

RL78/G24

DALI-2 Control Gear 基本(102) 調光(207) 調色(209) サンプルアプリケーション

要旨

本アプリケーションノートは、RL78/G24 マイクロコントローラを用いた DALI (Digital Addressable Lighting Interface)の通信を行うサンプルアプリケーションについて説明します。

サンプルアプリケーションを2種類用意しており、それぞれ Control Gear として動作します。対応している DALI 規格は IEC 62386-101 Edition2.0 (以下、101ed.2.0), IEC 62386-102 Edition2.0 (以下、102ed.2.0), IEC 62386-207 Edition1.0 (以下、207ed.1.0), IEC 62386-209 Edition1.0 (以下、209ed.1.0)です。各規格の組み合わせは2023年9月現在、DALI-2 認証に対応することが出来る組み合わせです。

規格に対応した通信を実現するために、RL78/G24 に搭載された周辺機能を用いてマンチェスタ符号化された DALI 信号の波形を処理しています。

本アプリケーションノートは DALI についての知識を既に持っていることを前提としています。DALI 規格の詳細については7. 参考ドキュメントを参照してください。

動作確認デバイス

RL78/G24

注意：各サンプルアプリケーションを他のマイコンへ適用する場合、そのマイコンの仕様に合わせて変更し、十分評価してください。

DALI-2 認証は半導体やソフトウェアのテストではありません。お客様で DALI-2 認証対応が可能な Control Gear として作りこみを行ってください。

目次

1. 仕様概要	4
2. 動作確認条件	6
3. ハードウェア説明	7
3.1 システム構成	7
3.2 使用端子一覧	8
4. ソフトウェア説明	8
4.1 環境構築方法	8
4.1.1 ソフトウェアのダウンロード	8
4.1.2 CS+のインストール	9
4.1.3 Renesas Flash Driver Type01 のインストール	9
4.1.4 DALI マスタコントローラ GUI のインストール	9
4.1.5 サンプルアプリケーションのインポート方法	9
4.1.6 Renesas Flash Driver の配置方法	10
4.1.7 サンプルアプリケーションのビルド方法	11
4.1.8 サンプルアプリケーションのハードウェア接続方法と設定スイッチ	11
4.1.8.1 ハードウェア接続	12
4.1.9 サンプルアプリケーションのデバッグ方法	13
4.2 動作手順	14
4.3 機能概要	17
4.3.1 機能ブロック図	17
4.3.2 ハードウェア非依存部	17
4.3.3 ハードウェア依存部	17
4.3.3.1 各サンプルアプリケーション共通機能	18
4.3.3.2 DALI102ed.2.0+207ed.1.0 サンプルアプリケーション固有機能	22
4.3.3.3 DALI102ed.2.0+207ed.1.0+209ed.1.0 サンプルアプリケーション固有機能	24
4.4 キャリブレーション	33
4.4.1 DALI 受信波形補正	33
4.4.2 DALI 送信波形補正	34
4.5 ソフトウェア構成	35
4.5.1 フォルダ構成	35
4.5.1.1 CPU で LED 調光制御を行うサンプルアプリケーション	36
4.5.1.2 FAA で LED 調光制御を行うサンプルアプリケーション	39
4.5.2 オプション・バイトの設定一覧	42
4.5.3 フローチャート	42
4.5.3.1 メインフロー	42
4.5.3.2 初期化	43
4.5.3.3 NVM 読み込み	43
4.5.3.4 開始処理	44
4.5.3.5 Control Gear タスク処理	45
5. 付録	46

5.1	DALI ドライバ	46
5.1.1	Backward Frame Transmitter (BFT).....	46
5.1.1.1	破損 Backward Frame の再現	46
5.1.2	Receiver (Rx).....	46
5.1.3	周辺機能	46
5.1.4	API 関数.....	47
6.	注意事項.....	49
7.	参考ドキュメント.....	49
	改訂記録.....	50

1. 仕様概要

本アプリケーションノートに記載するサンプルアプリケーションは以下の2つです。サンプルアプリケーションはそれぞれ LED 調光制御を CPU で行うプロジェクトと FAA(Flexible Application Accelerator)で行うプロジェクトの2種類存在します。

- DALI102ed.2.0+207ed.1.0 サンプルアプリケーション

Control Gear 機能 (102ed.2.0)に加え、LED 光源の拡張機能 (207ed.1.0)を搭載したサンプルアプリケーションです。拡張機能の主な内容として、線形/非線形の調光カーブの切り替え、フェード拡張、LED 光源特有の機能があります。

本サンプルアプリケーションは3 logical units で動作し、LED の故障状態については RL78/G24 DC/DC LED Control Evaluation Board 搭載のシリアル通信機能を用いて PC 上で設定・確認することが出来ます。

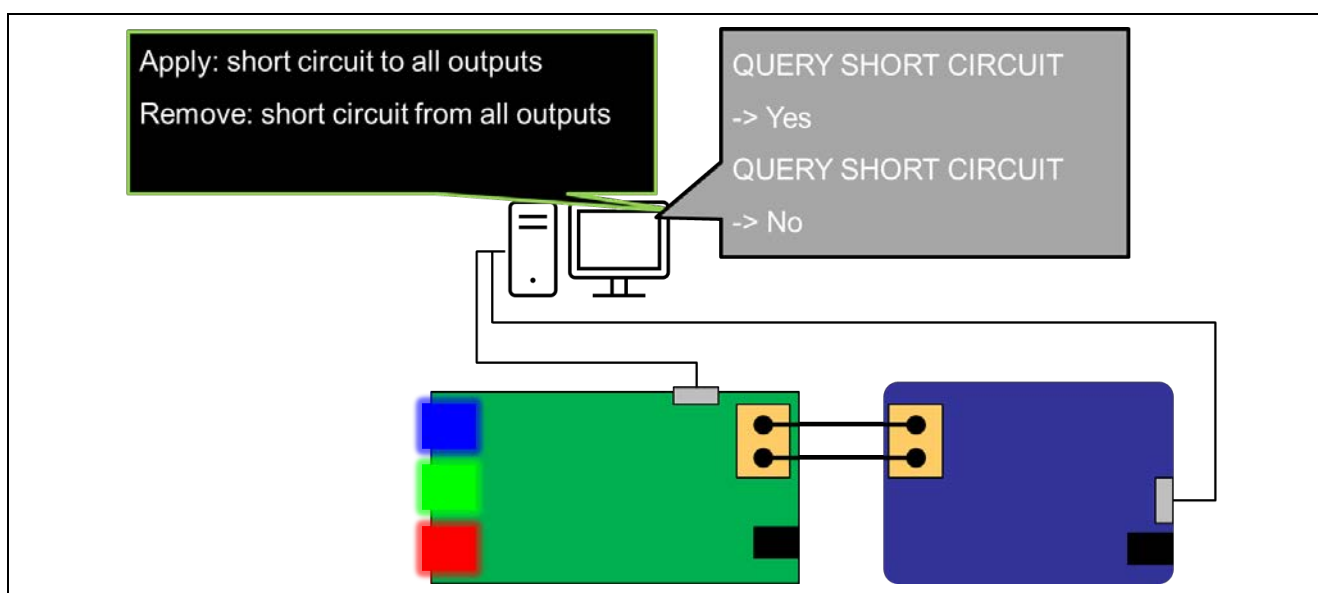


図 1-1 102ed.2.0+207ed.1.0 サンプルアプリケーション動作概要図

注：シリアルでの故障通知・復帰通知はあくまでもサンプルアプリケーションとして動作確認をするためのものです。実際の Control Gear に組み込む場合は仕様に合わせて組み込みを行ってください。

- DALI102ed.2.0+207ed.1.0+209ed.1.0 サンプルアプリケーション

Control Gear 機能 (102ed.2.0)に LED 光源の拡張機能 (207ed.1.0)と色彩制御の拡張機能 (209ed.1.0)を搭載したサンプルアプリケーションです。

LED 光源の拡張機能の主な内容として、線形/非線形の調光カーブの切り替え、フェード拡張、LED 光源特有の機能があります。

色彩制御の拡張機能の主な内容として、光源の調色方式(colour type)が 3 種類 (X,Y / RGBWAF / Tc)定義されています。

本サンプルアプリケーションは、1 logical unit で動作します。LED 光源の拡張機能として LED の故障状態については RL78/G24 DC/DC LED Control Evaluation Board 搭載のシリアル通信機能を用いて PC 上で設定・確認することが出来ます。

色彩制御の拡張機能として色温度 (Tc)、XY、RGBWAF に対応した調色を行っており、colour type は動作中に変更することが可能です。各 colour type にて使用する LED は以下となります。

Tc : 青 LED (高温域)と赤 LED (低温域)を使用します。

XY : 赤 LED、緑 LED、青 LED を使用します。

RGBWAF : 赤 LED、緑 LED、青 LED を使用し、動作中にいずれかの LED 調色を有効化/無効化することが可能です。

また、100ms 毎に actual level と現在有効になっている調色方式の調色値(colour value)を確認しいずれかの値に変更があった際にのみ PC 上のシリアル通信モニタに actual level 及び colour value を出力します。

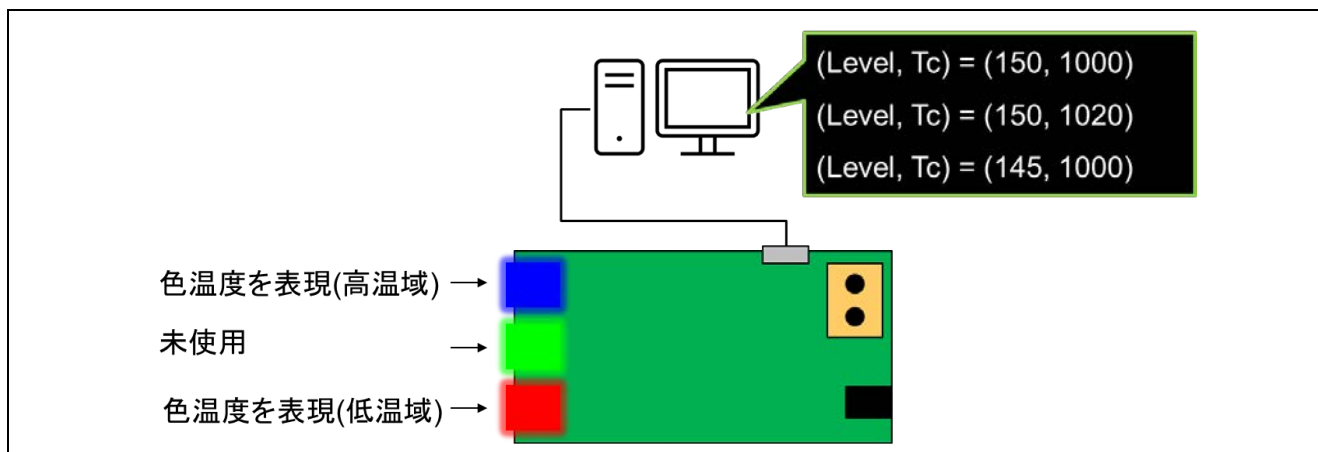


図 1-2 102ed.2.0+209ed.1.0 サンプルアプリケーション(colour type : Tc)動作概要図

注：シリアルへの表示動作はあくまでもサンプルアプリケーションとして Tc, XY 又は RGBWAF の colour value を確認するためのものです。

注：シリアル通信モニタに表示される Level は actual level を指します。

本アプリケーションノートでは、102ed.2.0 + 207ed.1.0 の組み合わせと 102ed.2.0 + 207ed.1.0 + 209ed.1.0 の組み合わせを取り上げていますが、組み替えることによって 102ed.2.0 単体、または 102ed.2.0, 209ed.1.0 を組み合わせた動作を行うことも可能です。

2. 動作確認条件

各サンプルアプリケーションは以下の環境で動作確認をしております。

表 2.1 動作確認環境

項目	内容
使用マイコン	RL78/G24 (R7F101GLG)
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 (f_{HOCO}) : 8MHz PLL 発振回路出力 (f_{PLL}) : 96MHz タイマ KB3 動作クロック : f_{PLL} = 96MHz
動作電圧	5.0V
統合開発環境 (CS+)	ルネサスエレクトロニクス製 CS+ for CC 8.10.00
C コンパイラ (CS+)	ルネサスエレクトロニクス製 CC-RL V1.12.01
統合開発環境 (IAR)	IAR システム製
C コンパイラ (IAR)	IAR Embedded Workbench for Renesas RL78 V4.21.1
ライブラリ	ルネサスエレクトロニクス製 Renesas Flash Driver Type01 V1.10
エミュレータ	E2 エミュレータ Lite
使用ボード	ルネサスエレクトロニクス製 RL78/G24 DC/DC LED Control Evaluation Board RX65N Cloud Kit+DALI-2 オプションボード RL78/G23 Lighting Communication Master Evaluation Board
使用ツール	DALI マスタコントローラ GUI V3.00.以降

3. ハードウェア説明

3.1 システム構成

システムの構成例を以下に示します。

各サンプルアプリケーションは RL78/G24 DC/DC LED Control Evaluation Board に対応した Control Gear 動作を行います。Control Gear は DALI システムのスレーブ機器にあたります。コマンド指示を出力するマスタ機器となる Application Controller が別途必要となります。本アプリケーションノートでは Application Controller の代わりに RL78/G23 Lighting Communication Master Evaluation Board もしくは RX65N Cloud Kit+DALI-2 オプションボードを使用します。

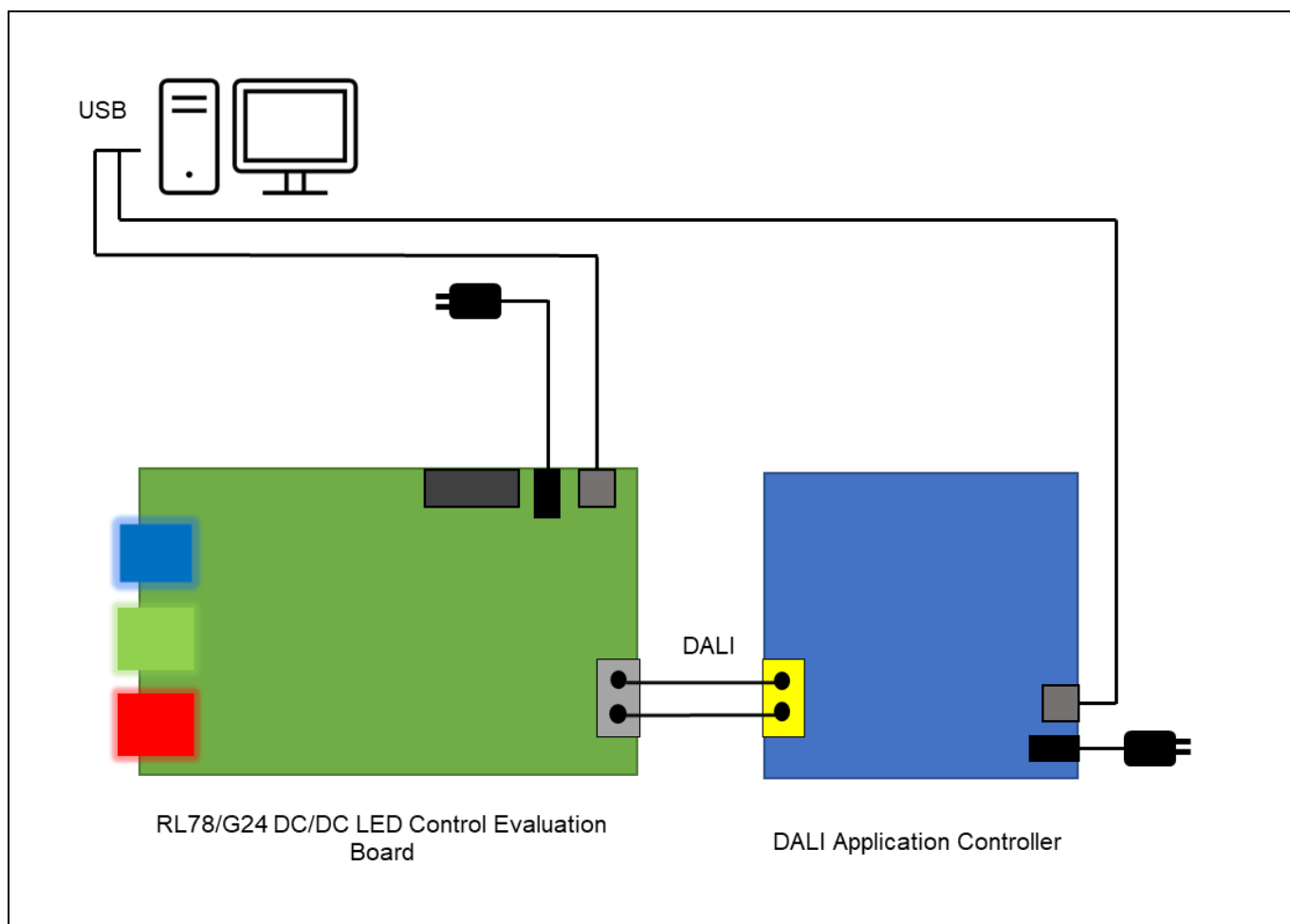


図 3-1 システム構成例

3.2 使用端子一覧

各サンプルアプリケーションで使用する端子と機能を以下に示します。

表 3.1 使用端子と機能

端子名	入出力	内容
P02/DALITxD0	出力	DALI 通信 (送信)
P03/DALIRxD0	入力	DALI 通信 (受信)
P50/RxD0	入力	UART 通信 (受信)
P51/TxD0	出力	UART 通信 (送信)
P12/TKBO00	出力	LED1 点灯
P13/TKBO01	出力	LED2 点灯
P14/TKBO10	出力	LED3 点灯
P120/IVCMP0	入力	LED1 点灯フィードバック
P00/IVCMP1	入力	LED2 点灯フィードバック
P01/IVCMP2	入力	LED3 点灯フィードバック

その他、RL78/G24 DC/DC LED Control Evaluation Board に関する機能については以下を参照してください。

- RL78/G24 DC/DC LED Control Evaluation Board ユーザーズ・マニュアル (R20UT5371)

4. ソフトウェア説明

4.1 環境構築方法

4.1.1 ソフトウェアのダウンロード

各サンプルアプリケーションを動作させるにあたって、以下のソフトウェアが必要になります。

ルネサスエレクトロニクスのホームページからダウンロードしてください。

- CS+ for CC
- Renesas Flash Driver type01 V1.10
- DALI マスタコントローラ GUI V3.00 以降

4.1.2 CS+のインストール

各サンプルアプリケーションのプログラム開発およびオンチップ・デバッグを行う場合は、CS+が必要になります。

インストール、その他基本操作についての詳細は以下ユーザーズマニュアルを参照してください。

- CS+ V8.10.00 統合開発環境ユーザーズマニュアル プロジェクト操作編 (R20UT5299)

4.1.3 Renesas Flash Driver Type01 のインストール

各サンプルアプリケーションはデータ・フラッシュ・ライブラリを含んでいません。そのため、ルネサスエレクトロニクスのホームページからダウンロードして、ライブラリファイルを登録していただく必要があります。

インストールについては以下ユーザーズマニュアルを参照してください。ライブラリファイルの登録については、4.1.6 章を参照してください。

- RL78 ファミリ Renesas Flash Driver RL78 Type01 ユーザーズマニュアル (R20UT4830)

4.1.4 DALI マスタコントローラ GUI のインストール

DALI マスタコントローラ GUI は各サンプルアプリケーションの評価を行う際に必要となります。

インストール手順は以下ユーザーズマニュアルを参照してください。

- DALI マスタコントローラ GUI ユーザーズマニュアル (R20UT0715)

4.1.5 サンプルアプリケーションのインポート方法

1. 提供プロジェクトファイルを解凍、任意の場所 (フォルダ) へ配置します。
2. CS+を起動し、「ファイル」タブ→「ファイルを開く」を選択します。
3. 1.で配置したフォルダ内の MTPJ ファイルを選択します。

4.1.6 Renesas Flash Driver の配置方法

動作させたいサンプルアプリケーションをインポート後、以下のディレクトリにルネサスエレクトロニクスからダウンロードしたデータ・フラッシュ・ライブラリを格納してください。

\DALI102_xxx_sample<DIR>	
\Library<DIR>	
\RFD<DIR>	Renesas Flash Driver を格納するフォルダです。
\userown	本サンプルアプリケーションに合わせて設定
r_rfd_common_userown.c	
\include	RFD からコピーしてください。
\rfd	
r_rfd.h	
r_rfd_compiler.h	
r_rfd_device.h	
r_rfd_memmap.h	
r_rfd_types.h	
r_typedefs.h	
r_rfd_common_api.h	
r_rfd_common_control_api.h	
r_rfd_common_extension_api.h	
r_rfd_common_get_api.h	
r_rfd_common_userown.h	
r_rfd_data_flash_api.h	
\source	
\commom	
r_rfd_common_api.c	
r_rfd_common_control_api.c	
r_rfd_common_extension_api.c	
r_rfd_common_get_api.c	
\dataflash	
r_rfd_data_flash_api.c	

4.1.7 サンプルアプリケーションのビルド方法

1. プロジェクト・ツリーでプロジェクトを右クリックし、「プロジェクトのビルド」を選択します。
2. ビルドが開始され、「出力」にビルドの状況が表示されるので“ビルド終了”というメッセージが表示されたらビルド完了です。

4.1.8 サンプルアプリケーションのハードウェア接続方法と設定スイッチ

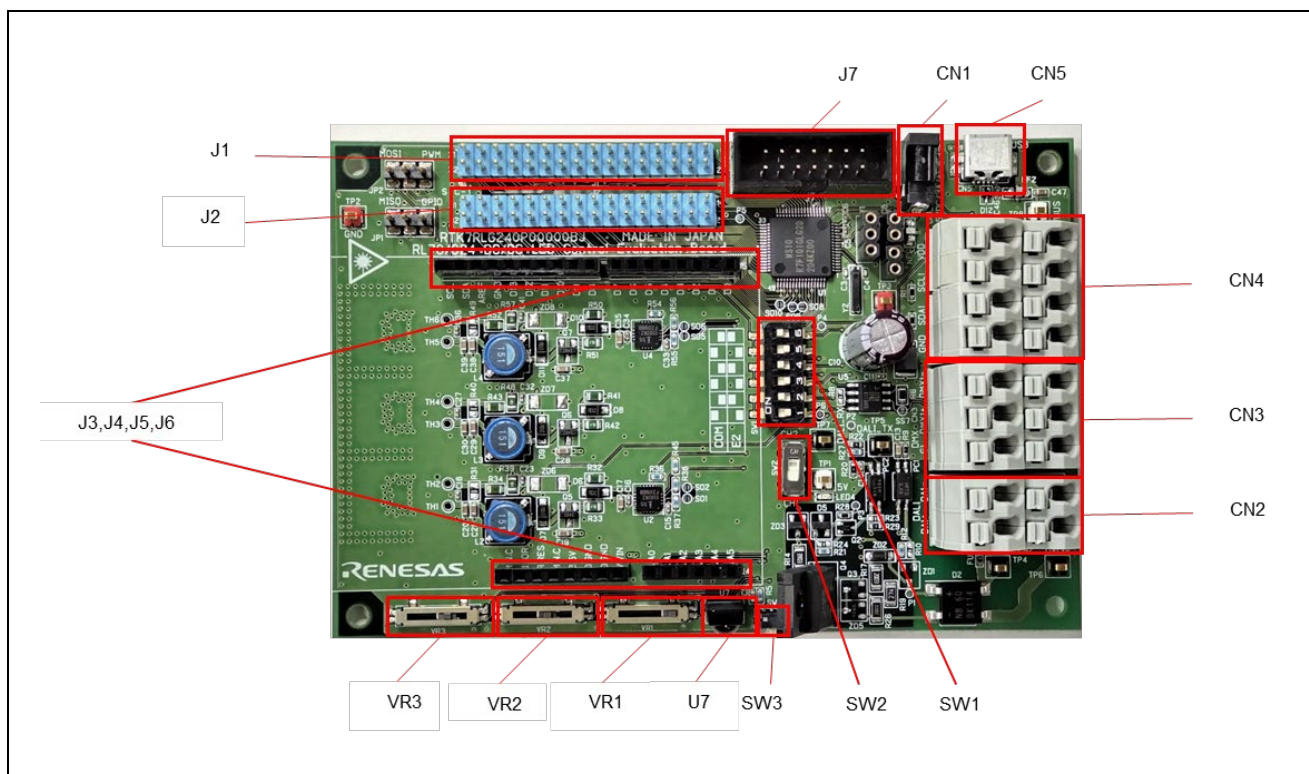


図 4-1RL78/G24 DC/DC LED Control Evaluation Board

RL78/G24 DC/DC LED Control Evaluation Board の設定スイッチ(SW1)の内容を変更する必要があります。設定スイッチの内容を以下に示します。

表 4.1 設定スイッチ SW1 (RL78/G24 DC/DC LED Control Evaluation Board)

SW1	設定
1	OFF
2	ON
3	OFF
4	ON
5	OFF
6	ON
7	OFF
8	ON

4.1.8.1 ハードウェア接続

ハードウェア接続は図 4-2 のようになります。

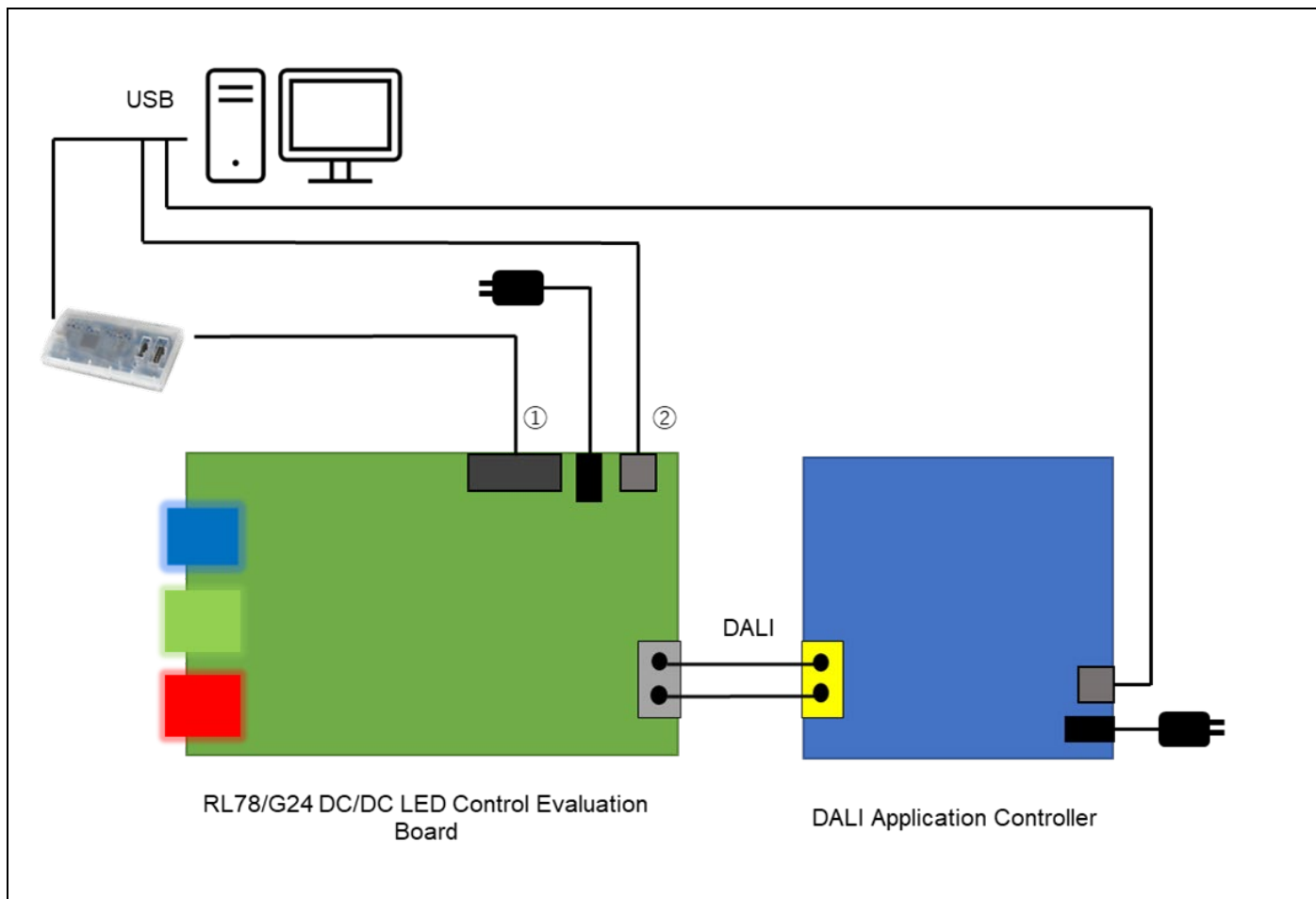


図 4-2 接続図

エミュレータによるデバッグを行う際は①を接続します。

シリアル通信モニタを使用する場合は②を接続します。

4.1.9 サンプルアプリケーションのデバッグ方法

1. プロジェクト・ツリーで「RL78 E2 Lite」を右クリックし「プロパティ」を開き、下図の設定になっていることを確認してください。

※ データフラッシュに格納されているデータを削除したい場合は「起動時にフラッシュ ROM を削除する」を「はい」に設定してください。



2.  ボタンをクリックしてマイクロコントローラへプログラムをダウンロードします。
3.  ボタンをクリックし、プログラムを実行します。



図 4-3 デバッグ画面設定内容

インストール、その他基本操作についての詳細は以下ユーザーズマニュアルを参照してください。

- CS+ V8.10.00 統合開発環境ユーザーズマニュアル プロジェクト操作編(R20UT5299)

4.2 動作手順

本アプリケーションノートでは、DALI マスタコントローラ GUI を使用して各サンプルアプリケーションを動作させる手順を以下に示します。

(1) 共通動作手順

1. RL78/G24 DC/DC LED Control Evaluation Board と、RL78/G24 DC/DC LED Control Evaluation Board を操作する Application Controller を接続後、各機器に電源を供給します。
接続方法は 3.1 章システム構成を参照してください。
2. シリアル通信ソフトウェア (Tera Term 等) を使用し、RL78/G24 DC/DC LED Control Evaluation Board と接続されている USB の COM ポートを選択後、接続します。接続設定は以下となります。

表 4.2 シリアル通信接続設定

項目	設定内容
通信ボーレート	9600
データビット	8bit
パリティ	none
ストップビット	1bit
フロー制御	none

3. サンプルアプリケーションをビルド後デバuggヘダウンロードし、デバugg実行します。
4. DALI マスタコントローラ GUI を起動後 Application Controller と接続されている USB の COM ポートを選択し、接続します。
5. DALI マスタコントローラ GUI を操作し、灯具の点灯状態が変わることを確認します。
例 1：接続されている灯具全てを MAX の光量で点灯させる→Broadcast を選択後、MAX ボタンを押す
例 2：接続されている灯具全てを消灯させる→Broadcast を選択後、OFF ボタンを押す

(2) DALI102ed.2.0+207ed.1.0 サンプルアプリケーションの場合

6. 4.2(1) 共通動作手順を実施後、DALI マスタコントローラ GUI の Manual Command を開き、Control Gear タブを選択します。
7. フェード時間を変更するコマンドを送信します。
例：フェード時間を 4s に設定後 400ms に変更する
 - (1)フェード時間を 4s に設定
 - RESET を送信
 - DTR0(6)を送信
 - SET FADE TIME を送信
 - DAPC コマンドを送信、4s かけて点灯状態が変化することを確認します。DALI マスタコントローラ GUI の Manual Command を開き、Control Gear タブを選択します。
 - (2)Fast fade time を 400ms に設定
 - RESET を送信
 - DTR0(16)を送信
 - 「with ENABLE DEVICE TYPE command」にチェックを入れて、STORE DTR AS FAST FADE TIME を送信
 - DAPC コマンドを送信、400ms で点灯状態が変化することを確認します。
8. LED の故障状態の確認は「4.3.3.2(2)デバッグ機能 (疑似異常発生機能)」に記載しているキーを押下し、故障状態を更新させます。
例：シリアルにて q を送信し、SHORT CIRCUIT を true に設定
9. DALI マスタコントローラ GUI にて、「QUERY SHORT CIRCUIT」を送信します。
例：DALI マスタコントローラ GUI にて、「with ENABLE DEVICE TYPE command」にチェックを入れて、「QUERY SHORT CIRCUIT」を送信
10. DALI マスタコントローラ GUI の Command Log で応答結果が更新した故障状態に合わせて変化していることを確認します。

(3) DALI102ed.2.0+207ed.1.0+209ed.1.0(all)サンプルアプリケーションの場合

(a) Colour type : Tc で色温度を変化させる

6. 4.2(1) 共通動作手順を実施後、DALI マスタコントローラ GUI の Send Code...から Manual command を開き、Control Gear タブを選択します。
7. 色温度 (Tc)を変更するコマンドを送信します。
例：色温度 (Tc)を 100[mirek]分暖かい色に変更する
 - DTR0 (100)を送信
 - 「with ENABLE DEVICE TYPE command」にチェックを入れて、STORE COLOUR TEMPERATURE Tc STEP INCREMENT を Broadcast で Twice 送信
 - COLOUR TEMPERATURE Tc STEP WARMER を Broadcast で送信
8. 灯具の点灯状態が可変することを確認します。
9. シリアル出力では、上記に合わせて可変値が出力されることを確認します。

(b) Colour type : RGBWAF で色を変化させる

6. 4.2(1) 共通動作手順を実施後、DALI マスタコントローラ GUI の Send Code... から Manual command を開き、Control Gear タブを選択します。
7. RGBWAF を変更するコマンドを送信します。
例 : RGBWAF で各原色を制御方法 : n で R=40%、G=100%、B=60%に変更する
 - DTR(128)を送信
 - 「with ENABLE DEVICE TYPE command」にチェックを入れて、SET TEMPORARY RGBWAF CONTROL を Broadcast で送信
 - DTR0(7)を送信
 - 「with ENABLE DEVICE TYPE command」にチェックを入れて、STORE ENABLED CHANNELS を Broadcast で Twice 送信
 - DTR0(100)を送信
 - DTR1(250)を送信
 - DTR2(150)を送信
 - 「with ENABLE DEVICE TYPE command」にチェックを入れて、SET TEMPORARY RGB DIMLEVEL を Broadcast で送信
 - 「with ENABLE DEVICE TYPE command」にチェックを入れて、ACTIVATE を Broadcast で送信
8. 灯具の点灯状態が可変することを確認します。
9. シリアル出力では、上記に合わせて可変値が出力されることを確認します。

(c) Colour type : XY で色を変化させる

6. 4.2(1) 共通動作手順を実施後、DALI マスタコントローラ GUI の Send Code... から Manual command を開き、Control Gear タブを選択します。
7. 座標点 XY を変更するコマンドを送信します。
例 : XY 空間上の $(x, y) = (0.5, 0.4)$ に色を変更する
 - DTR0(0)を送信
 - DTR1(128)を送信
 - 「with ENABLE DEVICE TYPE command」にチェックを入れて、SET TEMPORARY x-COORDINATE を Broadcast で送信
 - DTR0(102)を送信
 - DTR1(102)を送信
 - 「with ENABLE DEVICE TYPE command」にチェックを入れて、SET TEMPORARY y-COORDINATE を Broadcast で送信
 - 「with ENABLE DEVICE TYPE command」にチェックを入れて、ACTIVATE を Broadcast で送信
8. 灯具の点灯状態が可変することを確認します。
9. シリアル出力では、上記に合わせて可変値が出力されることを確認します。

その他、DALI マスタコントローラ GUI に関する詳細な操作方法は DALI マスタコントローラ GUI ユーザーズマニュアル (R20UT0715) を参照してください。

コマンド等の詳細は 7 章参考ドキュメントの各 DALI 規格書を参照してください。

4.3 機能概要

4.3.1 機能ブロック図

各サンプルアプリケーションの機能ブロック図を以下に示します。

※ DALI207 ライブラリは DALI102ed.2.0+207ed.1.0 サンプルアプリケーションにのみ、DALI209 ライブラリは DALI102ed.2.0+209ed.1.0 サンプルアプリケーションにのみ含まれます。

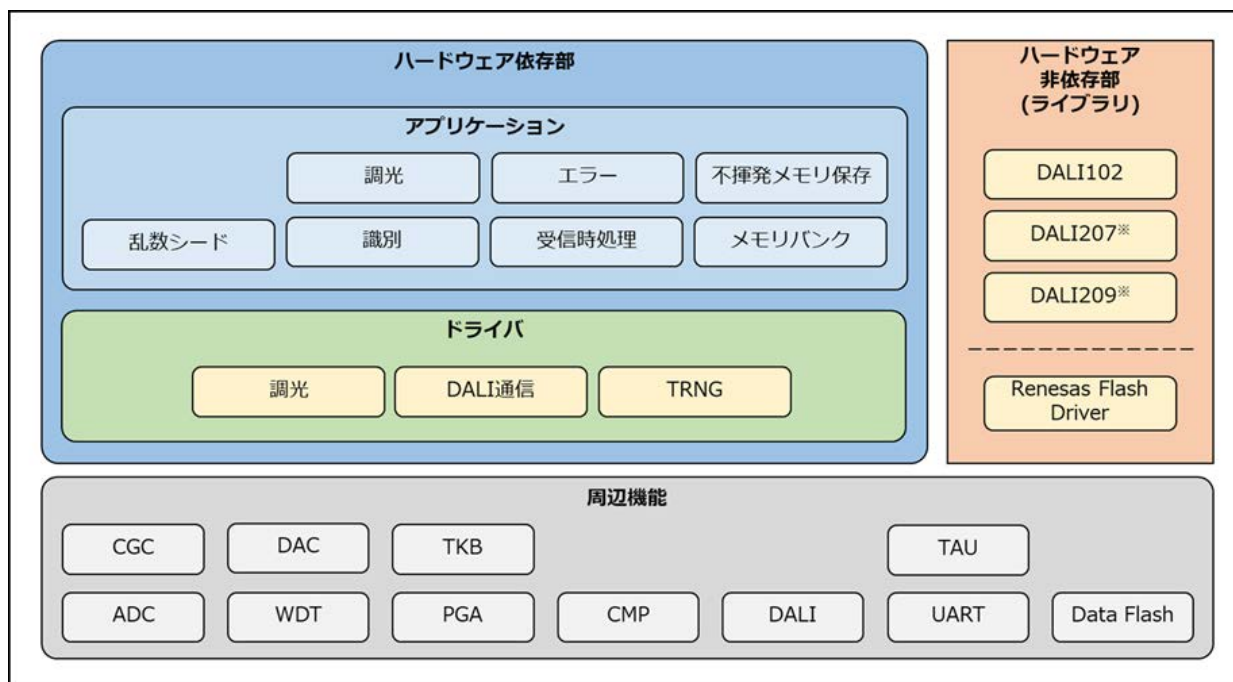


図 4-4 機能ブロック図

4.3.2 ハードウェア非依存部

ハードウェア非依存部は DALI ライブラリにて対応しています。詳細は各ライブラリのユーザーズマニュアルを参照してください。

Renesas Flash Driver は不揮発性メモリ保存に使用されます。Renesas Flash Driver の詳細は RL78 ファミリー Renesas Flash Driver RL78 Type01 ユーザーズマニュアル (R20UT4830) を参照してください。

4.3.3 ハードウェア依存部

ハードウェア依存部では、RL78/G24 DC/DC LED Control Evaluation Board のハードウェア構成に合わせた実装を行っています。DALI 製品を開発される際は規格と使用目的に合わせた動作への検討・変更を行い、十分に評価を行ってください。

各サンプルアプリケーションに実装している機能を以降の章に示します。

4.3.3.1 各サンプルアプリケーション共通機能

(1) 調光ドライバ

Control Gear の出力となる LED 灯具の調光を行うドライバです。タイマ KB 機能と PGA 機能、コンパレータ機能を使用して各 LED の点灯・消灯を制御します。調光対象の LED は 3 つあり、各々の灯具の状態を取得・調光を指示します。

(2) DALI 通信ドライバ

101ed.2.0 にて要求される DALI 通信仕様を満たしたドライバです。

DALI の送受信は、DALI モジュールを使用しており、System Failure や settling time の検出は TAU 機能を使用しています。

(3) 不揮発性メモリ保存

データフラッシュへの保存が必要な変数に関しては、以下のタイミングでデータフラッシュへの保存を行います。

- ライブラリから変数に対して変化要求が発生してから 1s 経過時
- 受信 Frame による保存要求を受け取った場合

また、DALI 規格では上記に加え電源 OFF を検出した際に保存を行う動作を推奨しています。本サンプルアプリケーションではボードが電源 OFF 検出に未対応の為、定期的に保存処理を行う仕様としています。

データフラッシュへのアクセスは Renesas Flash Driver Type01 を使用します。データフラッシュは 4 つのブロック※(ブロック 0~3)で構成され、ブロック 0,1 は更新頻度の高いデータの保存、ブロック 2,3 は更新頻度の低いデータの保存に割り当てられています。

※ここで言うブロックとは、データフラッシュの物理ブロック(256byte)4 つを組み合わせた仮想ブロック(1024byte)のことを指します。

ブロック構成を図 4-5 に示します。

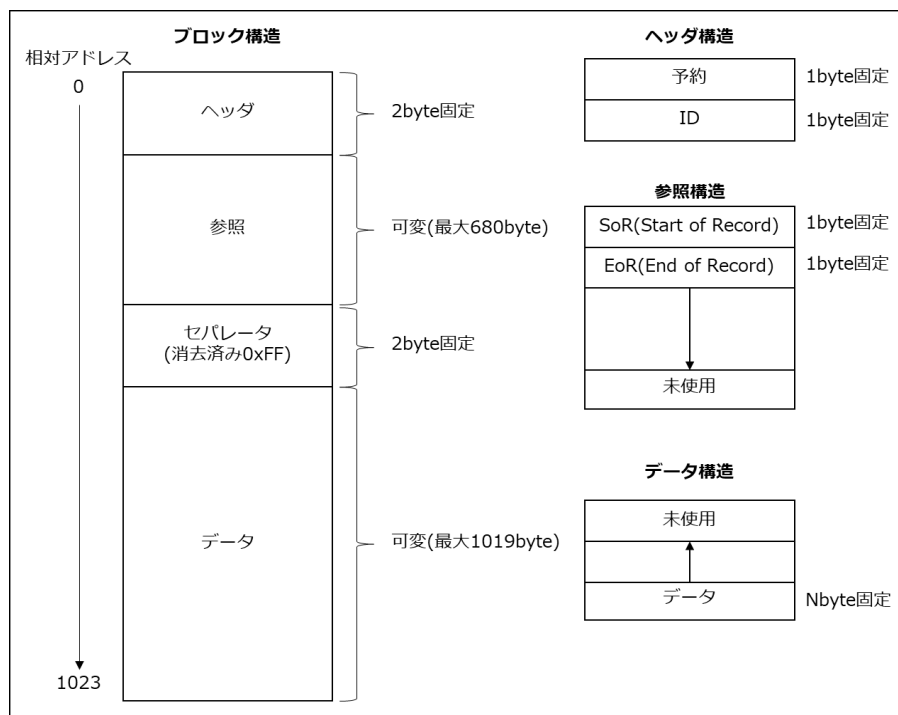


図 4-5 ブロック構造図

(a) ヘッダ構造

ヘッダ部分には ID を割り当て、その値から書き込みまたは消去するブロックの選択を行います。

表 4.3 ヘッダ構造

ヘッダ	有効値	無効値
ID	0 または 1	有効値以外の値

更新頻度の高いブロック 0,1 と更新頻度の低いブロック 2,3 において、偶数ブロックと奇数ブロックの両方が有効な場合は、以下のルールでブロック選択を行います。

- 奇数ブロックと偶数ブロックの ID が等しい場合：奇数ブロックを選択
- 奇数ブロックと偶数ブロックの ID が異なる場合：偶数ブロックを選択

(b) 参照構造

データ書き込み時に SoR、書き込み完了時に EoR を設定します。
 起動時に各値を使用して、最新のデータが保存されている場所を取得します。

表 4.4 参照構造

参照	有効値	無効値
SoR	0x12：書き込み開始 0xFF：未使用	有効値以外の値
EoR	0x34：書き込み完了 0xFF：未完了	有効値以外の値

(c) データ構造

データの書き込みは下位アドレスから上位アドレスへ向かって行います。

(d) ノンブロッキングな処理の優先順位

初期化や読み出しはブロッキングして行いますが、書き込みや消去はノンブロッキングで処理されます。その時の優先順位は下記となります。

1. 高頻度ブロックのデータ書き込み
2. 低頻度ブロックのデータ書き込み
3. 高頻度ブロックのデータ消去
4. 低頻度ブロックのデータ消去

(e) 消去のトリガ

初期化時に有効ではないブロックの消去を行います。最初の書き込みが成功した時に、もう片方のブロックを消去します。

(4) メモリバンク

規格に規定されているメモリバンクと、メモリバンクを保存・管理する処理を実装します。各サンプルアプリケーションでは規格に必要なメモリバンクを各 logical unit 分定義し、管理を行います。

(5) Frame 受信時の処理

DALI 通信ドライバより受け取った受信 Frame を DALI ライブラリの関数に渡し、必要があれば backward frame を有効期間内に DALI 通信ドライバを使用して送信します。

また、破損 backward frame を送信する必要がある場合は、破損 backward frame の送信要求を DALI 通信ドライバへ通知し送信を行います。

(6) 調光処理

各 logical unit のステータスに応じた調光量を DALI ライブラリから取得し、調光ドライバを経由して灯具に反映させます。識別動作中には点滅処理を行います。異常が発生している場合、DALI ライブラリへ異常を通知し異常内容によっては各 LED を消灯させます。

(7) 識別処理

各サンプルアプリケーションでは識別動作モード中、該当する logical unit の LED を点滅させます。

(8) エラー処理

ソフトウェア動作中に発生した異常を検出し、異常内容によっては DALI ライブラリの関数へ通知を行います。本サンプルアプリケーションではデバッグ機能 (疑似異常発生機能) で異常を発生させます。

- 疑似異常発生機能について

本来は灯具の状態から判断し故障検出・復帰を DALI ライブラリへ通知し故障状態の管理を行いますが、灯具一体型かつ電源 OFF 検出未対応の為、故障検出・復帰検出を疑似的に発生させます。故障検出・復帰検出は該当操作キーをシリアルにて送信することで設定させます。該当キーを押下した際、応答結果として設定した内容が PC 上に接続されたシリアル通信モニタに出力されます。

各サンプルアプリケーション共通で設定可能な故障内容を以下に示します。

表 4.5 各サンプルアプリケーション共通故障内容一覧

項目	関連する故障	操作キー	説明
control gear failure	CONTROL GEAR FAILURE	0, 1	機器の故障 0: 機器の故障状態を解消 1: 機器の故障状態に設定
failure procedure	SHORT CIRCUIT	-*	短絡回路の検出*
	OPEN CIRCUIT	-*	開回路の検出*
	failure status の全ての故障	c	すべての異常をクリア

※各サンプルアプリケーションにて対応するキーや内容が異なるため、使用するサンプルアプリケーションの疑似異常発生機能を参照してください。

注 1: 未設定のキーを送信した場合、シリアル出力画面に"unknown key"が表示されます。

注 2: control gear failure 設定は 102 規格の controlGearFailure ビットに反映されます。この設定はデータフラッシュメモリ(以降、NVM)に保存されません。

注 3: failure procedure 設定はテストで要求される物理的な操作を表しています。これを操作するとライブラリが管理する actualLevel や current protector の設定を考慮した上で lampFailure や failure status 等に反映されます。この設定は NVM に保存されます。

(9) 乱数・シード値の生成処理

DALI ライブラリにて使用する乱数のシード値を真性乱数発生器 (TRNG) で生成します。

4.3.3.2 DALI102ed.2.0+207ed.1.0 サンプルアプリケーション固有機能

207ed.1.0に対応した機能を以下に示します。

(1) failure status について

207ed.1.0 では、4.3.3.1(8)エラー処理に加え、オプションで以下の故障を検出します。

- SHORT CIRCUIT
- OPEN CIRCUIT
- LOAD DECREASE
- LOAD INCREASE
- CURRENT PROTECTOR
- THERMAL SHUT DOWN
- THERMAL OVERLOAD
- REFERENCE SYSTEM POWER FAILED

(2) デバッグ機能 (疑似異常発生機能)

207ed.1.0 では、4.3.3.1(8)章エラー処理に加え以下の故障を管理します。

該当操作キーの大文字/小文字にて故障状態の設定/解消を行います。キーに割り当てられている異常内容を下表に示します。

表 4.6 207ed.1.0 用操作キーと対応する故障内容一覧

項目	関連する故障	操作キー	説明
failure procedure	SHORT CIRCUIT	q, Q	短絡回路の検出 q: 全ての出力を短絡状態に設定 Q: 全ての出力の短絡状態を解消
	OPEN CIRCUIT	w, W	開回路の検出 w: 全ての出力を開回路状態に設定 W: 全ての出力の開回路状態を解消
	LOAD DECREASE	e, E	負荷減少の検出 e: 一部の負荷の接続を切ります E: 一部の負荷を再接続します
	LOAD INCREASE	r, R	負荷増加の検出 r: 追加の負荷を接続 R: 追加した負荷の接続を切ります
	CURRENT PROTECTOR	t, T	電流保護有効状態 t: underload/overload の発生を設定 T: underload/overload 状態を解消
	THERMAL SHUT DOWN	y, Y	過熱保護の検出 y: 過熱を検出 (負荷の過熱) Y: 過熱状態の解消
	THERMAL OVERLOAD	u, U	過熱による光レベル低下の検出 u: 過熱を検出 (負荷の停止) U: 過熱状態の解消
	REFERENCE SYSTEM POWER FAILED	i, I	LED 負荷の基準測定結果 i: 測定結果=失敗を設定 I: 測定結果=成功を設定
	failure status の全ての故障	c	すべての異常をクリア
	reference measurement の終了	f	LED 負荷の基準測定を終了

注 1: reference measurement 処理は即座に完了させることができます。何もしない場合は 15 分で完了します。測定結果は failure procedure 設定 (i, I) で設定します。

注意: 本アプリケーションノートでは、エラー処理を確認するために、疑似的な発生機能を実装しています。実際の製品に実装する際は、7章参考ドキュメントの DALI 規格書: 102ed.2.1 と 207ed.1.0 を参照して実装してください。

4.3.3.3 DALI102ed.2.0+207ed.1.0+209ed.1.0 サンプルアプリケーション固有機能

209ed.1.0に対応した機能を以下に示します。

(1) LED 調色機能(Tc)

本サンプルアプリケーションでは、1つの logical unit に対して2つの LED (低温域 LED、高温域 LED)で調色を表現します。

例として、色温度 (Tc)が低い時は「低温域 LED の調光率を高く、高温域 LED の調光率を低く」設定します。色温度 (Tc)の変化に伴って各 LED の調光率は変化しますが、各 LED の調光率の和は、常に actual level で指定された調光率と一致するように設定します。

色温度 (Tc)と各温域 LED の関係図と、総調光率 (actual level)と色温度 (Tc)の関係図を以下に示します。

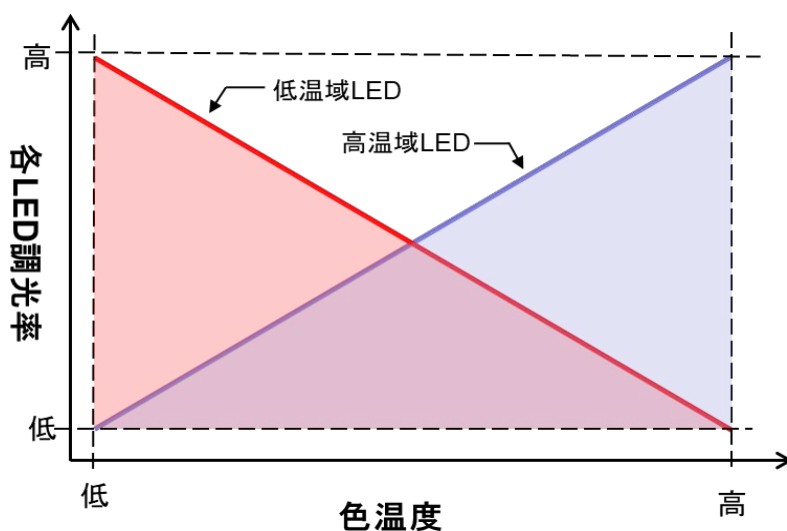


図 4-6 色温度と各温域 LED の関係

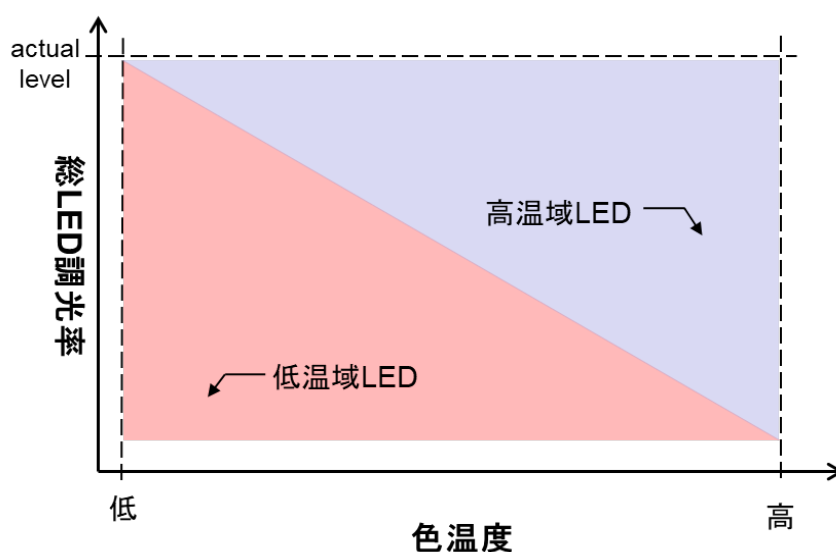


図 4-7 総調光率 (actual level)と色温度 (Tc)の関係

各 LED の調光率は actual level によっても変化します。actual level が変化することにより各 LED の調光率は変化しますが、低温域 LED と高温域 LED の調光率の比は変わりません。

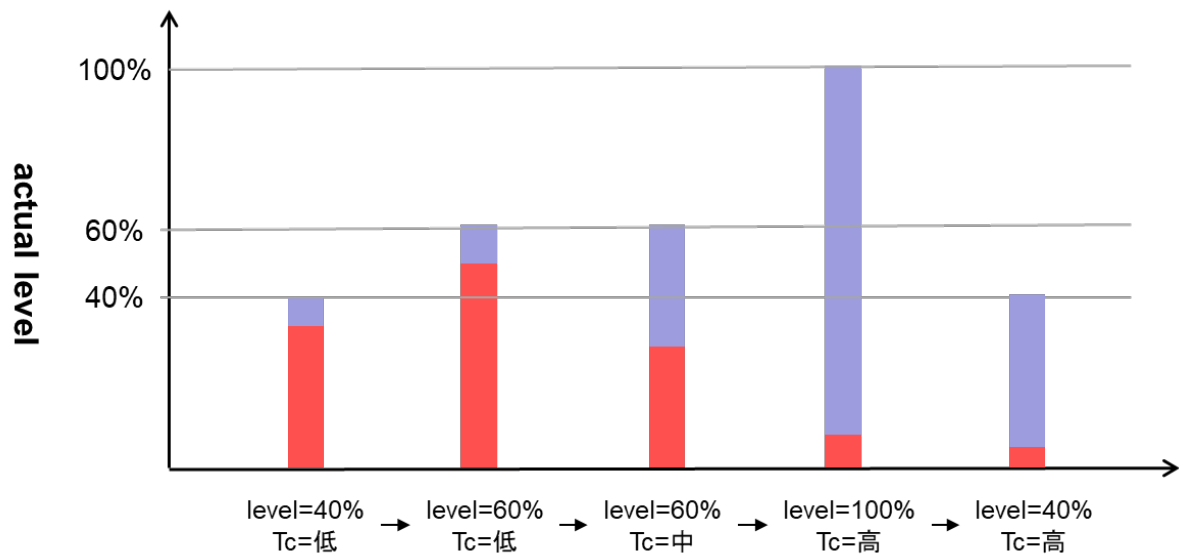


図 4-8 総調光率 (actual level) と各温度域 LED の調光率の比

LED の物理的な最小調光率として PHM が規定されています。本サンプルアプリケーションの調光/調色ドライバを使用する際は、209ed.1.0 を拡張していないときの PHM と異なる値を設定する必要があります。その理由として、actual level で設定される調光率を低温域 LED と高温域 LED の 2 つの調光率の和とするために、actual level に PHM を設定すると少なくともいずれかの LED が PHM を下回る調光率になってしまう為です。

また、209ed.1.0 には physical coolest と physical warmest という色温度 (Tc) の物理的制限が規定されています。本サンプルアプリケーションでは各 LED が有効な actual level 範囲内で物理的な最小調光率を下回らないよう PHM, physical coolest, physical warmest の 3 変数を設定する必要があります。

なお、RL78/G24 DC/DC LED Control Evaluation Board の各 LED の最小調光率 (PHM) が 70 (0.658%) であり、本サンプルアプリケーションでは 3 変数を以下の設定とすることで物理的制限を満たすようにしています。

- PHM : 170 (10.091%)
- physical coolest : 0x0096
- physical warmest : 0x01C2

色温度 Tc	0x0064	~	0x0096 (MAXの25%)	~	0x012C	~	0x01C2 (MAXの75%)	~	0x01F4
actual level(PHM)	170(10.091%)								
高温域LED調光率	10.091% (100%*10.091%)	~	7.568% (75%*10.091%)	~	5.0455% (50%*10.091%)	~	2.522% (25%*10.091%)	~	0% (0%*10.091%)
低温域LED調光率	0% (0%*10.091%)	~	2.522% (25%*10.091%)	~	5.0455% (50%*10.091%)	~	7.568% (75%*10.091%)	~	10.091% (100%*10.091%)
総LED調光率 (低温LED+高温LED)	10.091%	~	10.091%	~	10.091%	~	10.091%	~	10.091%

図 4-9 本サンプルでの物理的制限を満たした各調光率動作表

(2) LED 調色機能(XY)

本サンプルアプリケーションでは、1つの logical unit に対して 3つの LED (赤、緑、青)で調色を表現します。指定した XY 座標点を RGB 変換し、各 LED がどれくらいの割合で出力すればよいかを算出します。そこに現在の調光率をかけることで、各 LED の調光率の和は常に actual level で指定された調光率と一致するように設定します。各 LED の調光率は actual level によっても変化します。actual level が変化することにより各 LED の調光率は変化しますが、各 LED の調光率の比は変わりません。

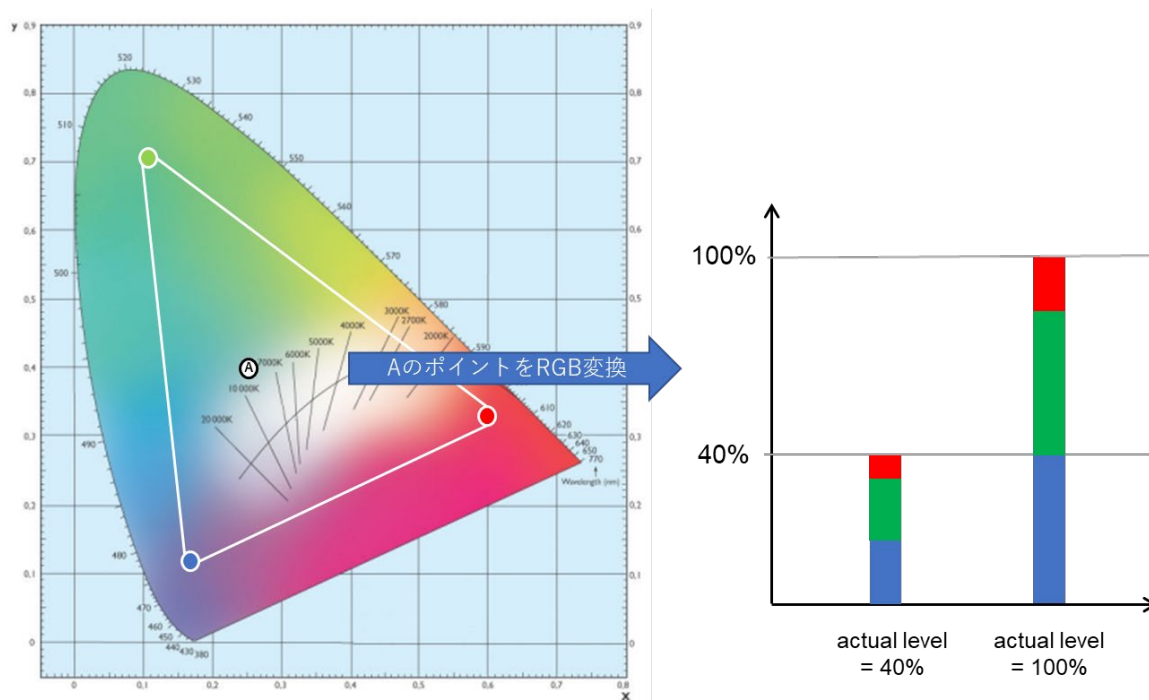


図 4-10 XY 座標点の RGB 変換と調光率の比

XY 調色機能では、XY 座標点が表示する色が使用する光源での「色表現が可能かどうか」を判断し、ライブラリへ通知する必要があります。例えば、使用する光源の配置位置により色が混ざらない範囲がある場合や、光源に出力制限があり光量が多く(又は足りず)色が表現できない場合等が挙げられます。色表現が行えない XY 座標点を示された場合、「色表現が可能なラインに沿って調色を行う」又は「色表現が行えない範囲となった時は現在の調色状態を保持し、再び色表現が可能な範囲となった時調色を再開する」動作を行う必要があります。

本サンプルアプリケーションでの「色表現が可能かどうか」の判定は 3 原色の各座標点で囲んだ三角形内部の色は全て「色表現可能」と判断をしています。各 LED の出力が物理的な最小調光率の PHM(physical minimum level)を下回ってしまう場合は PHM で出力するようにしています。

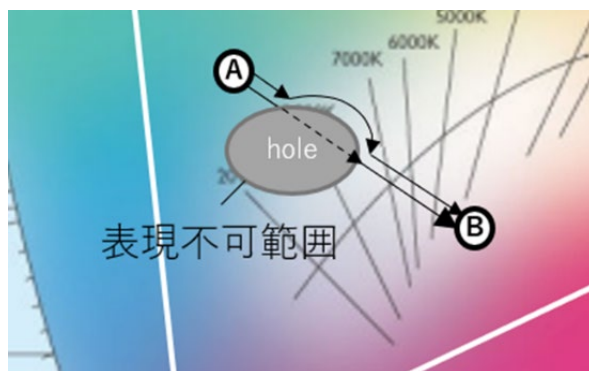
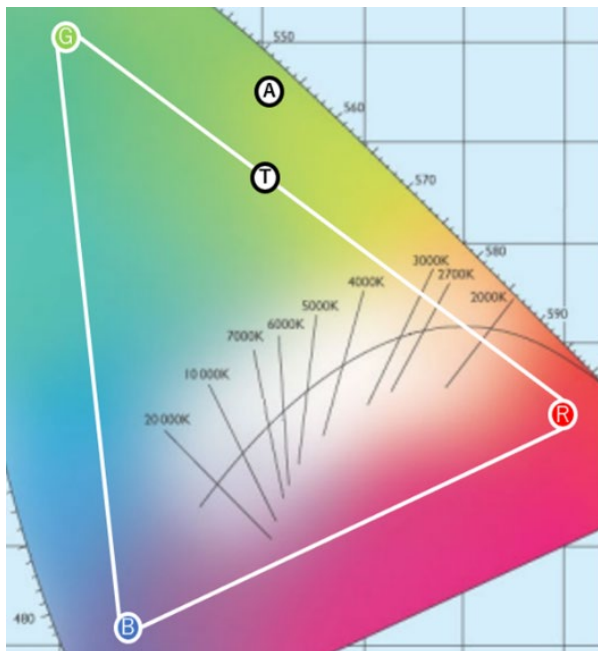


図 4-11 色表現が行えない範囲と動作例

また、取得する XY 座標点が常に「表現可能な範囲内」の座標点を取得できるよう、座標点の範囲外判断を行っています。これに伴い、使用している各 LED の座標点(primary_xy_coordinate_red / primary_xy_coordinate_green / primary_xy_coordinate_blue)を設定する必要があります。

本サンプルアプリケーションでは各 LED の座標点を以下のように設定し、範囲外判断を行えるようにしています。

- primary_xy_coordinate_red[DALI209_INDEX_XY_X] : 0xB43B (0.70)
- primary_xy_coordinate_red[DALI209_INDEX_XY_Y] : 0x4BC0 (0.29)
- primary_xy_coordinate_green[DALI209_INDEX_XY_X] : 0x29E0 (0.16)
- primary_xy_coordinate_green[DALI209_INDEX_XY_Y] : 0xB823 (0.71)
- primary_xy_coordinate_blue[DALI209_INDEX_XY_X] : 0x212E (0.12)
- primary_xy_coordinate_blue[DALI209_INDEX_XY_Y] : 0x11CB (0.06)



A: 範囲外の座標点
T: 範囲内の座標点

R: 赤色LEDが表す座標点
G: 緑色LEDが表す座標点
B: 青色LEDが表す座標点

図 4-12 各 LED 座標点による範囲と範囲外座標点/範囲内座標点の例

各設定値や「色表現可能かどうか」を判断する関数の詳細は RL78 ファミリー Control Gear ライブラリ ユーザーズマニュアル 調色(209)編を参照してください。

(3) LED 調色機能(RGBWAF)

本サンプルアプリケーションでは、1つの logical unit に対して3つの LED (赤、緑、青)で調色を表現します。RGBWAF 調色機能には White , Amber , Freecolour を使用することが可能ですが、RL78/G24 DC/DC LED Control Evaluation Board では赤、青、緑以外の灯具は実装していない為無効チャンネルとなります。本サンプルアプリケーションでは使用可能な LED のチャンネルを以下のように設定します。

- `rgbwaf_channels_present` : (`DALI209_ENABLE_CH_RGBWAF_R` | `DALI209_ENABLE_CH_RGBWAF_G` | `DALI209_ENABLE_CH_RGBWAF_B`)

使用するチャンネルは `enabled channels` の設定により動作中に無効/有効の切り替えを行うことが可能です。設定変更例を下表に示します。

表 4.7 enabled channels の内訳と設定例

bit	7	6	5	4	3	2	1	0	enabled channels
ch 内訳	-	-	Free	Amber	White	Blue	Green	Red	
変更例 1	-	-	無効	無効	無効	有効	無効	有効	0x05
	0	0	0	0	0	1	0	1	
変更例 2	-	-	無効	有効	無効	有効	有効	有効	0x07 *
	0	0	0	1	0	1	1	1	
* <code>rgbwaf_channels_present</code> で有効にしていないチャンネル(White, Amber, FreeColour)を <code>enabled channels</code> で有効設定しても反映されません。									

また、RGBWAF 調色機能には、Normalised colour control と Extended colour control の2つの制御方法があります。Normalised colour control では、`enabled channels` と colour value、調光率から調色率 (LightOutput)を算出します。Extended colour control では、`enabled channels` と colour value、調光率と灯具の電力比 (power ratio)から LightOutput を算出します。

Normalised colour control と Extended colour control は動作中に切り替えることが可能です。このため、制御方法と電力比 (power ratio)はあらかじめ設定しておく必要があります。電力比 (power ratio)の設定値に関する詳細は RL78 ファミリー Control Gear ライブラリ ユーザーズマニュアル 調色(209)編を参照してください。

取得した調色率(LightOutput)は各 LED の調光率に変換することで各 LED の調光率の和は常に actual level で指定された調光率と一致するようになっています。このため、LightOutput を算出する為に指定された actual level を調光率にしてライブラリへ通知する必要があります。

ただし、LightOutput には物理的な最小調光率の PHM(physical minimum level)は考慮されておりません。また、Tc や XY のように出力範囲の制限も無い為、PHM を下回る際は出力状態を保持する等のケアが必要です。

本サンプルアプリケーションでは各 LED の出力が PHM を下回ってしまう場合、PHM で出力するようにしています。

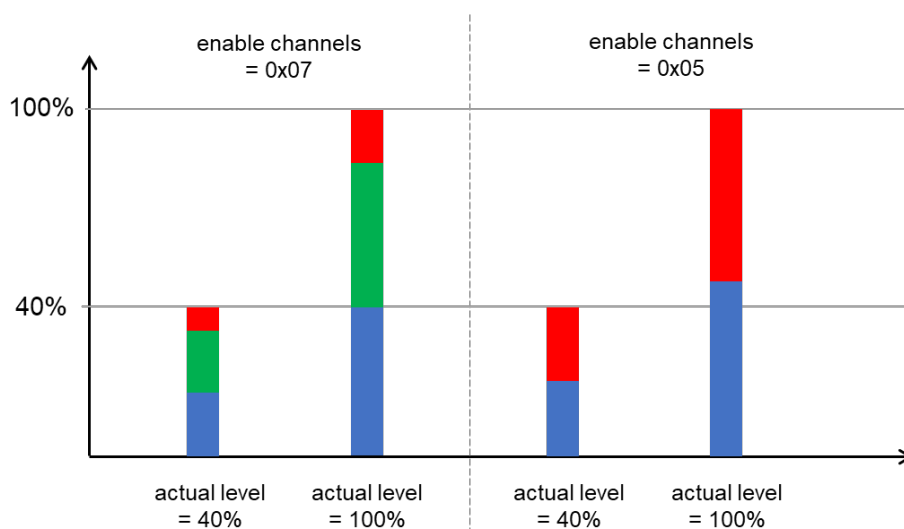


図 4-13 enabled channels と LightOutput の関係図

(4) colour type 切り替え機能(colour type change)

colour type change 機能は、現在有効な colour type を動作中に変更することを目的とします。使用する colour type を TC, XY, RGBWAF のうち、2 つ以上の組み合わせで設定されたときのみ有効な機能となります。

例えば、colour type を XY と Tc の 2 種類をサポートする設定で、現在有効な colour type は XY とします。colour type XY で動作中に colour type Tc を指定することで、有効な colour type が Tc に切り替わります。

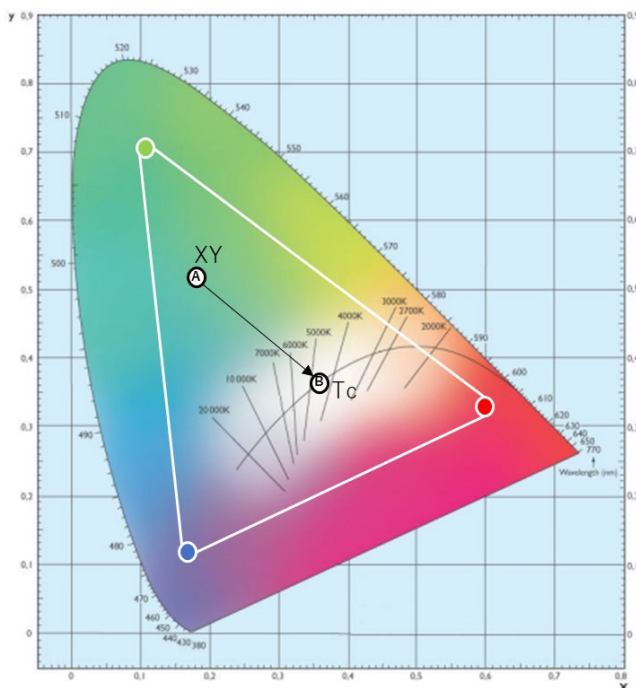


図 4-14 colour type change の動作例

colour type を変更する際に色をフェードさせる場合、各 colour type に対応した変換処理をライブラリへ登録する必要があります。フェードを必要としない場合、変換処理の登録は必要ありません。変換処理に関する詳細は RL78 ファミリー Control Gear ライブラリ ユーザーズマニュアル 調色(209)編を参照してください。

(5) デバッグ機能 (調光/調色モニタリング機能)

色(colour value)の変化を数値として表示するためにシリアル出力を行います。

100ms 毎に actual level と現在有効になっている調色方式の調色値(colour value)を確認しいずれかの値に変更があった場合にのみ PC 上のシリアル通信モニタに actual level 及び colour value を出力します。

- 色温度(Tc)による調色時の出力画面例

画面動作例：actual level=100%の状態、色温度を 150[mirek]→200[mirek]→250[mirek]と変化させた場合

```
(Level, Tc) = (0xFE, 0x0096)
(Level, Tc) = (0xFE, 0x00C8)
(Level, Tc) = (0xFE, 0x00FA)
.
.
.
```

図 4-15 シリアル出力画面(色温度 Tc)

- XY による調色時の出力画面例

画面動作例：actual level=100%の状態、XY 空間の座標点を (x, y) = (0.58, 0.32) →(0.2, 0.63) →(0.15, 0.13)と変化させた場合

```
(Level, coordinate_x, coordinate_y) = (0xFE, 0x963E, 0x52DD)
(Level, coordinate_x, coordinate_y) = (0xFE, 0x3362, 0xA21E)
(Level, coordinate_x, coordinate_y) = (0xFE, 0x2742, 0x225B)
.
.
.
```

図 4-16 シリアル出力画面(XY)

- RGBWAF による調色時の出力画面例

画面動作例：Normalised colour control 設定で以下のように変化させた場合

1. enabled channels= RED, GREEN, BLUE の出力が有効

RED=33.33%, GREEN=33.33%,BLUE=33.33%出力

2. enabled channels= RED, BLUE の出力が有効

RED=50%, BLUE=50%出力

3. enabled channels= RED, GREEN, BLUE の出力が有効

RED=22.22%, GREEN=33.33%,BLUE=44.44%出力

```
lightOutput(R, G, B, W, A, F) = (0x0D05, 0x0D05, 0x0D05, -, -, -)
lightOutput(R, G, B, W, A, F) = (0x1388, -, 0x1388, -, -, -)
lightOutput(R, G, B, W, A, F) = (0x08AE, 0x0D05, 0x115C, -, -, -)
.
.
.
```

図 4-17 シリアル出力画面(RGBWAF)

(6) デバッグ機能 (疑似異常発生機能)

209ed.1.0 では、4.3.3.1(8)章エラー処理と同様に該当操作キーの大文字/小文字にて故障状態の設定/解消を行います。キーに割り当てられている異常内容を下表に示します。

表 4.8 209ed.1.0 用操作キーと対応する故障内容一覧

項目	関連する故障	操作キー	説明
failure procedure	SHORT CIRCUIT	q, Q a, A z, Z	短絡回路の検出 q : 灯具(赤)の出力を短絡状態に設定 Q : 灯具(赤)の出力の短絡状態を解消 a : 灯具(緑)の出力を短絡状態に設定 A : 灯具(緑)の出力の短絡状態を解消 z : 灯具(青)の出力を短絡状態に設定 Z : 灯具(青)の出力の短絡状態を解消
	OPEN CIRCUIT	w, W s, S x, X	開回路の検出 w : 灯具(赤)の出力を開回路状態に設定 W : 灯具(赤)の出力の開回路状態を解消 s : 灯具(緑)の出力を開回路状態に設定 S : 灯具(緑)の出力の開回路状態を解消 x : 灯具(青)の出力を開回路状態に設定 X : 灯具(青)の出力の開回路状態を解消
	failure status の全ての故障	C	すべての異常をクリア

注意 : 本アプリケーションノートでは、エラー処理を確認するために、疑似的な発生機能を実装しています。実際の製品に実装する際は、7章参考ドキュメントの DALI 規格書 : 102ed.2.1 と 209ed.1.0 を参照して実装してください。

4.4 キャリブレーション

外部デバイス(フォトカプラ)の特性、劣化により DALI 受信波形や DALI 送信波形の Duty が変化する場合、RXWR1 レジスタや TXWR1 レジスタを使用して該当波形の補正が必要になります。

4.4.1 DALI 受信波形補正

RXWR1 レジスタを設定することで受信波形の Low レベルもしくは High レベルを伸長することが可能になり、結果的に受信波形の Duty 比を補正することができます。

本サンプルにおいての DALI 受信波形を補正する手順を以下に示します。

1. RL78/G24 DC/DC LED Control Evaluation Board と、RL78/G24 DC/DC LED Control Evaluation Board を操作する Application Controller を接続後、各機器に電源を供給します。
2. サンプルプロジェクト r_main.c の「RXD_CALIBRATION」マクロを有効にしてビルド後ダウンロードし、デバッグ実行します。
3. オシロスコープのチャンネルを下記表のように設定します。

表 4-13 オシロスコープ設定例

チャンネル	接続先	設定
CH1	DALI 通信バス	電圧レンジ: 5V/div
CH2	RL78/G24 DC/DC LED Control Evaluation Board J1-2 (RXDMON レジスタの同期出力)	電圧レンジ: 2V/div
その他の設定		
<ul style="list-style-type: none"> ・時間レンジは 100us/div 程度を推奨。 ・チャンネル 2 に対し立下り 2.5V 付近でトリガする設定とする。 		

4. DALI Master Controller GUI から QUERY GEAR PRESENT コマンドを Broadcast 送信します。
5. オシロスコープのトリガで取得した CH1 の波形より下記の一点を測定します。
6. 立下り時の 8V 地点から立上り時の 8V 地点までの half bit の Low 幅
7. オシロスコープのトリガで取得した CH2 の波形より下記の一点を測定します。
8. 立下り時の 8V 地点から立上り時の 8V 地点までの half bit の Low 幅
9. 測定結果①を基準に測定結果②の誤差を求め、その誤差が可能な限り 0 に近づくように補正する値を RXWR1 レジスタに設定します。
10. サンプルプロジェクト「r_main.c」の「RXD_CALIBRATION」マクロを無効にしてビルド/書き込みを行う。

また、上記の RXWR1 レジスタ設定で受信波形が補整しきれない場合は DALI ビットタイミング違反しきい値レジスタを使用してタイムスロットの調整を行って下さい。

RXWR1 レジスタの設定方法は RL78/G24 ユーザーズマニュアル ハードウェア編(R01UH0961)を参照してください。

4.4.2 DALI 送信波形補正

1. RL78/G24 DC/DC LED Control Evaluation Board と、RL78/G24 DC/DC LED Control Evaluation Board を操作する Application Controller を接続後、各機器に電源を供給します。
2. サンプルプロジェクトをビルド後、デバッグ実行します。
3. オシロスコープのチャンネルを下記表のように設定します。

表 4-13 オシロスコープ設定例

チャンネル	接続先	設定
CH1	DALI 通信バス	電圧レンジ : 5V/div
CH2	ターゲットボードの TP5(DALI_TX)	電圧レンジ : 2V/div
その他設定 ・ 時間レンジは 100us/div 程度を推奨。 ・ チャンネル 2 に対し立下り 2.5V 付近でトリガする設定とする。		

4. DALI Master Controller GUI から QUERY GEAR PRESENT コマンドを Broadcast 送信します。
5. オシロスコープのトリガで取得した CH1 の波形より下記の一点を測定します。
 - ① 立下り時の 8V 地点から立上り時の 8V 地点までの half bit の Low 幅
6. ①の測定結果が可能な限り half bit の Typical 値に近づくように補正する値を TXWR1 レジスタに設定します。

TXWR1 レジスタの設定方法は RL78/G24 ユーザーズマニュアル ハードウェア編(R01UH0961)を参照してください。

4.5 ソフトウェア構成

各サンプルアプリケーションのソフトウェア構成を記載します。

4.5.1 フォルダ構成

本サンプルアプリケーションはLEDの調光制御をCPU上で行うものとFAA(Flexible Application Accelerator)上で行うものの2種類を提供しています。それぞれのフォルダ構成を以降の章に示します。

4.5.1.1 CPUでLED調光制御を行うサンプルアプリケーション

表 4.9 フォルダ構成 (1/3)

フォルダ、ファイル名	説明
DALI102_20X_sample	サンプルアプリケーションのフォルダ
\App<DIR>	アプリケーションプログラム格納フォルダ
r_cg.c	Control Gear モジュールのソースファイル
r_cg.h	Control Gear モジュールのヘッダファイル
r_common.h	共通定義ヘッダファイル
r_debug.c	デバッグモジュールのソースファイル
r_debug.h	デバッグモジュールのヘッダファイル
r_lamp.c ^{*1}	調光モジュールのソースファイル ^{*1}
r_lamp.h ^{*1}	調光モジュールのヘッダファイル ^{*1}
r_lamp_tc.c ^{*2}	209ed.1.0の色温度(Tc)対応調光モジュールのソースファイル ^{*2}
r_lamp_tc.h ^{*2}	209ed.1.0の色温度(Tc)対応調光モジュールのヘッダファイル ^{*2}
r_lamp_xy.c ^{*2}	209ed.1.0のXY対応調光モジュールのソースファイル ^{*2}
r_lamp_xy.h ^{*2}	209ed.1.0のXY対応調光モジュールのヘッダファイル ^{*2}
r_lamp_rgbwaf.c ^{*2}	209ed.1.0のRGBWAF対応調光モジュールのソースファイル ^{*2}
r_lamp_rgbwaf.h ^{*2}	209ed.1.0のRGBWAF対応調光モジュールのヘッダファイル ^{*2}
r_main.c	メインソースファイル
r_memory_bank.c	メモリバンクモジュールのソースファイル
r_memory_bank.h	メモリバンクモジュールのヘッダファイル
r_memory_banks.c	複数のメモリバンクを管理するモジュールのソースファイル
r_memory_banks.h	複数のメモリバンクを管理するモジュールのヘッダファイル
r_nvm.c	データ・フラッシュ・ライブラリ対応 NVM モジュールのソースファイル
r_nvm.h	データ・フラッシュ・ライブラリ対応 NVM モジュールのヘッダファイル
r_random.c	乱数生成モジュールのソースファイル
r_random.h	乱数生成モジュールのヘッダファイル
r_unit0_memory_bank.c	メモリバンクの unit0 定義用ソースファイル
r_unit0_memory_bank.h	メモリバンクの unit0 定義用ヘッダファイル
r_unit1_memory_bank.c	メモリバンクの unit1 定義用ソースファイル
r_unit1_memory_bank.h	メモリバンクの unit1 定義用ヘッダファイル
r_unit2_memory_bank.c	メモリバンクの unit2 定義用ソースファイル
r_unit2_memory_bank.h	メモリバンクの unit2 定義用ヘッダファイル
\Driver<DIR>	ドライバプログラム格納フォルダ
Config_DALI.c	DALI 周辺機能設定用ソースファイル
Config_DALI.h	DALI 周辺機能設定用ヘッダファイル
Config_DALI_user.c	DALI 周辺機能設定ユーザ用ソースファイル
r_cg_dali.h	DALI 周辺機能定義用ヘッダファイル
r_dali101.c	DALI101ed.2.0 対応ドライバのソースファイル
r_dali101.h	DALI101ed.2.0 対応ドライバのヘッダファイル
r_dali101_bft.c	DALI101ed.2.0 対応ドライバ(送信)のソースファイル
r_dali101_bft.h	DALI101ed.2.0 対応ドライバ(送信)のヘッダファイル
r_dali101_rx.c	DALI101ed.2.0 対応ドライバ(受信)のソースファイル
r_dali101_rx.h	DALI101ed.2.0 対応ドライバ(受信)のヘッダファイル
r_led.c	調光ドライバのソースファイル
r_led.h	調光ドライバのヘッダファイル
r_led1.c	LED1 用調光ドライバのソースファイル
r_led1.h	LED1 用調光ドライバのヘッダファイル
r_led2.c	LED2 用調光ドライバのソースファイル
r_led2.h	LED2 用調光ドライバのヘッダファイル
r_led3.c	LED3 用調光ドライバのソースファイル
r_led3.h	LED3 用調光ドライバのヘッダファイル

表 4.10 フォルダ構成 (2/3)

フォルダ、ファイル名	説明
DALI102_20X_sample	サンプルアプリケーションのフォルダ
\Driver<DIR>	ドライバプログラム格納フォルダ
r_port.c	PORT 用ソースファイル
r_port.h	PORT 用ヘッダファイル
r_trng.c	TRNG 用ソースファイル
r_trng.h	TRNG 用ヘッダファイル
\DefaultBuild<DIR> (CC-RL)	プロジェクト出力ファイル格納フォルダ (CC-RL 開発環境のみ)
DALIXXX_sample.mot	ファームウェアへの書き込みに使用するモトローラ・S タイプ・フォーマット形式のファイル
\Debug\Exe<DIR> (IAR)	プロジェクト出力ファイル格納フォルダ (IAR 開発環境のみ)
DALIXXX_sample.mot	ファームウェアへの書き込みに使用するモトローラ・S タイプ・フォーマット形式のファイル
\Library<DIR>	ライブラリ格納フォルダ
\DALI102<DIR>	DALI102ed.2.0 ライブラリ格納フォルダ
r_dali_102_cc_gen2_v1_01.lib (CC-RL)	DALI102ed.2.0 対応ライブラリファイル
r_dali_102_iar_gen2_v1_01.a (IAR)	
r_dali102_api.h	DALI102ed.2.0 対応ライブラリのヘッダファイル
r_dali102_xxxx.h	DALI102ed.2.0 対応ライブラリのヘッダファイル (ライブラリ内にて使用)
\DALI207<DIR>* ¹	DALI207ed.1.0 ライブラリ格納フォルダ* ¹
r_dali_207_cc_gen2_v1_01.lib (CC-RL)	DALI207ed.1.0 対応ライブラリファイル* ¹
r_dali_207_iar_gen2_v1_01.a (IAR)	
r_dali207_api.h	DALI207ed.1.0 対応ライブラリのヘッダファイル* ¹
r_dali207_xxxx.h	DALI207ed.1.0 対応ライブラリのヘッダファイル (ライブラリ内にて使用)* ¹
\DALI209<DIR>* ²	DALI209ed.1.0 ライブラリ格納フォルダ* ²
r_dali_209_cc_gen2_v2_00.lib (CC-RL)	DALI209ed.1.0 対応ライブラリファイル* ²
r_dali_209_iar_gen2_v2_00.a (IAR)	
r_dali209_api.h	DALI209ed.1.0 対応ライブラリのヘッダファイル* ²
r_dali209_xxxx.h	DALI209ed.1.0 対応ライブラリのヘッダファイル (ライブラリ内にて使用)* ²
\RFD<DIR>	ルネサスエレクトロニクス製データ・フラッシュ・ライブラリを格納しているフォルダです。 ※このディレクトリ以下はルネサス エレクトロニクスからデータ・フラッシュ・ライブラリをダウンロードして、ライブラリファイルを登録していただく必要があります。詳細は 4.1.3 章、4.1.6 章を参照してください。
\src	スマート・コンフィグレータ生成ファイル格納フォルダ
smc_gen	スマート・コンフィグレータ生成フォルダ

表 4.11 フォルダ構成 (3/3)

フォルダ、ファイル名	説明
DALI102_20X_sample	サンプルアプリケーションのフォルダ
\Utility<DIR>	共通プログラム格納フォルダ
r_timer16.c	タイマモジュールのソースファイル
r_timer16.h	タイマモジュールのヘッダファイル
r_user_typedefs.h	タイプ定義ヘッダファイル

※1 : 207 サンプルアプリケーションにのみ格納されています。

※2 : 209 サンプルアプリケーションにのみ格納されています。

4.5.1.2 FAA で LED 調光制御を行うサンプルアプリケーション

表 4.12 フォルダ構成 (1/3)

フォルダ、ファイル名	説明
DALI102_20X_sample	サンプルアプリケーションのフォルダ
\App<DIR>	アプリケーションプログラム格納フォルダ
r_cg.c	Control Gear モジュールのソースファイル
r_cg.h	Control Gear モジュールのヘッダファイル
r_common.h	共通定義ヘッダファイル
r_debug.c	デバッグモジュールのソースファイル
r_debug.h	デバッグモジュールのヘッダファイル
r_lamp.c ^{*1}	調光モジュールのソースファイル ^{*1}
r_lamp.h ^{*1}	調光モジュールのヘッダファイル ^{*1}
r_lamp_tc.c ^{*2}	209ed.1.0 の色温度 (Tc) 対応調光モジュールのソースファイル ^{*2}
r_lamp_tc.h ^{*2}	209ed.1.0 の色温度 (Tc) 対応調光モジュールのヘッダファイル ^{*2}
r_lamp_xy.c ^{*2}	209ed.1.0 の XY 対応調光モジュールのソースファイル ^{*2}
r_lamp_xy.h ^{*2}	209ed.1.0 の XY 対応調光モジュールのヘッダファイル ^{*2}
r_lamp_rgbwaf.c ^{*2}	209ed.1.0 の RGBWAF 対応調光モジュールのソースファイル ^{*2}
r_lamp_rgbwaf.h ^{*2}	209ed.1.0 の RGBWAF 対応調光モジュールのヘッダファイル ^{*2}
r_main.c	メインソースファイル
r_memory_bank.c	メモリバンクモジュールのソースファイル
r_memory_bank.h	メモリバンクモジュールのヘッダファイル
r_memory_banks.c	複数のメモリバンクを管理するモジュールのソースファイル
r_memory_banks.h	複数のメモリバンクを管理するモジュールのヘッダファイル
r_nvm.c	データ・フラッシュ・ライブラリ対応 NVM モジュールのソースファイル
r_nvm.h	データ・フラッシュ・ライブラリ対応 NVM モジュールのヘッダファイル
r_random.c	乱数生成モジュールのソースファイル
r_random.h	乱数生成モジュールのヘッダファイル
r_unit0_memory_bank.c	メモリバンクの unit0 定義用ソースファイル
r_unit0_memory_bank.h	メモリバンクの unit0 定義用ヘッダファイル
r_unit1_memory_bank.c	メモリバンクの unit1 定義用ソースファイル
r_unit1_memory_bank.h	メモリバンクの unit1 定義用ヘッダファイル
r_unit2_memory_bank.c	メモリバンクの unit2 定義用ソースファイル
r_unit2_memory_bank.h	メモリバンクの unit2 定義用ヘッダファイル
\Driver<DIR>	ドライバプログラム格納フォルダ
Config_DALI.c	DALI 周辺機能設定用ソースファイル
Config_DALI.h	DALI 周辺機能設定用ヘッダファイル
Config_DALI_user.c	DALI 周辺機能設定ユーザ用ソースファイル
r_cg_dali.h	DALI 周辺機能定義用ヘッダファイル
r_dali101.c	DALI101ed.2.0 対応ドライバのソースファイル
r_dali101.h	DALI101ed.2.0 対応ドライバのヘッダファイル
r_dali101_bft.c	DALI101ed.2.0 対応ドライバ (送信) のソースファイル
r_dali101_bft.h	DALI101ed.2.0 対応ドライバ (送信) のヘッダファイル
r_dali101_rx.c	DALI101ed.2.0 対応ドライバ (受信) のソースファイル
r_dali101_rx.h	DALI101ed.2.0 対応ドライバ (受信) のヘッダファイル
r_led.c	調光ドライバのソースファイル
r_led.h	調光ドライバのヘッダファイル

表 4.13 フォルダ構成 (2/3)

フォルダ、ファイル名	説明
DALI102_20X_sample	サンプルアプリケーションのフォルダ
\Driver<DIR>	ドライバプログラム格納フォルダ
r_port.c	PORT 用ソースファイル
r_port.h	PORT 用ヘッダファイル
r_trng.c	TRNG 用ソースファイル
r_trng.h	TRNG 用ヘッダファイル
\DefaultBuild<DIR> (CC-RL)	プロジェクト出力ファイル格納フォルダ (CC-RL 開発環境のみ)
DALIXXX_sample.mot	ファームウェアへの書き込みに使用するモトローラ・S タイプ・フォーマット形式のファイル
\Debug\Exe<DIR> (IAR)	プロジェクト出力ファイル格納フォルダ (IAR 開発環境のみ)
DALIXXX_sample.mot	ファームウェアへの書き込みに使用するモトローラ・S タイプ・フォーマット形式のファイル
\Library<DIR>	ライブラリ格納フォルダ
\DALI102<DIR>	DALI102ed.2.0 ライブラリ格納フォルダ
r_dali_102_cc_gen2_v1_01.lib (CC-RL)	DALI102ed.2.0 対応ライブラリファイル
r_dali_102_iar_gen2_v1_01.a (IAR)	
r_dali102_api.h	DALI102ed.2.0 対応ライブラリのヘッダファイル
r_dali102_xxxx.h	DALI102ed.2.0 対応ライブラリのヘッダファイル (ライブラリ内にて使用)
\DALI207<DIR>* ¹	DALI207ed.1.0 ライブラリ格納フォルダ* ¹
r_dali_207_cc_gen2_v1_01.lib (CC-RL)	DALI207ed.1.0 対応ライブラリファイル* ¹
r_dali_207_iar_gen2_v1_01.a (IAR)	
r_dali207_api.h	DALI207ed.1.0 対応ライブラリのヘッダファイル* ¹
r_dali207_xxxx.h	DALI207ed.1.0 対応ライブラリのヘッダファイル (ライブラリ内にて使用)* ¹
\DALI209<DIR>* ²	DALI209ed.1.0 ライブラリ格納フォルダ* ²
r_dali_209_cc_gen2_v2_00.lib (CC-RL)	DALI209ed.1.0 対応ライブラリファイル* ²
r_dali_209_iar_gen2_v2_00.a (IAR)	
r_dali209_api.h	DALI209ed.1.0 対応ライブラリのヘッダファイル* ²
r_dali209_xxxx.h	DALI209ed.1.0 対応ライブラリのヘッダファイル (ライブラリ内にて使用)* ²
\RFID<DIR>	ルネサスエレクトロニクス製データ・フラッシュ・ライブラリを格納しているフォルダです。 ※このディレクトリ以下はルネサス エレクトロニクスからデータ・フラッシュ・ライブラリをダウンロードして、ライブラリファイルを登録していただく必要があります。詳細は 4.1.3 章、4.1.6 章を参照してください。
\src	スマート・コンフィグレート生成ファイル格納フォルダ
smc_gen	スマート・コンフィグレート生成フォルダ

表 4.14 フォルダ構成 (3/3)

フォルダ、ファイル名	説明
DALI102_20X_sample	サンプルアプリケーションのフォルダ
\Utility<DIR>	共通プログラム格納フォルダ
r_timer16.c	タイマモジュールのソースファイル
r_timer16.h	タイマモジュールのヘッダファイル
r_user_typedefs.h	タイプ定義ヘッダファイル

※1 : 207 サンプルアプリケーションにのみ格納されています。

※2 : 209 サンプルアプリケーションにのみ格納されています。

4.5.2 オプション・バイトの設定一覧

各サンプルアプリケーションのオプション・バイト設定を以下に示します。

表 4.15 オプション・バイト設定一覧

アドレス	設定値	内容
000C0H / 010C0H	01110001B	ウォッチドッグ・タイマ動作許可 (リセット解除後、カウント開始)
000C1H / 010C1H	11111011B	LVD リセット・モード (立ち下がり : 2.97V 立ち上がり : 2.91V)
000C2H / 010C2H	11101010B	HS モード、高速オンチップ・オシレータ : 8MHz
000C3H / 010C3H	10000100B	オンチップ・デバッグ許可

4.5.3 フローチャート

4.5.3.1 メインフロー

各サンプルアプリケーションのメインフローを以下に示します。

初期化処理終了後、Control Gear 処理を実行します。

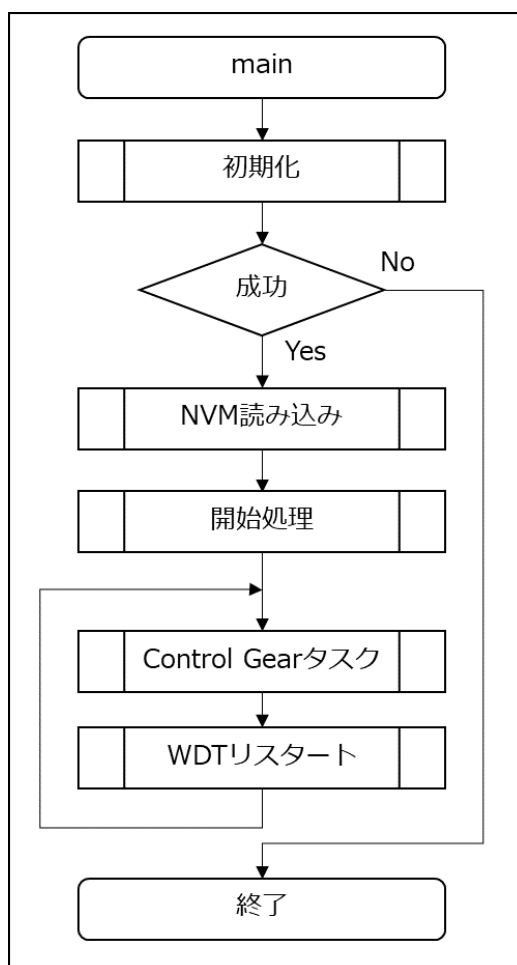


図 4-18 メインフロー図

4.5.3.2 初期化

初期化処理のフローを以下に示します。

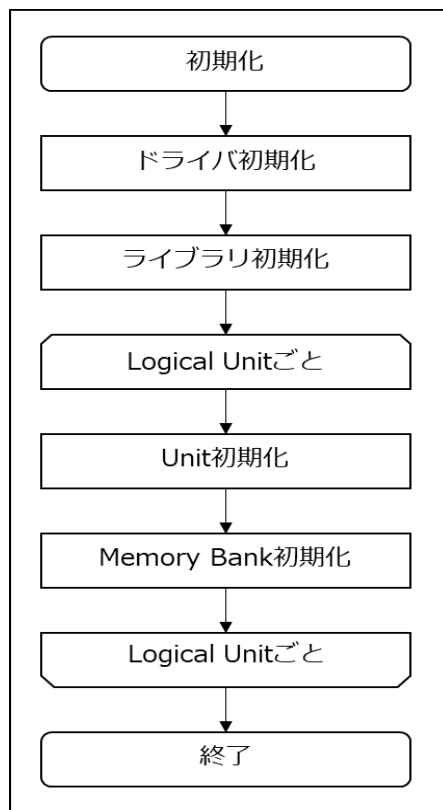


図 4-19 初期化フロー図

4.5.3.3 NVM 読み込み

初期化処理後に実施する NVM 読み込み処理のフローを以下に示します。

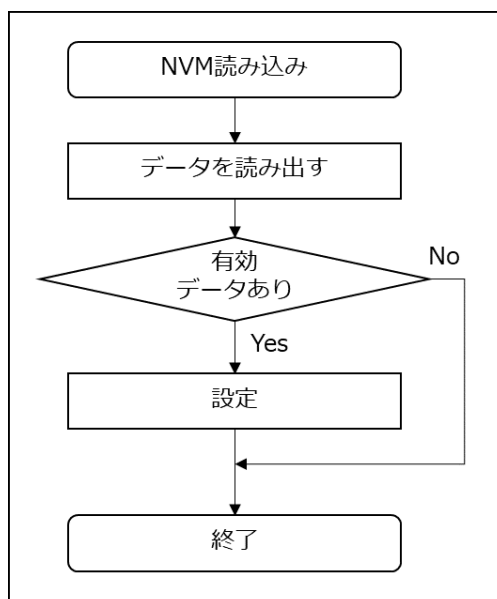


図 4-20 NVM 読み込み処理フロー図

4.5.3.4 開始処理

Control Gear 動作開始処理のフローを以下に示します。

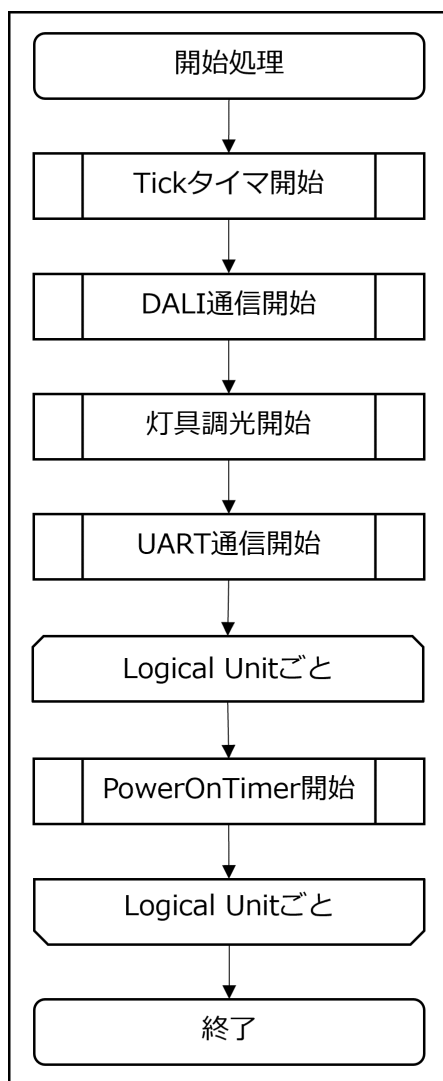


図 4-21 開始処理フロー図

4.5.3.5 Control Gear タスク処理

Control Gear タスク処理のフローを以下に示します。

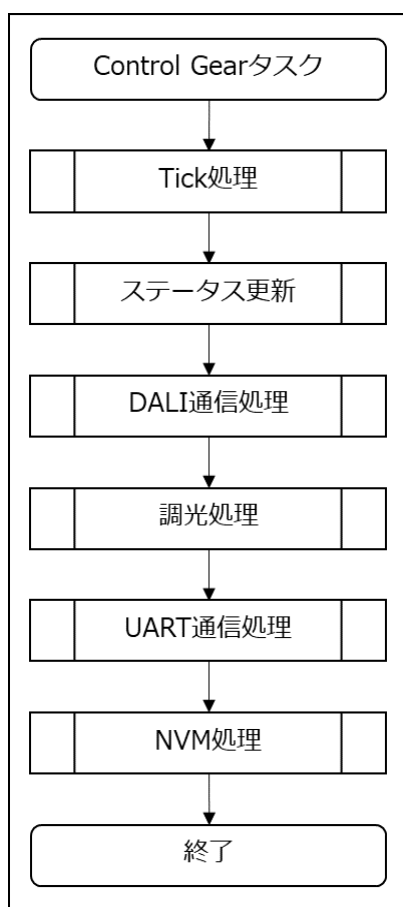


図 4-22 Control Gear タスク処理フロー図

5. 付録

5.1 DALI ドライバ

本サンプルアプリケーションに実装されている DALI ドライバの概要を以下に示します。

5.1.1 Backward Frame Transmitter (BFT)

DALI モジュールを使用して実装します。Forward Frame を受信してから Backward Frame を送信するまでの settling time (5.5ms~10.5ms)は精度を必要としないため、1ms 単位のソフトウェアタイマで送信までの待ち時間を調整します。

5.1.1.1 破損 Backward Frame の再現

インターバル・タイマを使用して一定時間(1300us~2000us)DALI バスを強制的に Low にして破損 Backward Frame を再現します。

注意：破損 Backward Frame は複数 logical unit の応答が異なった場合に送信します。単一の logical unit の場合は実装不要です。

5.1.2 Receiver (Rx)

DALI モジュール、インターバル・タイマを使用して実装します。

DALI の送受信は、DALI モジュールを使用しており、System Failure や settling time の検出は TAU 機能を使用しています。

5.1.3 周辺機能

DALI ドライバで使用する周辺機能を以下に示します。

表 5.1 DALI ドライバで使した周辺機能

機能	用途
P02/DALITxD0	送信処理：DALI 送信機能
P03/DALIRxD0	受信処理：DALI 受信機能
TAU0-CH2	受信処理：インターバル・タイマ(100us)
TAU0-CH3	送信処理：インターバル・タイマ(1458us)
DALI	DALI モジュール

5.1.4 API 関数

DALI ドライバの API 関数一覧と各関数の概要を以下に示します。

表 5.2 DALI ドライバの API 関数一覧

関数名	説明
R_DALI101_Init	DALI ドライバの初期化
R_DALI101_Start	DALI ドライバの起動
R_DALI101_Tick1ms	DALI ドライバの内部動作を 1ms 進行
R_DALI101_Task	DALI ドライバのタスク
R_DALI101_GetReceivedFrame	Frame の受信
R_DALI101_SystemFailureIsDetected	System Failure の検出
R_DALI101_SendBackwardFrame	Backward Frame の送信

表 5.3 R_DALI101_Init

関数	R_DALI101_Init	DALI ドライバの初期化
引数	void	-
戻り値	void	-

表 5.4 R_DALI101_Start

関数	R_DALI101_Start	DALI ドライバの起動
引数	void	-
戻り値	void	-

表 5.5 R_DALI101_Tick1ms

関数	R_DALI101_Tick1ms	DALI ドライバの内部動作を 1ms 進行
引数	void	-
戻り値	void	-

表 5.6 R_DALI101_Task

関数	R_DALI101_Task	DALI ドライバのタスク
引数	void	-
戻り値	void	-

表 5.7 R_DALI101_GetReceivedFrame

関数	R_DALI101_GetReceivedFrame	Frame の受信
引数	uint32_t * p_data	Frame data
	bool * p_twice	Send-twice
戻り値	int8_t	<0 : Error Frame 0 : 受信なし >0 : 受信 Frame のビット数

表 5.8 R_DALI101_SystemFailureIsDetected

関数	R_DALI101_SystemFailureIsDetected	System Failure の検出
引数	void	-
返り値	bool	true : 検出 false : 未検出

表 5.9 R_DALI101_SendBackwardFrame

関数	R_DALI101_SendBackwardFrame	Backward Frame の送信
引数	uint8_t data	Backward Frame データ
返り値	void	-

6. 注意事項

ユーザ規定部はボード RL78/G24 DC/DC LED Control Evaluation Board に合わせた構成・動作となっています。

- 各サンプルアプリケーションが提供する動作はあくまでもサンプルです。ユーザ規定部は本来ユーザが動作を決定し実装する部分となりますので、使用目的に合わせた設計変更・評価を行ってください。
- 異常系の動作はユーザの使用目的・動作に合わせ別途検討してください。

7. 参考ドキュメント

RL78/G24 ユーザーズマニュアル ハードウェア編 (R01UH0961J)

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

CS+ V8.10.00 統合開発環境ユーザーズマニュアル プロジェクト操作編(R20UT5299)

DALI マスタコントローラ GUI ユーザーズマニュアル (R20UT0715)

RL78/G24 DC/DC LED Control Evaluation Board ユーザーズ・マニュアル (R20UT5371)

RL78 ファミリ Renesas Flash Driver RL78 Type01 ユーザーズマニュアル (R20UT4830)

(最新版をルネサスエレクトロニクスホームページから入手してください。)

テクニカルアップデート／テクニカルニュース

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

DALI 規格書

IEC 62386-101:2014+AMD1:2018 CSV Consolidated version (ed.2.1)

IEC 62386-102:2014+AMD1:2018 CSV Consolidated version (ed.2.1)

IEC 62386-207:2009 (ed.1.0)

IEC 62386-209:2011 (ed.1.0)

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	Sep. 01 2023	-	初版

製品ご使用上の注意事項

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

1. 静電気対策

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

2. 電源投入時の処置

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

3. 電源オフ時における入力信号

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

4. 未使用端子の処理

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

5. クロックについて

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

6. 入力端子の印加波形

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

7. リザーブアドレス（予約領域）のアクセス禁止

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

8. 製品間の相違について

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

ご注意書き

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

標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット等

高品質水準： 輸送機器（自動車、電車、船舶等）、交通制御（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、これに限られません。）から生じる責任を負うものではありません。当社は、当社製品または当社製品が使用されたあらゆるシステムが、不正な改変、攻撃、ウイルス、干渉、ハッキング、データの破壊または窃盗その他の不正な侵入行為（「脆弱性問題」といいます。）によって影響を受けないことを保証しません。当社は、脆弱性問題に起因したまたはこれに関連して生じた損害について、一切責任を負いません。また、法令において認められる限りにおいて、本資料および当社ハードウェア/ソフトウェア製品について、商品性および特定目的との合致に関する保証ならびに第三者の権利を侵害しないことの保証を含め、明示または黙示のいかなる保証も行いません。
8. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
10. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
11. 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
12. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものいたします。
13. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
14. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。

注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

(Rev.5.0-1 2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

www.renesas.com

お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

www.renesas.com/contact/

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。すべての商標および登録商標は、それぞれの所有者に帰属します。