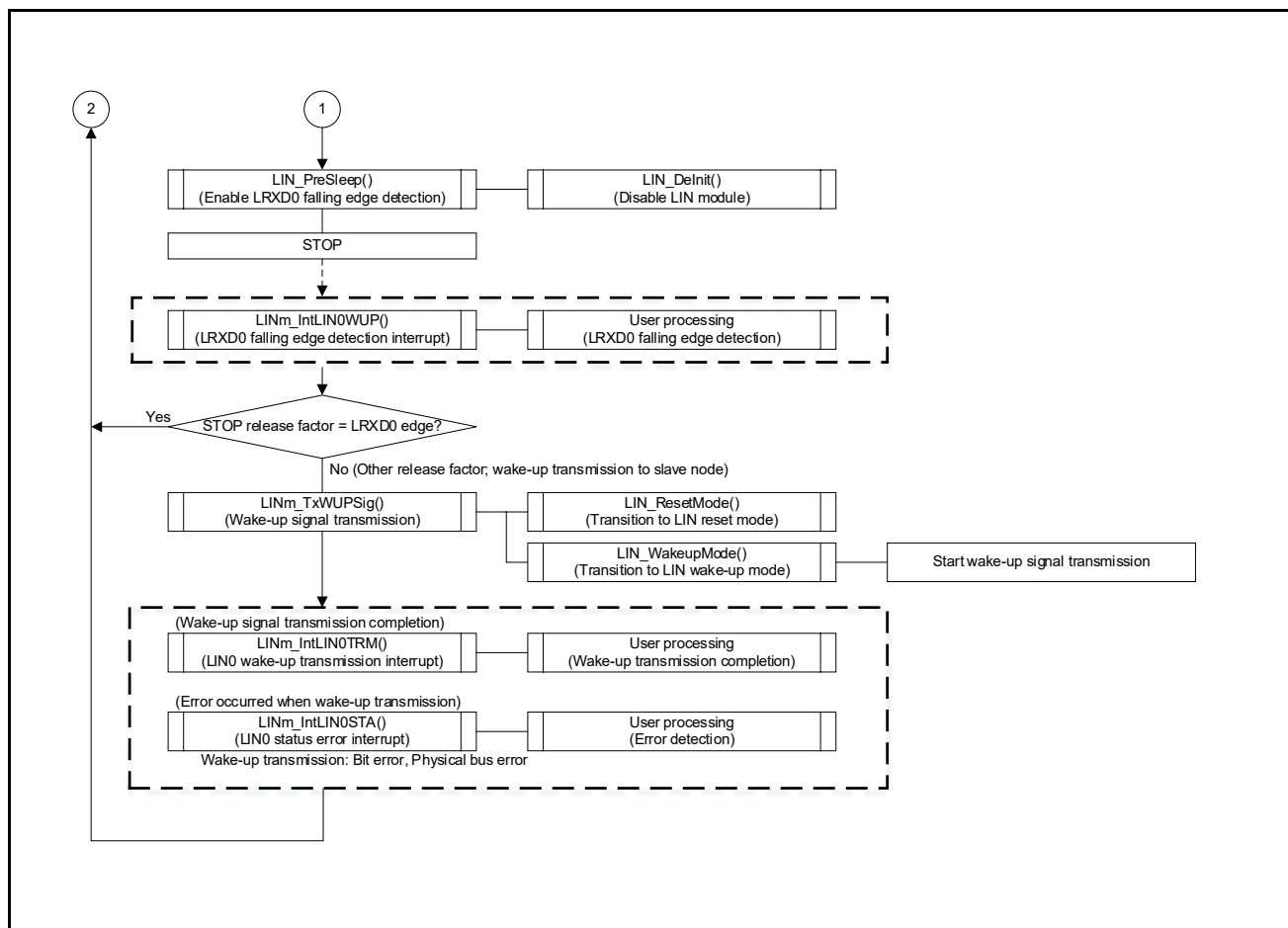
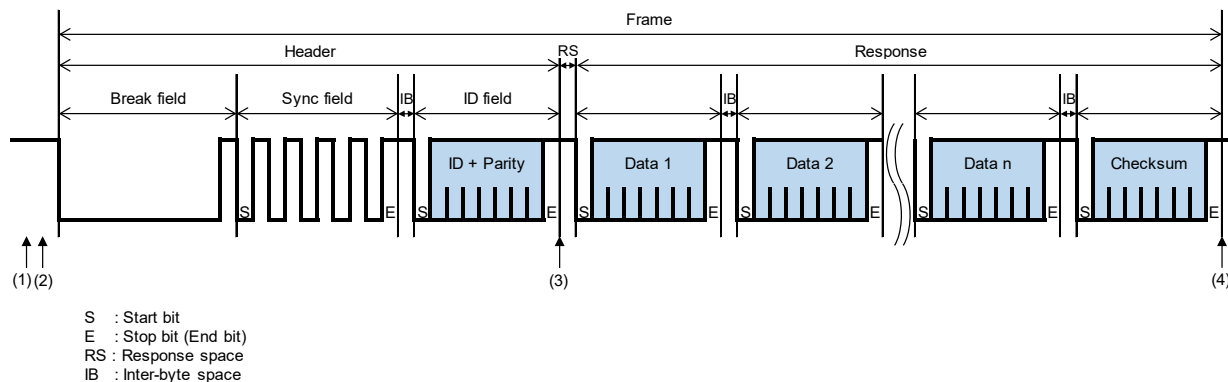


図 2.1 LIN Communication Process in Master Mode (2/2)

2.2 LIN フレームのタイミング (フレーム・セパレート・モード)

本アプリケーションノートで示す LIN マスタ通信では、ヘッダとレスポンスを別々の送信要求により開始するフレーム・セパレート・モードを使用しています。LIN マスタにおけるヘッダ送信と、レスポンス送受信のタイミング図を以下に示します。



- (1) Initialize the LIN master module.
- (2) Enable the device to transmit a header (LTRC0←01H).
 - Configure the response direction as reception (LDFC0.RFT←0) or transmission (LDFC0.RFT←1).
 - Specify the response length with the RFDL[3:0] bits of the LDFC0 register.
 - Start transmitting a header (LTRC0←01H).
- (3) The INTLIN0TRM interrupt occurs. Make sure the device has transmitted a header (LST0.HTRC = 1) and no error has been detected (LST0.ERR = 0). Subsequently, the following processing is performed for receiving or transmitting the response.
 - When transmitting a response, write data to the LDB0x registers (x = 1 to 8).
 - Start receiving or transmitting a response (LTRC0←02H).
- (4) INTLIN0RVC or INTLIN0TRM interrupt occurs. Make sure no error has been detected and read the received response from the LDB0x registers (x = 1 to 8).

図 2.2 LIN Master Header Transmission and Response Transmission / Reception Timing

2.3 LIN マスタ通信状態と SFR

LIN/UART モジュール（RLIN3）は、LINn 送信割り込み／LINn 受信完了割り込み／LINn 受信ステータス割り込みを生成します。また、各割り込みは複数の割り込み要因を持ちます。詳しくは、対象製品のユーザズ・マニュアルを参照ください。本アプリケーションノートでは、RLIN3 モジュールのステータスレジスタを読み出して割り込み要因を判定しています。通信状態と SFR の関係を、送信／受信それぞれに関して以下に示します。

表 2.1 LIN Communication State and SFRs (LINn successful transmission interrupt)

LIN Communication State Transmission	SFRs				
	LST0			LDLC0	LMST0
	bit3(ERR)	bit7(HTRC)	bit0(FTC)	RFT	
Communication error * Processed in the LIN0 status interrupt	1	-	-	-	-
Header transmission completion	0	1	-	0	-
Response not requested				1	
Response requested		0	1	-	03H
Response transmission completion				-	#03H
Wake-up transmission completion				-	-
Illegal interrupt occurred			0	-	-

:- Not judged by software.

Remarks: For details, see section 3.4.1.

表 2.2 LIN Communication State and SFRs in Master Mode (LINn successful reception interrupt)

LIN Communication State Reception	SFRs	
	LST0	
	bit3(ERR)	bit1(FRC)
Communication error * Processed in the LIN0 status interrupt	1	-
Response reception completion	0	1
Illegal interrupt occurred		0

:- Not judged by software.

Remarks: For details, see section 3.4.2.

3. LIN マスタ通信ドライバ

本アプリケーションノートで示す LIN マスタ通信のドライバ・プログラム例について説明します。

3.1 関数一覧

本ドライバの関数一覧と使用する変数を以下に示します。

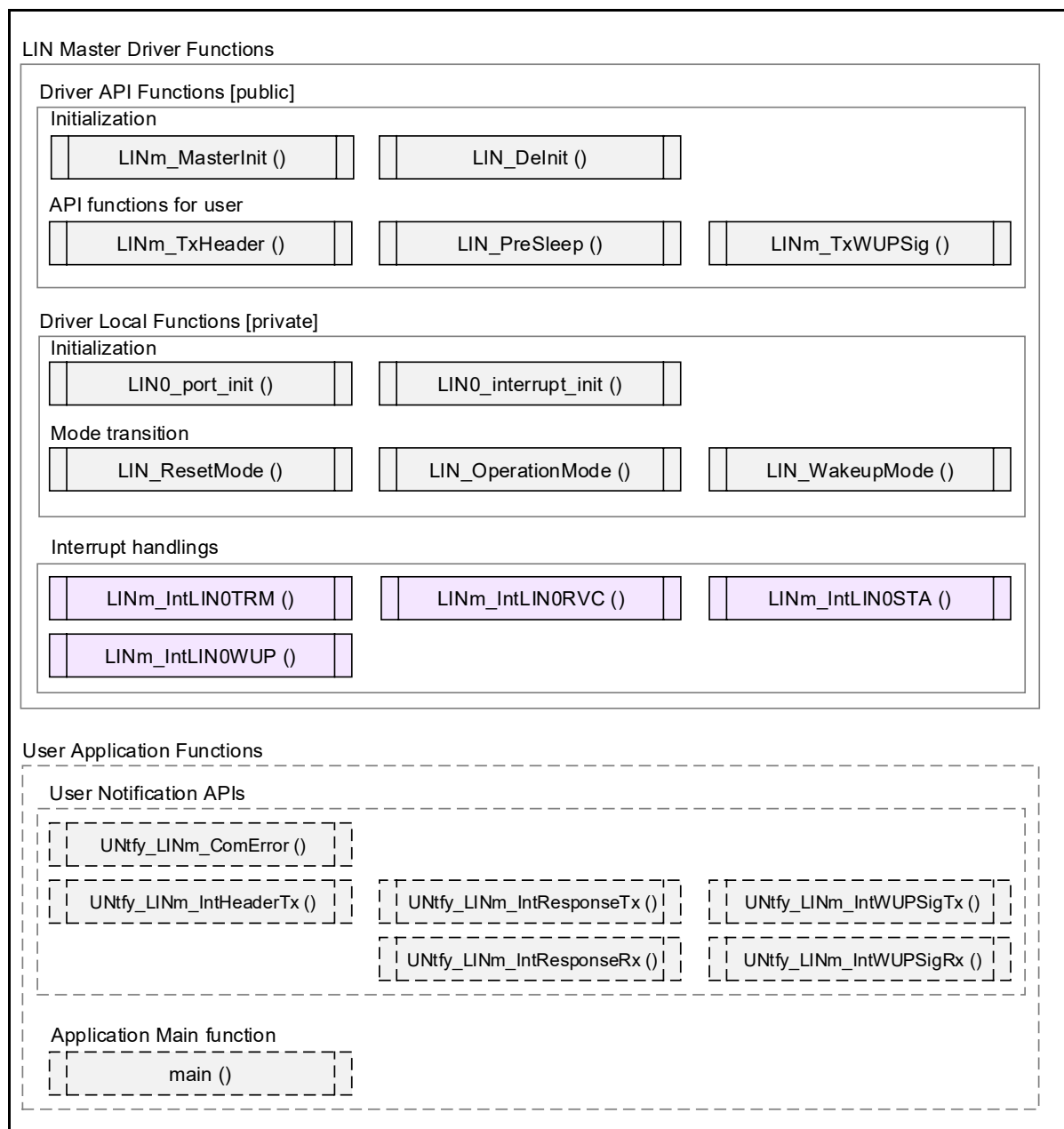


図 3.1 Functions of LIN Communication in Master Mode

表 3.1 Function List for LIN Communication in Master Mode

Function Name	Description	Public / Private
void LINm_MasterInit (void)	This function initializes LIN (master mode), port and interrupt functions.	public
void LIN_DelInit (void)	This function disables LIN (master) module.	public
void LINm_TxHeader (unsigned char u8a_LDFC_value)	This function is the transmission process of LIN header.	public
void LIN_PreSleep (void)	This function is process before transition to sleep mode.	public
void LINm_TxWUPSig (void)	This function transmits the LIN wake-up signal.	public
void LIN0_port_init (void)	Initialization for LIN related port function	private
void LIN0_interrupt_init (void)	Initialization for LIN related interrupt function	private
void LIN_ResetMode (void)	This function transitions to LIN reset mode.	private
void LIN_OperationMode (void)	This function transitions to LIN operation mode.	private
void LIN_WakeupMode (void)	This function transitions to LIN wake-up mode.	private
void LINm_IntLIN0TRM (void)	This function is the LIN0 transmission completion interrupt processing. (Header transmission / Response transmission / Wake-up transmission)	public
void LINm_IntLIN0RVC (void)	This function is the LIN0 response reception completion interrupt processing. (Response reception / Wake-up reception) Note	public
void LINm_IntLIN0STA (void)	This function is the interrupt processing when LIN0 status error is occurred. - Header transmission: Bit error / Physical bus error detection - Response transmission: Bit error / Response timeout error detection - Response reception: Response timeout error / Framing error / Checksum error detection - Wake-up transmission: Bit error / Physical bus error detection	public
void LINm_IntLIN0WUP (void)	This function is the LRXD0 falling edge detection interrupt processing.	public

Note: In this application note, wake-up detection is detected in LRXD0 falling edge. Wake-up reception interrupt is not used.

表 3.2 Variables Used for LIN Communication in Master Mode

Variable Name	Description
unsigned char g_u8LINm_TxBUFF[9]	g_u8LINm_TxBUFF[0]: Set requested ID data g_u8LINm_TxBUFF[1 to 8]: Set response transmission data
unsigned char g_u8LINm_RxBUFF[9]	g_u8LINm_RxBUFF[0]: Not used g_u8LINm_RxBUSS[1 to 8]: Store response reception data

表 3.3 User Notification Function APIs for LIN Master Driver

Function API Name	Description
void UNTfy_LINm_ComError (unsigned short u16a_errorinfo)	User's processing for error detections. bit[7:0]: Store the LEST0 register value (When INTLIN0STA occurred) bit[15:8]: When INTLIN0STA occurred: 00H When INTLIN0RVC occurred: 80H (illegal interrupt) When INTLIN0TRM occurred: 40H (illegal interrupt)
void UNTfy_LINm_IntHeaderTx (void)	User's processing for header transmission completion.
void UNTfy_LINm_IntResponseTx (void)	User's processing for response transmission completion.
void UNTfy_LINm_IntResponseRx (void)	User's processing for response reception completion.
void UNTfy_LINm_IntWUPSigTx (void)	User's processing for wake-up signal transmission completion.
void UNTfy_LINm_IntWUPSigRx (void)	User's processing for wake-up signal reception completion.

3.2 ドライバ API 関数 [Public]

本アプリケーションノートで示す LIN マスタ通信ドライバ API 関数について説明します。

3.2.1 LIN マスタ・モジュール初期化

表 3.4 LIN Master Initialization Function (LINm_MasterInit)

Syntax	void LINm_MasterInit (void)	
Parameters	In	None
	Out	None
Local variables	None	
Return value	None	
Call functions	LIN_ResetMode (): Transition to LIN reset mode. LIN0_port_init (): LIN0 port function initialization. LIN0_interrupt_init (): LIN0 interrupt function initialization. LIN_OperationMode (): Transition to LIN operation mode.	
Description	This function initializes LIN (master mode), port and interrupt functions.	
Note	Call from upper layer with interrupt disabled.	

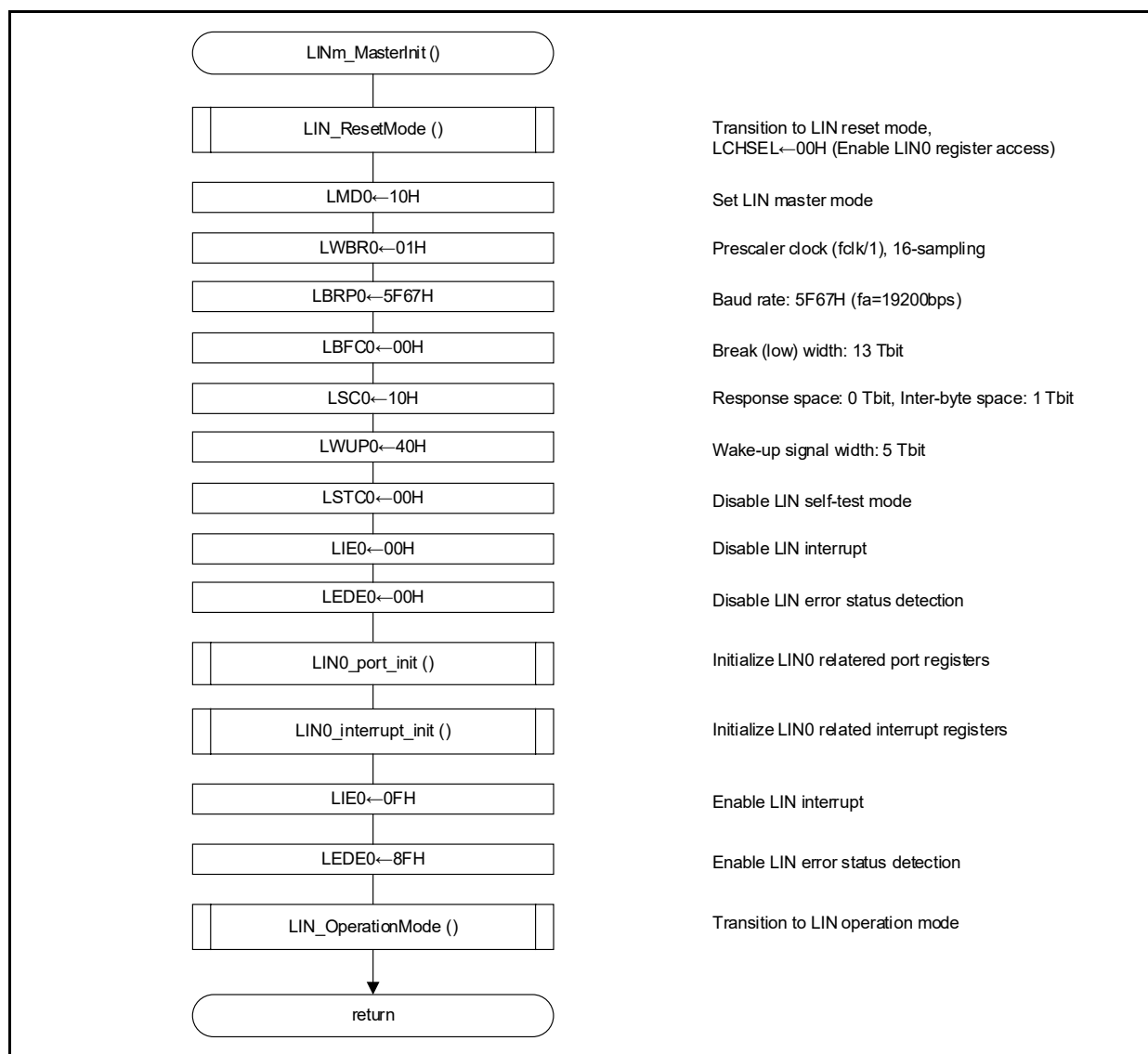


図 3.2 LIN Master Initialization Procedure (LINm_MasterInit)

3.2.2 LIN モジュール初期化解除

表 3.5 LIN De-Initialization Function (LIN_DeInit)

Syntax		void LIN_DeInit (void)	
Parameters	In	None	
	Out	None	
Local variables		None	
Return value		None	
Call functions		None	
Description		This function disables LIN module.	
Note		Call this function when GCSC and GINT bits of IAWCTL register are 0.	

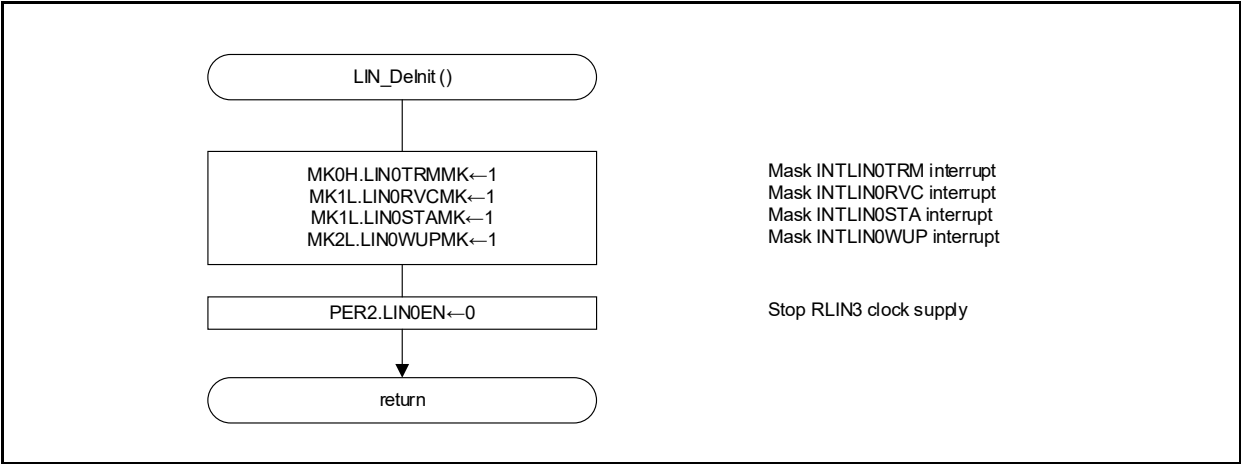


図 3.3 LIN De-Initialization Procedure (LIN_DeInit)

3.2.3 LIN マスタ・ヘッダ送信

表 3.6 LIN Master Header Transmission Function (LINm_TxHeader)

Syntax		void LINm_TxHeader (unsigned char u8a_LDFC_value)	
Parameters	In	unsigned char	u8a_LDFC_value LDFC0 setting value
		- bit3-0 (RFDL[3:0]):	Response field length [0 to 8]
		- bit4 (RFT):	Response direction [0: Reception, 1: Transmission]
		- bit5 (CSM):	Checksum type [0: classic-checksum, 1: enhanced-checksum]
		- bit6 (FSM):	1: Frame separate mode
		- bit7 (LSS):	0: Next transmission/reception data group is the last one
		unsigned char	g_u8LINm_TxBUFF[0] Transmission ID setting
		unsigned char	g_u8LINm_TxBUFF[1:8] Response transmission data setting
	Out	None	
Local variables		const unsigned char	u8TBL_ParityCalc[0x40u] Parity bit calculation of requested ID (Refer to Chapter. 6)
Return value		None	
Call functions		None	
Description		This function is the transmission process of LIN header.	
Note		Call this function when GINT bit of IAWCTL register is 0.	

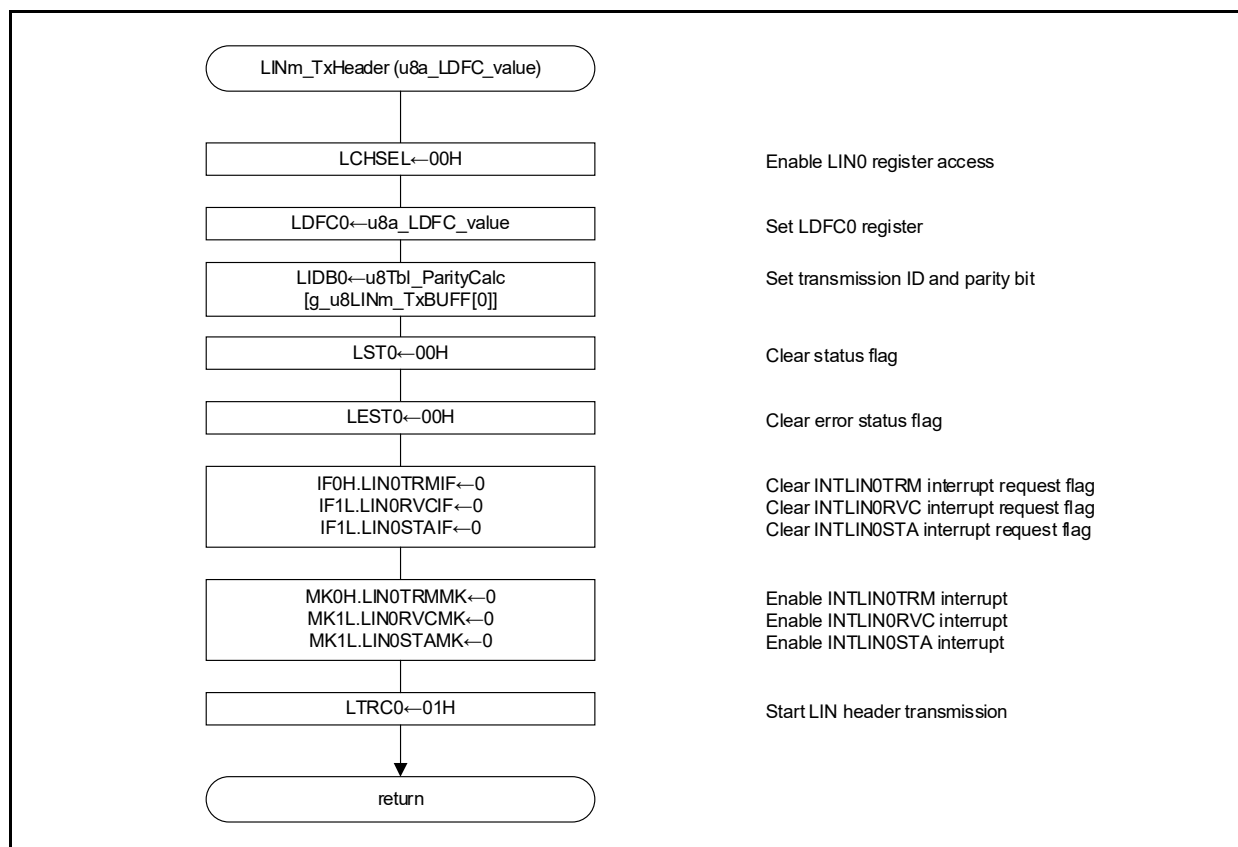


図 3.4 LIN Master Header Transmission Procedure (LINm_TxHeader)

3.2.4 スリープ遷移前処理

表 3.7 Before Sleep Mode Transition Processing Function (LIN_PreSleep)

Syntax	void LIN_PreSleep (void)	
Parameters	In	None
	Out	None
Local variables	None	
Return value	None	
Call functions	LIN_DeInit (): Disable LIN module function	
Description	This function is process before transition to sleep mode.	
Note	Call from upper layer with interrupt disabled.	
	Call this function when GINT bit of IAWCTL register is 0.	

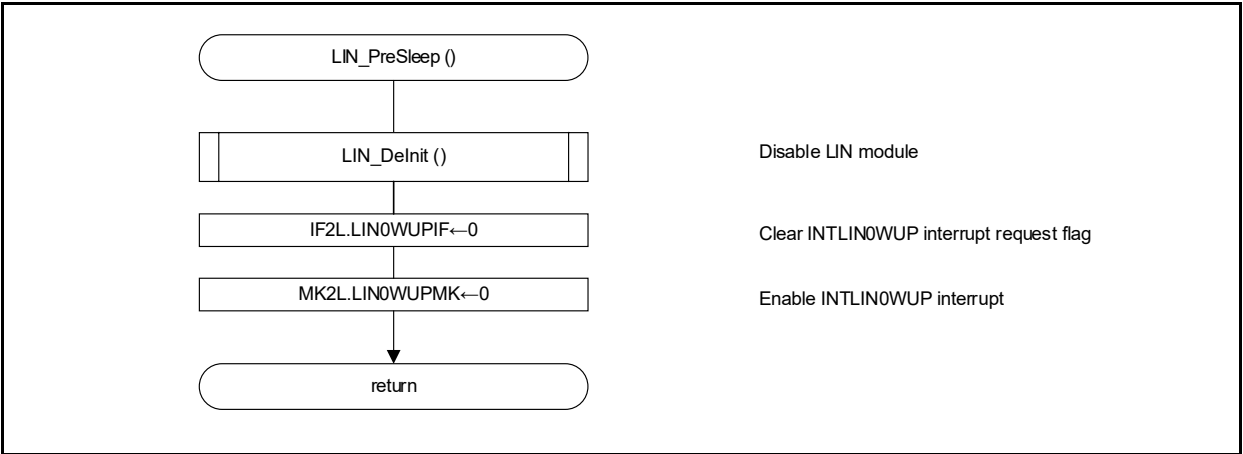


図 3.5 Before Sleep Mode Transition Processing Procedure (LIN_PreSleep)

3.2.5 LIN マスタ・ウェイクアップ信号送信

表 3.8 LIN Master Wake-up Signal Transmission Function (LINm_TxWUPSig)

Syntax	void LINm_TxWUPSig (void)	
Parameters	In	None
	Out	None
Local variables	None	
Return value	None	
Call functions	LIN_ResetMode ():	LIN reset mode transition function
	LIN_WakeupMode ():	LIN wake-up mode transition function
Description	This function transmits the LIN wake-up signal.	
Note	Call this function when GINT bit of IAWCTL register is 0.	

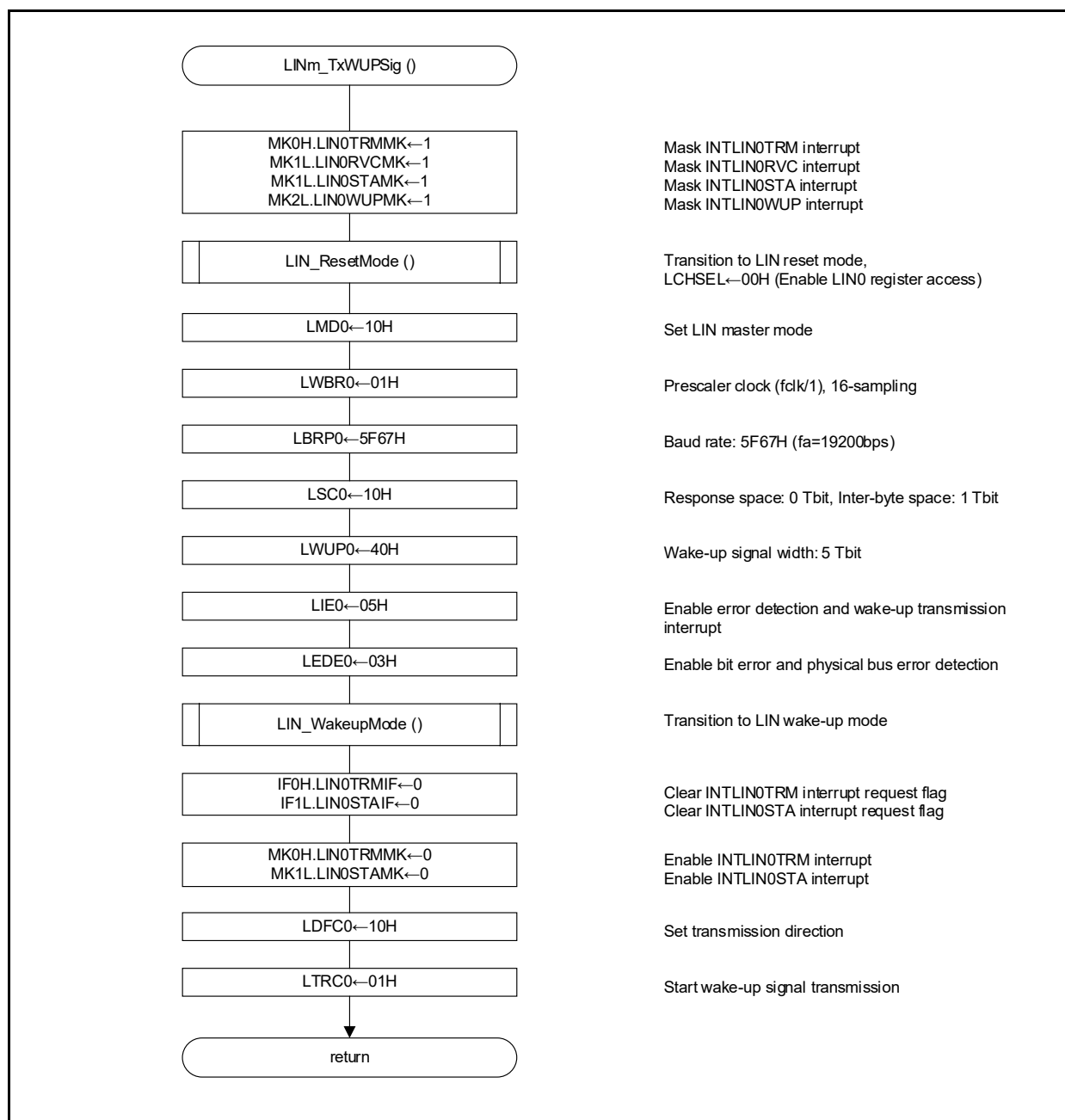


図 3.6 LIN Master Wake-up Signal Transmission Procedure (LINm_TxWUPSig)

3.3 ドライバ内部関数 [Private]

本アプリケーションノートで示す LIN マスタ通信ドライバ内部関数について説明します。

3.3.1 LIN0 ポート初期化

表 3.9 LIN0 Port Initialization Function (LIN0_port_init)

Syntax	void LIN0_port_init (void)	
Parameters	In	None
	Out	None
Local variables	None	
Return value	None	
Call functions	None	
Description	This function initializes LIN0 related ports.	
Note	Call this function when GPORT bit of IAWCTL register is 0.	

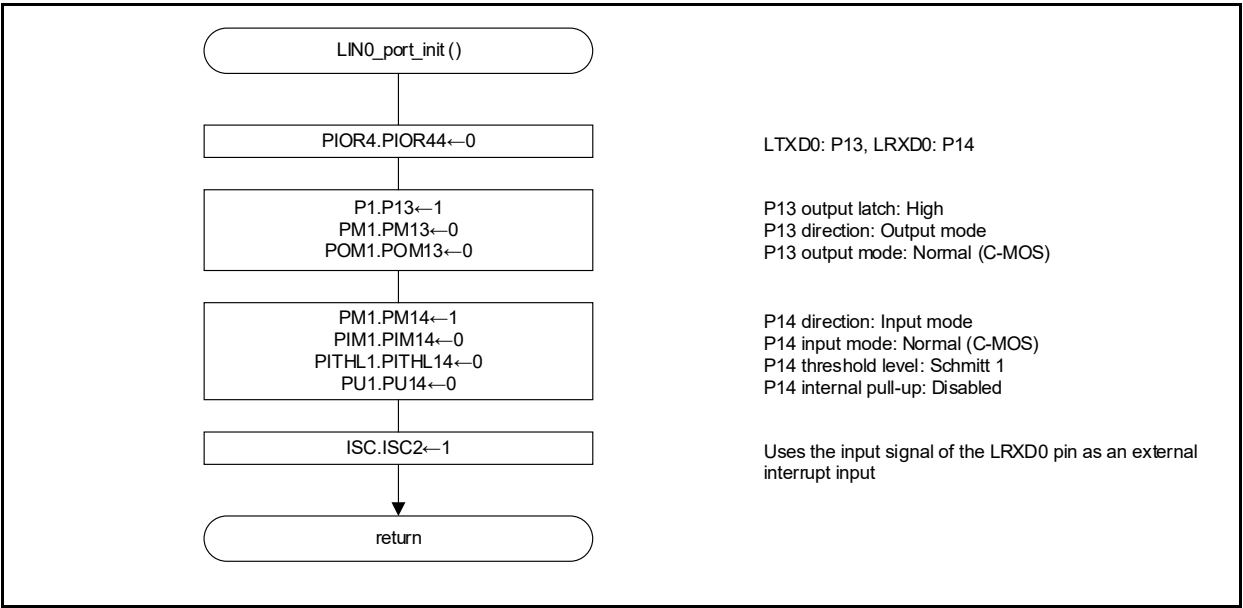


図 3.7 LIN0 port Initialization Procedure (LIN0_port_init)

3.3.2 LIN0 割り込み初期化

表 3.10 LIN0 Interrupt Initialization Function (LIN0_interrupt_init)

Syntax	void LIN0_interrupt_init (void)	
Parameters	In	None
	Out	None
Local variables	None	
Return value	None	
Call functions	None	
Description	This function initializes LIN0 related interrupts.	
Note	Call this function when GINT bit of IAWCTL register is 0.	

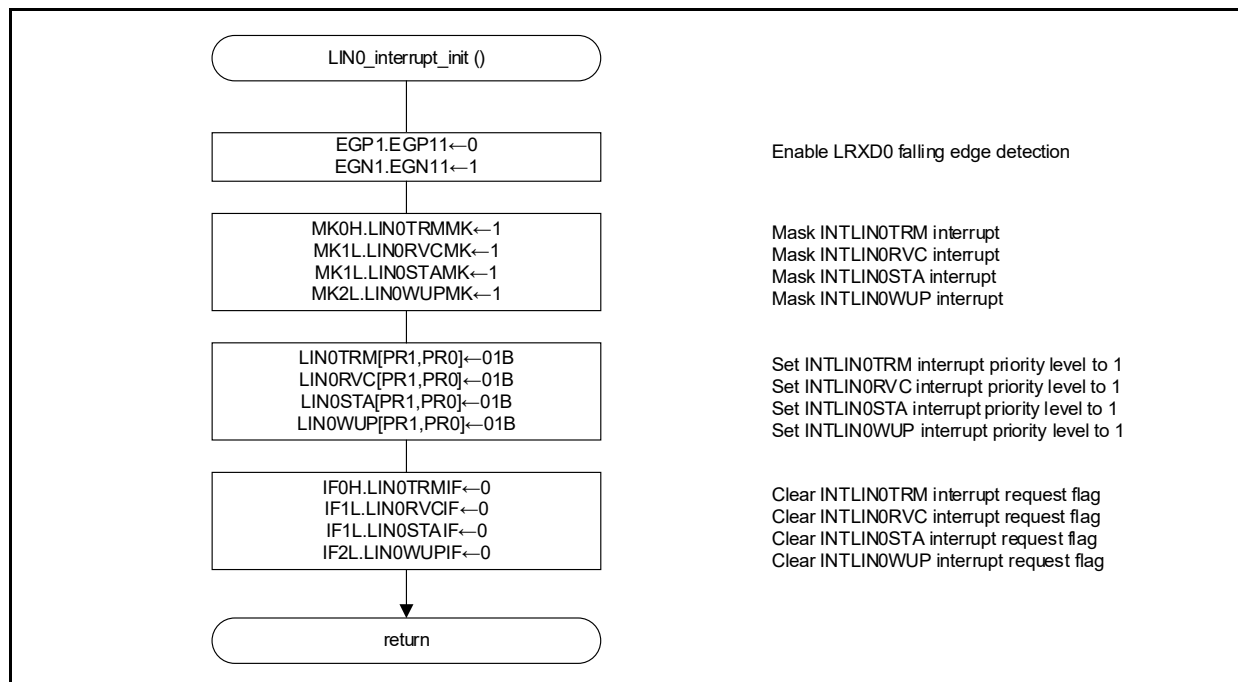


図 3.8 LIN0 Interrupt Initialization Procedure (LIN0_interrupt_init)

3.3.3 LIN リセット・モード遷移

表 3.11 Transition to LIN Reset Mode Function (LIN_ResetMode)

Syntax	void LIN_ResetMode (void)	
Parameters	In	None
	Out	None
Local variables	None	
Return value	None	
Call functions	None	
Description	This function transitions to LIN reset mode.	
Note	Call this function when GCSC bit of IAWCTL register is 0.	

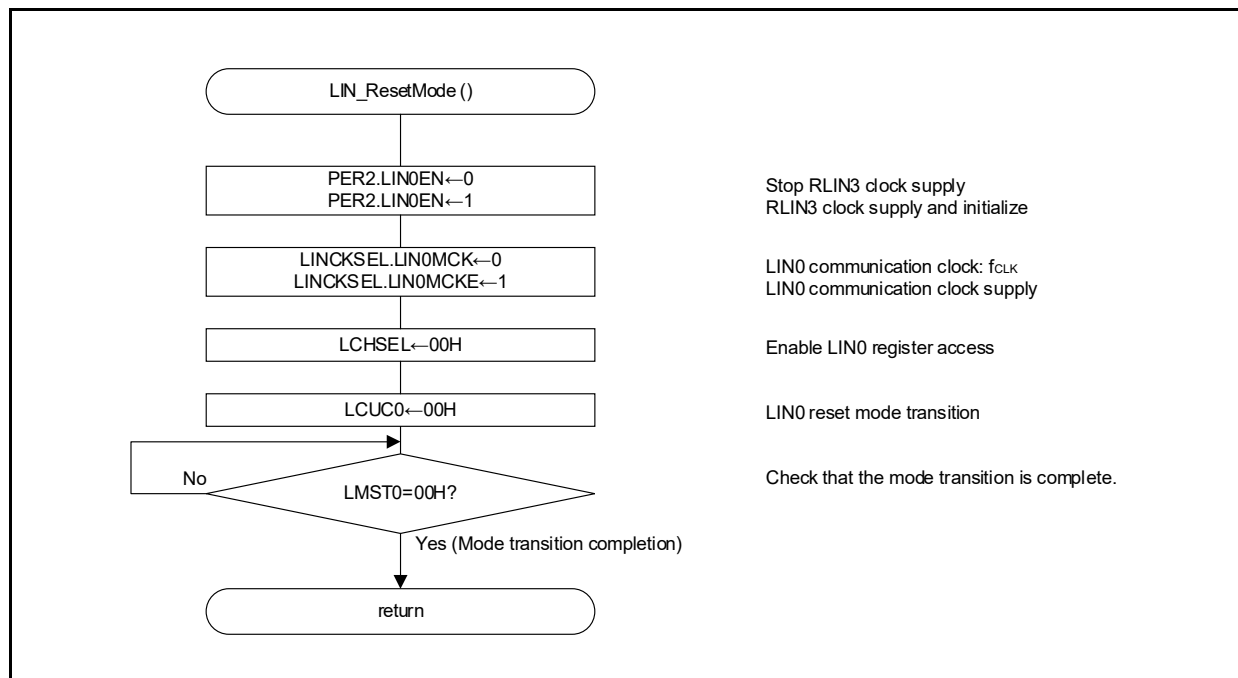


図 3.9 Transition to LIN Reset Mode Procedure (LIN_ResetMode)

3.3.4 LIN オペレーション・モード遷移

表 3.12 Transition to LIN Operation Mode Function (LIN_OperationMode)

Syntax	void LIN_OperationMode (void)	
Parameters	In	None
	Out	None
Local variables	None	
Return value	None	
Call functions	None	
Description	This function transitions to LIN operation mode.	

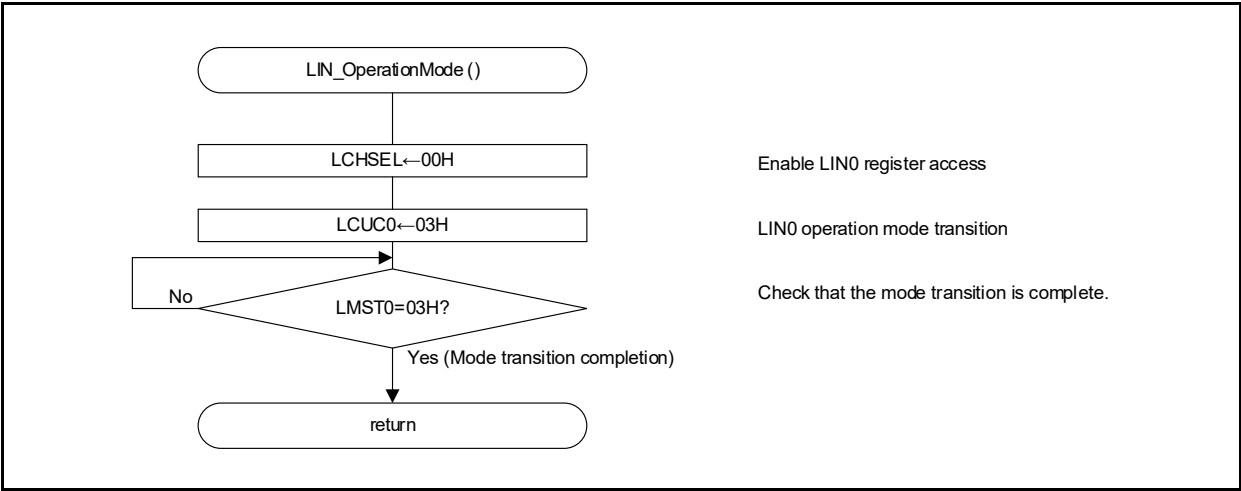


図 3.10 Transition to LIN Operation Mode Procedure (LIN_OperationMode)

3.3.5 LIN ウェイクアップ・モード遷移

表 3.13 Transition to LIN Wake-up Mode Function (LIN_WakeupMode)

Syntax	void LIN_WakeupMode (void)	
Parameters	In	None
	Out	None
Local variables	None	
Return value	None	
Call functions	None	
Description	This function transitions to LIN wake-up mode.	

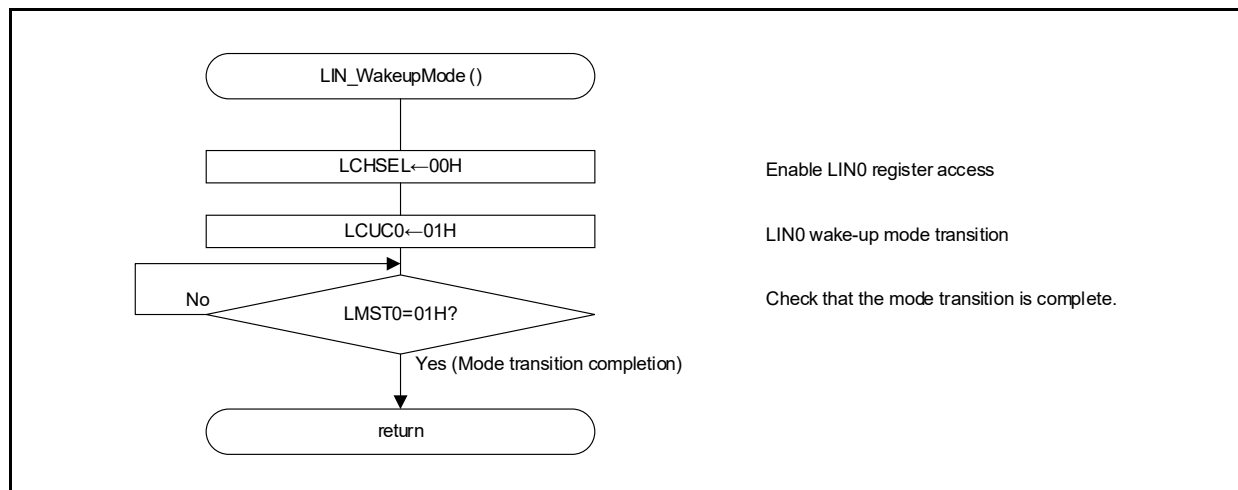


図 3.11 Transition to LIN Wake-up Mode Procedure (LIN_WakeupMode)

3.4 割り込み処理

本アプリケーションノートで示す LIN マスタ通信ドライバ割り込み処理関数について説明します。

3.4.1 LIN マスタ送信完了割り込み処理

表 3.14 LIN Master Transmission Completion Interrupt Processing Function (LINm_IntLIN0TRM)

Syntax	void LINm_IntLIN0TRM (void)	
Parameters	In	unsigned char g_u8LINm_TxBUFF[1:8]: Response transmission data setting* *: Set before call LINm_TxHeader () function
	Out	None
Local variables	unsigned char u8a_RD_LST0:	Store LST0 register
	- bit0 (FTC):	Response / Wake-up transmission completion
	- bit3 (ERR):	LIN communication error detection
	- bit7 (HTRC):	Header transmission completion
	unsigned short u16a_errorinfo:	4000H (Illegal interrupt has occurred)
Return value	None	
Call functions	UNTfy_LINm_IntHeaderTx ():	Header transmission completion processing
	UNTfy_LINm_IntResponseTx ():	Response transmission completion processing
	UNTfy_LINm_IntWUPSigTx ():	Wake-up transmission completion processing
	UNTfy_LINm_ComError (u16a_errorinfo):	Illegal interrupt processing
Description	This function is the LIN0 transmission completion interrupt processing. - LIN0 header transmission completion - LIN0 response transmission completion - LIN0 wake-up signal transmission completion	

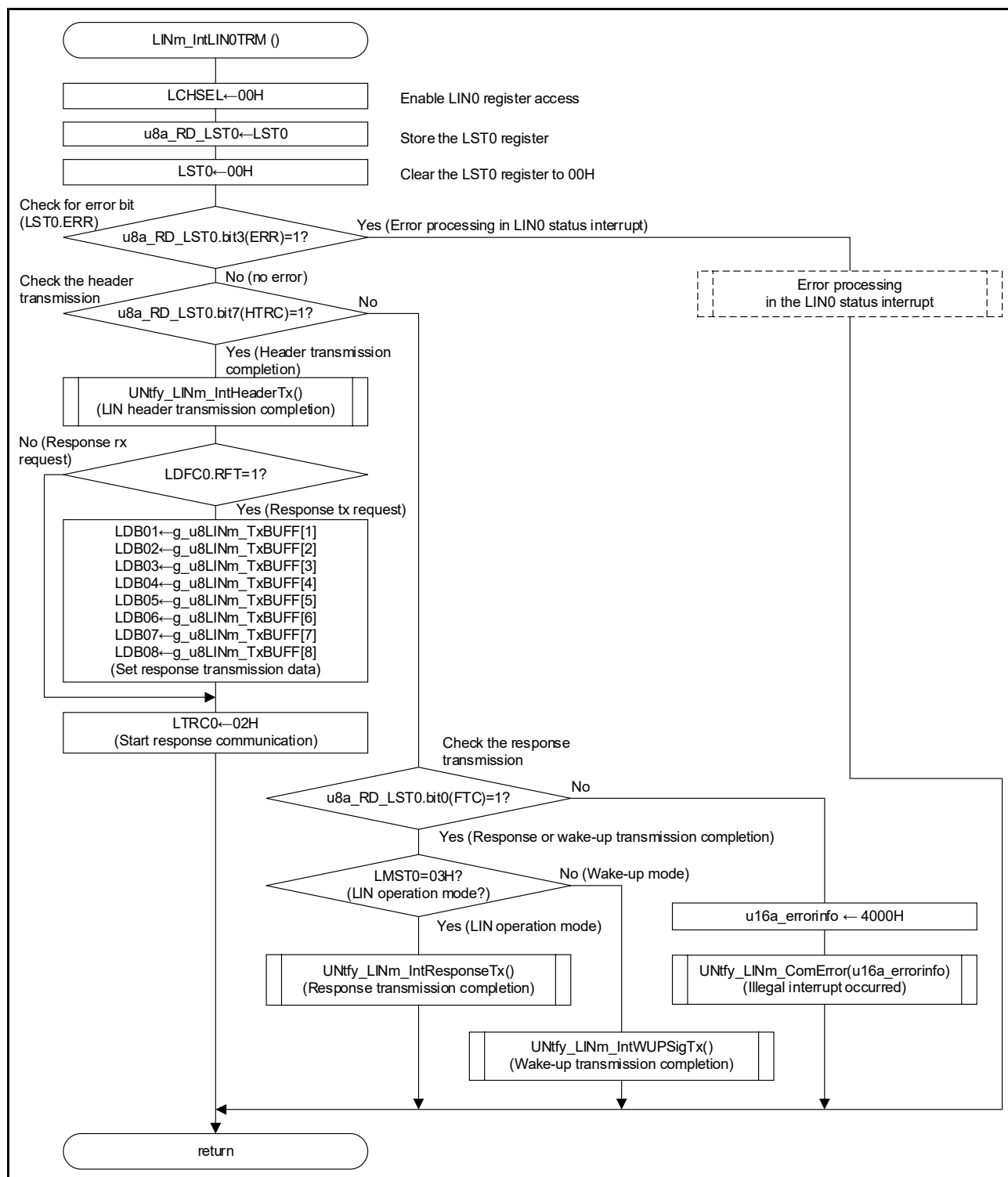


図 3.12 LIN Master Transmission Completion Interrupt Processing Procedure (LINm_IntLIN0TRM)

3.4.2 LIN マスタ受信完了割り込み処理

表 3.15 LIN Master Reception Completion Interrupt Processing Function (LINm_IntLIN0RVC)

Syntax		void LINm_IntLIN0RVC (void)		
Parameters	In	None		
	Out	unsigned char	g_u8LINm_RxBUFF[1:8]:	Response reception data
Local variables		unsigned char	u8a_RD_LST0:	Store LST0 register
		- bit1 (FRC):	Response reception completion	
		- bit3 (ERR):	LIN communication error detection	
		unsigned short	u16a_errorinfo:	8000H (Illegal interrupt has occurred)
Return value		None		
Call functions		UNTfy_LINm_IntResponseRx ():		Response reception completion processing
		UNTfy_LINm_ComError (u16a_errorinfo):		Illegal interrupt processing
Description		This function is the LIN0 response reception completion interrupt processing.		

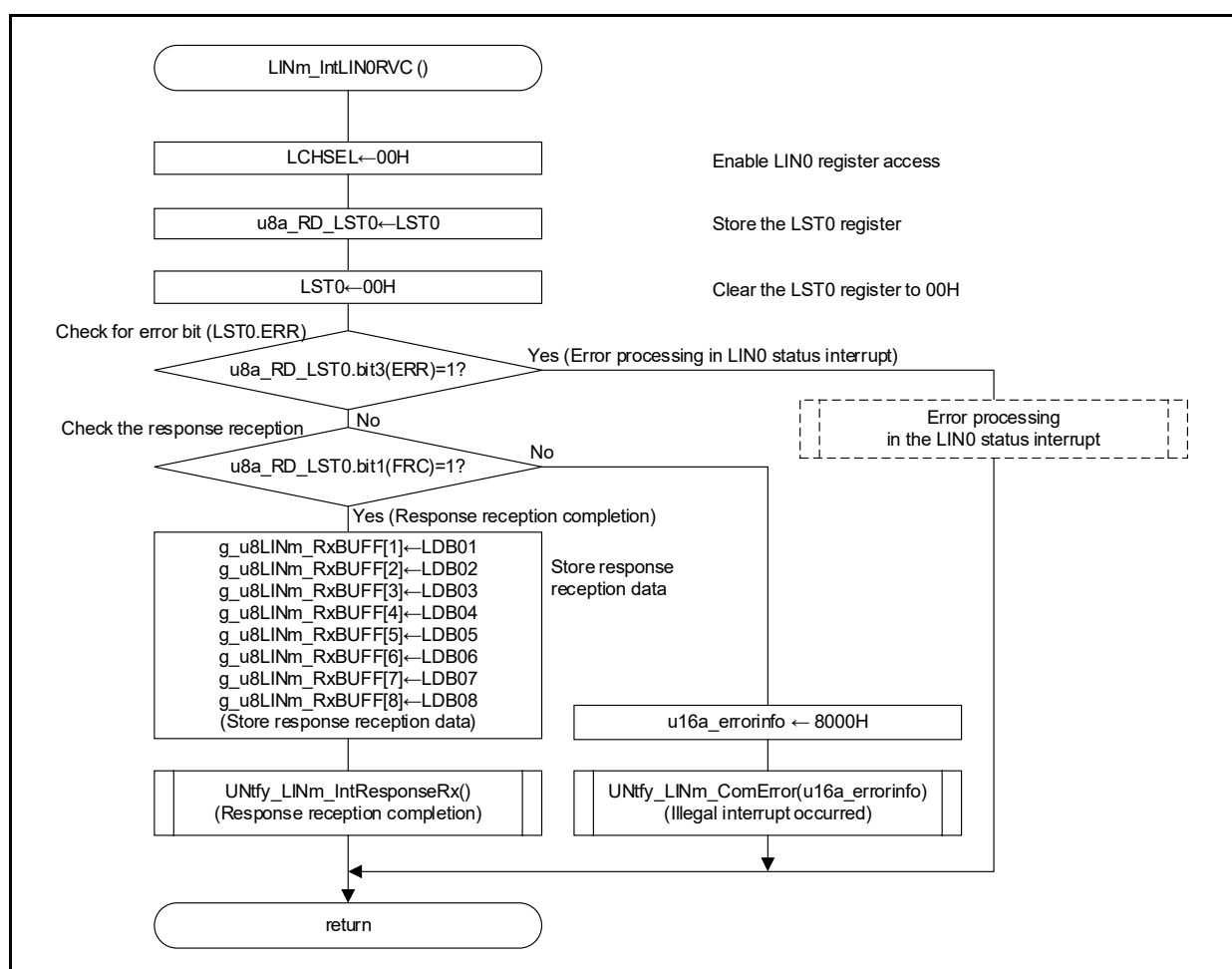


図 3.13 LIN Master Reception Completion Interrupt Processing Procedure (LINm_IntLIN0RVC)

3.4.3 LIN マスタ・ステータス・エラー割り込み処理

表 3.16 LIN Master Status Error Interrupt Processing Function (LINm_IntLIN0STA)

Syntax	void LINm_IntLIN0STA (void)	
Parameters	In	None
	Out	None
Local variables	unsigned char	u8a_RD_LEST0: Store LEST0 register
	- bit0 (BER):	Bit error detection in header / response / wake-up transmission
	- bit1 (PBER):	Physical bus error detection in header / wake-up transmission
	- bit2 (FTER):	Response timeout error detection in response tx/rx
	- bit3 (FER):	Framing error detection in response reception
	- bit5 (CSER):	Checksum error detection in response reception
	- bit7 (RPER):	Response preparation error detection in response reception
	unsigned short	u16a_errorinfo: Error information value
	- bit[7:0]:	u8a_RD_LEST0 value
	- bit[15:8]:	00H
Return value	None	
Call functions	UNtfy_LINm_ComError (u16a_errorinfo): Error detection processing	
Description	This function is the interrupt processing when LIN0 status error is occurred.	

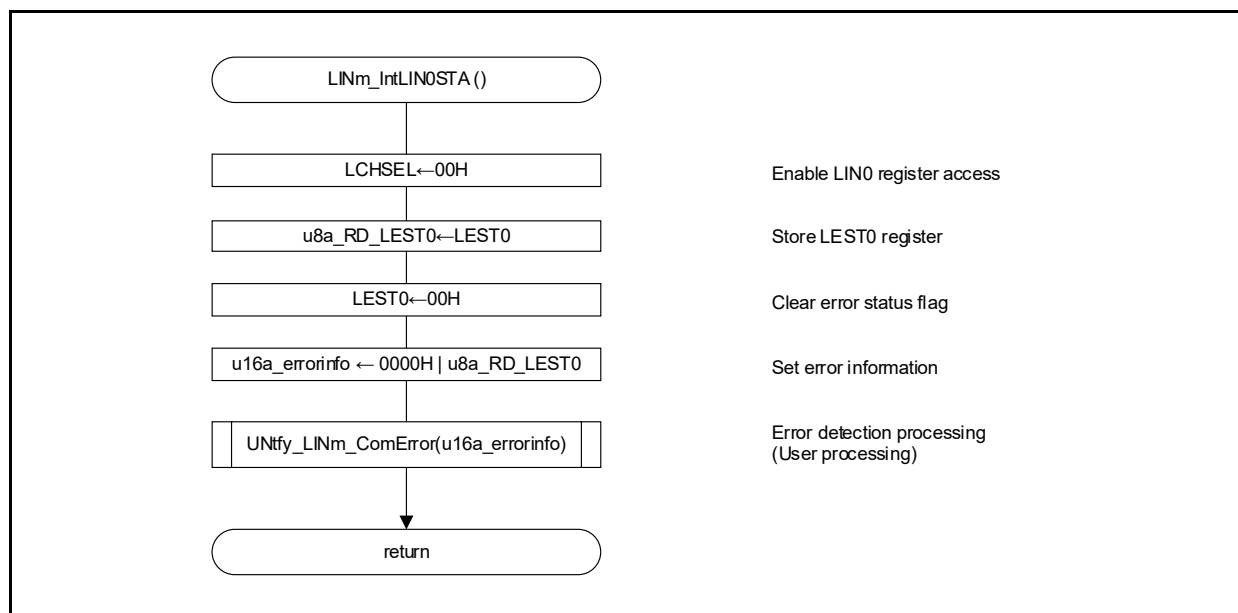


図 3.14 LIN Master Status Error Interrupt Processing Procedure (LINm_IntLIN0STA)

3.4.4 LIN マスタ・ウェイクアップ割り込み処理

表 3.17 LIN Master Wake-up Signal Detection Interrupt Processing Function (LINm_IntLIN0WUP)

Syntax		void LINm_IntLIN0WUP (void)
Parameters	In	None
	Out	None
Local variables		None
Return value		None
Call functions		UNTfy_LINm_IntWUPSigRx (): LRXD0 falling edge detection
Description		INTLIN0WUP interrupt service routine

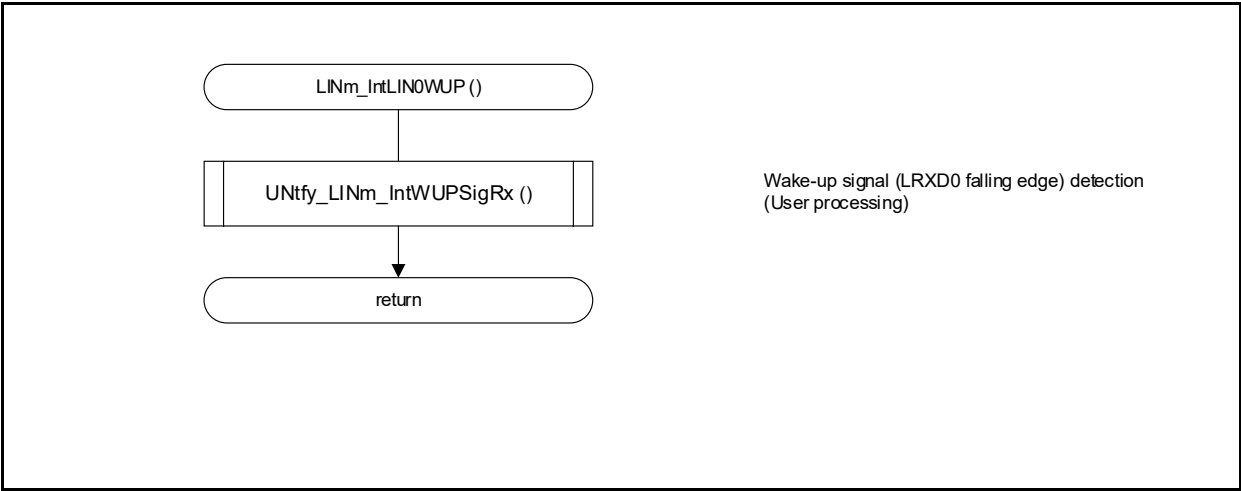


図 3.15 LIN Master Wake-up Signal Detection Interrupt Processing Procedure (LINm_IntLIN0WUP)

3.5 ユーザ通知関数 [API]

本アプリケーションノートで示す LIN マスタ通信ドライバ・ユーザ通知関数 API について説明します。

3.5.1 エラー通知関数 API

表 3.18 Error Notification Function API (UNTfy_LINm_ComError)

Syntax		void UNTfy_LINm_ComError (unsigned short u16a_errorinfo)	
Parameters	In	unsigned short u16a_errorinfo:	Error information value
		- 00xxH:	Communication error has occurred. "xxH": LEST0 register value
		- 4000H:	Illegal interrupt (INTLIN0TRM) has occurred.
		- 8000H:	Illegal interrupt (INTLIN0RVC) has occurred.
	Out	None	
Local variables		- (User processing)	
Return value		- (User processing)	
Call functions		- (User processing)	
Description		This function notifies that an error has occurred.	

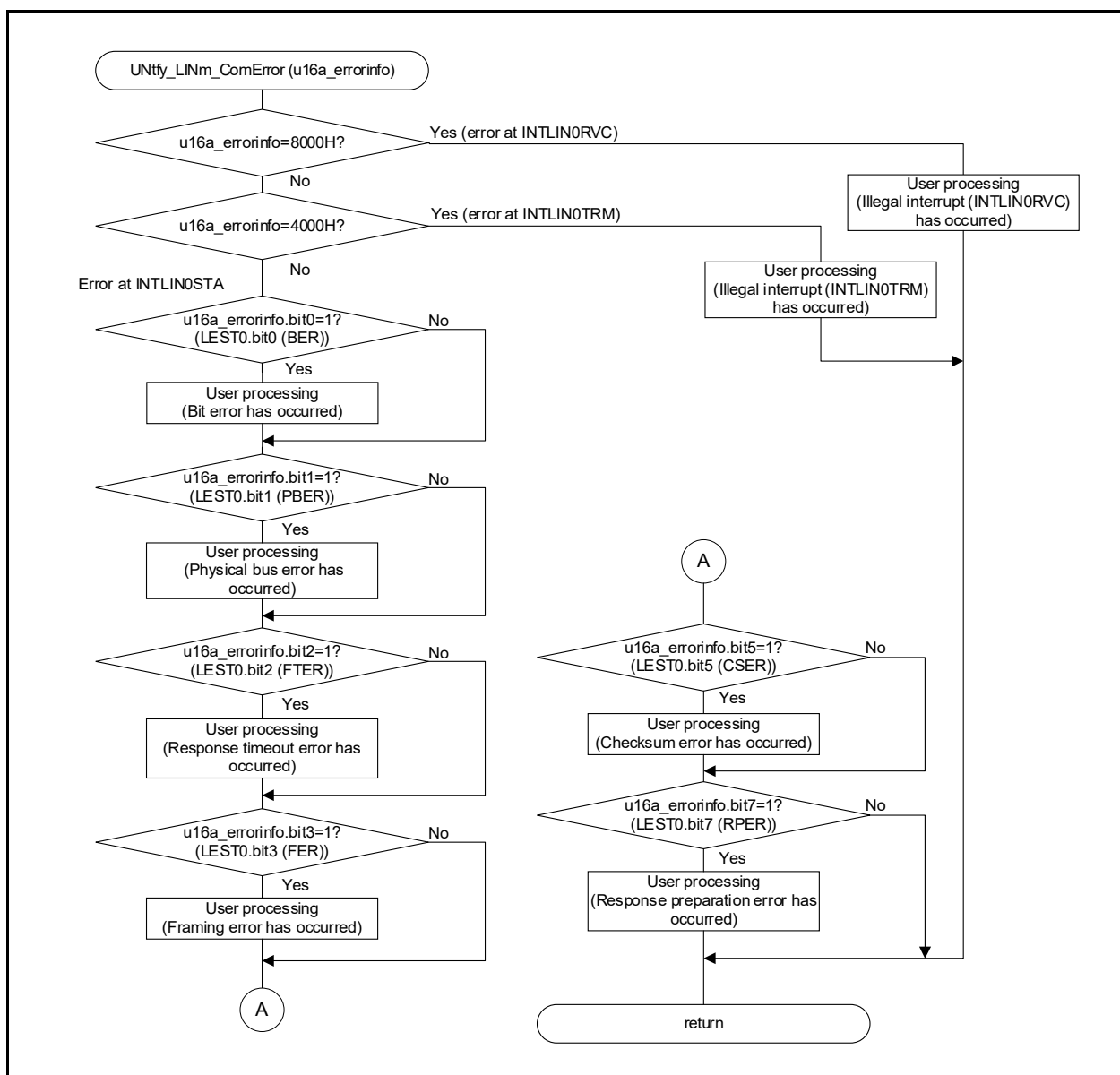


図 3.16 Example of Error Notification Procedure (UNTfy_LINm_ComError)

3.5.2 ヘッダ送信完了通知関数 API

表 3.19 Header Transmission Completion Notification Function API (UNtfy_LINm_IntHeaderTx)

Syntax	void UNtfy_LINm_IntHeaderTx (void)	
Parameters	In	None
	Out	None
Local variables	- (User processing)	
Return value	- (User processing)	
Call functions	- (User processing)	
Description	This function notifies that the header transmission is completed.	

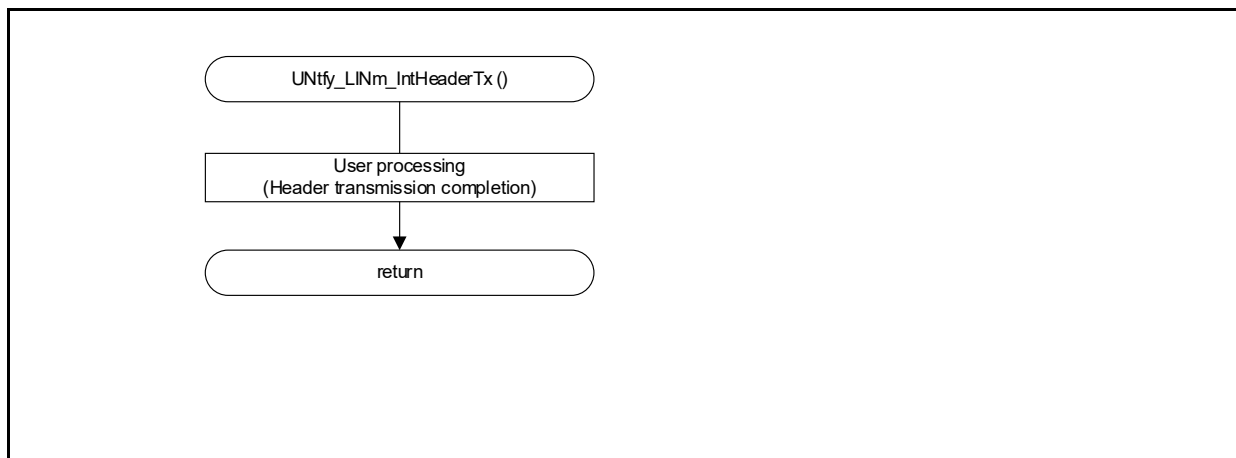


図 3.17 Example of Header Transmission Completion Notification Procedure (UNtfy_LINm_IntHeaderTx)

3.5.3 レスポンス送信完了通知関数 API

表 3.20 Response Transmission Completion Notification Function API (UNtfy_LINm_IntResponseTx)

Syntax	void UNtfy_LINm_IntResponseTx (void)	
Parameters	In	None
	Out	None
Local variables	- (User processing)	
Return value	- (User processing)	
Call functions	- (User processing)	
Description	This function notifies that the response transmission is completed.	

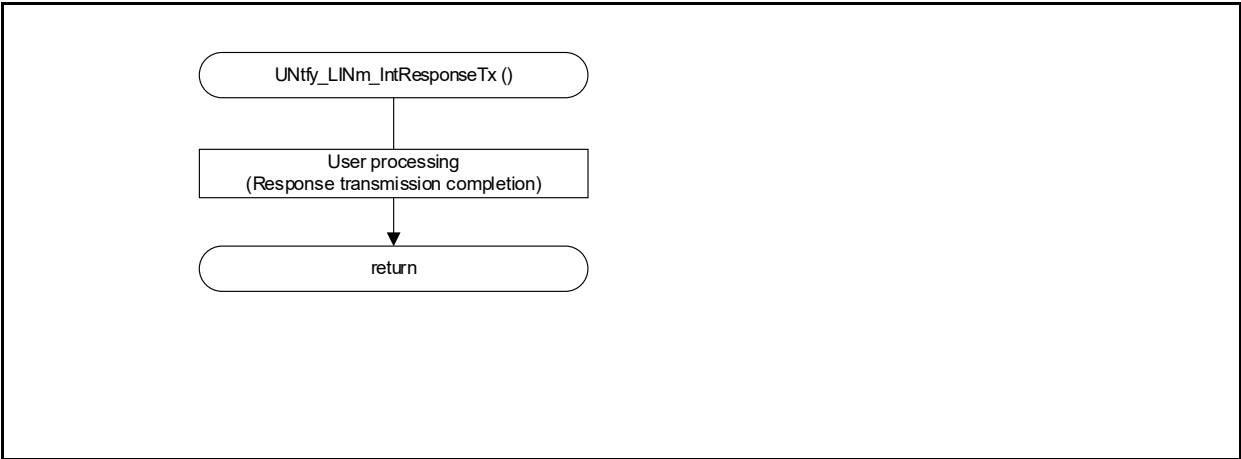


図 3.18 Example of Response Transmission Completion Notification Procedure (UNtfy_LINm_IntResponseTx)

3.5.4 レスポンス受信完了通知関数 API

表 3.21 Response Reception Completion Notification Function API (UNtfy_LINm_IntResponseRx)

Syntax		void UNtfy_LINm_IntResponseRx (void)	
Parameters	In	unsigned char g_u8LINm_RxBUFF[1:8]:	Response reception data
	Out	None	
Local variables		- (User processing)	
Return value		- (User processing)	
Call functions		- (User processing)	
Description		This function notifies that the response reception is completed.	

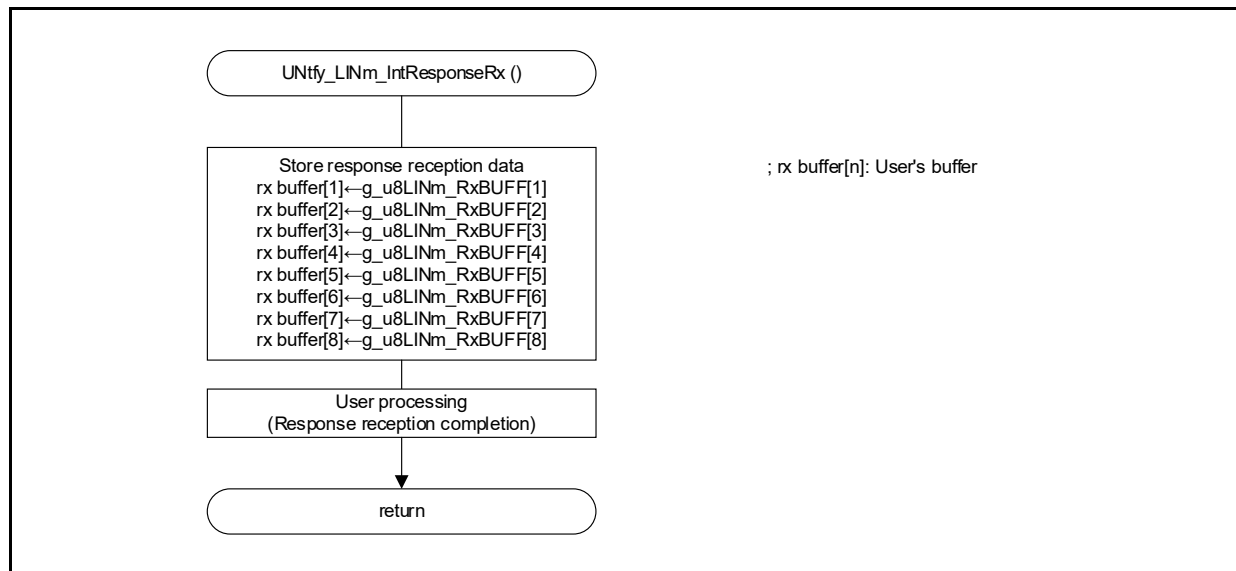


図 3.19 Example of Response Reception Completion Notification Procedure (UNtfy_LINm_IntResponseRx)

3.5.5 ウェイクアップ信号送信完了通知関数 API

表 3.22 Wake-up Signal Transmission Completion Notification Function API
(UNTfy_LINm_IntWUPSigTx)

Syntax	void UNTfy_LINm_IntWUPSigTx (void)	
Parameters	In	None
	Out	None
Local variables	- (User processing)	
Return value	- (User processing)	
Call functions	- (User processing)	
Description	This function notifies that the wake-up signal transmission is completed.	

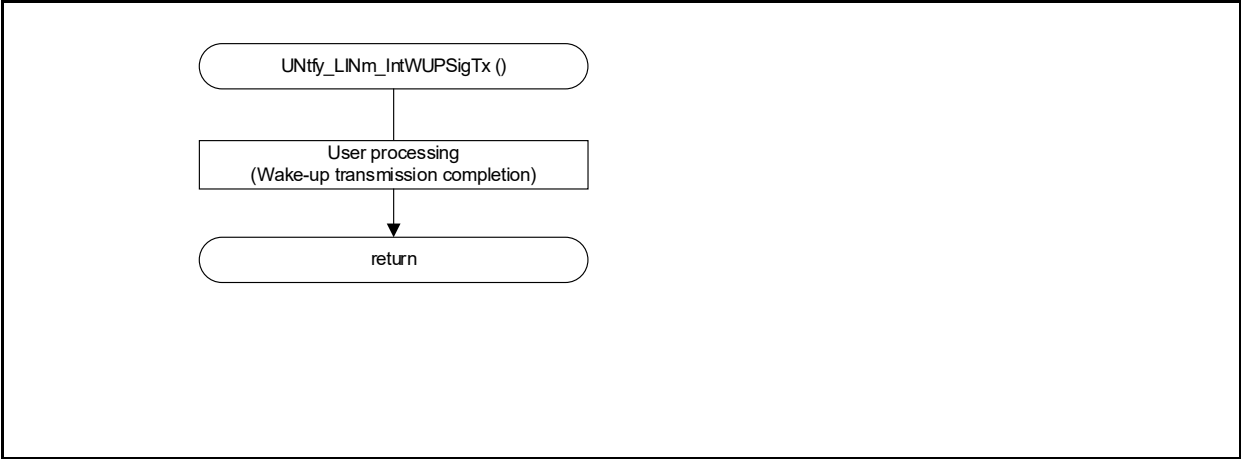


図 3.20 Example of Wake-up Signal Transmission Completion Notification Procedure
(UNTfy_LINm_IntWUPSigTx)

3.5.6 ウェイクアップ信号受信通知関数 API

表 3.23 Wake-up Signal Reception Notification Function API (UNtfy_LINm_IntWUPSigRx)

Syntax	void UNtfy_LINm_IntWUPSigRx (void)	
Parameters	In	None
	Out	None
Local variables	- (User processing)	
Return value	- (User processing)	
Call functions	- (User processing)	
Description	This function notifies that the LRXD0 falling edge is detected.	

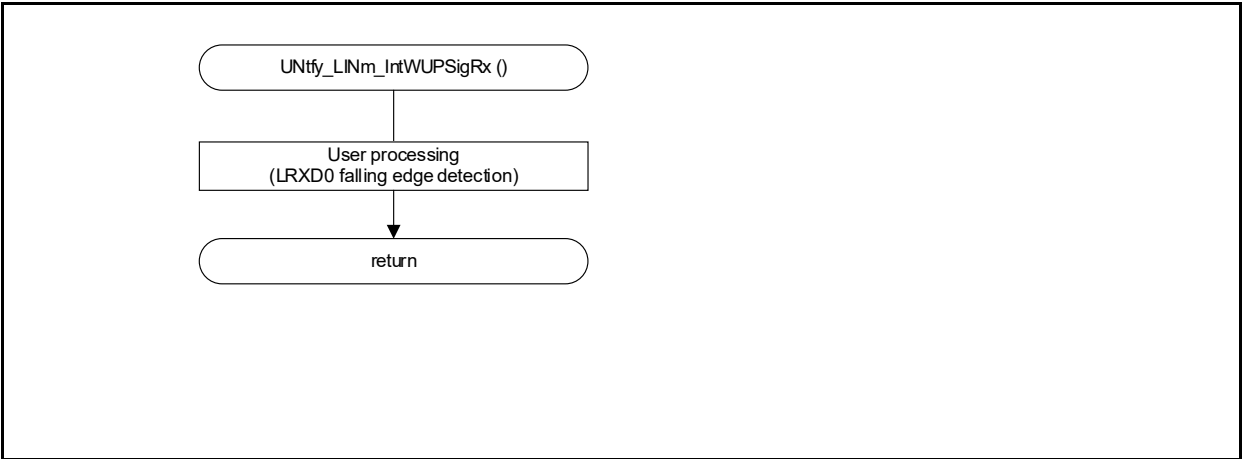


図 3.21 Example of Wake-up Signal Reception Notification Procedure (UNtfy_LINm_IntWUPSigRx)

4. LIN / UART モジュール（RLIN3）モード遷移

LCUCn レジスタの OM1, OM0 ビットで動作モードを選択した後で LMSTn レジスタの OMM1, OMM0 ビットを読み出すことで、LIN 動作モードの移行を確認することができます。

モード移行後の LIN 関連レジスタの設定は必ずモード移行の完了を確認した後で行ってください。

モード移行の最大時間を以下に示します。

表 4.1 LIN / UART Module (RLIN3) Mode Transition Times

Mode Transition	Transition Conditions	Transition Time ^{Note}
LIN reset mode → LIN operation mode	Set: LCUCn.[OM1,OM0]=11B Check: LMSTn.[OMM1, OMM0]=11B	(Max.) $3 \cdot f_{CLK} + 4 \cdot LINn$ communication clock source
LIN reset mode → LIN wake-up mode	Set: LCUCn.[OM1,OM0]=01B Check: LMSTn.[OMM1, OMM0]=01B	
LIN operation mode → LIN reset mode	Set: LCUCn.[OM1,OM0]=00B	
LIN wake-up mode → LIN reset mode	Check: LMSTn.[OMM1, OMM0]=00B	

Note. The time it takes for LMSTn.[OMM1,OMM0] to change after setting LCUCn.[OM1,OM0].

5. 参考資料

本アプリケーションノートの参考資料を以下に示します。参照の際は、ルネサスエレクトロニクスホームページから最新版を入手してください。

- RL78/F13, F14 ユーザーズマニュアル ハードウェア編 Rev.2.10
- RL78/F15 ユーザーズマニュアル ハードウェア編 Rev.1.00

6. 付録

ヘッダ ID 番号 (0~63) と、実際に送信する ID 値 (パリティを付加) の対応表を以下に示します。

表 6.1 LIN Header ID List

Header ID		Actual transmitted ID value (Note)	Response data length	Header ID		Actual transmitted ID value (Note)	Response data length
Decimal	Hexadecimal	Hexadecimal		Decimal	Hexadecimal	Hexadecimal	
0	0x00	0x80	2	32	0x20	0x20	4
1	0x01	0xC1	2	33	0x21	0x61	4
2	0x02	0x42	2	34	0x22	0xE2	4
3	0x03	0x03	2	35	0x23	0xA3	4
4	0x04	0xC4	2	36	0x24	0x64	4
5	0x05	0x85	2	37	0x25	0x25	4
6	0x06	0x06	2	38	0x26	0xA6	4
7	0x07	0x47	2	39	0x27	0xE7	4
8	0x08	0x08	2	40	0x28	0xA8	4
9	0x09	0x49	2	41	0x29	0xE9	4
10	0x0A	0xCA	2	42	0x2A	0x6A	4
11	0x0B	0x8B	2	43	0x2B	0x2B	4
12	0x0C	0x4C	2	44	0x2C	0xEC	4
13	0x0D	0x0D	2	45	0x2D	0xAD	4
14	0x0E	0x8E	2	46	0x2E	0x2E	4
15	0x0F	0xCF	2	47	0x2F	0x6F	4
16	0x10	0x50	2	48	0x30	0xF0	8
17	0x11	0x11	2	49	0x31	0xB1	8
18	0x12	0x92	2	50	0x32	0x32	8
19	0x13	0xD3	2	51	0x33	0x73	8
20	0x14	0x14	2	52	0x34	0xB4	8
21	0x15	0x55	2	53	0x35	0xF5	8
22	0x16	0xD6	2	54	0x36	0x76	8
23	0x17	0x97	2	55	0x37	0x37	8
24	0x18	0xD8	2	56	0x38	0x78	8
25	0x19	0x99	2	57	0x39	0x39	8
26	0x1A	0x1A	2	58	0x3A	0xBA	8
27	0x1B	0x5B	2	59	0x3B	0xFB	8
28	0x1C	0x9C	2	60	0x3C	0x3C	8
29	0x1D	0xDD	2	61	0x3D	0x7D	8
30	0x1E	0x5E	2	(62)	(reserved)	(reserved)	—
31	0x1F	0x1F	2	(63)	(reserved)	(reserved)	—

Note: P0(bit6) = ID0⊕ID1⊕ID2⊕ID4, P1(bit7) = !(ID1⊕ID3⊕ID4⊕ID5), P: Parity, ⊕: XOR, !: INV

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	Sep.30.21	－	初版作成

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本ドキュメントおよびテクニカルアップデートを参照してください。

1. 静電気対策

CMOS 製品の取り扱いの際は静電気防止を心がけてください。CMOS 製品は強い静電気によってゲート絶縁破壊を生じることがあります。運搬や保存の際には、当社が出荷梱包に使用している導電性のトレーやマガジンケース、導電性の緩衝材、金属ケースなどを利用し、組み立て工程にはアースを施してください。プラスチック板上に放置したり、端子を触ったりしないでください。また、CMOS 製品を実装したボードについても同様の扱いをしてください。

2. 電源投入時の処置

電源投入時は、製品の状態は不定です。電源投入時には、LSI の内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. 電源オフ時における入力信号

当該製品の電源がオフ状態のときに、入力信号や入出力プルアップ電源を入れないでください。入力信号や入出力プルアップ電源からの電流注入により、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。資料中に「電源オフ時における入力信号」についての記載のある製品は、その内容を守ってください。

4. 未使用端子の処理

未使用端子は、「未使用端子の処理」に従って処理してください。CMOS 製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI 周辺のノイズが印加され、LSI 内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。

5. クロックについて

リセット時は、クロックが安定した後、リセットを解除してください。プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

6. 入力端子の印加波形

入力ノイズや反射波による波形歪みは誤動作の原因になりますので注意してください。CMOS 製品の入力がノイズなどに起因して、 V_{IL} (Max.) から V_{IH} (Min.) までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定の場合はもちろん、 V_{IL} (Max.) から V_{IH} (Min.) までの領域を通過する遷移期間中にチャタリングノイズなどが入らないように使用してください。

7. リザーブアドレス（予約領域）のアクセス禁止

リザーブアドレス（予約領域）のアクセスを禁止します。アドレス領域には、将来の拡張機能用に割り付けられている リザーブアドレス（予約領域）があります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

8. 製品間の相違について

型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。同じグループのマイコンでも型名が違くと、フラッシュメモリ、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。回路、ソフトウェアおよびこれらに関連する情報を使用する場合、お客様の責任において、お客様の機器・システムを設計ください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含まれます。以下同じです。）に関し、当社は、一切その責任を負いません。
2. 当社製品または本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
3. 当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
4. 当社製品を組み込んだ製品の輸出入、製造、販売、利用、配布その他の行為を行うにあたり、第三者保有の技術の利用に関するライセンスが必要となる場合、当該ライセンス取得の判断および取得はお客様の責任において行ってください。
5. 当社製品を、全部または一部を問わず、改造、改変、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改変、複製、リバースエンジニアリング等により生じた損害に関し、当社は、一切その責任を負いません。
6. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。

標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット等

高品質水準： 輸送機器（自動車、電車、船舶等）、交通制御（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等

当社製品は、データシート等により高信頼性、**Harsh environment** 向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じて、当社は一切その責任を負いません。

7. あらゆる半導体製品は、外部攻撃からの安全性を 100%保証されているわけではありません。当社ハードウェア／ソフトウェア製品にはセキュリティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害（当社製品または当社製品が使用されているシステムに対する不正アクセス・不正使用を含みますが、これに限りません。）から生じる責任を負うものではありません。当社は、当社製品または当社製品が使用されたあらゆるシステムが、不正な改変、攻撃、ウイルス、干渉、ハッキング、データの破壊または窃盗その他の不正な侵入行為（「脆弱性問題」といいます。）によって影響を受けないことを保証しません。当社は、脆弱性問題に起因したまたはこれに関連して生じた損害について、一切責任を負いません。また、法令において認められる限りにおいて、本資料および当社ハードウェア／ソフトウェア製品について、商品性および特定目的との合致に関する保証ならびに第三者の権利を侵害しないことの保証を含め、明示または黙示のいかなる保証も行いません。
8. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は、データシート等において高信頼性、**Harsh environment** 向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
10. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
11. 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
12. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものいたします。
13. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
14. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。

注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

(Rev.5.0-1 2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

www.renesas.com

お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

www.renesas.com/contact/

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の

商標です。すべての商標および登録商標は、それぞれの所有者に帰属します。