

## RL78/G11

R01AN4903JJ0100

## 屋内用換気警報機 GCC

Rev.1.00  
2019.10.03

## 要旨

本アプリケーションノートでは、IDT 社ガスセンサモジュール(ZMOD4410)を RL78/G11 で制御し、屋内空気質を測定する例を説明します。RL78/G11 のシリアル・インタフェース IICA を使用して ZMOD4410 制御と LCD キャラクタディスプレイ制御を実現します。

## 動作確認デバイス

RL78/G11

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 .....                      | 3  |
| 1.1 室内空気質 (IAQ) .....            | 4  |
| 1.2 ZMOD4410 用 I2C 通信プロトコル ..... | 4  |
| 1.3 ZMOD4410 用ライブラリ .....        | 4  |
| 1.4 LCD キャラクタディスプレイ .....        | 4  |
| 1.5 警告灯 .....                    | 5  |
| 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 フォルダ構成 .....                 | 8  |
| 4.2 continuous.c の変更点 .....      | 9  |
| 4.3 r_cg_iica.c の変更点 .....       | 10 |
| 4.4 動作概要 .....                   | 11 |
| 4.5 オプション・バイトの設定一覧 .....         | 12 |
| 4.6 定数一覧 .....                   | 12 |
| 4.7 関数一覧 .....                   | 13 |
| 4.8 関数仕様 .....                   | 14 |
| 5. サンプルコード .....                 | 23 |

1. 仕様

本アプリケーションノートでは、IDT 社ガスセンサモジュール (ZMOD4410) を RL78/G11 で制御し、屋内空気質を測定します。IDT 社が提供しているライブラリを使用して ZMOD4410 の測定値から、IAQ (Indoor-Air-Quality) 指標を計算し、LCD キャラクタディスプレイに結果を表示します。空気品質に応じて警告灯 (LED) を点灯します。

表 1.1 使用する周辺機能とその用途

| 周辺機能  | 用途                  |
|-------|---------------------|
| IICA0 | ZMOD4410 との通信       |
| IICA1 | LCD キャラクタディスプレイとの通信 |

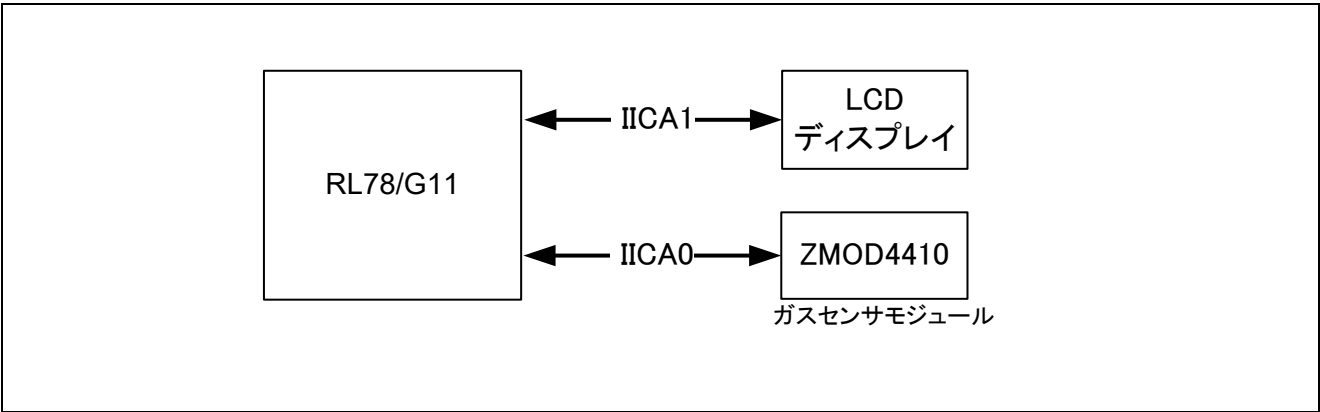


図 1.1 屋内空気品質測定システムの全体概要

## 1.1 室内空気質 (IAQ)

室内空気質 (Indoor Air Quality, IAQ) は、空気の清潔さ、人体にとっての快適性を表します。IDT 社屋内空気質センサ (ZMOD4410) では、空気中の全揮発性有機化合物 (TVOC) を測定し、IDT 社の評価指標に当てはめて屋内空気質を示します。IDT 社の評価指標については、ZMOD4410 マニュアルを参照してください。

## 1.2 ZMOD4410 用 I2C 通信プロトコル

ZMOD4410 との通信に I2C 通信を用います。ZMOD4410 の内部 RAM にアクセスして各種設定、計測データの取得を行います。

ZMOD4410 は、スタンダードモード及びファストモードに対応しており、スレーブアドレスは 32H (デフォルト値) です。データプロトコル構造については、ZMOD4410 のマニュアルを参照してください。

## 1.3 ZMOD4410 用ライブラリ

ZMOD4410 用ライブラリおよびライブラリの詳細は、IDT 社ホームページから入手してください。

<https://ja.idt.com/products/sensor-products/gas-sensors/zmod4410-indoor-air-quality-sensor-platform>  
(2019 年 5 月 14 日提供)

本アプリケーションノートでは、ライブラリ拡張子を .lib から .a にリネームしています。また、コード圧縮のため、ライブラリ付属のサンプルプログラムから printf 関数をコメントアウトし、プロジェクト全体に対して GCC コンパイラの最適化オプション (-O1) を使用しています。なお、ZMOD4410 の動作モードとして Continuous-mode を使用します (Low-power-mode は使用しません)。

## 1.4 LCD キャラクタディスプレイ

LCD キャラクタディスプレイには、HD44780 互換品 (16×2 表示、I2C 通信) を使用します。算出した IAQ 値を図 1.3 のフォーマットで表示します。

|   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|--|
| M | E | A | S | U | R | E | M | E | N | T | : |  |  |  |  |  |
| I | A | Q | : |   | 3 | . | 4 | 5 |   |   |   |  |  |  |  |  |

図 1.2 LCD キャラクタディスプレイの表示パターン

1.5 警告灯

ZMOD4410 の測定値から算出した IAQ rating 値に応じて警告灯 (LED) を下表の通りに点灯させます。

表 1.2 警告灯の点灯パターン

| IAQ rating  | TVOC (mg/m3) | LED1 (緑) | LED2 (黄) | LED3 (赤) |
|-------------|--------------|----------|----------|----------|
| ≤ 2.99      | < 1.0        | ON       | OFF      | OFF      |
| 3.00 ~ 4.99 | 1.0 ~ 10.0   | OFF      | ON       | OFF      |
| ≥ 5.00      | > 10.0       | OFF      | OFF      | ON       |

## 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おります。

表 2.1 動作確認条件

| 項目                 | 内容                                                                                                                        |
|--------------------|---------------------------------------------------------------------------------------------------------------------------|
| 使用マイコン             | RL78/G11 (R5F1056A)                                                                                                       |
| 動作周波数              | 高速オンチップ・オシレータ・クロック (f <sub>IH</sub> ) : 48MHz<br>CPU/周辺ハードウェア・クロック : 24MHz                                                |
| 動作電圧               | 3.3V<br>LVD 動作 (V <sub>LVD</sub> ) : リセット・モード TYP. 2.75V<br>電源立ち上がり時 2.76V~2.87V<br>電源立ち下がり時 2.70V~2.81V                  |
| 統合開発環境 (e2studio)  | ルネサスエレクトロニクス製<br>e2studio V7.4.0                                                                                          |
| C コンパイラ (e2studio) | GCC for Renesas RL78 4.9.2.201801<br>最適化オプション: -O1                                                                        |
| ZMOD4410ライブラリ、API  | IDT_ZMOD4410-Air-Quality-eCO2-Firmware_SWR_20190514<br>IDT_ZMOD4410-Odor-Firmware_SWR_20190514<br>CONTINUOUS_MODEでのみ動作を確認 |

表 2.2 コード生成ツール設定値

| 項目    | 内容                                                                                                                     |
|-------|------------------------------------------------------------------------------------------------------------------------|
| IICA0 | 転送モード: シングルマスタ<br>カウントクロック設定: fCLK/2<br>自局アドレス設定: 16<br>動作モード設定: 標準, 50000 bps<br>通信完了割り込み優先順位: レベル3<br>コールバック機能設定: なし |
| IICA1 | 転送モード: シングルマスタ<br>カウントクロック設定: fCLK/2<br>自局アドレス設定: 16<br>動作モード設定: 標準, 50000 bps<br>通信完了割り込み優先順位: レベル3<br>コールバック機能設定: なし |

### 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ノートで使用するハードウェア構成例を示します。

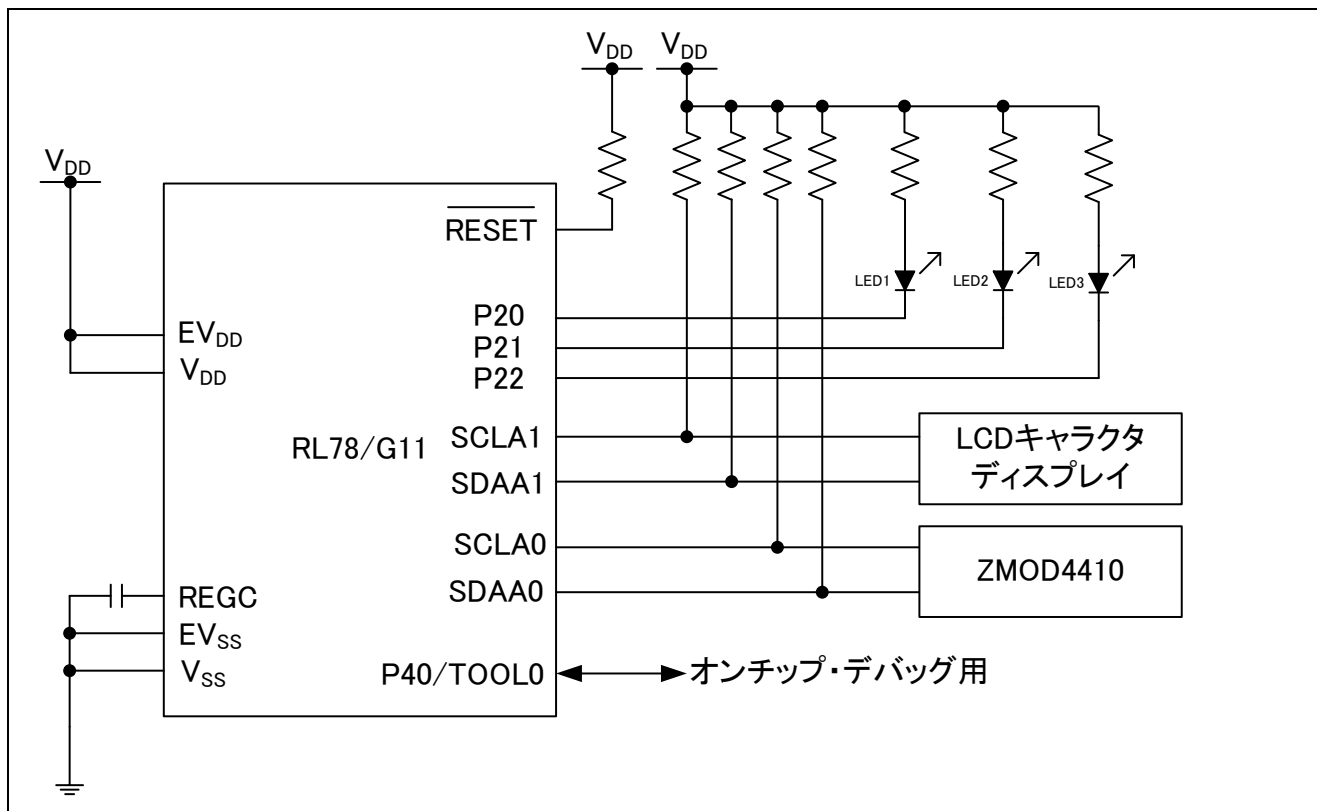


図 3.1 ハードウェア構成

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)。

注意 2. VDD は LVD にて設定したリセット解除電圧 ( $V_{LVD}$ ) 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                         |
|-------|-----|----------------------------|
| SCLA0 | 出力  | I2Cシリアル・クロック (ZMOD4410)    |
| SDAA0 | 出力  | I2Cシリアル・データ・バス (ZMOD4410)  |
| SCLA1 | 出力  | I2Cシリアル・クロック (LCDディスプレイ)   |
| SDAA1 | 出力  | I2Cシリアル・データ・バス (LCDディスプレイ) |
| P20   | 出力  | LED1 (緑)                   |
| P21   | 出力  | LED2 (黄)                   |
| P22   | 出力  | LED3 (赤)                   |

## 4. ソフトウェア説明

### 4.1 フォルダ構成

本アプリケーションノートのフォルダ構成を表 4.1 に示します。

表 4.1 ディレクトリ構成

| ファイル名                             | 説明                                 | 同梱<br>ファイル | コード生<br>成ツール<br>を使用 |
|-----------------------------------|------------------------------------|------------|---------------------|
| ¥an-r01an4903xx0100-rl78g11       | <DIR> 本アプリケーションのフォルダ               |            |                     |
| r01an4903xx0100-rl78g11.pdf       | 本アプリケーションノート                       | ✓          |                     |
| ¥src                              | <DIR> プログラム格納用フォルダ                 |            |                     |
| ¥api                              | <DIR> ZMOD4410 ライブラリ制御用 API 格納フォルダ |            |                     |
| continuous.c                      | ZMOD4410 ライブラリ API                 |            |                     |
| continuous.h                      | ZMOD4410 ライブラリ API ヘッドファイル         |            |                     |
| zmod44xx_config.h                 | ZMOD4410 ライブラリ API ヘッドファイル         |            |                     |
| zmod44xx_types.h                  | ZMOD4410 ライブラリ API ヘッドファイル         |            |                     |
| zmod44xx.c                        | ZMOD4410 ライブラリ API                 |            |                     |
| zmod44xx.h                        | ZMOD4410 ライブラリ API ヘッドファイル         |            |                     |
| ¥lib                              | <DIR> ZMOD4410 ライブラリ格納フォルダ         |            |                     |
| eco2.h                            | ZMOD4410 ライブラリ ヘッドファイル             |            |                     |
| iaq.h                             | ZMOD4410 ライブラリ ヘッドファイル             |            |                     |
| odor.h                            | ZMOD4410 ライブラリ ヘッドファイル             |            |                     |
| r_cda.h                           | ZMOD4410 ライブラリ ヘッドファイル             |            |                     |
| tvoc.h                            | ZMOD4410 ライブラリ ヘッドファイル             |            |                     |
| zmod4410_calibration.h            | ZMOD4410 ライブラリ ヘッドファイル             |            |                     |
| libeco2_rl78.a                    | ZMOD4410 ライブラリ                     |            |                     |
| libiaq_rl78.a                     | ZMOD4410 ライブラリ                     |            |                     |
| libodor_rl78.a                    | ZMOD4410 ライブラリ                     |            |                     |
| libr_cda_rl78.a                   | ZMOD4410 ライブラリ                     |            |                     |
| libtvoc_rl78.a                    | ZMOD4410 ライブラリ                     |            |                     |
| libzmod4410_calibration_co_rl78.a | ZMOD4410 ライブラリ                     |            |                     |
| ¥src                              | <DIR> ソースファイル格納用フォルダ               |            |                     |
| r_cg_cgc_user.c                   | CGC モジュール                          | ✓          | ✓                   |
| r_cg_cgc.c                        | CGC モジュール                          | ✓          | ✓                   |
| r_cg_cgc.h                        | CGC モジュール ヘッドファイル                  | ✓          | ✓                   |
| r_cg_hardware_setup.c             | システム初期化モジュール                       | ✓          | ✓                   |
| r_cg_iica_user.c                  | IICA モジュール                         | ✓          | ✓                   |
| r_cg_iica.c                       | IICA モジュール                         | ✓          | ✓ (—)               |
| r_cg_iica.h                       | IICA モジュール ヘッドファイル                 | ✓          | ✓                   |
| r_cg_interrupt_handlers.h         | 割り込みハンドラ ヘッドファイル                   | ✓          | ✓                   |
| r_cg_macrodriver.h                | ヘッドファイル                            | ✓          | ✓                   |
| r_cg_main.c                       | メイン処理モジュール                         | ✓          | ✓                   |
| r_cg_port_user.c                  | Port モジュール                         | ✓          | ✓                   |
| r_cg_port.c                       | Port モジュール                         | ✓          | ✓                   |
| r_cg_port.h                       | Port モジュール ヘッドファイル                 | ✓          | ✓                   |
| r_cg_reset_program.asm            | リセットプログラム                          | ✓          | ✓                   |
| r_cg_userdefine.h                 | ユーザ定義ファイル                          | ✓          | ✓                   |
| r_cg_vector_table.c               | ベクタテーブル                            | ✓          | ✓                   |
| r_lcd.c                           | LCD キャラクタディスプレイ制御モジュール             | ✓          |                     |
| r_lcd.h                           | LCD キャラクタディスプレイ制御モジュール ヘッド         | ✓          |                     |
| r_rl78_i2c.c                      | IICA インタフェース                       | ✓          |                     |
| r_rl78_i2c.h                      | IICA インタフェース ヘッドファイル               | ✓          |                     |

補足 "<DIR>" は、ディレクトリを意味します。

## 4.2 continuous.c の変更点

本アプリケーションノートでは、IDT 社サンプルプログラムに対し、下記の通り、ソースコードを変更しています。

- printf 関数のコメントアウト

変更箇所は、continuous.c の全範囲。

- iaq 値を LCD ディスプレイ、LED に出力する関数 "R\_DRAW\_IAQ()" をコール (r\_cg\_main.c 内で定義)

変更箇所は下記の通り。

|            | 変更前                                                                                   | 変更後                                                                                                       |
|------------|---------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------|
| cont_run() | <pre>/* calculate IAQ index */<br/>iaq = calc_iaq(r_mox, r_cda, &amp;tvoc_par);</pre> | <pre>/* calculate IAQ index */<br/>iaq = calc_iaq(r_mox, r_cda, &amp;tvoc_par);<br/>R_DrawIAQ(iaq);</pre> |

### 4.3 r\_cg\_iica.c の変更点

本アプリケーションノートでは、コード生成ツールで出力されたコードに対して、下記の通り、色づけした箇所をコメントアウトしています。再度、コード生成ツールでプログラムを自動生成したときは、下記の通りに色づけした箇所をコメントアウトしてください。

|                        | 変更前                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | 変更後                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| R_IICA0_Master_Receive | <pre>MD_STATUS R_IICA0_Master_Receive(uint8_t adr, uint8_t * * const rx_buf, uint16_t rx_num, uint8_t wait) {     MD_STATUS status = MD_OK;      IICAMK0 = 1U; /* disable INTIICA0 interrupt */      if (1U == IICBSY0)     {         /* Check bus busy */         IICAMK0 = 0U; /* enable INTIICA0 interrupt */         status = MD_ERROR1;     }     else     {         STT0 = 1U; /* generate a start condition */         IICAMK0 = 0U; /* enable INTIICA0 interrupt */     } }</pre> | <pre>MD_STATUS R_IICA0_Master_Receive(uint8_t adr, uint8_t * * const rx_buf, uint16_t rx_num, uint8_t wait) {     MD_STATUS status = MD_OK;      IICAMK0 = 1U; /* disable INTIICA0 interrupt */      /* IICBSY0 = 1U */     /* IICBSY0 */     /* Check bus busy */     /* IICAMK0 = 0U; /* enable INTIICA0 interrupt */     /* IICBSY0 = MD_ERROR1 */     /* IICBSY0 */     {         STT0 = 1U; /* generate a start condition */         IICAMK0 = 0U; /* enable INTIICA0 interrupt */     } }</pre> |
| R_IICA0_Master_Send    | <pre>MD_STATUS R_IICA0_Master_Send(uint8_t adr, uint8_t * * const tx_buf, uint16_t tx_num, uint8_t wait) {     MD_STATUS status = MD_OK;      IICAMK0 = 1U; /* disable INTIICA0 interrupt */      if (1U == IICBSY0)     {         /* Check bus busy */         IICAMK0 = 0U; /* enable INTIICA0 interrupt */         status = MD_ERROR1;     }     else     {         STT0 = 1U; /* generate a start condition */         IICAMK0 = 0U; /* enable INTIICA0 interrupt */     } }</pre>    | <pre>MD_STATUS R_IICA0_Master_Send(uint8_t adr, uint8_t * * const tx_buf, uint16_t tx_num, uint8_t wait) {     MD_STATUS status = MD_OK;      IICAMK0 = 1U; /* disable INTIICA0 interrupt */      /* IICBSY0 = 1U */     /* IICBSY0 */     /* Check bus busy */     /* IICAMK0 = 0U; /* enable INTIICA0 interrupt */     /* IICBSY0 = MD_ERROR1 */     /* IICBSY0 */     {         STT0 = 1U; /* generate a start condition */         IICAMK0 = 0U; /* enable INTIICA0 interrupt */     } }</pre>    |

#### 4.4 動作概要

本アプリケーションノートでは、初期設定後、ZMOD4410 を利用して IAQ 指標の測定を開始します。測定した結果を LCD ディスプレイに表示し、IAQ に応じた LED を点灯させます。

- ① IICA、LCD キャラクタディスプレイの初期設定を行います。
- ② ZMOD4410 からデバイス情報を読み出します。
- ③ ZMOD4410 のキャリブレーション用パラメータを計算します。
- ④ ZMOD4410 の初期設定、計測用設定を行います。
- ⑤ ZMOD4410 の計測を開始します。
- ⑥ ZMOD4410 の初期化完了を待ちます。
- ⑦ ZMOD4410 での計測完了を待ちます。
- ⑧ ZMOD4410 から計測結果を読み出します。
- ⑨ 初期化完了直後から 10 回目までの測定結果は棄却します（測定結果の較正処理）。
- ⑩ 計測結果から IAQ 値を計算します。
- ⑪ IAQ 値を LCD ディスプレイへ表示し、IAQ 値に応じた LED を点灯します。
- ⑫ ZMOD4410 が待機状態になるまで待ちます。
- ⑬ ⑦～⑫を繰り返します。

## 4.5 オプション・バイトの設定一覧

表 4.2 にオプション・バイトの設定一覧を示します。

表 4.2 オプション・バイト設定一覧

| アドレス          | 設定値       | 内容                                                                                  |
|---------------|-----------|-------------------------------------------------------------------------------------|
| 000C0H/010C0H | 11101111B | ウォッチドッグ・タイマ動作禁止<br>(リセット解除後、カウント停止)                                                 |
| 000C1H/010C1H | 01111111B | LVD 動作 (VLVD) : リセット・モード TYP. 2.75V<br>電源立ち上がり時 2.76V~2.87V<br>電源立ち下がり時 2.70V~2.81V |
| 000C2H/010C2H | 11101000B | HS モード<br>高速オンチップ・オシレータ・クロック 24MHz                                                  |
| 000C3H/010C3H | 10000100B | オンチップ・デバッグ許可                                                                        |

## 4.6 定数一覧

表 4.3 にサンプルプログラムで使用する定数を示します。

表 4.3 定数一覧

| 定数名             | 設定値            | 内容                         | 定義ファイル             |
|-----------------|----------------|----------------------------|--------------------|
| LCD_SLAVE_ADDR  | A0H            | LCDキャラクタディスプレイの<br>I2Cアドレス | r_ccg_userdefine.h |
| FCLK_MHZ        | 24             | CPU周波数 [MHz]               | lcd.h              |
| LOOPNUM_1MS     | FCLK_MHZ * 143 | 1msウェイト関数用ループ数             | lcd.h              |
| CONTINUOUS_MODE | —              | —                          | プロジェクト設定           |

## 4.7 関数一覧

表 4.4 にサンプルプログラムで使用する関数を示します。

表 4.4 関数

| 関数名                 | 概要                      |
|---------------------|-------------------------|
| main                | メイン関数 (コード生成ツールにより生成)   |
| R_MAIN_UserInit     | ユーザ初期化関数                |
| R_RL78_I2C_Read     | ZMOD4410のレジスタからデータ読み出し  |
| R_RL78_I2C_Write    | ZMOD4410のレジスタへのデータ書き込み  |
| R_LCD_Init          | LCDキャラクタディスプレイの初期化関数    |
| R_LCD_WwriteCommand | LCDキャラクタディスプレイにコマンド送信   |
| R_LCD_WriteData     | LCDキャラクタディスプレイに描画データを送信 |
| R_WaitMilliseconds  | 引数で指定された時間 (ms) のウェイト   |
| R_Wait1MilliSecond  | 1msのウェイト                |

## 4.8 関数仕様

サンプルコードの関数仕様とフローチャートを示します。

| [関数名] main |                                                  |
|------------|--------------------------------------------------|
| 概 要        | メイン関数                                            |
| ヘッダ        | —                                                |
| 宣 言        | —                                                |
| 説 明        | メイン・ユーザ初期設定関数を実行後、ZMOD4410 を制御して IAQ 値の計測を開始します。 |
| 引 数        | なし                                               |
| リターン値      | なし                                               |
| 備考         | なし                                               |

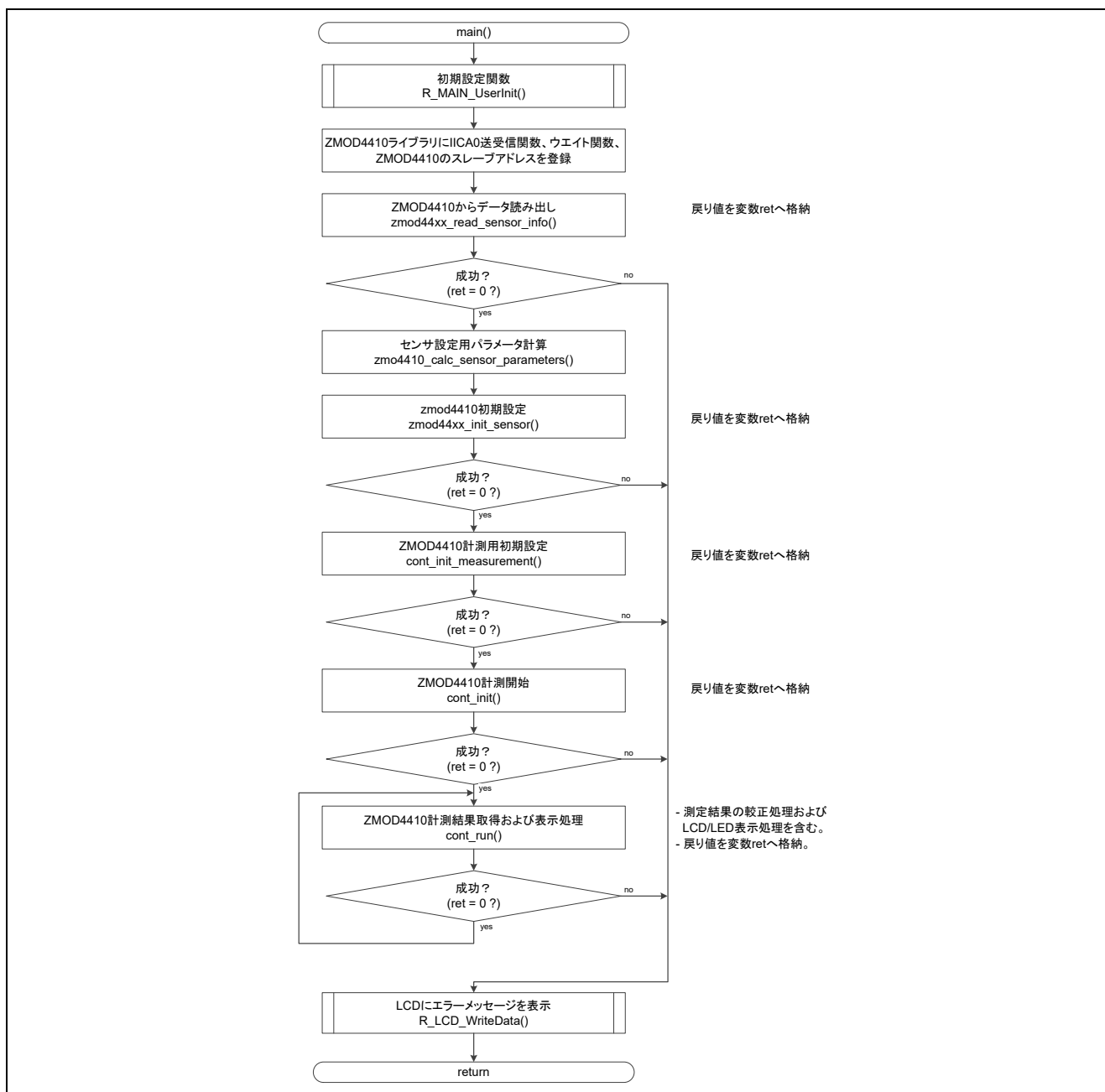


図 5.1 メイン関数

|                       |                                      |
|-----------------------|--------------------------------------|
| [関数名] R_MAIN_UserInit |                                      |
| 概 要                   | ユーザ初期設定関数                            |
| ヘッダ                   | —                                    |
| 宣 言                   | static void R_MAIN_UserInit(void);   |
| 説 明                   | EI 命令で割り込み許可にした後、LCD モジュールの初期化を行います。 |
| 引 数                   | なし                                   |
| リターン値                 | なし                                   |
| 備考                    | なし                                   |

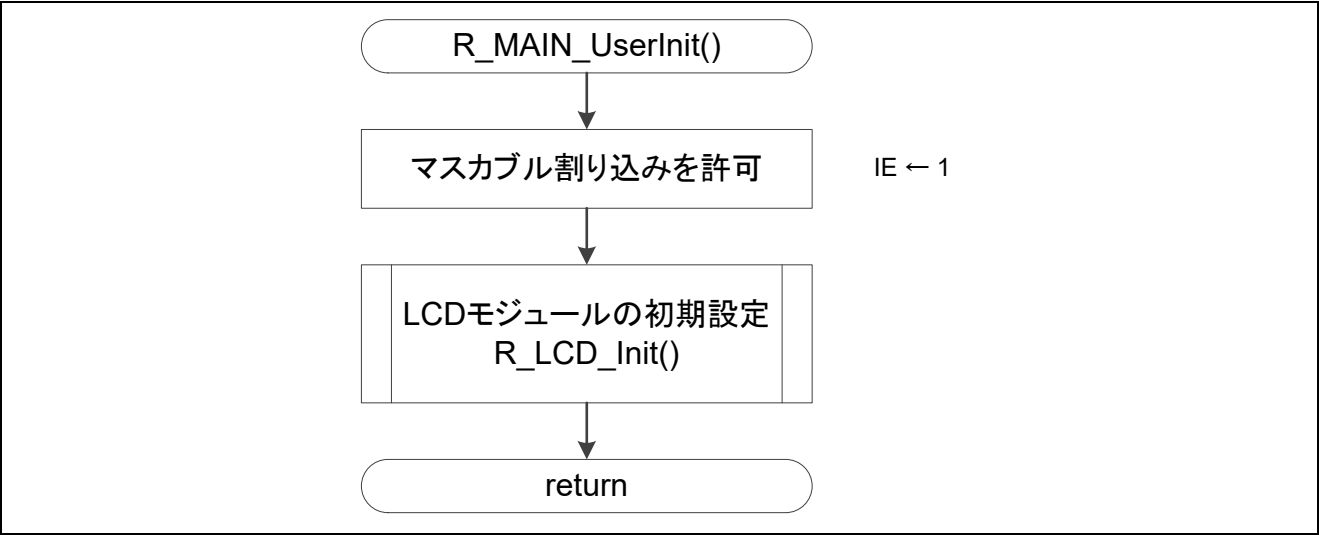


図 5.2 ユーザ初期設定関数

## [関数名] R\_RL78\_I2C\_Read

|       |                                                                                                                                                                                              |
|-------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 概 要   | ZMOD4410 のレジスタからデータ読み出し                                                                                                                                                                      |
| ヘッダ   | —                                                                                                                                                                                            |
| 宣 言   | int8_t R_RL78_I2C_Read ( uint8_t i2c_addr, uint8_t reg_addr, const uint8_t *buf, uint8_t len);                                                                                               |
| 説 明   | ZMOD4410 のレジスタからデータを読み出します。本関数を、i2c データ読み出し関数として ZMOD4410 のライブラリに登録してください。                                                                                                                   |
| 引 数   | i2c_addr                    ZMOD4410 のスレーブアドレス<br>reg_addr                    読み出し元レジスタアドレス<br>buf                         読み出しデータ格納用バッファアドレス<br>len                         読み出しデータ数 [Byte] |
| リターン値 | 0: 成功, else: 失敗                                                                                                                                                                              |
| 備考    | R_IICA_0_Master_Send()と、R_IICA_0_Master_Receive()は、実行成功時に MD_OK を返します。(r_cg_macrodriver.h で定義)                                                                                               |

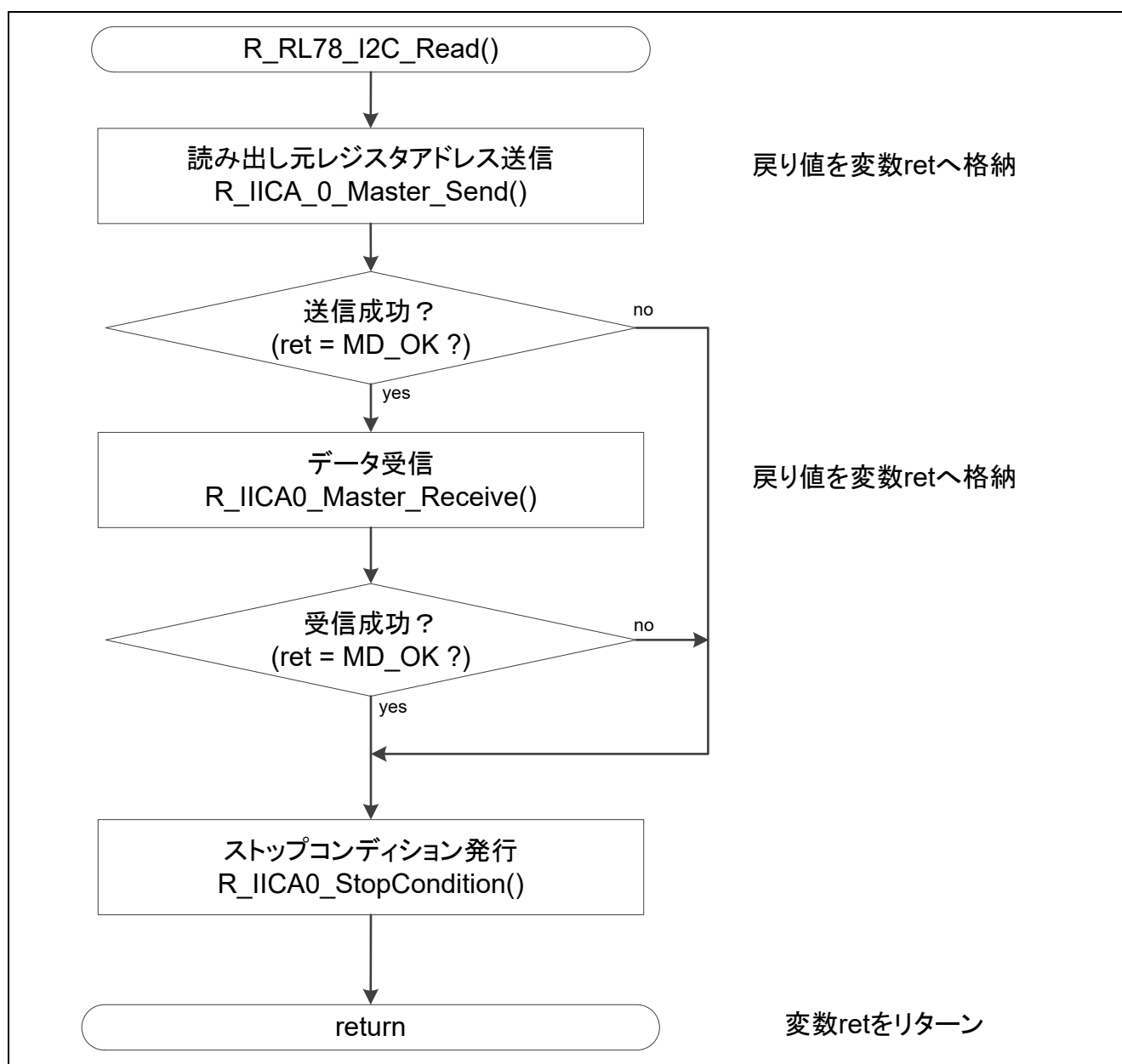


図 5.3 ZMOD4410 のレジスタからのデータ読み出し関数

## [関数名] R\_RL78\_I2C\_Write

|       |                                                                                                                                                                                              |
|-------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 概 要   | ZMOD4410 のレジスタへのデータ書き込み                                                                                                                                                                      |
| ヘッダ   | —                                                                                                                                                                                            |
| 宣 言   | int8_t R_RL78_I2C_Write(uint8_t i2c_addr, uint8_t reg_addr, const uint8_t *buf, uint8_t len);                                                                                                |
| 説 明   | ZMOD4410 のレジスタへデータを書き込みます。本関数を、i2c データ書き込み関数として ZMOD4410 のライブラリに登録してください。                                                                                                                    |
| 引 数   | i2c_addr                    ZMOD4410 のスレーブアドレス<br>reg_addr                    書き込み先レジスタアドレス<br>buf                         書き込みデータ格納用バッファアドレス<br>len                         書き込みデータ数 [Byte] |
| リターン値 | 0: 成功, else: 失敗                                                                                                                                                                              |
| 備考    | R_IICA_0_Master_Send()と、R_IICA_0_Master_Receive()は、実行成功時に MD_OK を返します。(r_cg_macrodriver.h で定義)                                                                                               |

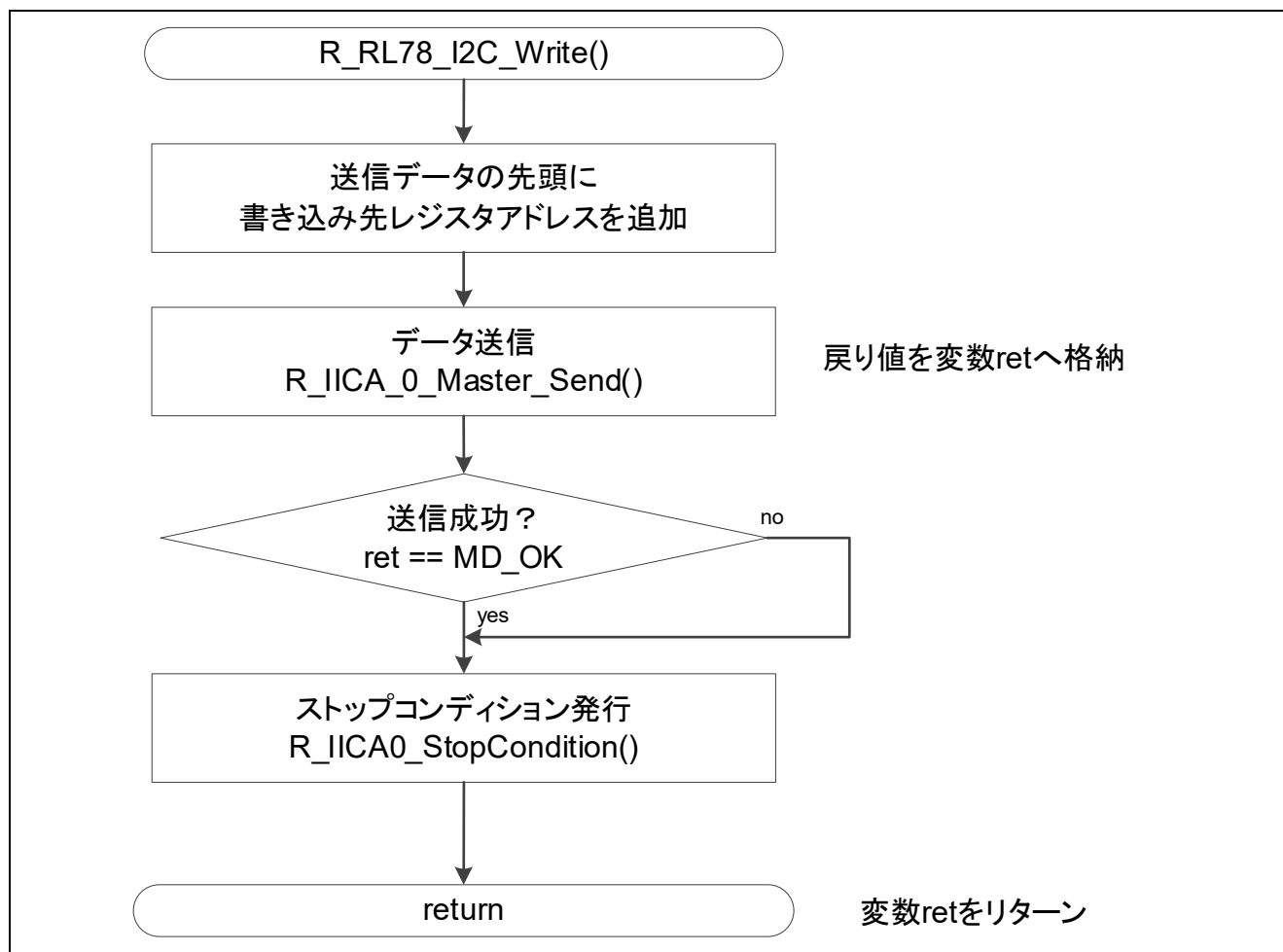


図 5.4 ZMOD4410 のレジスタへのデータ書き込み関数

## [関数名] R\_LCD\_Init

|       |                        |
|-------|------------------------|
| 概 要   | LCD キャラクタディスプレイの初期化関数  |
| ヘッダ   | lcd.h                  |
| 宣 言   | void R_LCD_Init(void); |
| 説 明   | LCD キャラクタの初期設定を行います。   |
| 引 数   | なし                     |
| リターン値 | なし                     |
| 備考    | なし                     |

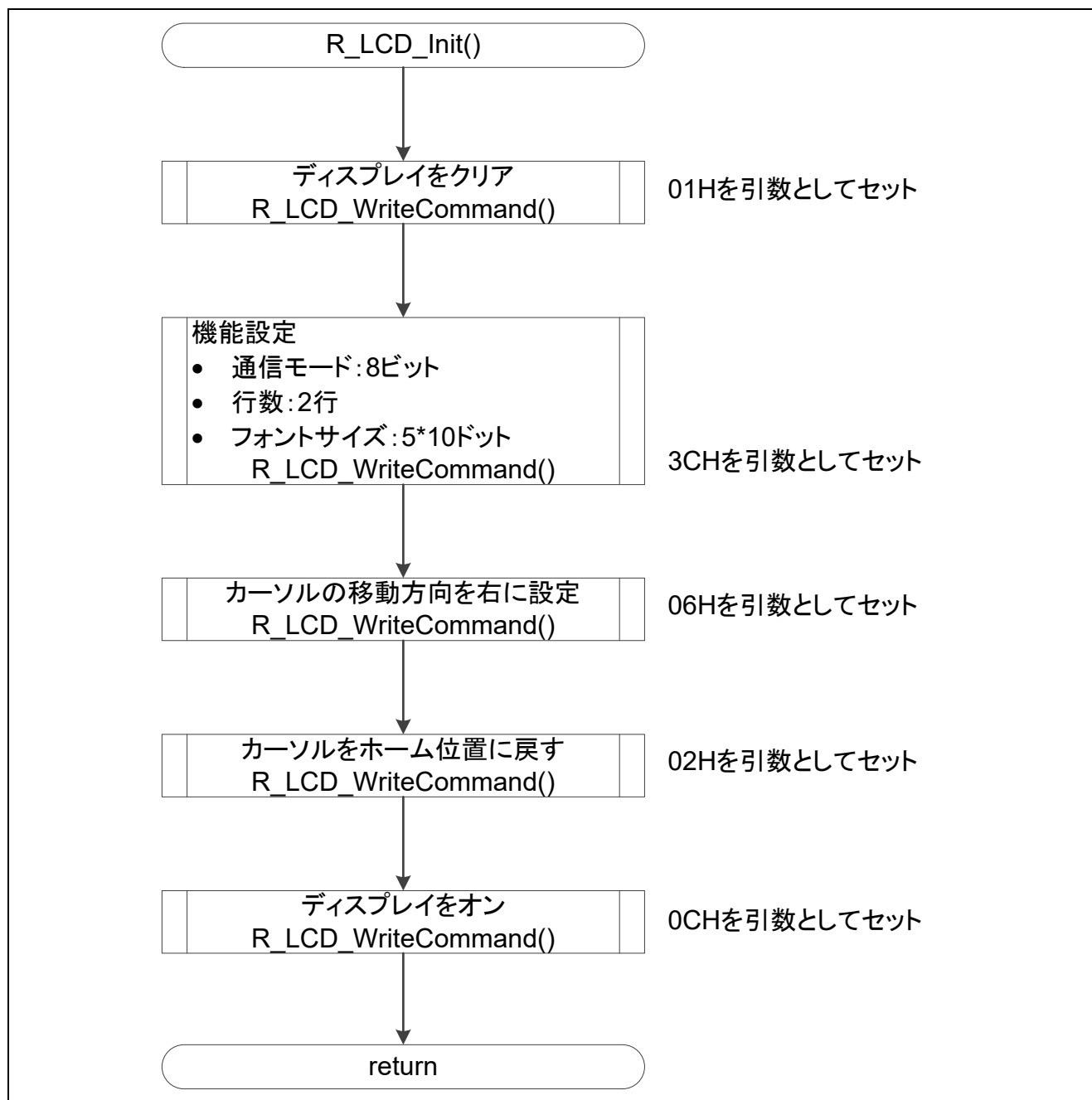


図 5.5 LCD キャラクタディスプレイ初期設定関数

---

[関数名] R\_LCD\_WriteCommand

---

|       |                                          |        |
|-------|------------------------------------------|--------|
| 概 要   | LCD キャラクタディスプレイへの制御コマンド送信関数              |        |
| ヘッダ   | lcd.h                                    |        |
| 宣 言   | void R_LCD_WriteCommand(uint8_t command) |        |
| 説 明   | LCD キャラクタディスプレイに引数で指定された制御コマンドを送信します。    |        |
| 引 数   | command                                  | 制御コマンド |
| リターン値 | なし                                       |        |
| 備考    | なし                                       |        |

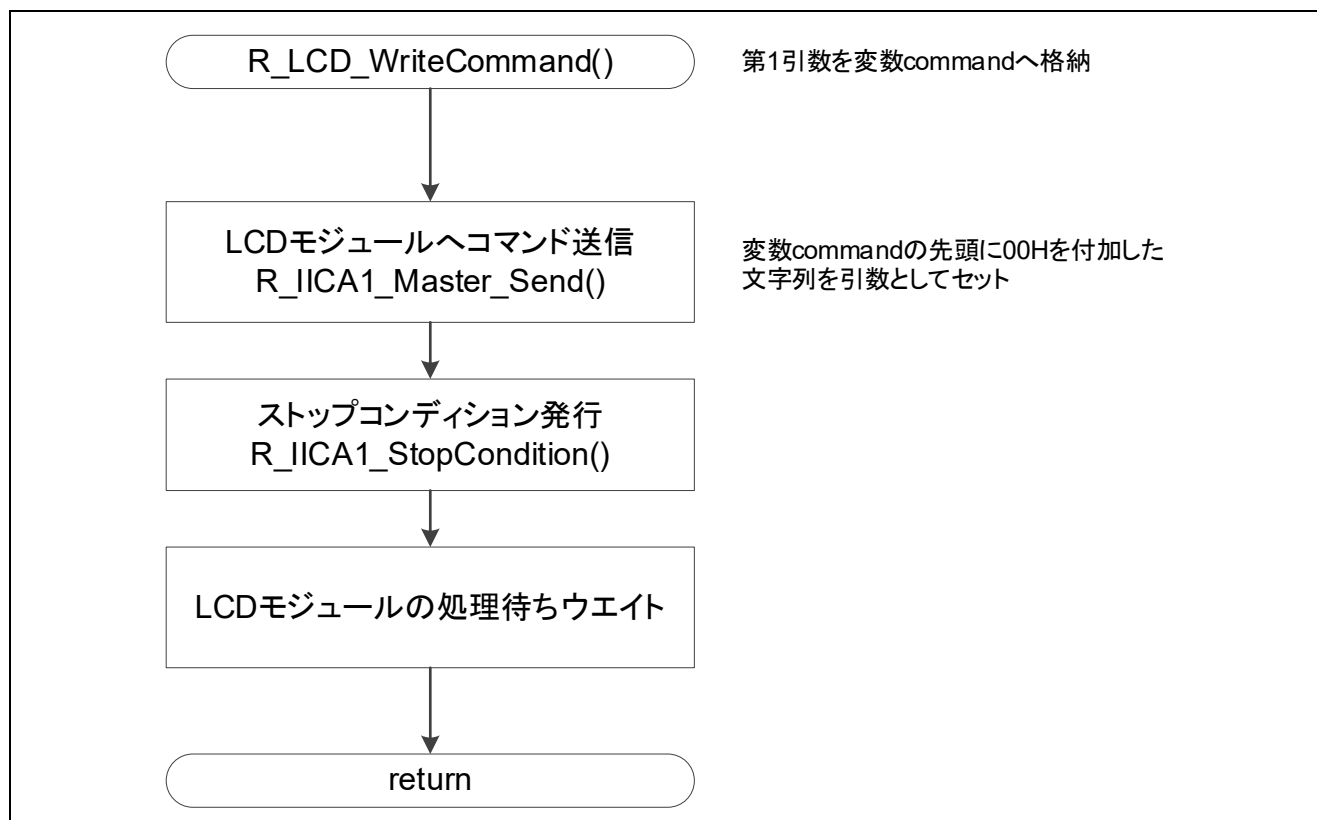


図 5.6 LCD キャラクタディスプレイコマンド送信関数

## [関数名] R\_LCD\_WriteData

|       |                                                       |                        |
|-------|-------------------------------------------------------|------------------------|
| 概 要   | LCD キャラクタディスプレイへのデータ送信関数                              |                        |
| ヘッダ   | lcd.h                                                 |                        |
| 宣 言   | void R_LCD_WriteData(uint8_t* data, uint8_t data_num) |                        |
| 説 明   | LCD キャラクタディスプレイに引数で指定されたデータを送信します。                    |                        |
| 引 数   | data                                                  | 送信データが格納されている領域の先頭アドレス |
|       | data_num                                              | 送信データのバイト数             |
| リターン値 | なし                                                    |                        |
| 備考    | なし                                                    |                        |

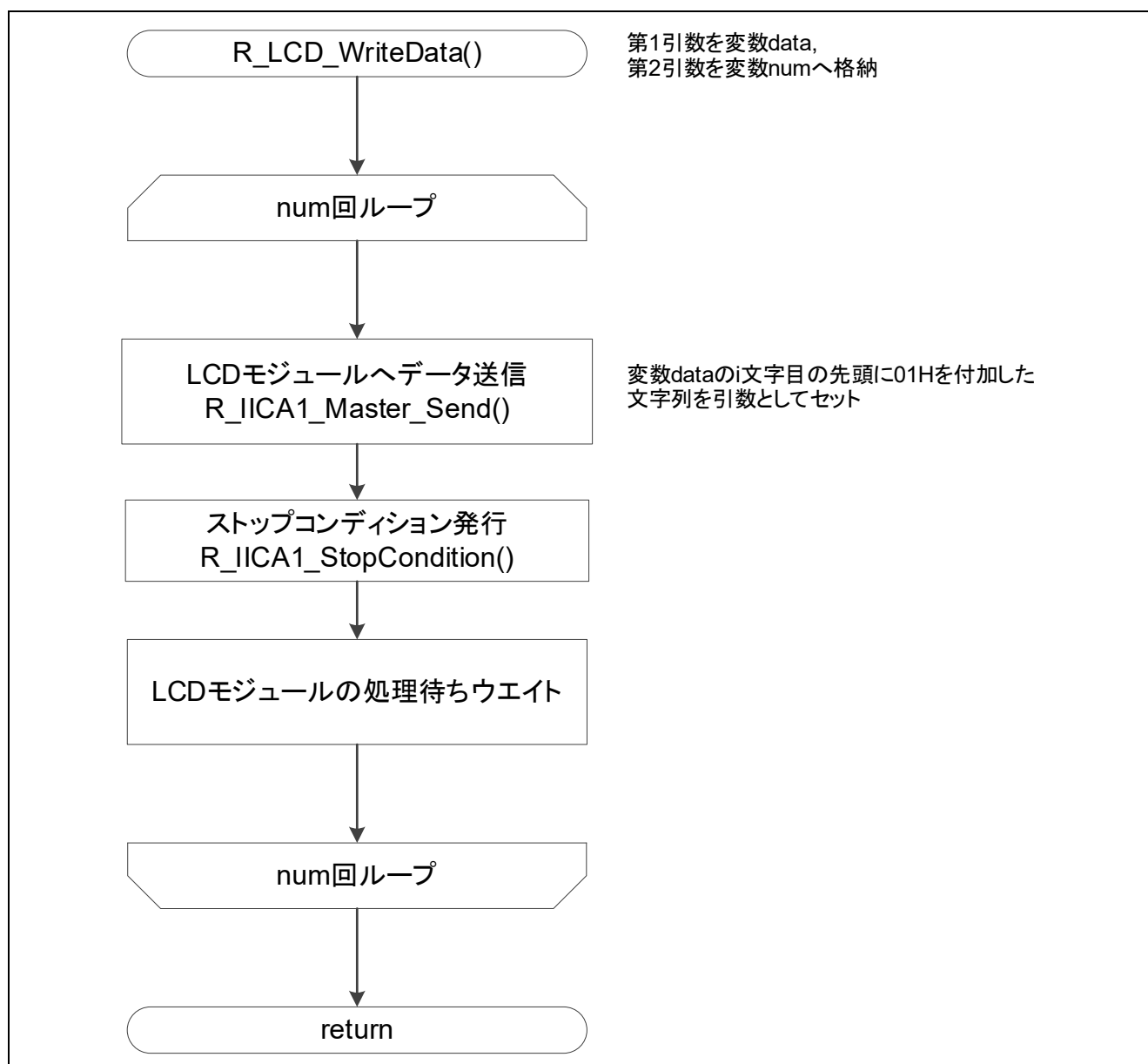


図 5.7 LCD キャラクタディスプレイ表示データ送信関数

|                          |                                                                                                                                            |
|--------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| [関数名] R_WaitMilliseconds |                                                                                                                                            |
| 概 要                      | ミリ秒ウエイト                                                                                                                                    |
| ヘッダ                      | r_lcd.h                                                                                                                                    |
| 宣 言                      | void R_WaitMilliseconds(uint16_t msec);                                                                                                    |
| 説 明                      | <ul style="list-style-type: none"><li>引数で指定された秒数分、R_Wait1MilliSecond 関数を呼び出します。</li><li>本関数をウエイト用関数として ZMOD4410 のライブラリに登録してください。</li></ul> |
| 引 数                      | uint16_t msec                      ウエイトミリ秒数                                                                                                |
| リターン値                    | なし                                                                                                                                         |
| 備考                       | なし                                                                                                                                         |

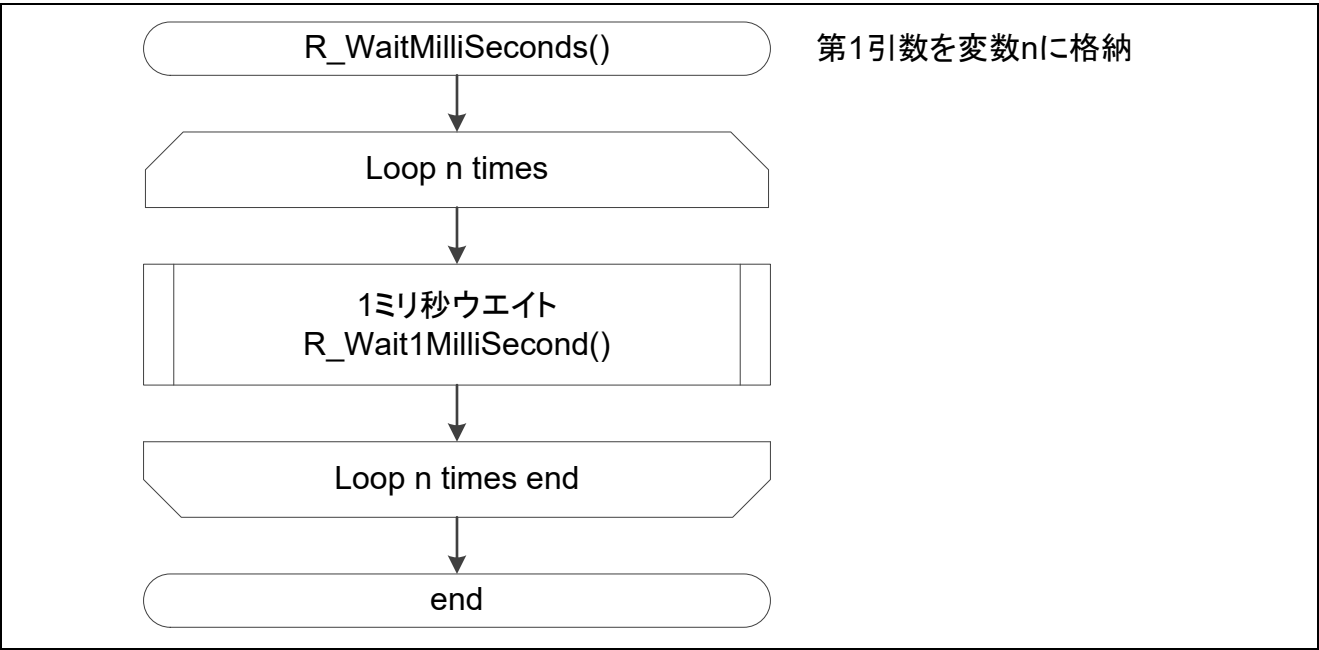


図 4.8 ミリ秒ウエイト

---

[関数名] R\_Wait1MilliSecond

---

|       |                                |
|-------|--------------------------------|
| 概 要   | 1 ミリ秒ウエイト                      |
| ヘッダ   | r_lcd.h                        |
| 宣 言   | void R_Wait1MilliSecond(void); |
| 説 明   | • CPU によって 1 ミリ秒ウエイトします。       |
| 引 数   | なし                             |
| リターン値 | なし                             |
| 備考    | なし                             |

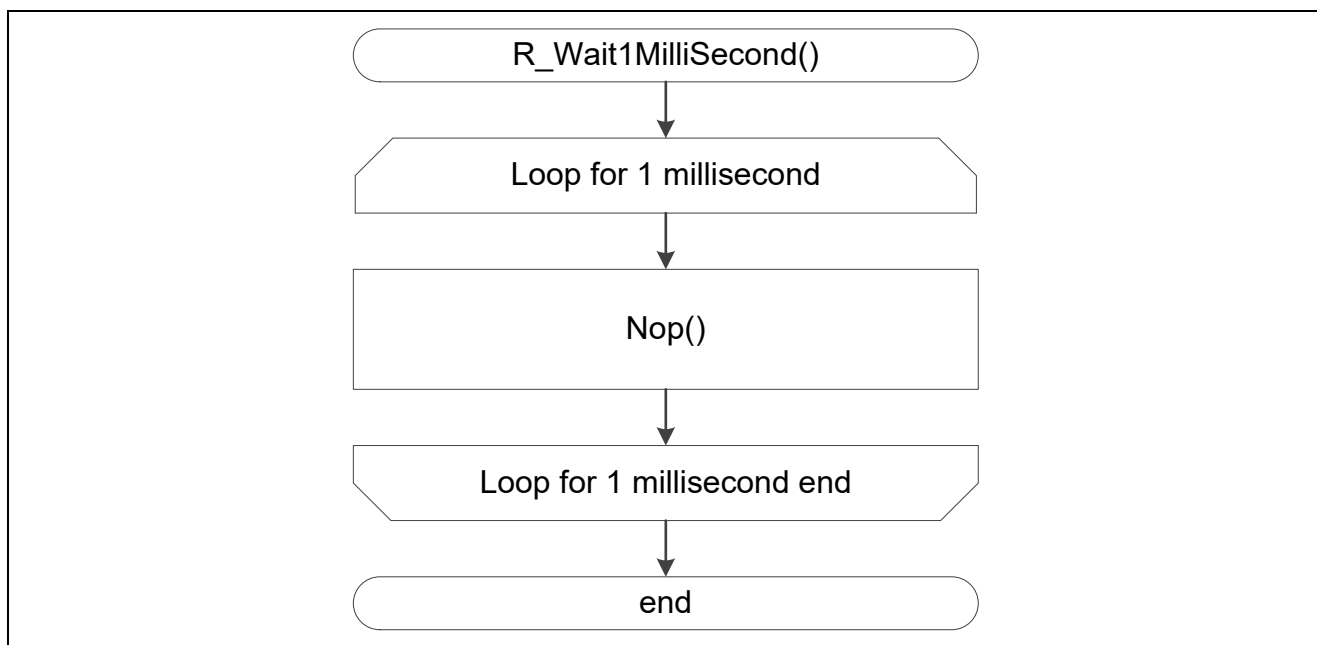


図 4.9 1 ミリ秒ウエイト

## 5. 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。

ZMOD4410 制御ライブラリは、下記 IDT 社ホームページから入手してください。

<https://ja.idt.com/products/sensor-products/gas-sensors/zmod4410-indoor-air-quality-sensor-platform>

### 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/>

お問い合わせ先

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

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 | 2019.10.03 | —    | 初版発行 |
|      |            |      |      |

## 製品ご使用上の注意事項

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

### 1. 静電気対策

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

### 2. 電源投入時の処置

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

### 3. 電源オフ時における入力信号

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

### 4. 未使用端子の処理

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

### 5. クロックについて

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

### 6. 入力端子の印加波形

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

### 7. リザーブアドレス（予約領域）のアクセス禁止

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

### 8. 製品間の相違について

型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。同じグループのマイコンでも型名が違えば、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。お客様の機器・システムの設計において、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には、お客様の責任において行ってください。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みます。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、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、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。
5. 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。
6. 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。
7. 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。
8. 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、当社は、一切その責任を負いません。
9. 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。
10. 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。
11. 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。
12. 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.4.0-1 2017.11)

## 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](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。