

## RL78/G23

### ELCL エッジ検出の間引き

---

#### 要旨

本アプリケーションノートでは、ロジック&イベント・リンク・コントローラ（ELCL）を使った入力信号のエッジ検出の間引き方法を説明します。ELCL を使うことで、ソフトウェアで実現していた機能をハードウェアで実現できるため、資源（ROM や RAM 等）を削減することが可能です。

#### 動作確認デバイス

RL78/G23

本アプリケーションノートを他のマイコンへ適用する場合、そのマイコンの仕様に合わせて変更し、十分評価してください。

## 目次

1. 仕様	3
2. 動作確認条件	5
3. ハードウェア説明	6
3.1 ハードウェア構成例	6
3.2 使用端子一覧	6
4. ソフトウェア説明	7
4.1 動作概要	7
4.2 フォルダ構成	9
4.3 オプション・バイトの設定一覧	10
4.4 定数一覧	10
4.5 変数一覧	10
4.6 関数一覧	11
4.7 関数仕様	11
4.8 フローチャート	13
4.8.1 メイン処理	13
4.8.2 ELCL 割り込み処理	14
4.8.3 ELCL のフリップフロップリセット処理	14
4.8.4 ウェイト処理	15
4.8.5 TAU0 チャンネル7 割り込み処理	15
5. 応用例	16
5.1 ELCL のコンポーネントの設定	16
5.2 r01an5613_elcl_edge.scfg	18
5.2.1 クロック	19
5.2.2 システム	19
5.2.3 r_bsp	19
5.2.4 Config_LVD0	19
5.2.5 Config_TAU0_7	19
5.2.6 Config_EdgeDetectionThinningFunction	19
5.2.7 Config_PORT	20
5.3 コンポーネント「ELCL エッジ検出の間引き機能」	20
5.3.1 ELCL レジスタの設定	21
6. サンプルコード	26
7. 参考ドキュメント	26
改訂記録	27

## 1. 仕様

本アプリケーションノートでは、ELCL を使って入力信号のエッジ検出の間引き、入力信号が 2 回に 1 回有効になる例を説明します。

図 1-1 に ELCL を使ったエッジ検出の間引きのシステム構成を示します。外部入力信号を INPUT A、間引く間隔を作るクロックを CLOCK B とし、間引きされた出力信号を OUTPUT C とします。

INPUT A からの入力信号に対するエッジ検出を 1 回目はマスクし、2 回目は有効にします。

図 1-1 システム構成

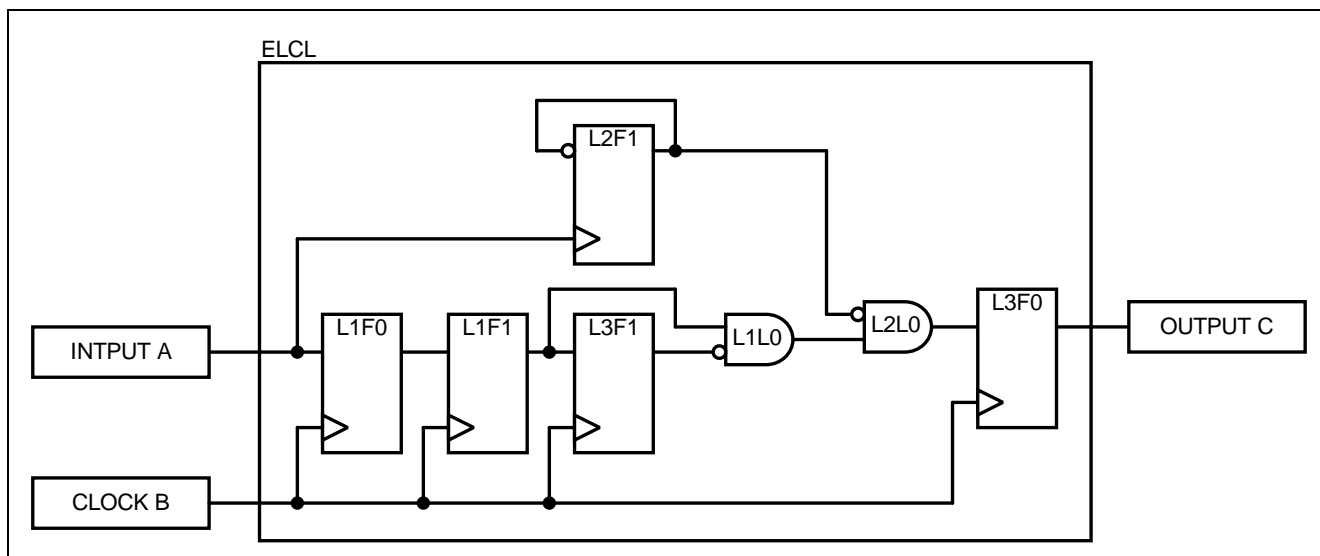
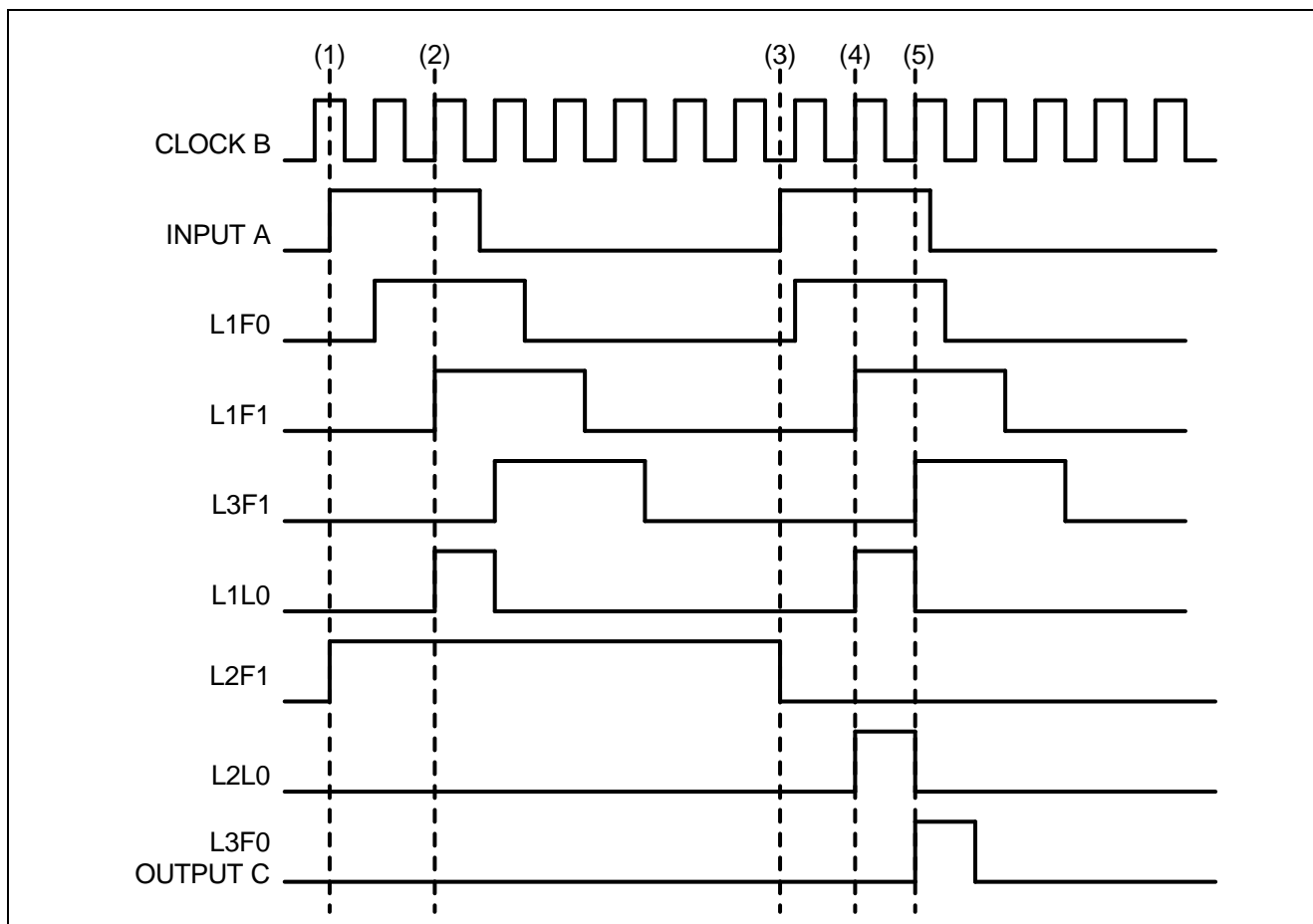


図 1-2 に図 1-1 のタイミングチャートを示します。

- (1) INPUT A からパルス信号が入力される (1 回目)
- (2) 立ち上がりエッジを検出する
- (3) INPUT A からパルス信号が入力される (2 回目)
- (4) 立ち上がりエッジを検出する
- (5) OUTPUT C からパルス信号が出力される

図 1-2 エッジ検出の間引きのタイミングチャート



## 2. 動作確認条件

本アプリケーションノートのサンプルコードは、下記の条件で動作を確認しています。

表 2-1 動作確認条件

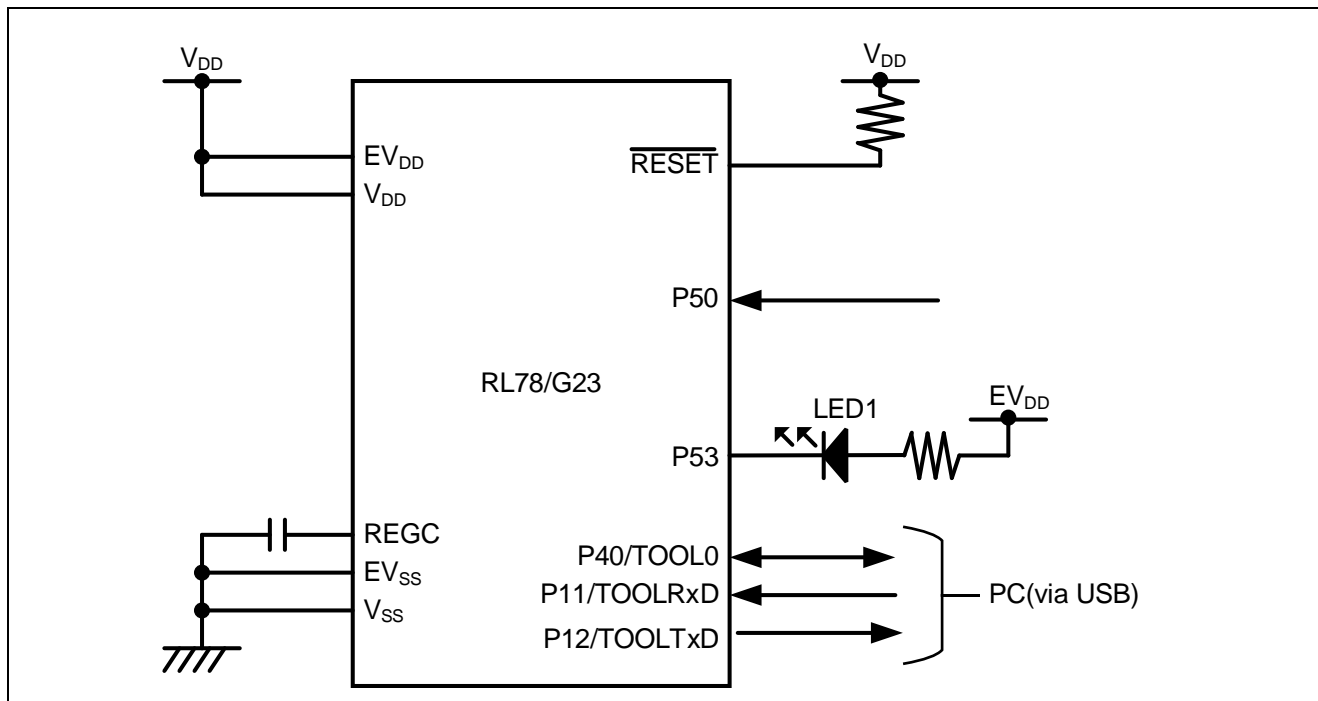
項目	内容
使用マイコン	RL78/G23 (R7F100GLG)
動作周波数	<ul style="list-style-type: none"> <li>高速オンチップ・オシレータ・クロック : 16MHz</li> <li>CPU/周辺ハードウェア・クロック : 16MHz</li> </ul>
動作電圧	<ul style="list-style-type: none"> <li>3.3V</li> <li>LVD0 動作 (<math>V_{LVD0}</math>) : リセット・モード 立ち上がり時 TYP. 1.90V 立ち下がり時 TYP. 1.86V</li> </ul>
統合開発環境 (CS+)	ルネサスエレクトロニクス製 CS+ for CC V8.07.00
C コンパイラ (CS+)	ルネサスエレクトロニクス製 CC-RL V1.11
統合開発環境 (e <sup>2</sup> studio)	ルネサスエレクトロニクス製 e <sup>2</sup> studio 2022-01 (22.01.0)
C コンパイラ (e <sup>2</sup> studio)	ルネサスエレクトロニクス製 CC-RL V1.11
統合開発環境 (IAR)	IAR システム製
C コンパイラ (IAR)	IAR Embedded Workbench for Renesas RL78 V4.21.1
スマート・コンフィグレータ	V.1.2.0
ボードサポートパッケージ (r_bsp)	V.1.13
エミュレータ	CS+、e <sup>2</sup> studio : COM ポート IAR : E2 エミュレータ Lite
使用ボード	RL78/G23 Fast Prototyping Board (RTK7RLG230CLG000BJ)

### 3. ハードウェア説明

#### 3.1 ハードウェア構成例

図 3-1 に本アプリケーションのサンプルコードで使用するハードウェア構成例を示します。

図 3-1 ハードウェア構成例



- 注意 1. この回路イメージは接続の概要を示す為に簡略化しています。実際に回路を作成される場合は、端子処理などを適切に行い、電気的特性を満たすように設計してください（入力専用ポートは個別に抵抗を介して  $V_{DD}$  又は  $V_{SS}$  に接続して下さい）。
- 注意 2.  $EV_{SS}$  で始まる名前の端子がある場合には  $V_{SS}$  に、 $EV_{DD}$  で始まる名前の端子がある場合には  $V_{DD}$  にそれぞれ接続してください。
- 注意 3.  $V_{DD}$  は LVD0 にて設定したリセット解除電圧 ( $V_{LVD0}$ ) 以上にしてください。

#### 3.2 使用端子一覧

表 3-1 に使用端子と機能を示します。

表 3-1 使用端子と機能

端子名	入出力	内容
P53	出力	LED1 点灯 (Low Active)
P50	入力	入力端子

注意 本アプリケーションノートは、使用端子のみを端子処理しています。実際に回路を作成される場合は、端子処理などを適切に行い、電気的特性を満たすように設計してください。

## 4. ソフトウェア説明

### 4.1 動作概要

本サンプルコードでは、P50 端子への入力信号のエッジ検出を行い、2 回に 1 回のエッジを有効にします。

図 4-1 にサンプルコードのシステム構成、図 4-2 にタイミングチャートを示します。

ELCL の入力信号として P50 を ELCL の出力信号として INTELCL を選択します。

図 4-1 サンプルコードのシステム構成

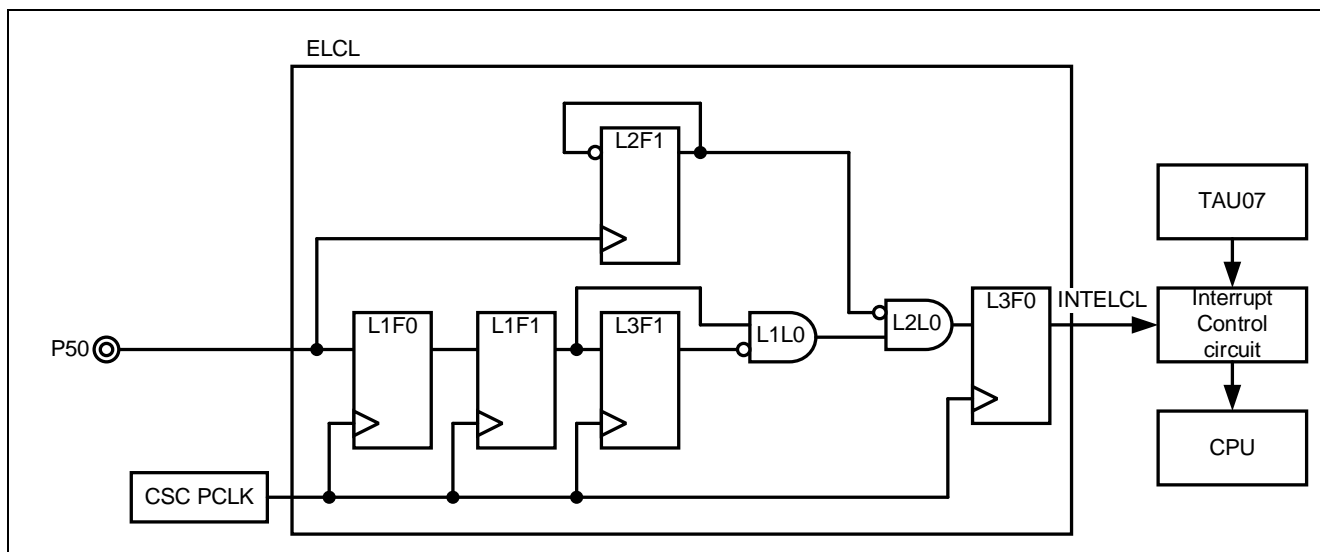
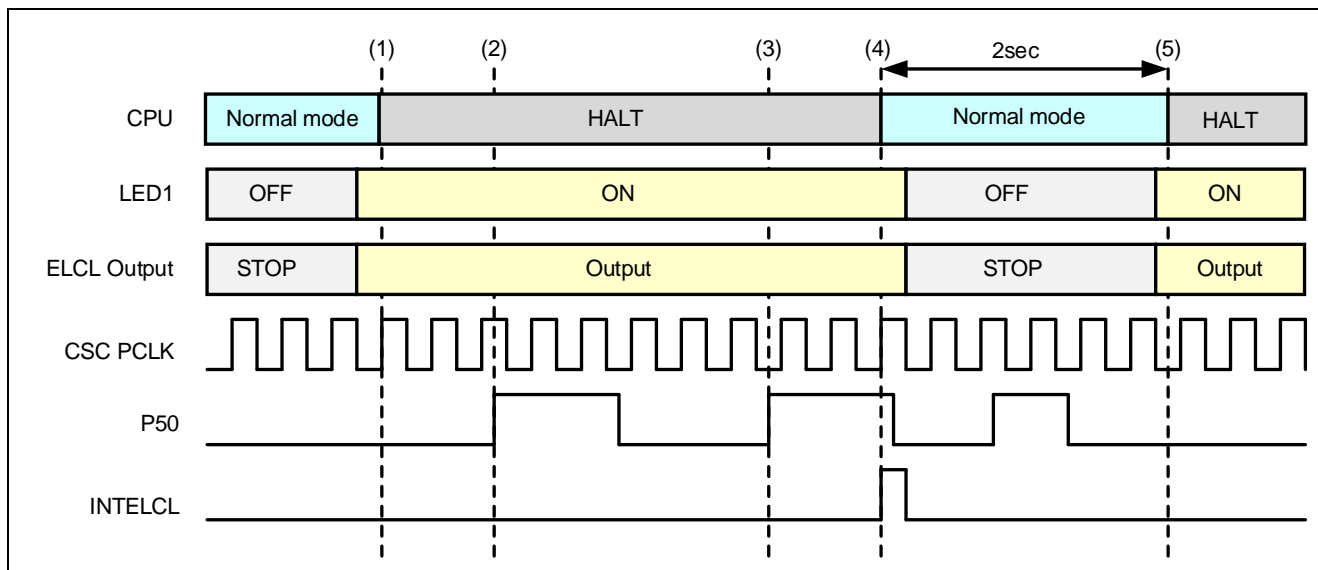


図 4-2 サンプルコードのタイミングチャート



- (1) LED1 を点灯および ELCL の出力信号を出力許可にし、HALT モードに遷移します
- (2) P50 にパルス信号が入力されエッジを検出する (1 回目)。INTELCL を出力しないので HALT モードは解除されません
- (3) P50 にパルス信号が入力されエッジを検出する (2 回目)
- (4) INTELCL を出力し、HALT モードは解除します。LED1 を消灯および ELCL の出力信号を出力禁止にします
- (5) 約 2 秒経過した後、ELCL の設定を初期化し、(1) に戻ります

(4) ~ (5) の期間に P50 にパルス信号が入力されても ELCL の出力は停止しているため、INTELCL は発生しません。



## 4.2 フォルダ構成

表 4-1 にサンプルコードの使用しているソースファイル／ヘッダファイルの構成を示します。なお、統合開発環境で自動生成されるファイル、bsp 環境のファイルは除きます。

表 4-1 フォルダ構成

フォルダ、ファイル名	説明	スマート・コンフィグレータを使用
¥r01an5613_elcl_edge<DIR> <sup>注3</sup>	サンプルコードのフォルダ	
¥src<DIR>	プログラム格納用フォルダ	
main.c	サンプルコードソースファイル	
¥smc_gen<DIR>	スマート・コンフィグレータ生成フォルダ	√
¥Config_EdgeDetectionThinningFunction<DIR>	ELCL 用プログラム格納フォルダ	√
Config_EdgeDetectionThinningFunction.c	ELCL 用ソースファイル	√
Config_EdgeDetectionThinningFunction.h	ELCL 用ヘッダファイル	√
Config_EdgeDetectionThinningFunction_user.c	ELCL 用割り込みソースファイル	√ <sup>注2</sup>
¥Config_PORT<DIR>	ポート用プログラム格納フォルダ	√
Config_PORT.c	ポート用ソースファイル	√
Config_PORT.h	ポート用ヘッダファイル	√
Config_PORT_user.c	ポート用割り込みソースファイル	√ <sup>注1</sup>
¥Config_TAU0_7<DIR>	TAU0 チャンネル 7 用プログラム格納フォルダ	√
Config_TAU0_7.c	TAU0 チャンネル 7 用ソースファイル	√
Config_TAU0_7.h	TAU0 チャンネル 7 用ヘッダファイル	√
Config_TAU0_7_user.c	TAU0 チャンネル 7 用割り込みソースファイル	√ <sup>注2</sup>
¥general<DIR>	初期化、共通プログラム格納フォルダ	√
¥r_bsp<DIR>	BSP 用プログラム格納フォルダ	√
¥r_config<DIR>	プログラム格納フォルダ	√

補足 ” <DIR> ” は、ディレクトリを意味します。

注 1. 本サンプルコードでは使用しません。

注 2. スマート・コンフィグレータで生成したファイルに割り込み処理ルーチンを追加しています。

注 3. IAR 版のサンプルコードは r01an5613\_elcl\_edge.ipcf を格納しています。ipcf ファイルについては、「RL78 スマート・コンフィグレータ ユーザーガイド : IAR 編 (R20AN0581)」を確認してください。

### 4.3 オプション・バイトの設定一覧

表 4-2 にオプション・バイト設定を示します。

表 4-2 オプション・バイト設定

アドレス	設定値	内容
000C0H/040C0H	1110 1111B (EFH)	ウォッチドッグ・タイマ動作停止 (リセット解除後、カウント停止)
000C1H/040C1H	1111 1110B (FEH)	LVD0 リセット・モード 検出電圧：立ち上がり 1.90V／立下り 1.86V
000C2H/040C2H	1110 1001B (E9H)	フラッシュ動作モード：高速メインモード 高速オンチップ・オシレータの周波数：16MHz
000C3H/040C3H	1000 0101B (85H)	オンチップ・デバッグ動作許可

### 4.4 定数一覧

表 4-3 にサンプルコードで使用する定数を示します。

表 4-3 サンプルコードで使用する定数

定数名	設定値	内容	ファイル
LED1	P5_bit.no3	P53	main.c
LED_ON	0	LED を ON するための設定値	main.c
LED_OFF	1	LED を OFF するための設定値	main.c
WAIT_TIME	2000	HALT モード解除後再度 HALT モードに遷移する間隔（2 秒毎）	main.c

### 4.5 変数一覧

表 4-4 にサンプルコードで使用するグローバル変数を示します。

表 4-4 サンプルコードで使用するグローバル変数

型	変数名	内容	使用関数
volatile uint16_t	g_ms_timer	ウエイト処理のカウント値	r_ms_delay, r_Config_TAU0_7_interrupt
uint8_t	g_elcl_interrupt	ELCL 割り込みの発生フラグ	main, r_Config_EdgeDetectionThinningFunction_interrupt

## 4.6 関数一覧

表 4-5 にサンプルコードで使用する関数を示します。ただし、スマート・コンフィグレータで生成された関数の内、変更を行っていないものは除きます。

表 4-5 関数一覧

関数名	概要	ソースファイル
main	メイン処理	main.c
r_Config_EdgeDetectionThinningFunction_interrupt	ELCL 割り込み処理 (HALT モード解除用)	Config_EdgeDetectionThinningFunction_user.c
r_elcl_reset_flipflop	ELCL のフリップフロップのリセット処理	Config_EdgeDetectionThinningFunction_user.c
r_ms_delay	HALT モード解除後のウエイト処理 (HALT モード間隔用)	Config_TAU0_7_user.c
r_Config_TAU0_7_interrupt	TAU0 チャンネル 7 割り込み処理 (HALT モード間隔用)	Config_TAU0_7_user.c

## 4.7 関数仕様

サンプルコードの関数仕様を示します。

### [関数名] main

概要	メイン処理
ヘッダ	r_smc_entry.h
宣言	void main (void);
説明	ELCL の初期化、ELCL の出力設定および割り込みの設定を行います LED1 を点灯し、HALT モードに移行します P50 に 2 回パルスが入力されると HALT モードが解除され、LED1 が消灯してフリップフロップを初期化します WAIT_TIME で設定した時間（約 2 秒）が経過後、再度 HALT モードに移行します
引数	なし
リターン値	なし
備考	なし

### [関数名] r\_Config\_EdgeDetectionThinningFunction\_interrupt

概要	ELCL 割り込み処理
ヘッダ	platform.h
宣言	#pragma interrupt r_Config_EdgeDetectionThinningFunction_interrupt (vect=INTELCL)
説明	g_elcl_interrupt = 1 に設定します
引数	なし
リターン値	なし
備考	なし

## [関数名] r\_elcl\_reset\_flipflop

---

概要	ELCL のフリップフロップのリセット処理
ヘッダ	platform.h
宣言	void r_elcl_reset_flipflop (void);
説明	ELCL 内のフリップフロップを初期化します
引数	なし
リターン値	なし
備考	ELLnCTL のビット 6 とビット 7 に 0 を設定するとフリップフロップはリセットされます

## [関数名] r\_ms\_delay

---

概要	ウエイト処理
ヘッダ	r_cg_macrodriver.h, r_cg_userdefine.h, Config_TAU0_7.h
宣言	void r_ms_delay (uint16_t msec);
説明	引数 msec で指定した時間 (ms) ウエイトします TAU0 チャンネル 7 を使ってカウントします。g_ms_timer が WAIT_TIME 未満の場合はポーリングし、WAIT_TIME 以上の場合はウエイト処理を完了します
引数	Msec
リターン値	なし
備考	なし

## [関数名] r\_Config\_TAU0\_7\_interrupt

---

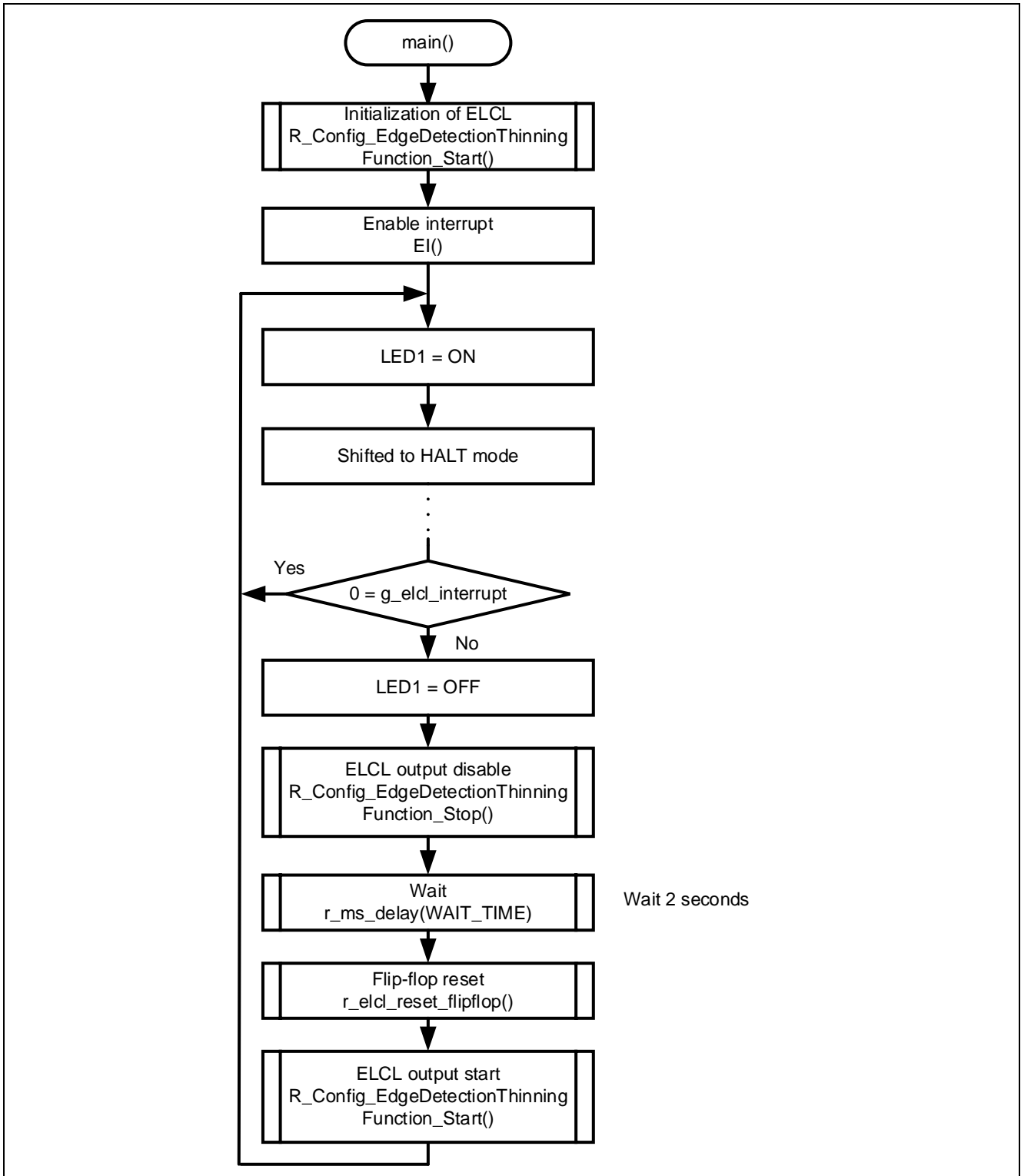
概要	TAU0 チャンネル 7 割り込み処理
ヘッダ	r_cg_macrodriver.h, r_cg_userdefine.h, Config_TAU0_7.h
宣言	#pragma interrupt r_Config_TAU0_7_interrupt (vect=INTTM07)
説明	TAU0 チャンネル 7 の INTTM07 による割り込み処理です。 HALT モード解除後、再度 HALT モードに遷移するまでの時間をカウントします。
引数	なし
リターン値	なし
備考	なし

4.8 フローチャート

4.8.1 メイン処理

図 4-3 にメイン処理のフローチャートを示します。

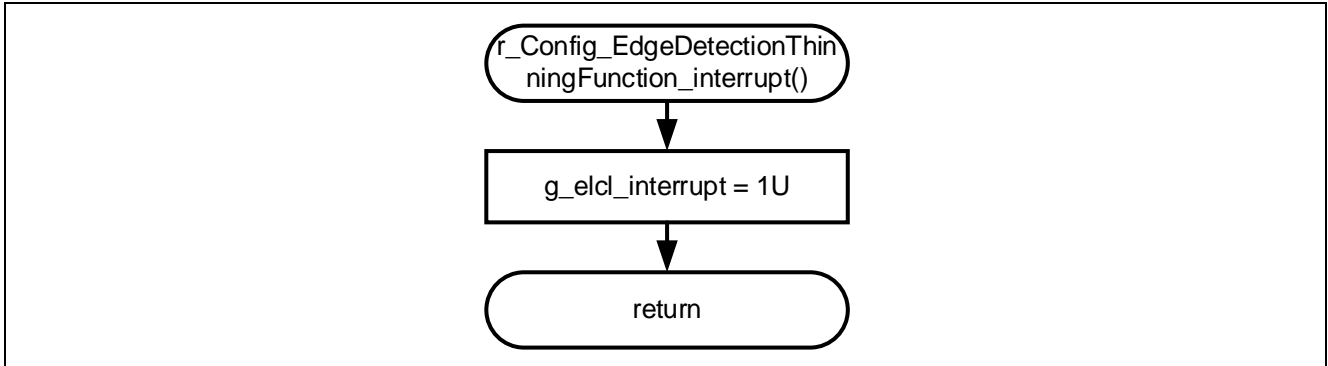
図 4-3 メイン処理



4.8.2 ELCL 割り込み処理

図 4-4 に ELCL 割り込み処理のフローチャートを示します。

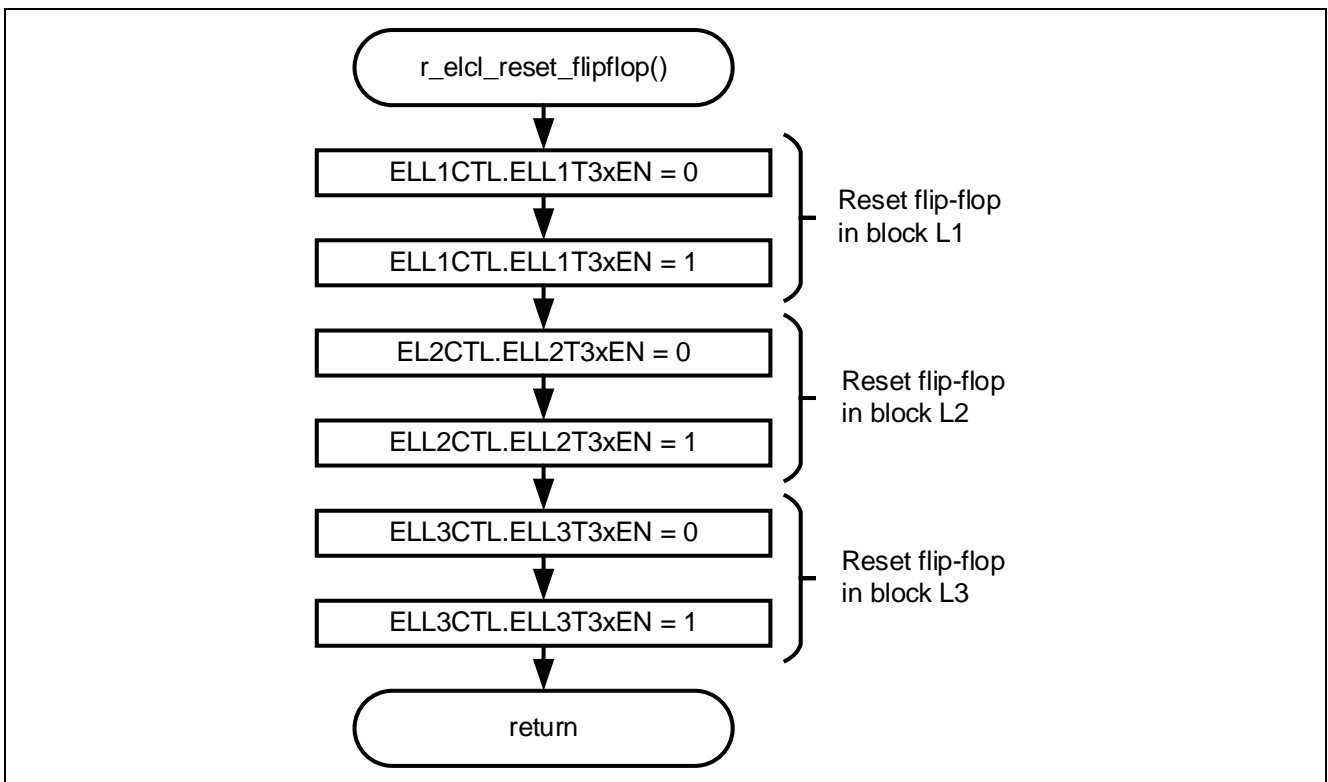
図 4-4 ELCL 割り込み処理



4.8.3 ELCL のフリップフロップリセット処理

図 4-5 に ELCL のフリップフロップリセット処理のフローチャートを示します。

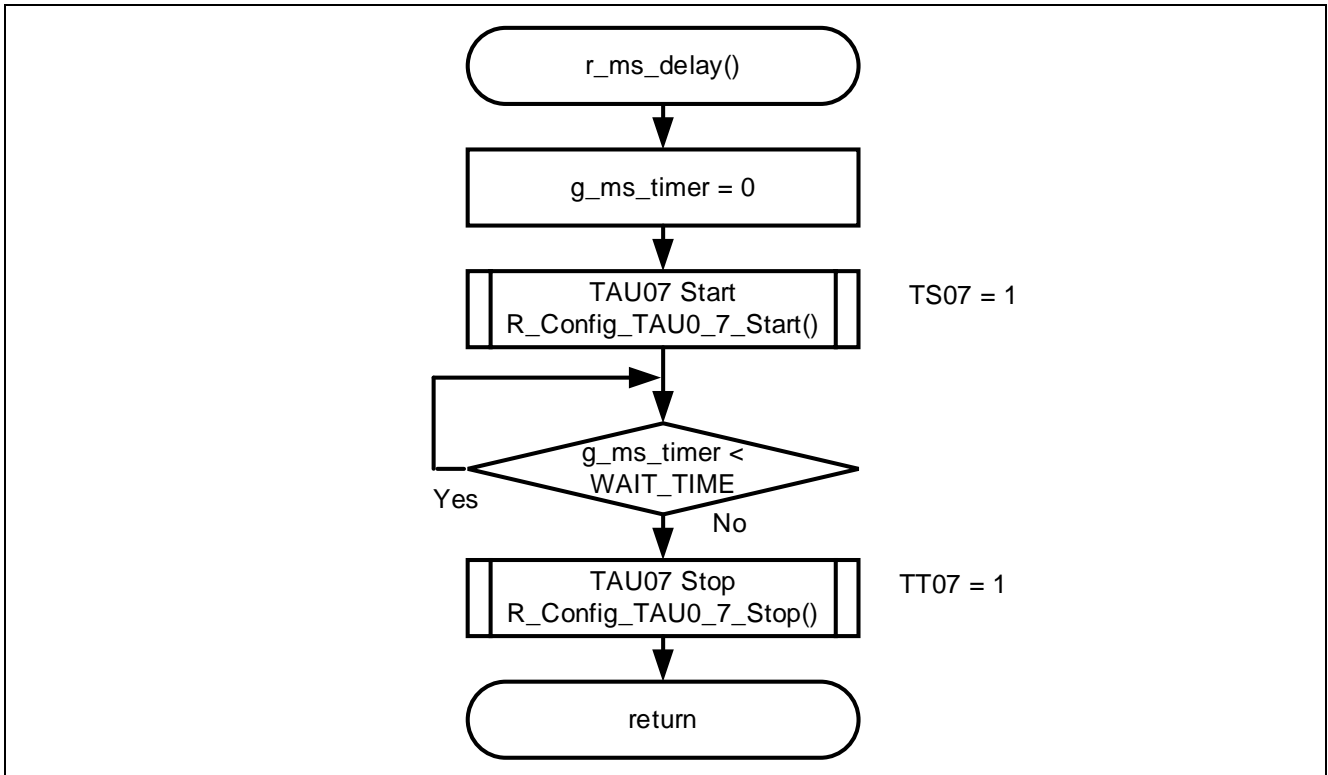
図 4-5 ELCL のフリップフロップリセット処理



4.8.4 ウェイト処理

図 4-6 にウェイト処理のフローチャートを示します。

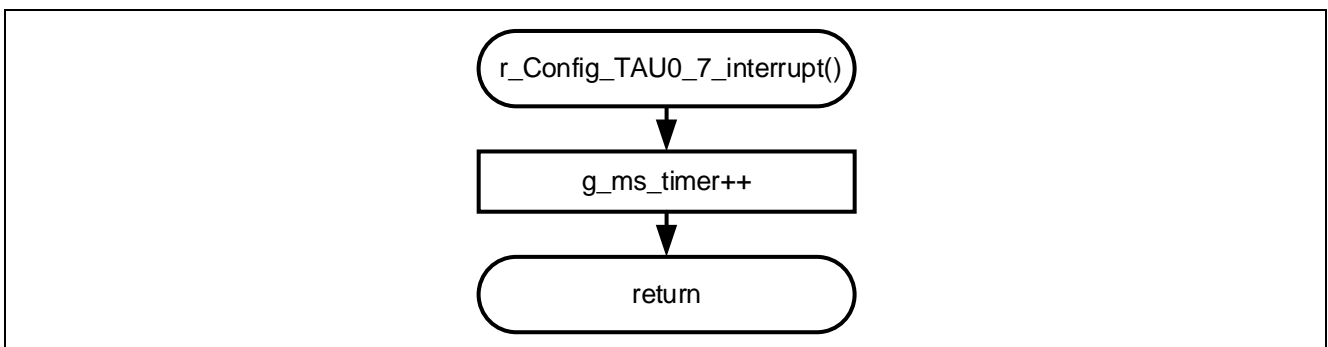
図 4-6 ウェイト処理



4.8.5 TAU0 チャンネル 7 割り込み処理

図 4-7 に TAU0 チャンネル 7 割り込み処理のフローチャートを示します。

図 4-7 TAU0 チャンネル 7 割り込み処理



## 5. 応用例

本アプリケーションノートは、サンプルコードの他に以下のスマート・コンフィグレータの設定ファイルを格納しています。

r01an5613\_elcl\_edge.scfg

ファイルの説明と使用する上での設定例および注意事項を以下に示します。

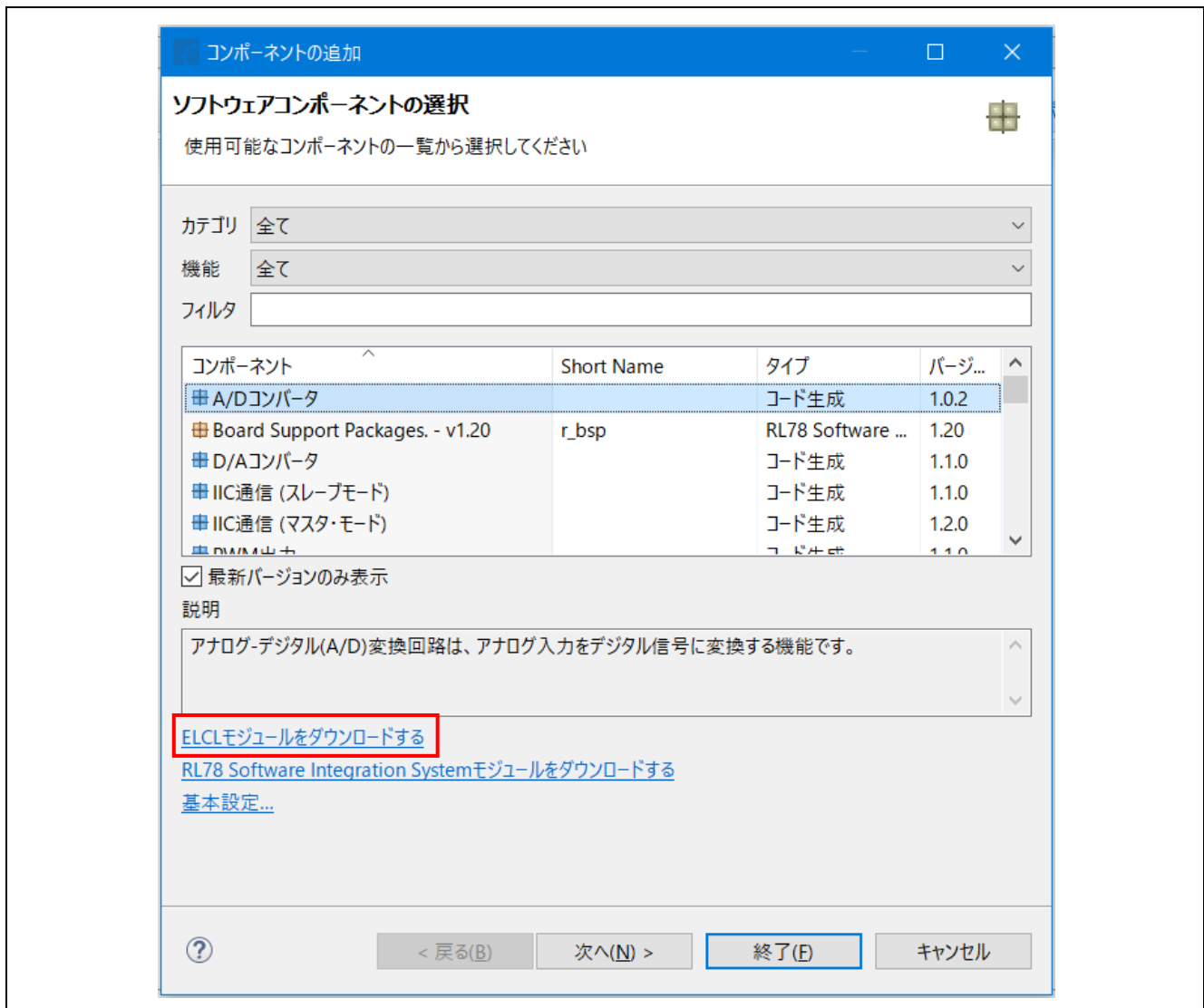
### 5.1 ELCL のコンポーネントの設定

ELCL コンポーネントを使用するためには ELCL コンテンツファイルのインストールが必要です。

手順を以下に示します。

1. スマート・コンフィグレータを起動してください。
2. 「コンポーネント」タグをクリックし、「コンポーネントの追加」をクリックしてください。
3. 図 5-1 に示す「コンポーネントの追加」のウィンドウが開きますので、「ELCL モジュールをダウンロードする」をクリックしてください。

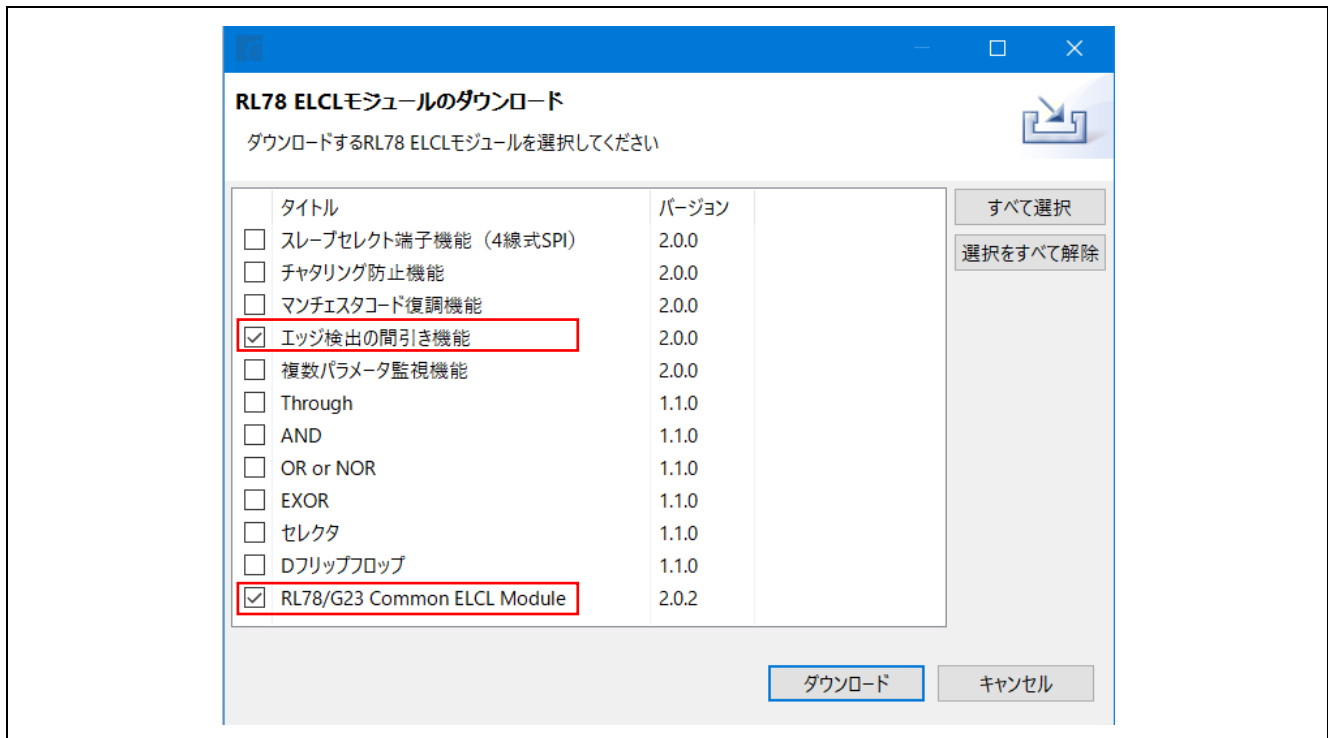
図 5-1 コンポーネントの追加





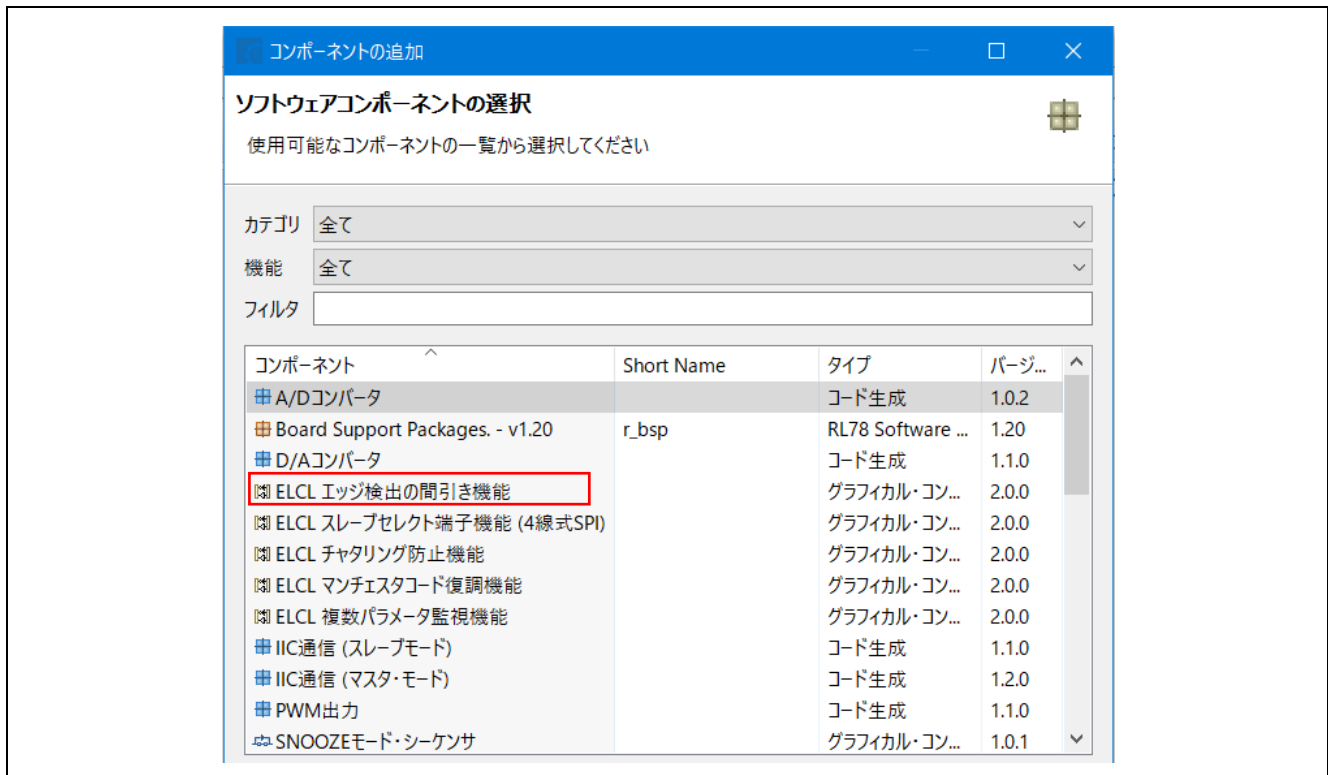
4. 「エッジ検出の間引き機能」を選択し、ダウンロードしてください。共通設定ファイル「RL78/G23 Common ELCL Module」もダウンロードしてください。

図 5-2 モジュールのダウンロード



5. ダウンロード完了後、「ELCL エッジ検出の間引き機能」が選択できることを確認してください。

図 5-3 モジュールの選択



## 5.2 r01an5613\_elcl\_edge.scfg

サンプルコードで使用しているスマート・コンフィグレータの設定ファイルです。スマート・コンフィグレータで設定されている全ての機能が含まれています。サンプルコードの設定は以下の通りです。

表 5-1 スマート・コンフィグレータの設定値

タグ名	コンポーネント	内容
クロック	-	動作モード：高速メインモード 2.4 (V)~5.5 (V) EV <sub>DD</sub> 設定：1.8V ≤ EV <sub>DD0</sub> < 5.5V 高速オンチップ・オシレータ：16MHz f <sub>IHP</sub> ：16MHz f <sub>CLK</sub> ：16MHz（高速オンチップ・オシレータ） f <sub>SXP</sub> ：32.768kHz（低速オンチップ・オシレータ）
システム	-	オンチップ・デバッグ動作設定：COM ポート <sup>注</sup> 疑似 RRM/DMM 機能設定：使用する Start/Stop 関数機能設定：使用しない トレース機能設定：使用する セキュリティ ID 設定：セキュリティ ID を設定する セキュリティ ID：0x00000000000000000000 セキュリティ ID 認証失敗時の設定：フラッシュ・メモリのデータを消去しない
コンポーネント	r_bsp	Start up select：Enable (use BSP startup) Control of invalid memory access detection：Disable RAM guard space (GRAM0-1)：Disabled Guard of control registers of port function (GPORT)：Disabled Guard of registers of interrupt function (GINT)：Disabled Guard of control registers of clock control function, voltage detector, and RAM parity error detection function (GCSC)：Disabled Data flash access control (DFLEN)：Disables Initialization of peripheral functions by Code Generator/Smart Configurator：Enable API functions disable：Enable Parameter check enable：Enable Setting for starting the high-speed on-chip oscillator at the times of release from STOP mode and of transitions to SNOOZE mode：High-speed Enable user warm start callback (PRE)：Unused Enable user warm start callback (POST)：Unused Watchdog Timer refresh enable：Unused
	Config_LVD0	動作モード設定：リセット・モード 電圧検出設定：リセット発生電圧 (V <sub>LVD0</sub> )：1.86 (V)
	Config_TAU0_7	コンポーネント：インターバル・タイマ 動作モード：16 ビット・カウンタ・モード リソース：TAU0_7 動作クロック：CK00 クロックソース：f <sub>CLK</sub> インターバル時間：1ms 割り込み設定：使用する 優先順位：レベル 3

表 5-2 スマート・コンフィグレータの設定値

タグ名	コンポーネント	内容
コンポーネント	Config_EdgeDetectionThinningFunction	コンポーネント : ELCL エッジ検出の間引き機能 Input signal selector : P50 Input signal selector : CSC PCLK Output signal selector : INTELCL
	Config_PORT	コンポーネント : ポート ポート選択 : PORT5 P53 : 出力 (1 を出力)

注. IAR 使用時は以下の設定にしてください。

オンチップ・デバッグ動作設定 : エミュレータを使う

エミュレータ設定 : E2 エミュレータ Lite

### 5.2.1 クロック

サンプルコードで使用するクロックの設定を行います。

### 5.2.2 システム

サンプルコードのオンチップ・デバッグ設定を行います。

「オンチップ・デバッグ動作設定」、「セキュリティ ID 認証失敗時の設定」は、「表 4-2 オプション・バイト設定」の「オンチップ・デバッグ動作許可」に影響を与えます。設定を変更する際は注意してください。

### 5.2.3 r\_bsp

サンプルコードのスタートアップの設定を行います。

### 5.2.4 Config\_LVDO

サンプルコードの電源管理の設定を行います。

「表 4-2 オプション・バイト設定」の「LVDO の設定」に影響を与えます。設定を変更する際は注意してください。

### 5.2.5 Config\_TAU0\_7

サンプルコードの TAU07 の設定を行います。

サンプルコードでは、HALT モード解除後、再度 HALT モードに遷移するまでの時間のカウンタに使用します。

### 5.2.6 Config\_EdgeDetectionThinningFunction

サンプルコードの ELCL の初期設定と出力を行います。

サンプルコードではエッジ検出を行う端子に P50、出力先に INTELCL を使用します。

詳細は、「5.3 コンポーネント「ELCL エッジ検出の間引き機能」」を参照してください。

## 5.2.7 Config\_PORT

サンプルコードのポートの設定を行います。

サンプルコードでは LED1 の制御に P53 を使用します。

## 5.3 コンポーネント「ELCL エッジ検出の間引き機能」

図 5-4 にコンポーネント「ELCL エッジ検出の間引き機能」、

表 5-3 に本コンポーネントの選択肢を示します。

図 5-4 コンポーネント「ELCL エッジ検出の間引き機能」

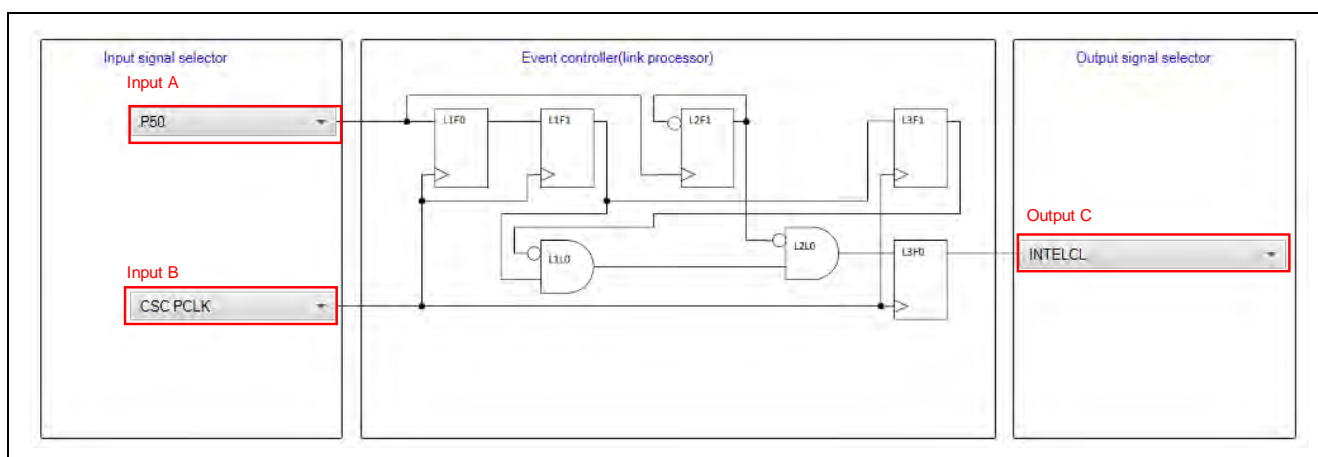


表 5-3 コンポーネント「ELCL エッジ検出の間引き機能」の選択肢

項目	選択肢	説明
Input A	P10	エッジ検出端子を選択
	P11	
	P12	
	P50	
	P51	
Input B <sup>注</sup>	CSC PCLK	エッジ検出クロックを選択
	HOCO HCLK_out	
	MOCO MCLK_out	
	SUB CLK SUBCLK_out	
Output C	INTELCL	出力先を選択
	DTC Start Trigger	
	SMS Start Trigger	
	A/D HW Trigger	
	D/A 0 HW Trigger	
	D/A 1 HW Trigger	
	CTSU HW Trigger	
	ITL Capture Trigger	

注. Input B は 16MHz 以下になるようにしてください。

## 5.3.1 ELCL レジスタの設定

表 5-4～表 5-8 にサンプルコードの ELCL レジスタの初期設定、図 5-5～図 5-9 にその時の ELCL の構成を示します。ELCL の全体構成は、図 4-1 を参照してください。

表 5-4 ELCL レジスタ設定 (入力)

レジスタ呼称	レジスタ名	設定値	備考
ELISEL0	入力信号選択レジスタ 0	05H	入力端子 P50 を選択
ELISEL1	入力信号選択レジスタ 1	1CH	ELCL 内部論理セルブロック L1 のフリップフロップ 0 出力を選択
ELISEL2	入力信号選択レジスタ 2	1DH	ELCL 内部論理セルブロック L1 のフリップフロップ 1 出力を選択
ELISEL3	入力信号選択レジスタ 3	1EH	ELCL 内部論理セルブロック L2 のフリップフロップ 1 出力を選択
ELISEL4	入力信号選択レジスタ 4	1FH	ELCL 内部論理セルブロック L3 のフリップフロップ 1 出力を選択
ELISEL6	入力信号選択レジスタ 6	05H	入力端子 P50 を選択
ELISEL10	入力信号選択レジスタ 10	1BH	CSC PCLK

図 5-5 ELCL の入力設定

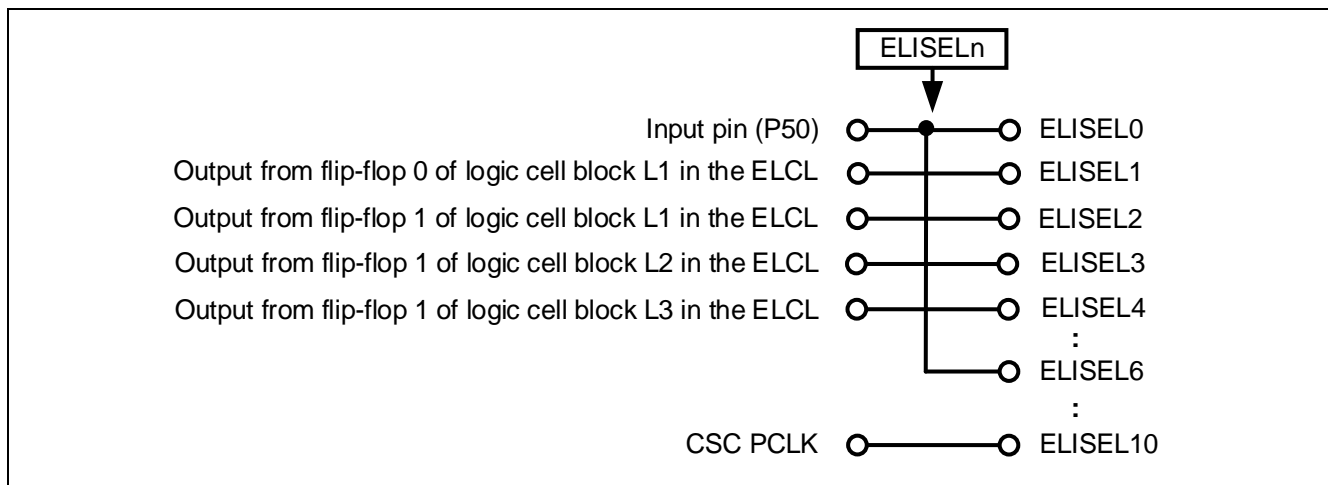


表 5-5 ELCL レジスタ設定 (論理セルブロック L1)

レジスタ呼称	レジスタ名	設定値	備考
ELL1SEL0	イベント・リンク L1 信号 選択レジスタ 0	01H	ELISEL0 の選択信号を L1 のリンク対象に選択
ELL1SEL1	イベント・リンク L1 信号 選択レジスタ 1	02H	ELISEL1 の選択信号を L1 のリンク対象に選択
ELL1SEL2	イベント・リンク L1 信号 選択レジスタ 2	03H	ELISEL2 の選択信号を L1 のリンク対象に選択
ELL1SEL3	イベント・リンク L1 信号 選択レジスタ 3	85H	ELISEL4 の選択信号を L1 のリンク対象に選択 (負論理)
ELL1SEL6	イベント・リンク L1 信号 選択レジスタ 6	05H	ELISEL10 の選択信号を L1 のリンク対象に選択
ELL1LNK0	イベント・リンク L1 出力 選択レジスタ 0	08H	ELL1SEL0 で選択した対象を論理セルブロック L1 のフリップフロップ 0 の入力にリンク
ELL1LNK1	イベント・リンク L1 出力 選択レジスタ 1	09H	ELL1SEL1 で選択した対象を論理セルブロック L1 のフリップフロップ 1 の入力にリンク
ELL1LNK2	イベント・リンク L1 出力 選択レジスタ 2	02H	ELL1SEL2 で選択した対象を論理セルブロック L1 の論理セル 0 の入力 1 にリンク
ELL1LNK3	イベント・リンク L1 出力 選択レジスタ 3	01H	ELL1SEL3 で選択した対象を論理セルブロック L1 の論理セル 0 の入力 0 にリンク
ELL1LNK6	イベント・リンク L1 出力 選択レジスタ 6	03H	ELL1SEL6 で選択した対象を論理セルブロック L1 のフリップフロップ 0、1 のクロックにリンク
ELL1CTL	論理セルブロック L1 制御 レジスタ	C1H	論理セルブロック L1 フリップフロップ 0、1 使用許可、論理セル 0 AND 回路選択

図 5-6 論理セル L1 の設定

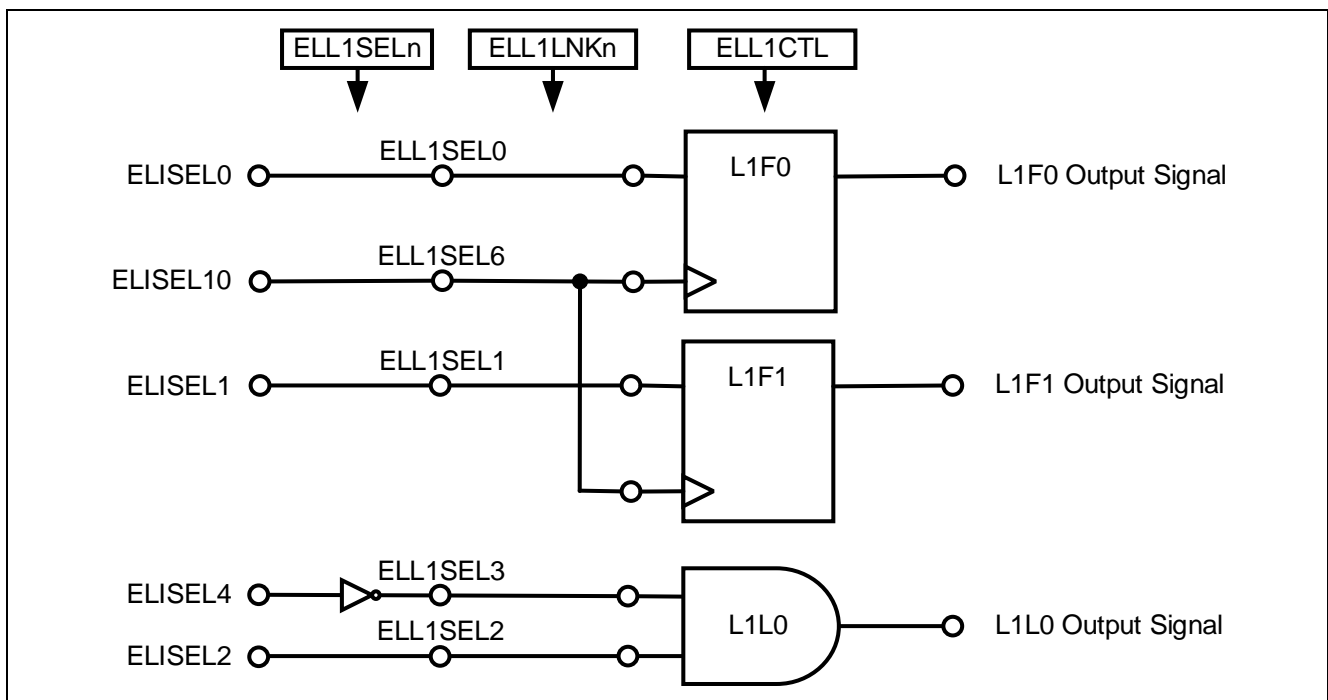


表 5-6 ELCL レジスタ設定 (論理セルブロック L2)

レジスタ呼称	レジスタ名	設定値	備考
ELL2SEL0	イベント・リンク L2 信号 選択レジスタ 0	84H	ELISEL3 の選択信号を L2 のリンク対象に選択 (負論理)
ELL2SEL1	イベント・リンク L2 信号 選択レジスタ 1	84H	ELISEL3 の選択信号を L2 のリンク対象に選択 (負論理)
ELL2SEL2	イベント・リンク L2 信号 選択レジスタ 2	0DH	論理セルブロック L1 の出力信号 0 を L2 のリンク 対象に選択
ELL2SEL6	イベント・リンク L2 信号 選択レジスタ 6	01H	ELISEL6 の選択信号を L2 のリンク対象に選択
ELL2LNK0	イベント・リンク L2 出力 選択レジスタ 0	09H	ELL2SEL0 で選択した対象を論理セルブロック L2 のフリップフロップ 1 の入力にリンク
ELL2LNK1	イベント・リンク L2 出力 選択レジスタ 1	01H	ELL2SEL1 で選択した対象を論理セルブロック L2 の論理セル 0 の入力 0 にリンク
ELL2LNK2	イベント・リンク L2 出力 選択レジスタ 2	02H	ELL2SEL2 で選択した対象を論理セルブロック L2 の論理セル 0 の入力 1 にリンク
ELL2LNK6	イベント・リンク L2 出力 選択レジスタ 6	02H	ELL2SEL6 で選択した対象を論理セルブロック L2 のフリップフロップ 1 のクロックにリンク
ELL2CTL	論理セルブロック L2 制御 レジスタ	81H	論理セルブロック L2 フリップフロップ 1 使用許 可、論理セル 0 AND 回路選択

図 5-7 論理セル L2 の設定

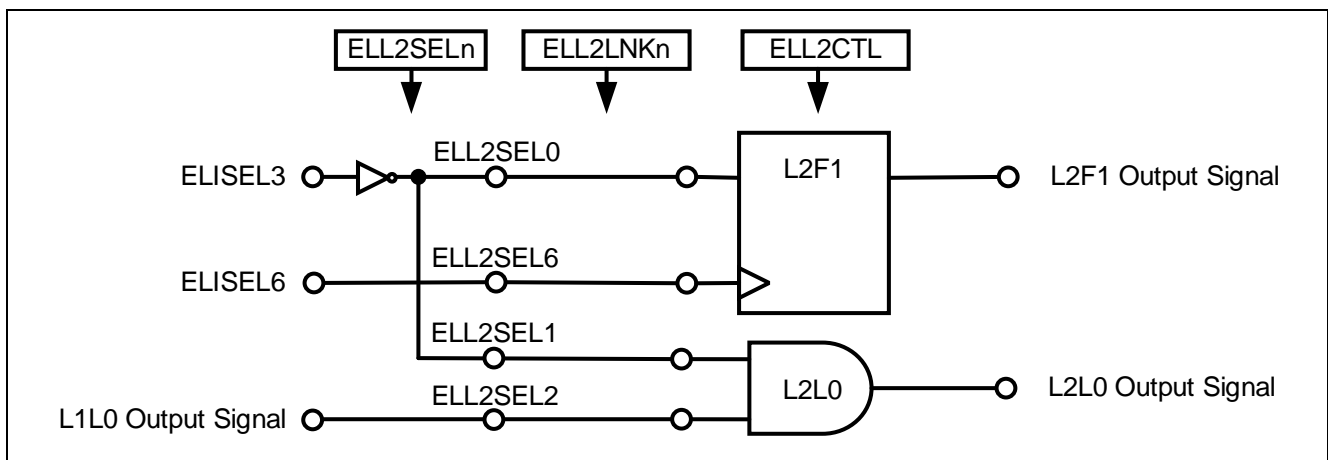


表 5-7 ELCL レジスタ設定 (論理セルブロック L3)

レジスタ呼称	レジスタ名	設定値	備考
ELL3SEL0	イベント・リンク L3 信号 選択レジスタ 0	0DH	論理セルブロック L2 の出力信号 0 を L3 のリンク対象に選択
ELL3SEL1	イベント・リンク L3 信号 選択レジスタ 1	03H	ELISEL2 の選択信号を L3 のリンク対象に選択
ELL3SEL6	イベント・リンク L3 信号 選択レジスタ 6	05H	ELISEL10 の選択信号を L3 のリンク対象に選択
ELL3LNK0	イベント・リンク L3 出力 選択レジスタ 0	08H	ELL3SEL0 で選択した対象を論理セルブロック L3 のフリップフロップ 0 の入力にリンク
ELL3LNK1	イベント・リンク L3 出力 選択レジスタ 1	09H	ELL3SEL1 で選択した対象を論理セルブロック L3 のフリップフロップ 1 の入力にリンク
ELL3LNK6	イベント・リンク L3 出力 選択レジスタ 6	03H	ELL3SEL6 で選択した対象を論理セルブロック L3 のフリップフロップ 0、1 のクロックにリンク
ELL3CTL	論理セルブロック L3 制御 レジスタ	C0H	論理セルブロック L3 フリップフロップ 0、1 使用許可

図 5-8 論理セル L3 の設定

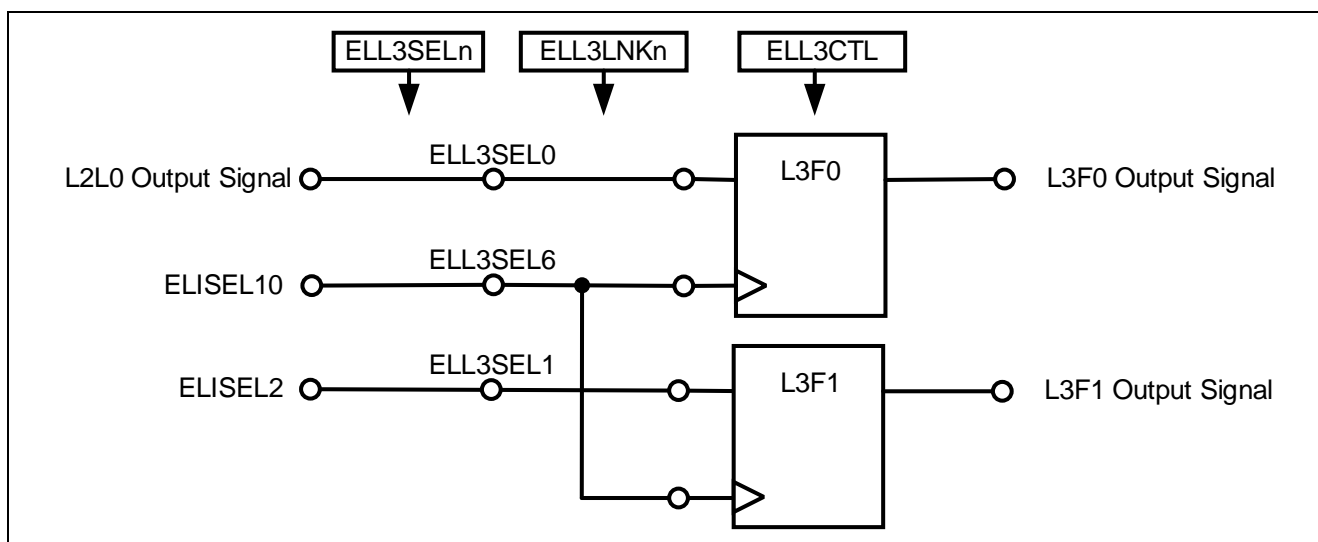
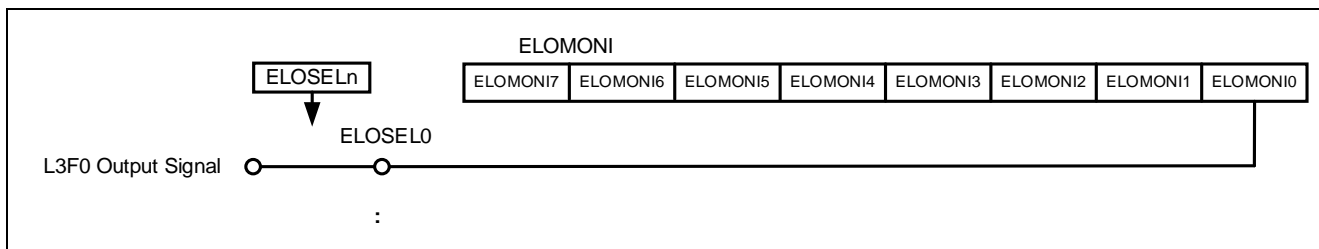




表 5-8 ELCL レジスタ設定 (出力)

レジスタ呼称	レジスタ名	設定値	備考
ELOSEL0	出力信号選択レジスタ 0	0EH	論理セルブロック L3 の出力信号[3]を選択

図 5-9 ELCL の出力設定



## 6. サンプルコード

サンプルコードは、ルネサスエレクトロニクスホームページから入手してください。

## 7. 参考ドキュメント

RL78/G23 ユーザーズマニュアル ハードウェア編 (R01UH0896J)

RL78 ファミリ ユーザーズマニュアル ソフトウェア編 (R01US0015J)

RL78 スマート・コンフィグレータ ユーザーガイド : CS+編 (R20AN0580J)

RL78 スマート・コンフィグレータ ユーザーガイド : e<sup>2</sup> studio 編 (R20AN0579J)

RL78 スマート・コンフィグレータ ユーザーガイド : IAR 編 (R20AN0581J)

(最新版をルネサスエレクトロニクスホームページから入手してください。)

テクニカルアップデート/テクニカルニュース

(最新版の情報をルネサスエレクトロニクスホームページから入手してください。)

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。

## 改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	Apr.13.21	-	初版発行
2.00	Mar.24.22	5	表 2-1 動作確認条件 動作電圧 立ち上がり時 TYP.1.875V ⇒ 1.90V 立ち下がり時 TYP.1.835V ⇒ 1.86V
		5	ツールバージョンを更新 表 2-1 動作確認条件 統合開発環境 (CS+) : CS+ for CC E8.05.00f ⇒ V8.07.00 C コンパイラ (CS+) : CC-RL V1.09 ⇒ V1.11 統合開発環境 (e <sup>2</sup> studio) : e <sup>2</sup> studio 2021-01 (21.1.0) ⇒ 2022-01 (22.1.0) C コンパイラ (e <sup>2</sup> studio) : CC-RL V1.09 ⇒ V1.11 スマート・コンフィグレータ : V.1.0.0 ⇒ V.1.2.0 統合開発環境 (IAR) : V4.20.1 ⇒ V4.21.1 ボードサポートパッケージ (r_bsp) : V.1.00 ⇒ V.1.13
		5, 6 18, 19	COM ポート対応に伴う変更 表 2-1 動作確認条件 エミュレータ : E2 エミュレータ Lite ⇒ CS+, e <sup>2</sup> studio : COM ポート IAR : E2 エミュレータ Lite 図 3-1 ハードウェア構成例 P11/TOOLRxD, P12/TOOLTxD 追加 表 5-1 スマート・コンフィグレータの設定値 注の追加
		5,10,18	ユーザーマニュアルの動作条件に合わせた変更 表 2-1 動作確認条件 動作周波数 高速オンチップ・オシレータ・クロック : 32MHz ⇒ 16MHz CPU/周辺ハードウェア・クロック : 32MHz ⇒ 16MHz 表 4-1 オプション・バイト設定 設定値 1110 1000B(E8H) ⇒ 1110 1001B(E9H) 内容 高速オンチップ・オシレータの周波数 : 32MHz ⇒ 16MHz 表 5-1 スマート・コンフィグレータの設定値 クロック 内容 高速オンチップ・オシレータ : 32MHz ⇒ 16MHz f <sub>IHP</sub> : 32MHz ⇒ 16MHz f <sub>CLK</sub> : 32000kHz ⇒ 16MHz (高速オンチップ・オシレータ)
		9	サンプルコード更新に伴い、表 4-1 フォルダ構成を更新 IAR 版サンプルコード更新に伴い、注 3 を追記
10	表 4-2 オプション・バイト設定 検出電圧 立ち上がり 1.875V/立下り 1.835V ⇒ 立ち上がり 1.90V/立下り 1.86V		

Rev.	発行日	改訂内容	
		ページ	ポイント
2.00	Mar.24.22	10	コンポーネント「ELCL エッジ検出の間引き機能」更新に伴う更新 表 7-1 サンプルコードで使用するグローバル変数 変数名 : g_elcl_interrupt 使用関数 r_elcl_interrupt ⇒ r_Config_EdgeDetectionThinningFunction_interrupt
		11	コンポーネント「ELCL エッジ検出の間引き機能」更新に伴う更新 表 4-5 関数一覧 r_elcl_start、r_elcl_stop、r_elcl_interrupt 削除 r_Config_EdgeDetectionThinningFunction_interrupt 追加 r_elcl_reset_flipflop,ソースファイル elcl_support.c ⇒ Config_EdgeDetectionThinningFunction_user.c
		11	IAR 版サンプルコード更新に伴う変更 4.7 関数仕様 [関数名]main, ヘッダ e <sup>2</sup> studio, CS+ : r_smc_entry.h IAR : ior7f100g.h, ior7f100g_ext.h, r_cg_macrodriver.h, Config_SMS.h, Config_ITL000_ITL001.h ⇒ r_smc_entry.h
		11	[関数名] main,説明 P50 に 2 回パルスが入力されると HALT モードを解除し、 LED1 が消灯してフリップフロップを初期化します ⇒ P50 に 2 回パルスが入力されると HALT モードが解除され、 LED1 が消灯してフリップフロップを初期化します
		11	[関数名] r_Config_EdgeDetectionThinningFunction_interrupt, 説明 INTELCL が発生すると HALT モードを解除します ⇒ g_elcl_interrupt = 1 に設定します
		11-12	サンプルコード更新に伴う変更 4.7 関数仕様 r_elcl_start、r_elcl_stop、r_elcl_interrupt 削除 r_Config_EdgeDetectionThinningFunction_interrupt 追加 [関数名] r_elcl_reset_flipflop,ヘッダ r_cg_macrodriver.h, elcl_support.h ⇒ platform.h
		13	図 4-3 メイン処理 Yes/No 分岐を修正

Rev.	発行日	改訂内容	
		ページ	ポイント
2.00	Mar.24.22	13-15, 16-17, 20	<p>コンポーネント「ELCL エッジ検出の間引き機能」更新に伴う更新</p> <p>図 4-3 メイン処理 関数名： R_Config_EdgeDetectionThinningFunction_Create() ⇒ R_Config_EdgeDetectionThinningFunction_Start() 図 4-6 ELCL 割り込み処理 関数名： r_elcl_interrupt() ⇒ r_Config_EdgeDetectionThinningFunction_interrupt()</p> <p>図 4-4 ELCL 出力開始処理 図 4-5 ELCL 出力停止処理 フローチャートの削除</p> <p>図 4-6 ELCL 割り込み処理 図番号：図 4-6 ⇒ 図 4-4 図 4-7 ELCL のフリップフロップリセット処理 図番号：図 4-7 ⇒ 図 4-5 図 4-8 ウェイト処理 図番号：図 4-8 ⇒ 図 4-6 図 4-9 TAU0 チャンネル 7 割り込み処理 図番号：図 4-9 ⇒ 図 4-7</p> <p>図 5-1 コンポーネントの追加 図 5-2 モジュールのダウンロード 図 5-3 モジュールの選択 図 5-4 コンポーネント「ELCL エッジ検出の間引き機能」図の更新</p>
		17,19-20	ELCL Edge Detection Thinning Function ⇒ ELCL エッジ検出の間引き機能
		18	<p>表 5-1 スマート・コンフィグレータの設定値 クロック：fsXL ⇒ fsXP コンポーネント Config_LVD0 リセット発生電圧 (VLVD0)：1.835 (V) ⇒ 1.86(V)</p>
		20	<p>コンポーネント「ELCL エッジ検出の間引き機能」更新に伴う更新</p> <p>表 7-2 コンポーネント「ELCL エッジ検出の間引き機能」の選択肢 Input A、Input B、Output C の選択肢を追加</p>
		26	<p>RL78 スマート・コンフィグレータ ユーザーガイドの追記</p> <p>7. 参考ドキュメント</p> <p>RL78 スマート・コンフィグレータ ユーザーガイド： CS+編 (R20AN0580J) RL78 スマート・コンフィグレータ ユーザーガイド： e<sup>2</sup> studio 編 (R20AN0579J) RL78 スマート・コンフィグレータ ユーザーガイド： IAR 編 (R20AN0581J)</p>

## 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しないでください。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えば、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）、交通管制（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も、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、これに限られません。）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（豊洲フォレスト）

[www.renesas.com](http://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/](http://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。