

RL78/G23

ELCL 複数パラメータ監視機能

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）を使った複数の入力信号（パラメータ）の監視方法を説明します。ELCL を使うことで、外付け部品や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（外付け部品、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	4
2. 動作確認条件	6
3. ハードウェア説明	7
3.1 ハードウェア構成例	7
3.2 使用端子一覧	7
4. ソフトウェア説明	8
4.1 動作概要	8
4.2 フォルダ構成	11
4.3 オプション・バイトの設定一覧	12
4.4 定数一覧	12
4.5 変数一覧	13
4.6 関数一覧	13
4.7 関数仕様	14
4.8 フローチャート	16
4.8.1 メイン処理	16
4.8.2 TAU0 のチャンネル変更処理	17
4.8.3 TAU0 のリセット処理	18
4.8.4 ELCL 割り込み処理	19
4.8.5 ELCL のフリップフロップリセット処理	19
4.8.6 INTP0 割り込み処理	20
4.8.7 ウェイト処理	21
4.8.8 TAU0 チャンネル7 割り込み処理	21
5. 応用例	22
5.1 ELCL のコンポーネントの設定	22
5.2 r01an5615_elcl_parameter.scfg	24
5.2.1 クロック	26
5.2.2 システム	26
5.2.3 r_bsp	26
5.2.4 Config_LVD0	26
5.2.5 Config_INTC	26
5.2.6 Config_TAU0_0、Config_TAU0_1、Config_TAU0_2、Config_TAU0_3	26
5.2.7 Config_TAU0_7	26
5.2.8 Config_MultipleParameterMonitoring	27
5.2.9 Config_PORT	27
5.3 コンポーネント「ELCL 複数パラメータ監視機能」	28
5.3.1 ELCL レジスタの設定	32
5.3.2 外部割り込み機能を持たない4つのポートを割り込みとして使用する例	37
5.3.3 入力を負論理にする方法	38
6. サンプルコード	39

7. 参考ドキュメント.....39

改訂記録.....40

1. 仕様

本アプリケーションノートでは、ELCL を使って 4 つの入力信号を論理和した出力信号を生成し、その出力信号を指定した周辺機器へ接続する例を説明します。

図 1-1 に ELCL の構成を図 1-2 にタイミングチャートを示します。

4 つの入力信号 (INPUT A、INPUT B、INPUT C、INPUT D) を 3 つの論理和 (L1L0、L1L1、L2L0) にリンクし、生成した信号を OUTPUT E へリンクします。また、4 つの入力信号は、フリップフロップ (L2F0、L2F1、L3F0、L3F1) にリンクし、フリップフロップの出力をモニタレジスタ (ELOMONI) にリンクします。これにより、モニタレジスタをリードすることで入力信号の状態を確認することが可能です。例えば、外部割り込み機能を持たない 4 つのポート (P20~P23) を INPUT A~INPUT D にリンクして、OUTPUT E のリンク先をイベント・リンク割り込み (INTELCL) とします。これにより、P20~P23 のいずれか 1 つでも変化すると INTELCL が発生し、外部割り込み機能を持たないポートを割り込みとして取り扱うことができます。

図 1-1 ELCL の構成

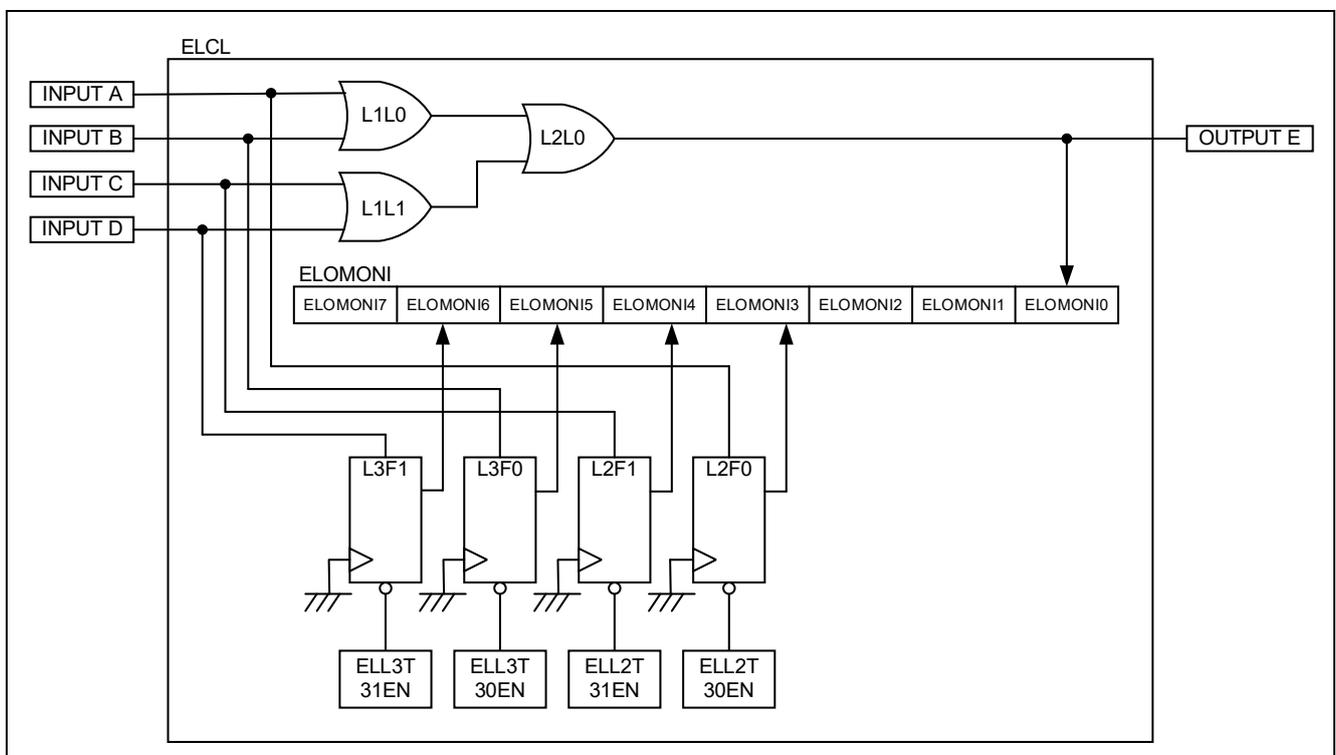
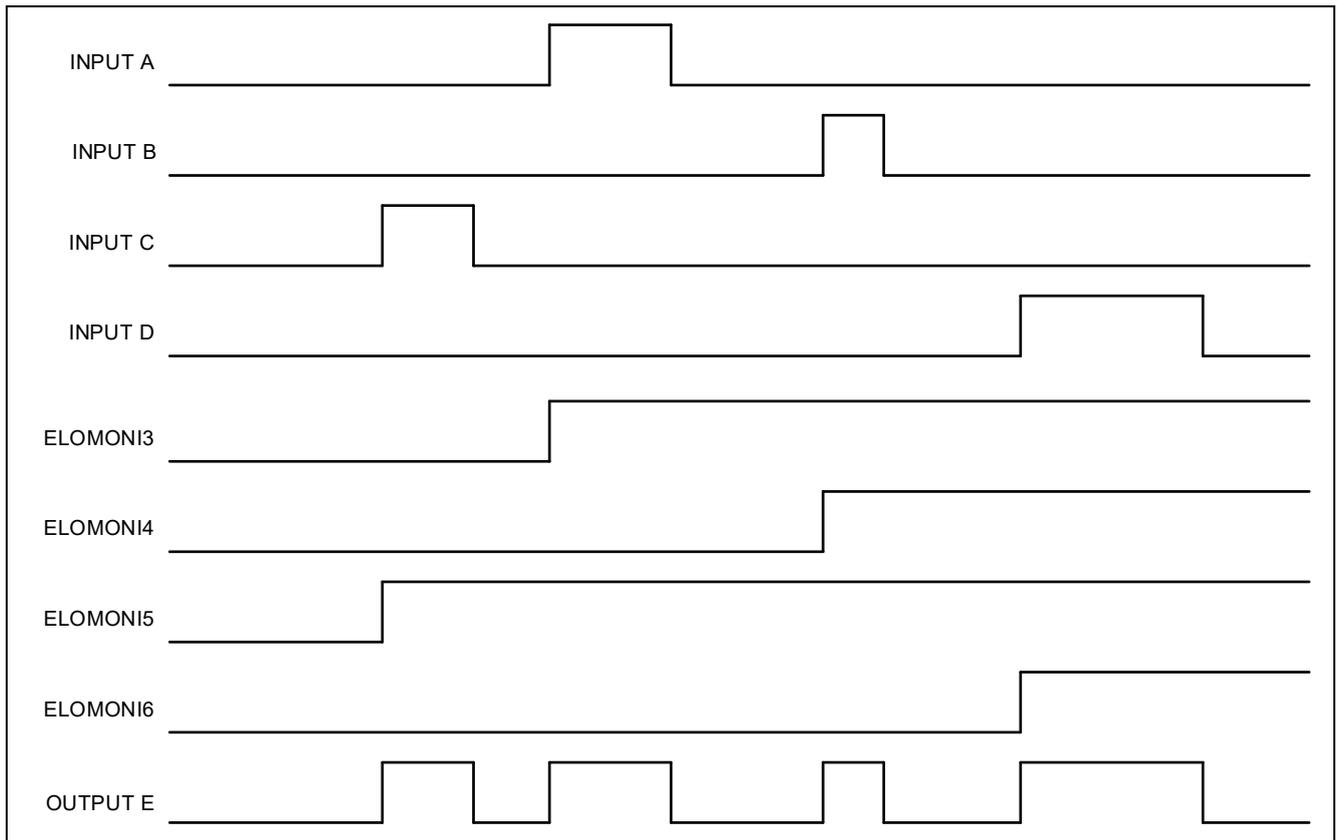


図 1-2 タイミングチャート



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

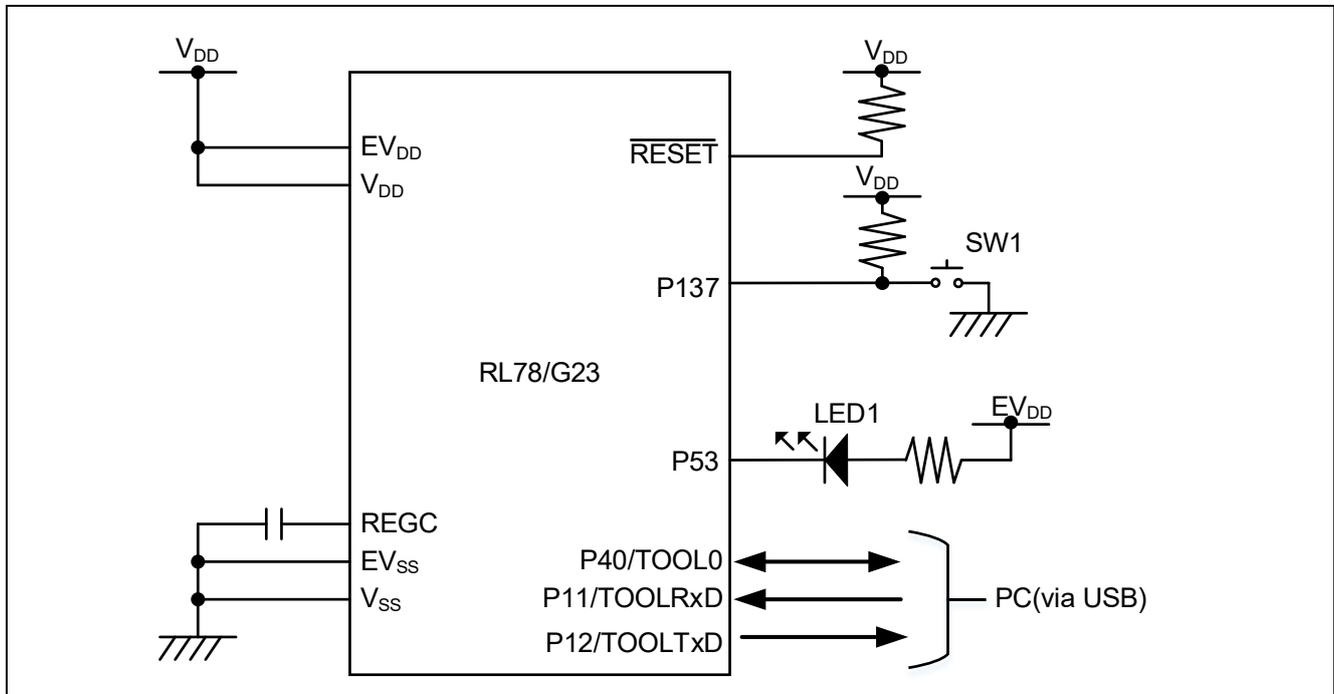
項目	内容
使用マイコン	RL78/G23 (R7F100GLG)
動作周波数	<ul style="list-style-type: none"> 高速オンチップ・オシレータ・クロック : 32MHz CPU/周辺ハードウェア・クロック : 32MHz
動作電圧	<ul style="list-style-type: none"> 3.3V LVD0 動作 (V_{LVD0}) : リセット・モード 立ち上がり時 TYP. 1.90V 立ち下がり時 TYP. 1.86V
統合開発環境 (CS+)	ルネサスエレクトロニクス製 CS+ for CC V8.07.00
C コンパイラ (CS+)	ルネサスエレクトロニクス製 CC-RL V1.11
統合開発環境 (e ² studio)	ルネサスエレクトロニクス製 e ² studio 2022-01 (22.01.0)
C コンパイラ (e ² 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 ² 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)
P137	入力	SW1 (Low Active)

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、4つのタイマ（TAU0 チャンネル0、TAU0 チャンネル1、TAU0 チャンネル2、TAU0 チャンネル3）をインターバル・タイマ・モードで動作させます。4つのタイマ完了割り込み（INTTM00、INTTM01、INTTM02、INTTM03）はELCLで論理和（OR）し、INTELCLとして出力します。INTELCLが発生するとポート（P53）の出力が反転し、LED1は点灯または消灯します。SW1を押下する毎に4つのタイマ出力の組み合わせが変化するため、それに応じてLED1の点灯間隔が変化します。

表 4-1 に動作概要を示します。No. (10) の次は No. (1) に戻ります。

表 4-1 動作概要（タイマの組み合わせ）

No.	g_timer_mode	SW1	TAU0				LED1
			チャンネル 0	チャンネル 1	チャンネル 2	チャンネル 3	
(1)	0	-	停止	停止	停止	停止	消灯
(2)	1	押下	停止	停止	停止	停止	消灯
(3)	1	-	カウント	停止	停止	停止	点滅 (約 2 秒間隔)
(4)	2	押下	停止	停止	停止	停止	消灯
(5)	2	-	カウント	カウント	停止	停止	点滅 (約 1 秒間隔)
(6)	3	押下	停止	停止	停止	停止	消灯
(7)	3	-	カウント	カウント	カウント	停止	点滅 (約 0.5 秒間隔)
(8)	4	押下	停止	停止	停止	停止	消灯
(9)	4	-	カウント	カウント	カウント	カウント	点滅 (約 0.25 秒間隔)
(10)	0	押下	停止	停止	停止	停止	消灯

図 4-1 に動作概要、図 4-2 にタイミングチャートを示します。

図 4-1 動作概要

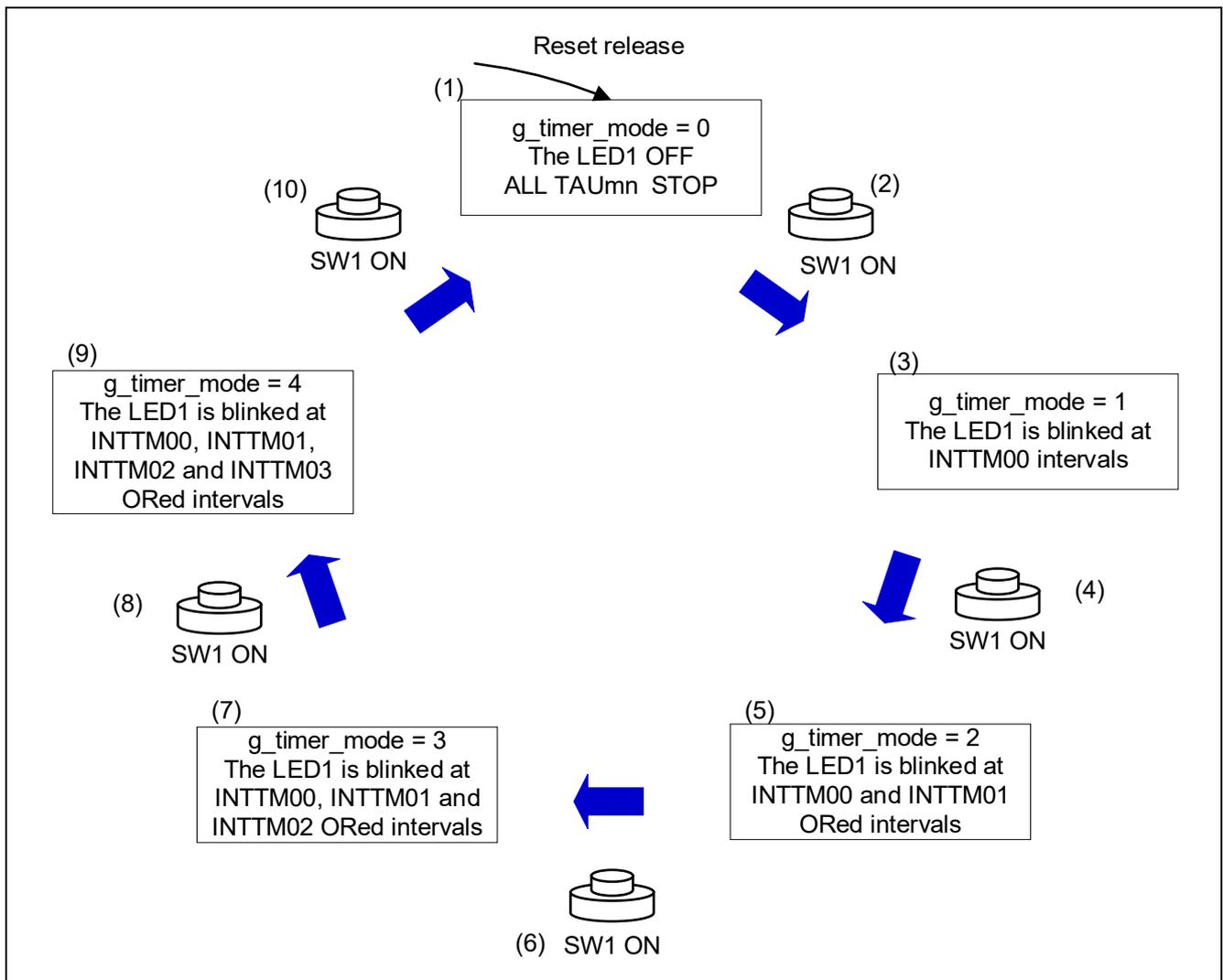


図 4-2 サンプルコードのタイミングチャート

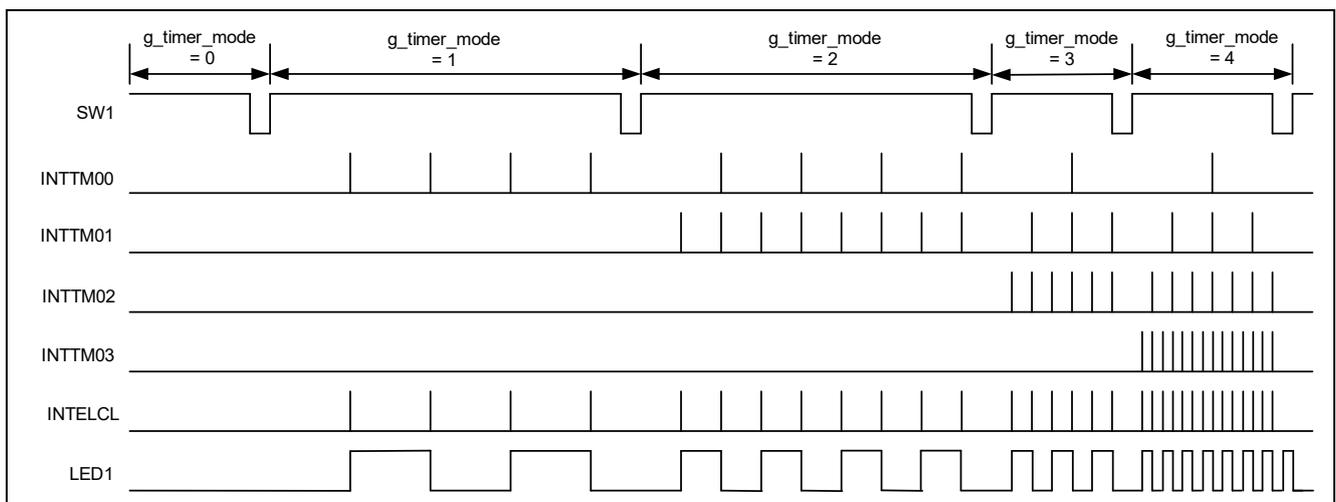
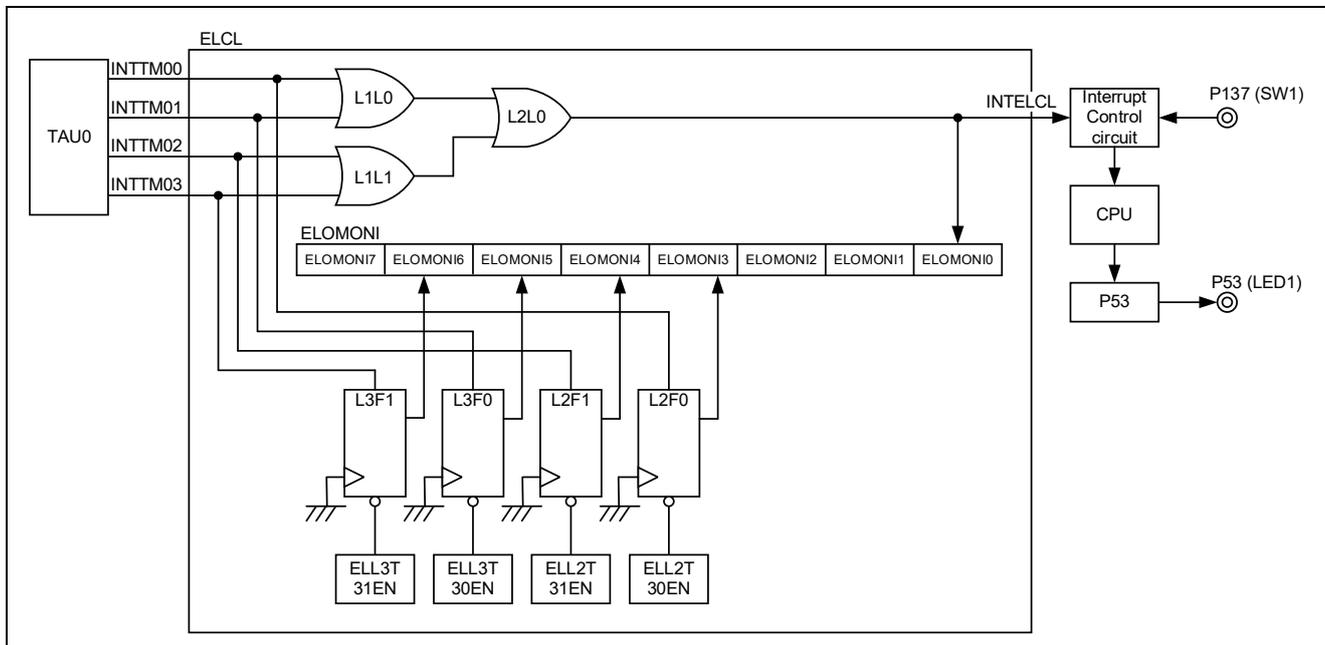


図 4-3 にサンプルコードのシステム構成を示します。

ELCL の入力信号として INTTM00、INTTM01、INTTM02、INTTM03 を、ELCL の出力信号として INTELCL を選択します。

図 4-3 サンプルコードのシステム構成



4.2 フォルダ構成

表 4-2 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 4-2 フォルダ構成 (1/2)

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
¥r01an5615_elcl_parameter<DIR>	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	
support_functions.c	関数用ソースファイル	
support_functions.h	関数用ヘッダファイル	
¥smc_gen<DIR>	スマート・コンフィグレータ生成フォルダ	√
¥Config_INTC<DIR>	割り込み用プログラム格納フォルダ	√
Config_INTC.c	INTP0 (SW1) 用ソースファイル	√
Config_INTC.h	INTP0 用ヘッダファイル	√
Config_INTC_user.c	INTP0 用割り込みソースファイル	√注2
¥Config_MultipleParameterMonitoring<DIR>	ELCL 用プログラム格納フォルダ	√
Config_MultipleParameterMonitoring.c	ELCL 用ソースファイル	√
Config_MultipleParameterMonitoring.h	ELCL 用ヘッダファイル	√
Config_MultipleParameterMonitoring_user.c	ELCL 用割り込みソースファイル	√注2
¥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	√注1
¥Config_TAU0_0<DIR>	TAU0 チャンネル 0 用プログラム格納フォルダ	√
Config_TAU0_0.c	TAU0 チャンネル 0 用ソースファイル	√
Config_TAU0_0.h	TAU0 チャンネル 0 用ヘッダファイル	√
Config_TAU0_0_user.c	TAU0 チャンネル 0 用割り込みソースファイル	√注1
¥Config_TAU0_1<DIR>	TAU0 チャンネル 1 用プログラム格納フォルダ	√
Config_TAU0_1.c	TAU0 チャンネル 1 用ソースファイル	√
Config_TAU0_1.h	TAU0 チャンネル 1 用ヘッダファイル	√
Config_TAU0_1_user.c	TAU0 チャンネル 1 用割り込みソースファイル	√注1
¥Config_TAU0_2<DIR>	TAU0 チャンネル 2 用プログラム格納フォルダ	√
Config_TAU0_2.c	TAU0 チャンネル 2 用ソースファイル	√
Config_TAU0_2.h	TAU0 チャンネル 2 用ヘッダファイル	√
Config_TAU0_2_user.c	TAU0 チャンネル 2 用割り込みソースファイル	√注1
¥Config_TAU0_3<DIR>	TAU0 チャンネル 3 用プログラム格納フォルダ	√
Config_TAU0_3.c	TAU0 チャンネル 3 用ソースファイル	√
Config_TAU0_3.h	TAU0 チャンネル 3 用ヘッダファイル	√
Config_TAU0_3_user.c	TAU0 チャンネル 3 用割り込みソースファイル	√注1

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。

表 4-3 フォルダ構成 (2/2)

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
¥r01an5615_elcl_parameter<DIR> ^{注3}	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	
¥smc_gen<DIR>	スマート・コンフィグレータ生成フォルダ	√
¥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	√ ^{注2}
¥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 r01an5615_elcl_parameter.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 4-4 にオプション・バイト設定を示します。

表 4-4 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 1000B (E8H)	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：32MHz
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 4-5 にサンプルコードで使用する定数を示します。

表 4-5 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
CHATTA_WAIT	100	チャタリング防止時間 (100ms)	Config_INTC_user.c

4.5 変数一覧

表 4-6 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 4-6 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	ウェイト処理のカウント値	r_ms_delay, r_Config_TAU0_7_interrupt
uint8_t	g_led_flag	INTELCL の発生フラグ main 関数でポーリングし、LED をトグルさせる。	main r_Config_MultipleParameterMon itoring_interrupt
uint8_t	g_timer_mode	タイマ動作モード	r_timer_update r_Config_MultipleParameterMon itoring_interrupt

4.6 関数一覧

表 4-7 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 4-7 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_timer_update	動作する TAU0 のチャンネル変更処理	support_functions.c
r_timer_reset	TAU0 の停止処理	support_functions.c
r_Config_MultipleParameterMonitoring_interrupt	ELCL 割り込み処理	Config_MultipleParameterMonitoring_user.c
r_elcl_reset_flipflop	ELCL 搭載フリップフロップのリセット処理	Config_MultipleParameterMonitoring_user.c
r_Config_INTC_intp0_interrupt	INTP0 割り込み処理	Config_INTC_user.c
r_ms_delay	チャタリング防止用ウェイト処理	Config_TAU0_7_user.c
r_Config_TAU0_7_interrupt	TAU0 チャンネル 7 割り込み処理 (チャタリング防止用)	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, support_functions.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 INTELCL から復帰するたびに P53 が反転し、LED1 が ON/OFF します ELCL 割り込み発生フラグ g_led_flag を 0 にします
引数	なし
リターン値	なし
備考	なし

[関数名] r_timer_update

概要	TAU0 動作チャンネル変更処理
ヘッダ	r_smc_entry.h, support_functions.h
宣言	void timer_update (void);
説明	g_timer_mode の値によって、動作する TAU のチャンネルの組み合わせを変え、TAU を開始/停止します
引数	なし
リターン値	なし
備考	なし

[関数名] r_timer_reset

概要	TAU0 リセット処理
ヘッダ	r_smc_entry.h, support_functions.h
宣言	void timer_reset (void);
説明	TAU0 を停止し、カウンタのリセット後、スタート待ち状態にします
引数	なし
リターン値	なし
備考	なし

[関数名] r_Config_MultipleParameterMonitoring_interrupt

概要	ELCL 割り込み処理
ヘッダ	platform.h
宣言	#pragma interrupt r_Config_MultipleParameterMonitoring_interrupt (vect=INTELCL)
説明	ELCL 割り込み発生フラグ g_led_flag を 1 にします
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	ELOMONI フラグをリセットするためにフリップフロップをリセットします
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_Config_INTC_intp0_interrupt

概要	INTP0 割り込み処理
ヘッダ	r_cg_macrodriver.h, r_cg_userdefine.h, Config_INTC.h, Config_TAU0_7.h
宣言	#pragma interrupt r_Config_INTC_intp0_interrupt (vect=INTP0)
説明	タイマの状態を示す g_timer_mode を設定します INTP0 割り込み発生フラグ g_timer_flag を 1 にします SW1 押下のチャタリング対策としてウエイト処理を実行します
引数	なし
リターン値	なし
備考	なし

[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 CHATTA_WAIT 未満の場合はポーリングし、CHATTA_WAIT 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

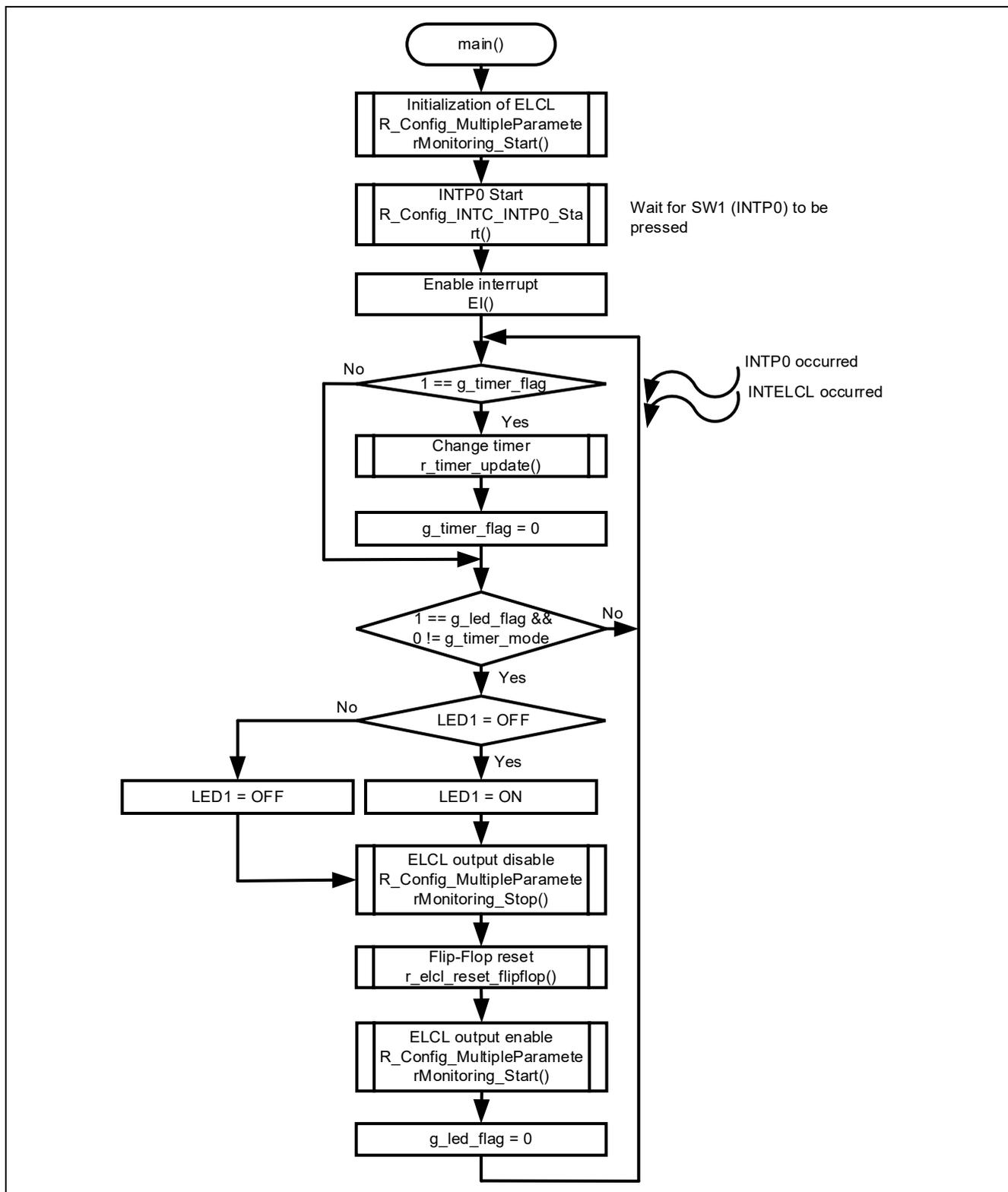
概要	TAU0 チャンネル 7 割り込み処理
ヘッダ	r_cg_macrodriver.h, r_cg_userdefine.h, Config_TAU0_7.h
宣言	#pragma interrupt r_Config_TAU0_7_interrupt (vect=INTTM07)
説明	TAU0 チャンネル 7 の INTTM07 による割り込み処理です。 g_ms_timer をカウントアップします。
引数	なし
リターン値	なし
備考	なし

4.8 フローチャート

4.8.1 メイン処理

図 4-4 にメイン処理のフローチャートを示します。

図 4-4 メイン処理



4.8.2 TAU0 のチャンネル変更処理

図 4-5 に TAU0 のチャンネル変更処理のフローチャートを示します。

図 4-5 TAU0 のチャンネル変更処理 (1/2)

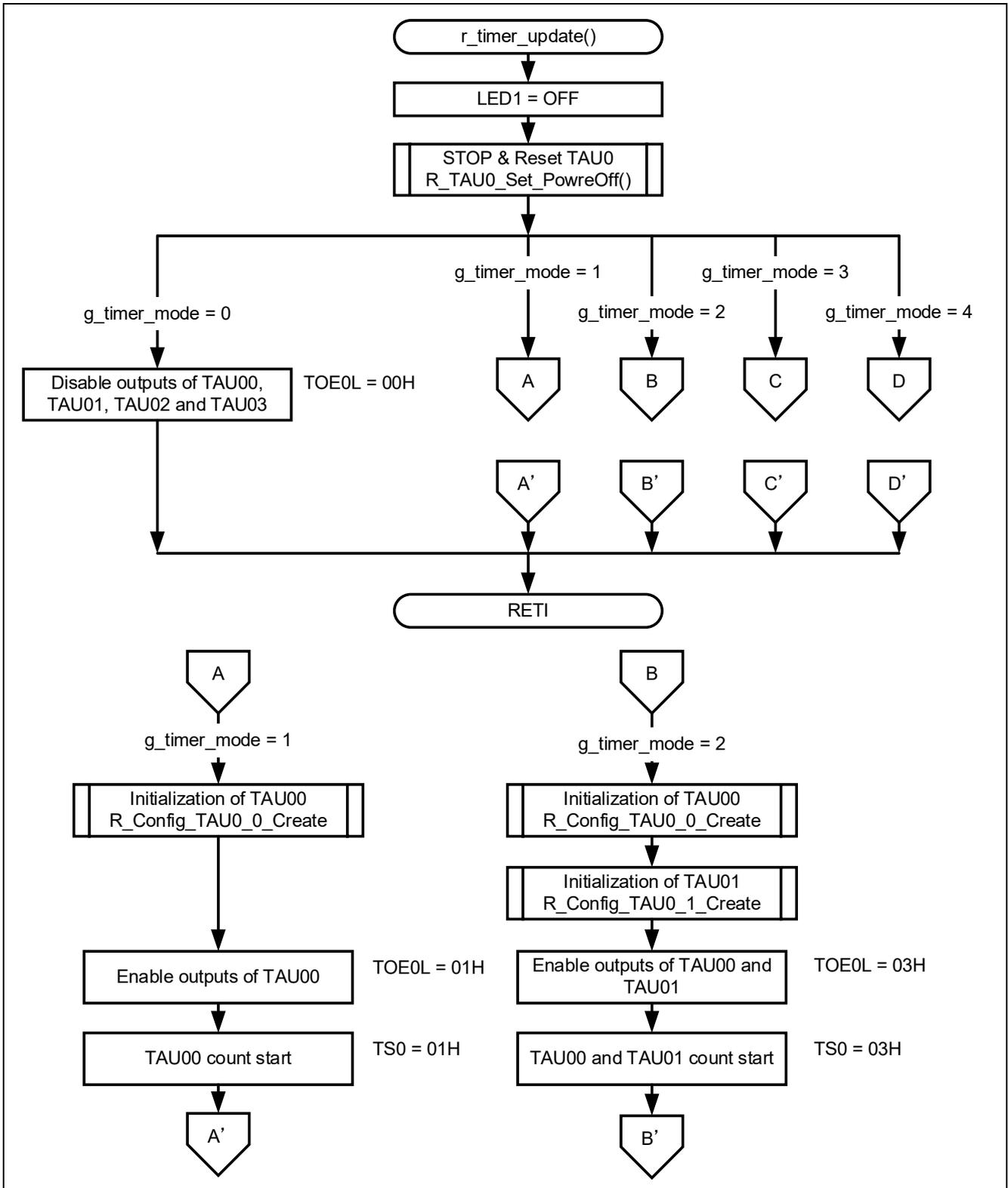
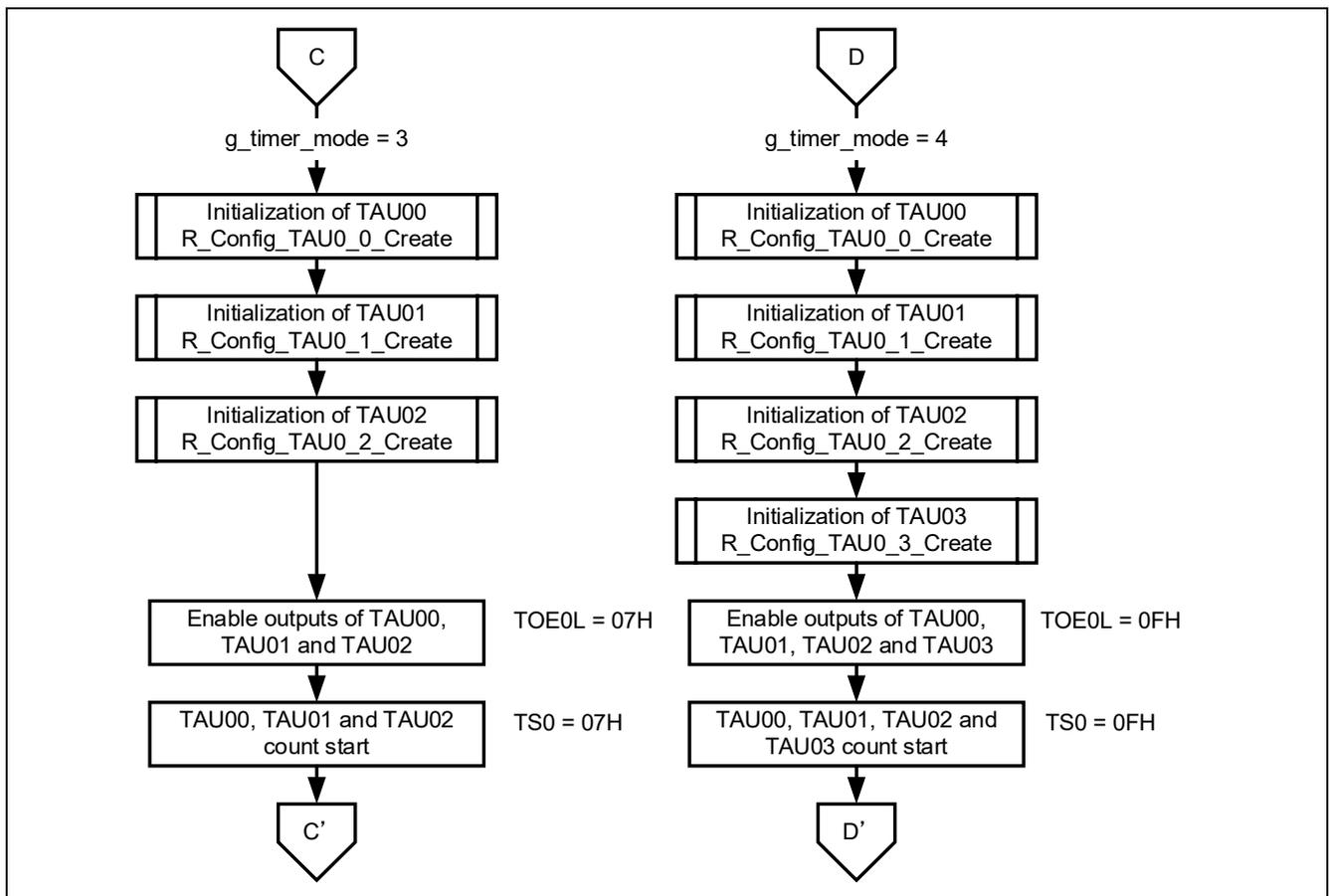


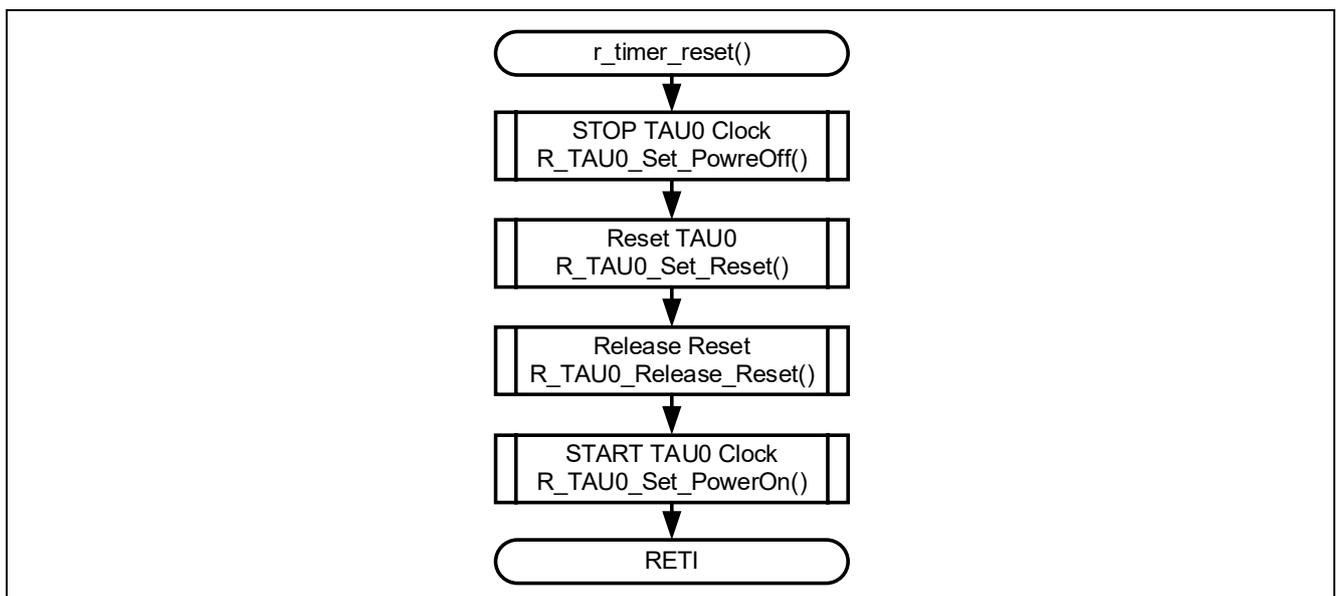
図 4-6 TAU0 のチャンネル変更処理 (2/2)



4.8.3 TAU0 のリセット処理

図 4-7 に TAU0 の停止処理のフローチャートを示します。

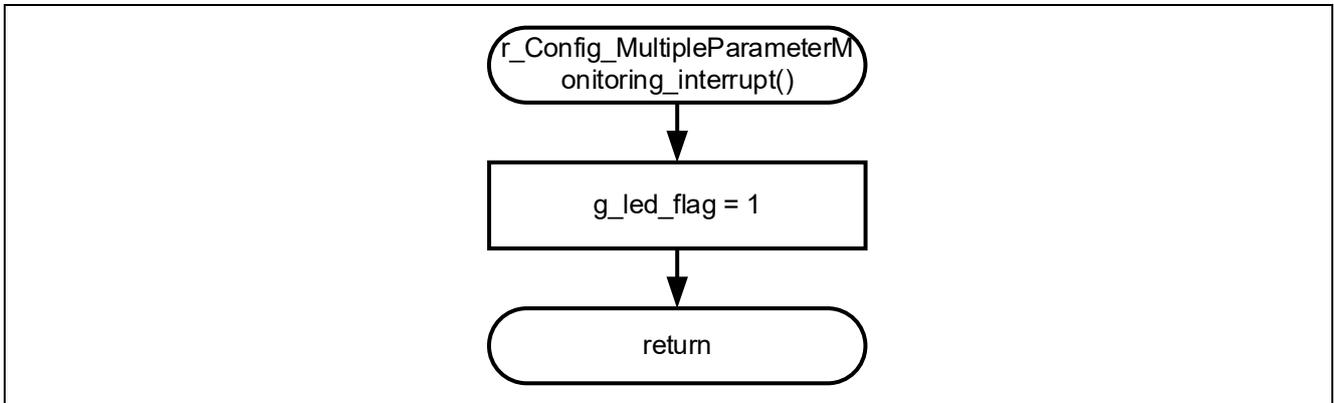
図 4-7 TAU0 のリセット処理



4.8.4 ELCL 割り込み処理

図 4-8 に ELCL 出力割り込み処理のフローチャートを示します。

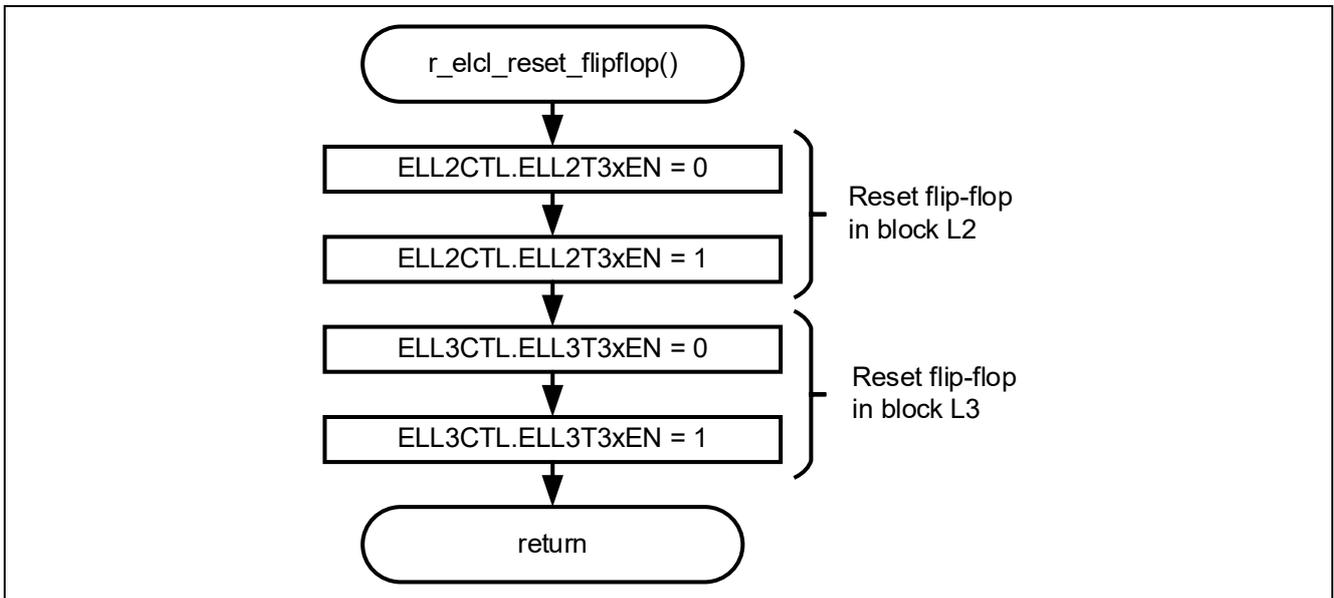
図 4-8 ELCL 割り込み処理



4.8.5 ELCL のフリップフロップリセット処理

図 4-9 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。

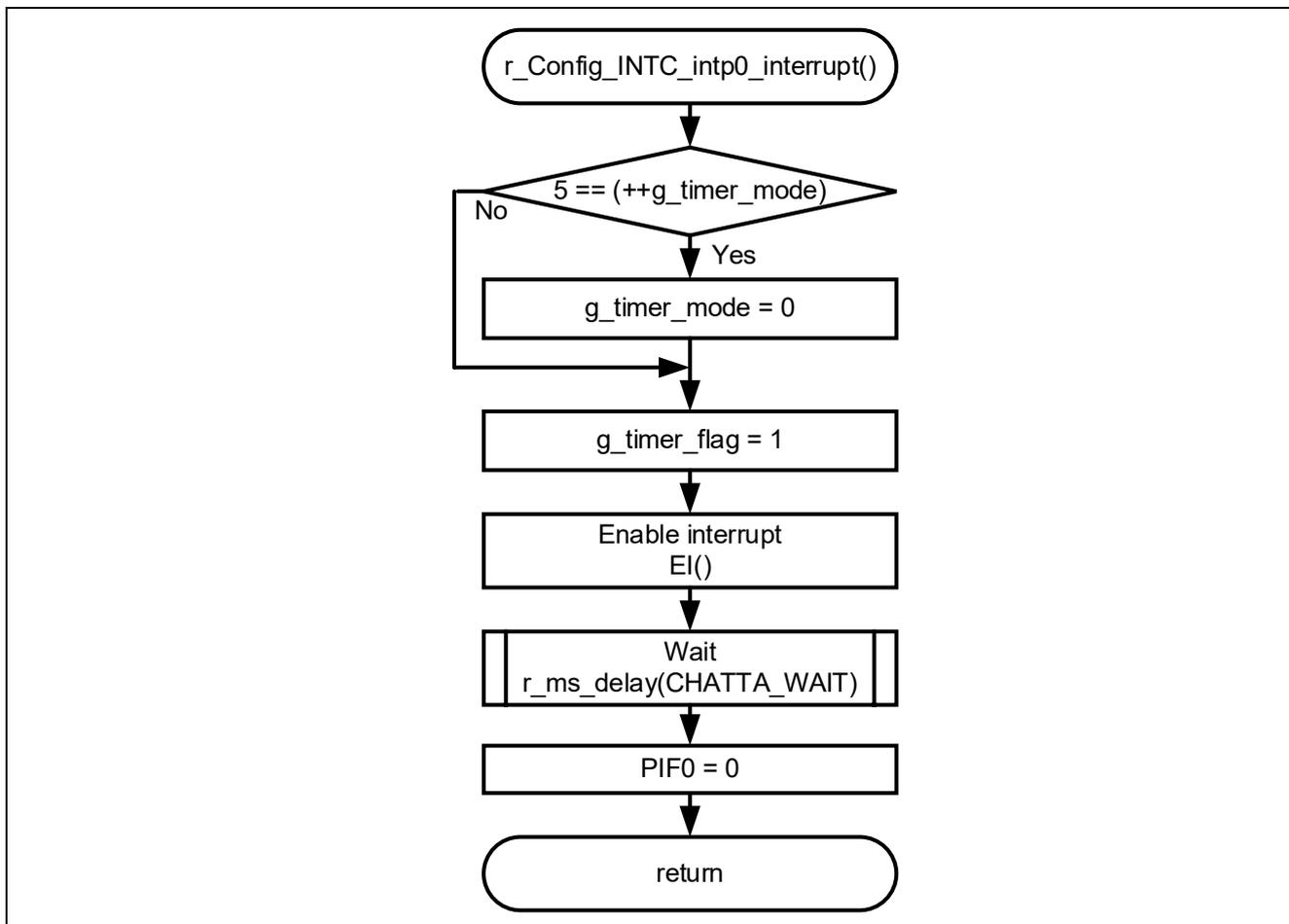
図 4-9 ELCL のフリップフロップリセット処理



4.8.6 INTP0 割り込み処理

図 4-10、に INTP0 割り込み処理のフローチャートを示します。

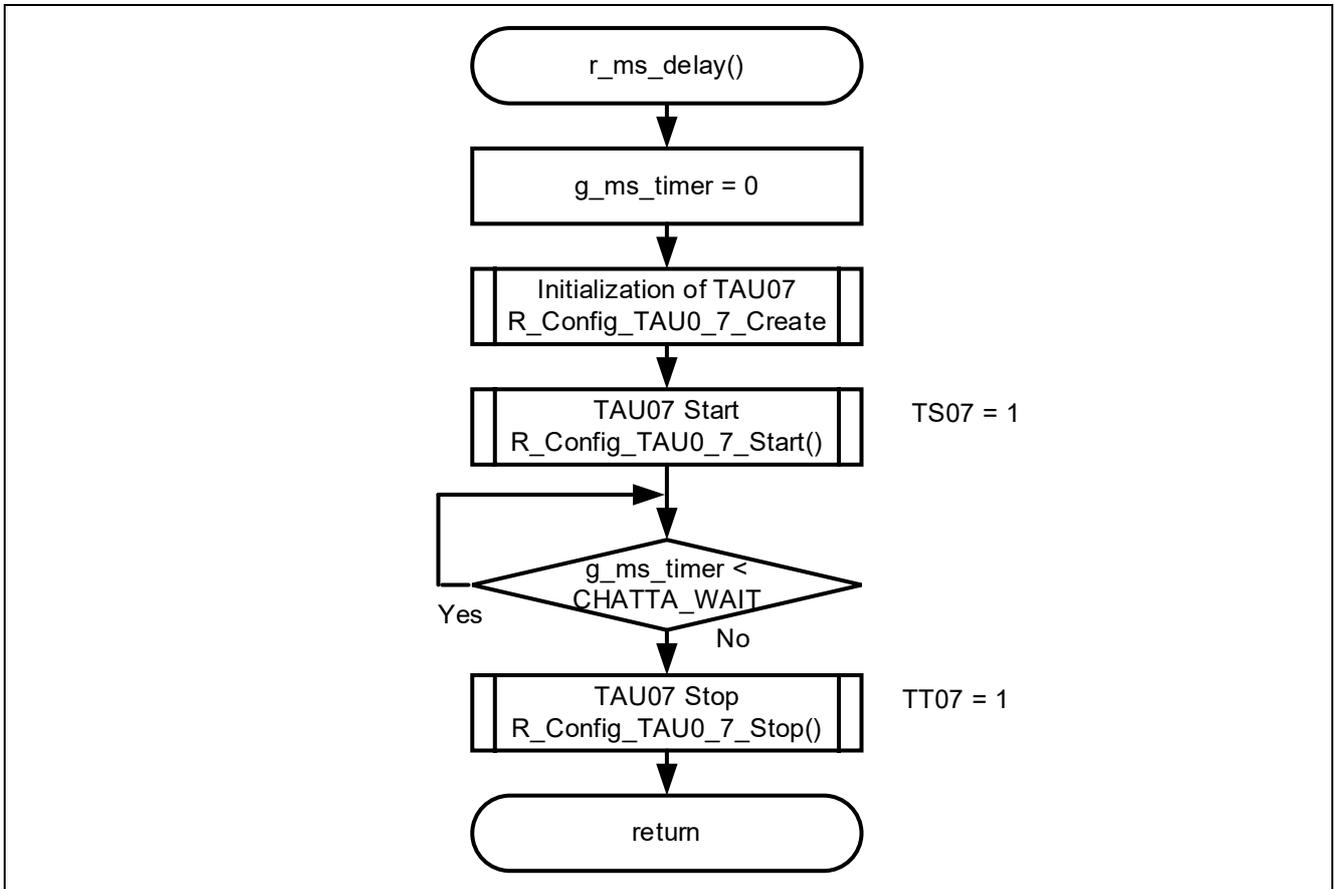
図 4-10 INTP0 割り込み処理



4.8.7 ウェイト処理

図 4-11 にウェイト処理のフローチャートを示します。

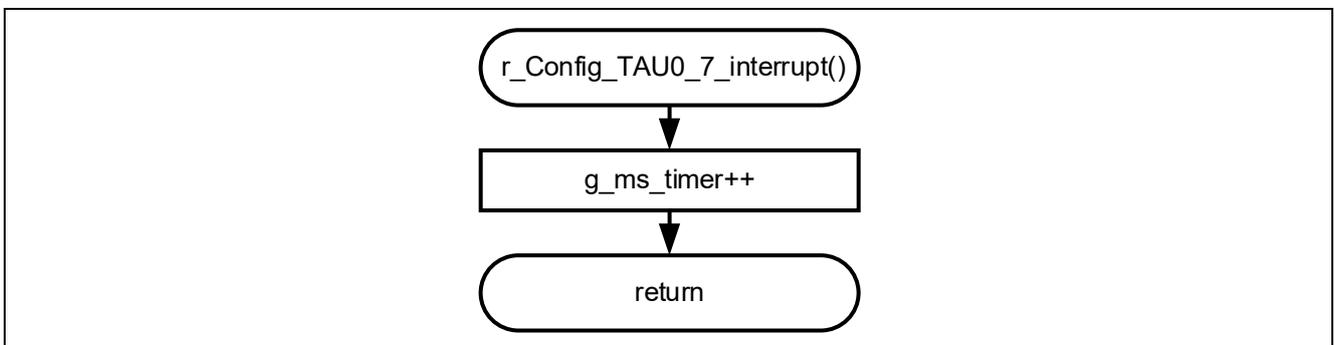
図 4-11 ウェイト処理



4.8.8 TAU0 チャンネル 7 割り込み処理

図 4-12 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 4-12 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。

r01an5615_elcl_parameter.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。

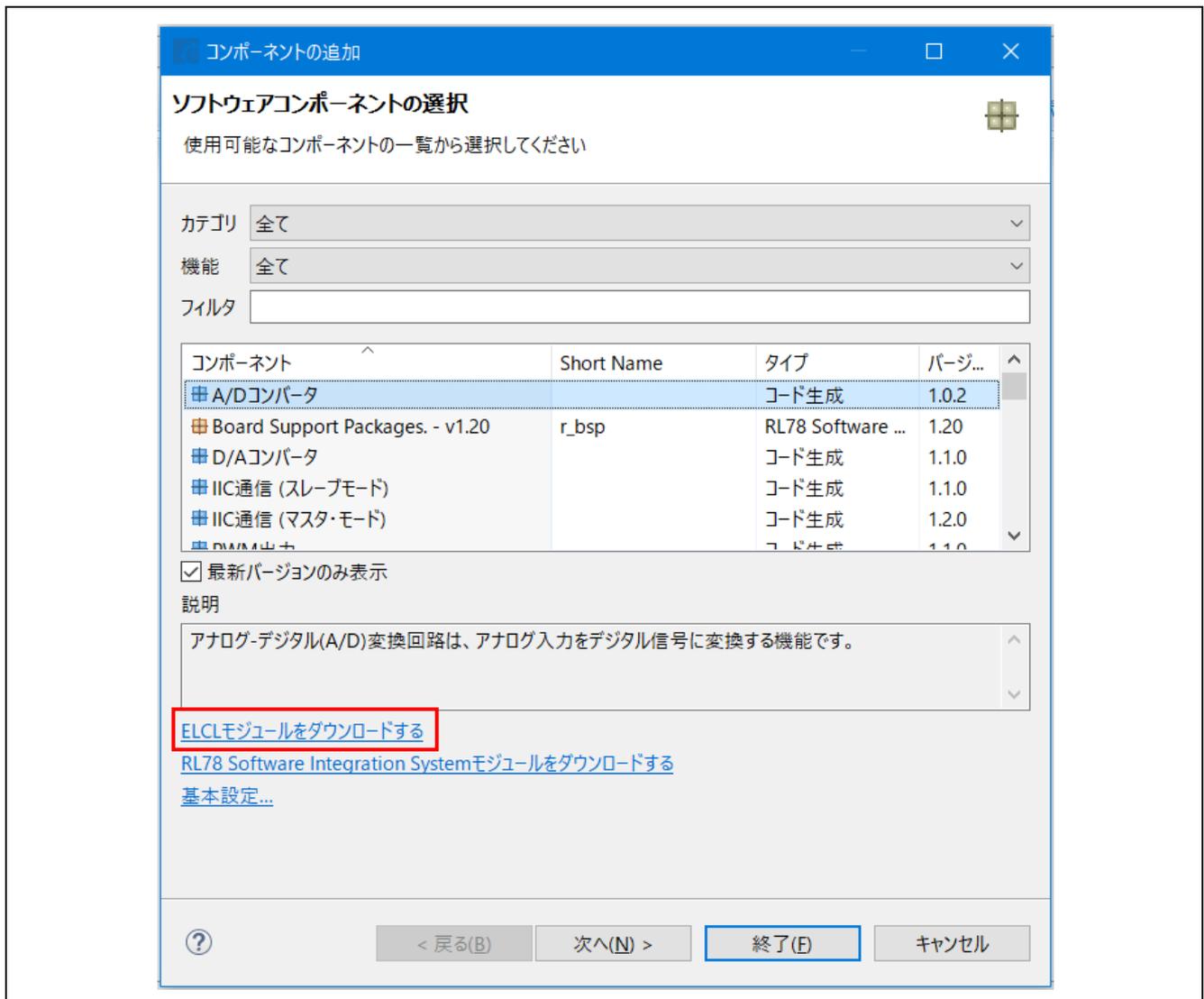
5.1 ELCL のコンポーネントの設定

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。

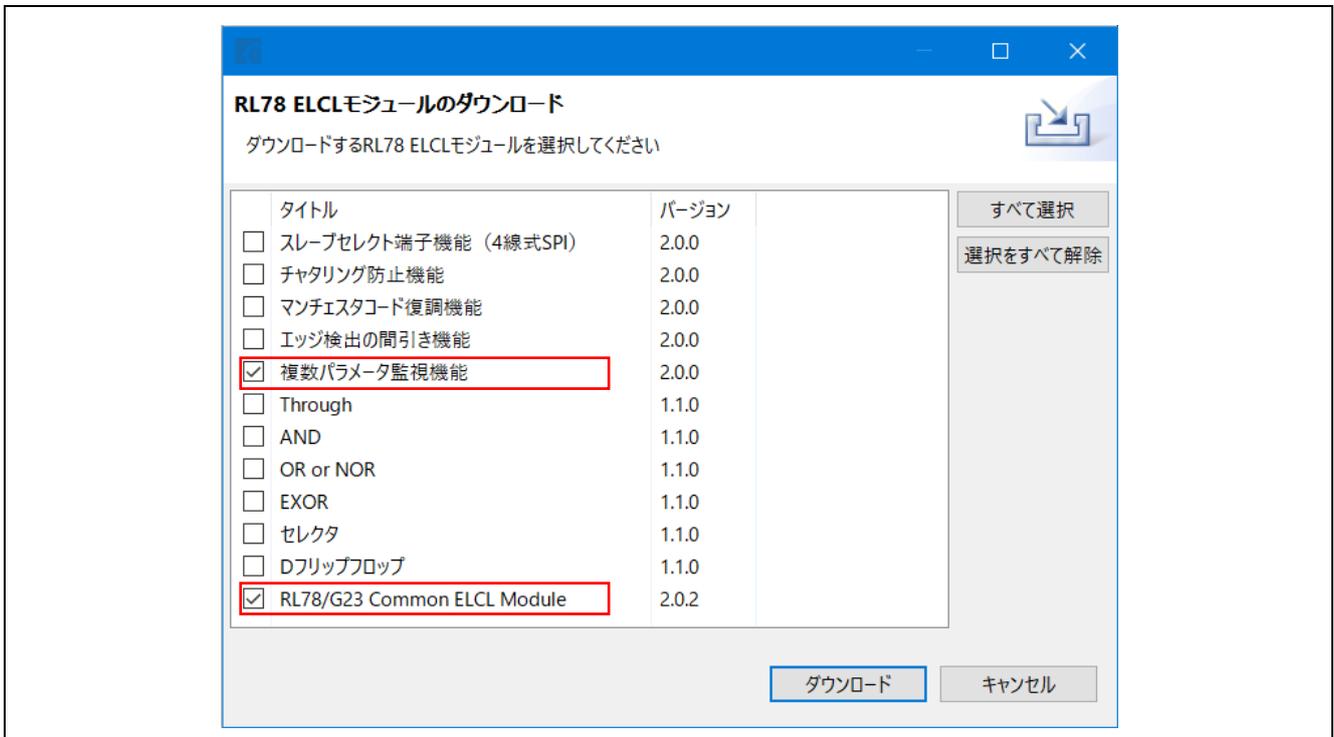
1. スマート・コンフィグレータを起動してください。
2. 「コンポーネント」タグをクリックし、「コンポーネントの追加」をクリックしてください。
3. 図 5-1 に示す「コンポーネントの追加」のウィンドウが開きますので、「ELCL モジュールをダウンロードする」をクリックしてください。

図 5-1 コンポーネントの追加



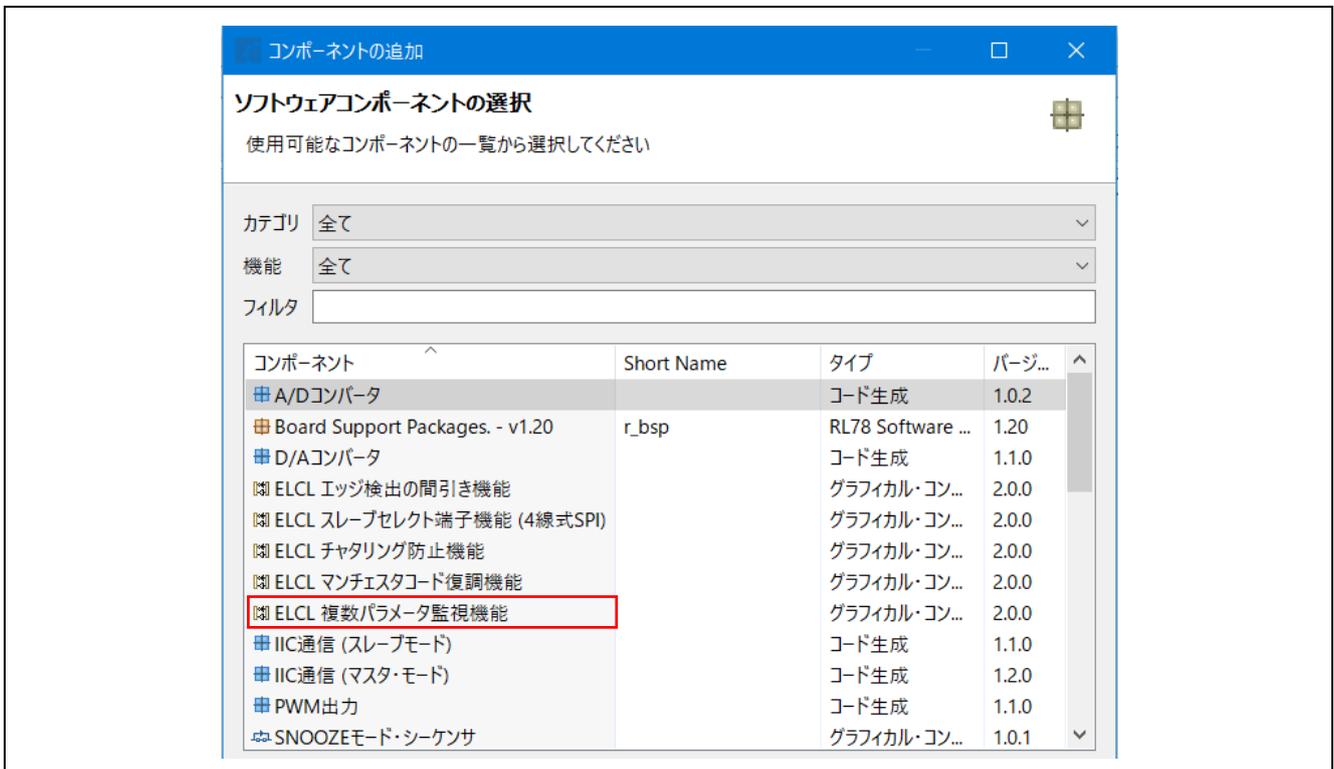
4. 「複数パラメータ監視機能」を選択し、ダウンロードしてください。共通設定ファイル「RL78/G23 Common ELCL Module」もダウンロードしてください。

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 複数パラメータ監視機能」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 r01an5615_elcl_parameter.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 _{DD} 設定：1.8V ≤ EV _{DD0} < 5.5V 高速オンチップ・オシレータ：32MHz f _{IHP} ：32MHz f _{CLK} ：32MHz（高速オンチップ・オシレータ） f _{SXP} ：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 ^注 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_LVDD0	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 _{LVDD0})：1.86 (V)
	Config_INTC	INTP0 設定：使用する 有効エッジ：立ち下がりエッジ 優先順位：レベル 3

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

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_TAU0_0	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_0 動作クロック：CK00 クロックソース： $f_{CLK}/2^{12}$ インターバル時間：4000 カウント 割り込み設定：使用しない
	Config_TAU0_1	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_1 動作クロック：CK00 クロックソース： $f_{CLK}/2^{12}$ インターバル時間：2000 カウント 割り込み設定：使用しない
	Config_TAU0_2	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_2 動作クロック：CK00 クロックソース： $f_{CLK}/2^{12}$ インターバル時間：1000 カウント 割り込み設定：使用しない
	Config_TAU0_3	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_3 動作クロック：CK00 クロックソース： $f_{CLK}/2^{12}$ インターバル時間：500 カウント 割り込み設定：使用しない
	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： $f_{CLK}/2^{12}$ 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 2
	Config_MultipleParameterMonitoring	コンポーネント：ELCL 複数パラメータ監視機能 Source A：ELISEL_6, INTTM00 Source B：ELISEL_7, INTTM01 Source C：ELISEL_8, INTTM02 Source D：ELISEL_9, INTTM03 Output signal selector：割り込み (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)

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 4-4 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_LVD0

サンプルコードの電源管理の設定を行います。

「表 4-4 オプション・バイト設定」の「LVD0 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_INTC

サンプルコードで使用する割り込みの設定を行います。

サンプルコードでは、外部マスカブル割り込み (INTP0) を設定します。INTP0 を使用しない場合は削除してください。

5.2.6 Config_TAU0_0、Config_TAU0_1、Config_TAU0_2、Config_TAU0_3

サンプルコードの TAU00、TAU01、TAU02、TAU03 の設定を行います。

サンプルコードでは「インターバル・タイマ」として使用し、チャンネルごとにインターバル時間を変えています。インターバル時間を変更すると LED が点滅する間隔が変わります。

TAU00、TAU01、TAU02、TAU03、個別の割り込みは使用していません。

5.2.7 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、INTP0 に対するチャタリング対策で使用します。INTP0 を使用しない場合、またはチャタリング対策が不要な場合は削除してください。

5.2.8 Config_MultipleParameterMonitoring

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入力に INTTM00、INTTM01、INTTM02、INTTM03 を選択して、出力に INTELCL を選択しています。

詳細は、「5.3 コンポーネント「ELCL 複数パラメータ監視機能」」を参照してください。

5.2.9 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 複数パラメータ監視機能」

図 5-4 にコンポーネント「ELCL 複数パラメータ監視機能」、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 複数パラメータ監視機能」

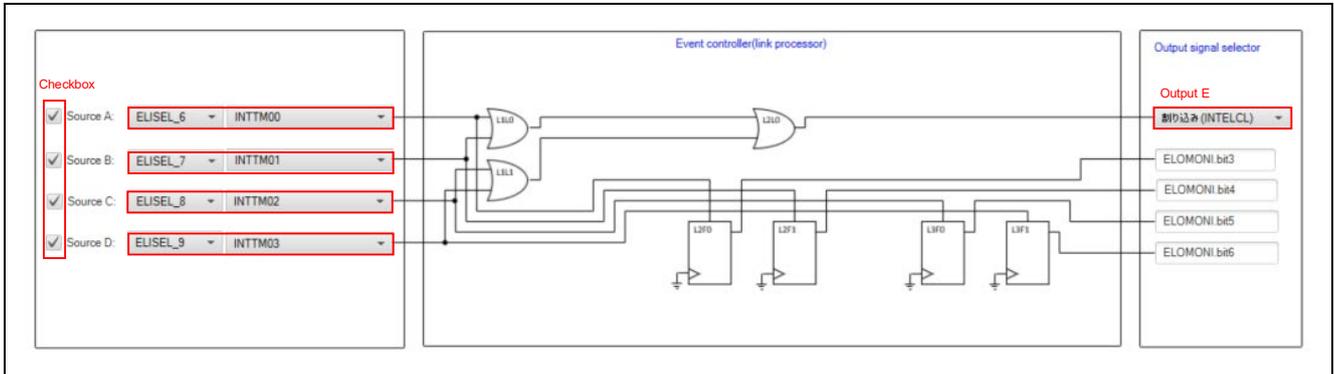


表 5-3 コンポーネント「ELCL 複数パラメータ監視機能」の選択肢

項目	選択肢		説明
Checkbox	チェックあり：入力要因有効 チェックなし：Low 固定		入力要因有効、または Low 固定を選択
Source A	ELISEL_0	表 5-4、表 5-5、表 5-6 に記載の信号 ^{注2}	入力要因を選択
Source B	~ 11 ^{注1}		
Source C	ELISEL_6	表 5-4、表 5-6 に記載の信号 ^{注2}	
Source D	~ 11 ^{注1}		
Output E	割り込み (INTELCL)		出力先を選択
	DTC 起動トリガ		
	SMS 起動トリガ		
	A/D コンバータハードウェア・トリガ		
	D/A コンバータ 0 ハードウェア・トリガ		
	D/A コンバータ 1 ハードウェア・トリガ		
	CTSU ハードウェア・トリガ		
	ITL キャプチャ・トリガ		

注 1. ELISEL_0 ~ 11、ELISEL_6 ~ 11 は、それぞれ ELISEL0 ~ 11、ELISEL6 ~ 11 に相当します。

注 2. Source A ~ B は入力要因の選択に ELISEL_0 ~ 11 の選択も必要です。Source C ~ D は入力要因の選択に ELISEL_6 ~ 11 の選択も必要です。表 5-4 ~ 表 5-6 を参照し正しく設定を行ってください。選択した組み合わせが正しくない場合、ワーニングが表示されます。

表 5-4 入力要因の選択肢 (ELISEL0 ~ 11)

対応する ELISELn レジスタ	入力要因
ELISEL0 ~ 11	SAU0 チャンネル 0 データ出力
	SAU0 チャンネル 1 データ出力
	SAU0 チャンネル 0 通信クロック出力
	SAU0 チャンネル 1 通信クロック出力
	P11
	P50
	UARTA0 通信クロック出力
	TAU0 チャンネル 0 出力
	TAU0 チャンネル 1 出力
	TAU0 チャンネル 2 出力
	TAU0 チャンネル 3 出力
	TAU0 チャンネル 4 出力
	TAU0 チャンネル 5 出力
	TAU0 チャンネル 6 出力
	TAU0 チャンネル 7 出力
	P10
	P51
	コンパレータ検出 0
	コンパレータ検出 1
	UARTA0 データ出力
	P12

表 5-5 入力要因の選択肢 (ELISEL0 ~ 5)

対応する ELISELn レジスタ	入力要因
ELISEL0	P20
	P120
	INTP6
	INTURE0/INTTM10
	INTTM16
	INTUT1
ELISEL1	P21
	P121
	INTP7
	INTURE1/INTTM11
	INTTM17
	INTUR1
ELISEL2	P22
	P122
	INTP8
	INTTM12
	INTST3/INTCSI30/INTIIC30
	INTIICA1
ELISEL3	P23
	P147
	INTP9
	INTSRE3/INTTM13H
	INTSR3/INTCSI31/INTIIC31
ELISEL4	P30
	P00
	INTKR
	INTTM14
	INTUTO
ELISEL5	P31
	P01
	INTTM13
	INTTM15
	INTUR0
ELISEL0 ~ 5	P137

表 5-6 入力要因の選択肢 (ELISEL6~11)

対応する ELISELn レジスタ	入力要因
ELISEL6	INTP0
	INTTM00
	INTTM06
	INTST2/INTCSI20/INTIIC20
	INTSR1/INTCSI11/INTIIC11
	INTSMSE
ELISEL7	INTP1
	INTTM01
	INTITL
	INTSR2/INTCSI21/INTIIC21
	INTSRE1/INTTM03H
	INTP10/INTCMP0
ELISEL8	INTP2
	INTTM02
	INTWDTI
	INTSRE2/INTM11H
	INTREMC
	INTP11/INTCMP1
ELISEL9	INTP3
	INTTM03
	INTRTC
	INTST0/INTCSI00/INTIIC00
	INTSR0/INTCSI01/INTIIC01
	INTCTSUWR
ELISEL10	INTP4
	INTTM04
	INTTM07
	INTSRE0/INTTM01H
	INTLVI
	INTCTSURD
ELISEL11	INTP5
	INTTM05
	INTIICA0
	INTST1/INTCSI10/INTIIC10
	INTAD
	INTCTSUFN
ELISEL6 ~ 11	CSC PCLK
	HOCO HCLK_out
	MOCO MCLK_out
	SUB_CLK SUBCLK_out

5.3.1 ELCL レジスタの設定

表 5-7～表 5-11 に 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 4-3 を参照してください。

表 5-7 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
ELISEL6	入力信号選択レジスタ 6	16H	INTTM00 を選択
ELISEL7	入力信号選択レジスタ 7	16H	INTTM01 を選択
ELISEL8	入力信号選択レジスタ 8	16H	INTTM02 を選択
ELISEL9	入力信号選択レジスタ 9	16H	INTTM03 を選択

図 5-5 ELCL の入力設定

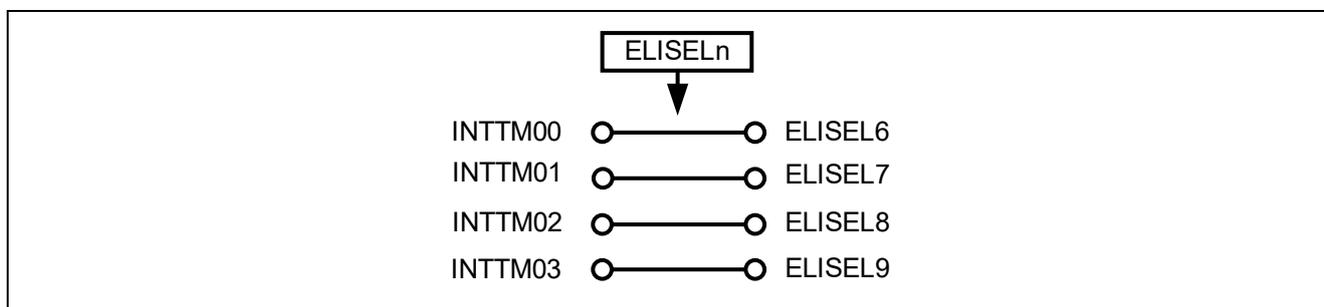


表 5-8 ELCL レジスタ設定 (論理セルブロック L1)

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	07H	ELISEL6 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	08H	ELISEL7 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	09H	ELISEL8 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	0AH	ELISEL9 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	01H	ELL1SEL0 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
ELL1LNK1	イベント・リンク L1 出力 選択レジスタ 1	02H	ELL1SEL1 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
ELL1LNK2	イベント・リンク L1 出力 選択レジスタ 2	03H	ELL1SEL2 で選択した対象を論理セルブロック L1 の論理セル 1 の入力 0 にリンク
ELL1LNK3	イベント・リンク L1 出力 選択レジスタ 3	04H	ELL1SEL3 で選択した対象を論理セルブロック L1 の論理セル 1 の入力 1 にリンク
ELL1CTL	論理セルブロック L1 制御 レジスタ	0AH	論理セル 0 は OR 回路選択 論理セル 1 は OR 回路選択

図 5-6 論理セル L1 の設定

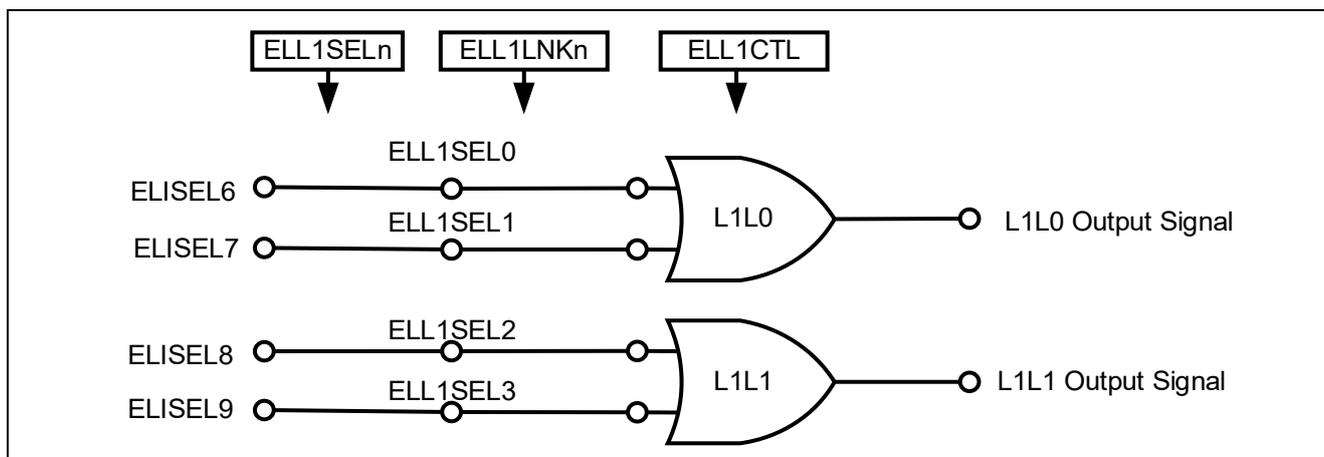


表 5-9 ELCL レジスタ設定 (論理セルブロック L2)

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	07H	ELISEL6 の選択信号を L2 のリンク対象に選択
ELL2SEL1	イベント・リンク L2 信号 選択レジスタ 1	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
ELL2SEL2	イベント・リンク L2 信号 選択レジスタ 2	0EH	論理セルブロック L1 の出力信号 1 を L2 のリンク対象に選択
ELL2SEL4	イベント・リンク L2 信号 選択レジスタ 4	03H	ELISEL8 の選択信号を 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	00H	選択なし (0 固定)
ELL2LNK0	イベント・リンク L2 出力 選択レジスタ 0	0AH	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 0 のセット制御にリンク
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
ELL2LNK4	イベント・リンク L2 出力 選択レジスタ 4	01H	ELL2SEL4 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 のセット制御にリンク
ELL2LNK6	イベント・リンク L2 出力 選択レジスタ 6	03H	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 0、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	C2H	論理セルブロック L2 フリップフロップ 0、1 使用許可、論理セル 0 OR 回路選択

図 5-7 論理セル L2 の設定

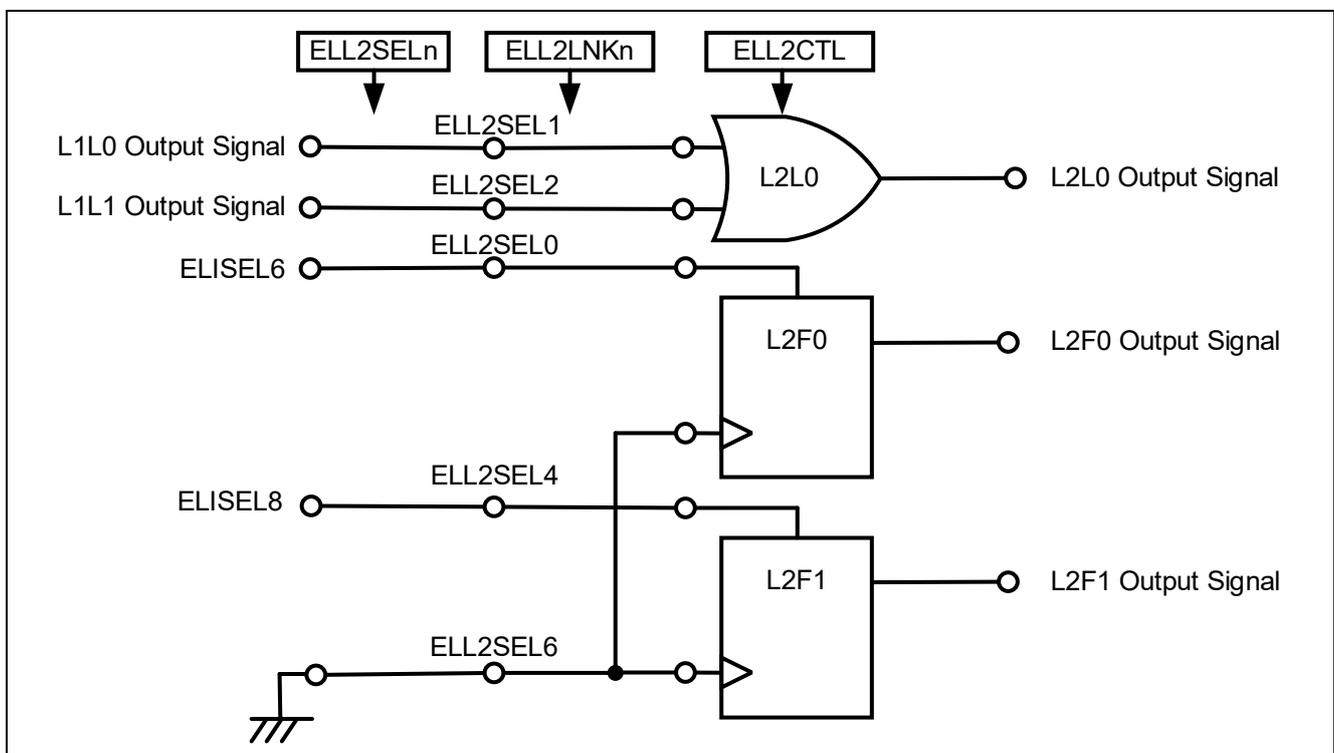


表 5-10 ELCL レジスタ設定 (論理セルブロック L3)

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	08H	ELISEL7 の選択信号を L3 のリンク対象に選択
ELL3SEL4	イベント・リンク L3 信号 選択レジスタ 4	04H	ELISEL9 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	00H	選択なし (0 固定)
ELL3LNK0	イベント・リンク L3 出力 選択レジスタ 0	0AH	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 のセット制御にリンク
ELL3LNK4	イベント・リンク L3 出力 選択レジスタ 4	01H	ELL3SEL4 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 のセット制御にリンク
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 のフリップフロップ 0、1 使用許可

図 5-8 論理セル L3 の設定

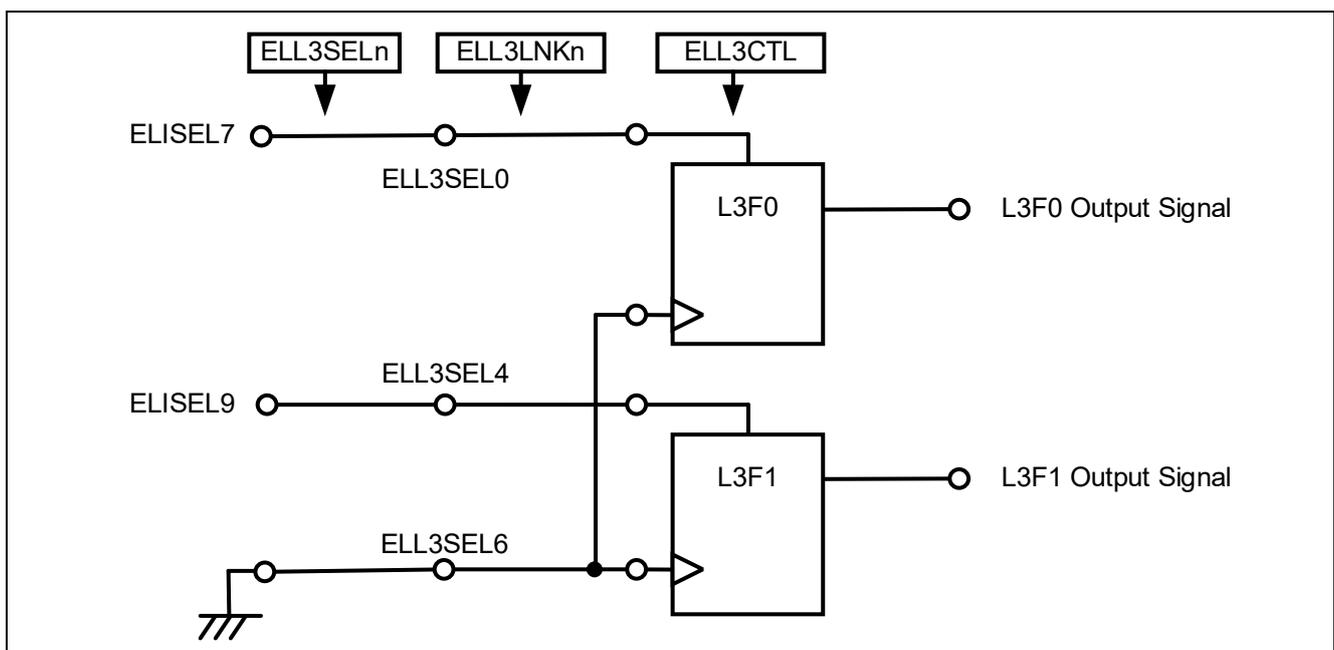
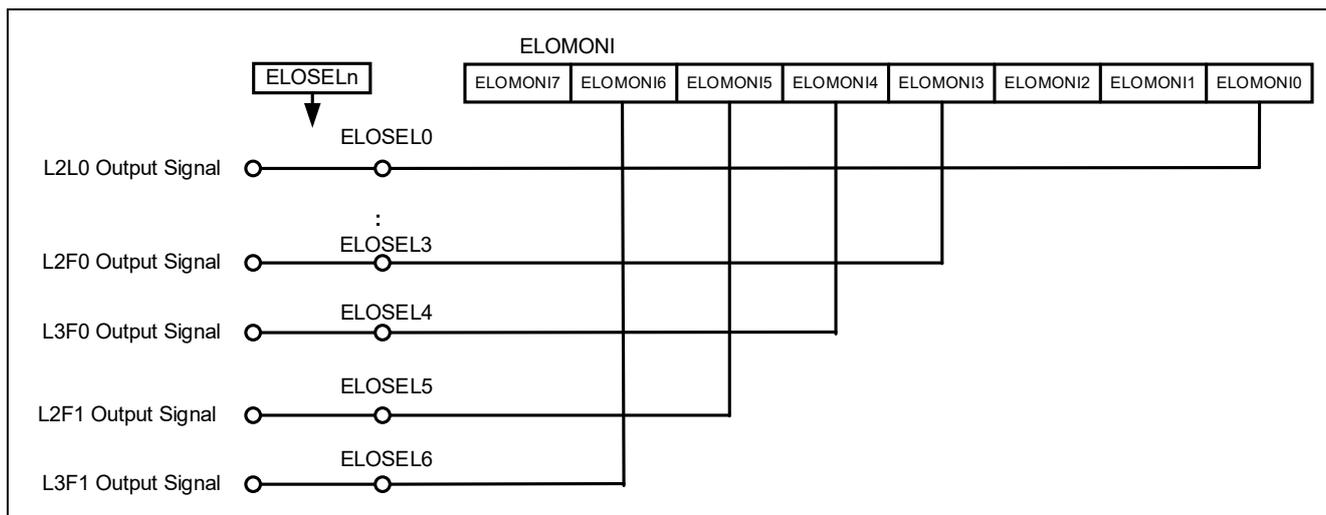


表 5-11 ELCL レジスタ設定 (出力)

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	06H	論理セルブロック L2 の出力信号[0]を選択
ELOSEL3	出力信号選択レジスタ 3	09H	論理セルブロック L2 の出力信号[3]を選択
ELOSEL4	出力信号選択レジスタ 4	0EH	論理セルブロック L3 の出力信号[3]を選択
ELOSEL5	出力信号選択レジスタ 5	0AH	論理セルブロック L2 の出力信号[4]を選択
ELOSEL6	出力信号選択レジスタ 6	0FH	論理セルブロック L3 の出力信号[4]を選択

図 5-9 ELCL の出力設定

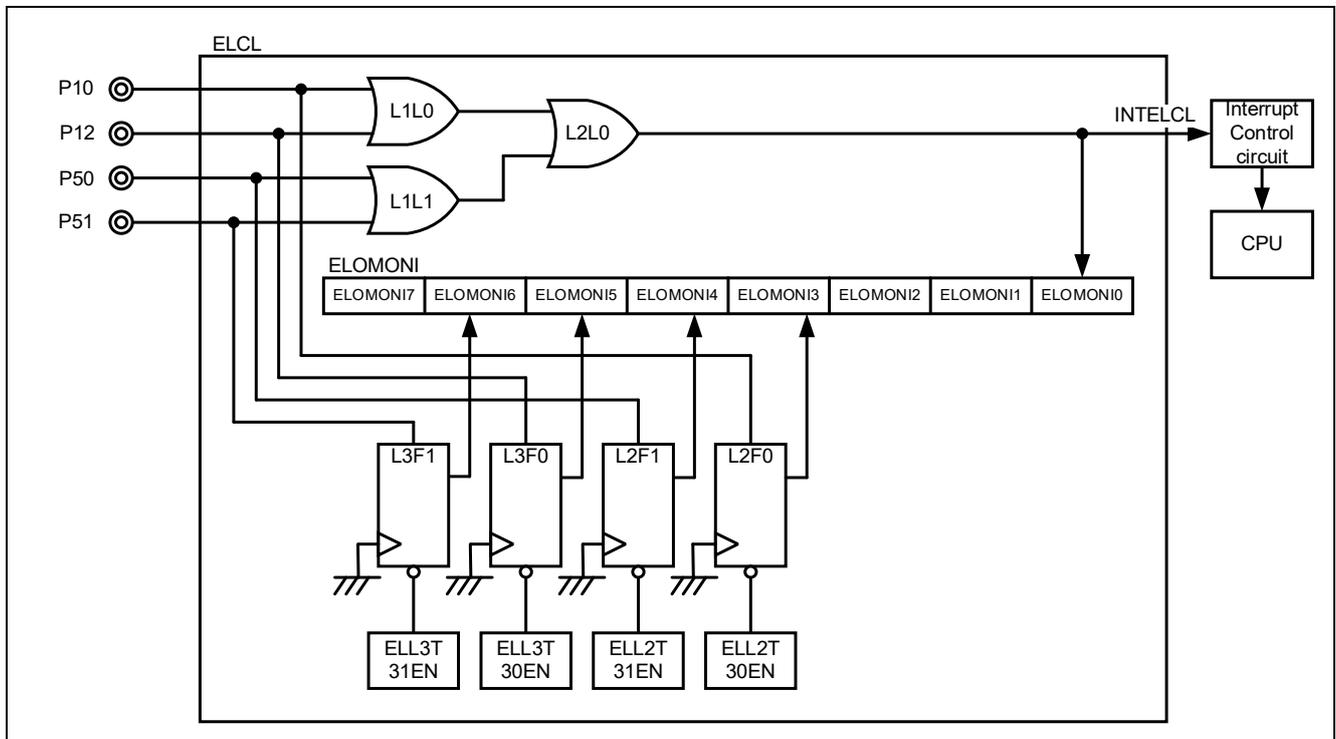


5.3.2 外部割り込み機能を持たない4つのポートを割り込みとして使用する例

外部割り込み機能を持たない4つのポートを割り込み（INTELCL）として使用することができます。

図 5-10 に構成例を示します。

図 5-10 構成例



本構成で実現する割り込み機能は、外部割り込み機能（INTPx）と以下の点で異なりますので、注意してください。

- INTELCL は、入力端子の OR 信号であり、OR された信号が“Low”→“High”に変更する時に割り込みが発生します。
- スマート・コンフィグレータからコードをそのまま出力すると入力端子の立ち上がりで割り込みが発生します。立ち下がりでの割り込みを発生させたい場合は、OR 回路に入力する際に負論理に変更してください。詳細は、「5.3.3 入力を負論理にする方法」を参照してください。
- 1つの入力に対して立ち上がり、立ち下がりと同時に設定することができません。いずれかを設定してください。

5.3.3 入力を負論理にする方法

コンポーネント「ELCL 複数パラメータ監視機能」は、4つの入力を正論理で OR します。サンプルコードを例に入力を負論理に変更する方法を説明します。

スマート・コンフィグレータで出力されたコードに対して、下記の通り、黄色で色づけした箇所を変更します。再度、スマート・コンフィグレータでプログラムを自動生成したときは、スマート・コンフィグレータの出力値（変更前）に上書きされます。自動生成する度に更新してください。

関数名 : R_Config_MultipleParameterMonitoring_Create	
変更前	変更後
<pre>void R_Config_MultipleParameterMonitoring_Create (void) { uint8_t p_isel0; uint8_t p_isel1; uint8_t p_isel2; uint8_t p_isel3; uint8_t p_isel2u; uint8_t p_isel3u; uint8_t p_osel0; uint8_t p_osel1; uint8_t p_osel2; ELISEL6=0x16U; p_isel0=0x07U; ELISEL7=0x16U; p_isel1=0x08U; ELISEL8=0x16U; p_isel2=0x09U; p_isel2u=0x03U; ELISEL9=0x16U; p_isel3=0x0AU; p_isel3u=0x04U; ELL1SEL0=p_isel0; ELL1SEL1=p_isel1; ELL1SEL2=p_isel2; ELL1SEL3=p_isel3; ELL1LNK0=0x01U; :</pre>	<pre>void R_Config_MultipleParameterMonitoring_Create (void) { uint8_t p_isel0; uint8_t p_isel1; uint8_t p_isel2; uint8_t p_isel3; uint8_t p_isel2u; uint8_t p_isel3u; uint8_t p_osel0; uint8_t p_osel1; uint8_t p_osel2; ELISEL6=0x16U; p_isel0=0x17U; ELISEL7=0x16U; p_isel1=0x18U; ELISEL8=0x16U; p_isel2=0x19U; p_isel2u=0x03U; ELISEL9=0x16U; p_isel3=0x1AU; p_isel3u=0x04U; ELL1SEL0=p_isel0; ELL1SEL1=p_isel1; ELL1SEL2=p_isel2; ELL1SEL3=p_isel3; ELL1LNK0=0x01U; :</pre>

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 : e² 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	6	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
		6	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 ² studio) : e ² studio 2021-01 (21.1.0) ⇒ 2022-01 (22.1.0) C コンパイラ (e ² 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 ボードサポートパッケージ (r_bsp) : V.1.00 ⇒ V.1.13
		6-7, 24	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 ² 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 1 の追加
		11-12	サンプルプログラム更新に伴い、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
		12	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
		13	表 4-6 サンプルコードで使用するグローバル変数 g_led_flag 内容 LED をトグルさせる。⇒ main 関数でポーリングし、LED をトグルさせる。
		13	コンポーネント「ELCL 複数パラメータ監視機能」更新に伴う更新 表 4-6 サンプルコードで使用するグローバル変数 g_led_flag、g_timer_mode 使用関数 r_elcl_interrupt ⇒ r_Config_MultipleParameterMonitoring_interrupt
		13	コンポーネント「ELCL 複数パラメータ監視機能」更新に伴い、表 4-7 関数一覧 を更新
14-15	サンプルプログラム更新に伴い、4.7 関数仕様 を更新		

Rev.	発行日	改訂内容	
		ページ	ポイント
2.00	Mar.24.22	16-23, 28	<p>コンポーネント「ELCL 複数パラメータ監視機能」更新に伴う更新</p> <p>図 4-4 メイン処理 関数名 : R_Config_MultipleParameterMonitoring_Create() ⇒ R_Config_MultipleParameterMonitoring_Start() ELCLIF = 0 ELCLMK = 0 ⇒ 削除</p> <p>図 4-10 ELCL 割り込み処理 関数名 : r_elcl_interrupt() ⇒ r_Config_MultipleParameterMonitoring_interrupt()</p> <p>図 4-9 ELCL 出力停止処理 図 4-8 ELCL 出力開始処理 フローチャートの削除</p> <p>図 4-10 ELCL 割り込み処理 図番号 : 図 4-10 ⇒ 図 4-8 図 4-11 ELCL のフリップフロップリセット処理 図番号 : 図 4-11 ⇒ 図 4-9 図 4-12 INTPO 割り込み処理 図番号 : 図 4-12 ⇒ 図 4-10 図 4-13 ウェイト処理 図番号 : 図 4-13 ⇒ 図 4-11 図 4-14 TAU0 チャンネル 7 割り込み処理 図番号 : 図 4-14 ⇒ 図 4-12</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 複数パラメータ監視機能」図の更新</p>
		23, 25, 27-28	<p>コンポーネント「ELCL 複数パラメータ監視機能」更新に伴う更新</p> <p>ELCL Multiple Parameter Monitoring Function ⇒ ELCL 複数パラメータ監視機能</p>
		24	<p>表 5-1 スマート・コンフィグレータの設定</p> <p>クロック : f_{sXL} ⇒ f_{sXP} コンポーネント Config_LVD0 リセット発生電圧 (V_{LVD0}) : 1.835 (V) ⇒ 1.86 (V)</p>

Rev.	発行日	改訂内容	
		ページ	ポイント
2.00	Mar.24.22	25	<p>コンポーネント「ELCL 複数パラメータ監視機能」更新に伴う更新</p> <p>コンポーネント : Config_MultipleParameterMonitoring</p> <p>Source A : INTTM00</p> <p>Source B : INTTM01</p> <p>Source C : INTTM02</p> <p>Source D : INTTM03</p> <p>Output signal selector : INTELCL</p> <p>⇒</p> <p>Source A : ELISEL_6 , INTTM00</p> <p>Source B : ELISEL_7 , INTTM01</p> <p>Source C : ELISEL_8 , INTTM02</p> <p>Source D : ELISEL_9 , INTTM03</p> <p>Output signal selector : 割り込み (INTELCL)</p>
		27	<p>コンポーネント「ELCL 複数パラメータ監視機能」更新に伴う更新</p> <p>5.2.8 Config_MultipleParameterMonitoring</p> <p>サンプルコードでは入力に TAU0 チャンネル 0、TAU0 チャンネル 1、TAU0 チャンネル 2、TAU0 チャンネル 3 を選択</p> <p>⇒</p> <p>サンプルコードでは入力に INTTM00、INTTM01、INTTM02、INTTM03 を選択</p>
		28	<p>コンポーネント「ELCL 複数パラメータ監視機能」更新に伴う更新</p> <p>表 5-3 コンポーネント「ELCL 複数パラメータ監視機能」の選択肢 項目にチェックボックスを追加</p> <p>Source A, B, C, D、Output E の選択肢を追加</p> <p>注 1、注 2 を追加</p>
		29-31	<p>コンポーネント「ELCL 複数パラメータ監視機能」更新に伴い追加</p> <p>表 5-4 入力要因の選択肢 (ELISEL0 ~ 11)</p> <p>表 5-5 入力要因の選択肢 (ELISEL0 ~ 5)</p> <p>表 5-6 入力要因の選択肢 (ELISEL6 ~ 11)</p>
		38	<p>コンポーネント「ELCL 複数パラメータ監視機能」更新に伴う更新</p> <p>5.3.3 入力を負論理にする方法</p> <p>現在のスマート・コンフィグレータ出力コードに合わせて更新</p>
		39	<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)</p> <p>RL78 スマート・コンフィグレータ ユーザーガイド : e² studio 編 (R20AN0579J)</p> <p>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、変更、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、変更、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。