

RENESAS TECHNICAL UPDATE

〒135-0061 東京都江東区豊洲 3-2-24 豊洲フォレシア
ルネサス エレクトロニクス株式会社問合せ窓口 <http://japan.renesas.com/contact/>E-mail: csc@renesas.com

製品分類	MPU & MCU	発行番号	TN-RL*-A0147A/J	Rev.	第1版
題名	誤記訂正通知 RL78/F23, F24 ユーザーズマニュアル Rev.1.10 の誤記通知		情報分類	技術情報	
適用製品	RL78/F23, F24 グループ	対象ロット等	関連資料	RL78/F23, F24 グループ ユーザーズマニュアル ハードウェア編 Rev.1.10 (R01UH0944JJ0110)	
		全ロット			

RL78/F23, F24 ユーザーズマニュアル ハードウェア編 Rev.1.10 (R01UH0944JJ0110) において、下記訂正がございます。

今回通知する訂正内容

No	訂正内容	R01UH0044JJ0110 の該当ページ	本通知の 該当ページ
1	PWM オプション・ユニット A (PWMOPA) を使用する際の注意事項の記載追加	P.549, P.653	P.2
2	シリアル・インタフェース IICA IICWL, IICWH レジスタの設定に関する記載追加	P.1029, P.1033	P.3
3	CAN インタフェース (RS-CANFD lite) 誤記訂正	P.1278, P.1279, P.1303, P.1310, P.1311, P.1321, P.1322, P.1330, P.1348, P.1351, P.1394, P.1395, P.1476, P.1501	P.4
4	SNOOZE ステータス出力設定例の誤記訂正	P.1613	P.12
5	CFERRCTRL レジスタのビット 2 の説明追加	P.1731	P.14
6	FLWE レジスタの説明追加	P.1738	P.14
7	フラッシュ・メモリの誤記訂正	P.1759, P.1767, P.1781, P.1782, P.1805, P.1806, P.1812, P.1814, P.1818, P.1819, P.1821, P.1822,	P.14
8	フラッシュ・メモリの説明追加	P.1767, P.1781, P.1818	P.21
9	電気的特性の誤記修正	P.1861, P.1879, P.1880, P.1890, P.1896, P.1914, P.1932, P.1933, P.1943, P.1949, P.1966, P.1984, P.1985, P.1995, P.2001	P.22
10	外形図 (32 ピン) の記載追加	P.2012	P.23

誤記訂正の該当箇所は、誤 太字下線、正 グレー・ハッチング で記載します。

No.1: PWM オプション・ユニット A (PWMOPA) を使用する場合の注意事項の記載追加**● 8.2.32 PWMDLY0 レジスタに注意 7 の記載を追加**

誤)

注意 1. PWMDLY0 レジスタは PWM 出力前に設定してください。

:

6. 本レジスタを遅延ありに設定しても TRDIO_{ji} 端子機能と同じ端子に兼用されている他の端子機能は遅延しません (j = A, B, C, D, i = 0, 1)。

正) 注意.7 の記載追加

注意 1. PWMDLY0 レジスタは PWM 出力前に設定してください。

:

6. 本レジスタを遅延ありに設定しても TRDIO_{ji} 端子機能と同じ端子に兼用されている他の端子機能は遅延しません (j = A, B, C, D, i = 0, 1)。

7. リセット同期 PWM モード、相補 PWM モード、PWM3 モード、および拡張相補 PWM モードを使用し、かつ PWM オプション・ユニット A (PWMOPA) を使用する場合、対象端子の TRDji[1, 0]ビットは "00B" (遅延なし) にしてください (i = 0, 1, j = A, B, C, D)。

● 8.6.3.8 注意事項に (8), (9) の記載を追加

誤)

- (1) タイマ RDe のパルス出力強制遮断が PWMOPA の出力遮断と同時に動作する場合の優先順位を次の表に示します。

:

- (7) コンパレータ 0 と INTP0 の入力パルス幅を PWMOPA 動作クロックの 1 サイクル以上に設定してください。

正) 注意事項 (8), (9) の記載追加

- (1) タイマ RDe のパルス出力強制遮断が PWMOPA の出力遮断と同時に動作する場合の優先順位を次の表に示します。

:

- (7) コンパレータ 0 と INTP0 の入力パルス幅を PWMOPA 動作クロックの 1 サイクル以上に設定してください。

- (8) リセット同期 PWM モード、相補 PWM モード、PWM3 モード、および拡張相補 PWM モードを使用し、かつ PWM オプション・ユニット A (PWMOPA) を使用する場合、対象端子の PWMDLY0 レジスタの TRDji[1, 0] ビットは、00B (遅延なし) にしてください (i = 0, 1, j = A, B, C, D)。

- (9) PWM 機能および拡張 PWM モードの時に、PWMDLY0 レジスタで PWM 出力を遅延し、かつ PWM オプション・ユニット A (PWMOPA) を使用した場合、PWMOPA の強制遮断解除後に遅延設定した端子から出力遮断直前に遅延回路に保持されたレベルが遅延設定値-1 のパルス分出力されます。

No.2: シリアル・インタフェース IICA IICWL, IICWH レジスタの設定に関する記載追加**● 16.3.7 IICWH0 レジスタ 備考の記載をレジスタ説明部に移動し、備考に記載を追加**

誤)

16.3.7 IICA ハイ・レベル幅設定レジスタ 0 (IICWH0)

シリアル・インタフェース IICA が、出力する SCLA0 端子信号のハイ・レベル幅を設定するレジスタです。

IICWH0 レジスタは、8 ビット・メモリ操作命令で設定します。

IICWH0 レジスタは、I²C が動作禁止 (IICA コントロール・レジスタ 00 (IICCTL00) のビット 7 (IICE0) = 0) のときに設定してください。

リセット信号の発生により、FFH になります。

備考 IICWL0, IICWH0 レジスタによる転送クロックの設定方法は「16.4.2 IICWL0, IICWH0 レジスタによる転送クロック設定方法」を参照してください。

正)

16.3.7 IICA ハイ・レベル幅設定レジスタ 0 (IICWH0)

シリアル・インタフェース IICA が、出力する SCLA0 端子信号のハイ・レベル幅を設定するレジスタです。

IICWH0 レジスタは、8 ビット・メモリ操作命令で設定します。

IICWH0 レジスタは、I²C が動作禁止 (IICA コントロール・レジスタ 00 (IICCTL00) のビット 7 (IICE0) = 0) のときに設定してください。

IICWH0 の設定方法については、「17.4.2 IICWL0, IICWH0 レジスタによる転送クロック設定方法」を参照してください。

リセット信号の発生により、FFH になります。

備考 IICWH0 の設定方法については、「17.4.2 IICWL0, IICWH0 レジスタによる転送クロック設定方法」を参照してください。

● 16.4.2 IICWL0, IICWH0 レジスタによる転送クロック設定方法 注意 3 の記載を追加

誤)

注意 2. 転送クロックを設定する場合は、fCLK の最低動作周波数に注意してください。シリアル・インタフェース IICA はモードによって fCLK の最低動作周波数が決められています。

ファースト・モード時 : fCLK = 3.5 MHz (Min.)

ファースト・モード・プラス時 : fCLK = 10 MHz (Min.)

標準モード時 : fCLK = 1 MHz (Min.)

正)

注意 2. 転送クロックを設定する場合は、fCLK の最低動作周波数に注意してください。シリアル・インタフェース IICA はモードによって fCLK の最低動作周波数が決められています。

ファースト・モード時 : fCLK = 3.5 MHz (Min.)

ファースト・モード・プラス時 : fCLK = 10 MHz (Min.)

標準モード時 : fCLK = 1 MHz (Min.)

3. シリアル・クロックの最小周期は、(IICWL0 + 1) + (IICWH0 + 1) になります。SDAA0, SCLA0 信号の立ち上がり時間 (tR) と立ち下がり時間 (tF) を考慮し、IICWL0, IICWH0 レジスタに設定する値を決めてください。

No.3: CAN インタフェース (RS-CANFD lite) 誤記訂正**● 18.3.2 C0CTRL.RTBO ビットの説明の誤記訂正**

誤)

• RTBO

CAN プロトコル・コントローラがバスオフ状態のときに、C0CTR.RTBO ビットを 1 に設定すると強制的にバスオフ状態からエラー・アクティブ状態になります。

バスオフ状態からエラー・アクティブ状態への遷移は、最大 1CAN ビット・タイム遅延します。

このビットを 1 にすると、C0STS.REC および C0STS.TEC ビットは 00H に初期化され、C0STS.BOSTS は 0 (バスオフ状態ではない) になります。他のレジスタは変化しません。

正)

• RTBO

CAN プロトコル・コントローラがバスオフ状態のときに、C0CTR.RTBO ビットを 1 に設定すると強制的にバスオフ状態からエラー・アクティブ状態になります。

このビットは、C0CTR.BOM[1:0]ビットが 00B (バスオフから復帰は ISO 11898-1 仕様準拠) のときに使用してください。

バスオフ状態からエラー・アクティブ状態への遷移は、最大 1CAN ビット・タイム遅延します。

このビットを 1 にすると、C0STS.REC および C0STS.TEC ビットは 00H に初期化され、C0STS.BOSTS は 0 (バスオフ状態ではない) になります。他のレジスタは変化しません。

● 18.3.12 GCTR.H.TSRST ビットの説明の誤記訂正

誤)

• TSRST

GCTR.TRST を 1 にすると、CAN タイム・スタンプ・レジスタ (GTSC) は 0000H にリセットされます。

このビットは、グローバル・スリープ・モードのときに書き込むことができません。グローバル・リセット・モード時に設定してください。

このビットを読むと、常に 0 が読み出されます。

正)

• TSRST

GCTR.TRST を 1 にすると、CAN タイム・スタンプ・レジスタ (GTSC) は 0000H にリセットされます。

このビットは、グローバル・スリープ・モードのときに書き込むことができません。また、グローバル・リセット・モード時はこのビットに書き込まないでください。

このビットを読むと、常に 0 が読み出されます。

● 18.3.16 GTSC.TS[15:0]ビットの説明追加

誤)

• TS[15:0]

タイム・スタンプ・カウンタの値を読み出すことができます。

GCFG.TSSS でクロック源を選択し、GCFG.TSP で選択されたクロック源を分周したクロックを使用してタイム・スタンプ・カウンタはフリーラン・カウント動作を行います。

動作モードからグローバル HALT モードまたはチャンネル HALT モードに遷移する場合、GTSC.TS の値は保証されません。

このビットは、グローバル・リセット・モード時に 0000H になります。

正)

• TS[15:0]

タイム・スタンプ・カウンタの値を読み出すことができます。

GCFG.TSSS でクロック源を選択し、GCFG.TSP で選択されたクロック源を分周したクロックを使用してタイム・スタンプ・カウンタはフリーラン・カウント動作を行います。

動作モードからグローバル HALT モードまたはチャンネル HALT モードに遷移する場合、GTSC.TS の値は保証されません。

このビットは、グローバル・リセット・モードまたはグローバル・スリープ・モードのときに書き込むことができません。

グローバル・リセット・モード時に 0000H になります。

● 18.3.17 GAFLECTR.AFLDAE ビットの誤記訂正

誤)

• AFLDAE

1 にすると受信ルールの書き込みを許可します。

受信ルールは、GAFLECTR.AFLDAE の設定にかかわらず読み出すことができます。

このビットは、グローバル・リセット・モードでは書き込むことができません。他のモードで設定してください。

正)

• AFLDAE

1 にすると受信ルールの書き込みを許可します。

受信ルールは、GAFLECTR.AFLDAE の設定にかかわらず読み出すことができます。

このビットは、グローバル・スリープ・モードでは書き込むことができません。他のモードで設定してください。

● 18.3.24 RMND.RMNS[15:0]ビットの説明追加

誤)

• RMNS[15:0]

対応する受信バッファにメッセージの格納を開始すると 1 になります。

このビットをクリアしたい場合、クリアするビットに 0 を、クリアしないビットには 1 の値を設定し、MOV 命令を使用して書き込んでください。

メッセージ格納中は、0 にクリアすることができません。

メッセージを格納する時間は RMNB.RMPLS が 000B (受信ペイロード・サイズ=8) の場合、6 クロック (pclk = fCLK) になります。RMNB.RMPLS が 000B より大きい場合、 $6 + [4 \text{ バイト毎に } 1]$ になります (64 バイト・ペイロードの場合、 $6 + (64-8)/4 = 20$ クロック)。

このビットは、グローバル・リセット・モード時は 0000H になります。

正)

• RMNS[15:0]

対応する受信バッファにメッセージの格納を開始すると 1 になります。

このビットをクリアしたい場合、クリアするビットに 0 を、クリアしないビットには 1 の値を設定し、MOV 命令を使用して書き込んでください。

メッセージ格納中は、0 にクリアすることができません。

メッセージを格納する時間は RMNB.RMPLS が 000B (受信ペイロード・サイズ=8) の場合、6 クロック (pclk = fCLK) になります。RMNB.RMPLS が 000B より大きい場合、 $6 + [4 \text{ バイト毎に } 1]$ になります (64 バイト・ペイロードの場合、 $6 + (64-8)/4 = 20$ クロック)。

このビットは、グローバル・リセット・モードまたはグローバル・スリープ・モードでは書き込むことができません。

グローバル・リセット・モード時は 0000H になります。

● 18.3.28 CFCC.CFIGCV[2:0]ビットの説明追加

誤)

• CFIGCV[2:0]

CFCC.CFM が 0 (受信 FIFO モードで使用) で CFCC.CFIM が 0 の場合、送受信 FIFO 受信割り込み要求を発生させる受信メッセージ格納数を設定します。

CFCC.CFDC に設定したバッファ数に対する分数で指定します。CFCC.CFDC が 001B (4 メッセージ) の場合、CFCC.CFIGCV は 001B (1/4 メッセージ格納)、011B (2/4 メッセージ格納)、101B (3/4 メッセージ格納) または 111B (バッファ・フル) から選択してください。詳細は、「18.8.2.1.5 FIFO 割り込みの設定」を参照してください。

このビットは、グローバル・リセット・モード時に設定してください。

正)

• CFIGCV[2:0]

CFCC.CFM が 0 (受信 FIFO モードで使用) で CFCC.CFIM が 0 の場合、送受信 FIFO 受信割り込み要求を発生させる受信メッセージ格納数を設定します。

CFCC.CFDC に設定したバッファ数に対する分数で指定します。CFCC.CFDC が 001B (4 メッセージ) の場合、CFCC.CFIGCV は 001B (1/4 メッセージ格納)、011B (2/4 メッセージ格納)、101B (3/4 メッセージ格納) または 111B (バッファ・フル) から選択してください。詳細は、「18.8.2.1.5 FIFO 割り込みの設定」を参照してください。

このビットは、グローバル・スリープ・モードでは書き込むことができません。

グローバル・リセット・モード時に設定してください。

● 18.3.28 CFCCL.CFITR ビットの誤記訂正

誤)

• CFITR

送受信 FIFO 送信モード時のインターバル・タイマの基準クロックを選択します。

0 のとき、fCLK を GCFG.ITRCP で分周したクロックになります。

1 のときは、fCLK を GCFG.ITRCP の設定値 × 10 で分周したクロックになります。

このビットは、グローバル・スリープ・モードでは書き込むことができません。CFCC.CFE が 0 (**送受信 FIFO 許可**) のときに書き換えてください。

正)

• CFITR

送受信 FIFO 送信モード時のインターバル・タイマの基準クロックを選択します。

0 のとき、fCLK を GCFG.ITRCP で分周したクロックになります。

1 のときは、fCLK を GCFG.ITRCP の設定値 × 10 で分周したクロックになります。

このビットは、グローバル・スリープ・モードでは書き込むことができません。CFCC.CFE が 0 (**送受信 FIFO 禁止**) のときに書き換えてください。

● 18.3.41 TMIEC.TMIE[3:0]ビットの誤記訂正

誤)

• TMIE[3:0]

送受信 FIFO 送信モード時のインターバル・タイマの基準クロックを選択します。このビットに 1 を書くと、対応する送信バッファの送信が完了すると割り込み要求を生成します。

送信バッファ割り込みの仕様については「18.12 割り込み」を参照してください。

このビットは、グローバル・スリープ・モードでは書き込むことができません。また、チャンネル・スリープ・モード時に **1 を設定しないでください。**

CFCC.CFTML ビットで送信バッファが送受信 FIFO バッファにリンク設定している場合は、対応するビットに 1 を書き込まないでください。

正)

• TMIE[3:0]

送受信 FIFO 送信モード時のインターバル・タイマの基準クロックを選択します。このビットに 1 を書くと、対応する送信バッファの送信が完了すると割り込み要求を生成します。

送信バッファ割り込みの仕様については「18.12 割り込み」を参照してください。

このビットは、グローバル・スリープ・モードでは書き込むことができません。また、チャンネル・スリープ・モード時に **書き込まないでください。**

CFCC.CFTML ビットで送信バッファが送受信 FIFO バッファにリンク設定している場合は、対応するビットに 1 を書き込まないでください。

● 18.3.43 THLSTS.THLIF ビットの誤記訂正

誤)

• THLIF

THLCC.THLIM で選択した送信履歴割り込み要因が発生すると 1 になります。

このビットは、1 のときに 0 を書き込むことでクリアすることができます。1 を書いても変化しません。プログラムによる 0 クリアと検出による 1 セットが同時の場合、このビットは 1 になります。

このビットは、グローバル動作モードまたはグローバル HALT モード時に書いてください。チャンネル・リセット・モードのときは 0 になります。

正)

• THLIF

THLCC.THLIM で選択した送信履歴割り込み要因が発生すると 1 になります。

このビットは、1 のときに 0 を書き込むことでクリアすることができます。1 を書いても変化しません。プログラムによる 0 クリアと検出による 1 セットが同時の場合、このビットは 1 になります。

このビットは、チャンネル動作モードまたはチャンネル HALT モード時に書いてください。チャンネル・リセット・モードのときは 0 になります。

● 18.3.43 THLSTS.THLELT ビットの誤記訂正

誤)

• THLELT

送信履歴バッファ・フルの状態、さらに新しい送信履歴データを格納しようとしたときに 1 になります。

この場合、新しいデータは破棄します。

このビットは、1 のときに 0 を書き込むことでクリアすることができます。1 を書いても変化しません。プログラムによる 0 クリアと検出による 1 セットが同時の場合、このビットは 1 になります。

このビットは、グローバル動作モードまたはグローバル HALT モード時に書いてください。チャンネル・リセット・モードのときは 0 になります。

正)

• THLELT

送信履歴バッファ・フルの状態、さらに新しい送信履歴データを格納しようとしたときに 1 になります。

この場合、新しいデータは破棄します。

このビットは、1 のときに 0 を書き込むことでクリアすることができます。1 を書いても変化しません。プログラムによる 0 クリアと検出による 1 セットが同時の場合、このビットは 1 になります。

このビットは、チャンネル動作モードまたはチャンネル HALT モード時に書いてください。チャンネル・リセット・モードのときは 0 になります。

● 18.5.9 CFIDH.CFRTR ビットの記載訂正

誤)

• CFRTR

CFCC.CFM が 0 (受信モード) の場合、送受信 FIFO バッファに格納された受信メッセージのフレーム・フォーマットを示します。

注意 1. **CAN-FD** にはリモート・フレームはありません。**CAN-FD** フレームを受信した場合、**RRS** の情報を示します。
CFCC.CFM が 1 (送信モード) の場合、送受信 FIFO バッファから送信するメッセージのデータ・フォーマットを設定します。

2. **CFFDCSTS.CFFDF** が 1 (**CAN-FD** フレーム) の場合、ドミナント (データ・フレーム) を送信します。

正)

• CFRTR

CFCC.CFM が 0 (受信モード) の場合、送受信 FIFO バッファに格納された受信メッセージのフレーム・フォーマットを示します。**CFCC.CFM が 1 (送信モード) の場合、送受信 FIFO バッファから送信するメッセージのデータ・フォーマットを設定します。**

注意 1. **CAN-FD** にはリモート・フレームはありません。**CAN-FD** フレームを受信した場合、**RRS** の情報を示します。
2. **CFFDCSTS.CFFDF** が 1 (**CAN-FD** フレーム) の場合、ドミナント (データ・フレーム) を送信します。

● 18.5.9 CFID.CFID[28:0] ビットの記載訂正

誤)

• CFID[28:0]

CFCC.CFM が 0 (受信モード) の場合、受信 FIFO バッファに格納されたメッセージの ID を示します。

CFCC.CFM が 1 (送信モード) の場合、送受信 FIFO バッファから送信するメッセージの ID を設定します。

ビット配置については、「18.4 ID フォーマット」を参照してください。

正)

• CFID[28:0]

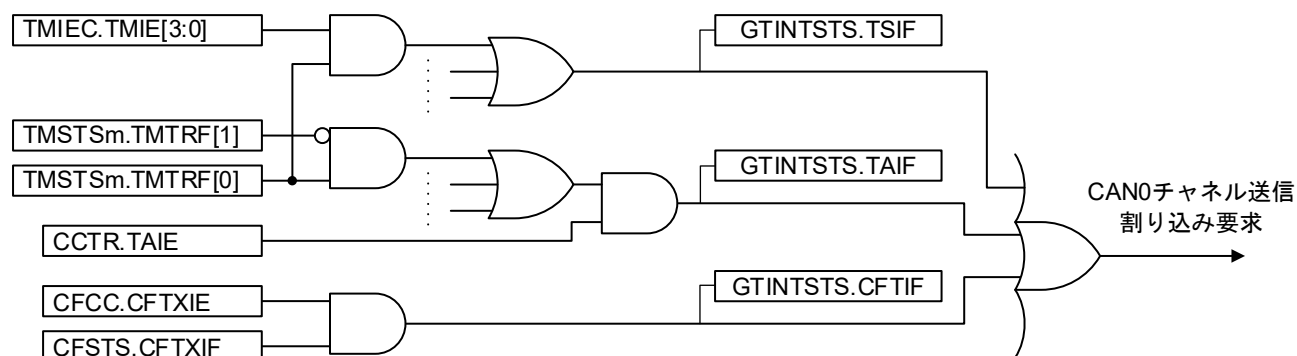
CFCC.CFM が 0 (受信モード) の場合、送受信 FIFO バッファに格納されたメッセージの ID を示します。

CFCC.CFM が 1 (送信モード) の場合、送受信 FIFO バッファから送信するメッセージの ID を設定します。

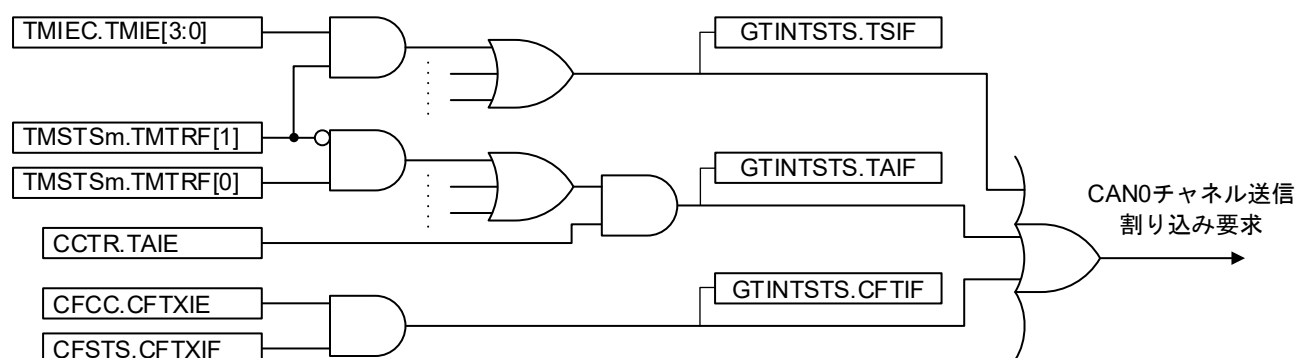
ビット配置については、「18.4 ID フォーマット」を参照してください。

● 18.12.1 図 18-45 図内の TMSTSm.TMTRF[1:0]の記載訂正

誤) TMSTSm.TMTRF[0]から信号 GTINTSTS.TSIF を生成



正) TMSTSm.TMTRF[1]から信号 GTINTSTS.TSIF を生成



● 18.17 注意事項に「18.12.2」で示す説明を追加**誤)**

- CAN モジュールを使用する場合、以下のクロック条件で使用してください。
 - fMP と fCLK は同じクロック源にしてください。また、MDIV レジスタで fMP/2 を選択し、fCLK = fMP/2 にしてください。
 - 高速オンチップ・オシレータは fMP と fCLK のクロック源として使用できますが、fCAN (CAN 通信クロック) には使用しないでください。
 - GCFG.DCS に 1 (CAN クロック源に X1 クロックを選択) を設定する場合、X1 クロックの周波数は、fCLK 以下としてください。
-

正)

- CAN モジュールを使用する場合、以下のクロック条件で使用してください。
 - fMP と fCLK は同じクロック源にしてください。また、MDIV レジスタで fMP/2 を選択し、fCLK = fMP/2 にしてください。
 - 高速オンチップ・オシレータは fMP と fCLK のクロック源として使用できますが、fCAN (CAN 通信クロック) には使用しないでください。
 - GCFG.DCS に 1 (CAN クロック源に X1 クロックを選択) を設定する場合、X1 クロックの周波数は、fCLK 以下としてください。

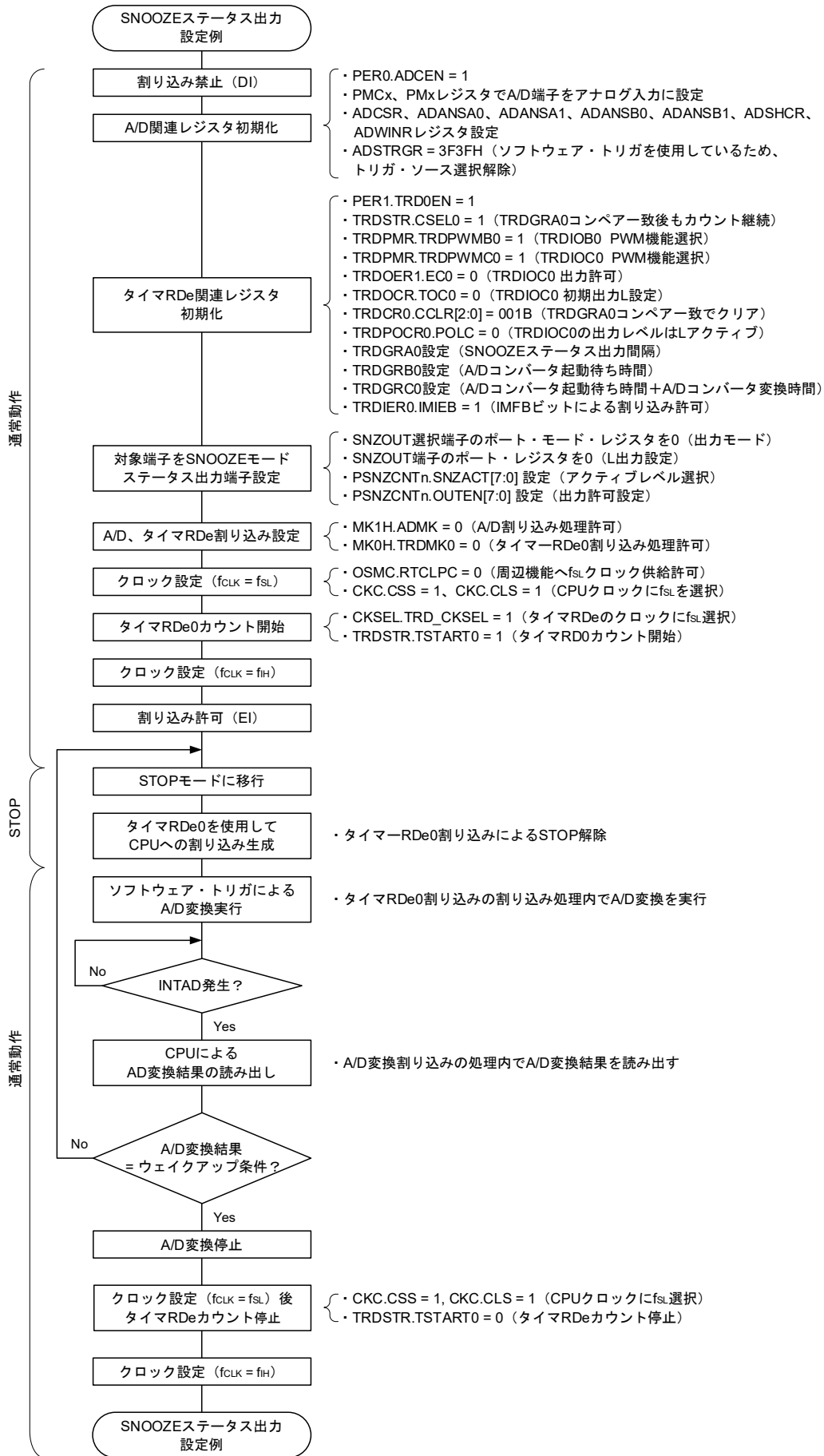
• RS-CANFD lite の割り込み処理

RS-CANFD lite の割り込みフラグ (RFSTSk レジスタの RFIF ビットなど) は、割り込み発生時に自動でクリアされないため、割り込み処理内で対応する割り込みフラグをすべてクリアする必要があります。詳細は「18.12.2 割り込み処理フロー」を参照ください。

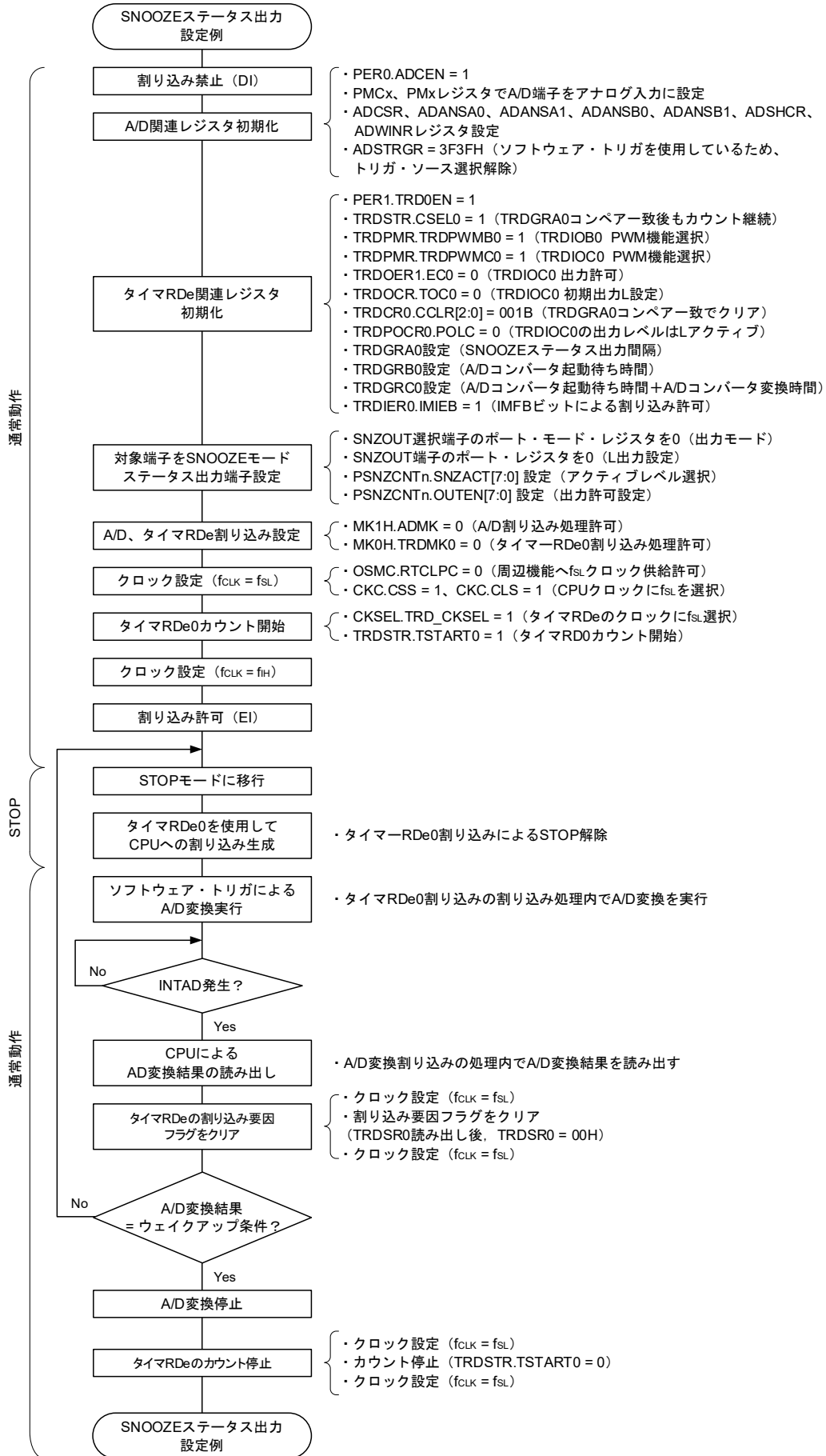
No.4: SNOOZE ステータス出力設定例の誤記訂正

誤)

図 23-10 SNOOZE ステータス出力設定例



正) 割り込み要因フラグのクリア処理を追加



No.5: CFERRCTLR レジスタのビット 2 の説明追加

誤)

ビット 2	使用不可
0	書き込み値は必ず 0 としてください。 <u>読み出し値は 0</u> です。

正)

ビット 2	使用不可
0	書く場合は 0 を書いてください。読んだ場合、動作モードにより読み出し値は異なります。 通常動作モード時：0 が読み出されます。 オンチップ・デバッグ・モード時：不定値が読み出されます。

No.6: FLWE レジスタの説明追加

誤)

注意 2. (FSSQ) の DCLR ビットが設定されていれば、FLWE はフラッシュ・メモリの ECC データとして有効です。

正)

注意 2. (FSSQ) の DCLR ビットが 0 の場合、FLWE レジスタはフラッシュ・メモリの ECC 領域シーケンサにより ECC データを格納します。DCLR ビットが 1 (ECC 領域シーケンサ停止) のときに、FLWE レジスタを診断セルフテストとして使用することができます。

No.7: フラッシュ・メモリの誤記訂正

● 31.2 ユーザ・オプション・バイト (000C0H/040C0H) の注意説明の誤記訂正

誤)

注意 EEPROM エミュレーション時でも、ウォッチドッグ・タイマの動作は継続します。ただし、これらの処置中には割り込みの受け付け時間が遅れるので、遅延を考慮し、オーバフロー時間およびウィンドウ・オープン期間を設定してください。

正)

注意 フラッシュ・メモリの書き換え時でも、ウォッチドッグ・タイマの動作は継続します。ウォッチドッグ・タイマのカウンタ・クリアの遅延を考慮してオーバフロー時間およびウィンドウ・オープン期間を設定してください。

● 32 セルフ・プログラミング説明の誤記訂正

誤)

・セルフ・プログラミング

フラッシュ・セルフ・プログラミング・コードを利用して、ユーザ・アプリケーション上でコード・フラッシュ・メモリの自己書き換えができます。詳細は「32.7 セルフ・プログラミング」を参照してください。

正)

・セルフ・プログラミング

ユーザ・アプリケーション上でコード・フラッシュ・メモリを書き換えることができます。詳細は「32.7 セルフ・プログラミング」を参照してください。

● 32 データ・フラッシュ・メモリ説明の誤記訂正

誤)

データ・フラッシュ・メモリは、データ・フラッシュ・コードを利用して、ユーザ・プログラム実行中に書き換えることができます（バックグランド・オペレーション）。

データ・フラッシュへのアクセスや書き込みについては、「32.9 データ・フラッシュ」を参照してください。

正)

データ・フラッシュ・メモリは、専用フラッシュ・メモリ・プログラマや外部デバイスによるシリアル・プログラミングを使用して書き込むことができます。また、ユーザ・プログラム実行中に書き換えることができます（バックグランド・オペレーション）。

データ・フラッシュへのアクセスや書き込みについては、「32.7 セルフ・プログラミング」および「32.9 データ・フラッシュ」を参照してください。

● 32.7 セルフ・プログラミング説明の誤記訂正

誤)

RL78/F23 および RL78/F24 は、ユーザ・プログラムを介してコード・フラッシュ・メモリを書き換えるためのセルフ・プログラミング機能をサポートしています。この関数を使用すると、ユーザ・アプリケーションでコード・フラッシュ・メモリを書き換えることができるため、フィールドでプログラムを更新することができます。

正)

RL78/F23 および RL78/F24 は、ユーザ・プログラムを介してコード・フラッシュ・メモリを書き換えるためのセルフ・プログラミング機能をサポートしています。この機能を使用すると、ユーザ・アプリケーションでコード・フラッシュ・メモリを書き換えることができるため、フィールドでプログラムを更新することができます。

● 32.7 セルフ・プログラミング説明 注意 2, 4 の誤記訂正

誤)

注意 2. セルフ・プログラミング中に割り込みを禁止するためには、通常動作モード時と同様に、DI 命令により IE フラグがクリア (0) されている状態でフラッシュ・セルフ・プログラミング・コードを実行してください。

注意 4. コード・フラッシュ・メモリまたはエクストラ領域を書き換えるには、セルフ・プログラミング・コードまたは値を RAM に配置します。コード・フラッシュ・メモリ・プログラミング・モードでは、コード・フラッシュ・メモリを参照することはできません。したがって、コード・フラッシュ・メモリ・プログラミング・モードでは、ROM (コード・フラッシュ・メモリ) から実行するユーザ・プログラムとそのデータをあらかじめ RAM にコピーして、RAM からプログラムの実行とデータの参照ができるようにしてください。

正)

注意 2. セルフ・プログラミング中に割り込みを禁止するためには、通常動作モード時と同様に、DI 命令により IE フラグがクリア (0) されている状態でフラッシュ・セルフ・プログラミングを実行してください。

注意 4. コード・フラッシュ・メモリまたはエクストラ領域を書き換えるには、セルフ・プログラミングを行うユーザ・プログラムを RAM に配置します。コード・フラッシュ・メモリ・プログラミング・モードでは、コード・フラッシュ・メモリを参照することはできません。したがって、コード・フラッシュ・メモリ・プログラミング・モードでは、ROM (コード・フラッシュ・メモリ) から実行するユーザ・プログラムとそのデータをあらかじめ RAM にコピーして、RAM からプログラムの実行とデータの参照ができるようにしてください。

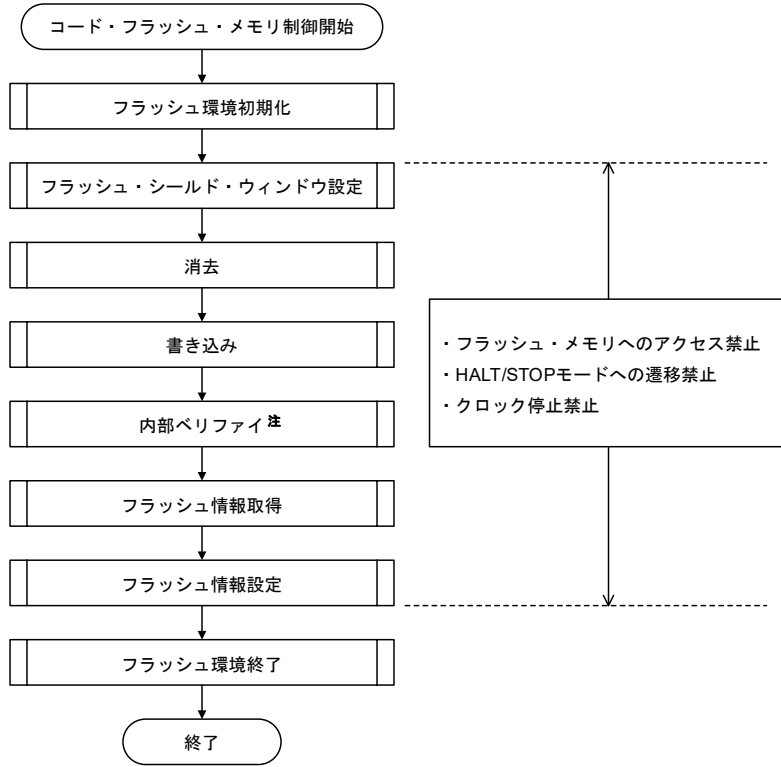
● 32.7.1 セルフ・プログラミング手順の説明および図 32-8 の誤記訂正

誤)

32.7.1 セルフ・プログラミング手順

フラッシュ・セルフ・プログラミング・コードを利用してコード・フラッシュ・メモリの書き換えを行う流れを示します。

図 32-8 セルフ・プログラミング（フラッシュ・メモリの書き換え）のフロー

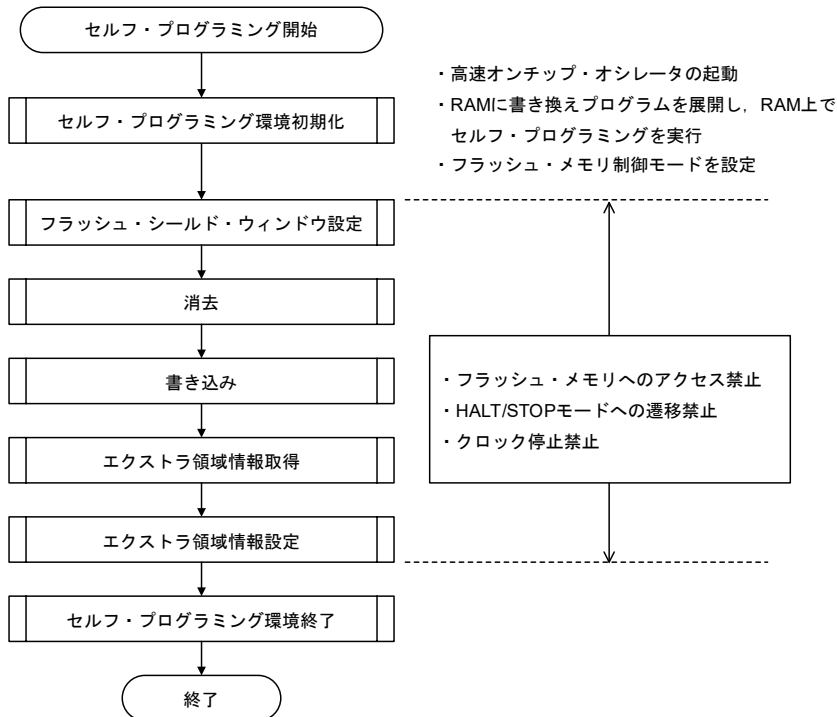


正)

32.7.1 セルフ・プログラミング手順

セルフ・プログラミングを利用してコード・フラッシュ・メモリの書き換えを行う流れを示します。

図 32-8 セルフ・プログラミング（フラッシュ・メモリの書き換え）のフロー



● 32.7.6.4 (2) 書き込み, (3) ブランク・チェックの誤記訂正**誤)**

(2) 書き込み：書き込みは、1 ワード (4 バイト) 単位のため、FLAPL レジスタの下位 2 ビットを 00B に設定します。

FLAPH と FLAPL レジスタ：対象のフラッシュ・メモリの先頭アドレス (例：0x002000)

FLSEDH と FLSEDL レジスタ：すべて“0”，または未設定 (例：0x000000)

FLWH と FLWL レジスタ：FLW7～FLW0 ビットのみが有効であるため、0x00000000～0x000000FF の範囲で書き込む値を設定してください。

(3) ブランク・チェック：ブランク・チェックは、1 ワード (4 バイト) 単位のため、FLAPL レジスタの下位 2 ビットを 00B に設定します。また、FLSEDL レジスタの下位 2 ビットを 11B に設定します。

FLAPH と FLAPL レジスタ：対象のフラッシュ・メモリの先頭アドレス (例：0x002000)

FLSEDH と FLSEDL レジスタ：対象のフラッシュ・メモリの終了アドレス (例：0x0023FF)

注 1 バイトのみブランク・チェックする場合は、FLSEDH と FLSEDL レジスタには、FLAPH と FLAPL レジスタと同じ値を設定します。

正)

(2) 書き込み：書き込みは、1 ワード (4 バイト) 単位のため、FLAPL レジスタの下位 2 ビットを 00B に設定します。

FLAPH と FLAPL レジスタ：対象のフラッシュ・メモリの先頭アドレス (例：0x002000)

FLSEDH と FLSEDL レジスタ：すべて“0”，または未設定 (例：0x000000)

FLWH と FLWL レジスタ：1 ワード (4 バイト) の書き込むデータを設定してください。

(3) ブランク・チェック：ブランク・チェックは、1 ワード (4 バイト) 単位のため、FLAPL レジスタの下位 2 ビットを 00B に設定します。また、FLSEDL レジスタの下位 2 ビットを 11B に設定します。

FLAPH と FLAPL レジスタ：対象のフラッシュ・メモリの先頭アドレス (例：0x002000)

FLSEDH と FLSEDL レジスタ：対象のフラッシュ・メモリの終了アドレス (例：0x0023FF)

注 1 ワード (4 バイト) のみブランク・チェックする場合は、FLSEDH と FLSEDL レジスタには、FLAPH と FLAPL レジスタと同じ値を設定します。

● 32.7.6.5 データ・フラッシュ・メモリ領域の書き換え操作の誤記訂正**誤)**

<コマンドの操作方法>

対象コマンドは、コード・フラッシュ・メモリ領域のブロック消去、内部ベリファイ、書き込み、ブランク・チェックです。

正)

<コマンドの操作方法>

対象コマンドは、データ・フラッシュ・メモリ領域のブロック消去、内部ベリファイ、書き込み、ブランク・チェックです。

● **32.7.7 ブート・スワップ機能の誤記訂正**

誤)

32.7.7 ブート・スワップ機能

セルフ・プログラミングにてブート・プログラム領域であるブート・クラスタ 0 注 の消去を行う前に、あらかじめ新しいブート・プログラムをブート・クラスタ 1 に書き込んでおきます。ブート・クラスタ 1 への書き込みが正常終了したら、RL78/F23, F24 内蔵のファームウェアのセット・インフォメーション機能で、このブート・クラスタ 1 とブート・クラスタ 0 をスワップし、ブート・クラスタ 1 をブート領域にします。このあと、本来の領域であるブート・クラスタ 0 へ消去や書き込みを行います。

正)

32.7.7 ブート・スワップ機能

セルフ・プログラミングにてブート・プログラム領域であるブート・クラスタ 0 注 の消去を行う前に、あらかじめ新しいブート・プログラムをブート・クラスタ 1 に書き込んでおきます。ブート・クラスタ 1 への書き込みが正常終了したら、セルフ・プログラミングでブート領域をブート・クラスタ 0 からブート・クラスタ 1 に変更します。このあと、本来の領域であるブート・クラスタ 0 へ消去や書き込みを行います。

● **32.7.8 フラッシュ・シールド・ウィンドウ機能 表 32-13 の誤記訂正**

誤)

表 32-13 フラッシュ・シールド・ウィンドウ機能の設定／変更方法とコマンドの関係

プログラミング条件	ウィンドウ範囲の設定／変更方法	実行コマンド	
		ブロック消去	書き込み
セルフ・プログラミング時	<u>フラッシュ・セルフ・プログラミング・コード</u> で、ウィンドウの先頭ブロック、最終ブロックを指定する	ウィンドウ範囲内のみブロック消去できる	ウィンドウ範囲内のみ書き込みできる
シリアル・プログラミング時	専用フラッシュ・メモリ・プログラマの GUI 上などで、ウィンドウの先頭ブロック、最終ブロックを指定する	ウィンドウ範囲外もブロック消去可能	ウィンドウ範囲外も書き込み可能

正)

表 32-13 フラッシュ・シールド・ウィンドウ機能の設定／変更方法とコマンドの関係

プログラミング条件	ウィンドウ範囲の設定／変更方法	実行コマンド	
		ブロック消去	書き込み
セルフ・プログラミング時	<u>フラッシュ・セルフ・プログラミング</u> で、ウィンドウの先頭ブロック、最終ブロックを指定する	ウィンドウ範囲内のみブロック消去できる	ウィンドウ範囲内のみ書き込みできる
シリアル・プログラミング時	専用フラッシュ・メモリ・プログラマの GUI 上などで、ウィンドウの先頭ブロック、最終ブロックを指定する	ウィンドウ範囲外もブロック消去可能	ウィンドウ範囲外も書き込み可能

● 32.7.10 セルフ・プログラミングの注意事項の誤記訂正

誤)

- (1) コード・フラッシュ・メモリまたはエクストラ領域の書き換え

コード・フラッシュ・メモリまたはエクストラ領域を書き換えるには、セルフ・プログラミング・コードまたは値を RAM に配置します。

- (3) データ・フラッシュ・メモリとエクストラ領域を操作するための前提条件

データ・フラッシュ・メモリ領域とエクストラ領域を操作する前に、データ・フラッシュ制御レジスタ (DFLCTL) の DFLEN ビットを 1 (データ・フラッシュ・メモリへのアクセスを有効) に設定してください。

- (6) オンチップ・デバッグを使用する場合の注意事項

オンチップ・デバッグを使用する場合は、フラッシュ・メモリ・シーケンサの動作開始から動作終了までの過程で、オンチップ・デバッグにブレーク・ポイントを設定しないでください。

正)

- (1) コード・フラッシュ・メモリまたはエクストラ領域の書き換え

コード・フラッシュ・メモリまたはエクストラ領域を書き換えるには、RAM に書き換えプログラミングを配置します。

- (3) データ・フラッシュ・メモリ領域を操作するための前提条件

データ・フラッシュ・メモリ領域を操作する前に、データ・フラッシュ制御レジスタ (DFLCTL) の DFLEN ビットを 1 (データ・フラッシュ・メモリへのアクセスを有効) に設定してください。

- (6) オンチップ・デバッグを使用する場合の注意事項

オンチップ・デバッグを使用してフラッシュ・セルフ・プログラミング中にブレークする場合は、CS+や e2studio などの統合開発環境のデバッグ設定で、「Flash のセルフ・プログラミングを行う」を [はい] に設定してください。詳しくは、統合開発環境のユーザーズ・マニュアルを参照してください。

● 32.8 セキュリティ設定の誤記訂正

誤)

32.8 セキュリティ設定

RL78/F23, F24 は、フラッシュ・メモリに書かれたユーザ・プログラムの書き換えを禁止するセキュリティ機能をサポートしており、第三者によるプログラム改ざん防止などに対応しています。

「Security Set」コマンドを使用することにより、次の操作をすることができます。

正)

32.8 セキュリティ設定

RL78/F23, F24 は、フラッシュ・メモリに書かれたユーザ・プログラムの書き換えを禁止するセキュリティ機能をサポートしており、第三者によるプログラム改ざん防止などに対応しています。

シリアル・プログラミングまたはセルフ・プログラミングで、次の操作をすることができます。

● 32.9.1 データ・フラッシュの概要の誤記訂正

誤)

- データ・フラッシュ・コードを利用することにより、ユーザ・プログラムでデータ・フラッシュ・メモリの書き換えが可能です。

- 注意 1. リセット解除後、データ・フラッシュは停止状態です。データ・フラッシュ使用時はデータ・フラッシュ・コントロール・レジスタ (DFLCTL) を必ず設定してください。
2. データ・フラッシュの書き換え中は、高速オンチップ・オシレータを動作させておく必要があります。高速オンチップ・オシレータを停止させている場合は、高速オンチップ・オシレータ・クロックを動作 (HIOSTOP = 0) させ、30 μ s 経過後にデータ・フラッシュ・コードを実行してください。

正)

- セルフ・プログラミングにより、ユーザ・プログラムでデータ・フラッシュ・メモリの書き換えが可能です。

- 注意 1. リセット解除後、データ・フラッシュは停止状態です。データ・フラッシュ使用時はデータ・フラッシュ・コントロール・レジスタ (DFLCTL) を必ず設定してください。
2. データ・フラッシュの書き換え中は、高速オンチップ・オシレータを動作させておく必要があります。高速オンチップ・オシレータを停止させている場合は、高速オンチップ・オシレータ・クロックを動作 (HIOSTOP = 0) させ、30 μ s 経過後にデータ・フラッシュ・メモリの書き換えを実行してください。

● 32.9.2 データ・フラッシュのアクセス手順の誤記訂正

誤)

32.9.2 データ・フラッシュへのアクセス手順

リセット解除後、データ・フラッシュは停止状態です。データ・フラッシュへアクセスするには、以下の手順で初期設定を行う必要があります。

初期設定後は、CPU 命令による読み出し、またはデータ・フラッシュ・プログラミング・コードによる読み出し／書き換えが可能です。

- 注意 3. データ・フラッシュの書き換え中は、高速オンチップ・オシレータを動作させておく必要があります。高速オンチップ・オシレータを停止させている場合は、高速オンチップ・オシレータ・クロックを動作 (HIOSTOP = 0) させ、30 μ s 経過後にデータ・フラッシュ・プログラミング・コードを実行してください。

正)

32.9.2 データ・フラッシュへのアクセス手順

リセット解除後、データ・フラッシュは停止状態です。データ・フラッシュへアクセスするには、以下の手順で初期設定を行う必要があります。

初期設定後は、CPU 命令による読み出し、またはセルフ・プログラミングを使用したユーザ・プログラムによる読み出し／書き換えが可能です。

- 注意 3. データ・フラッシュの書き換え中は、高速オンチップ・オシレータを動作させておく必要があります。高速オンチップ・オシレータを停止させている場合は、高速オンチップ・オシレータ・クロックを動作 (HIOSTOP = 0) させ、30 μ s 経過後にデータ・フラッシュ・メモリの書き換えを実行してください。

No.8: フラッシュ・メモリの説明追加**● 32 フラッシュ・メモリ消去時の注意説明を追加**

誤)

フラッシュ・メモリのプログラミング方法は、次のとおりです。

コード・フラッシュ・メモリは、フラッシュ・メモリ・プログラマまたは外部デバイス（UART 通信）によるシリアル・プログラミングもしくは、セルフ・プログラミングで書き換えることができます。

正)

フラッシュ・メモリのプログラミング方法は、次のとおりです。

コード・フラッシュ・メモリは、フラッシュ・メモリ・プログラマまたは外部デバイス（UART 通信）によるシリアル・プログラミングもしくは、セルフ・プログラミングで書き換えることができます。

消去されたフラッシュ・メモリ領域を CPU が読み出すと、コード・フラッシュ ECC エラー検出割り込みが発生します。コード・フラッシュ・メモリ領域のすべての未使用領域には、書き込みを行ってください。未使用領域に書き込むデータは、FFH を推奨します。CPU が FFH コードをフェッチすると、不正命令の実行による内部リセットが発生します。

● 32.7 セルフ・プログラミング説明 注意 5 の説明訂正

誤)

注意 5. 消去されたフラッシュ・メモリ領域を CPU が読み出すと、コード・フラッシュ ECC エラー検出割り込みが発生します。コード・フラッシュ・メモリ領域の書き込まれていない領域には、データ (FFH) を書き込むことをお勧めします。CPU が FFH コードをフェッチすると、不正命令の実行による内部リセットが発生します。

正)

注意 5. セルフ・プログラミング中は、割り込みを禁止 (DI 状態) してください。セルフ・プログラミングを中断して、割り込みを使用したい場合は、非プログラマブル・モードに遷移して、コード・フラッシュ・メモリ ECC 機能の割り込みをマスクした後で、割り込みを許可 (EI 状態) してください。

消去されたフラッシュ・メモリ領域を CPU が読み出すと、コード・フラッシュ ECC エラー検出割り込みが発生します。コード・フラッシュ・メモリ領域のすべての未使用領域には、書き込みを行ってください。

未使用領域に書き込むデータは、FFH を推奨します。CPU が FFH コードをフェッチすると、不正命令の実行による内部リセットが発生します。

● 32.7.10 セルフ・プログラミングの注意事項の説明訂正

誤)

(2) コード・フラッシュ・メモリ領域の消去

消去されたフラッシュ・メモリ領域を CPU が読み出すと、コード・フラッシュ ECC エラー検出割り込みが発生します。コード・フラッシュ・メモリ領域の書き込まれていない領域には、データ (FFH) を書き込むことをお勧めします。CPU が FFH コードをフェッチすると、不正命令の実行による内部リセットが発生します。

正)

(2) コード・フラッシュ・メモリ領域の消去

セルフ・プログラミング中は、割り込みを禁止 (DI 状態) してください。セルフ・プログラミングを中断して、割り込みを使用したい場合は、非プログラマブル・モードに遷移して、コード・フラッシュ・メモリ ECC 機能の割り込みをマスクした後で、割り込みを許可 (EI 状態) してください。

消去されたフラッシュ・メモリ領域を CPU が読み出すと、コード・フラッシュ ECC エラー検出割り込みが発生します。コード・フラッシュ・メモリ領域のすべての未使用領域には、書き込みを行ってください。未使用領域に書き込むデータは、

FFH を推奨します。CPU が FFH コードをフェッチすると、不正命令の実行による内部リセットが発生します。

No.9: 電気的特性の誤記修正

● 36.2.4, 37.2.4, 38.2.4 PLLDIV1 ビットの誤記修正

誤)

発振子	略号	条件	MIN.	TYP.	MAX.	単位	
PLL 出力周波数 (中央値)	f _{PLL}	f _{MAIN} : 8 MHz または 16 MHz, PLLMULA = 1, PLLMUL = 0	PLLDIV0 = 0, FPLLDIV = 0, PLLDIV1 = 0		f _{PLL} × 10/2		MHz
			PLLDIV0 = 0, FPLLDIV = 1, PLLDIV1 = 1		f _{PLL} × 10		MHz

正)

発振子	略号	条件	MIN.	TYP.	MAX.	単位	
PLL 出力周波数 (中央値)	f _{PLL}	f _{MAIN} : 8 MHz または 16 MHz, PLLMULA = 1, PLLMUL = 0	PLLDIV0 = 0, FPLLDIV = 0, PLLDIV1 = 0		f _{PLL} × 10/2		MHz
			PLLDIV0 = 0, FPLLDIV = 1, PLLDIV1 = 1		f _{PLL} × 10		MHz

● 36.5.1, 37.5.1, 38.5.1 (2), (3), (9) の誤記修正

誤)

(2), (3) : 注 5. t_{KCY1} ≥ 4/f_{CLK} も満たす必要があります。

(9) : 注 3. t_{KCY1} ≥ 4/f_{CLK} も満たす必要があります。

正)

(2), (3) : 注 5. t_{KCY1} ≥ 4/f_{MCK} も満たす必要があります。

(9) : 注 3. t_{KCY1} ≥ 4/f_{MCK} も満たす必要があります。

● 36.5.2, 37.5.2, 38.5.2 表内の条件記載の誤記修正

誤)

項目	略号	条件	標準モード		ファースト・モード		ファースト・モード・プラス		単位
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCLA0 クロック周波数	f _{SCL}	ファースト・モード・プラス : 10 MHz ≤ f _{CLK}					0	1000	kHz
		ファースト・モード : 3.5 MHz ≤ f _{CLK}			0	400			kHz
		標準モード : 1 MHz ≤ f _{CLK}	0	100					kHz

正)

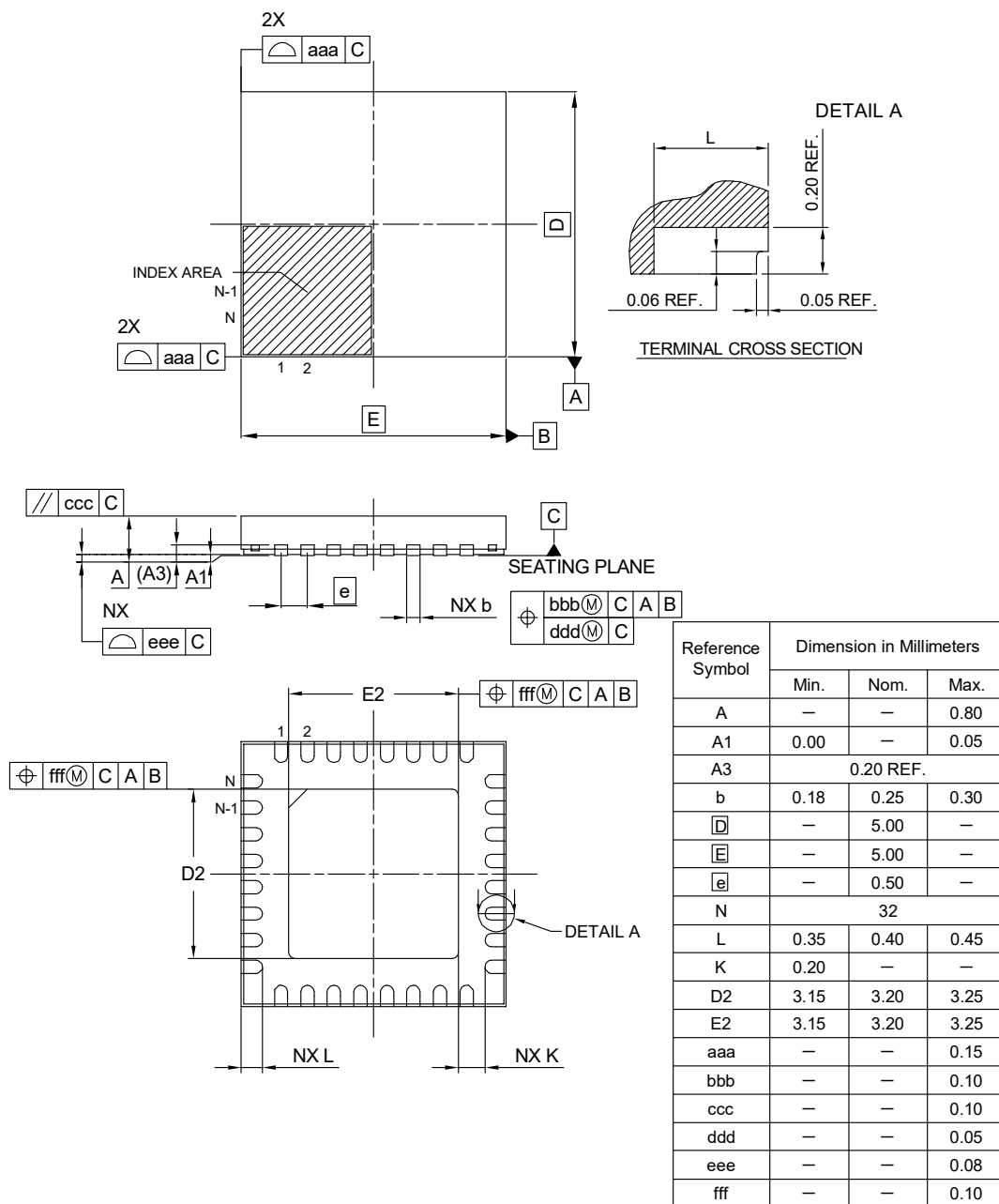
項目	略号	条件	標準モード		ファースト・モード		ファースト・モード・プラス		単位
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCLA0 クロック周波数	f _{SCL}	ファースト・モード・プラス : 10 MHz ≤ f _{MCK}					0	1000	kHz
		ファースト・モード : 3.5 MHz ≤ f _{MCK}			0	400			kHz
		標準モード : 1 MHz ≤ f _{MCK}	0	100					kHz

No.10: 外形図 (32ピン) の記載追加

誤)

39.1 32ピン製品

JEITA Package code	RENESAS code	MASS(TYP.)[g]
P-HWQFN032-5x5-0.50	PWQN0032KF-B	0.06

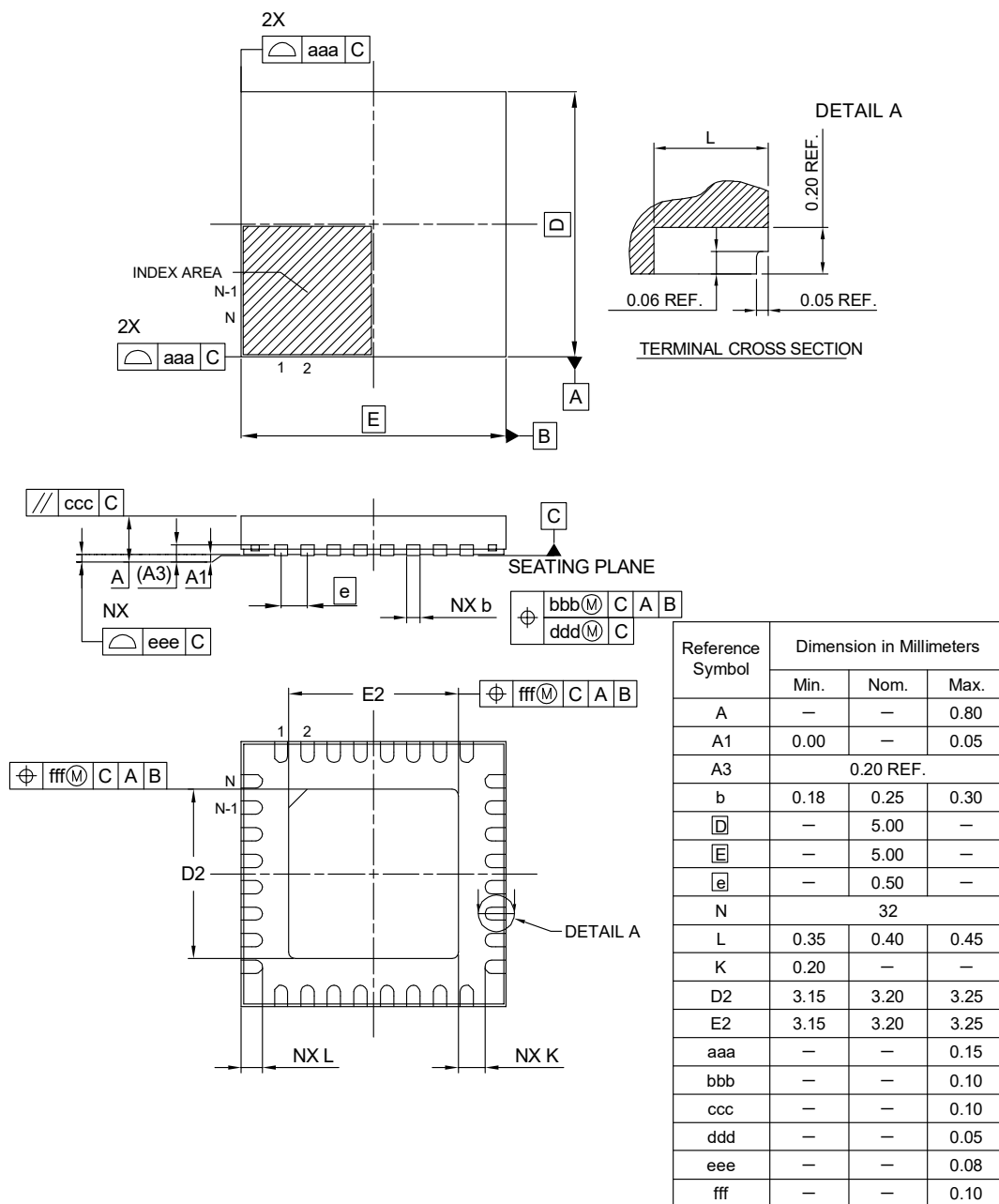


正)

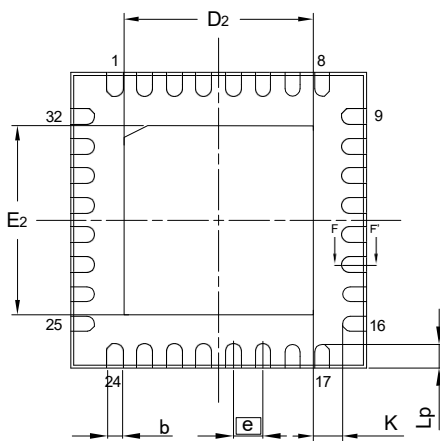
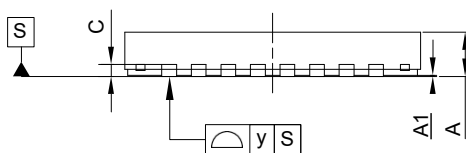
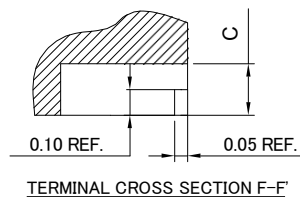
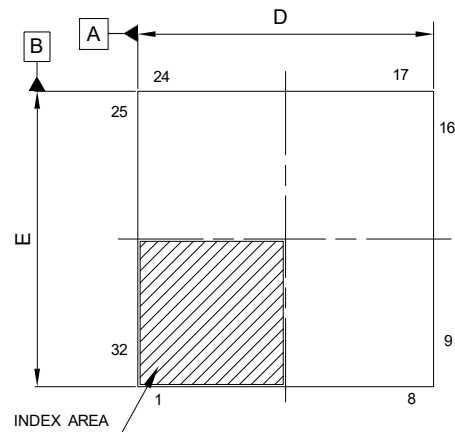
39.1 32ピン製品

外形図は2種類あります。どちらの外形図を使用しているかは、製品捺印で確認できます。
 詳細については、当社 Web サイトの製品パッケージ情報を参照ください。

JEITA Package code	RENESAS code	MASS(TYP.)[g]
P-HWQFN032-5x5-0.50	PWQN0032KF-B	0.06



JEITA Package code	RENESAS code	MASS(TYP.)[g]
P-HWQFN32-5x5-0.50	PWQN0032KH-A	0.06



Reference Symbol	Dimension in Millimeters		
	Min.	Nom.	Max.
D	4.85	5.00	5.15
E	4.85	5.00	5.15
A	—	—	0.80
A ₁	0.00	—	0.05
b	0.18	0.25	0.30
e	0.50 BSC		
L _p	0.35	0.40	0.45
y	—	—	0.08
c	—	0.20	—
K	0.20	—	—
D ₂	—	3.20	—
E ₂	—	3.20	—