

RL78/G15

PWM を使用したヒータ温度制御

要旨

本アプリケーションノートでは、マグカップ型電気なべ（クッキングマグカップ）を想定し、PWM 機能を用いた小型ヒータの温度制御方法について説明します。

PWM 信号を MOSFET に入力し、その ON/OFF 動作でヒータの発熱量を制御します。サーミスタ入力により検出した実際の温度と設定温度の差分にもとづき、ヒータのフィードバック制御を行います。それにより、設定温度までの加熱、規定温度を設定時間までの維持が可能になります。

動作確認デバイス

| | |
|-------|---------------------------------------|
| 評価ボード | RL78/G15-20pin Fast Prototyping Board |
| ヒータ | AL-DC-A6-30 アルミ箔フレキシブルヒータ |
| サーミスタ | 103AT-2 AT Thermistor |

本アプリケーションノートを他のマイコンやヒータ、サーミスタへ適用する場合、それらの仕様に合わせて変更し、十分評価してください。またヒータは高温になりますので、やけどや火災にご注意ください。

目次

| | |
|--|----|
| 1. 仕様 | 4 |
| 1.1 仕様概要 | 4 |
| 1.2 動作詳細 | 4 |
| 2. 動作確認条件 | 10 |
| 3. ハードウェア説明 | 11 |
| 3.1 ハードウェア構成例 | 11 |
| 3.2 使用端子一覧 | 12 |
| 4. ソフトウェア説明 | 12 |
| 4.1 使用する周辺機能 | 12 |
| 4.2 オプション・バイトの設定一覧 | 13 |
| 4.3 フォルダ構成 | 14 |
| 4.4 変数一覧 | 15 |
| 4.5 定数一覧 | 16 |
| 4.6 列挙体一覧 | 19 |
| 4.6.1 e_system_status_t 列挙体 | 19 |
| 4.6.2 e_mode_select_t 列挙体 | 19 |
| 4.6.3 e_stew_status_t 列挙体 | 19 |
| 4.6.4 e_button_status_t 列挙体 | 20 |
| 4.7 関数一覧 | 21 |
| 4.8 関数仕様 | 22 |
| 4.9 フローチャート | 27 |
| 4.9.1 main 関数のフローチャート | 27 |
| 4.9.2 system_init 関数のフローチャート | 33 |
| 4.9.3 r_Config_ADC_interrupt 関数のフローチャート | 34 |
| 4.9.4 calculate_resistance 関数のフローチャート | 34 |
| 4.9.5 calculate_temperature 関数のフローチャート | 35 |
| 4.9.6 r_get_temperature 関数のフローチャート | 36 |
| 4.9.7 r_Config_INTC_intp0_interrupt 関数のフローチャート | 37 |
| 4.9.8 r_Config_IT_interrupt 関数のフローチャート | 38 |
| 4.9.9 r_start_stew_mode 関数のフローチャート | 39 |
| 4.9.10 r_control_stew_mode 関数のフローチャート | 40 |
| 4.9.11 r_led_sequence 関数のフローチャート | 42 |
| 4.9.12 display_setting_led 関数のフローチャート | 43 |
| 4.9.13 r_display_temp_led 関数のフローチャート | 44 |
| 4.9.14 r_display_time_led 関数のフローチャート | 45 |
| 4.9.15 r_control_heater_pwm 関数のフローチャート | 46 |
| 4.9.16 r_start_boil_mode 関数のフローチャート | 47 |
| 4.9.17 r_stew_heater_pwm 関数のフローチャート | 48 |
| 4.9.18 r_get_target_temp 関数のフローチャート | 49 |
| 4.9.19 r_stop_pwm_output 関数のフローチャート | 50 |
| 4.9.20 r_Config_TAU0_2_interrupt 関数のフローチャート | 51 |
| 4.10 応用例 | 52 |

| | |
|--|----|
| 4.10.1 r01an7611_rl78g15_heater_control.scfg | 52 |
| 4.10.1.1 クロック | 54 |
| 4.10.1.2 システム | 54 |
| 4.10.1.3 r_bsp..... | 54 |
| 4.10.1.4 Config_INTC..... | 54 |
| 4.10.1.5 Config_IT | 54 |
| 4.10.1.6 Config_TAU0_0..... | 55 |
| 4.10.1.7 Config_TAU0_2..... | 55 |
| 4.10.1.8 Config_ADC | 55 |
| 4.10.2 温度、時間設定の変更方法 | 56 |
| | |
| 5. プロジェクトのインポート方法..... | 57 |
| 5.1 e ² studio での手順 | 57 |
| 5.2 CS+ での手順 | 58 |
| | |
| 6. 参考資料..... | 59 |
| | |
| 改訂記録 | 60 |

1. 仕様

1.1 仕様概要

本アプリケーションノートでは、PWM 信号で MOSFET を制御し、ヒータの温度制御を行います。サーミスタの温度検出をもとに PWM 信号のデューティ比を変化させヒータの発熱量を調整します。PWM 信号を調整することで、ヒータの発熱量を細かく制御し、安定した温度管理を実現します。

1.2 動作詳細

本アプリケーションノートでは、PWM 機能を使用したヒータの温度制御の使用例を示しています。

Fast Prototyping Board (FPB) 上のボタン (SW2) を押すことで、動作モード、温度、時間の設定を行い、ヒータの温度制御を行います。各設定の詳細は表 1-1 に示します。

表 1-1 設定一覧

| 動作モード | 温度設定 | 時間設定 |
|-----------------|-------------------------|-----------------------|
| 湯沸かし (Boil) モード | 40°C、55°C、70°C (3 段階) | — |
| 煮込み (Stew) モード | 70°C (固定) ^{注1} | 10 分、40 分、70 分 (3 段階) |

注1. 本アプリケーションノートで使用しているヒータの仕様上 70°Cとしています。

動作モード LED が点滅中にボタンを 1 秒以上長押しすることで動作モードの切り替えを行うことができます。ボタンを短押しすると湯沸かし (Boil) モードの場合は温度設定、煮込み (Stew) モードの場合は時間設定を変更します。設定した温度や時間は、3 段階の LED 表示で行います。

各種設定後、3 秒以上ボタン操作がない場合、設定した動作モードでヒータの制御を開始します。

どちらの動作モードも動作中は該当する動作モード LED が点灯します。動作中は温度や時間に合わせて LED の点灯パターンが変化します。また、動作中にボタンを 2 秒以上長押しすることで動作を停止し、スタンバイモードへ遷移します。動作中に 80°C を超えた場合は緊急停止します。再度動作を開始する場合はリセットを行ってください。

本アプリケーションの状態遷移図を図 1-1 に、各状態の説明を表 1-2 に示します。

図 1-1 温度制御の状態遷移図

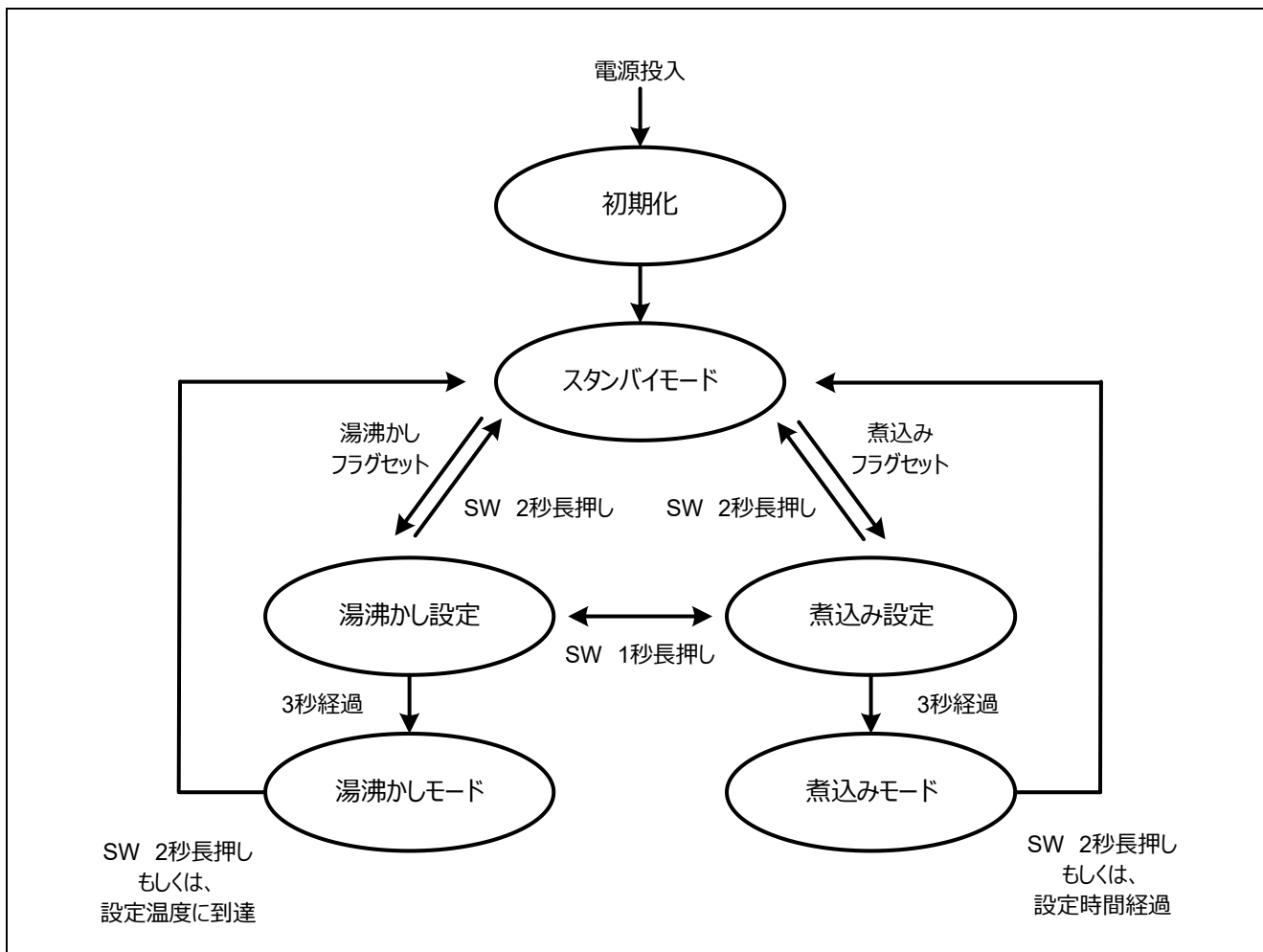
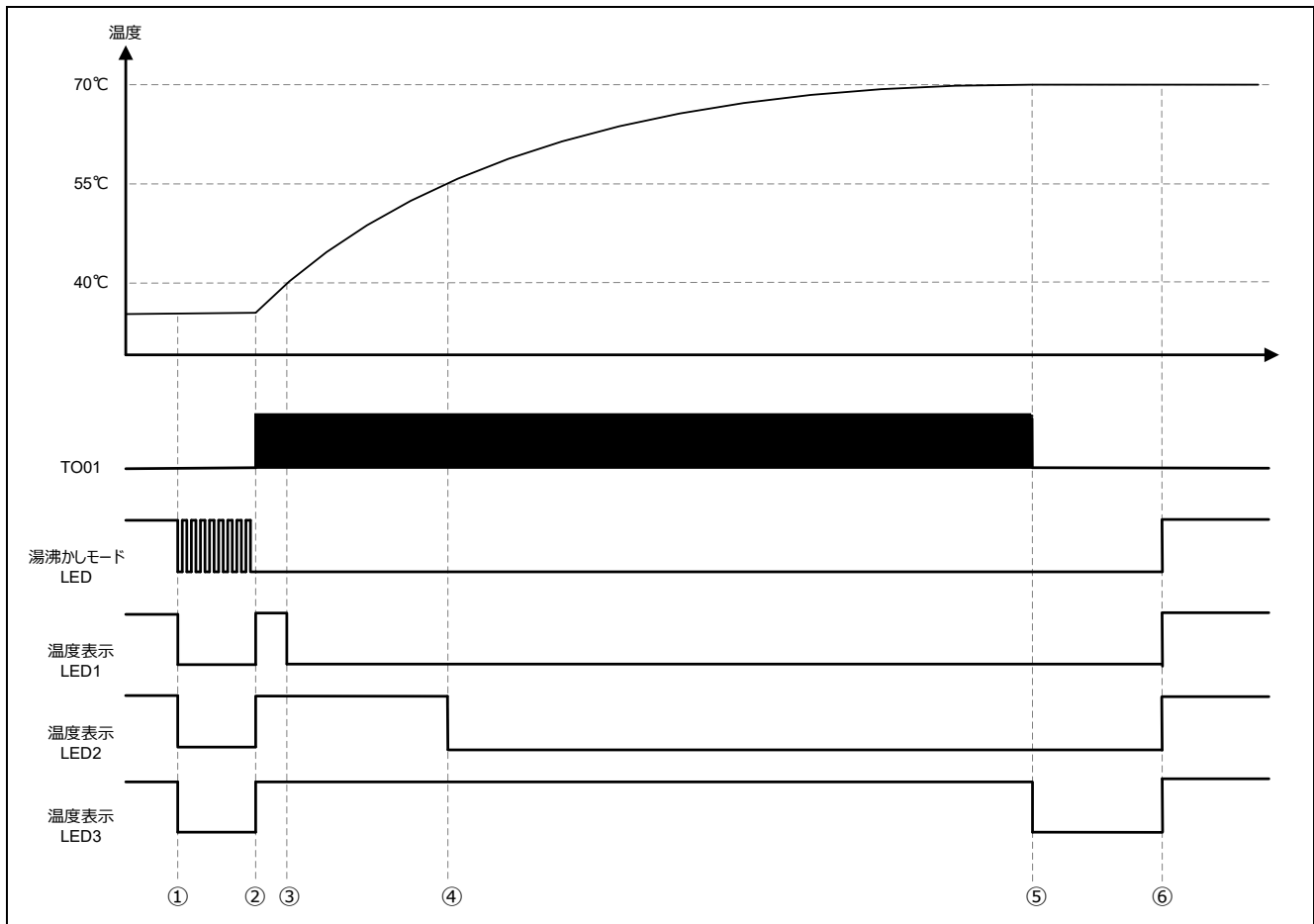


表 1-2 各状態の遷移要因と処理内容

| 状態 | 遷移要因 | 処理内容 |
|---------|--|--|
| 初期状態 | <ul style="list-style-type: none"> 電源投入時 | <ul style="list-style-type: none"> 初期化处理 LED 全点灯 |
| スタンバイ | <ul style="list-style-type: none"> 初期化处理の完了 湯沸かし/煮込み設定のキャンセル (SW 2 秒長押し) 湯沸かし/煮込みモードのキャンセル (SW 2 秒長押し) 湯沸かしモードで設定温度に到達 煮込みモードで設定時間の経過 | <ul style="list-style-type: none"> ボタン操作 (湯沸かし/煮込み設定) の待機 湯沸かしフラグセット (初回のみ) インターバル・タイマの停止 LED 全消灯 |
| 湯沸かし設定 | <ul style="list-style-type: none"> スタンバイモード中のボタン操作 (湯沸かしフラグセット時) 煮込み設定中の動作モード変更 (SW 1 秒長押し) | <ul style="list-style-type: none"> 湯沸かし温度を 3 段階設定 (湯沸かし LED 点滅) 動作モード変更受付 湯沸かし設定キャンセル受付 湯沸かしフラグセット |
| 煮込み設定 | <ul style="list-style-type: none"> スタンバイモード中のボタン操作 (煮込みフラグセット時) 湯沸かし設定中の動作モード変更 (SW 1 秒長押し) | <ul style="list-style-type: none"> 煮込み時間を 3 段階設定 (煮込み LED の点滅) 動作モード変更受付 煮込み設定キャンセル受付 煮込みフラグセット |
| 湯沸かしモード | <ul style="list-style-type: none"> 湯沸かし設定中にボタン操作がない状態が 3 秒経過 | <ul style="list-style-type: none"> 設定温度になるまで PWM 信号を出力する 現在の温度の LED 点灯表示 湯沸かしモードキャンセル受付 設定温度へ達したらスタンバイモードへ遷移する |
| 煮込みモード | <ul style="list-style-type: none"> 煮込み設定中にボタン操作がない状態が 3 秒経過 | <ul style="list-style-type: none"> 規定温度 (70°C) に到達後、設定時間の間、規定温度を維持する PWM 信号を出力する 現在の時間の LED 点灯表示 煮込みモードキャンセル受付 設定時間が経過したらスタンバイモードへ遷移する |

湯沸かし（Boil）モードは設定温度に達すると自動で運転を停止するモードです。動作詳細を図 1-2 に示します。温度上昇のグラフはイメージです。

図 1-2 湯沸かし（Boil）モード動作詳細（70°C設定時）

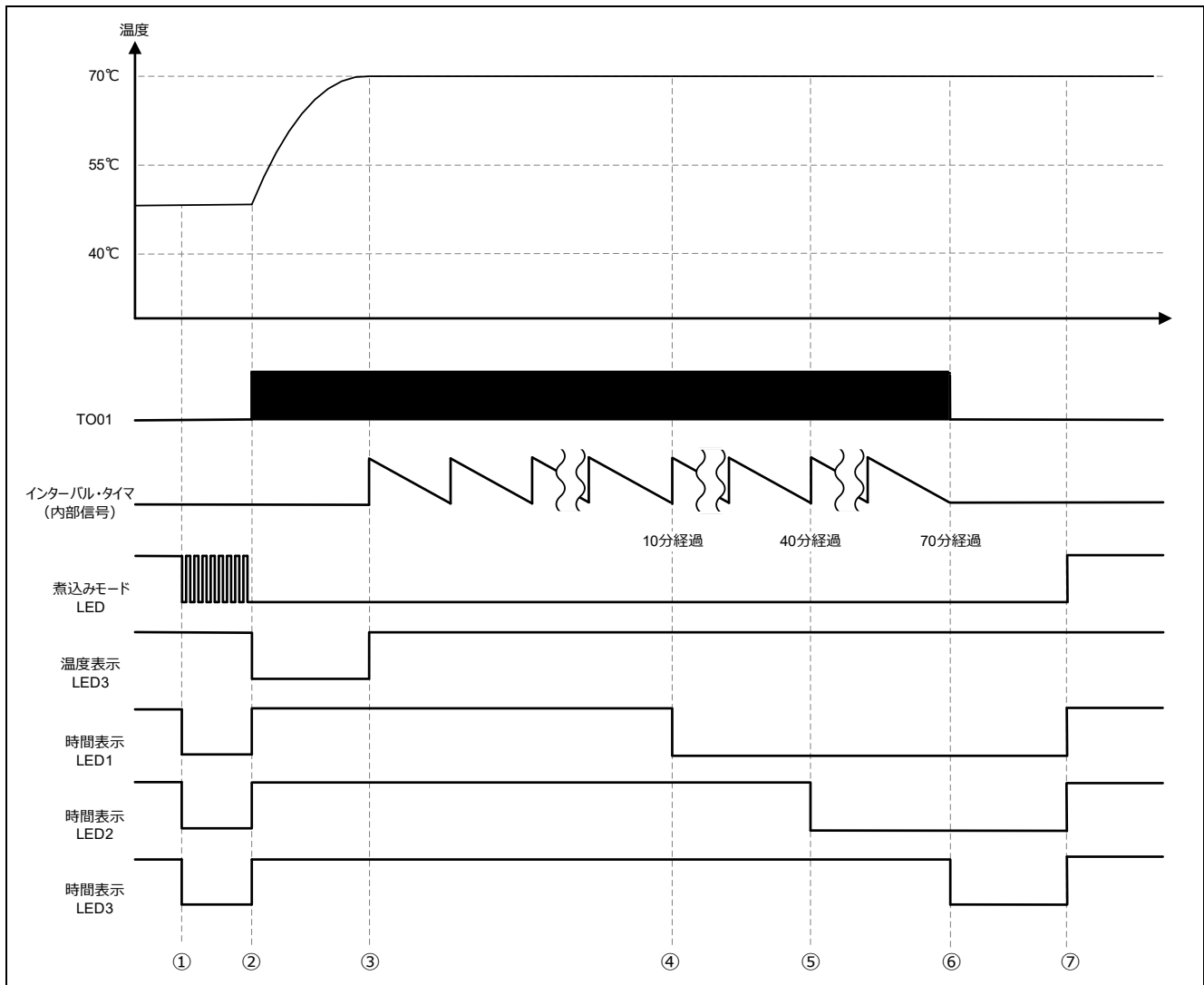


備考 TO01 波形の黒塗りは PWM 出力の High/Low の連続変化を示します。
波形は図 1-4 を参照してください。
また、LED はアクティブ Low で点灯します。

- ① 湯沸かしモード LED が点滅し、温度設定を行います。
(図の場合は 70°C に設定しているため温度表示 LED 全点灯)
- ② 設定後、3 秒以上ボタン操作がない場合ヒータ動作を開始します。動作中はサーミスタで検出した温度と設定温度との差分にもとづき、PWM 信号のデューティ比を変化させ、ヒータの発熱量をフィードバック制御します。また、湯沸かしモード LED と現在温度に合わせて温度表示 LED を点灯します。
(図の場合は 40°C を下回っているため温度表示 LED 全消灯)
- ③ 40°C に達したタイミングで温度表示 LED1 を点灯します。
- ④ 55°C に達したタイミングで温度表示 LED2 を点灯します。
- ⑤ 70°C（設定温度）に達したタイミングで温度表示 LED3 を点灯します。また、ヒータ動作を停止します。
- ⑥ スタンバイモードに遷移後、すべての LED を消灯します。

煮込み（Stew）モードは規定温度（70℃）に達してから指定時間が経過すると自動で運転を停止するモードです。動作詳細を図 1-3 に示します。温度上昇のグラフはイメージです。

図 1-3 煮込み（Stew）モード動作詳細（70分設定時）



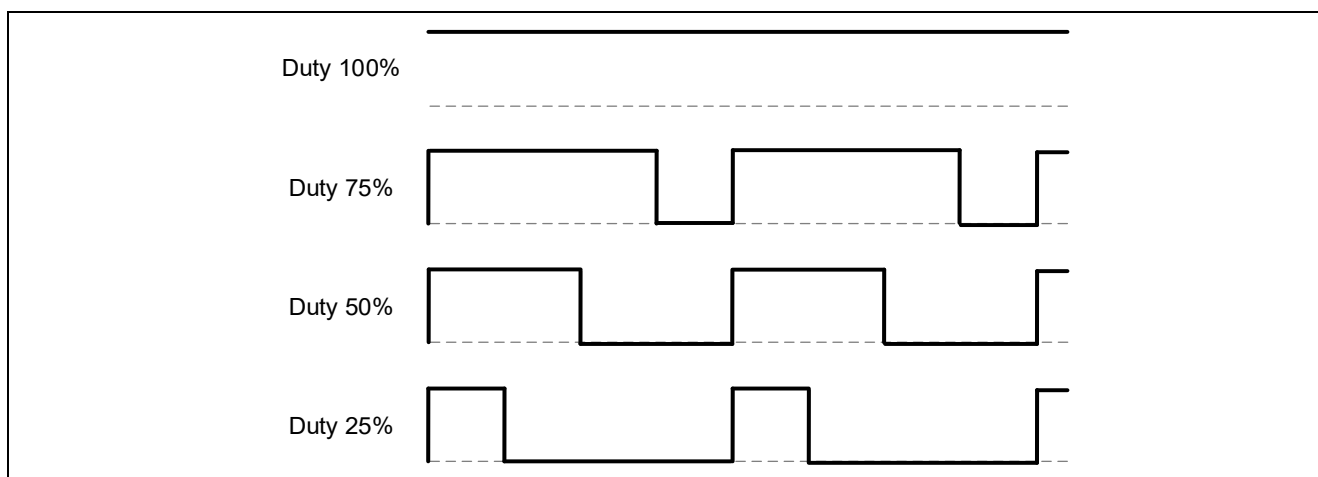
備考 TO01 波形の黒塗りは PWM 出力の High/Low の連続変化を示します。
 波形は図 1-4 を参照してください。
 また、LED はアクティブ Low で点灯します。

- ① 煮込みモード LED が点滅し、時間設定を行います。
(図の場合は 70 分に設定しているため時間表示 LED 全点灯)
- ② 設定後、3 秒以上ボタン操作がない場合ヒータ動作を開始します。動作中はサーミスタで検出した温度と設定温度との差分にもとづき、PWM 信号のデューティ比を変化させ、ヒータの発熱量をフィードバック制御します。また、湯沸かしモード LED と温度表示 LED3 のみを点灯します。時間表示 LED を全消灯します。
- ③ 規定温度 (70°C) に達したタイミングで温度表示 LED3 を消灯し、インターバル・タイマの動作を開始します。
- ④ 規定温度 (70°C) に達してから 10 分経過したタイミングで時間表示 LED1 を点灯します。
- ⑤ 規定温度 (70°C) に達してから 40 分経過したタイミングで時間表示 LED2 を点灯します。
- ⑥ 規定温度 (70°C) に達してから 70 分 (設定時間) 経過したタイミングで温度表示 LED3 を点灯します。また、ヒータ動作とインターバル・タイマを停止します。
- ⑦ スタンバイモードに遷移後、すべての LED を消灯します。

TO01 の PWM 波形はデューティ 100%、75%、50%、25% の 4 パターンがあります。現在温度と設定温度の差分にもとづき、デューティ比が設定されます。

波形を以下に示します。

図 1-4 TO01 の PWM 波形



湯沸かしモード及び煮込みモードにおける、設定温度までの温度差分と出力する PWM 信号のデューティ比の対応表を以下に示します。

表 1-3 設定温度までの温度差分と出力する PWM 信号のデューティ比の対応表

| 温度差分 | 30°Cより大きい | 20°Cより大きい | 10°Cより大きい | 10°C以下 |
|--------|-----------|-----------|-----------|--------|
| デューティ比 | 100% | 75% | 50% | 25% |

2. 動作確認条件

本サンプルプログラムは、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

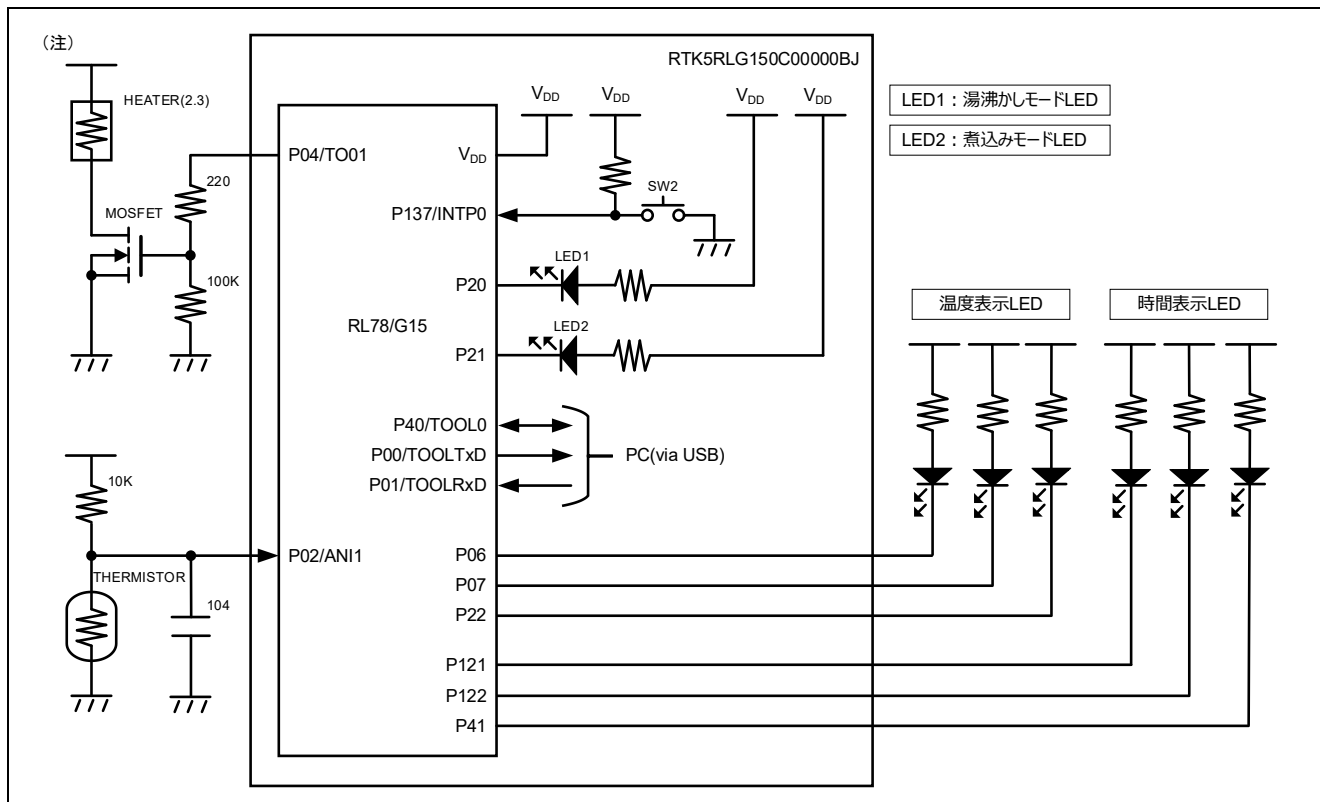
| 項目 | 内容 |
|---------------------------------|---|
| 使用マイコン | RL78/G15 (R5F12068ASP) |
| 使用ボード | RL78/G15 Fast Prototyping Board (RTK5RLG150C00000BJ) |
| 動作周波数 | <ul style="list-style-type: none"> 高速オンチップ・オシレータ・クロック : 16MHz CPU/周辺ハードウェア・クロック : 16MHz |
| 動作電圧 | <ul style="list-style-type: none"> 5V SPOR 検出電圧設定 立ち上がり時 TYP. 2.90V 立ち下がり時 TYP. 2.84V |
| 統合開発環境 (CS+) | ルネサスエレクトロニクス製 <ul style="list-style-type: none"> CS+ for CC V8.12.0 |
| C コンパイラ (CS+) | ルネサスエレクトロニクス製 CC-RL V1.14 |
| 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 2024-10 (24.10.1) |
| C コンパイラ (e ² studio) | ルネサスエレクトロニクス製 CC-RL V1.14 |
| スマート・コンフィグレータ | V.1.11.0 |
| ボードサポートパッケージ (r_bsp) | V.1.70 |
| ヒータ | AL-DC-A6-30 アルミ箔フレキシブルヒータ |
| サーミスタ | 103AT-2 AT Thermistor |

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 本アプリケーションノートに記載のヒータを使用する場合、70°C以上の温度にするためには9V以上の電源を供給する必要があります。詳細はヒータの電気的特性情報をご確認ください。

備考 サーマスタとMOSFETのGNDは離して接続してください。

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 |
|------------|-----|------------------------------|
| P02 / ANI1 | 入力 | サーミスタに接続、抵抗値を取得し、A/D 変換を行う |
| P04 / TO01 | 出力 | リレー (MOSFET) に接続、PWM 信号を出力する |
| P20 | 出力 | 湯沸かしモード選択 LED |
| P21 | 出力 | 煮込みモード選択 LED |
| P06 | 出力 | 湯沸かしモード 3 段階の温度表示 LED |
| P07 | 出力 | |
| P22 | 出力 | |
| P121 | 出力 | 煮込みモード 3 段階の温度表示 LED |
| P122 | 出力 | |
| P41 | 出力 | |
| P137/INTP0 | 入力 | SW 入力 |

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 使用する周辺機能

本サンプルプログラムで使用する周辺機能を以下に示します。

表 4-1 使用する周辺機能一覧

| 周辺機能 | 用途 |
|------------------------|--|
| 割り込みコントローラ (INTC) | 外部割り込み (立ち下りエッジ) でボタン押下検知 |
| A/D コンバータ (ADC) | サーミスタの抵抗値を取得する (10 ビット分解能) |
| タイマ・アレイ・ユニット (TAU) | PWM マスタ (周期設定) |
| | PWM スレーブ (デューティ比制御)、PWM 出力 |
| | ボタン長押し (モード切り替え、スタンバイ遷移) 判定、動作モード確定判定、チャタリング対策 |
| 12 ビット・インターバル・タイマ (IT) | 煮込みモードの時間管理 |
| PORT | LED 制御 |

4.2 オプション・バイトの設定一覧

本サンプルプログラムのオプション・バイト設定を以下に示します。

表 4-2 オプション・バイト設定

| アドレス | 設定値 | 内容 |
|-----------------|-------------------|---|
| 000C0H / 020C0H | 1110 1111B (0xEF) | 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 / 020C1H | 1111 1011B (0xFB) | P125 の端子制御 : RESET入力 SPOR 検出電圧設定 立ち上がり時 TYP. 2.90V(2.76V ~ 3.02V) 立ち下がり時 TYP. 2.84V(2.70V ~ 2.96V) |
| 000C2H / 020C2H | 1111 1001B (0xF9) | 高速オンチップ・オシレータ・クロック : 16MHz 動作電圧範囲 : 2.4V ~ 5.5V |
| 000C3H / 020C3H | 1000 0101B (0x85) | オンチップ・デバッグ許可 |

4.3 フォルダ構成

本サンプルプログラムのソースファイル／ヘッダファイルの構成を以下に示します。

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 4-3 フォルダ構成

| フォルダ、ファイル名 | 説明 | スマート・コンフィグレータを使用 |
|--|-----------------------|------------------|
| ¥r01an7611_rl78g15_heater_control<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 | |
| ¥smc_gen<DIR> | スマート・コンフィグレータ生成フォルダ | √ |
| ¥Config_ADC<DIR> | ADC 用プログラム格納フォルダ | √ |
| Config_ADC.c | ADC 用ソースファイル | √ |
| Config_ADC.h | ADC 用ヘッダファイル | √ |
| Config_ADC_user.c | ADC 用割り込み、温度計測ソースファイル | √ |
| ¥Config_INTC<DIR> | INTC 用プログラム格納フォルダ | √ |
| Config_INTC.c | INTC 用ソースファイル | √ |
| Config_INTC.h | INTC 用ヘッダファイル | √ |
| Config_INTC_user.c | INTC 用割り込みソースファイル | √ |
| ¥Config_IT<DIR> | IT 用プログラム格納フォルダ | √ |
| Config_IT.c | IT 用ソースファイル | √ |
| Config_IT.h | IT 用ヘッダファイル | √ |
| Config_IT_user.c | IT 用割り込みソースファイル | √ |
| ¥Config_PORT<DIR> | PORT 用プログラム格納フォルダ | √ |
| Config_PORT.c | PORT 用ソースファイル | √ |
| Config_PORT.h | PORT 用ヘッダファイル | √ |
| Config_PORT_user.c | LED 制御用ソースファイル | √ |
| ¥Config_TAU0_0 <DIR> | TAU0_0 用プログラム格納フォルダ | √ |
| Config_TAU0_0.c | TAU0_0 用ソースファイル | √ |
| Config_TAU0_0.h | TAU0_0 用ヘッダファイル | √ |
| Config_TAU0_0_user.c | ヒータ制御用ソースファイル | √ |
| ¥Config_TAU0_2<DIR> | TAU0_2 用プログラム格納フォルダ | √ |
| Config_TAU0_2.c | TAU0_2 用ソースファイル | √ |
| Config_TAU0_2.h | TAU0_2 用ヘッダファイル | √ |
| Config_TAU0_2_user.c | TAU0_2 用割り込みソースファイル | √ |

補足 ” <DIR>” は、ディレクトリを意味します。

4.4 変数一覧

本サンプルプログラムで使用する変数一覧を以下に示します。

表 4-4 変数一覧

| 型 | 変数名 | 内容 | ファイル |
|-------------------|-------------------------|------------------------------|--|
| e_system_status_t | g_system_status | アプリケーションの状態を示す変数 | main.c Config_TAU0_0_user.c |
| e_mode_select_t | g_mode_flag | ヒータの動作モードを示す変数 | main.c |
| uint8_t | g_boil_mode_initialized | 湯沸かしモード用初期化フラグ | main.c |
| int8_t | g_current_temp | 温度を示す変数 | main.c Config_TAU0_0_user.c |
| uint8_t | g_adc_done | A/D 割り込みフラグ | Config_ADC_user.c |
| e_stew_status_t | g_stew_status | 煮込みモードの状態を示す変数 | Config_IT_user.c main.c |
| uint16_t | g_stew_time_count | 煮込みモードの時間カウント用変数 | Config_IT_user.c main.c |
| uint16_t | g_stew_time_set | 煮込みモードの時間カウント設定用変数 | Config_IT_user.c Config_PORT_user.c |
| uint8_t | g_pwm_started | PWM 出力フラグ | Config_IT_user.c main.c Config_TAU0_0_user.c |
| uint8_t | g_led_blink_flag | LED 点滅制御用 0 : オフ、1 : オン | Config_IT_user.c main.c |
| uint8_t | g_led_blink_count | LED 点滅判定カウンター | Config_IT_user.c |
| uint8_t | g_heater_status | 温度状態フラグ 0 : 正常動作、1 : 異常停止 | Config_TAU0_0_user.c Config_IT_user.c |
| uint16_t | g_button_press_time | ボタン押下時間 | Config_TAU0_2_user.c main.c Config_INTC_user.c |
| uint8_t | g_button_released | ボタンリリース判定 | Config_TAU0_2_user.c main.c Config_INTC_user.c |
| uint16_t | g_no_operation_time | ボタン操作がない時間 | Config_TAU0_2_user.c main.c Config_INTC_user.c |
| uint8_t | g_button_1sec_detected | ボタン 1 秒以上押下検出 | Config_TAU0_2_user.c main.c |
| uint8_t | g_button_2sec_detected | ボタン 2 秒以上押下検出 | Config_TAU0_2_user.c main.c |

4.5 定数一覧

本サンプルプログラムで使用する定数一覧を以下に示します。

表 4-5 定数一覧 (1/3)

| 定数名 | 設定値 | 内容 | ファイル |
|----------------------|--------|----------------------|---|
| SERIES_R | 10000 | 直列抵抗 (10kΩ) | r_cg_userdefine.h Config_ADC_user.c |
| ADC_MAX | 1023 | 10 ビット ADC の 最大値 | r_cg_userdefine.h Config_ADC_user.c |
| TEMP_ABNORMAL | 80 | 異常温度 | r_cg_userdefine.h Config_IT_user.c Config_TAU0_0_user.c |
| TARGET_TEMP_LO W | 40 | 湯沸かしモードの 設定温度 (低) | r_cg_userdefine.h Config_PORT_user.c Config_TAU0_0_user.c |
| TARGET_TEMP_MI D | 55 | 湯沸かしモードの 設定温度 (中) | r_cg_userdefine.h Config_PORT_user.c Config_TAU0_0_user.c |
| TARGET_TEMP_HI GH | 70 | 湯沸かしモードの 設定温度 (高) | r_cg_userdefine.h Config_PORT_user.c Config_TAU0_0_user.c |
| TEMP_HYSTERESI S | 2 | 温度ヒステリシス | r_cg_userdefine.h Config_TAU0_0_user.c |
| DUTY_100 | 0x3E80 | デューティ比 100% | r_cg_userdefine.h Config_TAU0_0_user.c |
| DUTY_75 | 0x2EDF | デューティ比 75% | r_cg_userdefine.h Config_TAU0_0_user.c |
| DUTY_50 | 0x1F40 | デューティ比 50% | r_cg_userdefine.h Config_TAU0_0_user.c |
| DUTY_25 | 0x0F9F | デューティ比 25% | r_cg_userdefine.h Config_TAU0_0_user.c |

表 4-6 定数一覧 (2/3)

| 定数名 | 設定値 | 内容 | ファイル |
|----------------------|------------------|-------------------------------------|---|
| STEW_TIME_SHORT | 10 | 煮込みモードの設定時間 (短) | r_cg_userdefine.h Config_IT_user.c |
| STEW_TIME_MIDDLE | 40 | 煮込みモードの設定時間 (中) | r_cg_userdefine.h Config_IT_user.c |
| STEW_TIME_LONG | 70 | 煮込みモードの設定時間 (長) | r_cg_userdefine.h Config_IT_user.c |
| STEW_TEMP | TARGET_TEMP_HIGH | 煮込みモードの設定温度 | r_cg_userdefine.h Config_IT_user.c Config_TAU0_0_user.c |
| PRESS_TIME_1SEC | 100 | 1 秒ボタン長押し | r_cg_userdefine.h main.c Config_TAU0_2_user.c |
| PRESS_TIME_2SEC | 200 | 2 秒ボタン長押し | r_cg_userdefine.h Config_TAU0_2_user.c |
| NO_PRESS_TIME | 300 | 3 秒間ボタン操作がない | r_cg_userdefine.h main.c |
| LED_DELAY_TIME | 100 | 初期化時の LED 点灯時間 (ms) | r_cg_userdefine.h Config_PORT_user.c |
| LED_MIN_ON_TIME | 100 | 温度表示 LED の最小点灯時間 (10ms × 100 = 1 秒) | r_cg_userdefine.h Config_PORT_user.c |
| LED_BOIL_MODE_ON() | P2_bit.no0 = 0U | 湯沸かしモード LED を点灯 | r_cg_userdefine.h main.c Config_PORT_user.c |
| LED_BOIL_MODE_OFF() | P2_bit.no0 = 1U | 湯沸かしモード LED を消灯 | r_cg_userdefine.h main.c Config_PORT_user.c |
| LED_BOIL_TEMP1_ON() | P0_bit.no6 = 0U | 温度表示 LED1 を点灯 | r_cg_userdefine.h Config_PORT_user.c |
| LED_BOIL_TEMP1_OFF() | P0_bit.no6 = 1U | 温度表示 LED1 を消灯 | r_cg_userdefine.h main.c Config_PORT_user.c |
| LED_BOIL_TEMP2_ON() | P0_bit.no7 = 0U | 温度表示 LED2 を点灯 | r_cg_userdefine.h Config_PORT_user.c |
| LED_BOIL_TEMP2_OFF() | P0_bit.no7 = 1U | 温度表示 LED2 を消灯 | r_cg_userdefine.h main.c Config_PORT_user.c |
| LED_BOIL_TEMP3_ON() | P2_bit.no2 = 0U | 温度表示 LED3 を点灯 | r_cg_userdefine.h Config_IT_user.c Config_PORT_user.c |
| LED_BOIL_TEMP3_OFF() | P2_bit.no2 = 1U | 温度表示 LED3 を消灯 | r_cg_userdefine.h main.c Config_PORT_user.c |

表 4-7 定数一覧 (3/3)

| 定数名 | 設定値 | 内容 | ファイル |
|----------------------|---|----------------|---|
| LED_STEW_MODE_ON() | P2_bit.no1 = 0U | 煮込みモード LED を点灯 | r_cg_userdefine.h main.c Config_PORT_user.c |
| LED_STEW_MODE_OFF() | P2_bit.no1 = 1U | 煮込みモード LED を消灯 | r_cg_userdefine.h main.c Config_PORT_user.c |
| LED_STEW_TIME1_ON() | P12_bit.no1 = 0U | 時間表示 LED1 を点灯 | r_cg_userdefine.h Config_PORT_user.c |
| LED_STEW_TIME1_OFF() | P12_bit.no1 = 1U | 時間表示 LED1 を消灯 | r_cg_userdefine.h main.c Config_PORT_user.c |
| LED_STEW_TIME2_ON() | P12_bit.no2 = 0U | 時間表示 LED2 を点灯 | r_cg_userdefine.h Config_PORT_user.c |
| LED_STEW_TIME2_OFF() | P12_bit.no2 = 1U | 時間表示 LED2 を消灯 | r_cg_userdefine.h main.c Config_PORT_user.c |
| LED_STEW_TIME3_ON() | P4_bit.no1 = 0U | 時間表示 LED3 を点灯 | r_cg_userdefine.h Config_PORT_user.c |
| LED_STEW_TIME3_OFF() | P4_bit.no1 = 1U | 時間表示 LED3 を消灯 | r_cg_userdefine.h main.c Config_PORT_user.c |
| LED_ALL_OFF() | do { ¥ LED_BOIL_MODE_OFF(); ¥ LED_STEW_MODE_OFF(); ¥ LED_BOIL_TEMP1_OFF(); ¥ LED_BOIL_TEMP2_OFF(); ¥ LED_BOIL_TEMP3_OFF(); ¥ LED_STEW_TIME1_OFF(); ¥ LED_STEW_TIME2_OFF(); ¥ LED_STEW_TIME3_OFF(); ¥ } while (0) | すべての LED を消灯 | r_cg_userdefine.h main.c Config_PORT_user.c |
| thermistor_table[] | -50°C~110°Cの範囲で B 定数を用いて計算した値を格納 | サーミスタの温度特性 | Config_ADC_user.c |

4.6 列挙体一覧

本サンプルプログラムで使用する列挙体一覧を以下に示します。

4.6.1 e_system_status_t 列挙体

本アプリケーションの状態を管理する列挙体です。

表 4-8 e_system_status_t 列挙体

| 定義 | 値 | 内容 | ファイル |
|---------------------|---|----------|----------------------|
| SYSTEM_INIT | 0 | 初期化状態 | r_cg_userdefine.h |
| SYSTEM_STANDBY | 1 | スタンバイモード | main.c |
| SYSTEM_BOIL_SETTING | 2 | 湯沸かし設定 | Config_TAU0_0_user.c |
| SYSTEM_STEW_SETTING | 3 | 煮込み設定 | |
| SYSTEM_BOIL_MODE | 4 | 湯沸かしモード | |
| SYSTEM_STEW_MODE | 5 | 煮込みモード | |

4.6.2 e_mode_select_t 列挙体

ヒータ動作のモード選択状態を管理する列挙体です。

表 4-9 e_mode_select_t 列挙体

| 定義 | 値 | 内容 | ファイル |
|-----------|---|---------|------------------------------|
| MODE_BOIL | 0 | 湯沸かしモード | r_cg_userdefine.h |
| MODE_STEW | 1 | 煮込みモード | main.c Config_PORT_user.c |

4.6.3 e_stew_status_t 列挙体

煮込みモードの状態を管理する列挙体です。

表 4-10 e_stew_status_t 列挙体

| 定義 | 値 | 内容 | ファイル |
|---------------|---|-------------|-------------------|
| STEW_IDLE | 0 | 煮込みモードの待機状態 | r_cg_userdefine.h |
| STEW_HEATING | 1 | 煮込みモードの加熱 | main.c |
| STEW_RUNNING | 2 | 煮込みモードの温度維持 | Config_IT_user.c |
| STEW_COMPLETE | 3 | 煮込み完了 | |

4.6.4 e_button_status_t 列挙体

ボタンの押下状況を管理する列挙体です。

表 4-11 e_button_status_t 列挙体

| 定義 | 値 | 内容 | ファイル |
|----------------|---|------------------|---|
| BUTTON_IDLE | 0 | ボタン未押下状態 | r_cg_userdefine.h main.c Config_IT_user.c |
| BUTTON_DETECT | 1 | ボタン押下開始 | |
| BUTTON_RELEASE | 2 | ボタンリリース待ち状態 | |
| BUTTON_STANDBY | 3 | スタンバイモード遷移後の待機状態 | |

4.7 関数一覧

本サンプルプログラムで使用する関数一覧を以下に示します。

表 4-12 関数一覧

| 関数名 | 概要 | ソースファイル |
|-------------------------------|-----------------------------------|--|
| main | メイン処理 | main.c |
| system_init | 初期化設定処理 | main.c |
| r_Config_ADC_interrupt | A/D 完了割り込み処理 | Config_ADC_user.c |
| calculate_resistance | A/D 変換結果からサーミスタの抵抗値の計算処理 | Config_ADC_user.c |
| calculate_temperature | サーミスタの抵抗値から温度の計算処理 | Config_ADC_user.c |
| r_get_temperature | A/D の変換値から温度 (°C) 取得処理 | main.c Config_ADC_user.c |
| r_Config_INTC_intp0_interrupt | SW2 の外部割り込み処理 | Config_INTC_user.c |
| r_Config_IT_interrupt | インターバル・タイマ割り込み処理 | Config_IT_user.c |
| r_start_stew_mode | 煮込みモードの設定時間をインターバル・タイマの割り込み回数変換処理 | main.c Config_IT_user.c |
| r_control_stew_mode | 煮込みモード全体のコントロール処理 | main.c Config_IT_user.c |
| r_led_sequence | LED の点灯・消灯のタイミングを一定にする処理 | main.c Config_PORT_user.c |
| display_setting_led | 設定内容を LED で表示する処理 | main.c Config_PORT_user.c |
| r_display_temp_led | 現在温度を LED で表示する処理 | main.c Config_PORT_user.c |
| r_display_time_led | 経過時間を LED で表示する処理 | main.c Config_PORT_user.c |
| r_control_heater_pwm | 現在の温度に応じて PWM 出力のデューティを変化させる処理 | main.c Config_TAU0_0_user.c |
| r_start_boil_mode | 湯沸かしモード開始処理 | main.c Config_TAU0_0_user.c |
| r_stew_heater_pwm | 煮込みモードの PWM の出力デューティ制御処理 | Config_TAU0_0_user.c Config_IT_user.c |
| r_get_target_temp | 湯沸かしモードの設定温度の取得処理 | main.c Config_TAU0_0_user.c Config_PORT_user.c |
| r_stop_pwm_output | PWM 出力を停止し、TO 出力端子を Low に固定する処理 | main.c Config_TAU0_0_user.c Config_IT_user.c |
| r_Config_TAU0_2_interrupt | TAU0_2 の割り込み処理 | Config_TAU0_2_user.c |

4.8 関数仕様

本サンプルプログラムの関数仕様を以下に示します。

[関数名] main

| | |
|-------|------------------|
| 概要 | メイン処理 |
| ヘッダ | なし |
| 宣言 | void main (void) |
| 説明 | ヒータ制御のメイン処理を行います |
| 引数 | なし |
| リターン値 | なし |
| 備考 | なし |

[関数名] system_init

| | |
|-------|------------------------|
| 概要 | 初期化設定処理 |
| ヘッダ | なし |
| 宣言 | void system_init(void) |
| 説明 | スタンバイモード遷移前の初期化設定を行います |
| 引数 | なし |
| リターン値 | なし |
| 備考 | なし |

[関数名] r_Config_ADC_interrupt

| | |
|-------|---|
| 概要 | A/D 完了割り込み処理 |
| ヘッダ | Config_ADC.h |
| 宣言 | static void __near r_Config_ADC_interrupt(void) |
| 説明 | A/D 動作完了フラグを立てます |
| 引数 | なし |
| リターン値 | なし |
| 備考 | なし |

[関数名] calculate_resistance

| | |
|-------|---|
| 概要 | A/D 変換結果からサーミスタの抵抗値の計算処理 |
| ヘッダ | Config_ADC.h |
| 宣言 | uint32_t calculate_resistance(uint16_t adc_value) |
| 説明 | A/D 変換結果からサーミスタの抵抗値を計算します |
| 引数 | adc_value : A/D 変換結果 |
| リターン値 | r : サーミスタの抵抗値 |
| 備考 | なし |

[関数名] calculate_temperature

| | |
|-------|--|
| 概要 | サーミスタの抵抗値から温度の計算処理 |
| ヘッダ | Config_ADC.h |
| 宣言 | int8_t calculate_temperature(uint16_t adc_value) |
| 説明 | サーミスタの抵抗値から温度を計算します |
| 引数 | adc_value : A/D 変換結果 |
| リターン値 | temp : 温度 |
| 備考 | なし |

[関数名] r_get_temperature

| | |
|-------|--------------------------------|
| 概要 | A/D 変換結果から温度の計算処理 |
| ヘッダ | Config_ADC.h |
| 宣言 | int8_t r_get_temperature(void) |
| 説明 | A/D 変換結果から温度を計算します |
| 引数 | なし |
| リターン値 | temperature : 温度 |
| 備考 | なし |

[関数名] r_Config_INTC_intp0_interrupt

| | |
|-------|--|
| 概要 | SW2 の外部割り込み処理 |
| ヘッダ | Config_INTC.h |
| 宣言 | static void __near r_Config_INTC_intp0_interrupt(void) |
| 説明 | SW2 押下の初期化を行います |
| 引数 | なし |
| リターン値 | なし |
| 備考 | なし |

[関数名] r_Config_IT_interrupt

| | |
|-------|--|
| 概要 | インターバル・タイマ割り込み処理 |
| ヘッダ | Config_IT.h |
| 宣言 | static void __near r_Config_IT_interrupt(void) |
| 説明 | 割り込み回数のカウントを行います |
| 引数 | なし |
| リターン値 | なし |
| 備考 | なし |

[関数名] r_start_stew_mode

| | |
|-------|--|
| 概要 | 煮込みモードの設定時間をインターバル・タイマの割り込み回数変換処理 |
| ヘッダ | Config_IT.h |
| 宣言 | void r_start_stew_mode(uint8_t stew_level) |
| 説明 | 煮込みモードの設定時間をインターバル・タイマの割り込み回数に置き換えます |
| 引数 | stew_level : 設定時間 |
| リターン値 | なし |
| 備考 | なし |

[関数名] r_control_stew_mode

| | |
|-------|---|
| 概要 | 煮込みモード全体のコントロール処理 |
| ヘッダ | Config_IT.h |
| 宣言 | void r_control_stew_mode(int8_t current_temp) |
| 説明 | 煮込みモード全体をコントロールします |
| 引数 | current_temp : 現在温度 |
| リターン値 | なし |
| 備考 | なし |

[関数名] r_led_sequence

| | |
|-------|---------------------------|
| 概要 | LED の点灯・消灯のタイミングを一定にする処理 |
| ヘッダ | Config_PORT.h |
| 宣言 | void r_led_sequence(void) |
| 説明 | LED の点灯・消灯のタイミングを一定にします |
| 引数 | なし |
| リターン値 | なし |
| 備考 | なし |

[関数名] display_setting_led

| | |
|-------|---|
| 概要 | 設定内容を LED で表示する処理 |
| ヘッダ | Config_PORT.h |
| 宣言 | void display_setting_led(uint8_t level, e_mode_select_t mode) |
| 説明 | 設定モードで点灯する LED を決定します |
| 引数 | level : 3 段階設定値 mode : 設定されている動作モード (湯沸かしモードまたは、煮込みモード) |
| リターン値 | なし |
| 備考 | なし |

[関数名] r_display_temp_led

| | |
|-------|---|
| 概要 | 現在温度を LED で表示する処理 |
| ヘッダ | Config_PORT.h |
| 宣言 | void r_display_temp_led(int8_t current_temp, uint8_t setting_level) |
| 説明 | 温度表示用 LED を制御し、現在温度を表します |
| 引数 | current_temp : 現在温度 setting_level : 3 段階設定値 |
| リターン値 | なし |
| 備考 | なし |

[関数名] r_display_time_led

| | |
|-------|--|
| 概要 | 経過時間を LED で表示する処理 |
| ヘッダ | Config_PORT.h |
| 宣言 | void r_display_time_led(uint16_t time_count) |
| 説明 | 時間表示用 LED を制御し、経過時間を表します |
| 引数 | time_count : インターバル・タイマの割り込みカウント数 |
| リターン値 | なし |
| 備考 | なし |

[関数名] r_control_heater_pwm

| | |
|-------|--|
| 概要 | 現在の温度に応じて PWM 出力のデューティを変化させる処理 |
| ヘッダ | Config_TAU0_0.h |
| 宣言 | void r_control_heater_pwm(int8_t current_temp, int8_t target_temp) |
| 説明 | 現在の温度に応じて PWM 出力のデューティを変化させます |
| 引数 | current_temp : A/D 変換で得られる現在温度 target_temp : 設定モードで設定した温度 |
| リターン値 | なし |
| 備考 | g_heater_status に 1 がセットされている場合、PWM 信号は出力されません。PWM 信号の再出力にはリセットが必要です |

[関数名] r_start_boil_mode

| | |
|-------|--|
| 概要 | 湯沸かしモード開始処理 |
| ヘッダ | Config_TAU0_0.h |
| 宣言 | void r_start_boil_mode(uint8_t boil_level) |
| 説明 | 湯沸かしモードの動作を開始します |
| 引数 | boil_level : 湯沸かしモードの設定温度 |
| リターン値 | なし |
| 備考 | なし |

[関数名] r_stew_heater_pwm

| | |
|-------|---|
| 概要 | 煮込みモードの PWM 出力のデューティ制御処理 |
| ヘッダ | Config_TAU0_0.h |
| 宣言 | void r_stew_heater_pwm(int8_t current_temp) |
| 説明 | 煮込みモードの PWM 出力のデューティを制御します |
| 引数 | current_temp : A/D 変換で得られる現在温度 |
| リターン値 | なし |
| 備考 | なし |

[関数名] r_get_target_temp

| | |
|-------|---|
| 概要 | 湯沸かしモードの設定温度の取得処理 |
| ヘッダ | Config_TAU0_0.h |
| 宣言 | int8_t r_get_target_temp(uint8_t level) |
| 説明 | 湯沸かしモードの設定温度を取得します |
| 引数 | level : 湯沸かしモードの温度設定 |
| リターン値 | 湯沸かしモードの設定温度 |
| 備考 | なし |

[関数名] r_stop_pwm_output

| | |
|-------|---------------------------------|
| 概要 | PWM 出力を停止し、TO 出力端子を Low に固定する処理 |
| ヘッダ | Config_TAU0_0.h |
| 宣言 | void r_stop_pwm_output(void) |
| 説明 | PWM 出力を停止し、TO 出力端子を Low に固定します |
| 引数 | なし |
| リターン値 | なし |
| 備考 | なし |

[関数名] r_Config_TAU0_2_interrupt

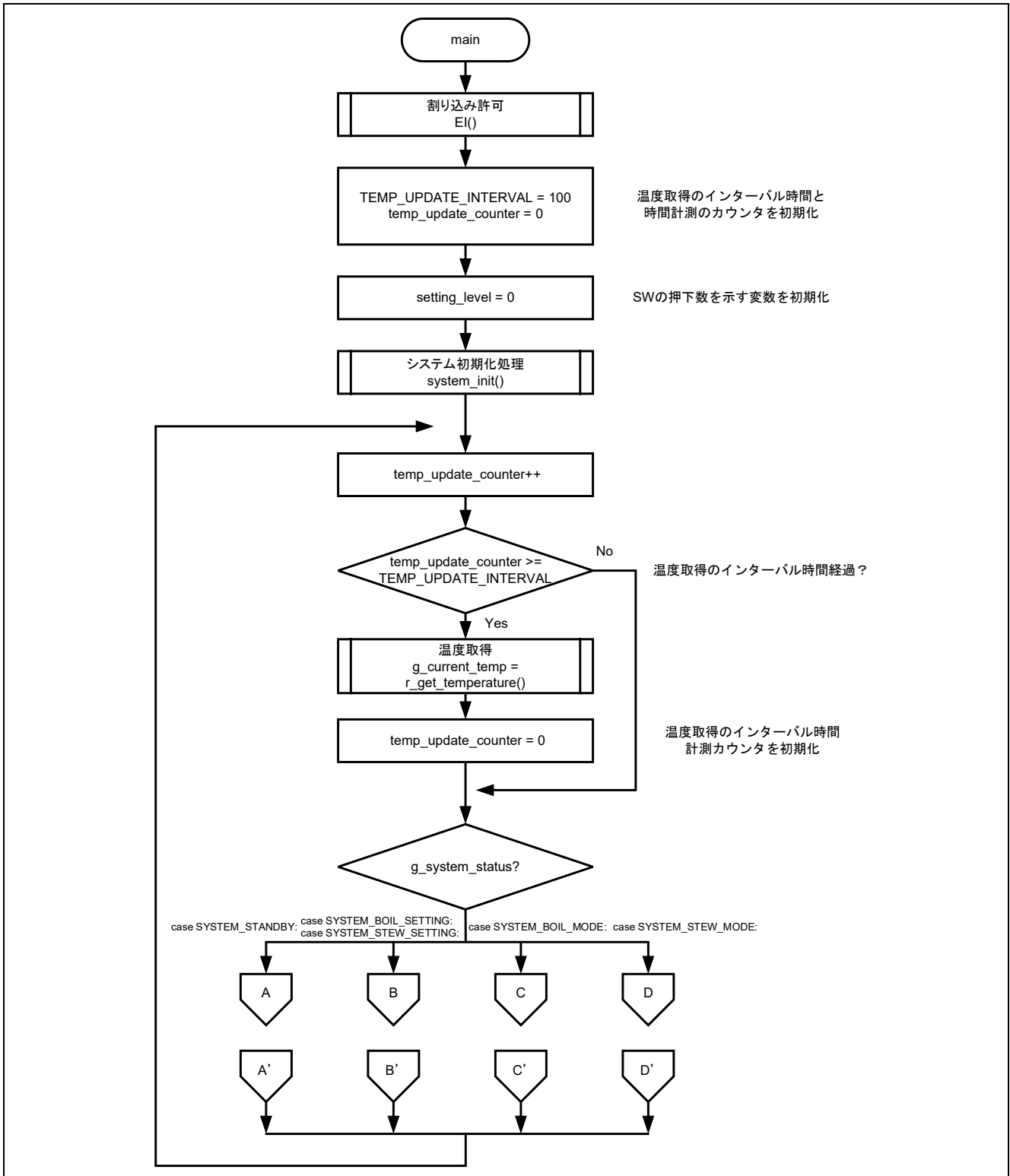
| | |
|-------|--|
| 概要 | TAU0_2 の割り込み処理 |
| ヘッダ | Config_TAU0_2.h |
| 宣言 | static void __near r_Config_TAU0_2_interrupt(void) |
| 説明 | ボタン長押しのカウントを行います |
| 引数 | なし |
| リターン値 | なし |
| 備考 | なし |

4.9 フローチャート

4.9.1 main 関数のフローチャート

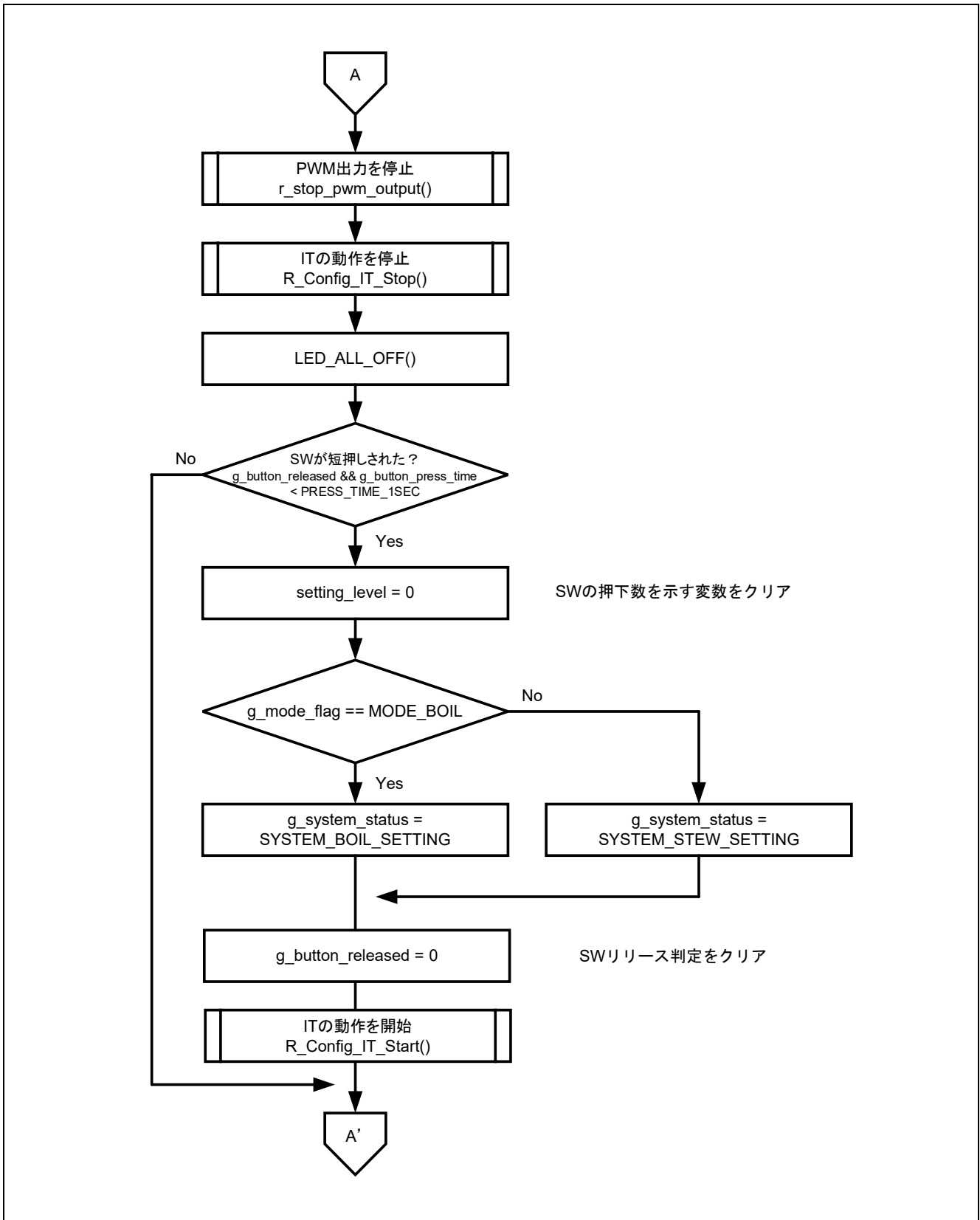
main 関数のフローチャートを以下に示します。

図 4-1 main 関数フローチャート (1/6)



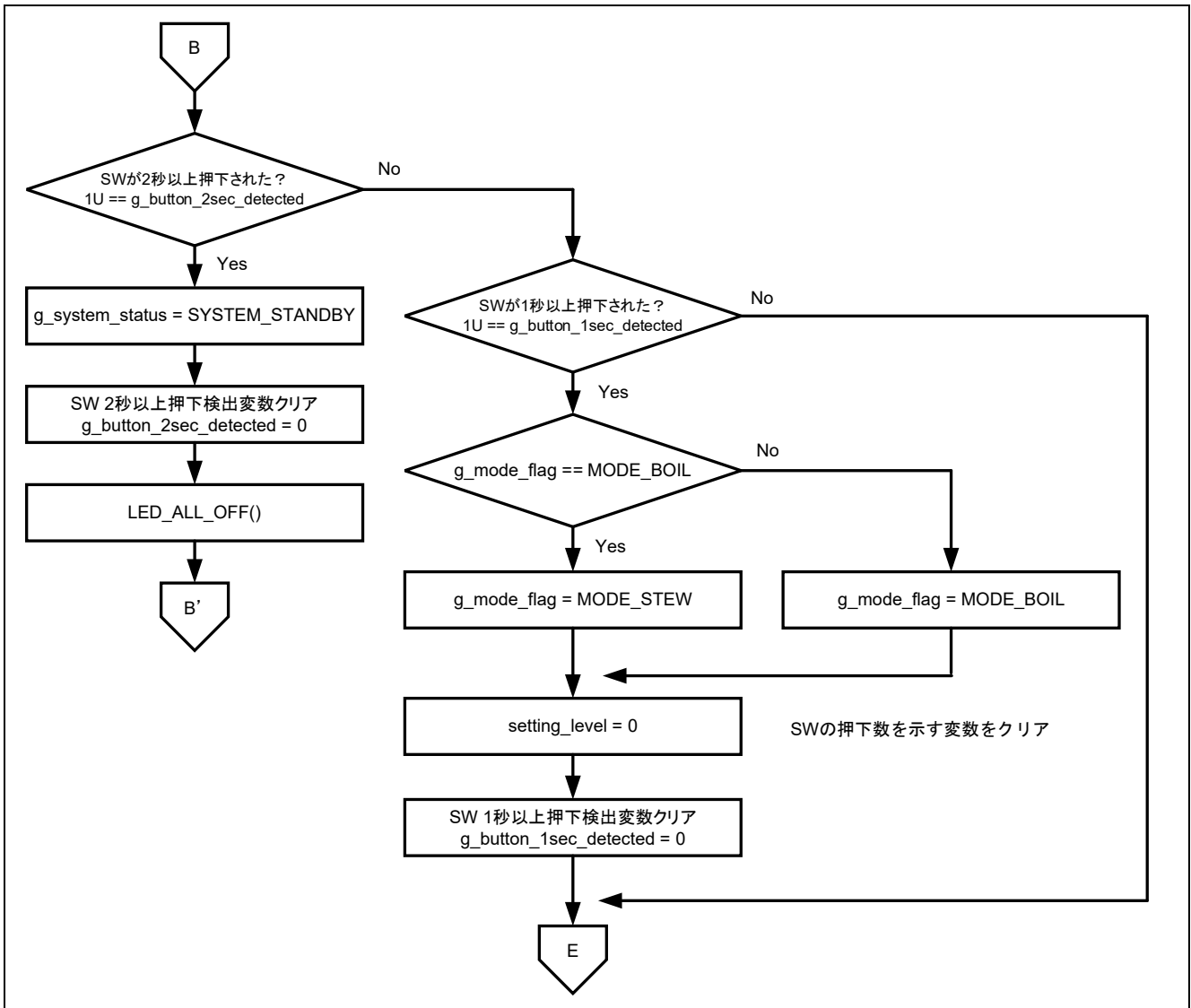
スタンバイモードの処理を以下に示します。

図 4-2 main 関数フローチャート (2/6)



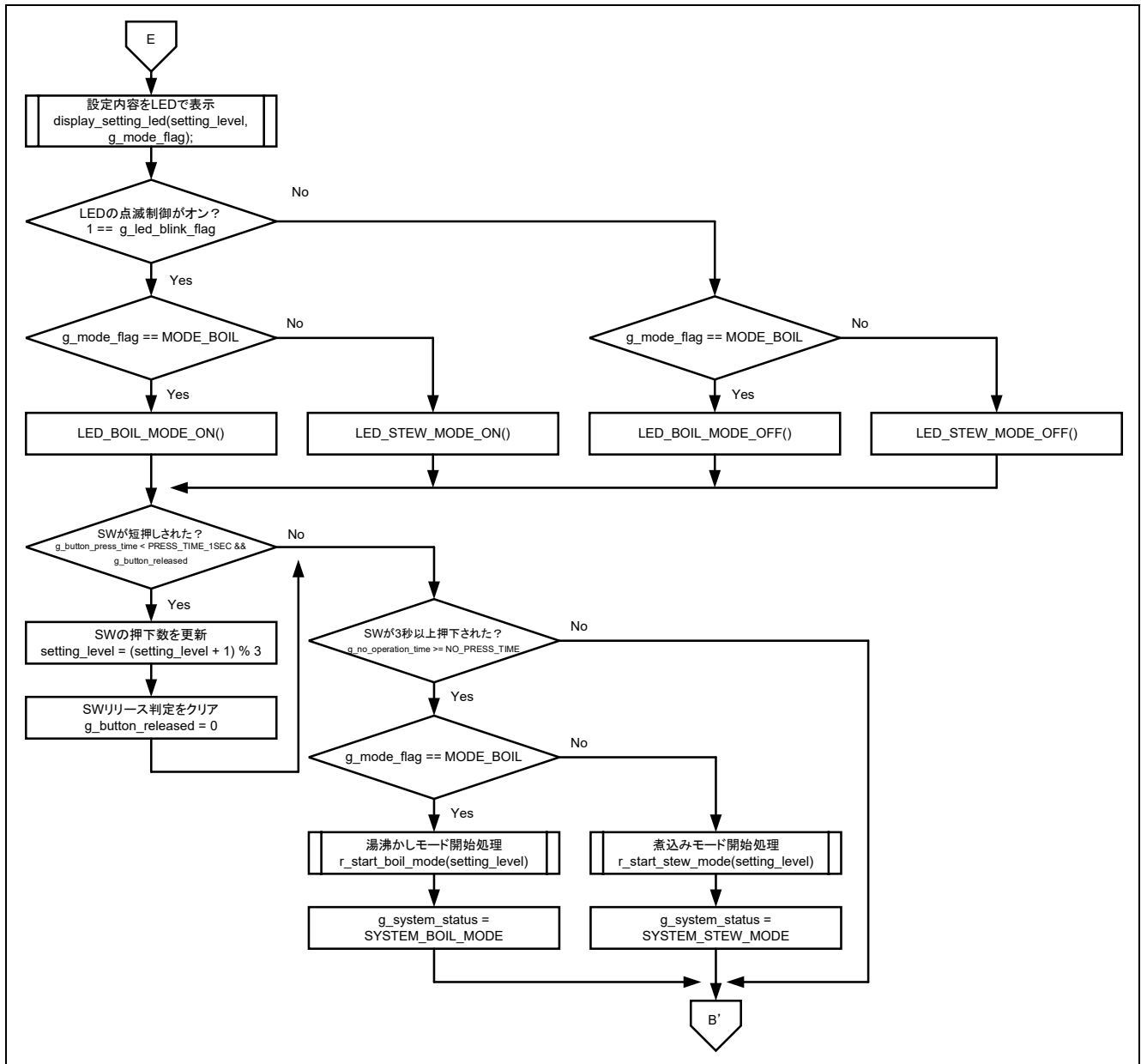
湯沸かし設定、煮込み設定の前半の処理を以下に示します。

図 4-3 main 関数フローチャート (3/6)



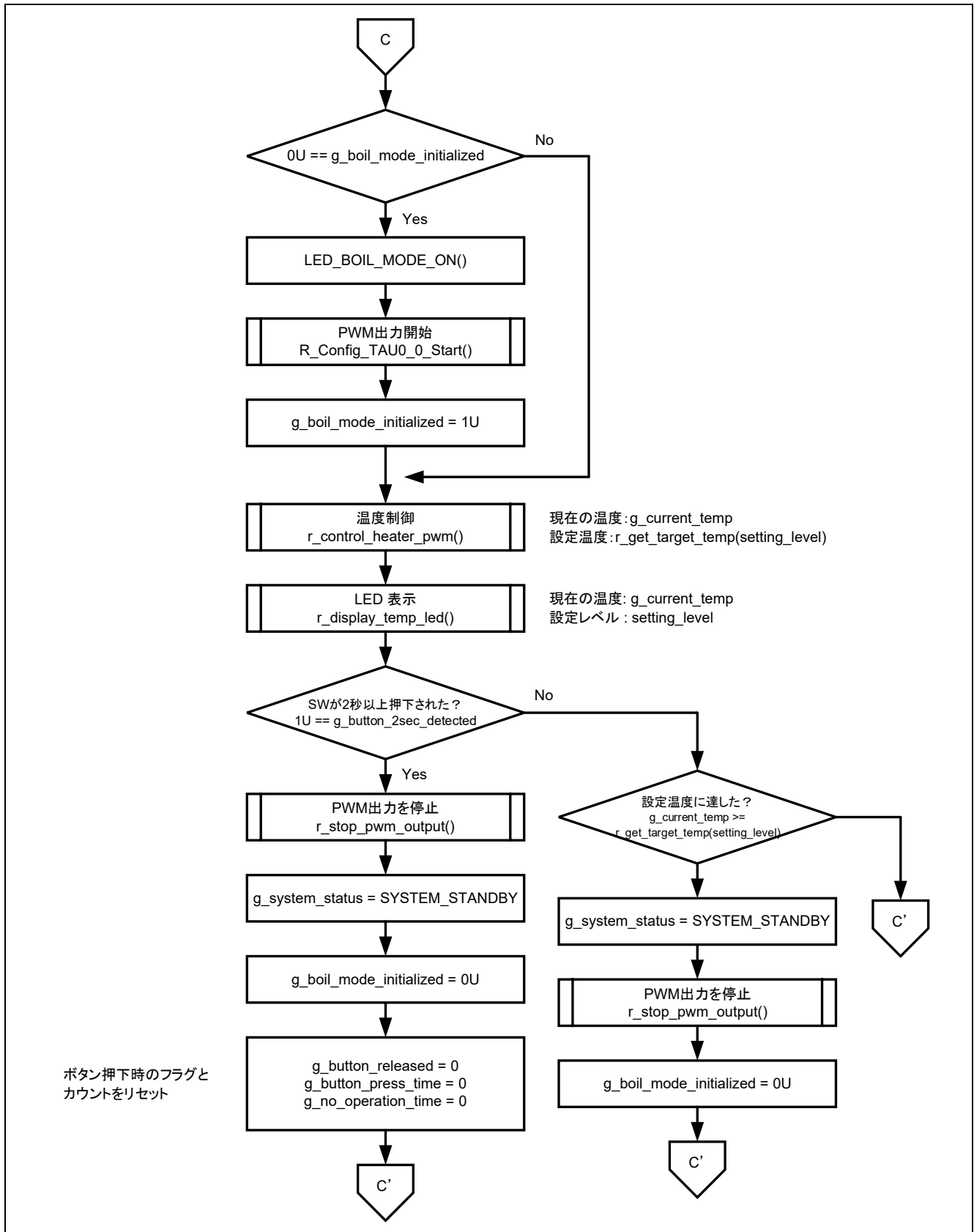
湯沸かし設定、煮込み設定の後半の処理を以下に示します。

図 4-4 main 関数フローチャート (4/6)



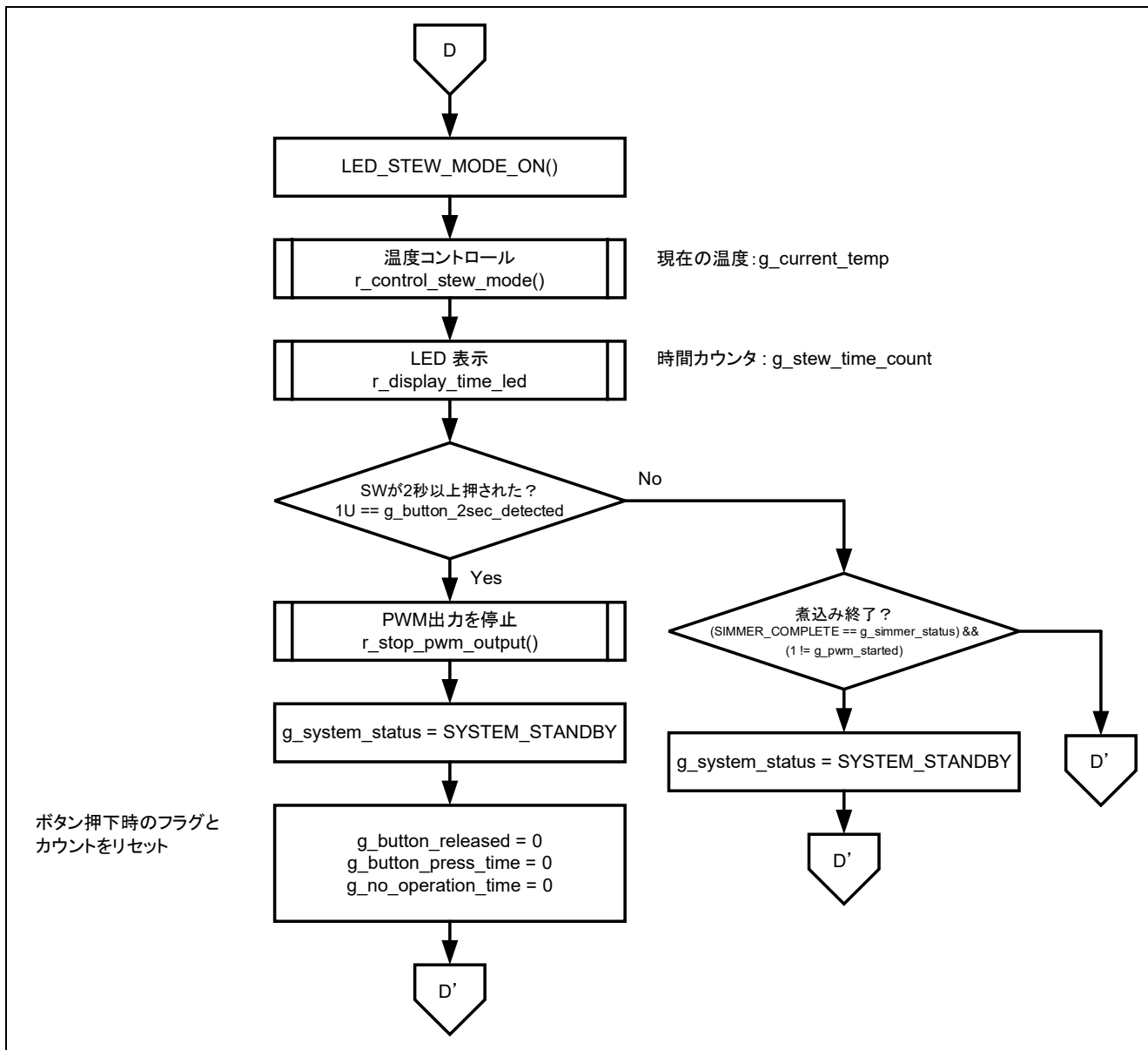
湯沸かしモード処理を以下に示します。

図 4-5 main 関数フローチャート (5/6)



煮込みモードの処理を以下に示します。

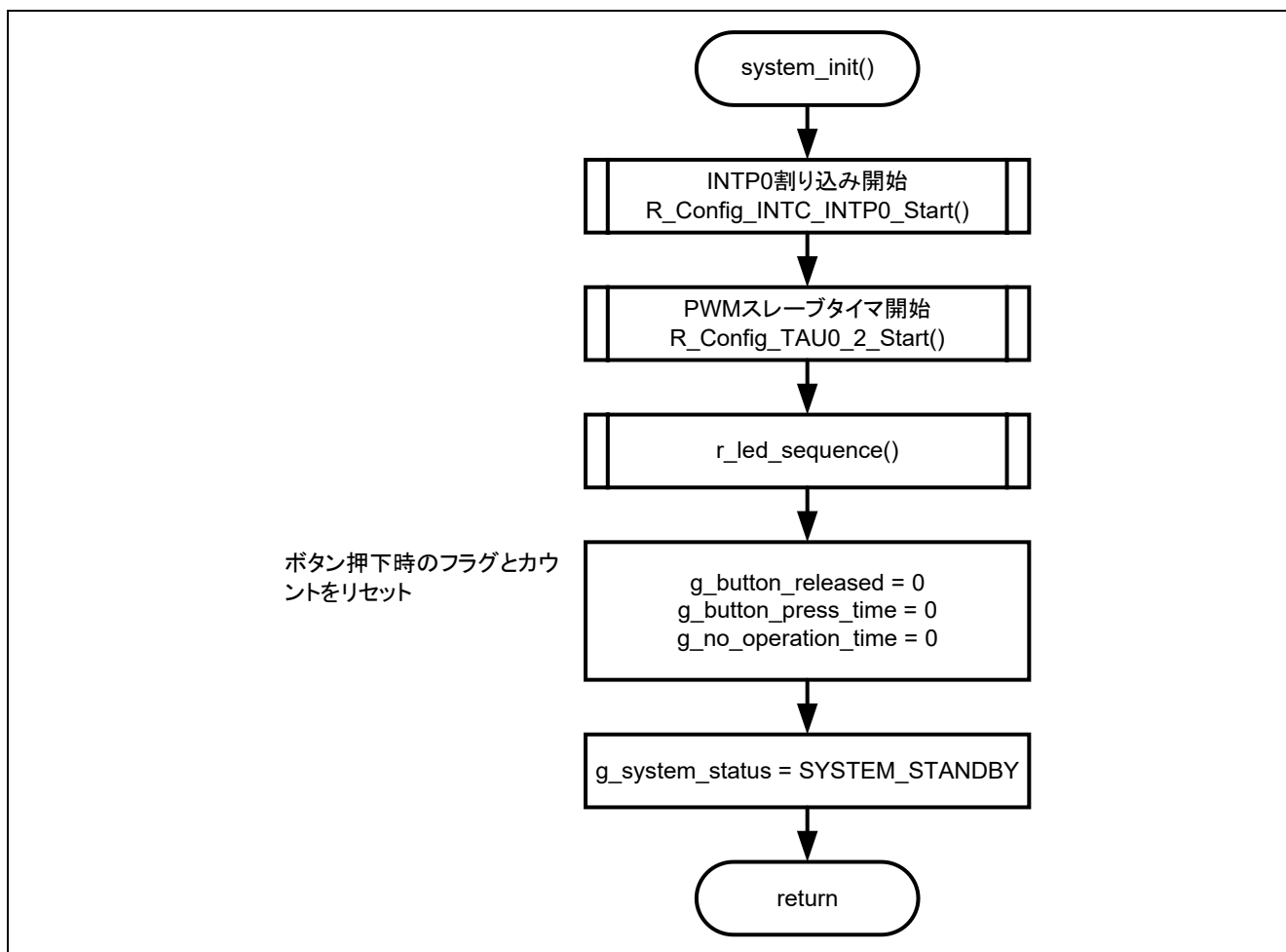
図 4-6 main 関数フローチャート (6/6)



4.9.2 system_init 関数のフローチャート

system_init 関数のフローチャートを以下に示します。

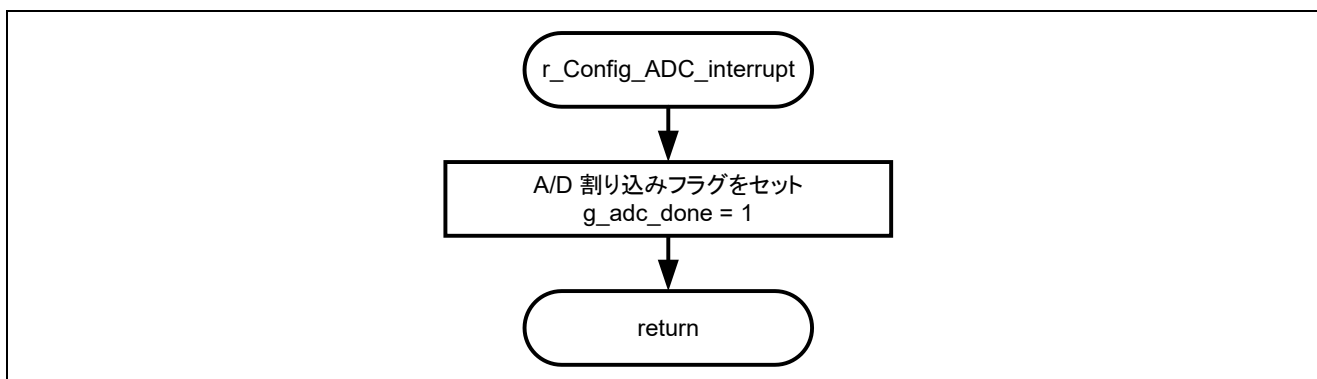
図 4-7 system_init 関数フローチャート



4.9.3 r_Config_ADC_interrupt 関数のフローチャート

r_Config_ADC_interrupt 関数のフローチャートを以下に示します。

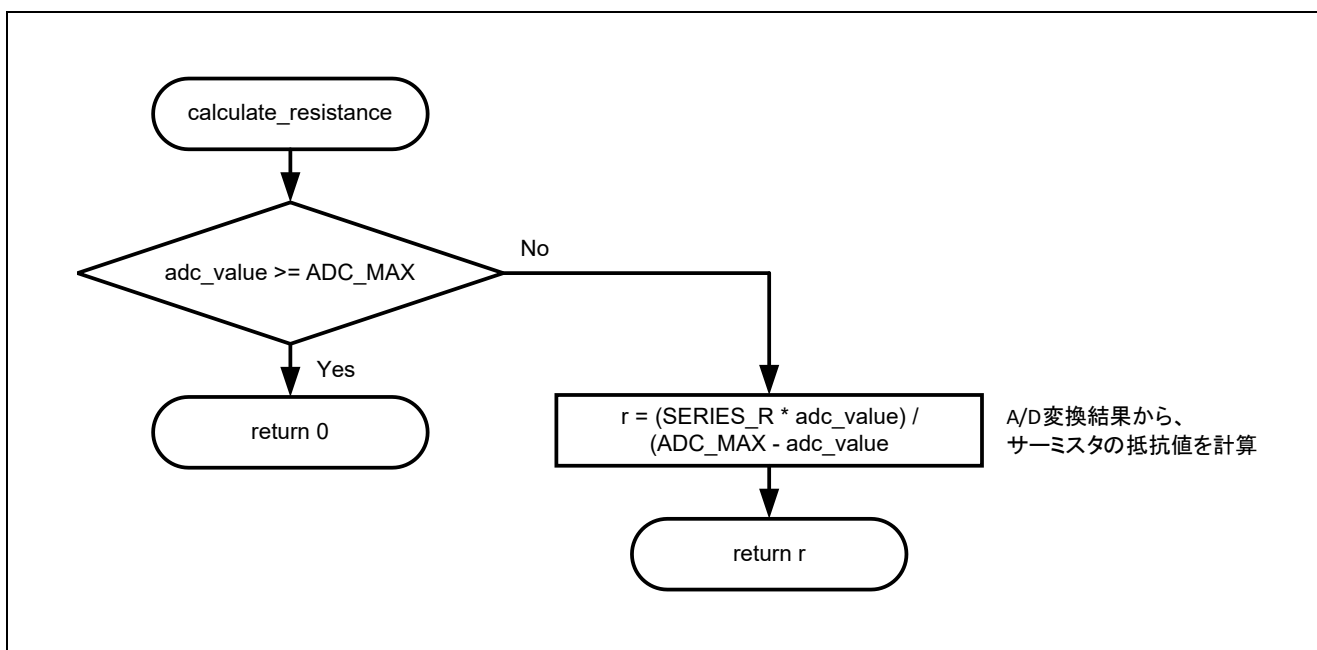
図 4-8 r_Config_ADC_interrupt 関数フローチャート



4.9.4 calculate_resistance 関数のフローチャート

calculate_resistance 関数のフローチャートを以下に示します。

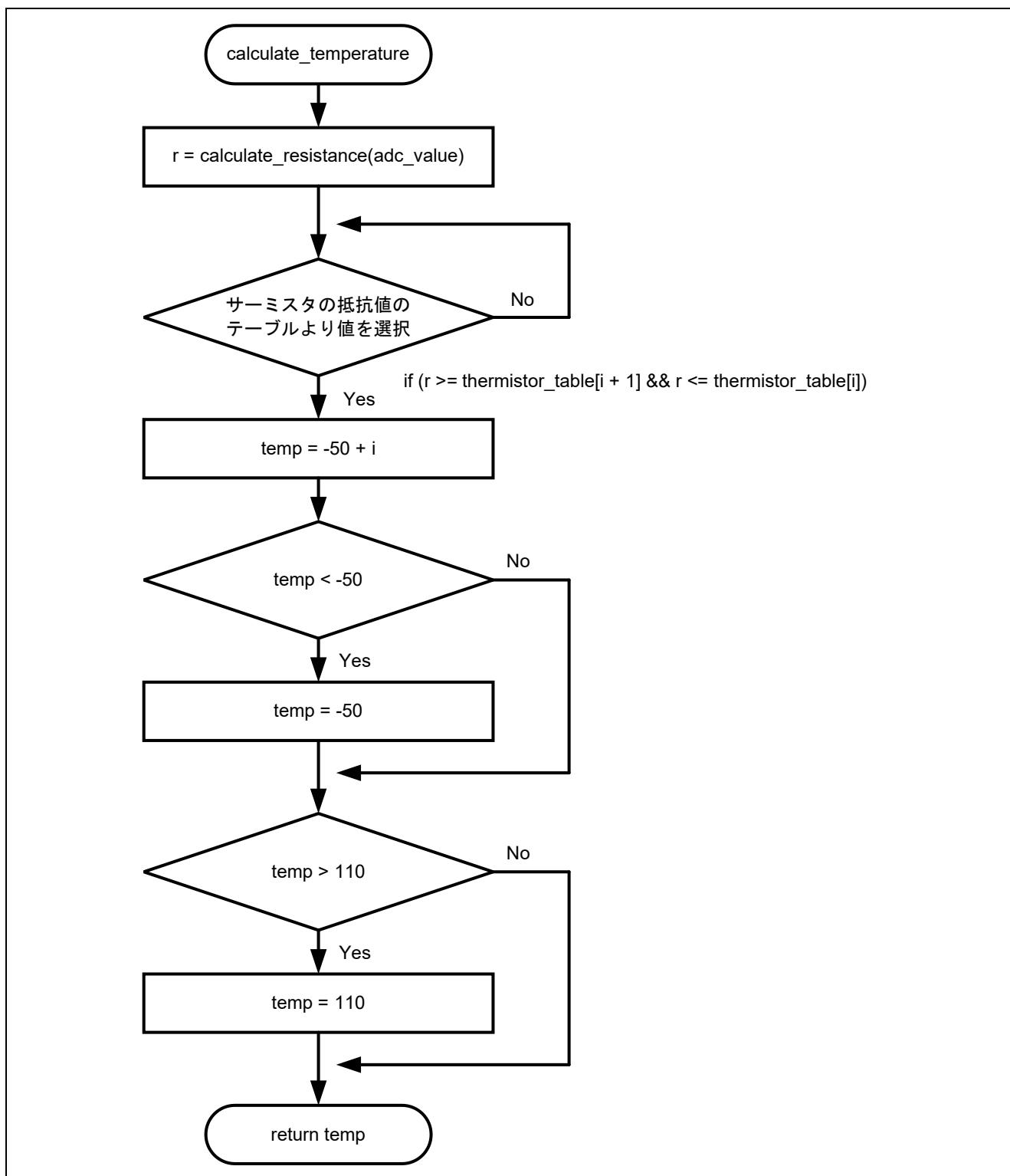
図 4-9 calculate_resistance 関数フローチャート



4.9.5 calculate_temperature 関数のフローチャート

calculate_temperature 関数のフローチャートを以下に示します。

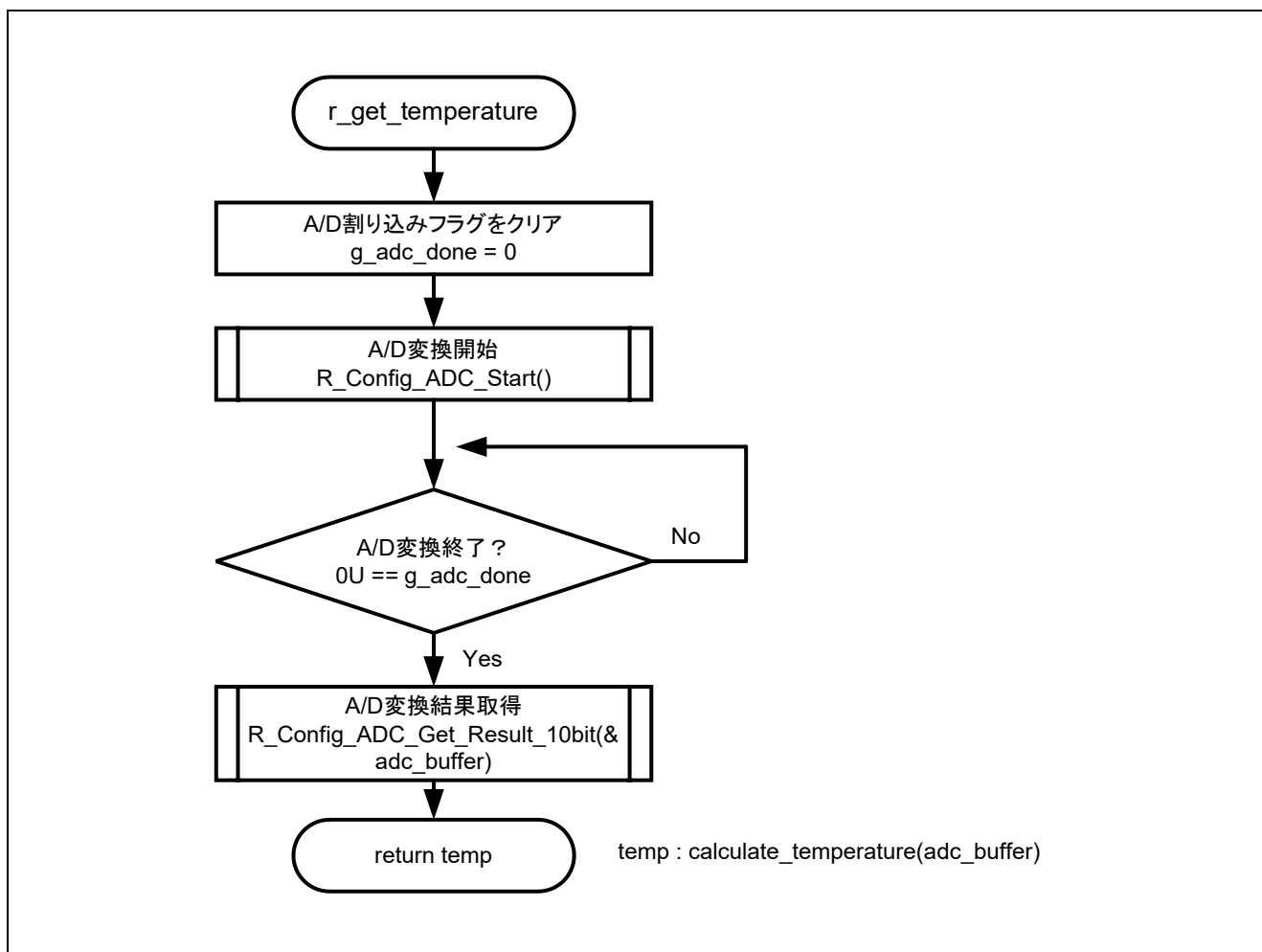
図 4-10 calculate_temperature 関数フローチャート



4.9.6 r_get_temperature 関数のフローチャート

r_get_temperature 関数のフローチャートを以下に示します。

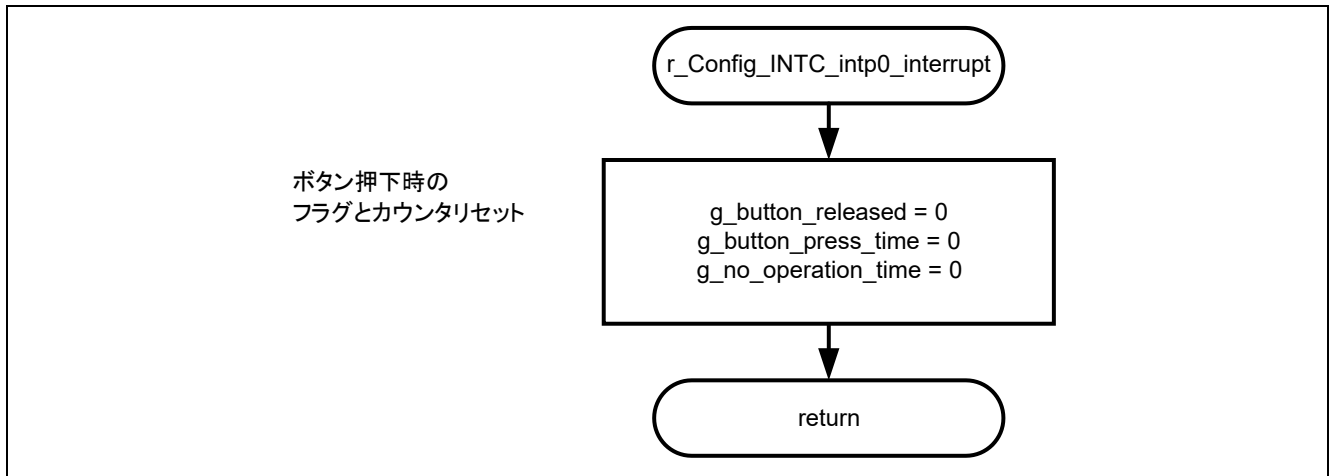
図 4-11 r_get_temperature 関数フローチャート



4.9.7 r_Config_INTC_intp0_interrupt 関数のフローチャート

r_Config_INTC_intp0_interrupt 関数のフローチャートを以下に示します。

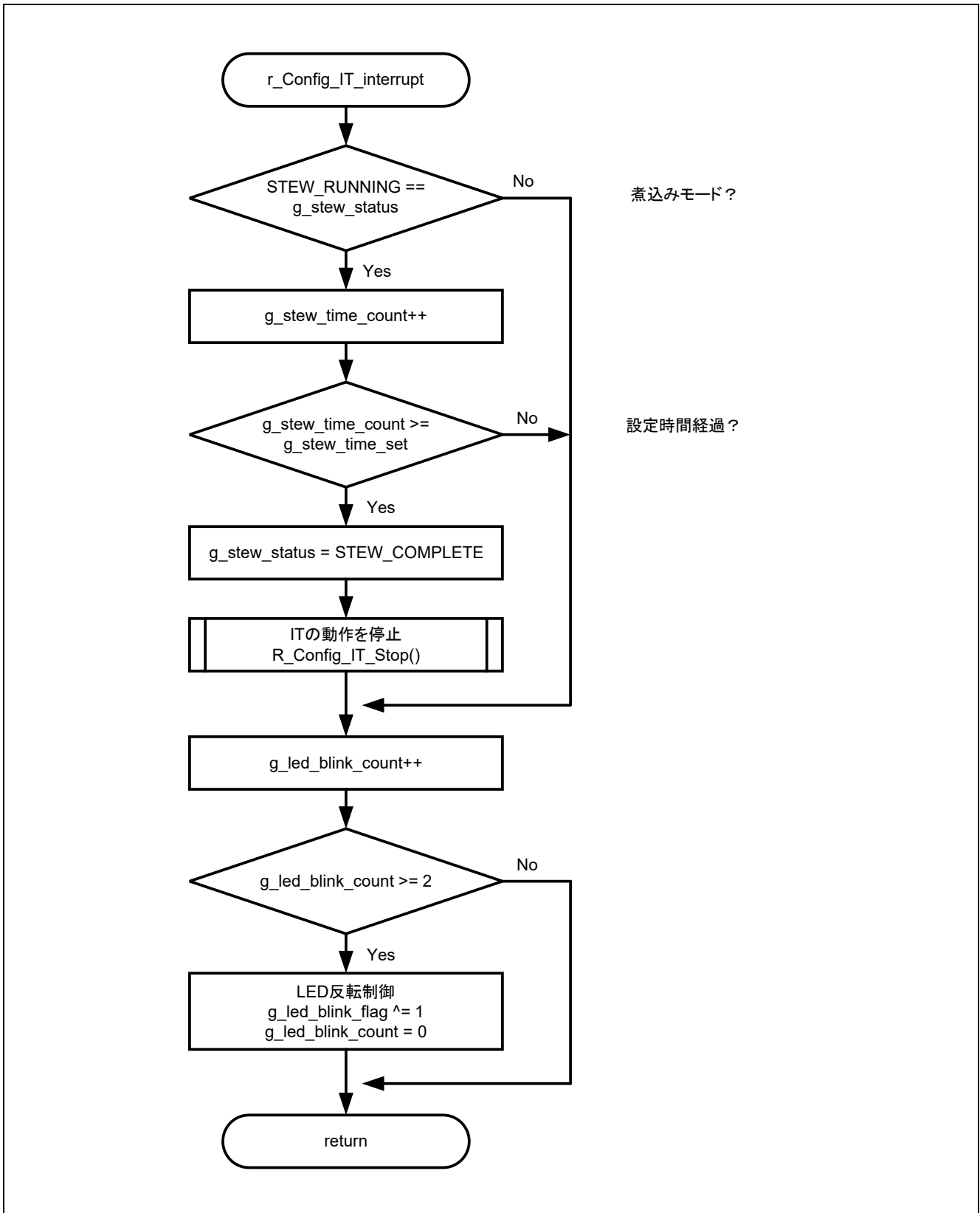
図 4-12 r_Config_INTC_intp0_interrupt 関数フローチャート



4.9.8 r_Config_IT_interrupt 関数のフローチャート

r_Config_IT_interrupt 関数のフローチャートを以下に示します。

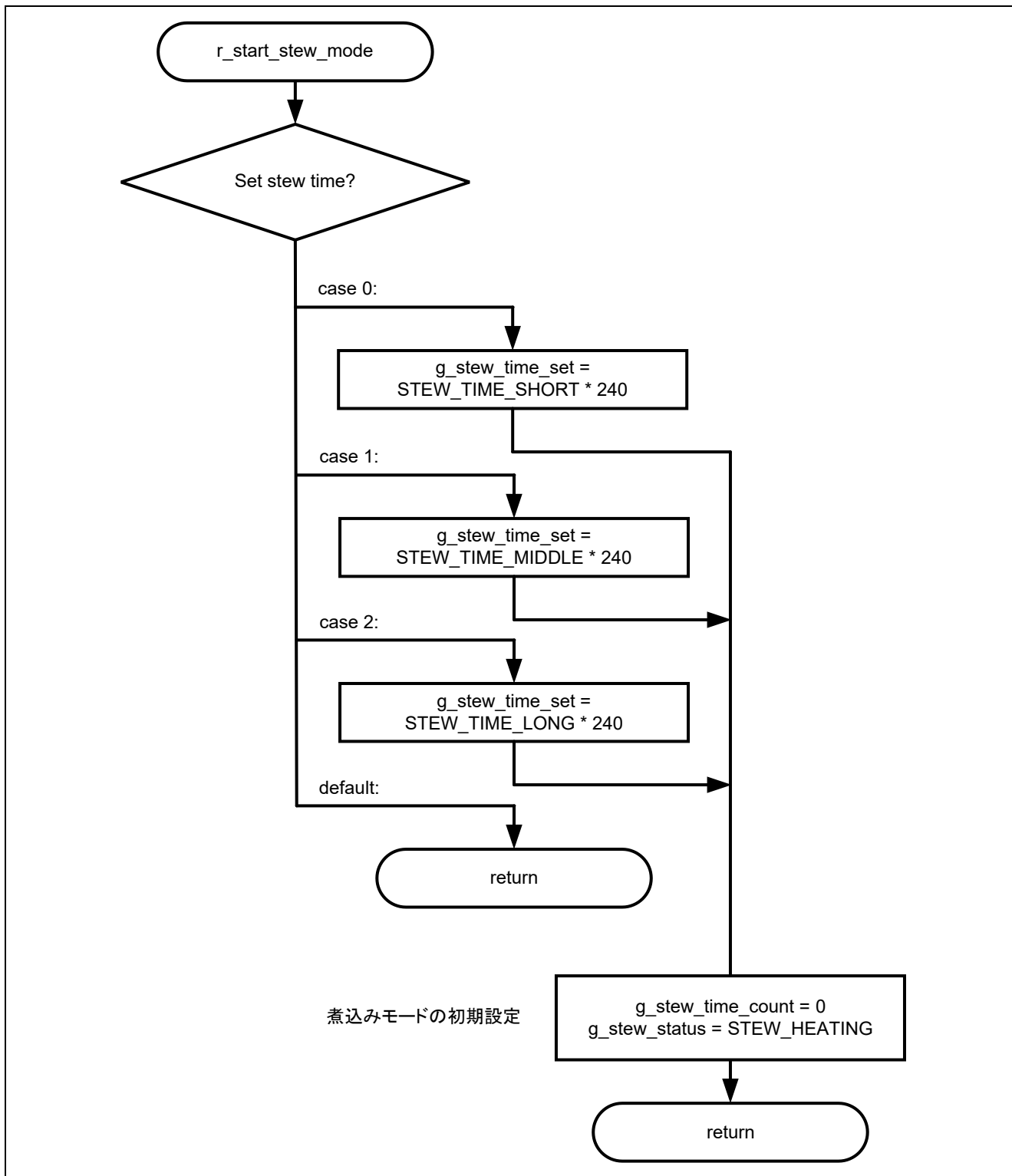
図 4-13 r_Config_IT_interrupt 関数フローチャート



4.9.9 r_start_stew_mode 関数のフローチャート

r_start_stew_mode 関数のフローチャートを以下に示します。

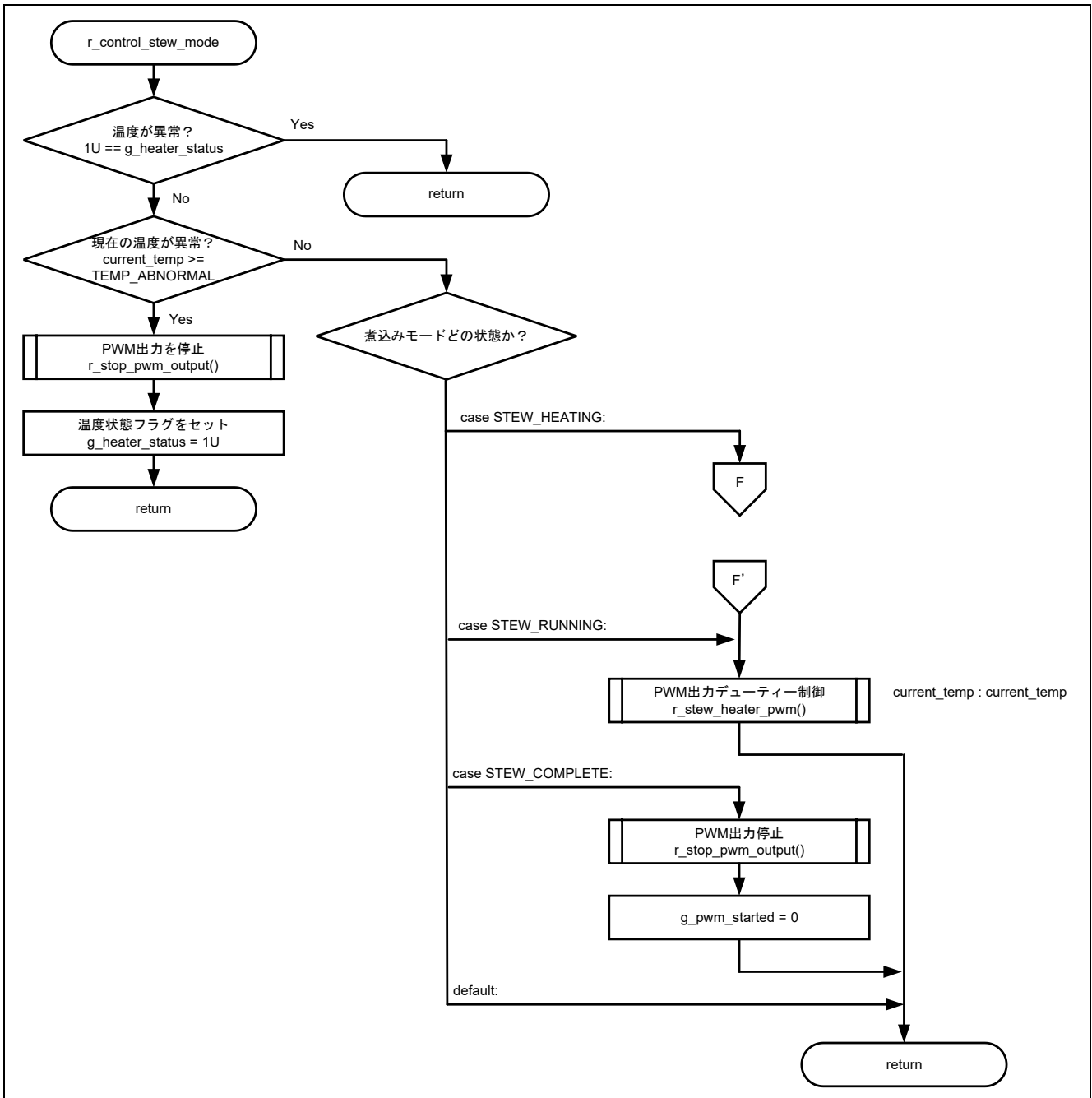
図 4-14 r_start_stew_mode 関数フローチャート



4.9.10 r_control_stew_mode 関数のフローチャート

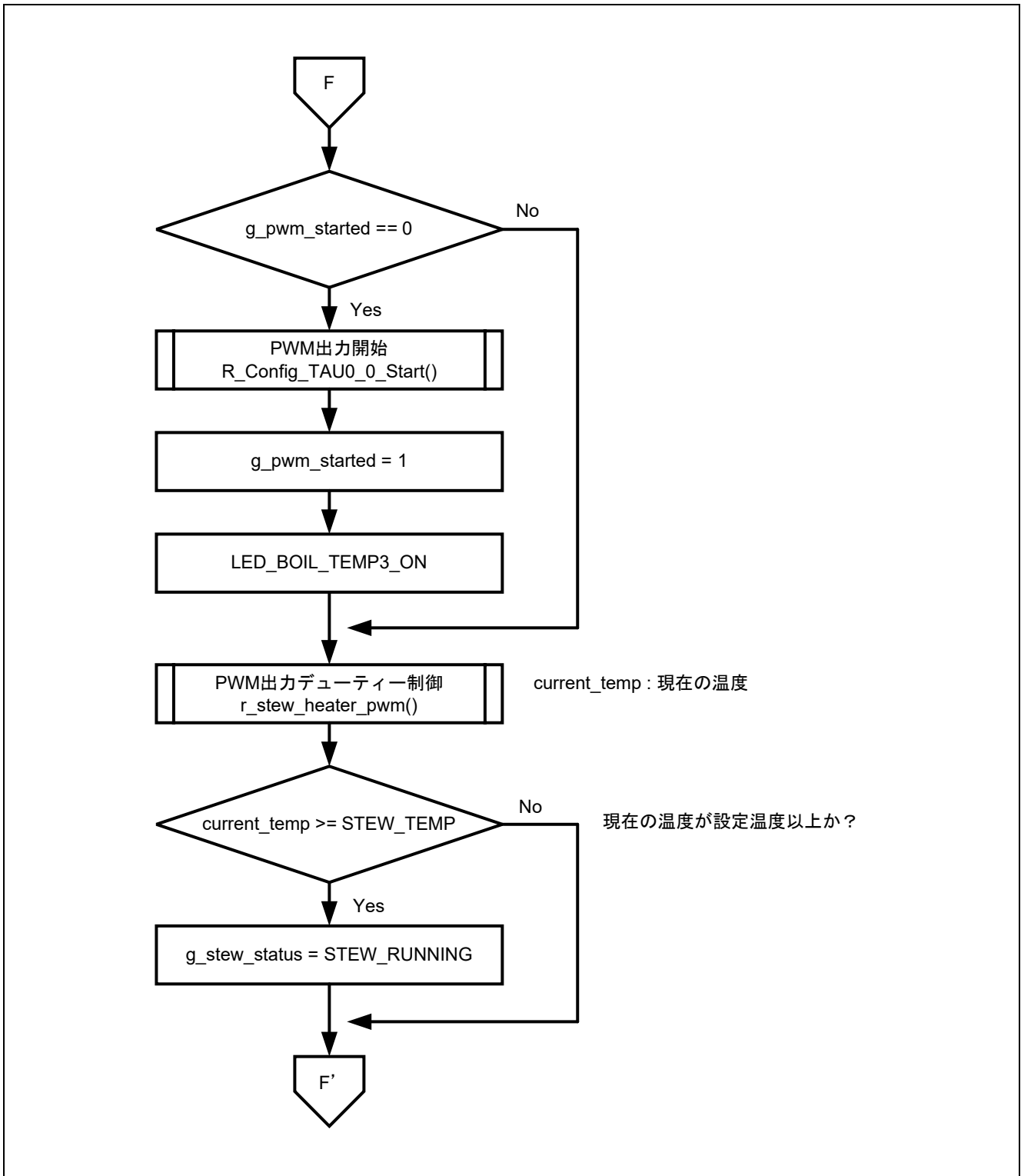
r_control_stew_mode 関数のフローチャートを以下に示します。

図 4-15 r_control_stew_mode 関数フローチャート (1/2)



煮込みモードの状態が加熱時の処理を以下に示します。

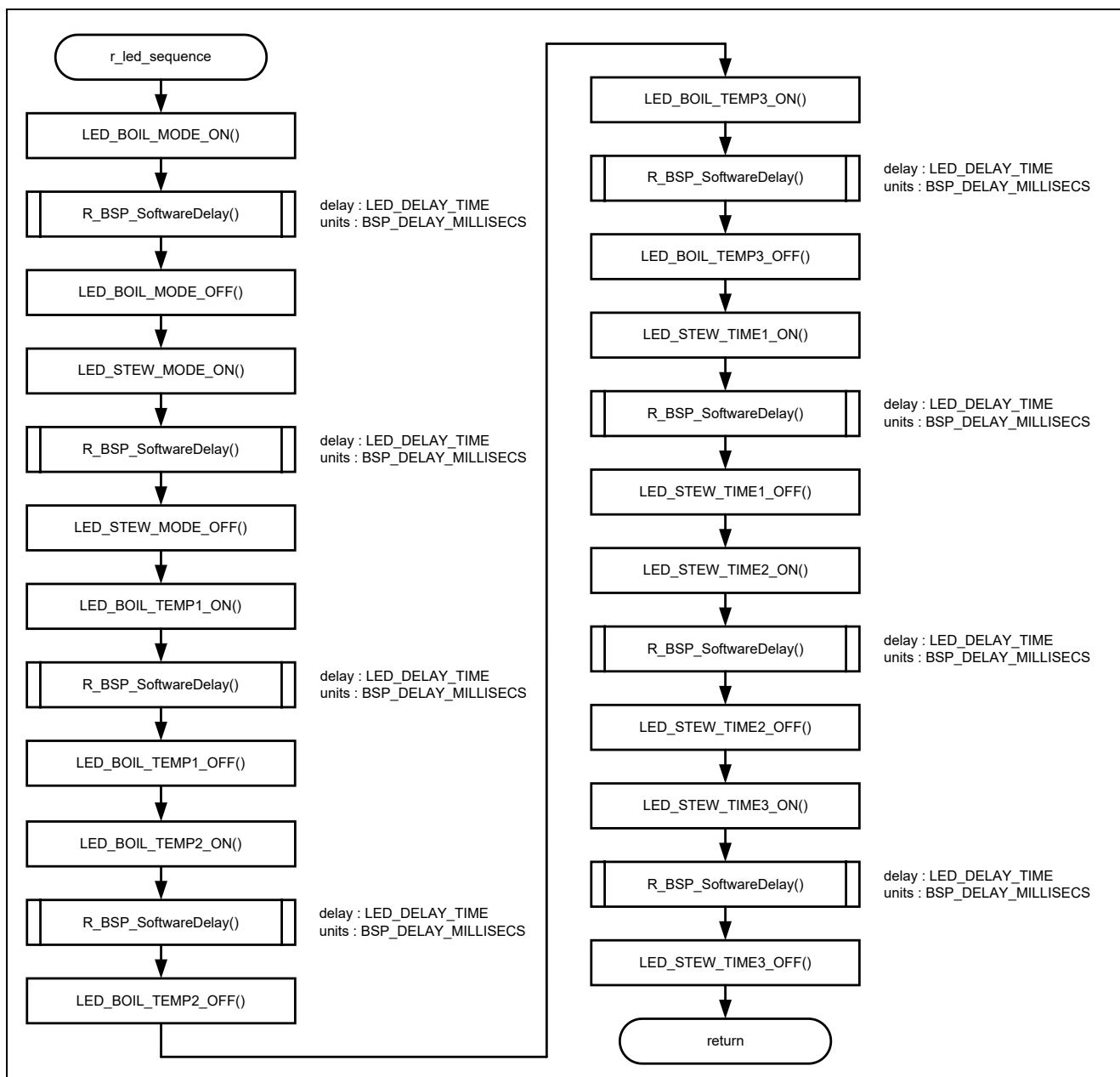
図 4-16 r_control_stew_mode 関数フローチャート (2/2)



4.9.11 r_led_sequence 関数のフローチャート

r_led_sequence 関数のフローチャートを以下に示します。

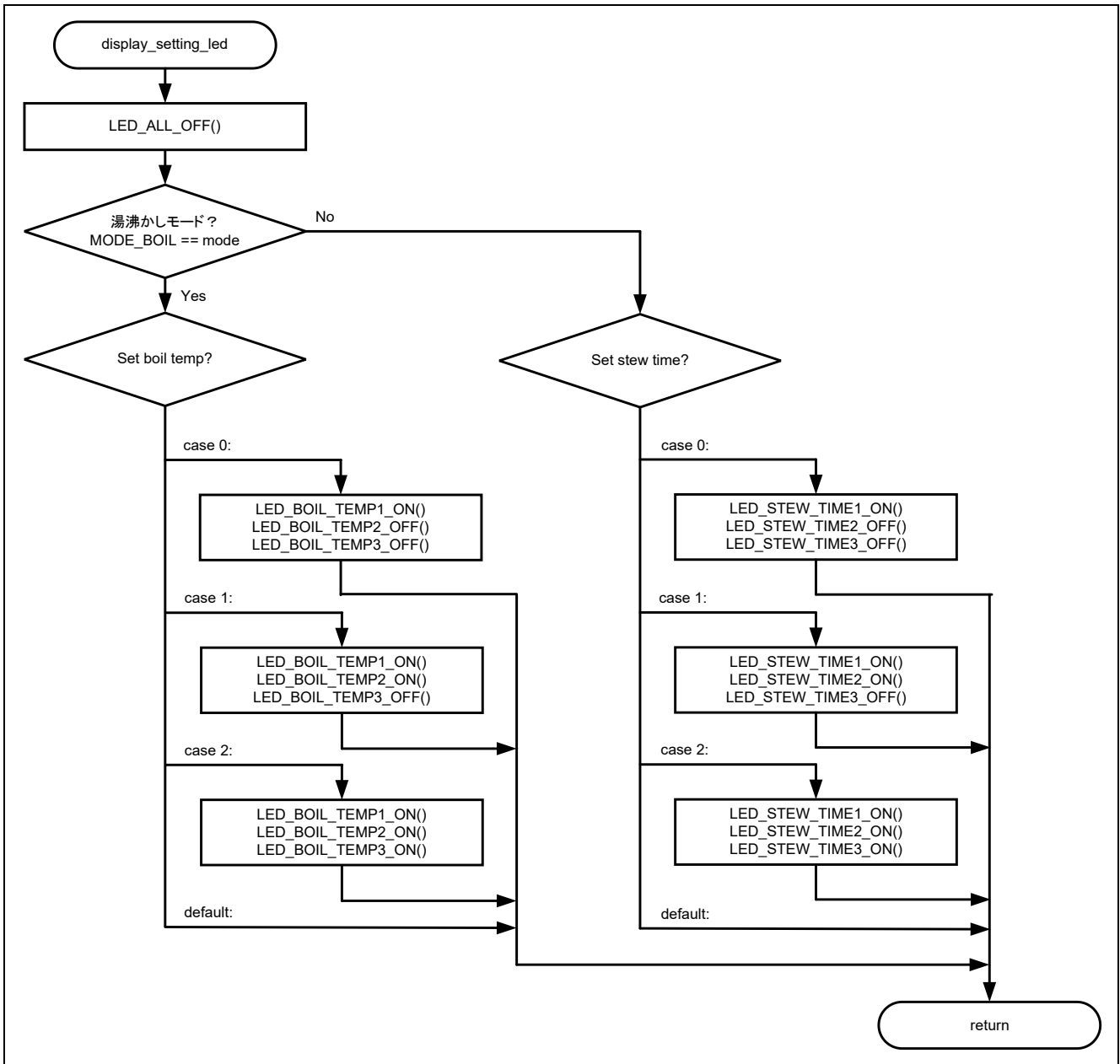
図 4-17 r_led_sequence 関数フローチャート



4.9.12 display_setting_led 関数のフローチャート

display_setting_led 関数のフローチャートを以下に示します。

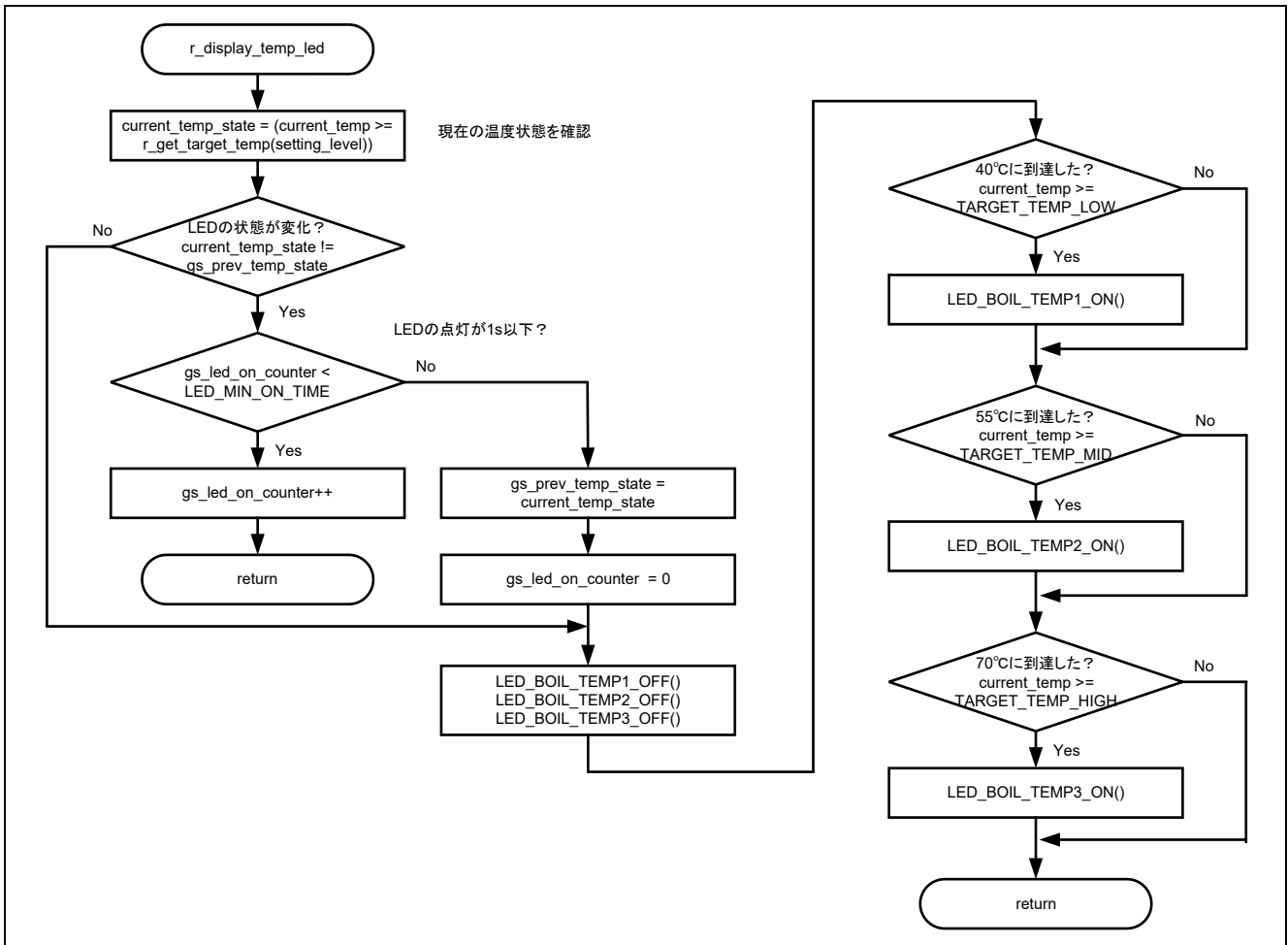
図 4-18 display_setting_led 関数フローチャート



4.9.13 r_display_temp_led 関数のフローチャート

r_display_temp_led 関数のフローチャートを以下に示します。

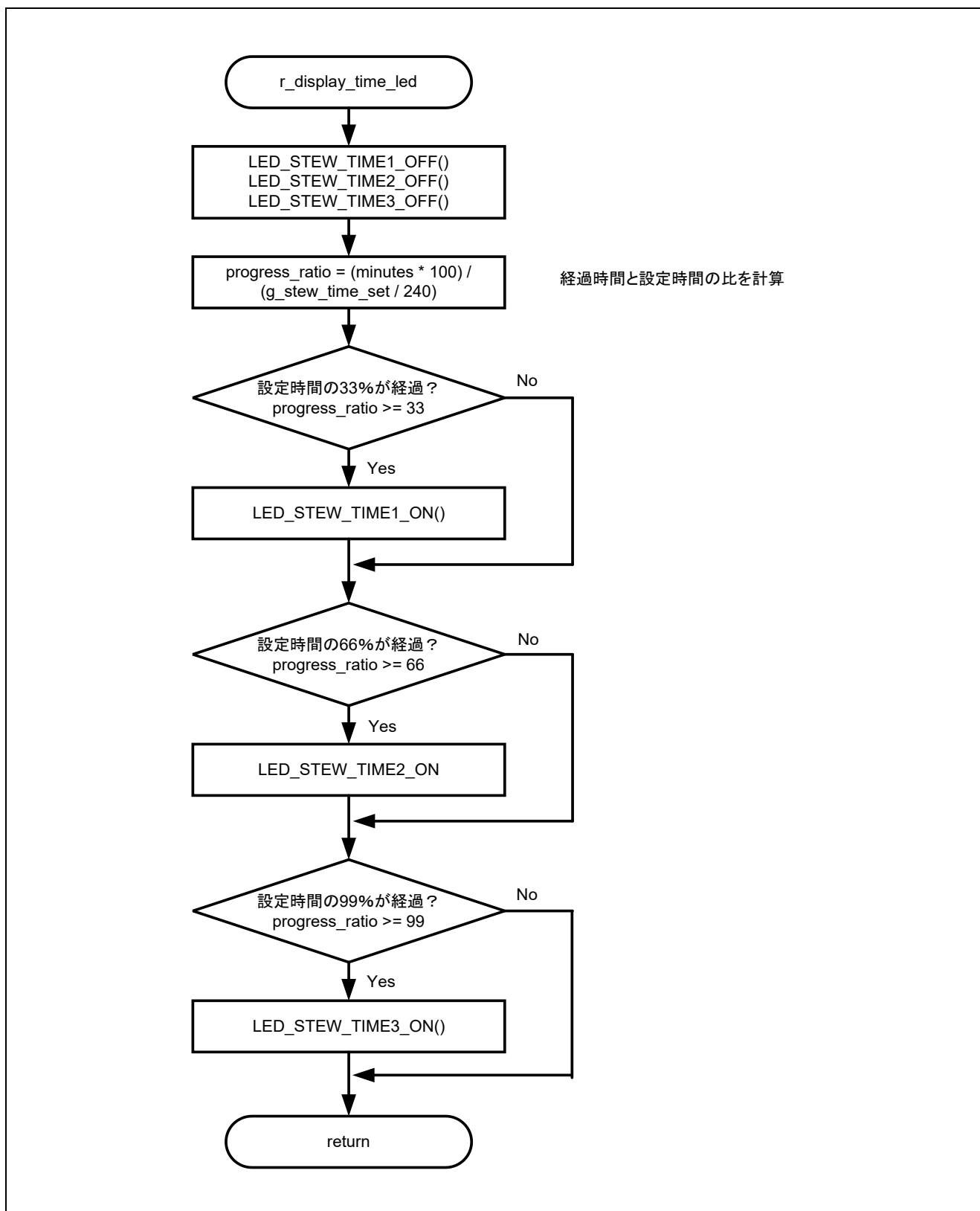
図 4-19 r_display_temp_led 関数フローチャート



4.9.14 r_display_time_led 関数のフローチャート

r_display_time_led 関数のフローチャートを以下に示します。

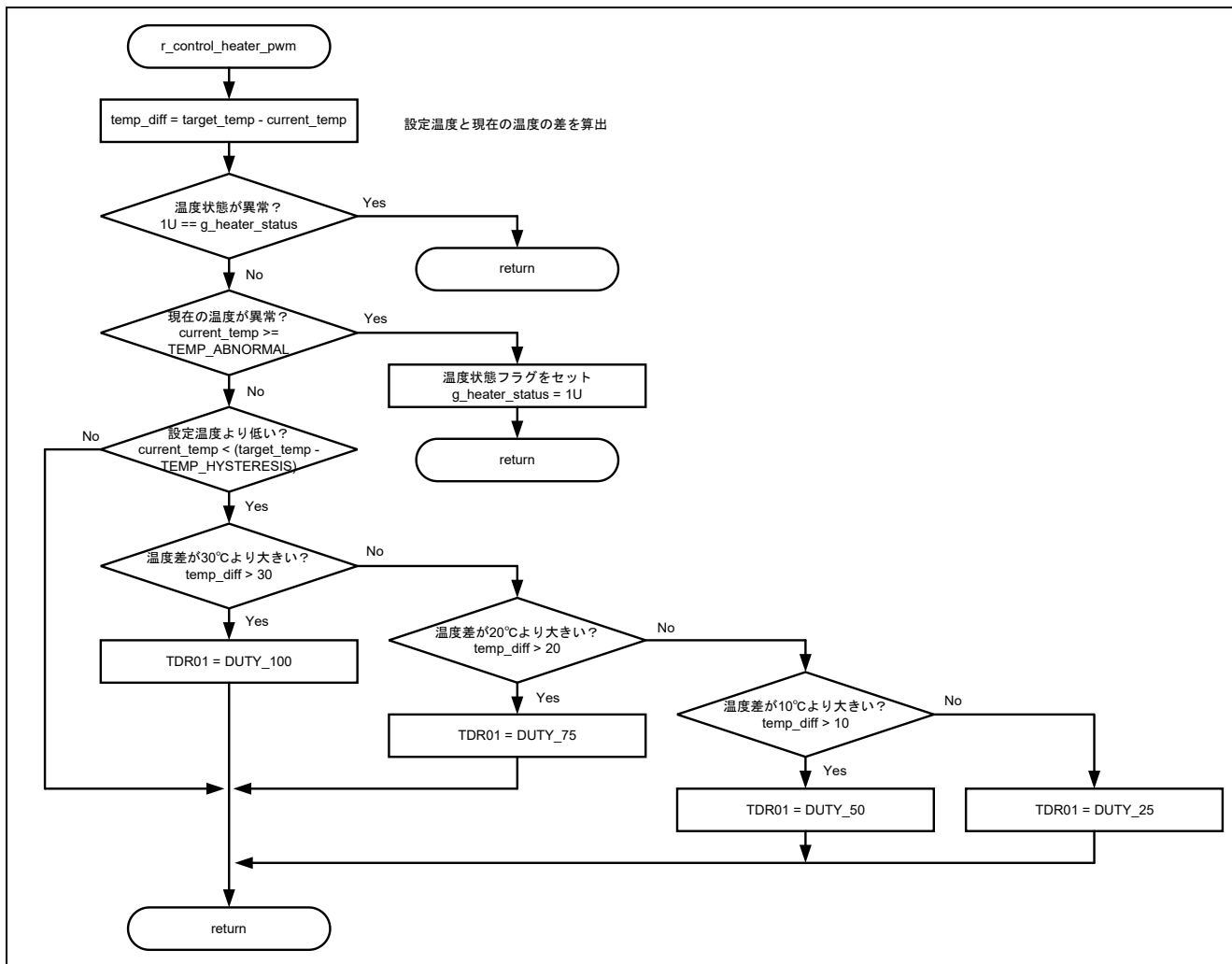
図 4-20 r_display_time_led 関数フローチャート



4.9.15 r_control_heater_pwm 関数のフローチャート

r_control_heater_pwm 関数のフローチャートを以下に示します。

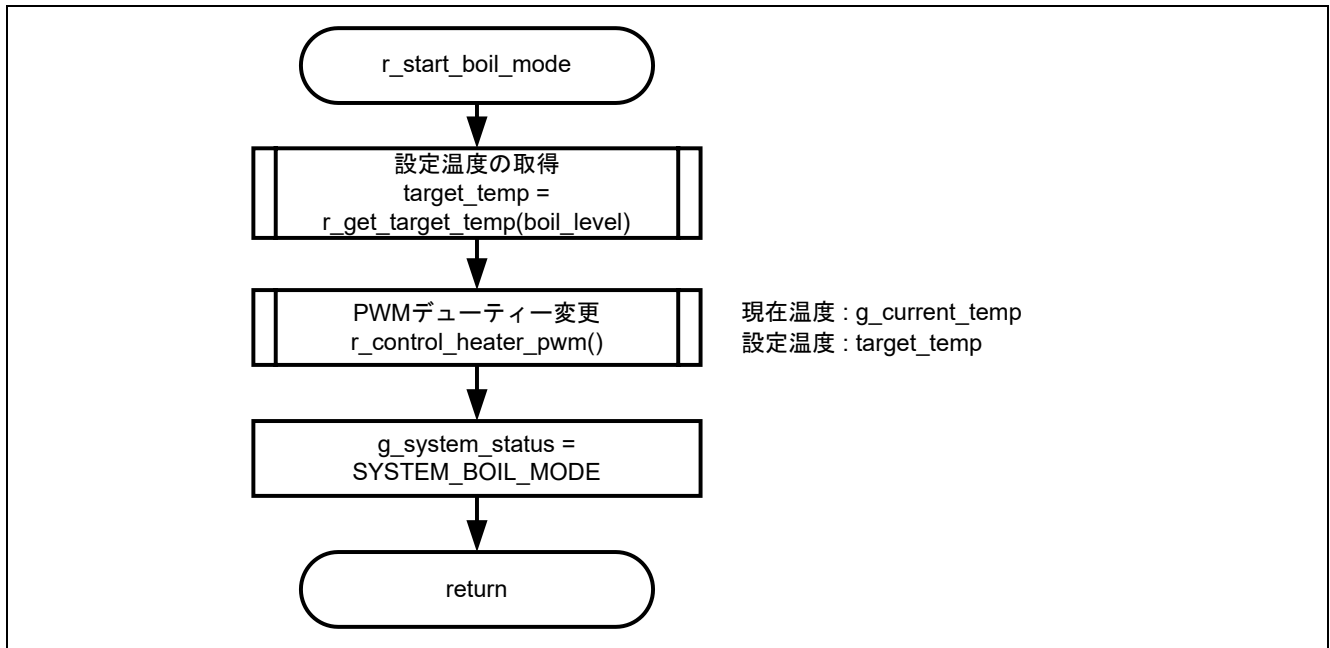
図 4-21 r_control_heater_pwm 関数フローチャート



4.9.16 r_start_boil_mode 関数のフローチャート

r_start_boil_mode 関数のフローチャートを以下に示します。

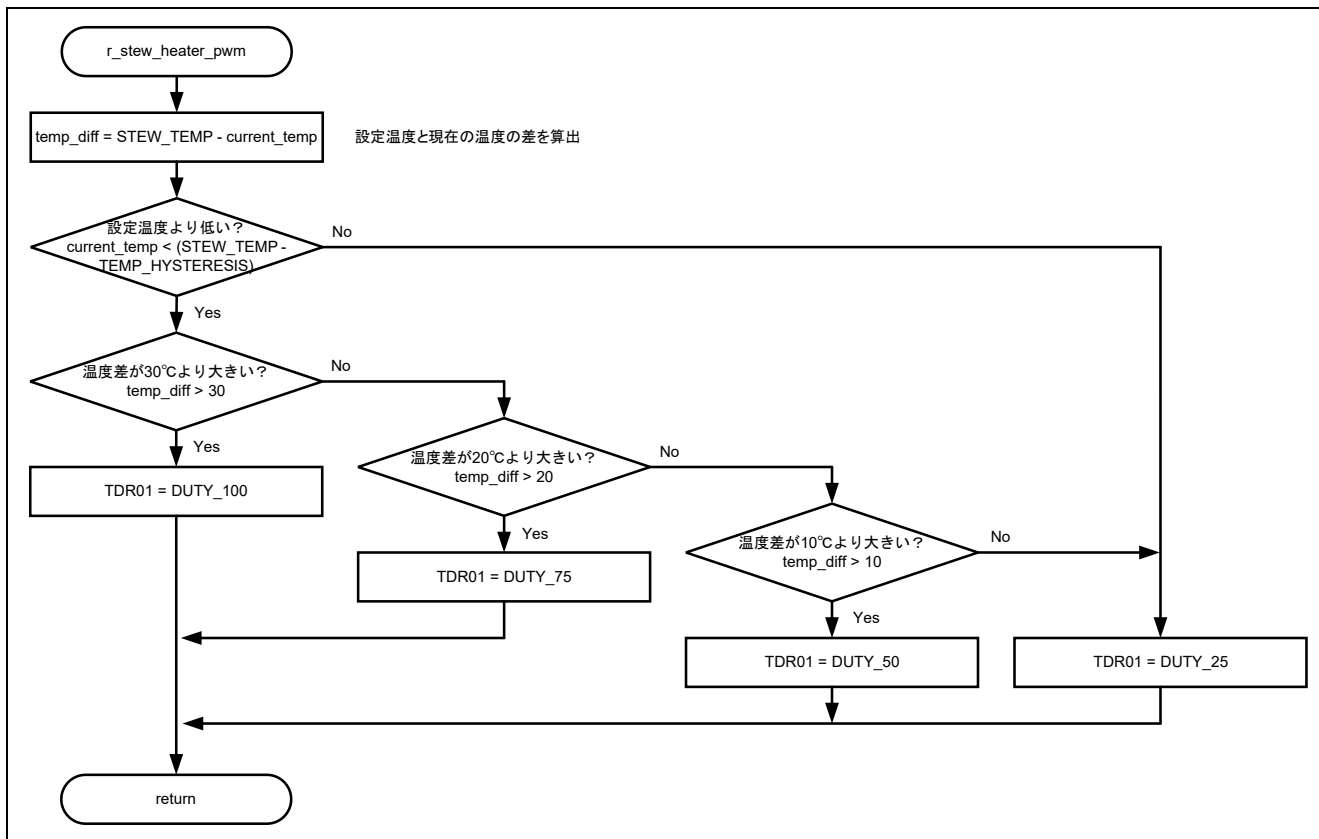
図 4-22 r_start_boil_mode 関数フローチャート



4.9.17 r_stew_heater_pwm 関数のフローチャート

r_stew_heater_pwm 関数のフローチャートを以下に示します。

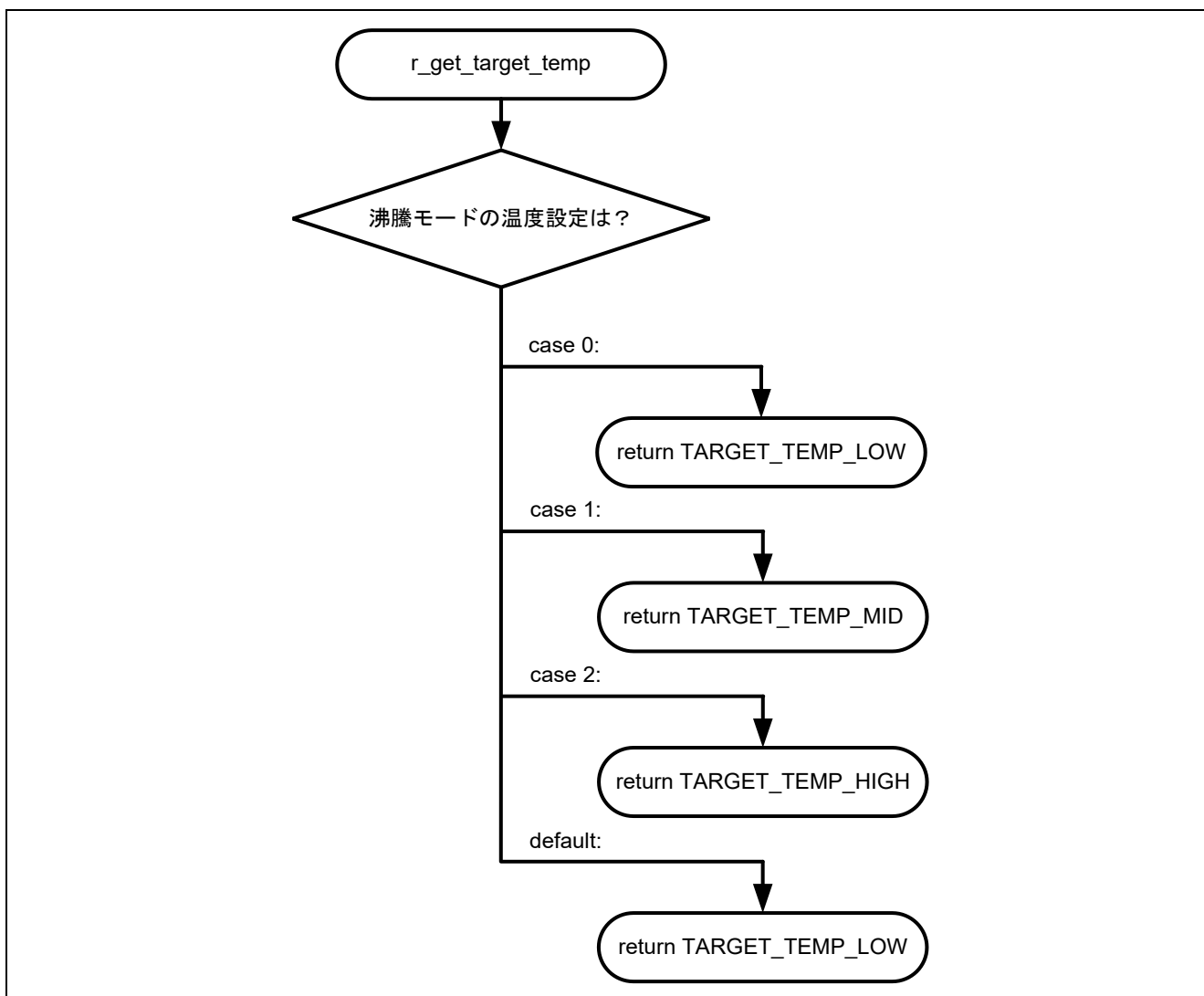
図 4-23 r_stew_heater_pwm 関数フローチャート



4.9.18 r_get_target_temp 関数のフローチャート

r_get_target_temp 関数のフローチャートを以下に示します。

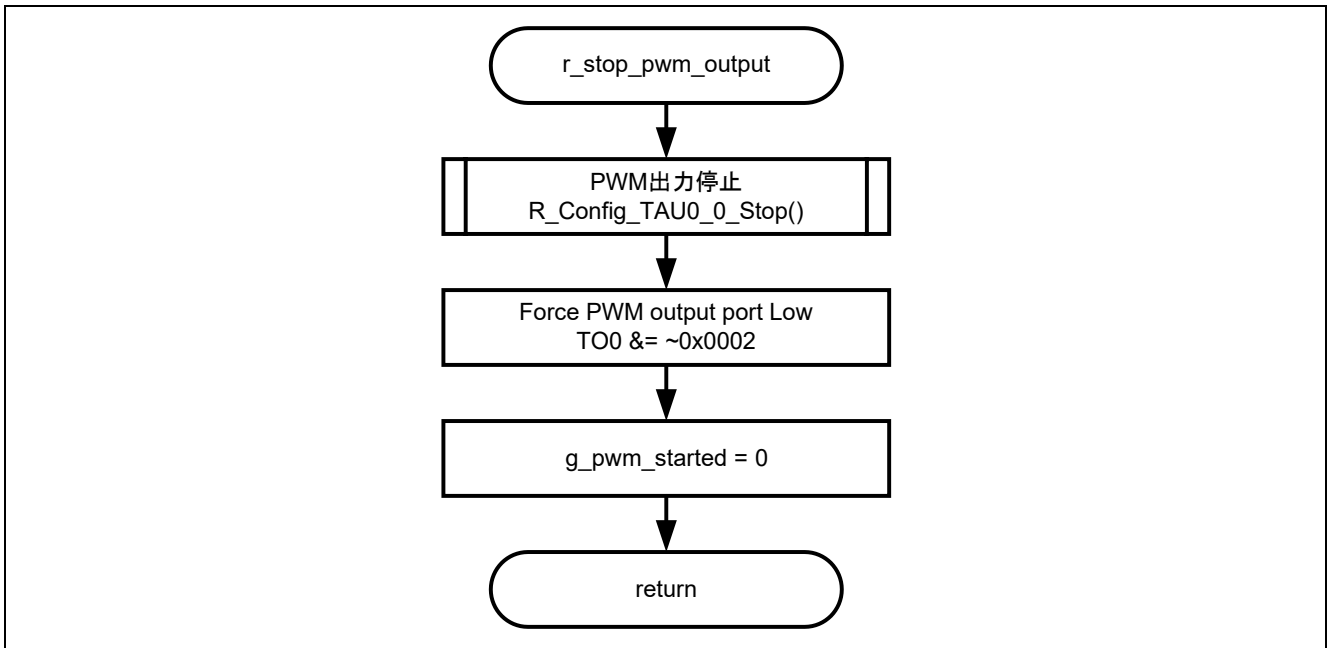
図 4-24 r_get_target_temp 関数フローチャート



4.9.19 r_stop_pwm_output 関数のフローチャート

r_stop_pwm_output 関数のフローチャートを以下に示します。

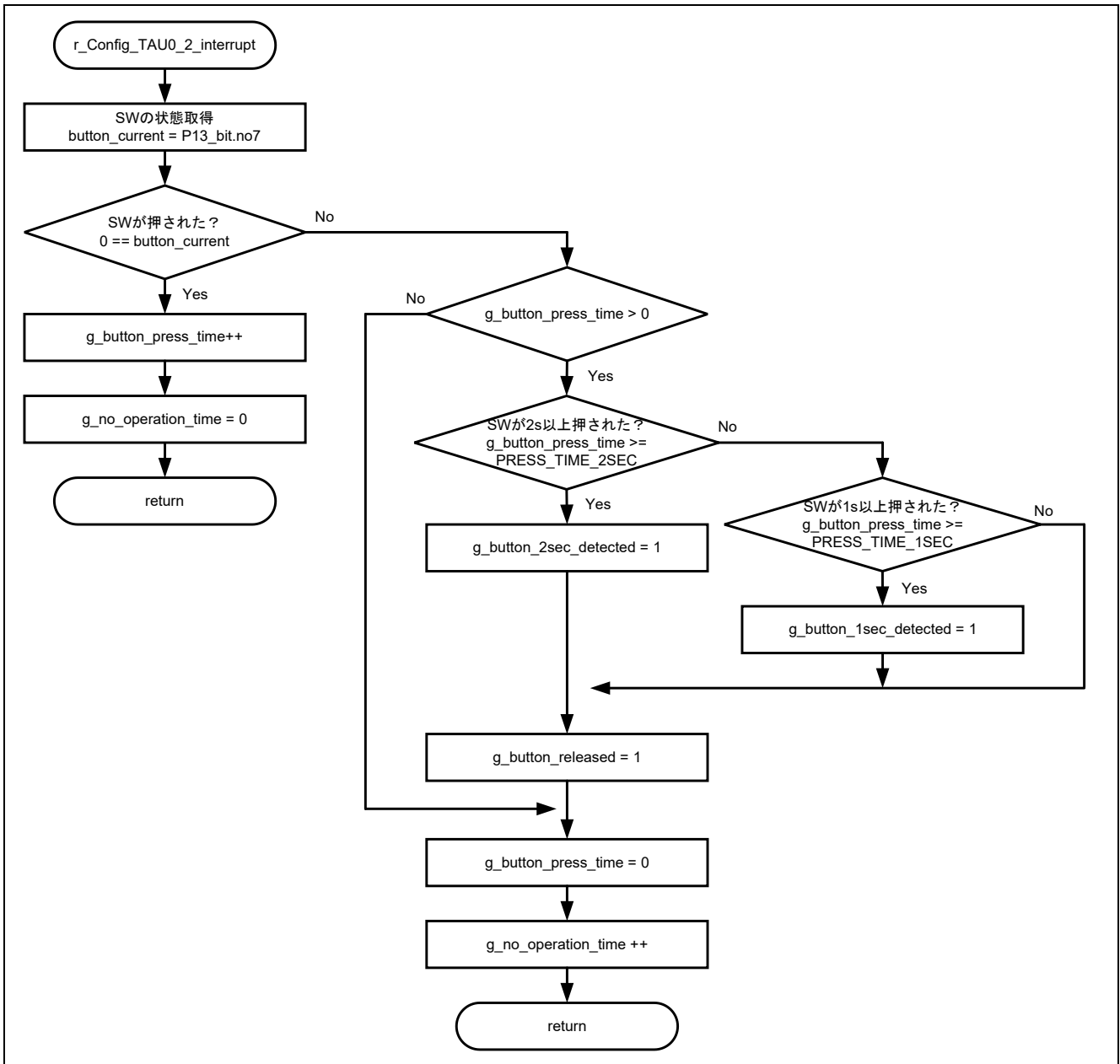
図 4-25 r_stop_pwm_output 関数フローチャート



4.9.20 r_Config_TAU0_2_interrupt 関数のフローチャート

r_Config_TAU0_2_interrupt 関数のフローチャートを以下に示します。

図 4-26 r_Config_TAU0_2_interrupt 関数フローチャート



4.10 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。

r01an7611_rl78g15_heater_control.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。

4.10.1 r01an7611_rl78g15_heater_control.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 4-13 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 2.7 (V) ~5.5 (V) EVDD 設定：2.7V ≤ EVDD0 < 5.5V 高速オンチップ・オシレータ：16MHz 低速オンチップ・オシレータ：15kHz 12 ビットインターバルタイマー用の動作クロックの供給に使用する fMAIN：16MHz fCLK：16000kHz（高速オンチップ・オシレータ） fIL：15kHz（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 疑似 RRM/DMM 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 RESET端子設定：使用する リセット発生電圧：2.52V |
| コンポーネント | r_bsp | Start up select：Enable (use BSP startup) Initialization of peripheral functions by Code Generator/Smart Configurator：Enable API functions disable(R_BSP_StartClock,R_BSP_StopCloc)：Disable API functions disable(R_BSP_GetFclkFreqHz)：Enable API functions disable(R_BSP_SetClockSource)：Disable API functions disable(R_BSP_ChangrClockSetting)：Disable API functions disable(R_BSP_SoftwareDelay)：Enable Parameter check enable：Enable Enable user warm start callback (PRE)：Unused Enable user warm start callback (POST)：Unused Watchdog Timer refresh enable：Unused |
| | Config_INTC | コンポーネント：割り込みコントローラ INTP0 設定：INTP0 有効エッジ：立ち下がりエッジ 優先順位：レベル 3（低優先順位） |

表 4-14 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 |
|-----|---------------|---|
| | Config_IT | コンポーネント：インターバル・タイマ 動作モード：12 ビット・カウンタ・モード リソース：IT インターバル時間：250ms 割り込み設定：使用する 優先順位：レベル 3（低優先順位） |
| | Config_TAU0_0 | コンポーネント：PWM 出力 リソース：TAU0_0 動作クロック：CK00 クロックソース：f _{CLK} 周期設定：1ms 割り込み設定：使用する 優先順位：レベル 3（低優先順位） PWM スレーブ選択設定：チャンネル 1 スレーブ (スレーブ 1) PWM デューティ設定：50% 初期出力値：0 出力レベル：アクティブ・ハイ 割り込み設定：使用する 優先順位：レベル 3（低優先順位） |
| | 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：CK01 クロックソース：f _{CLK} / 2 ⁸ インターバル時間：10ms カウント開始時に INTTM02 割り込みを発生しない 割り込み設定：使用する 優先順位：レベル 3（低優先順位） |
| | Config_ADC | コンポーネント：A/D コンバータ リソース：ADC コンパレータ動作設定：許可 分解能設定：10 ビット A/D チャンネルの選択：AN1 変換時間モード：標準 1 変換時間：184 / f _{CLK} 割り込み設定：使用する 優先順位：レベル 3（低優先順位） |

表 4-15 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 |
|-----|-------------|--|
| | Config_PORT | コンポーネント：ポート ポート選択：PORT0、PORT2、PORT4、PORT12、 P06：出力（1 を出力） P07：出力（1 を出力） P20：出力（1 を出力） P21：出力（1 を出力） P22：出力（1 を出力） P41：出力（1 を出力） P121：出力（1 を出力） P122：出力（1 を出力） |

4.10.1.1 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。

4.10.1.2 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、「4.2 オプション・バイトの設定一覧」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。

4.10.1.3 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。

4.10.1.4 Config_INTC

サンプルコードの INTP0 の設定を行います。

サンプルコードでは、SW2 押下の立ち下がリエッジを検知します。割り込みベクタ処理にて、SW2 を押下した際の初期化処理を行います。

4.10.1.5 Config_IT

サンプルコードの IT の設定を行います。

サンプルコードでは煮込みモードの時間管理に使用します。

4.10.1.6 Config_TAU0_0

サンプルコードの TAU00 の設定を行います。

サンプルコードでは、ヒータの過熱制御用の PWM 出力に使用します。

4.10.1.7 Config_TAU0_2

サンプルコードの TAU02 の設定を行います。

サンプルコードでは、ボタン長押し判定、動作モードの確定判定、チャタリング対策に使用します。

4.10.1.8 Config_ADC

サンプルコードの ADC の設定を行います。

サンプルコードでは、サーミスタの抵抗値の取得に使用します。

4.10.2 温度、時間設定の変更方法

本サンプルコードは以下の温度、時間設定がされています。各設定を変更する場合は以下の定数の値を変更してください。変更の際はヒータやサーミスタの仕様に合わせて十分に評価をして設定を行ってください。一定温度以上はヒータに供給する電圧調整が必要です。

表 4-16 各種設定

| 設定 | 定数名 | 値 |
|-------------|------------------|------------------|
| 湯沸かしモード温度設定 | TARGET_TEMP_LOW | 40 |
| | TARGET_TEMP_MID | 55 |
| | TARGET_TEMP_HIGH | 70 |
| 煮込みモード温度設定 | STEW_TEMP | TARGET_TEMP_HIGH |
| 煮込みモード時間設定 | STEW_TIME_SHORT | 10 |
| | STEW_TIME_MIDDLE | 40 |
| | STEW_TIME_LONG | 70 |
| 緊急停止温度設定 | TEMP_ABNORMAL | 80 |

5. プロジェクトのインポート方法

サンプルプログラムは e² studio のプロジェクト形式で提供しています。本章では、e² studio および CS+ ヘプロジェクトをインポートする方法を示します。インポート完了後、ビルドおよびデバッガの設定を確認してください。

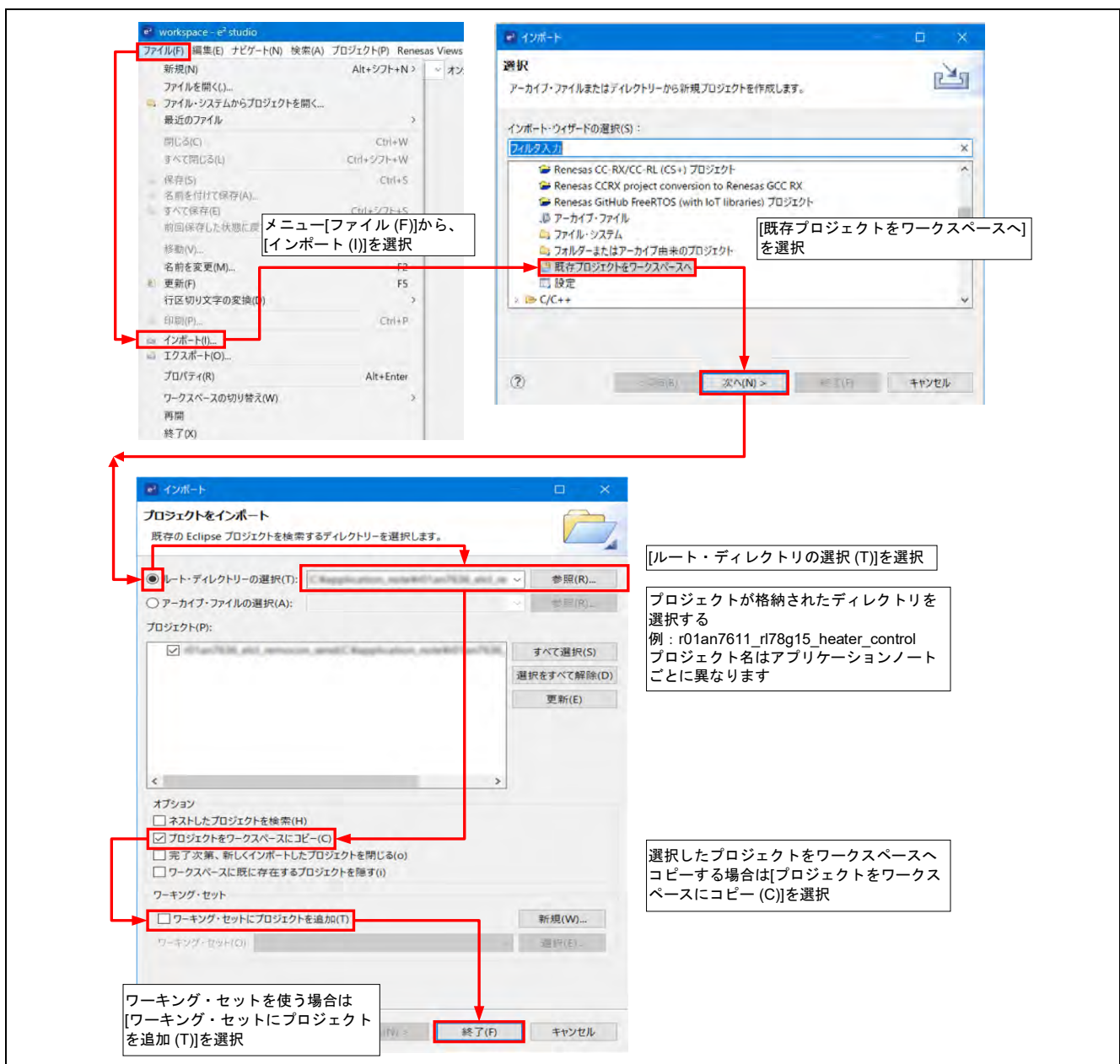
5.1 e² studio での手順

e² studio でご使用になる際は、以下の手順で e² studio にインポートしてください。

なお、e² studio で管理するプロジェクトのフォルダ名、およびそのフォルダに至るファイルパスには、空白文字の他、半角カナ文字、全角文字、半角記号 (特に '\$', '#', '%') が混じらないようにしてください。

(使用する e² studio のバージョンによっては画面が異なる場合があります。)

図 5-1 プロジェクトを e² studio にインポートする方法



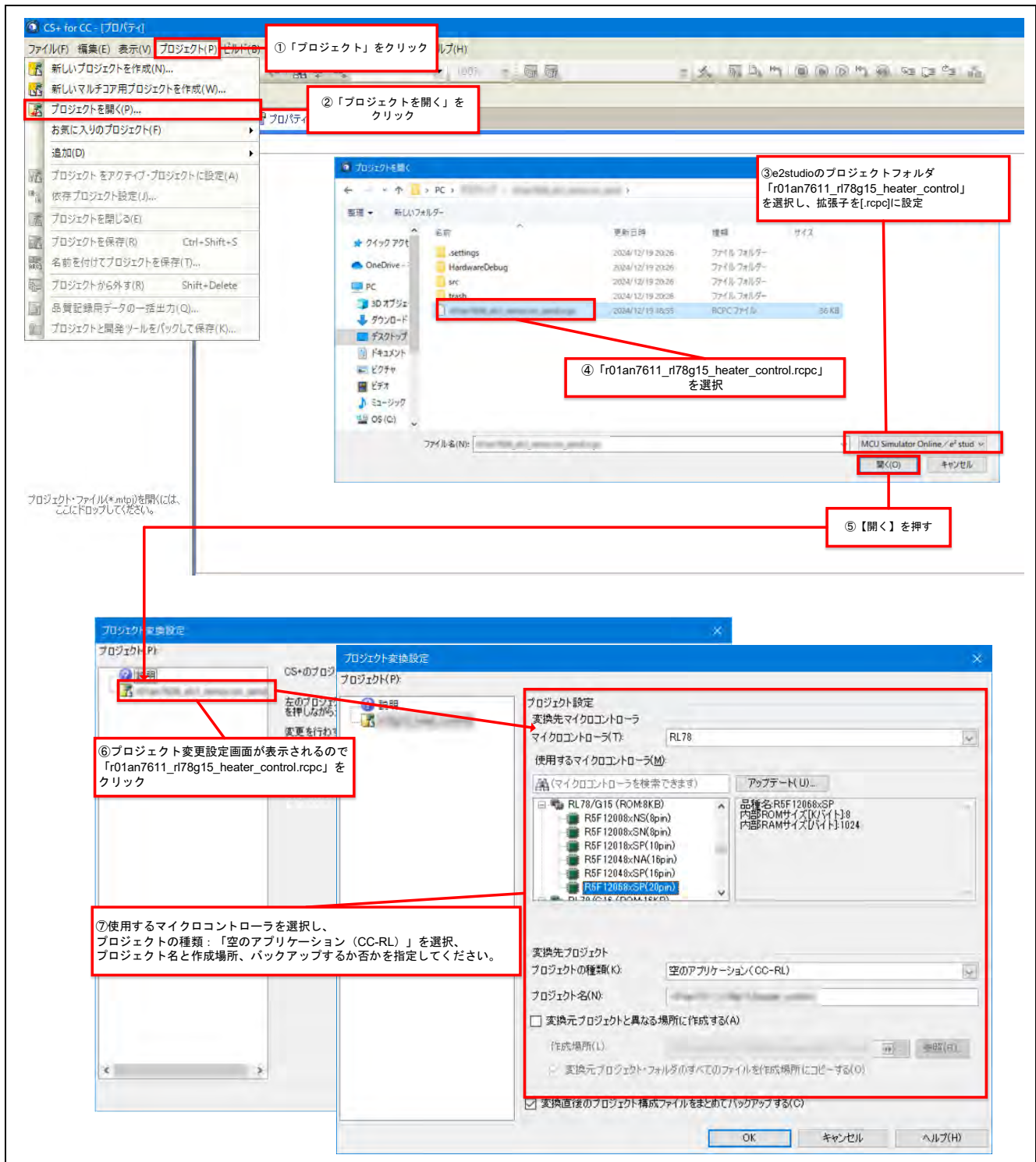
5.2 CS+ での手順

CS+ でご使用になる際は、以下の手順で CS+ にインポートしてください。

なお、CS+で管理するプロジェクトのフォルダ名、およびそのフォルダに至るファイルパスには、空白文字の他、半角カナ文字、全角文字、半角記号（特に'\$','#','%'）が混じらないようにしてください。

（使用する CS+ のバージョンによっては画面が異なる場合があります。）

図 5-2 プロジェクトを CS+ にインポートする方法



6. 参考資料

- RL78/G15 ユーザーズマニュアル ハードウェア編 (R01UH0959J)
- RL78 ファミリ ユーザーズマニュアル ソフトウェア編 (R01US0015)
- 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)
(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。)

改訂記録

| Rev. | 発行日 | 改訂内容 | |
|------|------------|------|------|
| | | ページ | ポイント |
| 1.00 | Mar.6.2025 | — | 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、切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

6. 入力端子の印加波形

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

7. リザーブアドレス（予約領域）のアクセス禁止

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

8. 製品間の相違について

型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。同じグループのマイコンでも型名が違えば、フラッシュメモリ、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。回路、ソフトウェアおよびこれらに関連する情報を使用する場合、お客様の責任において、お客様の機器・システムを設計ください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含みます。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. 当社製品または本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
 3. 当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
 4. 当社製品を組み込んだ製品の輸出入、製造、販売、利用、配布その他の行為を行うにあたり、第三者保有の技術の利用に関するライセンスが必要となる場合、当該ライセンス取得の判断および取得はお客様の責任において行ってください。
 5. 当社製品を、全部または一部を問わず、改造、改変、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改変、複製、リバースエンジニアリング等により生じた損害に関し、当社は、一切その責任を負いません。
 6. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット等
高品質水準： 輸送機器（自動車、電車、船舶等）、交通制御（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、これに限りません。）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としたします。
 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

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。

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/