

## RL78/ G10

### 簡易 IIC による IIC バス制御（割り込み編）

R01ANxxxxJJ0100

Rev. 1.00

2015.12.30

---

#### 要旨

本アプリケーションノートでは、シリアル・アレイ・ユニットの簡易 IIC 機能を用いた IIC バスの制御方法を示します。

#### 対象デバイス

RL78/G10

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あわせて変更し、十分評価してください。

## 目次

1. 仕様 .....	4
2. IIC バスの基本的な内容 .....	4
2.1 はじめに .....	4
2.1.1 通信とプロトコル .....	4
2.1.2 IIC バスの仕様のとらえ方 .....	5
2.1.3 IIC バスのプロトコルのとらえ方 .....	5
2.2 使用する LCD 表示モジュールのプロトコル .....	6
3. 動作確認条件 .....	7
4. 関連アプリケーションノート .....	7
5. ハードウェア説明 .....	8
5.1 ハードウェア構成例 .....	8
5.2 使用端子一覧 .....	8
6. ソフトウェア説明 .....	8
6.1 動作概要 .....	8
6.2 コード生成での設定内容 .....	9
6.3 オプション・バイトに反映される設定一覧 .....	13
6.4 定数一覧 .....	13
6.5 変数一覧 .....	14
6.6 関数一覧 .....	15
6.7 関数仕様 .....	15
6.8 フローチャート .....	20
6.8.1 初期設定関数 .....	20
6.8.2 システム関数 .....	21
6.8.3 CPU クロックの設定 .....	22
6.8.4 入出力ポート設定 .....	22
6.8.5 SAU0 の設定 .....	23
6.8.6 IIC00 の設定 .....	25
6.8.7 タイマ・アレイ・ユニットの設定 .....	31
6.8.8 WDT の設定 .....	37
6.8.9 メイン処理 .....	38
6.8.10 R_MAIN_UserInit 処理 .....	39
6.8.11 500ms 待ち .....	40
6.8.12 100ms 待ち .....	40
6.8.13 LCD モジュールの初期設定 .....	41
6.8.14 LCD モジュールへのコマンド送信 .....	42
6.8.15 LCD モジュールへのデータ送信 .....	43
6.8.16 WDT 割り込み処理 .....	44
6.8.17 IIC00 と TM01 の起動準備 .....	45
6.8.18 データ送信起動処理 .....	47
6.8.19 データ受信起動処理 .....	48
6.8.20 IIC00 送信完了待ち .....	49
6.8.21 IIC00 状態確認 .....	49
6.8.22 ストップ・コンディション発行処理 .....	50
図 6.23 にストップ・コンディション発行処理のフローチャートを示します。 .....	50
6.8.23 コマンド処理待ち .....	52
6.8.24 20us 待ち .....	52
6.8.25 スタート・コンディション発行処理 .....	54
6.8.26 IIC00 動作停止処理 .....	57
6.8.27 5us 待ち処理 .....	58
図 6.28 に 5us 待ち処理のフローチャートを示します。 .....	58

6.8.28 IIC00 割り込み処理 .....	59
図 6.29 に IIC00 割り込み処理のフローを示します。 .....	59
6.8.29 TM01 割り込み処理.....	61
7. CA-78K0R への移植について .....	66
7.1 追加修正するファイル .....	66
7.1.1 準備作業 .....	66
7.1.2 ファイルのコピー.....	66
7.1.3 コピーしたファイルのプロジェクトへの追加 .....	66
7.1.4 “r_IIC_LIB.c”の修正 .....	66
7.2 r_cg_main.c の変更 .....	66
7.2.1 CC-RL プロジェクトからのコピー .....	66
7.2.2 インクルード・ファイルの追加 .....	67
7.3 ビルド.....	67
8. サンプルコード .....	68
9. 参考ドキュメント.....	68

## 1. 仕様

本アプリケーションノートでは、IIC バスを使用する上で知っていたほうがよいと思われる基本的な項目および、RL78 シリーズに内蔵されているシリアル・アレイ・ユニットの簡易 IIC 機能を用いた IIC バスで接続した周辺機能の制御方法を示します。

実際に IIC バスの物理層を制御する処理関数部は"r\_IIC\_LIB.c"として独立させています。メインの処理を行う r\_cg\_main.c からは、"r\_IIC\_LIB.c"で提供されている関数を使用して IIC バスを使用します。"r\_IIC\_LIB.c"部分とコード生成での内蔵周辺機能の初期設定を変更することで、IICA への置き換えができることを考慮しています。



なお、簡易 IIC の制御には割り込み（ベクタ割り込み）を使用します。

今回は、16 文字×2 行を表示する LCD 表示モジュールを IIC バスで制御します。

## 2. IIC バスの基本的な内容

### 2.1 はじめに

IIC バスはいろいろな制約があって、使い方が難しそうだという声をよく耳にします。これは、通信に対して、送りたいときに送信したら相手は勝手に受信してくれ、データが欲しいときには通信相手が都合よく送ってきてくれると考えているからではないかと思われる。これは通信では「無手順」と呼ばれていて、古くはマイコンとテレタイプや CRT ターミナルのような端末との通信で一般に使われていました。

#### 2.1.1 通信とプロトコル

しかし、マイコンと周辺機能を内蔵した外部の IC または他のマイコンを接続して通信を行う場合には、これではきちんとした通信とは言えません。

きちんとした通信は、通信相手の状態を考慮しながら、決まったルール（プロトコル）に基づいてデータのやり取りを行うものです。そのため、前もってルールを明確にしておき、そのルールに従って通信を行う必要があります。

その点で、IIC バスでは通信の基本的なプロトコルが規定されているので、それに従うだけで済むと考えるべきです。

他の通信方法、例えば 3 線式シリアル通信（CSI）でもプロトコルは必要です。CSI では、マスタが通信を始める前にスレーブの通信準備ができていないと、正常な通信は行えません。これを満足させるには、スレーブからビジー状態であることを示す信号をマスタに戻す必要があります。（このためには、3 線式ではなく、4 線式になってしまいます。）このようなことをきちんと対応しないと、ビットずれが発生してしまい、この状態が CSI のインタフェースを初期化するまで続いてしまいます。この制御は厳密に考えると結構面倒な処理が必要になります。ここらは、RL78/G13 の CSI 通信のアプリケーションノートに例が記載されています。

実際に、マイコンが CSI（または SPI）を用いて制御するような周辺はかなりのものが EEPROM や D/A と言った単機能のハードウェアで、スレーブは常に通信が可能な状態になっているので、マイコン側では通信の手順を特に意識しなくても済んでいるだけなのです。

なお、EEPROM では単純なデータのやり取りだけでは制御できませんが、それは通信ではなく EEPROM の制御方法と考えられているようです。

一方、IIC バスはスレーブが準備できているかをチェックする手段が組み込まれていて、他に信号を追加することなく、2本の信号線だけで通信が可能です。

もうひとつの一般的な通信方式である調歩同期通信ですが、UART 機能を用いて比較的簡単に通信を行えるので、初心者に好まれているようです。上記の端末との通信方式はほとんどがこの調歩同期通信による無手順通信です。しかし、調歩同期通信でもきちんとした通信を行うには、それなりのプロトコルが必要になります（実際に、ほとんどの端末はフロー制御をサポートしています）。UART のレジスタを見ると、3つのエラー・フラグがあります。つまり、これは使用者がそのようなエラーに対する処理を行う必要があることを表しています。しかも、このエラー・フラグは当然ながら受信側にしかありません。そのために、受信側でエラーが発生したことをデータの送信側に連絡する必要があります。ここで良く用いられるのが再送要求です。データの送信に対して、受信側は正しくデータを受け取ったことを示すために、送信側に ACK 応答を行います。データが正しく受け取れなかったときには NACK 応答を行って、送信側に再送を要求します。これらの処理を使用者がソフトで実現する（場合によっては通信用のプロトコル・スタックを利用する）必要があります。

IIC バスには受信側から送信側に受信可能であることを通知する ACK 応答機能が組み込まれているので、この点では使用者が頭を悩ます必要はありません。

IIC バスでは、CAN や LIN のように上位階層までプロトコルが決められていない分、ある程度自由に使うことが可能な通信手段です。だからこそ、ボード内でのいろんな周辺機能との通信に用いられています。

### 2.1.2 IIC バスの仕様のとらえ方

IIC バスを難しくしている要因のひとつにマルチマスタ機能があります。しかし、簡易 IIC では、マルチマスタには対応していないので、難しいことを検討する必要はありません。

### 2.1.3 IIC バスのプロトコルのとらえ方

IIC バスの通信は全てマスタが管理します。スレーブが独自にできることは、NACK 応答をすることだけです。（簡易 IIC ではスレーブからウェイトをかけることはできません。）それ以外はマスタからの指示に従って通信を行う必要があります。ただし、IIC バスでは、基本的なプロトコルだけを規定していて、上位のプロトコルは規定していないので、そこはユーザが規定する必要があります。つまり、この部分にスレーブの仕様を反映する、スレーブとしてそのような場合にどうするかを定義することになります。

#### (a) IIC バスの仕様として決まっていること

仕様書には「通常、アドレス指定されたレシーバは、（省略）、各バイトが終了するたびにアクノリッジを生成しなければなりません。

スレーブ・レシーバがアドレス確認を行なうことができない場合（例えば、リアルタイム機能を実行しているために受信できないような場合）、そのスレーブはデータ・ラインを“H”の状態に保持しなければなりません。（省略）スレーブがデータを受信することができない場合、スレーブは次に送られてくる最初のバイトに対してアクノリッジを生成しないことによってそのことを示します。（省略）マスタがレシーバとなる場合、スレーブから送信された最後のデータ・バイトに対してアクノリッジをしないことによって、マスタはスレーブ・トランスミッタにデータの終わりを知らせます」と記載されています。

つまり、「マスタ受信時の NACK 応答はエラーではなく通信の終了」で、「スレーブからの NACK 応答は通信を行えない状況にある」ことを表します。決して、スレーブからの NACK 応答は受信データにエラーがあったことを意味してはいません。

#### (b) コントローラで決まっていること

RL78 に搭載されている簡易 IIC 機能については、ハードウェアマニュアルを参照してください。上でもふれたように、簡易 IIC では、スレーブがマスタ（RL78）にウェイトをかけることができません。

#### (c) ユーザが定義すること

例えば、IIC バスの仕様書の「7ビット・アドレスのフォーマット」には「以前にアクセスされたメモリー・ロケーションの自動加算または自動減算についての決定は、全てデバイスの設計者が行ないます。」と明記されていますし、これ以外にも定義しておいた方がよいことがあります。

例えば、RL78/G13 の IIC バスのアプリケーションノートの例のように、16 バイトのデータしか扱わないのに 17 バイト目のデータの読み出し指示が来た場合にどうするかです。

簡易 IIC にはスレーブ機能はないので、詳細は割愛します。

このように、IIC バスのプロトコルに従って通信を行えば、IIC バスのスレーブとして十分です。このような IIC バスの手順を満足した以外に、ローカルな通信であれば、さらに IIC バスの仕様の一部をサポートしないことも考えられます。

IIC バスのマスタはそのような仕様のスレーブの制御を実現するためにはどうすればよいか検討すればよいわけです（細かいところはマスタの指示（=IIC バスのプロトコル）に従うが、大きな動作はスレーブの仕様に合わせるという訳です）。

つまりは、最初にスレーブの仕様を明確化するのが重要です。特に、簡易 IIC ではスレーブの処理時間を待つ機能がないので、制御プログラム側で適当な待ち合わせ時間を確保するような処理が必須です。

そのため、ここでは簡易 IIC の転送完了割り込みでタイマを起動して、一つの通信から次の通信までに時間を置くようにしています。

## 2.2 使用する LCD 表示モジュールのプロトコル

最初に、IIC バスでの通信で大きな影響があるスレーブとして制御する LCD 表示モジュールの仕様を明確にします。このモジュールの特徴としては、受信機能だけをサポートしていることと、通信形式が固定されていること、および通信と通信の間隔を長くとることが必要です。

対象デバイス : ACM1602NI-FLW-FBW-M01 (DISPLAYTRONIC)

表示仕様 : アルファベット&カナを 16 文字×2 行

通信 I/F : IIC バス。通信速度 85kbps（コード生成での指定値）。受信機能のみ

アドレス : 0xA0

通信形式 : ST + スレーブ・アドレス + コントロールバイト + コマンド/データ + SP のみ  
(ST : スタート・コンディション、SP : ストップ・コンディション)

コントロールバイト : 0x00 なら次はコマンド、0x80 なら次はデータ

起動時間 : 最低 15ms（電源供給後通信を開始するまでの時間）

通信間隔 : 最低 5ms

### 3. 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 3.1 動作確認条件

項目	内容
使用マイコン	RL78/G10 (R5F10Y16)
動作周波数	● 高速オンチップ・オシレータ (HOCO) クロック : 24MHz ● CPU/周辺ハードウェア・クロック : 24MHz
動作電圧	3.3V (2.9V~3.6V で動作可能) セレクトابل・パワーオン・リセット電圧 : 2.75V
統合開発環境	ルネサス エレクトロニクス製 CS+ V3.01.00
アセンブラ	ルネサス エレクトロニクス製 CC-RL V1.01.00
使用ボード	RL78/G10 ターゲット・ボード (QB-R5F10Y16-TB) (+ LCD 表示モジュール (ACM1602NI-FLW-FBW-M01) )

### 4. 関連アプリケーションノート

本アプリケーションノートに関連するアプリケーションノートを以下に示します。併せて参照してください。

RL78/G10 初期設定 CC-RL (R01AN2668J) アプリケーションノート

5. ハードウェア説明

5.1 ハードウェア構成例

図 5.1 に本アプリケーションノートで使用するハードウェア構成例を示します。

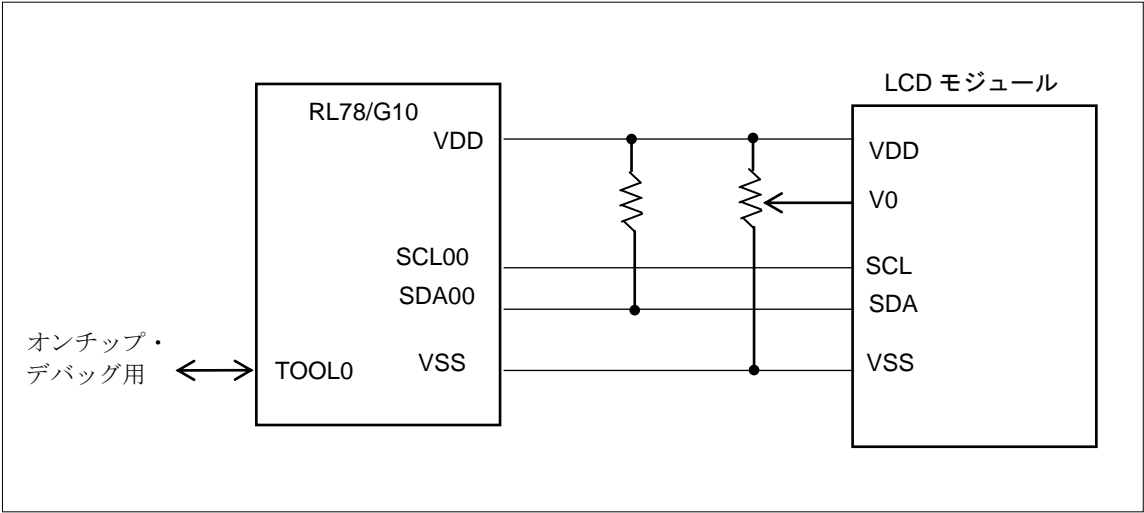


図 5.1 ハードウェア構成

- 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 VDD 又は Vss に接続して下さい）。
- 2 VDD はセレクトブル・パワーオン・リセット電圧（V<sub>SPOR</sub>）以上にしてください。

5.2 使用端子一覧

表 5.1 表 5.4 に使用端子と機能使用端子と機能を示します。

表 5.1 使用端子と機能

端子名	入出力	内容
SDA00	入出力	IIC 通信データ信号
SCL00	出力	IIC 通信クロック信号

6. ソフトウェア説明

6.1 動作概要

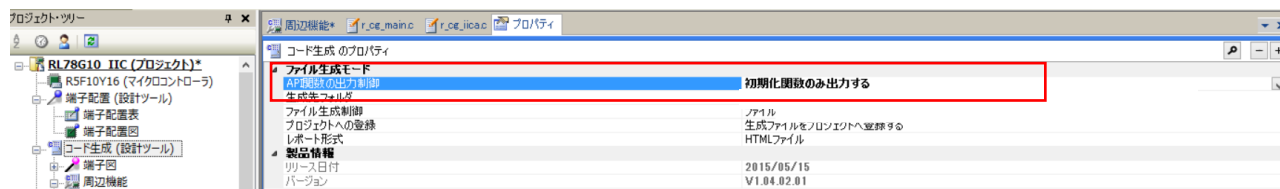
本アプリケーションノートでは、内蔵周辺機能の初期設定だけは CS+のコード生成機能を利用します。

内蔵周辺機能の初期設定が完了したら、12 ビット・インターバル・タイマで 0.5s 待ちます。その後、IICA0 を使用して IIC バスで接続した LCD 表示モジュールを初期設定し、初期データを表示させます。その後、0.5s 待ってから 0.1s ごとに表示データを 1 キャラクタずつ右端からシフト・インしていきます。16 キャラクタ全てが表示されたら、前に戻って、表示を繰り返します。



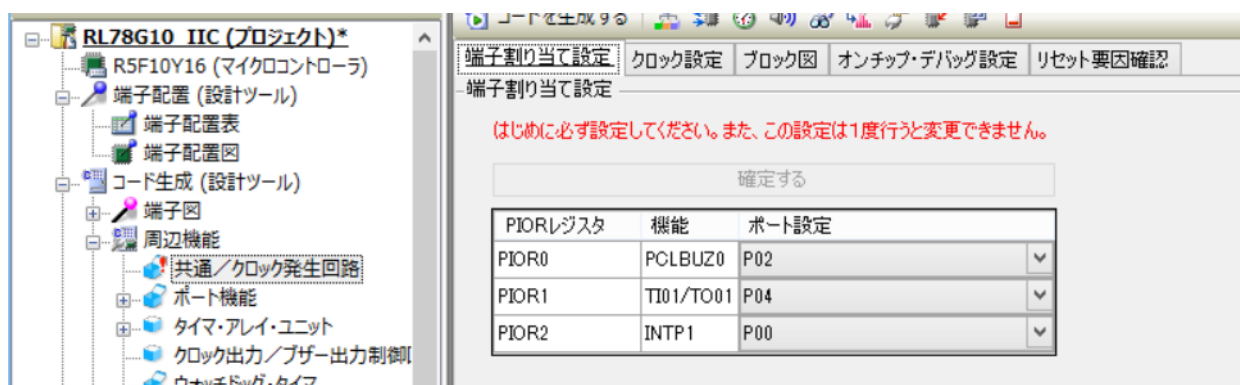
## 6.2 コード生成での設定内容

プロパティの「ファイル生成モード」の「API 関数の出力制御」を「初期化関数のみ出力する」に設定し、以下の設定を行います。



### (1) クロック発生回路の設定

#### (a) 端子割り当て設定：そのまま確定



#### (b) クロック設定

- ・動作モード設定：高速メイン・モード  $2.7(V) \leq VDD \leq 5.5(V)$
- ・メイン・システム・クロック (fMAIN) 設定：高速オンチップオシレータクロック (fIH)
- ・高速オンチップオシレータクロック設定：20 (MHz)
- ・高速システム・クロック設定：動作をチェックしない (16 ピン製品のみ)
- ・低速内蔵発振クロック (fIL) 設定：周波数 15 (kHz)
- ・CPU と周辺クロック設定：20000 (fIM) (kHz)
- ・RESET 端子設定：使用する (P125)



## (c) オンチップ・デバッグ設定

- ・オンチップ・デバッグ動作設定：使用する
- ・RRM 機能設定：使用しない
- ・セキュリティ ID 設定：セキュリティ ID を設定する

端子割り当て設定	クロック設定	ブロック図	オンチップ・デバッグ設定	リセット要因確認
- オンチップ・デバッグ動作設定 -				
<input type="radio"/> 使用しない		<input checked="" type="radio"/> 使用する		
- RRM/DMM機能設定 -				
<input checked="" type="radio"/> 使用しない		<input type="radio"/> 使用する		
- セキュリティID設定 -				
<input checked="" type="checkbox"/> セキュリティIDを設定する				
セキュリティID		0x00000000000000000000		

## (d) リセット要因確認

- ・リセット要因を確認する関数を入力する：チェックを「外す

端子割り当て設定	クロック設定	ブロック図	オンチップ・デバッグ設定	リセット要因確認
- 関数出力設定 -				
<input type="checkbox"/> リセット要因を確認する関数を入力する				

## (2) ポートの設定

全て初期値（使用しない）のまま

## (3) タイマの設定

## (a) 一般設定 チャンネル 1：ディレイカウント機能

一般設定	チャンネル0	チャンネル1
- 機能 -		
チャンネル 0	使用しない	
チャンネル 1	ディレイ・カウント機能	

## (b) チャンネル 1

- ・動作モード設定：16 ビット
  - ・ディレイカウント時間設定：5  $\mu$ s
- これ以外は初期状態のまま

一般設定	チャンネル0	チャンネル1
- 動作モード設定 -		
<input checked="" type="radio"/> 16ビット		<input type="radio"/> 下位8ビット
- ディレイ・カウント時間設定 -		
<input type="checkbox"/> TI01端子入力信号のノイズ・フィルタ使用		
外部イベント・エッジ選択(TI01)		立下りエッジ
ディレイ		5 $\mu$ s (実際の値: 5)
<input type="checkbox"/> 動作中のスタート・トリガを有効とする		
- 割り込み設定 -		
<input checked="" type="checkbox"/> タイマ・チャンネル1のカウント完了で割り込み発生(INTTM01)		
優先順位		低

## (4) クロック出力／ブザー出力の設定

初期値（使用しない）のまま

#### (5) ウォッチドッグ・タイマの設定

- ・ウォッチドッグ・タイマ動作設定：使用する
- ・HALT/STOP モード時の動作設定：許可
- ・オーバフロー時間： $(2^{11}-1)/f_{IL}$ （136.47ms）
- ・割り込み設定：75%割り込みを発生する

ウォッチドッグ・タイマ動作設定	
<input type="radio"/> 使用しない	<input checked="" type="radio"/> 使用する
HALT/STOPモード時の動作設定	
<input checked="" type="radio"/> 許可	<input type="radio"/> 停止
オーバフロー時間設定	
オーバフロー時間	$(2^{11}-1)/f_{IL}$ 136.47 (ms)
割り込み設定	
<input checked="" type="checkbox"/> オーバフロー時間の75% + $3/(f_{IL} \times 4)$ 到達時にインターバル割り込みを発生する(INTWDTI)	
優先順位	低

#### (6) A/D コンバータの設定

初期値（使用しない）のまま

#### (7) シリアルの設定

(a)チャンネル チャンネル 0：IIC00 に設定

チャンネル	UART0	CSI00	IIC00
機能			
チャンネル0	IIC00		
チャンネル1	使用しない		

#### (b)IIC00 の設定

- ・転送レート：85000(bps)を設定

その他は初期状態のまま。

チャンネル	UART0	CSI00	IIC00
転送レート設定			
転送レート	85000 (bps) (実際の値: 84745.763)		
割り込み設定			
転送完了割り込み優先順位(INTIIC00)	低		
コールバック機能設定			
<input checked="" type="checkbox"/> マスタ送信完了	<input checked="" type="checkbox"/> マスタ受信完了	<input checked="" type="checkbox"/> マスタエラー	

#### (8) 割り込みの設定

全て初期値（チェックなし）のまま

#### (9) キー割り込み機能

全て初期値（チェックなし）のまま

#### (10) セレクトابل・パワーオン・リセットの設定

- ・リセット発生電圧：2.84（V）

動作モード設定	
リセット発生電圧(VSPDR)	2.84 (V)

### 6.3 オプション・バイトに反映される設定一覧

表 6.1 にオプション・バイトに反映される値の例を示します。

表 6.1 オプション・バイト反映値

アドレス	設定値	内容
0x000C0	0b11111001	ウォッチドッグ・タイマ オーバフロー時間：2 <sup>11</sup> -1 HALT/STOP モード時、カウンタ動作許可
0x000C1	0b11110111	2.84V (2.70V~2.96V)、RESET 端子
0x000C2	0b11111001	HOCO :20MHz
0x000C3	0b10000101	オンチップ・デバッグ許可

### 6.4 定数一覧

表 6.2 に サンプルコードで使用する定数 ~~サンプルコードで使用する定数~~ を示します。

表 6.2 サンプルコードで使用する定数

定数名	設定値	内容
slaveaddr	0xA0	スレーブのアドレス
COMBYTE	0x00	LCD モジュールのコマンド指定コード
DATABYTE	0x80	LCD モジュールのデータ指定コード
CLRDISP	0x01	表示クリア命令
HOMEPOSI	0x02	ホーム・ポジション命令
LCD_Mode	0b00111000	動作モード命令
DISPON	0b00001100	表示オン命令（カーソル非表示）
ENTRY_Mode	0b00000110	入力でのカーソル右移動命令
dispdata1	"Renesas RL78/G10 "	LCD に表示するデータ

## 6.5 変数一覧

表 6.3 にサンプルコードで使用する変数一覧を示します。g\_iic00\_status 以下の変数は、IIC00 のハードウェア制御用ライブラリで使用する変数です。

表 6.3 サンプル コードで 使用する 変数 Type	Variable Name	Contents	Function Used
uint8_t	g_write_commmmand	コマンド送信用バッファ	set_command()
uint8_t	g_write_data	表示データ送信用バッファ	set_data()
uint8_t	g_05s_status	0.5 秒経過フラグ	R_MAIN_UserInit() wait_100ms() wait_500ms() r_it_interrupt()
uint8_t	g_05s_count	0.5 秒経過確認用カウンタ	R_MAIN_UserInit() wait_100ms() r_it_interrupt()
uint8_t	g_iic00_status	IIC00 の動作状態を格納	R_IIC_Master_Send R_IIC_Master_Receive R_IIC_check_comstate R_IIC_wait_comend() r_inttm03_interrupt
uint8_t	gp_iic00_rx_address	受信データ格納用ポインタ	R_IIC_Master_Receive() r_inttm03_interrupt()
uint16_t	g_iic00_rx_len	受信するデータ数	R_IIC_Master_Receive() r_inttm03_interrupt()
uint16_t	g_iic00_rx_cnt	受信したデータ数	R_IIC_Master_Receive() r_inttm03_interrupt()
uint16_t	gp_iic00_tx_address	送信データ読み出しポインタ	R_IIC_Master_Send() r_inttm03_interrupt()
uint16_t	g_iic00_tx_cnt	残り送信データ数	R_IIC_Master_Send() r_inttm03_interrupt()
uint8_t	g_TM03flag	20 $\mu$ 秒計測中フラグ	r_iic00_interrupt() r_inttm03_interrupt()
uint8_t	g_TM03cnt	5 $\mu$ 秒のカウンタ	r_inttm03_interrupt()

## 6.6 関数一覧

表 6.4 に IIC バス制御で使用する関数一覧を示します。関数 R\_IIC\_Master\_Send 以下は IIC00 のハードウェア制御用ライブラリの関数です。

表 6.4 関数一覧

関数名	概要
init_LCD	LCD モジュールの初期設定を行う
set_data	LCD モジュールに表示データを送信する
set_command	LCD モジュールにコマンドを送信する
wait_100ms	表示データ更新用に 100ms 待つ
wait_500ms	次の表示タイミング用に 500ms 待つ
r_it_interrupt	WDTI でおおよそ 100ms のインターバル割り込み処理を行う
IIC_TM03_init	IIC00 と TM01 の起動準備を行う
R_IIC_Master_Send	指定したデータのスレーブへの送信を起動する
R_IIC_Master_Receive	スレーブからのデータ受信を起動する
R_IIC_wait_comend	IIC バスでの通信完了を待つ
R_IIC_check_comstate	IIC バスの通信状態を確認する
R_IIC_StopCondition	IIC バスにストップ・コンディションを発行する
wait_time	LCD モジュールのコマンド処理時間を待つ
wait_20us	LCD モジュールのタイミング調整で 20us 待つ
R_IIC00_Stop	IIC00 の動作を停止する
R_IIC00_start_condition	IIC バスにスタート・コンディションを発行する
wait_5us	IIC バスの制御で 5us 待つ
r_iic00_interrupt	IIC00 の転送完了 TM01 を起動する
r_inttm01_interrupt	IIC00 の転送完了処理を行う

## 6.7 関数仕様

サンプルコードの関数仕様を示します。

### [関数名] init\_LCD

概要	LCD モジュールの初期設定処理
ヘッダ	
宣言	void init_LCD(void);
説明	LCD モジュールを初期化し、表示モードやデータ入力モードを設定します。 (表示位置自動更新、カーソル表示なし)
引数	なし
リターン値	なし
備考	なし

### [関数名] set\_data

概要	LCD モジュールへの表示データ送信処理
ヘッダ	r_cg_macrodriver.h
宣言	MD_STATUS set_data(uint8_t datacode);
説明	LCD モジュールに表示データを送信します。
引数	表示データ
リターン値	[MD_OK] の場合：データ送信完了 それ以外の場合：データ送信失敗
備考	なし

## [関数名] set\_command

---

概要	LCD モジュールへのコマンド送信処理
ヘッダ	r_cg_macrodriver.h
宣言	MD_STATUS set_command(uint8_t comcode);
説明	LCD モジュールに動作モードを指定するコマンドを送信します。
引数	コマンド
リターン値	[MD_OK]の場合：コマンド送信完了 それ以外の場合：コマンド送信失敗
備考	なし
備考	なし

## [関数名] wait\_100ms

---

概要	100ms 待ち
ヘッダ	
宣言	void wait_100ms(void);
説明	インターバル・タイマを用いて 100ms の時間を待ちます。
引数	なし
リターン値	なし
備考	なし

## [関数名] wait\_500ms

---

概要	500ms 待ち
ヘッダ	
宣言	void wait_500ms(void);
説明	インターバル・タイマを用いて 500ms の時間を待ちます。
引数	なし
リターン値	なし
備考	なし

## [関数名] r\_it\_interrupt

---

概要	100ms のインターバル・タイマ割り込み処理
ヘッダ	r_cg_macrodriver.h
宣言	#pragma interrupt INTWDTI r_it_interrupt rb2 __interrupt static void r_it_interrupt(void)
説明	ウォッチドッグ・タイマの 75%割り込みでの 100ms 程度の割り込み（INTWDTI）で起動され、変数 g_05s_count をカウントして 500ms を待ちます。 500ms 経過したら、変数 g_05s_count をクリアし、変数 g_05s_status に 1 をセットします。
引数	なし
リターン値	なし
備考	なし



以下の関数は、IIC00 のハードウェア制御用ライブラリで提供されている関数です。

[関数名] IIC\_TM01\_init

概要	IIC00 と TM01 を起動する
ヘッダ	r_cg_macrodriver.h、r_IIC_LIB.h
宣言	void IIC_TM01_init(void);
説明	IIC00 の割り込みを許可し、TM03 をトリガ待ちに設定します。
引数	なし
リターン値	なし
備考	なし

[関数名] R\_IIC\_Master\_Send

概要	指定したデータのスレーブへの送信を起動	
ヘッダ	r_cg_macrodriver.h、r_IIC_LIB.h	
宣言	MD_STATUS R_IIC_Master_Send(uint8_t adr, uint8_t * const tx_buf, uint16_t tx_num);	
説明	IIC バスの状態を確認し、スタート・コンディションを発行し、スレーブ・アドレスを送信します。	
引数	第 1 引数	スレーブ・アドレス
	第 2 引数	送信データのポインタ
	第 3 引数	送信するデータ数
リターン値	MD_OK	
備考	なし	

[関数名] R\_IIC\_Master\_Receive

概要	スレーブからのデータ受信を起動	
ヘッダ	r_cg_macrodriver.h、r_IIC_LIB.h	
宣言	MD_STATUS R_IIC_Master_Receive(uint8_t adr, uint8_t * const rx_buf, uint16_t rx_num);	
説明	IIC バスの状態を確認し、スタート・コンディションを発行し、スレーブ・アドレスを送信します。	
引数	第 1 引数	スレーブ・アドレス
	第 2 引数	受信データのポインタ
	第 3 引数	受信するデータ数
リターン値	MD_OK	
備考	なし	

[関数名] R\_IIC\_wait\_comend

概要	IC バスでの通信完了待ち
ヘッダ	r_cg_macrodriver.h、r_IIC_LIB.h
宣言	MD_STATUS R_IIC_wait_comend(void);
説明	IIC バスでの通信完了を待ち、通信結果を戻します。
引数	なし
リターン値	[MD_OK] の場合：正常に処理を完了 それ以外の場合：処理エラー
備考	なし

## [関数名] R\_IIC\_check\_comstate

---

概要	IIC バスでの通信完了待ち
ヘッダ	r_cg_macrodriver.h、r_IIC_LIB.h
宣言	MD_STATUS R_IIC_check_comstate(void);
説明	IIC バスでの通信状態を戻します。
引数	なし
リターン値	0x00 : 正常に処理を完了 0x01 : マスタ送信処理中 0x41 : マスタ受信処理中 0x81 : マスタ受信のためのスレーブ・アドレス送信中
備考	なし

## [関数名] R\_IIC\_StopCondition

---

概要	IIC バスにストップ・コンディションを発行
ヘッダ	r_cg_macrodriver.h
宣言	MD_STATUS R_IIC_StopCondition(void);
説明	ストップ・コンディションを発行します。
引数	なし
リターン値	MD_OK
備考	なし

## [関数名] wait\_time

---

概要	LCD モジュールのコマンド処理時間待ち
ヘッダ	r_cg_macrodriver.h
宣言	void wait_time(uint8_t time);
説明	LCD モジュールに送信したコマンドの処理時間待ちで使します。
引数	待ち時間（20us 単位）
リターン値	なし
備考	なし

## [関数名] wait\_20us

---

概要	20us 待ち
ヘッダ	
宣言	void wait_20us(void);
説明	TM03 のディレイカウント機能を用いて 20us の時間を待ちます。
引数	なし
リターン値	なし
備考	なし

## [関数名] R\_IIC00\_start\_condition

---

概要	IIC バスにスタート・コンディションを発行
ヘッダ	r_cg_macrodriver.h
宣言	void R_IIC00_start_condition (void);
説明	スタート・コンディションを発行します。
引数	なし
リターン値	なし
備考	なし

## [関数名] R\_IIC00\_Stop

---

概要	IIC00 の動作を停止
ヘッダ	r_cg_macrodriver.h
宣言	void R_IIC00_Stop (void);
説明	スタート・コンディションやストップ・コンディション発行のために、IIC00 の動作を停止します。
引数	なし
リターン値	なし
備考	なし

---

## [関数名] wait\_5us

---

概要	5us 待ち
ヘッダ	
宣言	void wait_20us(void);
説明	TM01 のディレイカウント機能を用いて 5us の時間を待ちます。
引数	なし
リターン値	なし
備考	なし

---

## [関数名] r\_iic00\_interrupt

---

概要	IIC00 の転送完了処理を行う
ヘッダ	r_cg_macrodriver.h
宣言	#pragma interrupt INTIIC00 r_iic00_interrupt __interrupt static void r_iic00_interrupt(void)
説明	IIC00 の通信完了割り込み（INTIIC00）で起動され、スレーブでの処理時間確保のための時間待ち（TM01）を起動します。
引数	なし
リターン値	なし
備考	実際の処理は INTTM01 で行う

---

## [関数名] r\_inttm01\_interrupt

---

概要	IIC00 の転送完了処理を行う
ヘッダ	r_cg_macrodriver.h
宣言	#pragma interrupt INTTM01 r_inttm01_interrupt __interrupt static void r_inttm01_interrupt (void)
説明	IIC00 の通信完了割り込み（INTIIC00）で起動され、スレーブでの処理時間確保のための時間待ちを行い、IIC00 通信の割り込み処理を行います。
引数	なし
リターン値	なし
備考	結果は変数 g_iic00_status に格納されます。 通信完了しても、IIC バスは開放しません。

---

## 6.8 フローチャート

図 6.1 に本アプリケーションノートの全体フローを示します。

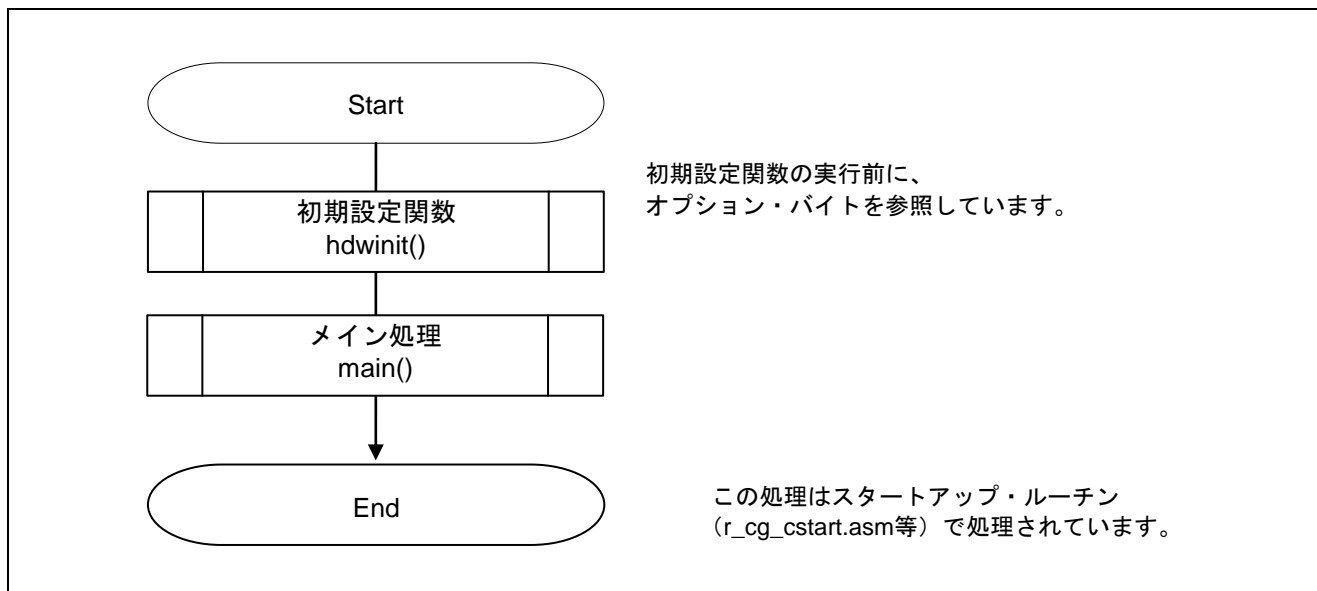


図 6.1 全体フロー

注 この処理は、スタートアップ・ルーチン（r\_cg\_cstart.asm 等）で処理されています。初期設定関数とメイン処理関数の呼び出しの間でメモリ関係の設定を行っています。

### 6.8.1 初期設定関数

図 6.2 に初期設定関数のフローチャートを示します。

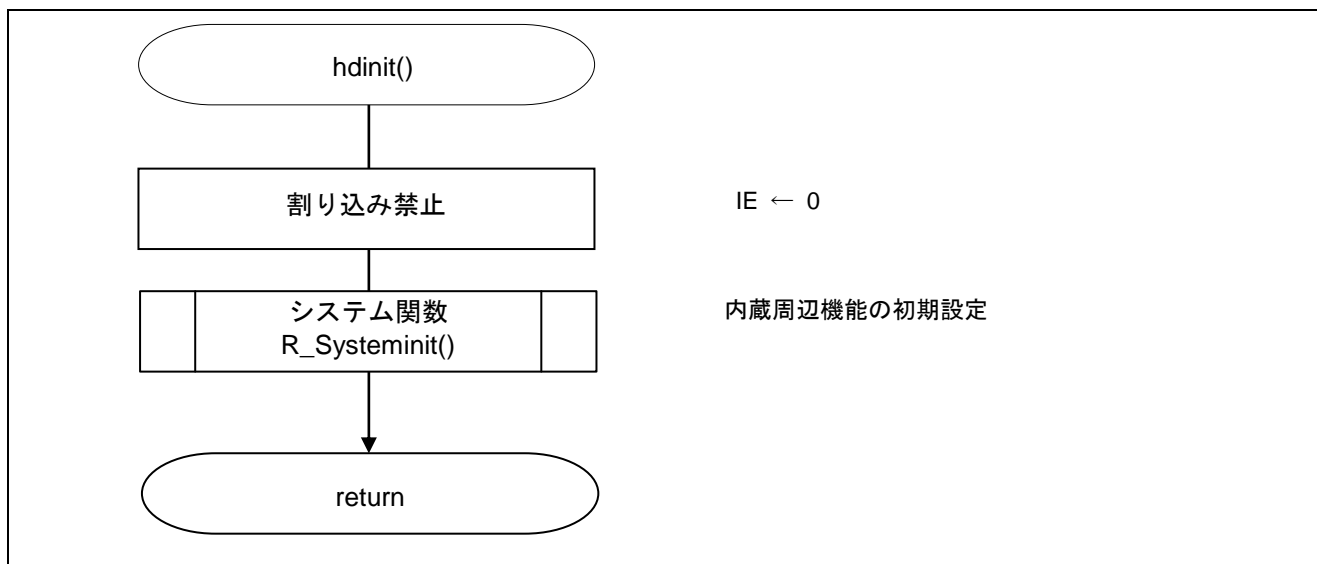


図 6.2 初期設定関数

## 6.8.2 システム関数

図 6.3 にシステム関数のフローチャートを示します。

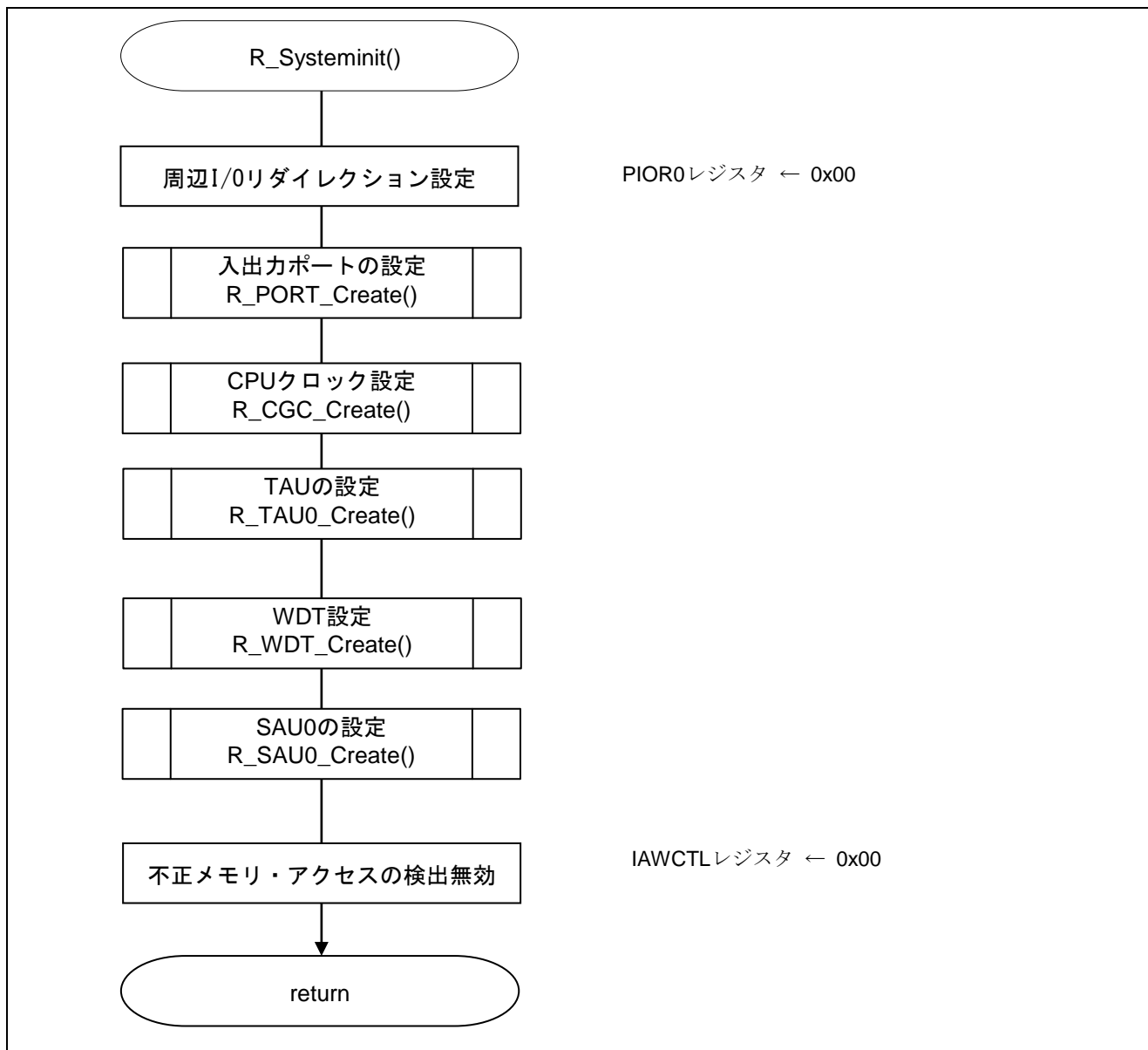


図 6.3 システム関数

### 6.8.3 CPU クロックの設定

図 6.4 に CPU クロック設定のフローチャートを示します。

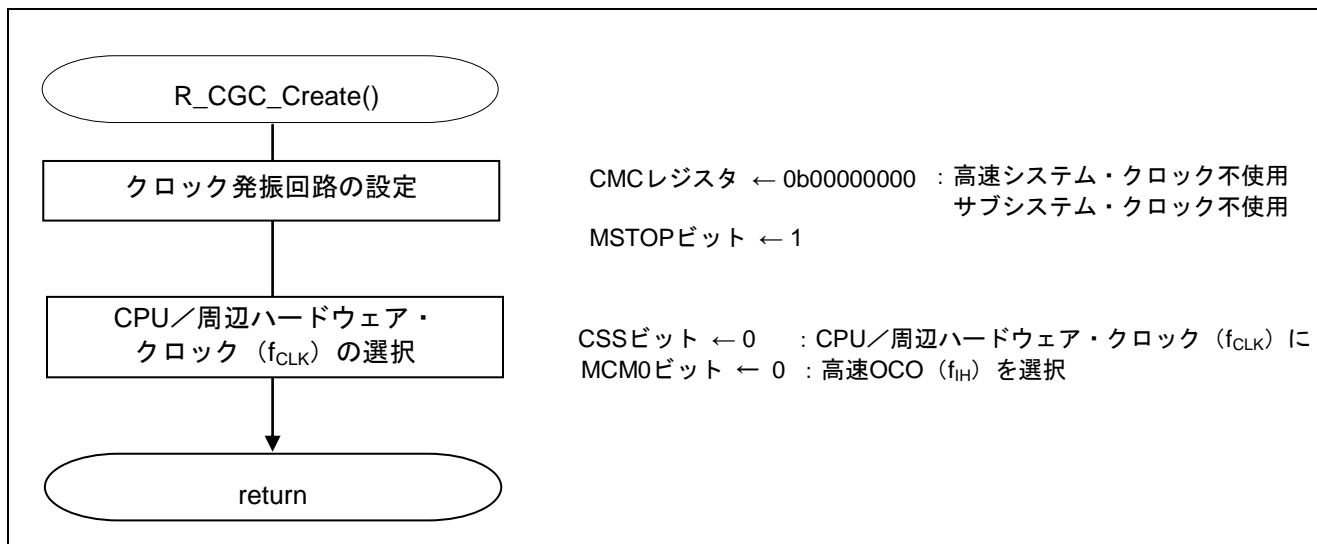


図 6.4 CPU クロックの設定

### 6.8.4 入出力ポート設定

図 6.5 に入出力ポート設定のフローチャートを示します。

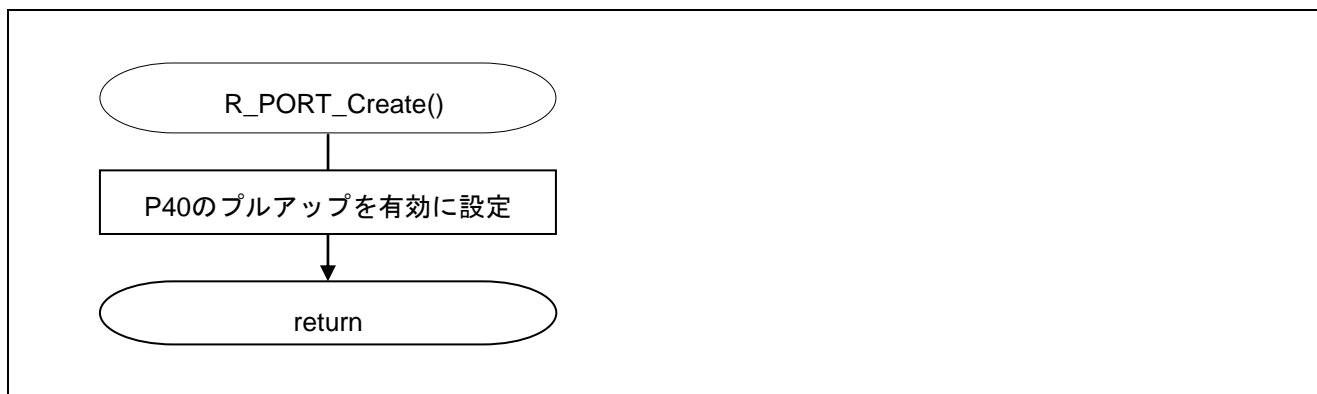


図 6.5 入出力ポート設定

**注意** 未使用のポートは、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。また、未使用の入力専用ポートは個別に抵抗を介して V<sub>DD</sub> 又は V<sub>SS</sub> に接続して下さい。

6.8.5 SAU0 の設定

図 6.6 に SAU0 の設定のフローチャートを示します。

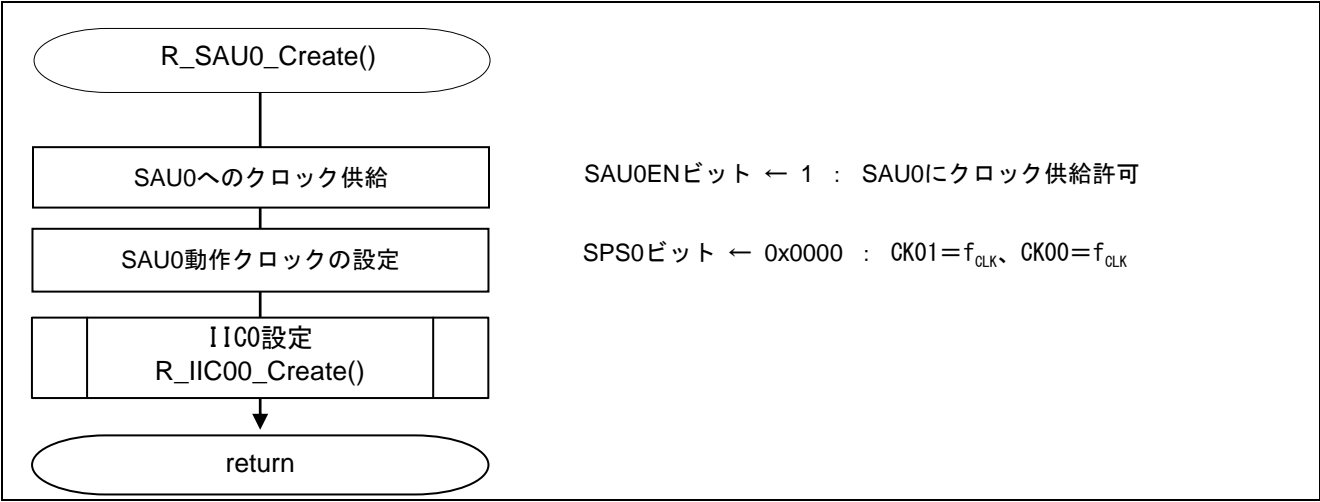


図 6.6 SAU0 の設定

SAU0 へのクロック 供給開始

- ・ 周辺イネーブル・レジスタ 0 (PER0)
- SAU0 へのクロック 供給を開始します

略号 : PER0

7	6	5	4	3	2	1	0
TMKAEN	CMPEN 注	ADCEN	IICA0EN 注	0	SAU0EN	0	TAU0EN
x	x	x	x	0	1	0	x

注 : 16 ピン製品のみ

ビット 4

SAU0EN	SAU0 の入力クロックの制御
0	入力クロック 供給停止
1	入力クロック 供給

SAU0 の動作クロック の設定

- ・ シリアル・クロック 選択レジスタ 0 (SPS0)
- CK00、CK01 を設定します

略号：SPS0

7	6	5	4	3	2	1	0
PRS 013	PRS 012	PRS 011	PRS 010	PRS 003	PRS 002	PRS 001	PRS 000
x	x	x	x	0	0	0	0

ビット 3－0

PRS 003	PRS 002	PRS 001	PRS 000	動作クロック（CK00）の選択					
					$f_{CLK}=$ 1MHz	$f_{CLK}=$ 2MHz	$f_{CLK}=$ 4MHz	$f_{CLK}=$ 16MHz	$f_{CLK}=$ 24MHz
0	0	0	0	$f_{CLK}$	1 MHz	2 MHz	4 MHz	16MHz	24 MHz
0	0	0	1	$f_{CLK}/2$	500 kHz	1 MHz	2MHz	8 MHz	12 MHz
0	0	1	0	$f_{CLK}/2^2$	250 kHz	500 kHz	1 MHz	4MHz	6 MHz
0	0	1	1	$f_{CLK}/2^3$	125 kHz	250 kHz	500 kHz	2 MHz	3 MHz
0	1	0	0	$f_{CLK}/2^4$	62.5 kHz	125 kHz	250 kHz	1 MHz	1.5 MHz
0	1	0	1	$f_{CLK}/2^5$	31.3 kHz	62.5 kHz	125 kHz	500 kHz	750 kHz
0	1	1	0	$f_{CLK}/2^6$	15.6 kHz	31.3 kHz	62.5 kHz	250 kHz	375 kHz
0	1	1	1	$f_{CLK}/2^7$	7.81 kHz	15.6 kHz	31.3 kHz	125 kHz	187.5 kHz
1	0	0	0	$f_{CLK}/2^8$	3.91 kHz	7.81 kHz	15.6 kHz	62.5 kHz	93.8 kHz
1	0	0	1	$f_{CLK}/2^9$	1.95 kHz	3.91 kHz	7.81 kHz	31.3 kHz	46.9 kHz
1	0	1	0	$f_{CLK}/2^{10}$	977 Hz	1.95 kHz	3.91 kHz	15.6 kHz	23.4 kHz
1	0	1	1	$f_{CLK}/2^{11}$	488 Hz	977 Hz	1.95 kHz	7.81 kHz	11.7 kHz
1	1	0	0	$f_{CLK}/2^{12}$	244 Hz	488 Hz	977 Hz	3.91 kHz	5.86 kHz
1	1	0	1	$f_{CLK}/2^{13}$	122 Hz	244 Hz	488 Hz	1.95 kHz	2.93 kHz
1	1	1	0	$f_{CLK}/2^{14}$	61 Hz	122 Hz	244 Hz	977 Hz	1.46 kHz
1	1	1	1	$f_{CLK}/2^{15}$	30.5 Hz	61 Hz	122 Hz	488 Hz	732 Hz

**注意** レジスタ設定方法の詳細については、RL78/G10 ユーザーズマニュアル ハードウェア編を参照してください。



## 6.8.6 IIC00 の設定

図 6.7 に IIC00 の設定のフローチャートを示します。

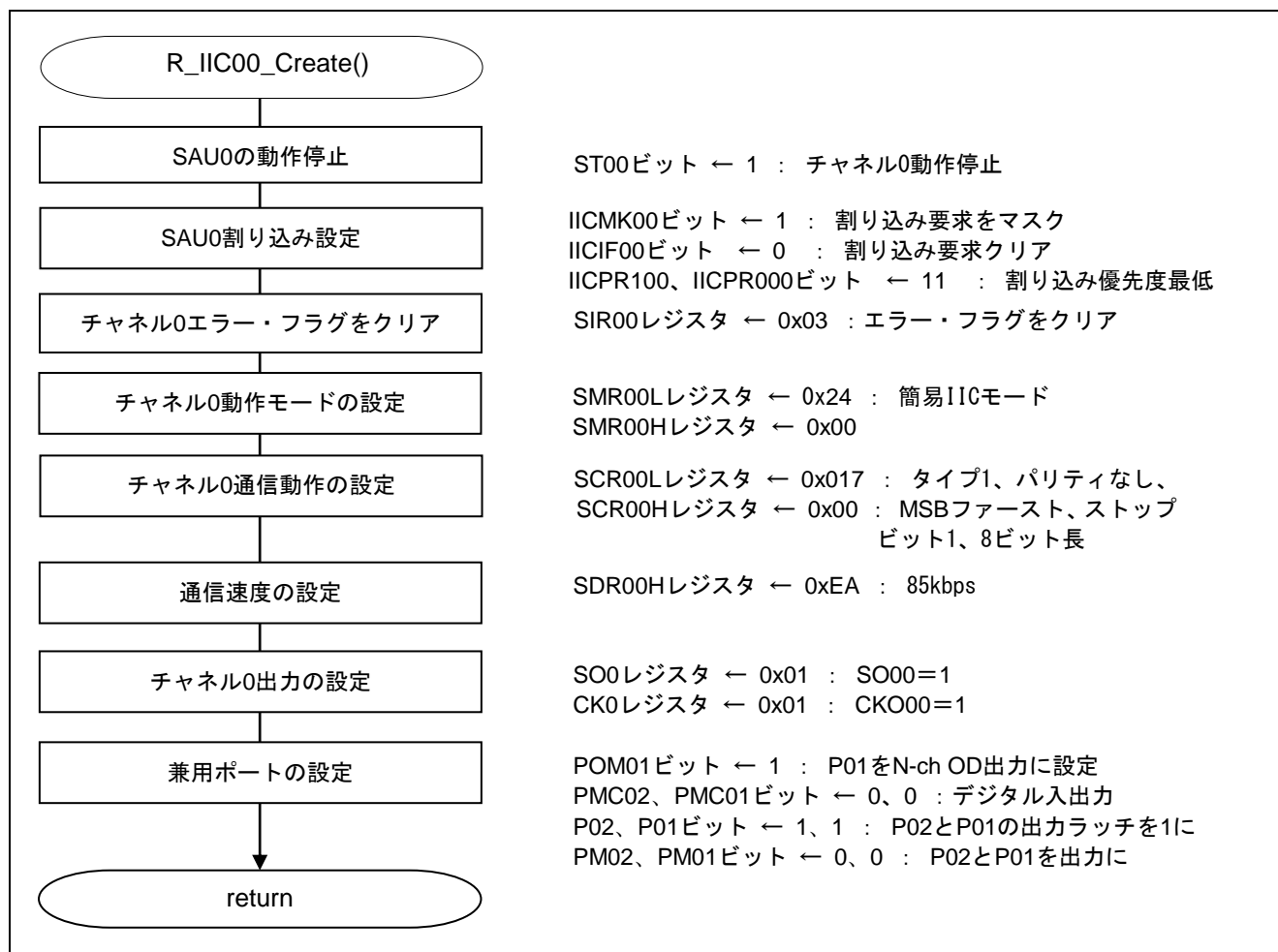


図 6.7 IIC00 の設定

## SAU0 の動作停止

- ・シリアル・チャンネル停止レジスタ 0 (ST0)  
チャンネル 0 を動作停止します

略号 : ST0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	ST01	ST00
0	0	0	0	0	0	x	1

ビット 0

ST00	チャンネル 0 の動作停止トリガ
0	トリガ動作せず
1	SE00 ビットを 0 にクリアし、通信動作を停止する

## IIC00 の割り込み制御

- ・割り込みマスク・フラグ・レジスタ（MK0L）の IICMK00 ビット  
IIC00 の割り込みをマスクします
- ・割り込み要求フラグ・レジスタ（IF0L）の IICIF00 ビット  
IIC00 の割り込み要求をマスクします
- ・優先順位指定フラグ・レジスタ（PR00L、PR10L）の IICPR000、IICPR010 ビット  
IIC00 の割り込み優先度を最低に設定します

略号：MK0L

ビット 3

IICMK00	INTIIC00 の割り込みの制御
0	割り込み処理許可
1	割り込み処理禁止

略号：IF0L

ビット 3

IICIF00	INTIIC00 の割り込み要求フラグ
0	割り込み要求信号が発生していない
1	割り込み要求信号が発生し、割り込み要求状態

略号：PR00L、PR10L

ビット 3

IICPR010	IICPR000	INTIIC00 の優先順位レベルの選択
0	0	レベル 0 を指定(高優先順位)
0	1	レベル 1 を指定
1	0	レベル 2 を指定
1	1	レベル 3 を指定(低優先順位)

## エラー・フラグのクリア

- ・シリアル・フラグ・クリア・トリガ・レジスタ 00（SIR00）  
エラー・フラグをクリアします

略号：SIR00（ビット 15～ビット 8 は 0 固定）

7	6	5	4	3	2	1	0
0	0	0	0	0	FECT00	PECT00	OVCT00
0	0	0	0	0	0	1	1

ビット 1, 0

PECT00,OVCT00	エラー・フラグのクリア・トリガ
0	クリアしない
1	クリアする

## チャンネル 0 動作モードの設定

- ・シリアル・モード・レジスタ 00H、00L（SMR00H、SMR00L）  
チャンネル 0 を簡易 IIC に設定します

略号：SMR00H、SMR00L

SMR00H								SMR00L							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
CKS00	CCS00	0	0	0	0	0	STS00	0	SIS00	1	0	0	MD002	MD001	MD000
<b>0</b>	<b>0</b>	0	0	0	0	0	<b>0</b>	0	<b>0</b>	1	0	0	<b>1</b>	<b>0</b>	<b>0</b>

## ビット 7

CKS00	チャンネル 0 の動作クロック ( $f_{MCK}$ ) の選択
<b>0</b>	SPS0 レジスタで設定した動作クロック CK00
1	SPS0 レジスタで設定した動作クロック CK01

## ビット 6

CCS00	チャンネル 0 の転送クロック ( $f_{CLK}$ ) の選択
<b>0</b>	CKS00 ビットで指定した動作クロック $f_{MCK}$ の分周クロック
1	SCK00 端子からの入力クロック $f_{SCK}$ (CSI モードのスレーブ転送)

## ビット 0

STS00	スタート・トリガ要因の選択
<b>0</b>	ソフトウェア・トリガのみ有効
1	RxD0 端子の有効エッジ (UART 受信時に選択)

## ビット 6

SIS00	UART モードでのチャンネル 0 の受信データのレベル反転の制御
<b>0</b>	入力される通信データは、そのまま取り込まれます。
1	入力される通信データは、反転して取り込まれます。

## ビット 2, 1

MD002	MD001	チャンネル 0 の動作モードの設定
0	0	CSI モード
0	1	UART モード
<b>1</b>	<b>0</b>	簡易 I2C モード
1	1	設定禁止

## ビット 0

MD000	チャンネル 0 の割り込み要因の選択
<b>0</b>	転送完了割り込み
1	バッファ空き割り込み

**注意** レジスタ設定方法の詳細については、RL78/G10 ユーザーズマニュアル ハードウェア編を参照してください。

## チャンネル 0 通信動作の設定

- ・シリアル通信動作設定レジスタ 00H、00L（SCR00H、SCR00L）  
タイプ 1、パリティなし、MSB ファースト、ストップ・ビット 1、8 ビット長に設定します

略号：SCR00H、SCR00L

SCR00H								SCR00L							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
TXE 00	RXE 00	DAP 00	CKP 00	0	EOC 00	PTC 001	PTC 000	DIR 00	0	SLC 001	SLC 000	0	1	DLS 001	DLS 000
0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1

ビット 7、6

TXE00	RXE00	チャンネル 0 の動作モードの設定
0	0	通信禁止
0	1	受信のみを行う
1	0	送信のみを行う
1	1	送受信を行う

ビット 5、4

DAP00	CKP00	CSI モードでのデータとクロックの位相選択
0	0	タイプ 1
0	1	タイプ 2
1	0	タイプ 3
1	1	タイプ 4

ビット 2

EOC00	エラー割り込み信号(INTSRE0)のマスク制御
0	エラー割り込み INTSRE0 発生を禁止する(INTSR0 が発生する)
1	エラー割り込み INTSRE0 の発生を許可する(エラー発生時、INTSR0 は発生しない)

ビット 1、0

PTC001	PTC000	UART モードでのパリティ・ビットの設定
0	0	パリティなし
0	1	0 パリティ
1	0	偶数パリティ
1	1	奇数パリティ

ビット 7

DIR00	CSI, UART モードでのデータ転送順序の選択
0	MSB ファーストで入出力を行う
1	LSB ファーストで入出力を行う

ビット 5、4

SLC001	SLC000	UART モードでのストップ・ビットの設定
0	0	ストップ・ビットなし
0	1	ストップ・ビット長 = 1 ビット
1	0	ストップ・ビット長 = 2 ビット
1	1	設定禁止

略号：SCR00H、SCR00L

SCR00H								SCR00L							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
TXE 00	RXE 00	DAP 00	CKP 00	0	EOC 00	PTC 001	PTC 000	DIR 00	0	SLC 001	SLC 000	0	1	DLS 001	DLS 000
0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1

ビット 1、0

DLS001	DLS000	CSI, UART モードでのデータ長の設定
0	0	通信禁止
0	1	9 ビット・データ長
1	0	7 ビット・データ長
1	1	8 ビット・データ長

## 通信速度の設定

- ・シリアル・データ・レジスタ 00H (SDR00H)  
85kbps に設定します

略号：SDR00H

15	14	13	12	11	10	9	0
1	1	1	0	1	0	1	0

## チャンネル 0 出力の設定

- ・シリアル・クロック出力レジスタ 0 (CKO0) シリアル出力レジスタ 0 (SO0)  
SCL00、SDA00 信号出力レベルを設定します

略号：CKO0、SO0

CKO0								SO0							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	0	0	0	0	0	CKO 01	CKO 00	0	0	0	0	0	0	SO 01	SO 00
0	0	0	0	0	0	x	1	0	0	0	0	0	0	x	1

ビット 0

CKO00	チャンネル 0 のシリアル・クロック出力
0	シリアル・クロック出力値が “0”
1	シリアル・クロック出力値が “1”

ビット 0

SO00	チャンネル 0 のシリアル・データ出力
0	シリアル・データ出力値が “0”
1	シリアル・データ出力値が “1”

## 兼用ポートの設定

- ・ポート出力モード・レジスタ 0 (POM0)  
N-ch オープン・ドレイン出力に設定します。
- ・ポート・モード・コントロール・レジスタ 0 (PMC0)  
デジタル入出力に設定します。
- ・ポート・レジスタ 0 (P0)  
出力ラッチに 1 を設定します。
- ・ポート・モード・レジスタ 0 (PM0)  
出力ポートに設定します。

略号：POM0

7	6	5	4	3	2	1	0
POM07 注	POM06 注	0	0	0	0	POM01	POM00
x	x	0	0	0	0	<b>1</b>	x

注：16 ピン製品のみ。10 ピン製品では 0

ビット 1

POM01	P01 端子の出力モードの選択
0	通常出力モード
<b>1</b>	<b>N-ch オープン・ドレイン出力</b>

略号：PMC0

7	6	5	4	3	2	1	0
PMC07 注	PMC06 注	PMC05 注	PMC04	PMC03	PMC02	PMC01	1
x	x	x	x	x	<b>0</b>	<b>0</b>	1

注：16 ピン製品のみ。10 ピン製品では 1

ビット 1

PMC02, PMC01	P02、P01 端子のデジタル入出力／アナログ入力の選択
<b>0</b>	<b>デジタル入出力</b>
1	アナログ入力

略号：P0

7	6	5	4	3	2	1	0
P07 注	P06 注	P05 注	P04	P03	P02	P01	P00
x	x	x	x	x	<b>1</b>	<b>1</b>	x

注：16 ピン製品のみ。10 ピン製品では 0

ビット 2, 1

P02, P01	P02, P01 出力ラッチの設定
0	0 を設定
<b>1</b>	<b>1 を設定</b>

略号：PM0

7	6	5	4	3	2	1	0
PM07 注	PM06 注	PM05 注	PM04	PM03	PM02	PM01	PM00
x	x	x	x	x	<b>0</b>	<b>0</b>	x

注：16 ピン製品のみ。10 ピン製品では 1

ビット 2, 1

PM02, PM01	P02, P01 端子の入出力モードの選択
<b>0</b>	<b>出力モード</b>
1	入力モード

**注意** レジスタ設定方法の詳細については、RL78/G10 ユーザーズマニュアル ハードウェア編を参照してください。

## 6.8.7 タイマ・アレイ・ユニットの設定

図 6.8 にタイマ・アレイ・ユニットの設定のフローチャートを示します。

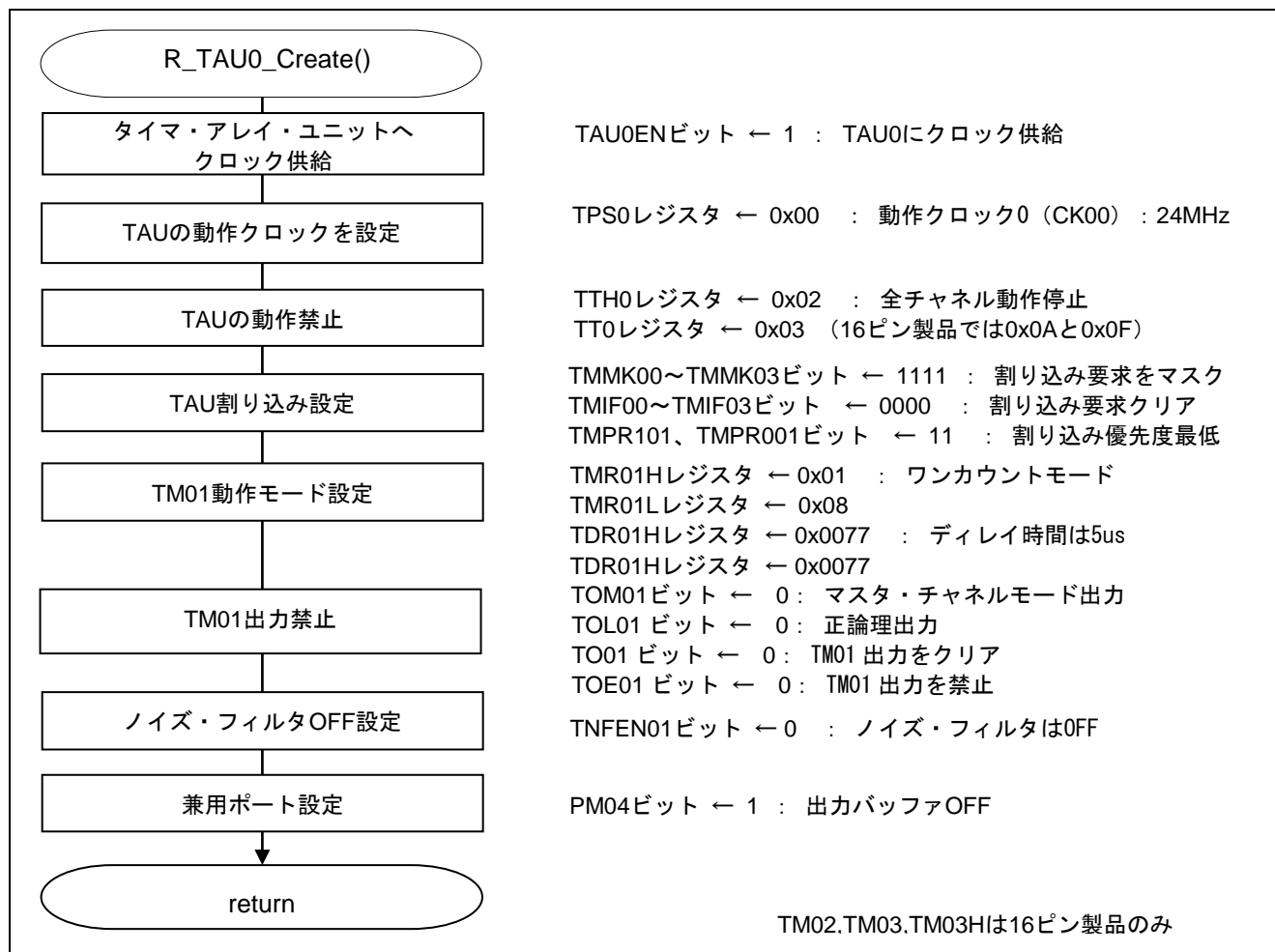


図 6.8 タイマ・アレイ・ユニットの設定

## タイマ・アレイ・ユニット 0 へのクロック供給開始

・周辺イネーブル・レジスタ 0 (PER0)

タイマ・アレイ・ユニット 0 へのクロック供給を開始します

略号 : PER0

7	6	5	4	3	2	1	0
TMKAEN <sup>注</sup>	CMPEN <sup>注</sup>	ADCEN	IICA0EN <sup>注</sup>	0	SAU0EN	0	TAU0EN
x	0	x	0	0	x	0	<b>1</b>

注 : 16 ピン製品のみ

## ビット 0

TAU0EN	タイマ・アレイ・ユニット 0 の入力クロックの制御
0	入力クロック供給停止
<b>1</b>	<b>入力クロック供給</b>

## タイマ・クロック周波数の設定

- ・タイマ・クロック選択レジスタ 0 (TPS0)
- タイマ・アレイ・ユニット 0 の動作クロックを選択

略号 : TPS0

7	6	5	4	3	2	1	0
PRS 013	PRS 012	PRS 011	PRS 010	PRS 003	PRS 002	PRS 001	PRS 000
x	x	x	x	0	0	0	0

ビット 3 - 0

PRS 003	PRS 002	PRS 001	PRS 000	動作クロック (CK00) の選択					
					f <sub>CLK</sub> = 1MHz	f <sub>CLK</sub> = 2MHz	f <sub>CLK</sub> = 4MHz	f <sub>CLK</sub> = 16MHz	f <sub>CLK</sub> = 24MHz
0	0	0	0	f <sub>CLK</sub>	1 MHz	2 MHz	4 MHz	16MHz	24 MHz
0	0	0	1	f <sub>CLK</sub> /2	500 kHz	1 MHz	2 MHz	8 MHz	12 MHz
0	0	1	0	f <sub>CLK</sub> /2 <sup>2</sup>	250 kHz	500 kHz	1 MHz	4MHz	6 MHz
0	0	1	1	f <sub>CLK</sub> /2 <sup>3</sup>	125 kHz	250 kHz	500 kHz	2 MHz	3 MHz
0	1	0	0	f <sub>CLK</sub> /2 <sup>4</sup>	62.5 kHz	125 kHz	250 kHz	1 MHz	1.5 MHz
0	1	0	1	f <sub>CLK</sub> /2 <sup>5</sup>	31.3 kHz	62.5 kHz	125 kHz	500 kHz	750 kHz
0	1	1	0	f <sub>CLK</sub> /2 <sup>6</sup>	15.6 kHz	31.3 kHz	62.5 kHz	250 kHz	375 kHz
0	1	1	1	f <sub>CLK</sub> /2 <sup>7</sup>	7.81 kHz	15.6 kHz	31.3 kHz	125 kHz	187.5 kHz
1	0	0	0	f <sub>CLK</sub> /2 <sup>8</sup>	3.91 kHz	7.81 kHz	15.6 kHz	62.5 kHz	93.8 kHz
1	0	0	1	f <sub>CLK</sub> /2 <sup>9</sup>	1.95 kHz	3.91 kHz	7.81 kHz	31.3 kHz	46.9 kHz
1	0	1	0	f <sub>CLK</sub> /2 <sup>10</sup>	977 Hz	1.95 kHz	3.91 kHz	15.6 kHz	23.4 kHz
1	0	1	1	f <sub>CLK</sub> /2 <sup>11</sup>	488 Hz	977 Hz	1.95 kHz	7.81 kHz	11.7 kHz
1	1	0	0	f <sub>CLK</sub> /2 <sup>12</sup>	244 Hz	488 Hz	977 Hz	3.91 kHz	5.86 kHz
1	1	0	1	f <sub>CLK</sub> /2 <sup>13</sup>	122 Hz	244 Hz	488 Hz	1.95 kHz	2.93 kHz
1	1	1	0	f <sub>CLK</sub> /2 <sup>14</sup>	61 Hz	122 Hz	244 Hz	977 Hz	1.46 kHz
1	1	1	1	f <sub>CLK</sub> /2 <sup>15</sup>	30.5 Hz	61 Hz	122 Hz	488 Hz	732 Hz

**注意** レジスタ設定方法の詳細については、RL78/G10 ユーザーズマニュアル ハードウェア編を参照してください。



## タイマ動作停止の設定

- ・タイマ・チャンネル停止レジスタ 0（TTH0、TT0）  
タイマ・チャンネルの動作停止を選択

略号：TTH0、TT0

TTH0								TT0							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	0	0	0	TT03H <sup>注</sup>	0	TT01H	0	0	0	0	0	TT03 <sup>注</sup>	TT02 <sup>注</sup>	TT01	TT00
0	0	0	0	1 <sup>注</sup>	0	1	0	0	0	0	0	1 <sup>注</sup>	1 <sup>注</sup>	1	1

注：16 ピン製品のみ

## ビット n

TT0n	チャンネル n の動作停止トリガ
0	トリガ動作しない
1	TE0n ビットが 0 にクリアされ、カウント動作を停止する（停止トリガ発生）

## タイマのカウント完了割り込みの設定

- ・割り込みマスク・フラグ・レジスタ（MK0H）の TMMK01 ビット  
割り込みマスクの設定
- ・割り込み要求フラグ・レジスタ（IF0H）の TMIF01 ビット  
割り込み要求フラグのクリア
- ・優先順位指定フラグ・レジスタ（PR00H、PR10H）の TMPR001、TMPR101 ビット  
TM03 の割り込み優先度を最低に設定します

略号：MK0H

## ビット 0

TMMK01	割り込み処理の制御
0	割り込み処理許可
1	割り込み処理禁止

略号：IF0H

## ビット 0

TMIF01	割り込み要求フラグ
0	割り込み要求信号が発生していない
1	割り込み要求信号が発生し、割り込み要求状態

略号：PR00H、PR10H

## ビット 0

TMPR101	TMPR001	INTTM01 の優先順位レベルの選択
0	0	レベル 0 を指定(高優先順位)
0	1	レベル 1 を指定
1	0	レベル 2 を指定
1	1	レベル 3 を指定(低優先順位)

**注意** レジスタ設定方法の詳細については、RL78/G10 ユーザーズマニュアル ハードウェア編を参照してください。

## チャンネル 1 の動作モードの設定

・ タイマ・モード・レジスタ 01（TMR01H、TMR01L）

動作クロック（ $f_{MCK}$ ）の選択

カウント・クロックの選択

スタート・トリガとキャプチャ・トリガの設定

タイマ入力の有効エッジ選択

動作モード設定

略号：TMR01H、TMR01L

TMR01H								TMR01L							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
CKS 011	0	0	CCS 01	SPLIT 01	STS 012	STS 011	STS 010	CIS 011	CIS 030	0	0	MD 013	MD 012	MD 011	MD 010
<b>0</b>	0	0	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>0</b>	<b>0</b>	0	0	<b>1</b>	<b>0</b>	<b>0</b>	<b>0</b>

## ビット 7

CKS011	チャンネル 1 の動作クロック（ $f_{MCK}$ ）の選択
<b>0</b>	<b>タイマ・クロック選択レジスタ 0（TPS0）で設定した動作クロック CK00</b>
0	タイマ・クロック選択レジスタ 0（TPS0）で設定した動作クロック CK02
1	タイマ・クロック選択レジスタ 0（TPS0）で設定した動作クロック CK01
1	タイマ・クロック選択レジスタ 0（TPS0）で設定した動作クロック CK03

## ビット 4

CCS01	チャンネル 1 のカウント・クロック（ $f_{CLK}$ ）の選択
<b>0</b>	<b>CKS011、CKS010 ビットで指定した動作クロック（<math>f_{MCK}</math>）</b>
1	TI03 端子からの入力信号の有効エッジ

## ビット 3

SPLIT01	チャンネル 1 の 8 ビット・タイマ／16 ビット・タイマ動作の選択
<b>0</b>	<b>16 ビット・タイマとして動作</b>
1	8 ビット・タイマとして動作

## ビット 2－0

STS012	STS011	STS010	チャンネル 1 のスタート・トリガ、キャプチャ・トリガの設定
0	0	0	ソフトウェア・トリガ・スタートのみ有効（他のトリガ要因を非選択にする）
<b>0</b>	<b>0</b>	<b>1</b>	<b>TI00 端子入力の有効エッジを、スタート・トリガ、キャプチャ・トリガの両方に使用</b>
0	1	0	TI00 端子入力の両エッジを、スタート・トリガとキャプチャ・トリガに分けて使用
1	0	0	マスタ・チャンネルの割り込み信号を使用（複数チャンネル連動動作機能のスレーブ・チャンネル時）
上記以外			設定禁止

**注意** レジスタ設定方法の詳細については、RL78/G10 ユーザーズマニュアル ハードウェア編を参照してください。

ビット 7－6

CIS011	CIS010	TI01 端子の有効エッジ選択
<b>0</b>	<b>0</b>	<b>立ち下がリエッジ</b>
0	1	立ち上がりエッジ
1	0	両エッジ（ロウ・レベル幅測定時） スタート・トリガ：立ち下がリエッジ、キャプチャ・トリガ：立ち上がりエッジ
1	1	両エッジ（ハイ・レベル幅測定時） スタート・トリガ：立ち上がりエッジ、キャプチャ・トリガ：立ち下がリエッジ

ビット 3－0

MD 013	MD 012	MD 011	MD 010	チャンネル 1 の動作 モードの設定	対応する機能	TCR のカウント動作
0	0	0	1/0	インターバル・タイマ・モード	インターバル・タイマ／方形波出力／分周器機能／PWM 出力（マスタ）	ダウン・カウント
0	1	0	1/0	キャプチャ・モード	入力パルス間隔測定	アップ・カウント
0	1	1	0	イベント・カウンタ・モード	外部イベント・カウンタ	ダウン・カウント
<b>1</b>	<b>0</b>	<b>0</b>	<b>1/0</b>	<b>ワンカウント・モード</b>	ディレイ・カウンタ／ワンショット・パルス出力／PWM 出力（スレーブ）	ダウン・カウント
1	1	0	0	キャプチャ&ワンカウント・モード	入力信号のハイ／ロウ・レベル幅測定	アップ・カウント
上記以外				設定禁止		

### ディレイ時間設定

- ・タイマ・データ・レジスタ 01（TDR01H、TDR01L）  
ディレイ時間を設定

略号：TDR01H、TDR01L

TDR01H								TDR01L							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	1	1	1	0	1	1	1

**注意** レジスタ設定方法の詳細については、RL78/G10 ユーザーズマニュアル ハードウェア編を参照してください。

### タイマ出力禁止設定

- ・タイマ出力モード・レジスタ 0 (TOM0)  
マスタ・モード出力に設定
- ・タイマ出力レベル・レジスタ 0 (TOL0)  
正論理出力に設定
- ・タイマ出力レジスタ 0 (TO0)  
出力を 0 に設定
- ・タイマ出力許可レジスタ 0 (TOE0)  
各チャンネルのタイマ出力許可／禁止の値設定

略号：TOM0

7	6	5	4	3	2	1	0
0	0	0	0	TOM03 注	TOM02 注	TOM01	0
0	0	0	0	<b>0</b>	x	x	0

注：16 ピン製品のみ

ビット 1

TOM01	チャンネル 1 のタイマ出力モードの制御
<b>0</b>	マスタ・チャンネル出力モード
1	スレーブ・チャンネル出力モード

略号：TOL0

7	6	5	4	3	2	1	0
0	0	0	0	TOL03 注	TOL02 注	TOL01	0
0	0	0	0	<b>0</b>	x	x	0

注：16 ピン製品のみ

ビット 1

TOL01	チャンネル 1 のタイマ出力レベルの制御
<b>0</b>	正論理出力(アクティブ・ハイ)
1	反転出力(アクティブ・ロウ)

略号：TO0

7	6	5	4	3	2	1	0
0	0	0	0	TO03 注	TO02 注	TO01	TO00
0	0	0	0	<b>0</b>	x	x	x

注：16 ピン製品のみ

ビット 1

TO01	チャンネル 1 のタイマ出力レベルの制御
<b>0</b>	ロウ・レベル
1	ハイ・レベル

略号：TOE0

7	6	5	4	3	2	1	0
0	0	0	0	TOE03 注	TOE02 注	TOE01	TOE00
0	0	0	0	<b>0</b>	x	x	x

注：16 ピン製品のみ

ビット 1

TOE01	チャンネル 1 のタイマ出力許可／禁止
<b>0</b>	カウント動作による TO01（タイマ・チャンネル出力ビット）の動作停止。
1	カウント動作による TO01（タイマ・チャンネル出力ビット）の動作許可。

**注意** レジスタ設定方法の詳細については、RL78/G10 ユーザーズマニュアル ハードウェア編を参照してください。

6.8.8 WDT の設定

図 6.9 に WDT 設定のフローチャートを示します。

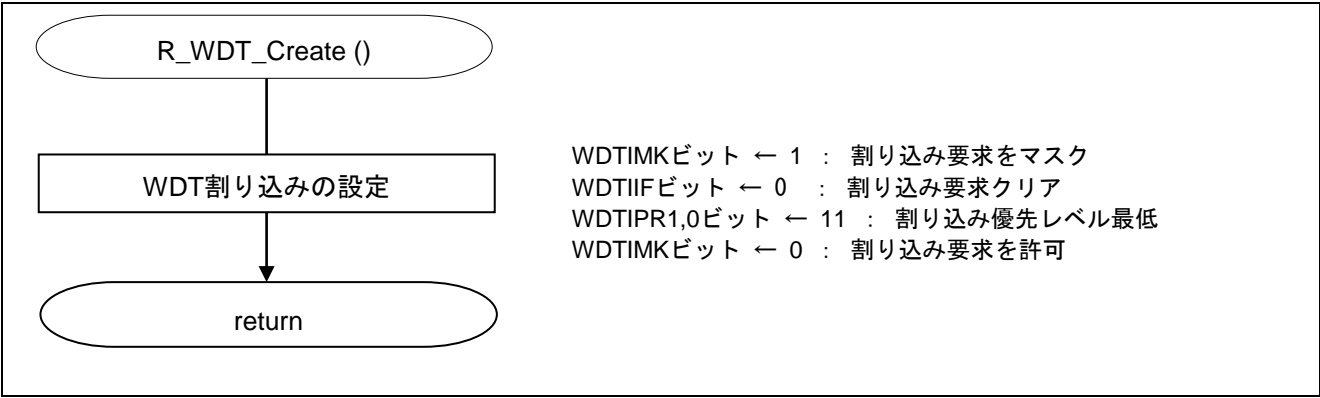


図 6.9 WDT の設定

WDT の割り込み制御

- ・ 割り込みマスク・フラグ・レジスタ（MK0L）の WDTIMK ビット  
WDT の割り込みをマスクします
- ・ 割り込み要求フラグ・レジスタ（IF0L）の WDTIIF ビット  
WDT の割り込み要求をクリアします
- ・ 優先順位指定フラグ・レジスタ（PR00L、PR10L）の WDTIPR0、WDTIPR1 ビット  
WDT の割り込み優先度を最低に設定します

略号：MK0L

ビット 0

WDTIMK	WDT の割り込みの制御
0	割り込み処理許可
1	割り込み処理禁止

略号：IF0L

ビット 0

WDTIIF	WDT の割り込み要求フラグ
0	割り込み要求信号が発生していない
1	割り込み要求信号が発生し、割り込み要求状態

略号：PR00L、PR10L

ビット 0

WDTIPR1	WDTIPR0	WDT の優先順位レベルの選択
0	0	レベル 0 を指定(高優先順位)
0	1	レベル 1 を指定
1	0	レベル 2 を指定
1	1	レベル 3 を指定(低優先順位)

## 6.8.9 メイン処理

図 6.10 にメイン処理のフローチャートを示します。

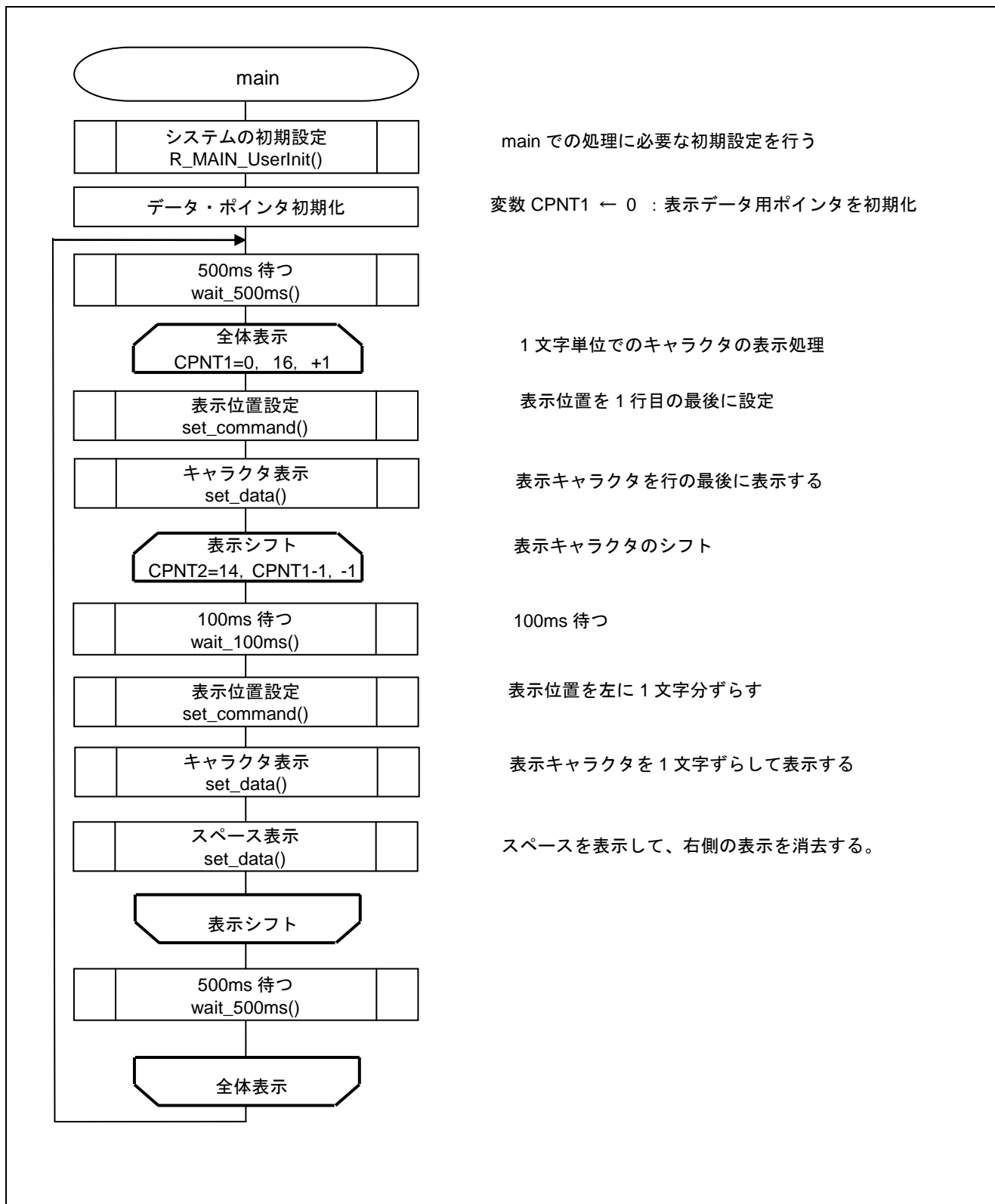
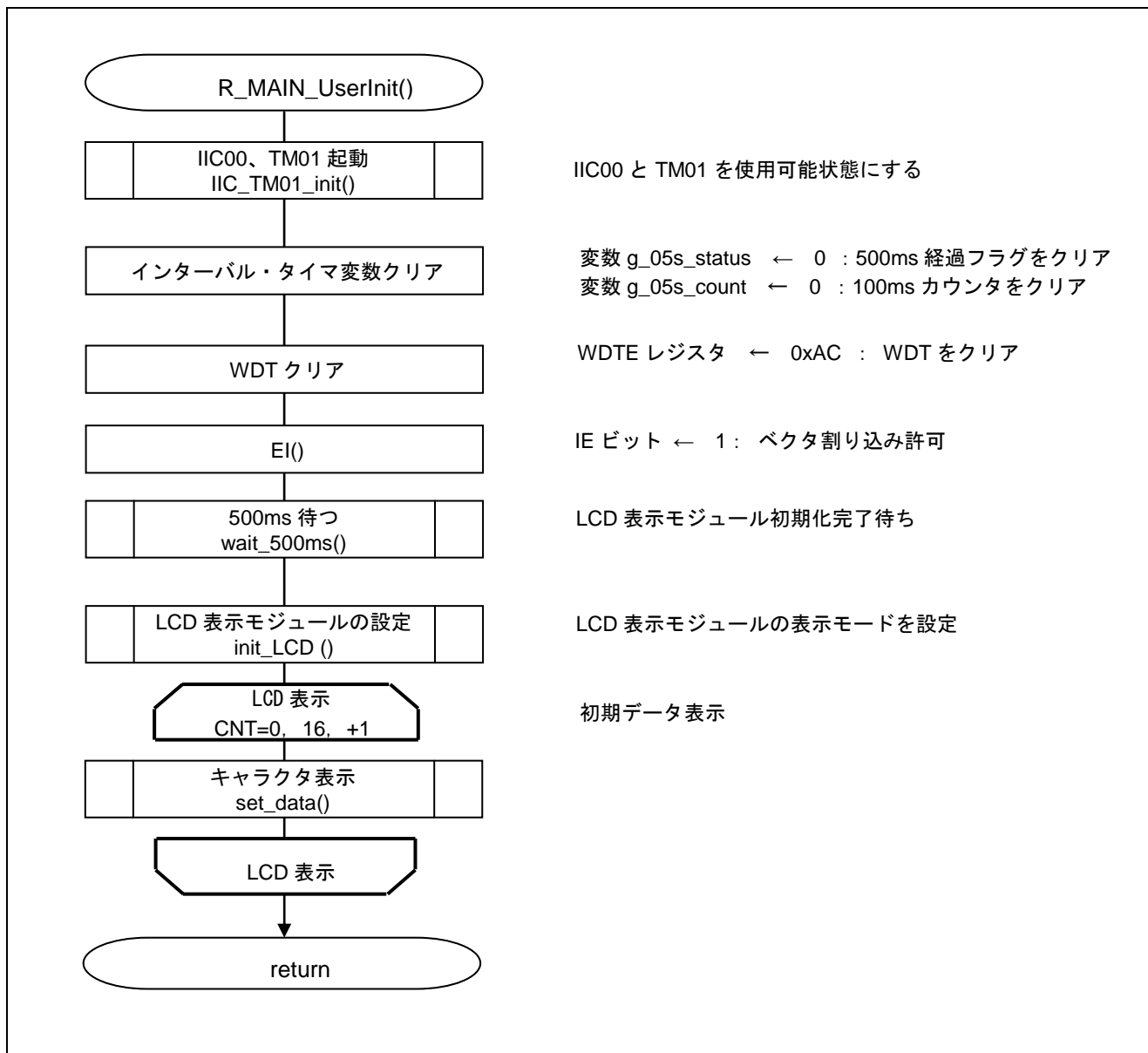


図 6.10 メイン処理

## 6.8.10 R\_MAIN\_UserInit 処理

図 6.11 に R\_MAIN\_UserInit 処理のフローチャートを示します。



## WDT クリア

・ウォッチドッグ・タイマ・イネーブル・レジスタ (WDTE)

WDT をクリア

略号 : WDTE

7	6	5	4	3	2	1	0
1	0	1	0	1	1	0	0

**注意** レジスタ設定方法の詳細については、RL78/G10 ユーザーズマニュアル ハードウェア編を参照してください。

## 6.8.11 500ms 待ち

図 6.12 に 500ms 待ち処理のフローチャートを示します。

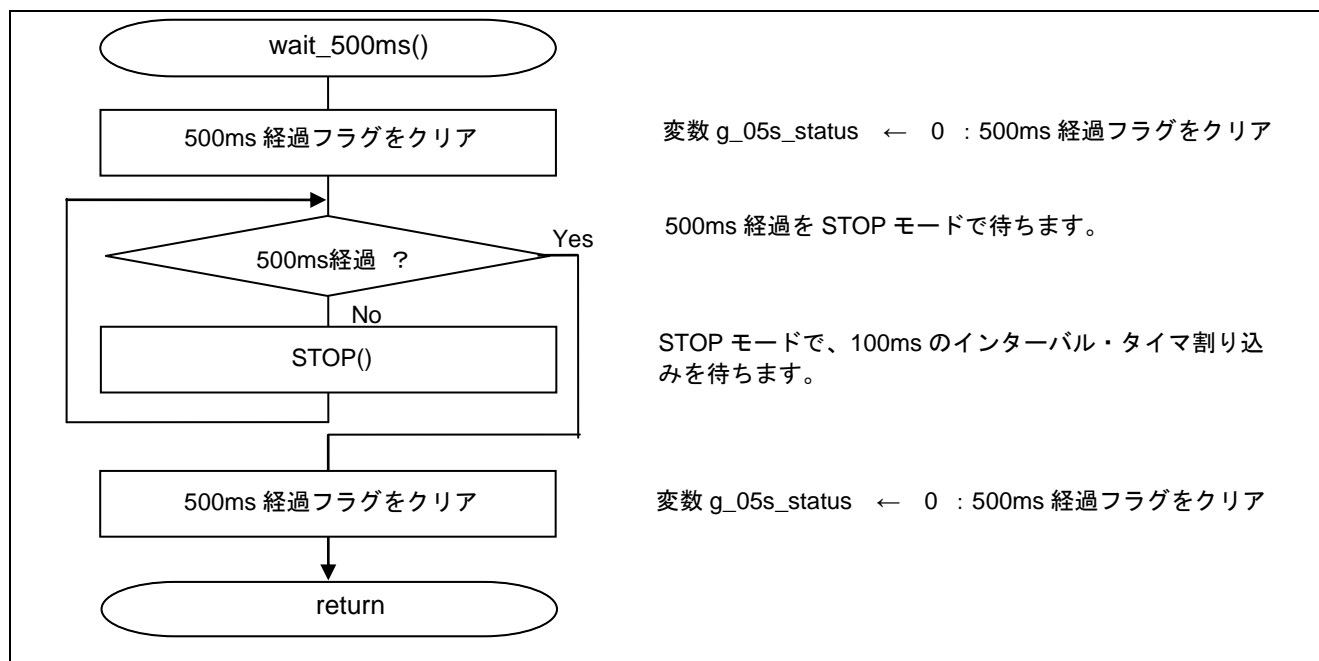


図 6.12 500ms 待ち処理

## 6.8.12 100ms 待ち

図 6.13 に 100ms 待ち処理のフローチャートを示します。

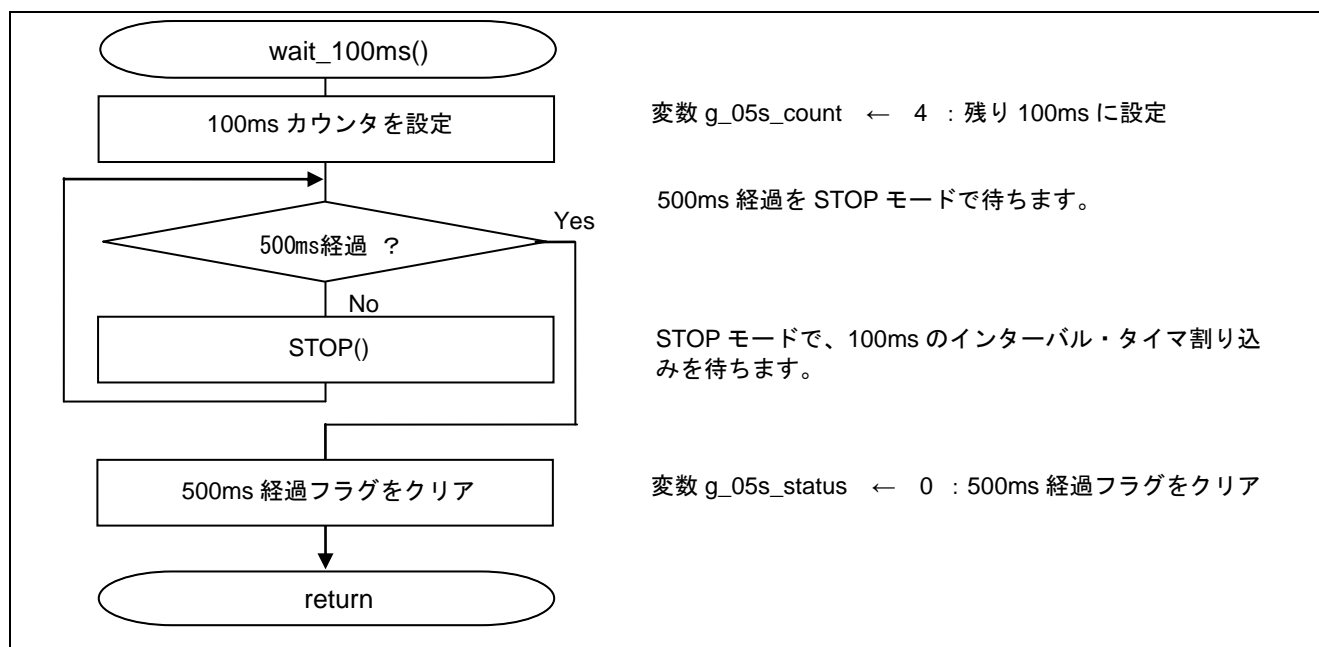


図 6.13 100ms 待ち処理



## 6.8.13 LCD モジュールの初期設定

図 6.14 に LCD モジュールの初期設定処理のフローチャートを示します。

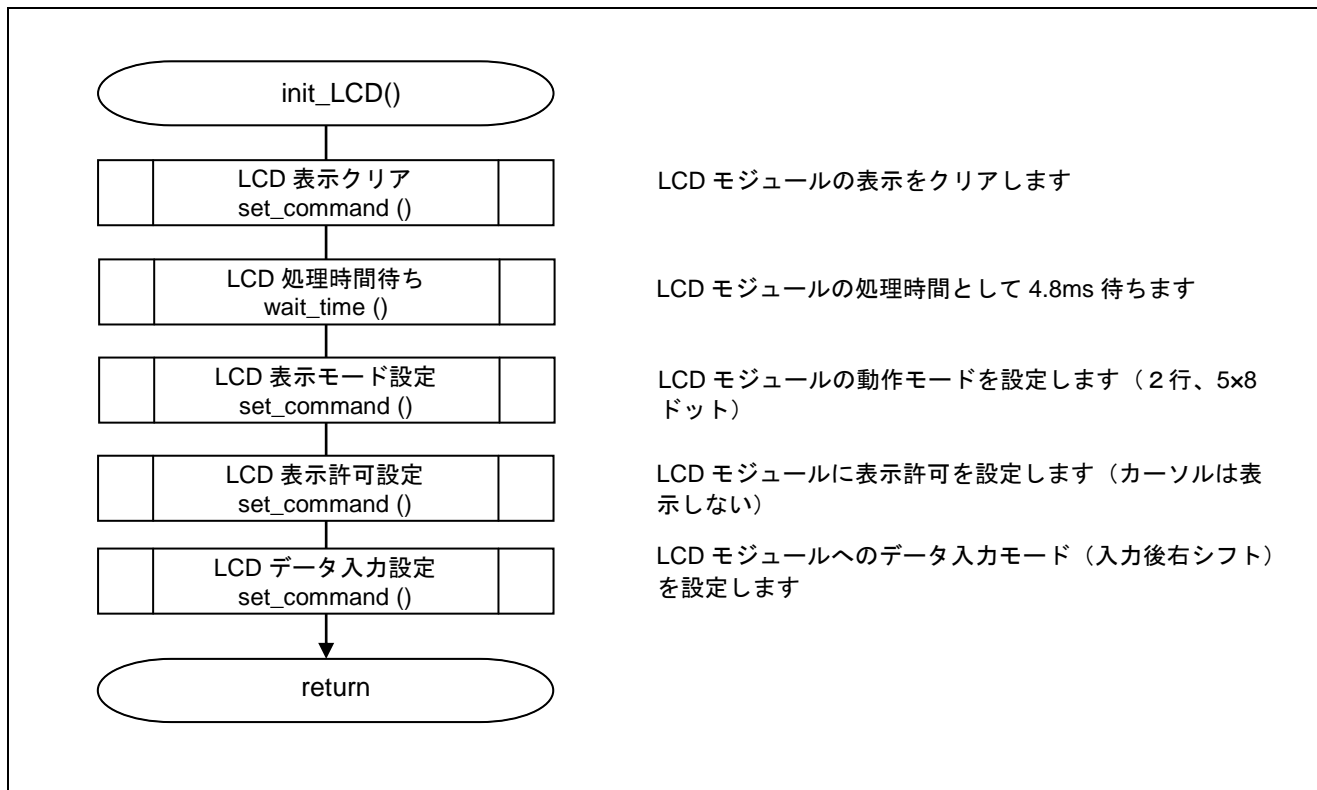


図 6.14 LCD モジュールの初期設定処理

## 6.8.14 LCD モジュールへのコマンド送信

図 6.15 に LCD モジュールへのコマンド送信処理のフローチャートを示します。

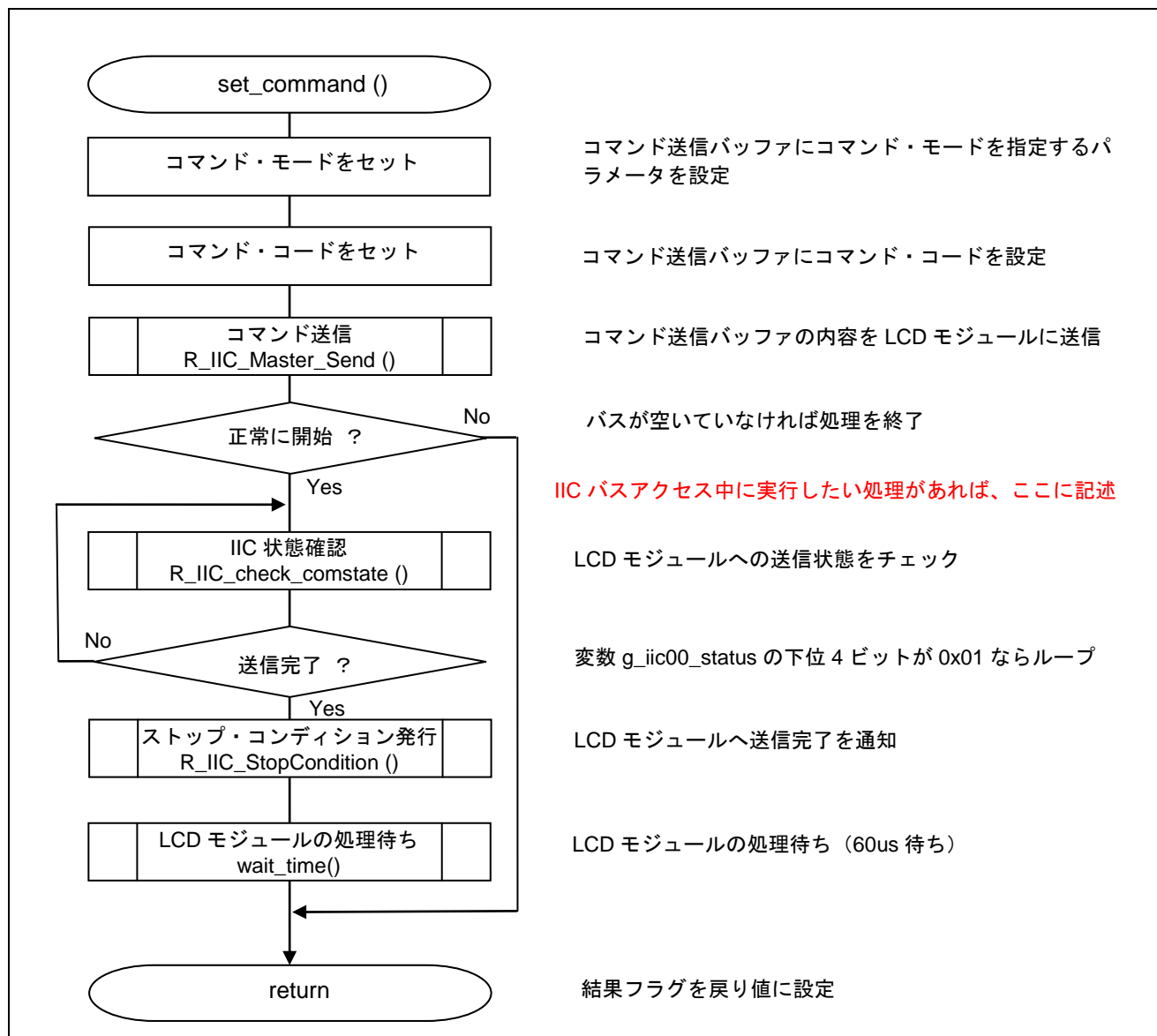


図 6.15 LCD モジュールへのコマンド送信処理

## 6.8.15 LCD モジュールへのデータ送信

図 6.16 に LCD モジュールへのデータ送信処理のフローチャートを示します。

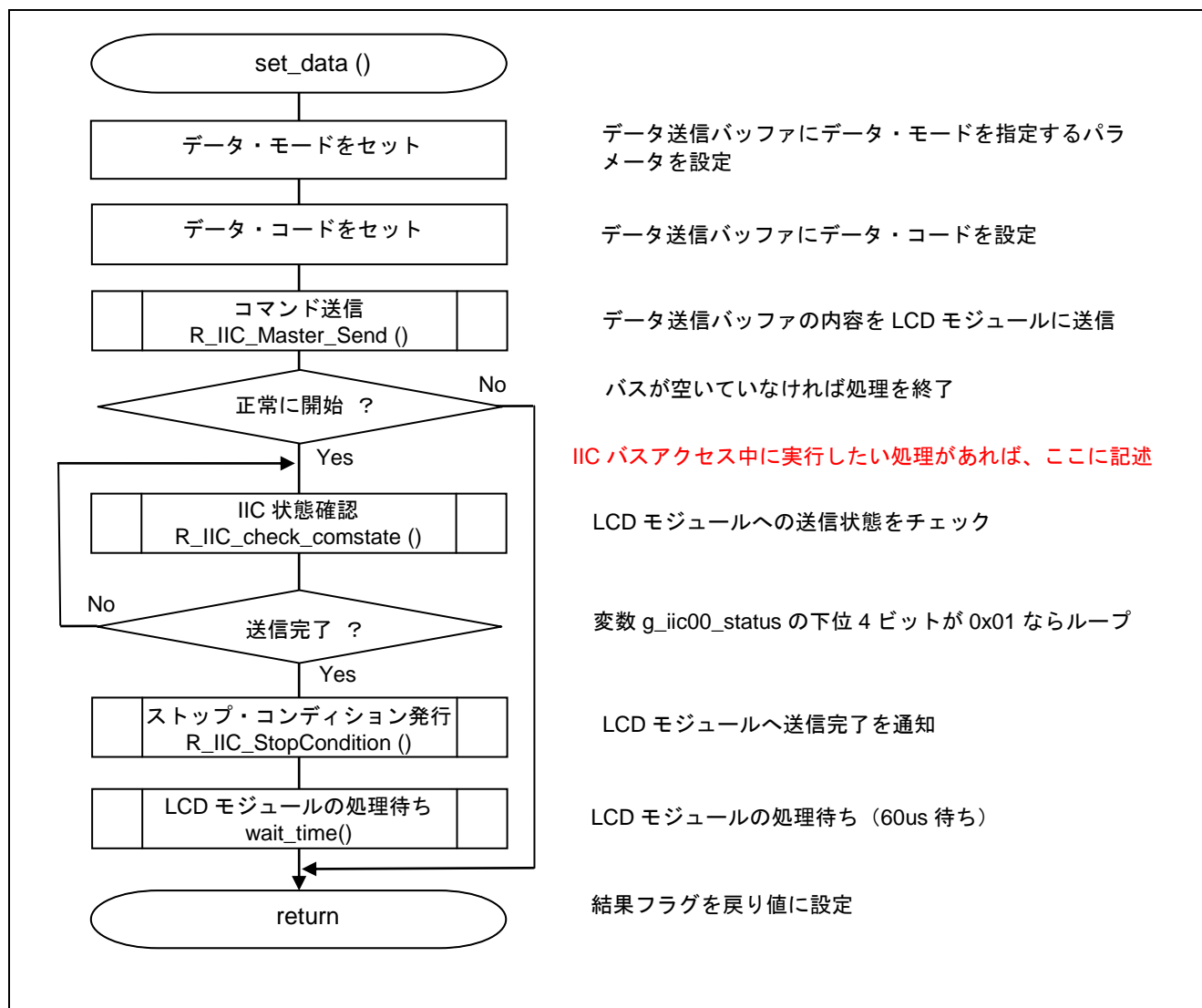


図 6.16 LCD モジュールへのデータ送信処理

## 6.8.16 WDT 割り込み処理

図 6.17 に WDT 割り込み処理のフローを示します。

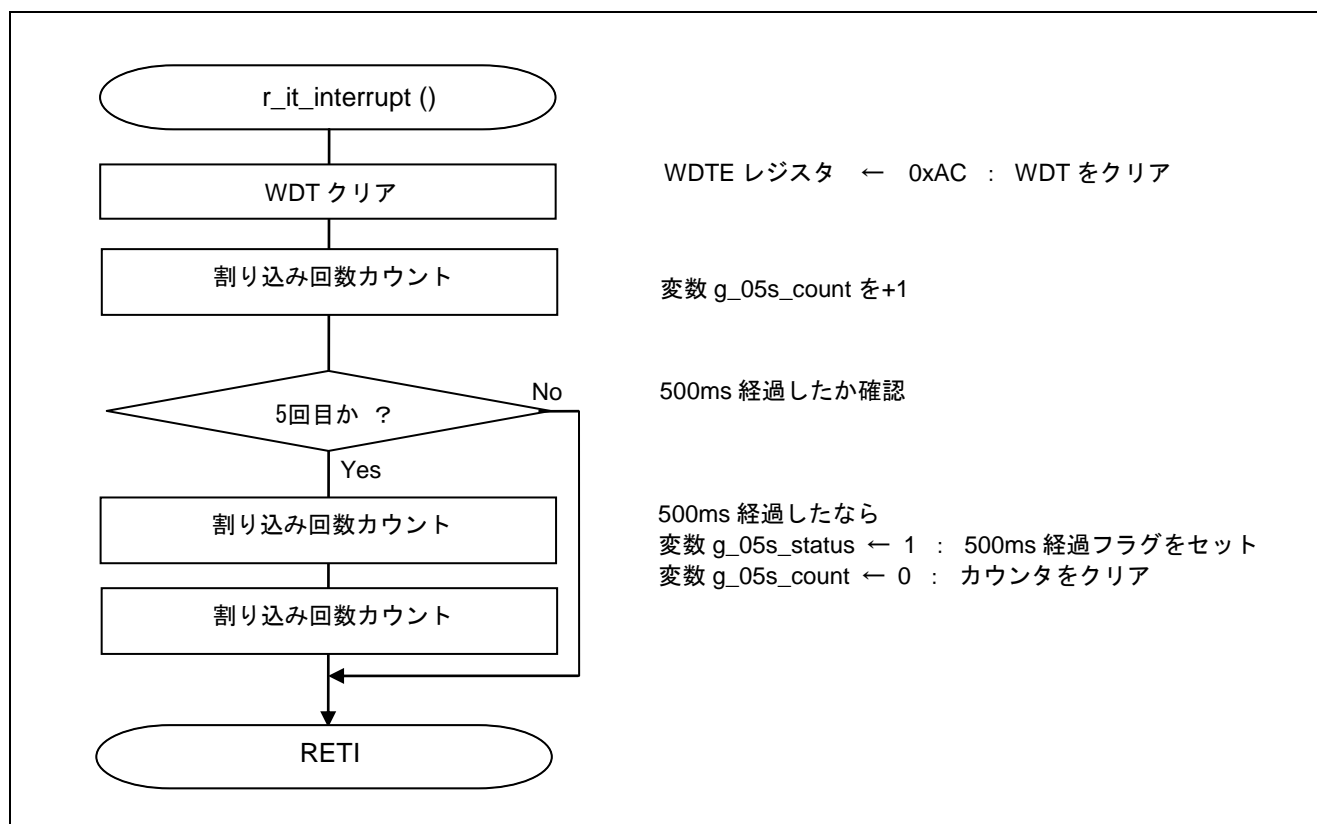


図 6.17 WDT 割り込み処理

## WDT クリア

・ウォッチドッグ・タイマ・イネーブル・レジスタ (WDTE)

WDT をクリア

略号 : WDTE

7	6	5	4	3	2	1	0
1	0	1	0	1	1	0	0

**注意** レジスタ設定方法の詳細については、RL78/G10 ユーザーズマニュアル ハードウェア編を参照してください。

## 6.8.17 IIC00 と TM01 の起動準備

図 6.18 に IIC00 と TM01 の起動準備処理のフローチャートを示します。

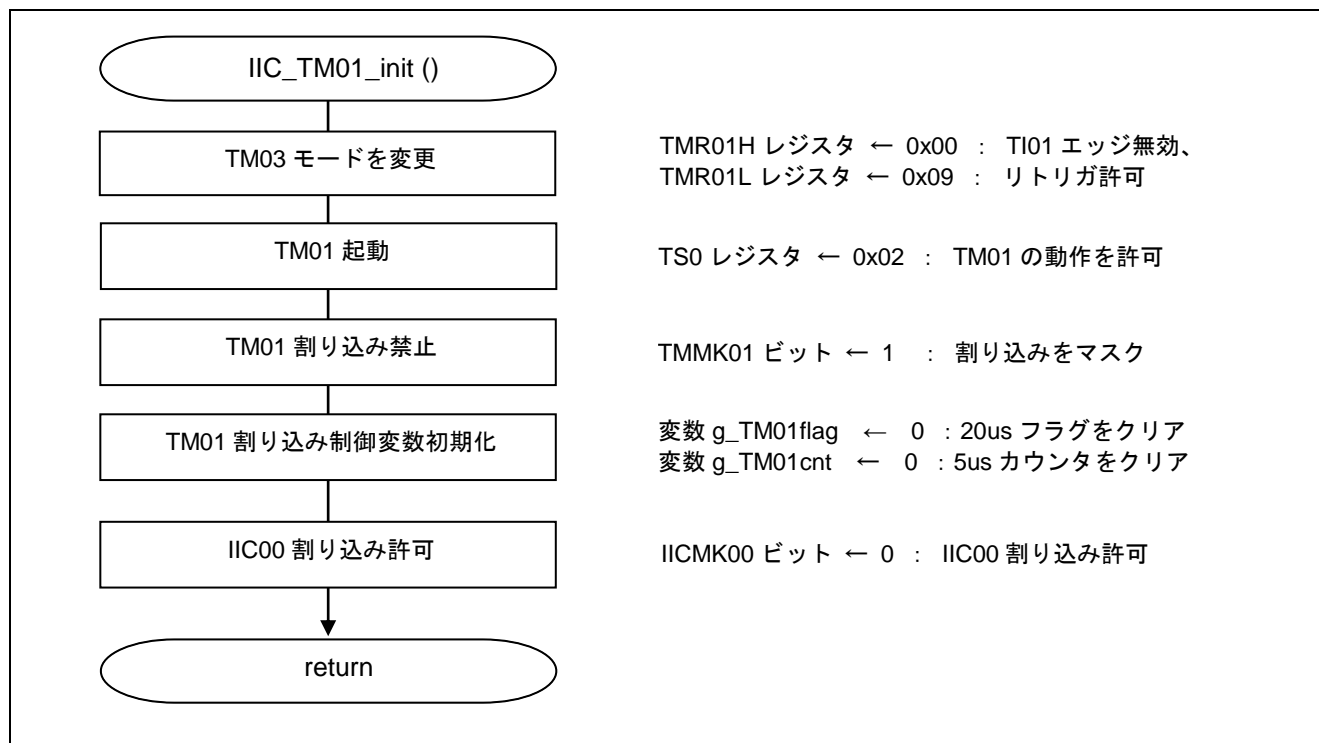


図 6.18 IIC00 と TM01 の起動準備処理

## タイマ 01 起動

- ・タイマ・モード・レジスタ 01 (TMR01)  
チャンネル 01 のエッジ検出禁止に変更  
カウント中の再トリガを許可に変更
- ・タイマ・チャンネル開始レジスタ 0 (TS0)  
チャンネル 01 の動作を許可
- ・割り込みマスク・フラグ・レジスタ (MK0H) の TMMK01 ビット  
TM01 の割り込みをマスクします

略号 : TMR01H、TMR01L

TMR01H								TMR01L							
7	6	15	4	3	2	1	0	7	6	5	4	3	2	1	0
CKS011	0	0	CCS01	SPLIT01	STS012	STS011	STS010	CIS011	CIS010	0	0	MD013	MD012	MD011	MD010
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1

ビット 7

CKS011	チャンネル 1 の動作クロック (f <sub>MCK</sub> ) の選択
0	タイマ・クロック選択レジスタ 0 (TPS0) で設定した動作クロック CK00

ビット 4

CCS01	チャンネル 1 のカウント・クロック (f <sub>TCLK</sub> ) の選択
0	CKS011、CKS010 ビットで指定した動作クロック (f <sub>MCK</sub> )

ビット 3

SPLIT01	チャンネル 1 の 8 ビット・タイマ/16 ビット・タイマ動作の選択
0	16 ビット・タイマとして動作

略号：TMR01H、TMR01L

TMR01H								TMR01L							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
CKS011	0	0	CCS01	SPLIT01	STS012	STS011	STS010	CIS011	CIS010	0	0	MD013	MD012	MD011	MD010
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1

ビット 2－0

STS012	STS011	STS010	チャンネル 1 のスタート・トリガ、キャプチャ・トリガの設定
0	0	0	ソフトウェア・トリガ・スタートのみ有効（他のトリガ要因を非選択にする）

ビット 7－6

CIS011	CIS010	TI01 端子の有効エッジ選択
0	0	立ち下がリエッジ

ビット 3－0

MD013	MD012	MD011	MD010	チャンネル 1 の動作モードの設定	対応する機能	TCR のカウント動作
1	0	0	1	ワンカウント・モード	ディレイ・カウンタ／ワンショット・パルス出力／PWM 出力（スレーブ）	ダウン・カウント

略号：TSH0、TS0

TSH0								TS0							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	TS03H <sup>注</sup>	0	TS01H	0	0	0	0	0	TS03 <sup>注</sup>	TS02 <sup>注</sup>	TS01	TS00
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

注：16 ピン製品のみ

ビット 1

TS01	チャンネル 01 の動作許可トリガ
0	トリガ動作しない
1	TE01 ビットが 1 にセットされ、カウント動作許可状態

略号：MK0H

ビット 0

TMMK01	割り込み処理の制御
0	割り込み処理許可
1	割り込み処理禁止

略号：MK0L

ビット 3

IICMK00	割り込み処理の制御
0	割り込み処理許可
1	割り込み処理禁止

**注意** レジスタ設定方法の詳細については、RL78/G10 ユーザーズマニュアル ハードウェア編を参照してください。

6.8.18 データ送信起動処理

図 6.19 にデータ送信起動処理のフローチャートを示します。

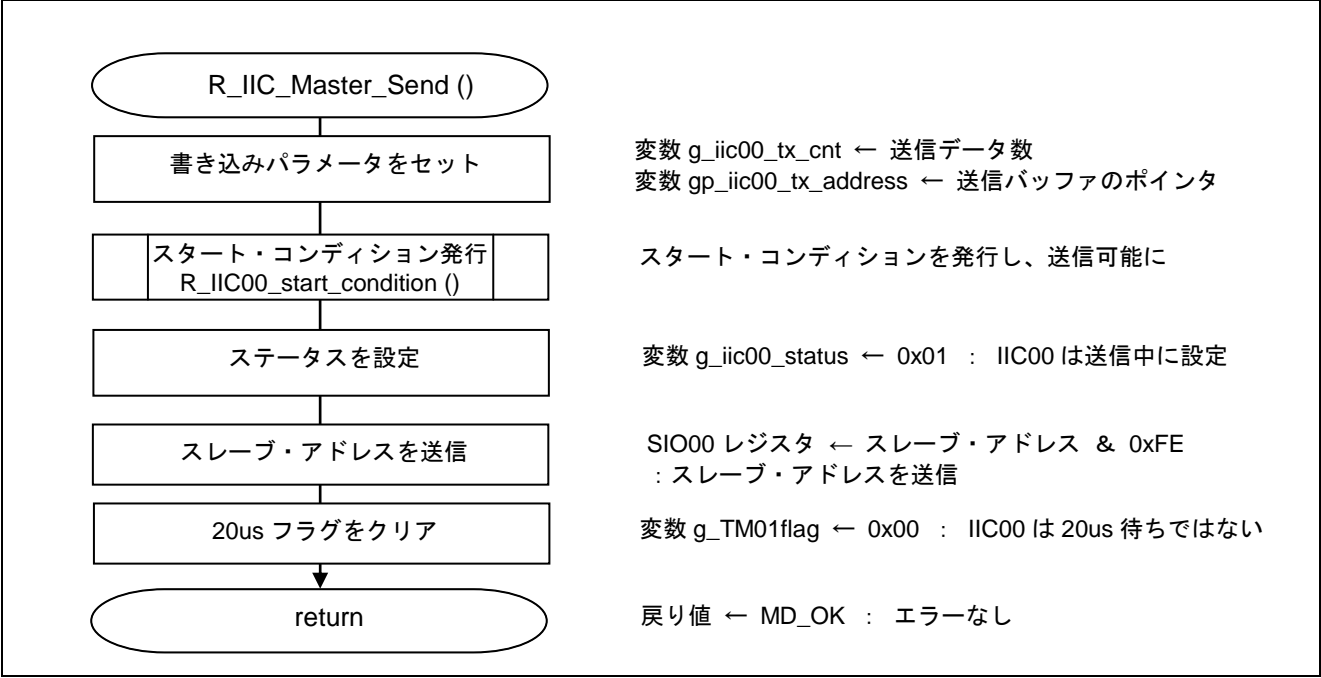


図 6.19 データ送信起動処理

スレーブ・アドレス送信

- ・シリアル・データ・レジスタ 00 (SIO00)
- 送信データ（スレーブ・アドレス）をセットします

略号 : SIO00

7	6	5	4	3	2	1	0
1	0	1	0	0	0	0	0

スレーブ・アドレスは 0xA0 です。

6.8.19     データ受信起動処理

図 6.20 にデータ受信起動処理のフローチャートを示します。

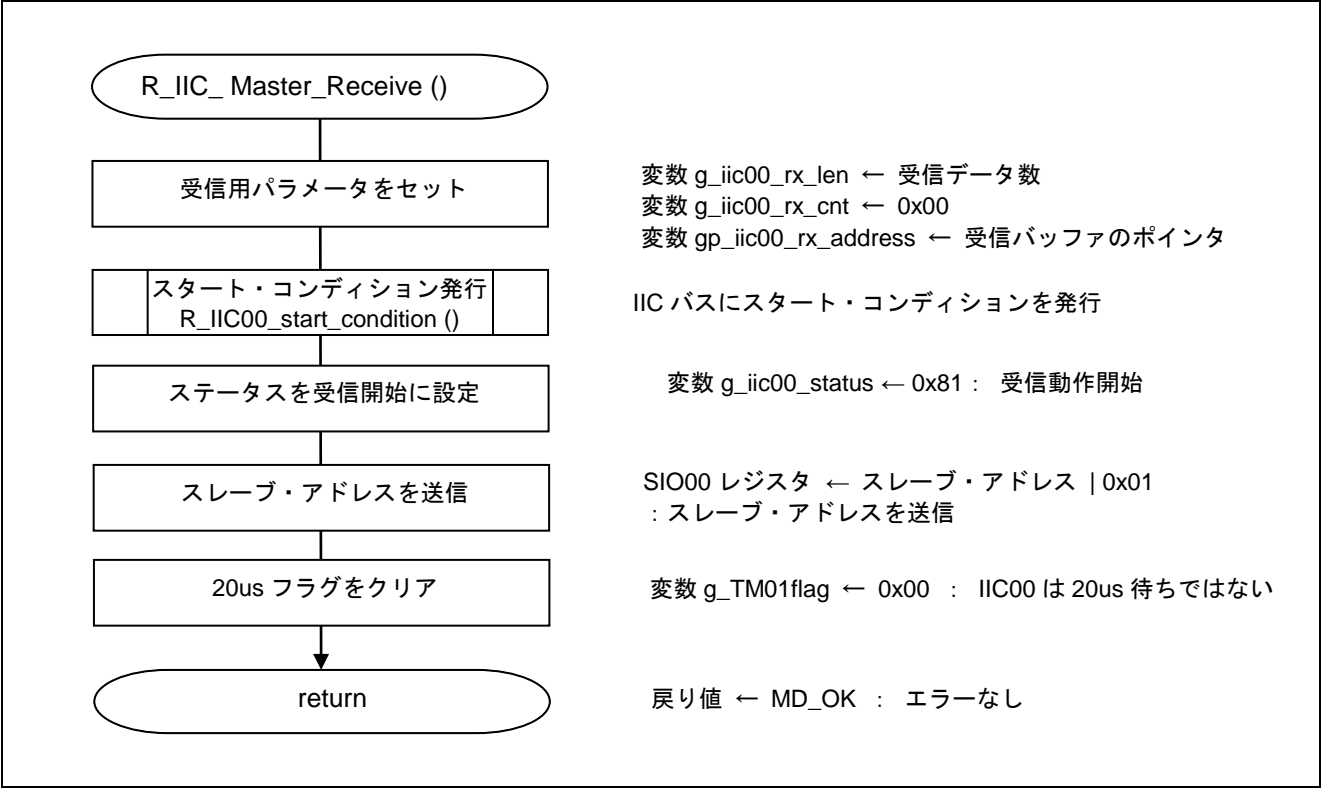


図 6.20     データ受信起動処理

アドレス送信

- ・シリアル・データ・レジスタ 00（SDR00）の下位（SIO00 レジスタ）

送信データ（スレーブ・アドレス）をセットします

略号：SIO00

7	6	5	4	3	2	1	0
1	0	1	0	0	0	0	1

スレーブ・アドレスは 0xA1 です。



## 6.8.20 IIC00 送信完了待ち

図 6.21 に IIC00 送信完了待ち処理のフローチャートを示します。

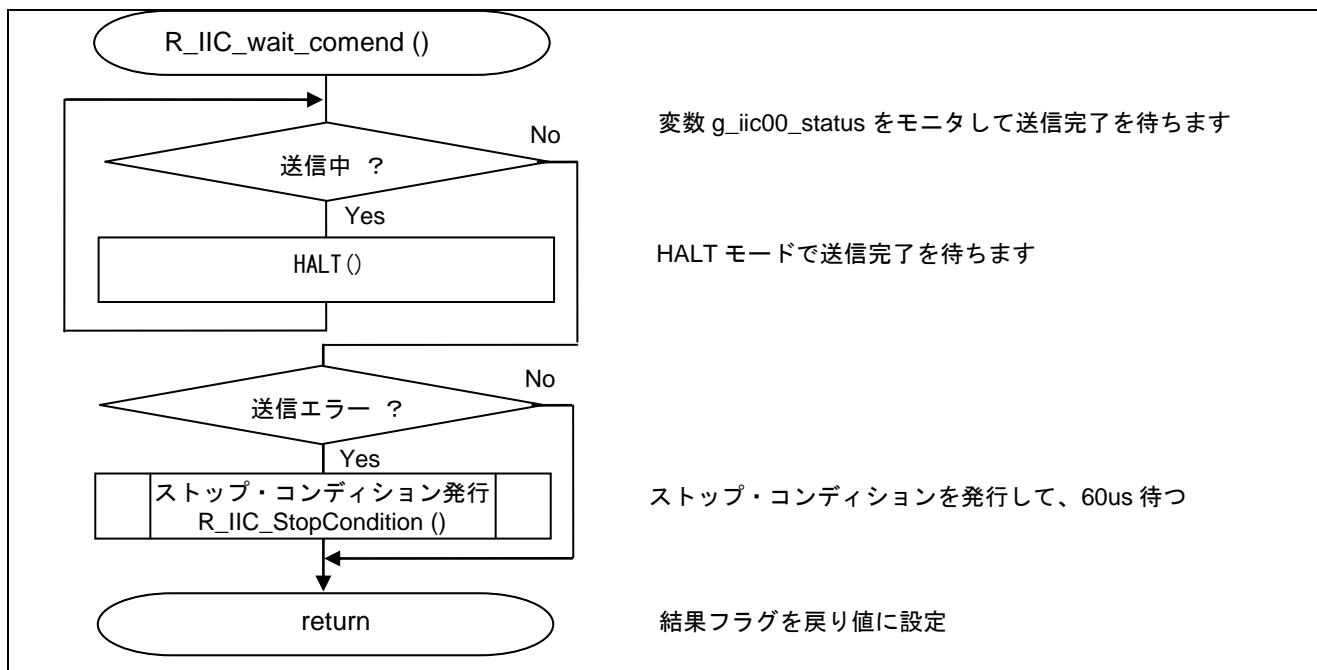


図 6.21 IIC00 送信完了待ち処理

## 6.8.21 IIC00 状態確認

図 6.22 に IIC00 状態確認処理のフローチャートを示します。

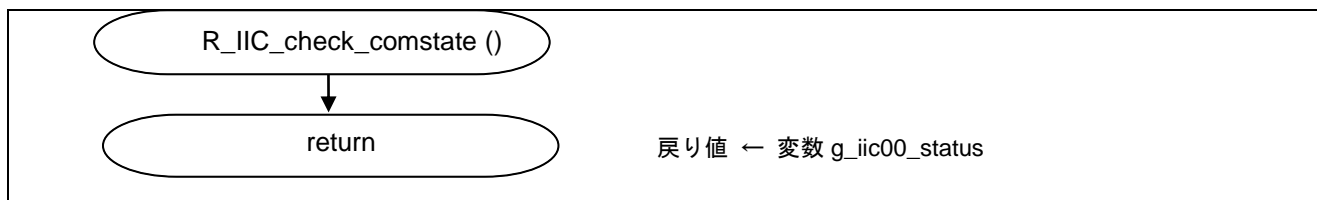


図 6.22 IIC00 状態確認処理

## 6.8.22 ストップ・コンディション発行処理

図 6.23 にストップ・コンディション発行処理のフローチャートを示します。

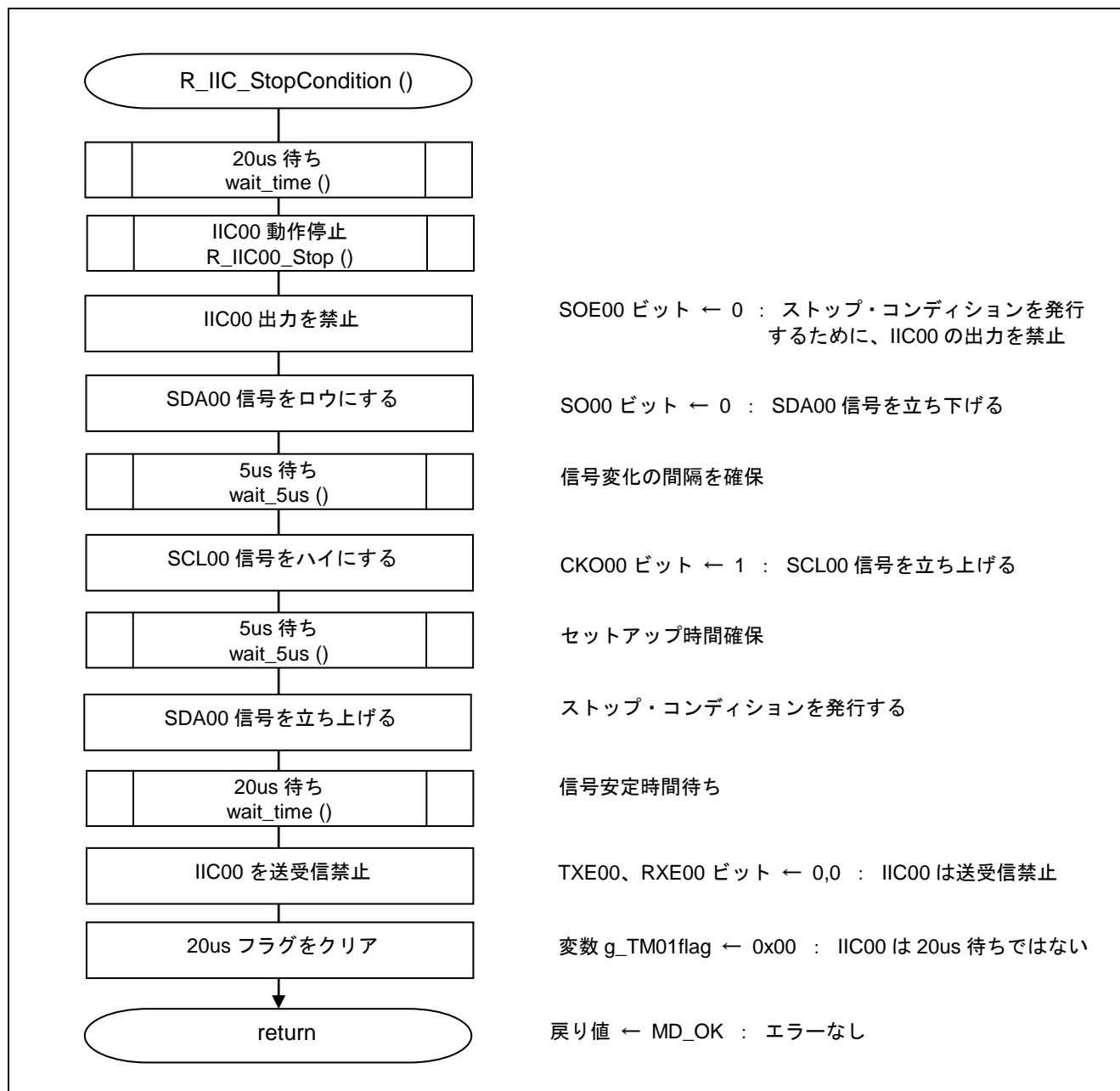


図 6.23 ストップ・コンディション発行処理

## IIC00 出力を禁止

- ・シリアル出力許可レジスタ 0 (SOE0) の SOE00 ビット
- IIC00 の出力を禁止します

略号：SOE0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	SOE01	SOE00
0	0	0	0	0	0	x	0

ビット 0

SOE00	チャンネル 0 のシリアル出力許可／停止
0	シリアル通信動作による出力停止
1	シリアル通信動作による出力許可

## SDA00、SCL00 信号の制御

- ・シリアル・クロック出力レジスタ 0（CKO0）、シリアル出力レジスタ 0（SO0）の CKO00 ビットと SO00 ビット
- SDA00 信号を立ち下げます
- SCL00 信号を立ち上げます
- SDA00 信号を立ち上げます

略号：CKO0、SO0

CKO0								SO0							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	0	0	0	0	0	CKO01	CKO00	0	0	0	0	0	0	SO01	SO00
0	0	0	0	0	0	x	<b>0</b>	0	0	0	0	0	0	0	<b>0/1</b>

ビット 0

SO00	チャンネル 0 のシリアル・データ出力
0	シリアル・データ出力値が “0”
1	シリアル・データ出力値が “1”

ビット 0

CKO00	チャンネル 0 のシリアル・クロック出力
0	シリアル・クロック出力値が “0”
1	シリアル・クロック出力値が “1”

## IIC00 の送受信禁止

- ・シリアル通信動作設定レジスタ 00H（SCR00H）の TXE00 ビットと RXE00 ビット
- 送信を禁止します
- 受信を禁止します

略号：SCR00H

15	14	13	12	11	10	9	0
TXE00	RXE00	DAP00	CKP00	0	EOC00	PTC001	PTC000
0	0	0	0	0	0	0	0

ビット 7、6

TXE00	RXE00	チャンネル 0 の動作モードの設定
0	0	通信禁止
0	1	受信のみを行う
1	0	送信のみを行う
1	1	送受信を行う（簡易 IIC では設定禁止）

**注意** レジスタ設定方法の詳細については、RL78/G10 ユーザーズマニュアル ハードウェア編を参照してください。

## 6.8.23 コマンド処理待ち

図 6.24 にコマンド処理待ち処理のフローチャートを示します。

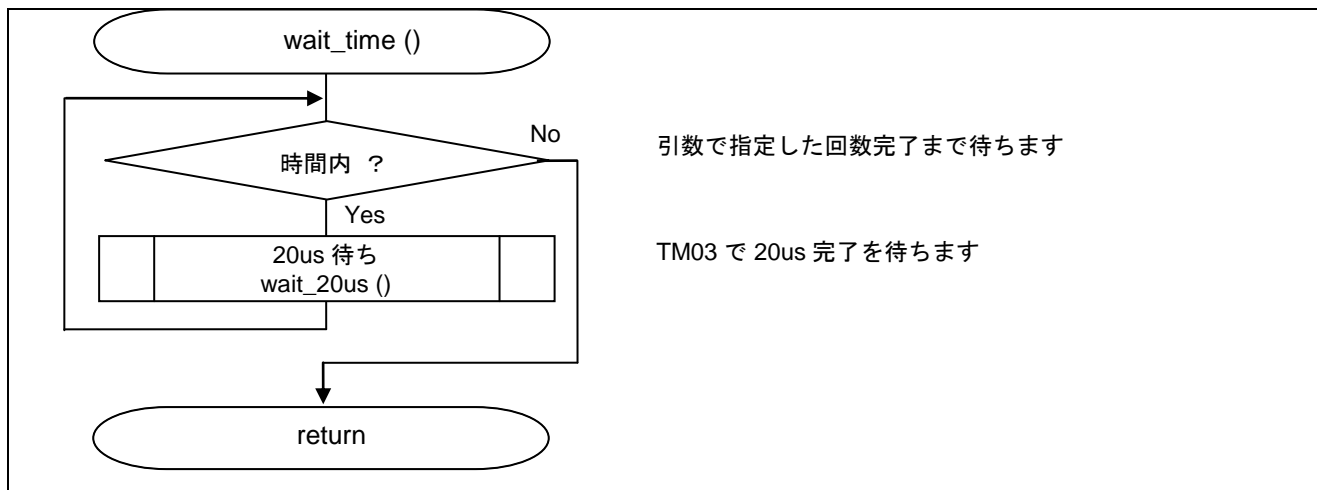


図 6.24 コマンド処理待ち処理

## 6.8.24 20us 待ち

図 6.25 に 20us 待ち処理のフローチャートを示します。

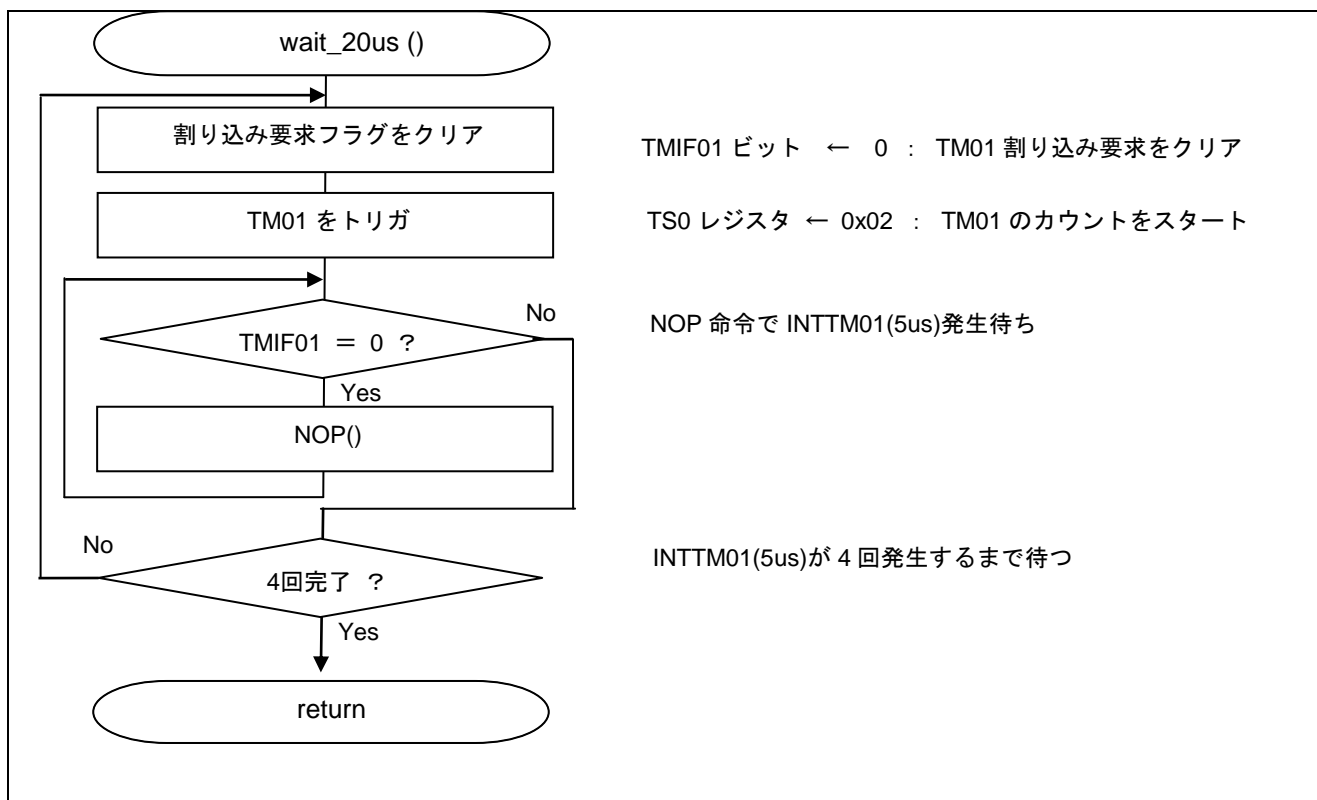


図 6.25 20us 待ち処理

## タイマ 01 起動トリガ

- ・割り込み要求フラグ・レジスタ (IF0H) の TMIF01 ビット  
TM01 の割り込み要求をクリアします
- ・タイマ・チャンネル開始レジスタ 0 (TS0)  
チャンネル 01 のカウント動作を開始

略号：IF0H

ビット 0

TMIF01	割り込み要求フラグ
<b>0</b>	割り込み要求信号が発生していない
1	割り込み要求信号が発生し、割り込み要求状態

略号：TS0

7	6	5	4	3	2	1	0
0	0	0	0	TS03 注	TS02 注	TS01	TS00
0	0	0	0	0	0	<b>1</b>	0

注：16 ピン製品のみ

ビット 1

TS01	チャネル 01 の動作許可トリガ
0	トリガ動作しない
<b>1</b>	カウント動作を開始トリガする

## 6.8.25 スタート・コンディション発行処理

図 6.26 にスタート・コンディション発行処理のフローチャートを示します。

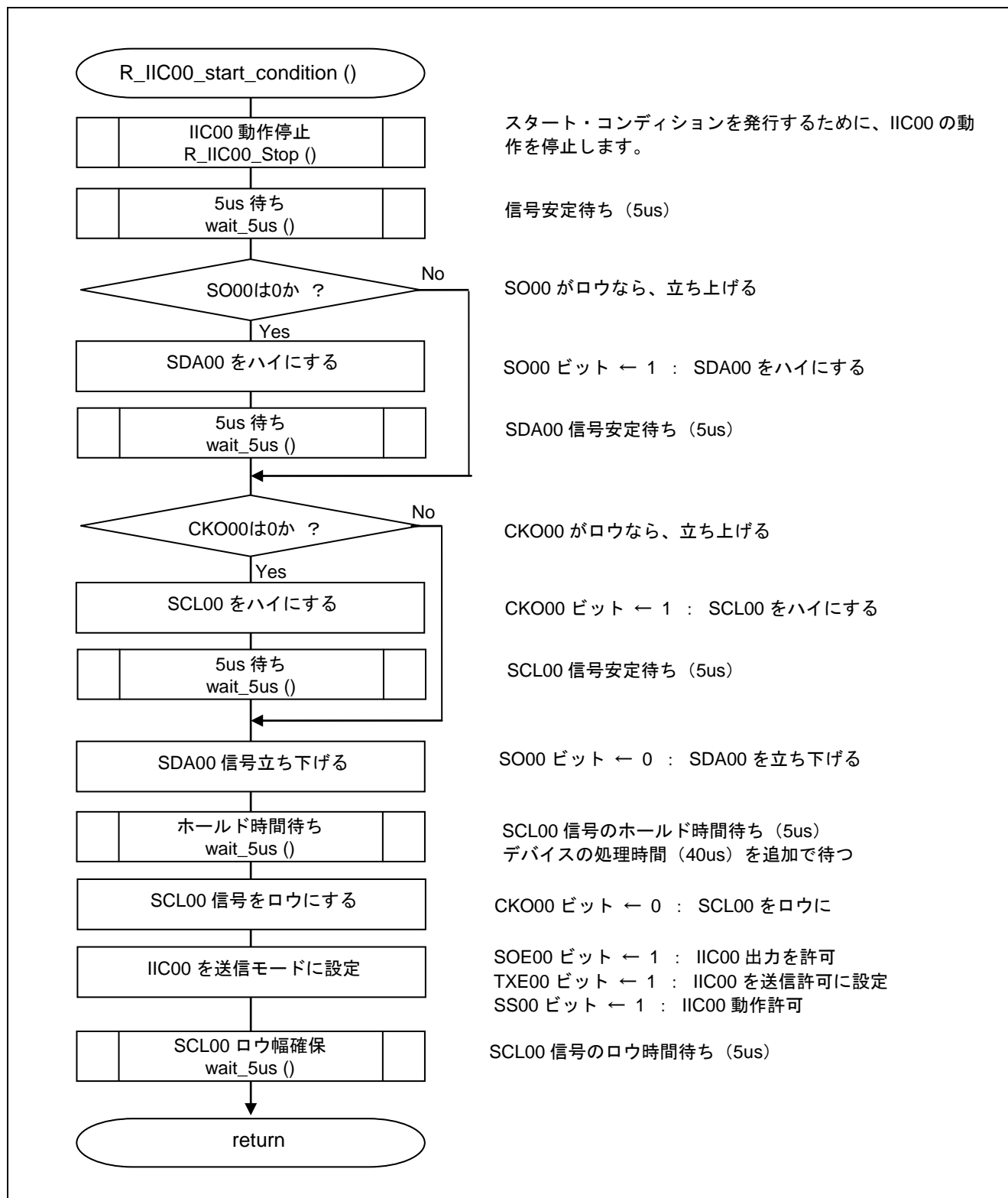


図 6.20 スタート・コンディション発行処理

## SO00 状態確認と立ち上げ

- ・シリアル出力レジスタ（SO0）の SO00 ビット  
SDA00 の確認及び立ち上げ

略号：SO0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	SO01	SO00
0	0	0	0	0	0	0	<b>1</b>

## ビット 0

SO00	チャンネル 0 のシリアル・データ出力
0	シリアル・データ出力値が “0”
<b>1</b>	シリアル・データ出力値が “1”

## CKO00 状態確認と立ち上げ

- ・シリアル出力レジスタ（SO0）の CKO00 ビット  
SCL00 の確認及び立ち上げ

略号：SO0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	CKO01	CKO00	0	0	0	0	0	0	SO01	SO00
0	0	0	0	0	0	x	<b>1</b>	0	0	0	0	0	0	0	1

## ビット 8

CKO00	チャンネル 0 のシリアル・クロック出力
0	シリアル・クロック出力値が “0”
<b>1</b>	シリアル・クロック出力値が “1”

## スタート・コンディション発行

- ・シリアル出力レジスタ（SO0）の SO00 ビット  
SDA00 の立ち下げ

略号：SO0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	SO01	SO00
0	0	0	0	0	0	0	<b>0</b>

## ビット 0

SO00	チャンネル 0 のシリアル・データ出力
<b>0</b>	シリアル・データ出力値が “0”
1	シリアル・データ出力値が “1”

## 送信準備

- ・シリアル・クロック出力レジスタ（CKO0）の CKO00 ビット  
SCL00 の立ち下げ

略号：CKO0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	CKO01	CKO00
0	0	0	0	0	0	x	<b>0</b>

## ビット 0

CKO00	チャンネル 0 のシリアル・クロック出力
<b>0</b>	シリアル・クロック出力値が “0”
1	シリアル・クロック出力値が “1”

## IIC00 送信モード設定

- ・シリアル出力許可レジスタ 0 (SOE0) の SOE00 ビット  
IIC00 出力を許可します
- ・シリアル通信動作設定レジスタ 00H (SCR00H) の TXE00 ビット  
IIC00 の送信を許可します
- ・シリアル・チャンネル開始レジスタ 0 (SS0) の SS00 ビット  
IIC00 の動作を許可します

略号 : SOE0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	SOE01	SOE00
0	0	0	0	0	0	x	1

ビット 0

SOE00	チャンネル 0 のシリアル出力許可／停止
0	シリアル通信動作による出力停止
1	シリアル通信動作による出力許可

略号 : SCR00H

7	6	5	4	3	2	1	0
TXE00	RXE00	DAP00	CKP00	0	EOC00	PTC001	PTC000
1	0	0	0	0	0	0	0

ビット 7、6

TXE00	RXE00	チャンネル 0 の動作モードの設定
0	0	通信禁止
0	1	受信のみを行う
1	0	送信のみを行う
1	1	送受信を行う（簡易 IIC では設定禁止）

略号 : SS0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	SS01	SS00
0	0	0	0	0	0	x	1

ビット 0

SS00	チャンネル 0 の動作開始トリガ
0	トリガ動作せず
1	SE00 ビットに 1 をセットし、通信待機状態に移移する

**注意** レジスタ設定方法の詳細については、RL78/G10 ユーザーズマニュアル ハードウェア編を参照してください。



## 6.8.26 IIC00 動作停止処理

図 6.27 に IIC00 動作停止処理のフローチャートを示します。

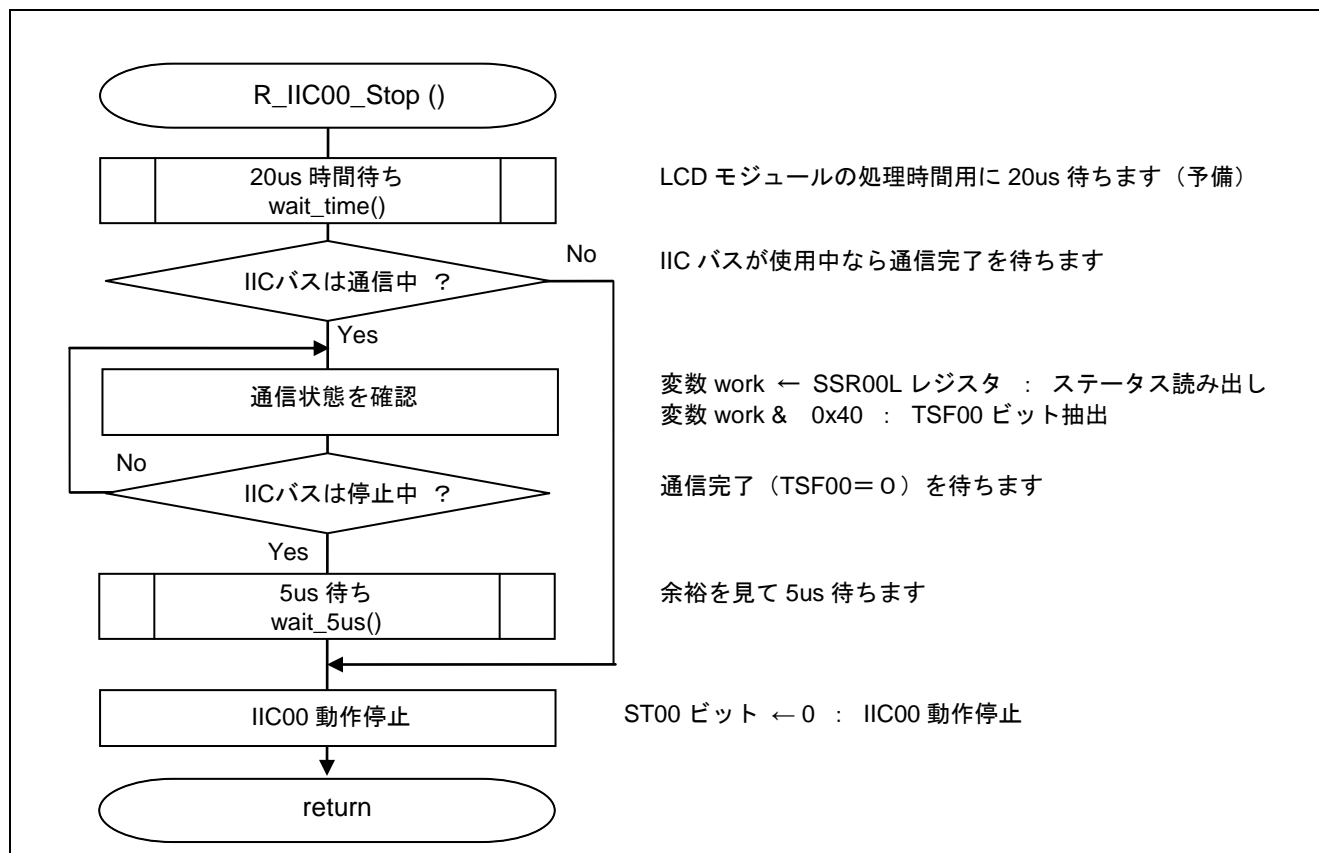


図 6.27 IIC00 動作停止処理

## IIC00 通信状態確認

- ・シリアル・ステータス・レジスタ 00 (SSR00) の TSF00 ビット
- 通信完了を待ちます

略号 : SSR00

7	6	5	4	3	2	1	0
0	TSF00	BFF00	0	0	FEF00	PEF00	OVF00
0	<b>0</b>	0	0	0	x	x	x

## ビット 5

TSF00	チャンネル 0 の通信状態表示フラグ
<b>0</b>	通信動作停止状態または通信動作待機状態
1	通信動作状態

## IIC00 動作停止

- ・シリアル・チャンネル停止レジスタ 0 (ST0) の ST00 ビット
- チャンネル 0 を停止します。

略号 : ST0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	ST01	ST00
0	0	0	0	0	0	x	<b>1</b>

## ビット 0

ST00	チャンネル 0 の動作停止トリガ
0	トリガ動作せず
<b>1</b>	SE00 ビットを 0 にクリアし、通信動作を停止する

## 6.8.27 5us 待ち処理

図 6.28 に 5us 待ち処理のフローチャートを示します。

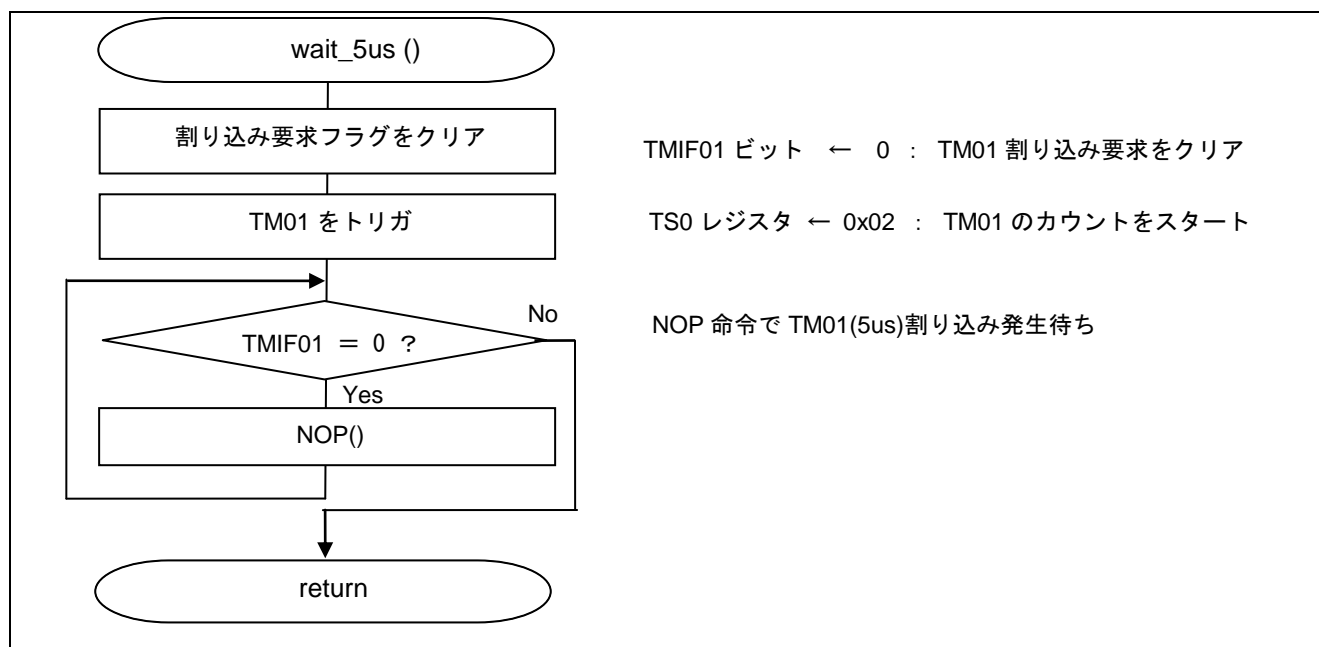


図 6.28 5us 待ち処理

## タイマのカウント完了割り込みのクリア

- ・割り込み要求フラグ・レジスタ（IF0H）の TMIF01 ビット  
割り込み要求フラグのクリア

略号：IF0H

ビット 0

TMIF01	割り込み要求フラグ
0	割り込み要求信号が発生していない
1	割り込み要求信号が発生し、割り込み要求状態

## タイマ 01 カウント開始

- ・タイマ・チャンネル開始レジスタ 0（TS0）  
チャンネル 01 のカウントを開始

略号：TS0

7	6	5	4	3	2	1	0
0	0	0	0	TS03	TS02	TS01	TS00
0	0	0	0	0	0	1	0

ビット 1

TS01	チャンネル 01 の動作許可トリガ
0	トリガ動作しない
1	カウント動作を開始トリガする

**注意** レジスタ設定方法の詳細については、RL78/G10 ユーザーズマニュアル ハードウェア編を参照してください。

6.8.28 IIC00 割り込み処理

図 6.29 に IIC00 割り込み処理のフローを示します。

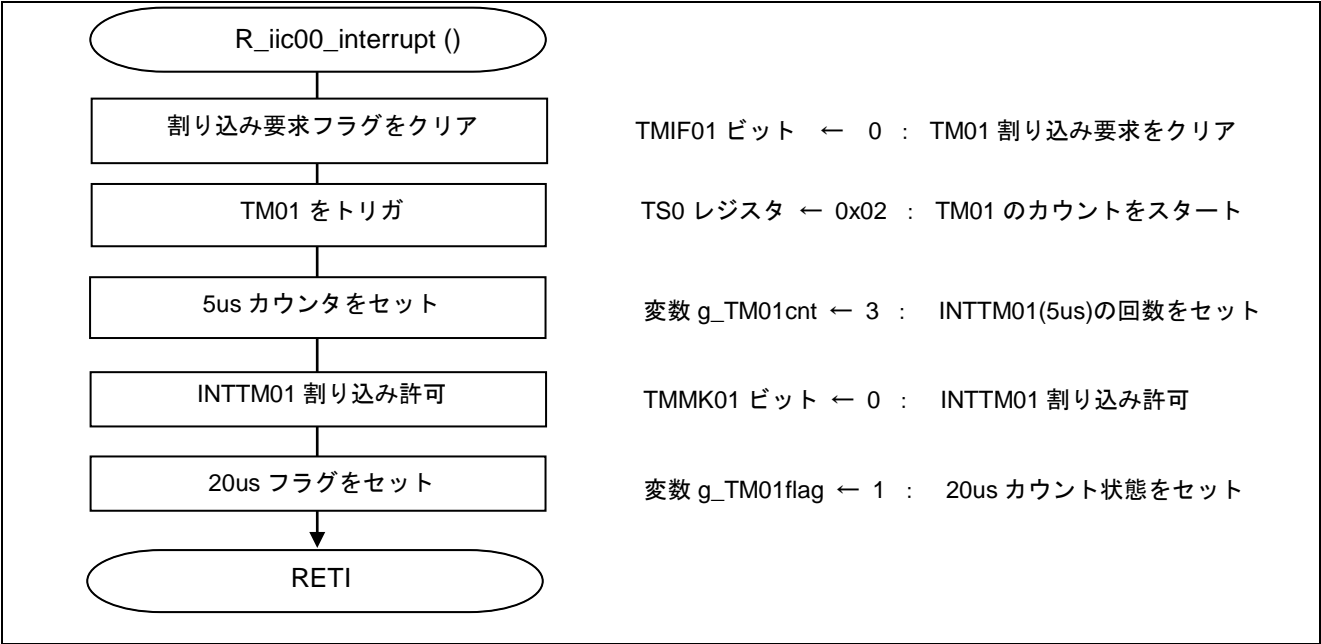


図 6.29 IIC00 割り込み処理

タイマのカウント完了割り込みのクリア

- ・割り込み要求フラグ・レジスタ（IF0H）の TMIF01 ビット  
割り込み要求フラグのクリア

略号：IF0H

ビット 0

TMIF01	割り込み要求フラグ
0	割り込み要求信号が発生していない
1	割り込み要求信号が発生し、割り込み要求状態

タイマ 01 起動

- ・タイマ・チャンネル開始レジスタ 0（TS0）  
チャンネル 01 のカウントを起動

略号：TS0

7	6	5	4	3	2	1	0
0	0	0	0	TS03	TS02	TS01	TS00
0	0	0	0	0	0	1	0

ビット 1

TS01	チャンネル 01 の動作許可トリガ
0	トリガ動作しない
1	カウント動作を開始トリガする

タイマ割り込みの許可

略号：MK0H

ビット 0

TMMK01	割り込み処理の制御
0	割り込み処理許可
1	割り込み処理禁止

## 6.8.29 TM01 割り込み処理

図 6.30～図 6.32 に TM01 割り込み処理のフローを示します。

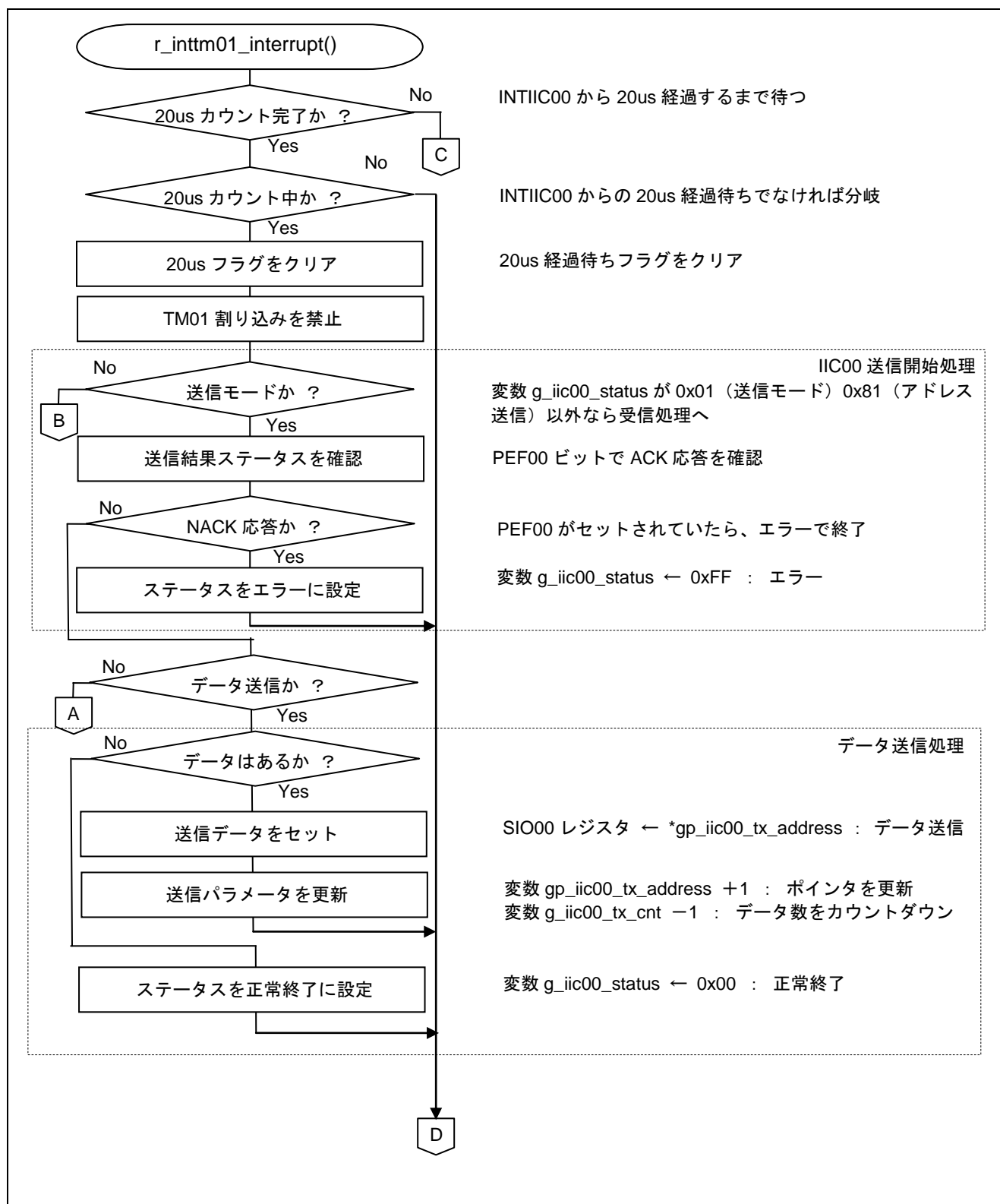


図 6.30 TM03 割り込み処理（1 / 3）

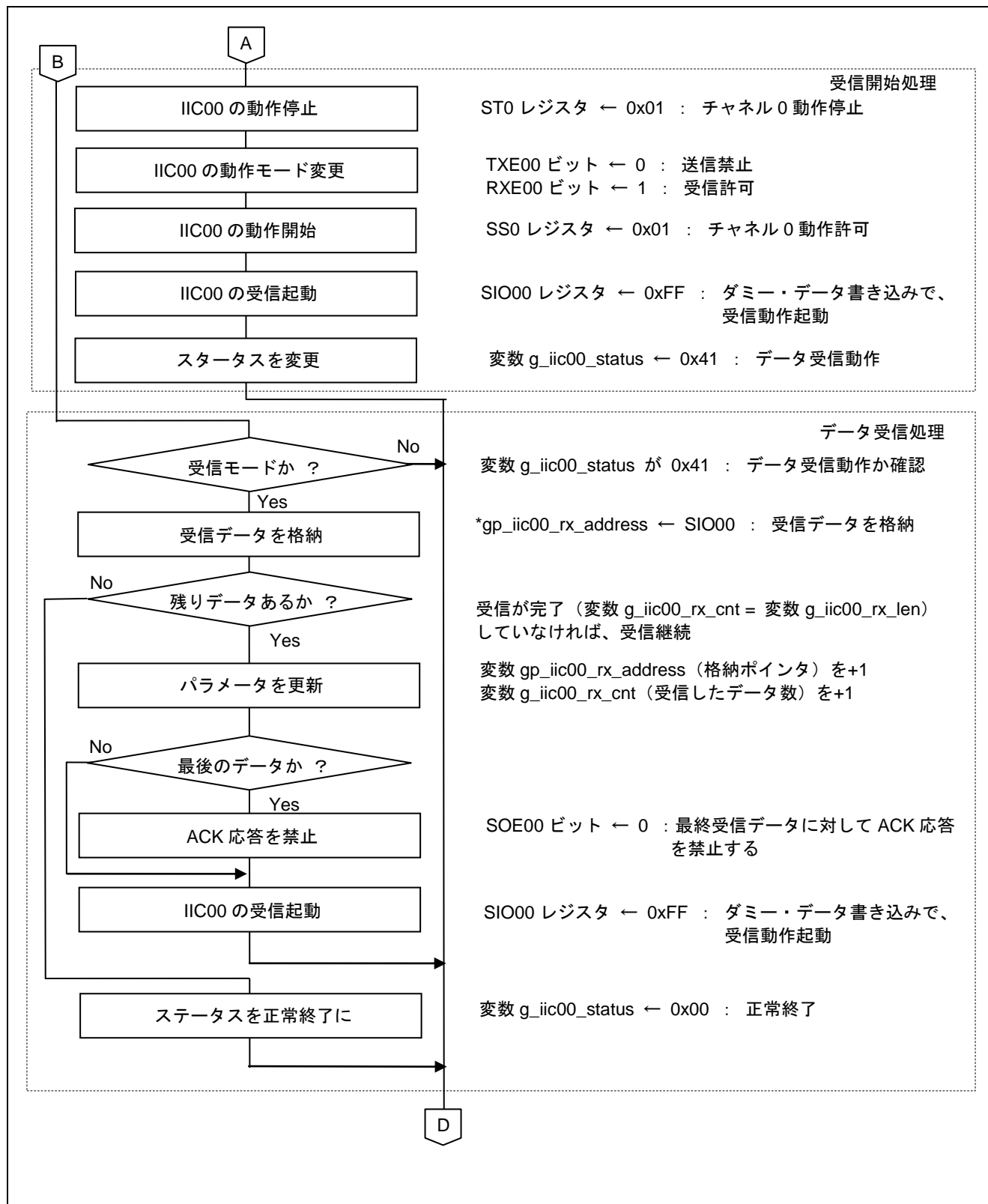


図 6.31 TM01 割り込み処理（2 / 3）

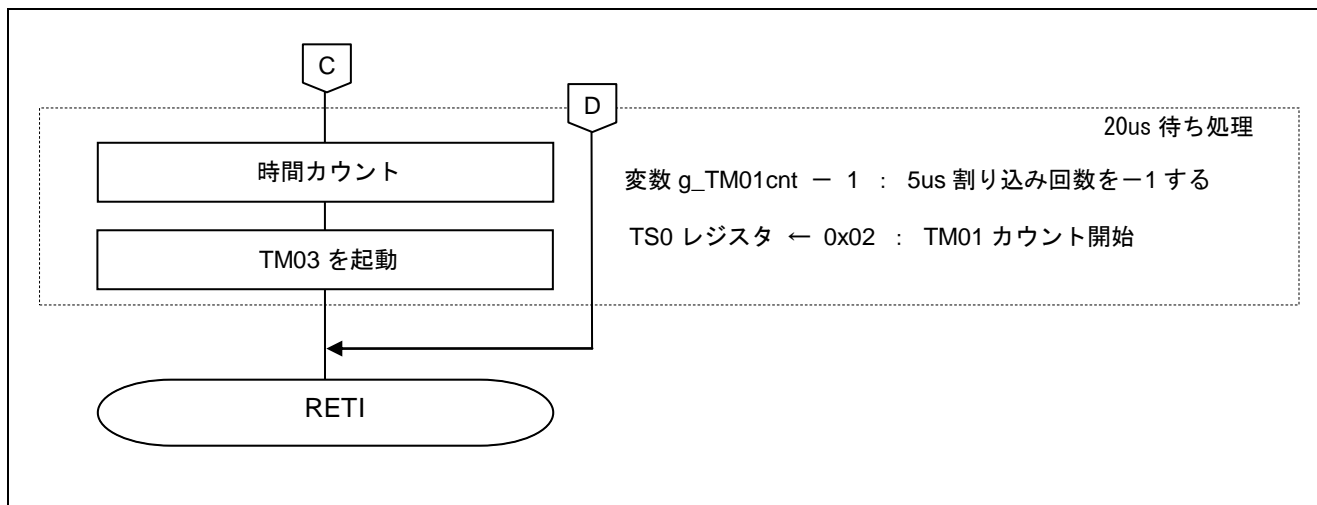


図 6.32 TM03 割り込み処理（3／3）

## IIC00 のステータス確認

- ・シリアル・ステータス・レジスタ 00（SSR00）の PEF00 ビット  
ACK 応答が確認

略号：SSR00

7	6	5	4	3	2	1	0
0	TSF00	BFF00	0	0	FEF00	PEF00	OVF00
0	x	x	0	0	x	<b>0/1</b>	x

ビット 1

PEF00	チャンネル 0 のパリティ・エラー検出フラグ
<b>0</b>	スレープから ACK 応答
<b>1</b>	スレープから NACK 応答

## IIC00 エラー・フラグのクリア

- ・シリアル・フラグ・クリア・トリガ・レジスタ 00（SIR00）  
エラー・フラグをクリアします

略号：SIR00

7	6	5	4	3	2	1	0
0	0	0	0	0	FECT00	PECT00	OVCT00
0	0	0	0	0	<b>0/1</b>	<b>0/1</b>	<b>0/1</b>

ビット 1

PECT00	チャンネル 0 のパリティ・エラー・フラグのクリア・トリガ
<b>0</b>	クリアしない
<b>1</b>	SSR00 レジスタの PEF00 ビットを 0 にクリアする

## データ送信

- ・シリアル・データ・レジスタ 00（SDR00）の下位（SIO00 レジスタ）  
送信データをセットします

略号：SIO00

7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x

## IIC00 動作停止

- ・シリアル・チャンネル停止レジスタ 0 (ST0) の ST00 ビット  
チャンネル 0 を停止します。

略号 : ST0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	ST01	ST00
0	0	0	0	0	0	0	<b>1</b>

ビット 0

ST00	チャンネル 0 の動作停止トリガ
0	トリガ動作せず
<b>1</b>	<b>SE00 ビットを 0 にクリアし、通信動作を停止する</b>

## IIC00 動作モード変更

- ・シリアル通信動作設定レジスタ (SCR00H) の TXE00 ビットと RXE00 ビット  
送信を禁止し、受診を許可します。

略号 : SCR00H

15	14	13	12	11	10	9	0
TXE00	RXE00	DAP00	CKP00	0	EOC00	PTC001	PTC000
<b>0</b>	<b>0</b>	0	0	0	0	0	0

ビット 7、6

TXE00	RXE00	チャンネル 0 の動作モードの設定
0	0	通信禁止
<b>0</b>	<b>1</b>	<b>受信のみを行う</b>
1	0	送信のみを行う
1	1	送受信を行う（簡易 IIC では設定禁止）

## IIC00 動作開始

- ・シリアル・チャンネル開始レジスタ 0 (SS0) の SS00 ビット  
チャンネル 0 を動作開始します

略号 : SS0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	SS01	SS00
0	0	0	0	0	0	0	<b>1</b>

ビット 0

SS00	チャンネル 0 の動作開始トリガ
0	トリガ動作せず
<b>1</b>	<b>SE00 ビットに 1 をセットし、通信待機状態に遷移する</b>

## データ受信の起動

- ・シリアル・データ・レジスタ 00 (SDR00) の下位 (SIO00 レジスタ)  
ダミー・データをセットします

略号 : SIO00

7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	1



## 受信データの読み出し

- ・シリアル・データ・レジスタ 00（SDR00）の下位（SIO00 レジスタ）  
受信したデータを読み出します

略号：SIO00

7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	1

## ACK 応答の禁止

- ・シリアル出力許可レジスタ 0（SOE0）の SOE00 ビット  
シリアル出力を禁止します。

略号：SOE0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	SOE01	SOE00
0	0	0	0	0	0	x	<b>0</b>

## ビット 0

SOE00	チャンネル 0 のシリアル出力許可／停止
<b>0</b>	シリアル通信動作による出力停止
1	シリアル通信動作による出力許可

## ACK 応答の禁止

- ・シリアル出力許可レジスタ 0（SOE0）の SOE00 ビット  
シリアル出力を禁止します。

## タイマ 01 起動

- ・タイマ・チャンネル開始レジスタ 0（TS0）  
チャンネル 01 のカウントを起動

略号：TS0

7	6	5	4	3	2	1	0
0	0	0	0	TS03	TS02	TS01	TS00
0	0	0	0	<b>1</b>	0	0	0

## ビット 1

TS01	チャンネル 01 の動作許可トリガ
0	トリガ動作しない
<b>1</b>	カウント動作を開始トリガする

## 7. CA-78K0R への移植について

### 7.1 追加修正するファイル

このサンプルコードの中で、「6.2 コード移植対象プログラム生成での設定内容」で示した設定でコード生成したコードに対して追加／変更するのは以下の3つのファイルのみです。

- ・ r\_cg\_main.c
- ・ r\_IIC\_LIB.c
- ・ r\_IIC\_LIB.h

#### 7.1.1 準備作業

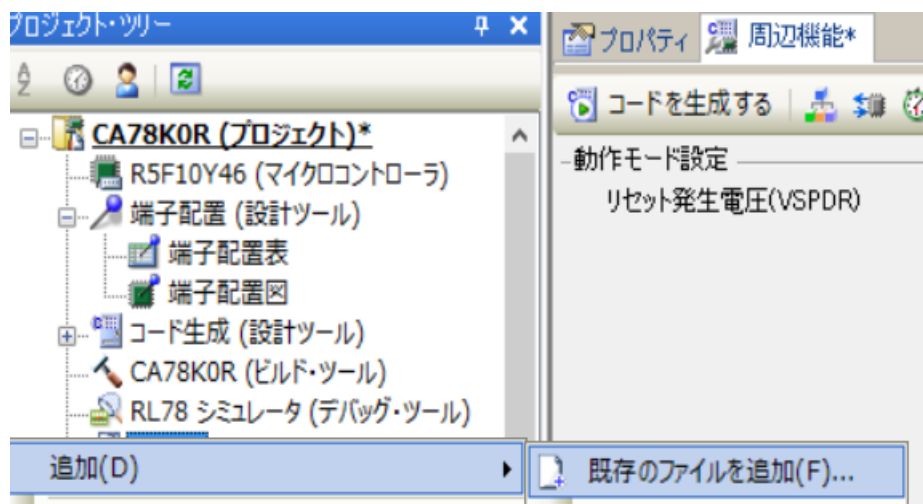
CS+CA-78K0R を起動し、「6.2 コード生成での設定内容」に従って、初期設定のみでコード生成しておきます。

#### 7.1.2 ファイルのコピー

“r\_IIC\_LIB.c”と”r\_IIC\_LIB.h”の2つのファイルをコピーします。

#### 7.1.3 コピーしたファイルのプロジェクトへの追加

プロジェクト・ツリーの「ファイル」をクリックして、「追加」を選択し、「既存のファイルを追加(F)...」を選択して、上で追加したファイルをプロジェクトに追加します。



#### 7.1.4 “r\_IIC\_LIB.c”の修正

36～41 行目の pragma 指令部分を 29～35 行目の include 指定部分の前に移動します。

### 7.2 r\_cg\_main.c の変更

コード生成だけでは、main 関数等の中身がないので、CC-RL のプロジェクトの r\_cg\_main.c を元にして作成します。

#### 7.2.1 CC-RL プロジェクトからのコピー

CC-RL のプロジェクトの cg\_src フォルダにある r\_cg\_main.c をコピーして、コード生成で作成された r\_cg\_main.c を置き換えます。

### 7.2.2 インクルード・ファイルの追加

43～49 行目にある下記の部分を 29 行目のインクルード宣言の前に移動します。

```
/******  
Pragma directive  
*****/  
/* Start user code for pragma. Do not edit comment generated here */  
#pragma interrupt INTWDTI r_it_interrupt  
/* End user code. Do not edit comment generated here */
```

## 7.3 ビルド

以上の変更が完了したら、ビルドします。

## 8. 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。

## 9. 参考ドキュメント

RL78/G10 ユーザーズマニュアル ハードウェア編 Rev.3.00 (R01UH0384J)

RL78 ファミリ ユーザーズマニュアル ソフトウェア編 Rev.2.00 (R01US0015J)

(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。)

## ホームページとサポート窓口

ルネサス エレクトロニクスホームページ

<http://japan.renesas.com/>

お問合せ先

<http://japan.renesas.com/inquiry>

改訂記録	RL78/G10 ソフトウェア（簡易 IIC による IIC バス制御（割り込み編））
------	--

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2015.12.30	—	初版発行

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。
--------------------------------

## 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、本文を参照してください。なお、本マニュアルの本文と異なる記載がある場合は、本文の記載が優先するものとします。

### 1. 未使用端子の処理

【注意】未使用端子は、本文の「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。未使用端子は、本文「未使用端子の処理」で説明する指示に従い処理してください。

### 2. 電源投入時の処置

【注意】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. リザーブアドレスのアクセス禁止

【注意】リザーブアドレスのアクセスを禁止します。

アドレス領域には、将来の機能拡張用に割り付けられているリザーブアドレスがあります。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。

### 4. クロックについて

【注意】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。

### 5. 製品間の相違について

【注意】型名の異なる製品に変更する場合は、事前に問題ないことをご確認下さい。

同じグループのマイコンでも型名が違うと、内部メモリ、レイアウトパターンの相違などにより、特性が異なる場合があります。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。



