

RL78/G1D ビーコンスタック

R01AN3045JJ0220

基本機能 サンプルプログラム

Rev.2.20
2020.06.19

要旨

本ソフトウェアは Bluetooth® Low Energy に対応した RL78/G1D で動作し、ビーコンとして情報発信するための Advertising と、Advertising パケットを受信するための Scanning、RF 特性を評価するための Direct Test Mode(DTM)を実行するサンプルプログラムです。

本サンプルプログラムは RL78/G1D 評価ボードの DIP スイッチの設定によって、ビーコンアプリケーションとスキャンアプリケーション、DTM アプリケーションのいずれかを実行します。ビーコンアプリケーションは、Non-connectable Undirected Advertising パケットの送信と Scannable Undirected Advertising パケットの送信を、RL78/G1D 評価ボードのスイッチの押下により交互に切り替えて実行、またはホストマシンから ASCII 形式 UART 通信コマンドを送信して実行します。スキャンアプリケーションは Advertising パケットを受信し、受信結果を UART から出力します。DTM アプリケーションは UART 制御により RF 受信テストと RF 送信テストを実行します。

動作確認デバイス

RL78/G1D 評価ボード(RTK0EN0001D01001BZ)

関連資料

資料名	資料番号	
	和文	英文
RL78/G1D		
ユーザーズマニュアル ハードウェア編	R01UH0515J	R01UH0515E
RL78/G1D 評価ボード		
ユーザーズマニュアル	R30UZ0048J	R30UZ0048E
E1 エミュレータ		
ユーザーズマニュアル	R20UT0398J	R20UT0398E
ユーザーズマニュアル別冊 (RL78 接続時の注意事項)	R20UT1994J	R20UT1994E
Renesas Flash Programmer V3.02 フラッシュ書き込みソフトウェア		
ユーザーズマニュアル	R20UT3841J	R20UT3841E
CC-RL コンパイラ		
ユーザーズマニュアル	R20UT3123J	R20UT3123E
RL78/G1D ビーコンスタック		
ユーザーズマニュアル	R01UW0171J	R01UW0171E

目次

1. 概要	5
1.1 ビーコンアプリケーション	6
1.2 ビーコンアプリケーション - ASCII コマンド制御	6
1.3 スキャンアプリケーション	7
1.4 DTM アプリケーション	7
2. 動作環境	8
3. ファイル構成	9
4. 操作方法	11
4.1 ファームウェアのビルド	12
4.1.1 CS+ for CC 使用時	12
4.1.2 CS+ for CC 使用時 (ビーコンアプリケーション-ASCII コマンド制御)	12
4.1.3 e ² studio 使用時	13
4.1.4 e ² studio 使用時 (ビーコンアプリケーション-ASCII コマンド制御)	13
4.2 ファームウェアの書き込み	14
4.3 動作確認	16
4.3.1 Advertising パケット送信の確認	17
4.3.2 Advertising パケット送信の確認 (ビーコンアプリケーション-ASCII コマンド制御)	18
4.3.3 Advertising パケット受信の確認	20
4.4 RF 特性の評価	22
4.5 消費電流の測定	23
4.5.1 測定環境の構築	23
4.5.2 評価ボード設定	24
4.5.3 消費電流の測定	25
5. 仕様	26
5.1 ビーコンアプリケーション	26
5.1.1 デフォルト Advertising 動作設定	26
5.2 ビーコンアプリケーション - ASCII コマンド制御	28
5.2.1 デフォルト Advertising 動作設定	28
5.2.2 ASCII 形式 UART 通信コマンド一覧	28
5.2.3 ASCII 形式 UART 通信コマンド	29
5.3 スキャンアプリケーション	44
5.3.1 ASCII 形式 UART 通信コマンド	44
5.3.2 バイナリ形式 UART 通信コマンド	49
5.4 DTM アプリケーション	54
5.4.1 Direct Test Mode	54
5.5 フラッシュメモリへのアクセス	56
5.5.1 コードフラッシュメモリへのアクセス	56
5.5.2 コードフラッシュメモリへのアクセス (ビーコンアプリケーション-ASCII コマンド制御)	57
5.6 使用ハードウェアリソース	58
5.7 コンパイラ	59
5.8 メモリモデル	59

5.9	プログラムサイズ	59
5.10	アドレスマップ	60
6.	設定	63
6.1	ハードウェア設定	63
6.1.1	MCU メイン・システム・クロック周波数	64
6.1.2	RF 送受信動作	65
6.1.3	RF 内蔵 DC-DC コンバータ	65
6.1.4	RF スロー・クロック供給源	65
6.1.5	RF 内蔵オシレータのキャリブレーション	65
6.1.6	RF 基準クロックの分周出力	66
6.1.7	RF 基準クロックの発振安定時間	66
6.1.8	システム動作設定アドレス	66
6.1.9	エナジーハーベスティング向けハードウェア設定	67
6.2	アプリケーション設定	68
6.2.1	アプリケーション選択設定	68
6.2.2	システム動作設定	69
6.2.3	システム動作設定 (ビーコンアプリケーション-ASCII コマンド制御)	70
6.2.4	Advertising 動作設定	71
6.2.5	Advertising データ	71
6.2.6	Advertising データ更新設定	73
6.2.7	White List 設定	73
7.	関数	74
7.1	関数一覧	74
7.1.1	ビーコンアプリケーション	74
7.1.2	スキャンアプリケーション	74
7.1.3	DTM アプリケーション	74
7.1.4	ビーコンアプリケーション-ASCII コマンド制御	75
8.	動作	76
8.1	状態遷移	76
8.1.1	ビーコンアプリケーション	76
8.1.2	スキャンアプリケーション	77
8.1.3	DTM アプリケーション	78
8.1.4	ビーコンアプリケーション-ASCII コマンド制御	79
8.2	シーケンス	80
8.2.1	RF 初期化	80
8.2.2	ビーコンアプリケーション	81
8.2.3	スキャンアプリケーション	82
8.2.4	DTM アプリケーション	83
8.2.5	ビーコンアプリケーション-ASCII コマンド制御	84
9.	Appendix	85
9.1	ビーコンスタックの消費電流	85
9.1.1	注意事項	85
9.1.2	測定環境	85

9.1.3	測定結果：ビーコンスタック起動	86
9.1.4	測定結果：周期的な Advertising パケット送信	90
9.2	デバイスアドレス	102
9.3	Advertising パケットフォーマット	103

1. 概要

サンプルプログラムのシステム構成を図 1-1 に示します。サンプルプログラムはビーコンアプリケーション、スキャンアプリケーション、DTM アプリケーション、ビーコンスタックで構成されます。またサンプルプログラムは RL78/G1D 評価ボードで動作します。

- ビーコンアプリケーション : 情報発信のための Advertising を実行
- スキャンアプリケーション : Advertising パケットを受信するための Scanning を実行
- DTM アプリケーション : RF 特性評価のための Direct Test Mode を実行
- ビーコンスタック : Advertising、Scanning、Direct Test Mode を実行する API を提供

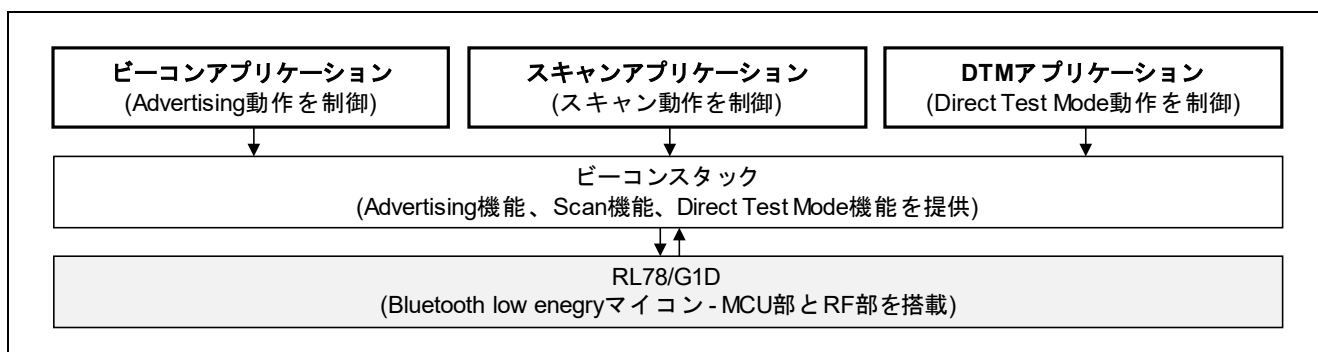


図 1-1 システム構成

ビーコンアプリケーション、スキャンアプリケーション、DTM アプリケーションの仕様については 5 章「仕様」を参照してください。

ビーコンスタックの仕様については『RL78/G1D ビーコンスタック ユーザーズマニュアル』(R01UW0171)を参照してください。

評価ボードの詳細については『RL78/G1D 評価ボード ユーザーズマニュアル』(R30UZ0048)を参照してください。

1.1 ビーコンアプリケーション

ビーコンアプリケーションの動作概要を図 1-2 に示します。ビーコンアプリケーションを起動するには、評価ボードの DIP スイッチ SW6-1 と SW6-4 を OFF に設定して電源供給を開始します。

ビーコンアプリケーションは起動後、Non-connectable Undirected Advertising パケット送信を開始します。評価ボードのスイッチ SW2 を押下すると、Non-connectable Undirected Advertising パケットの送信を停止します。再度 SW2 を押下すると、Scannable Undirected Advertising パケットの送信を開始します。その後 SW2 を押下すると Scannable Undirected Advertising パケットの送信を停止します。このようにスイッチ SW2 を使用することで、Non-connectable Undirected Advertising パケットと Scannable Undirected Advertising パケットの送信開始と送信停止が可能です。

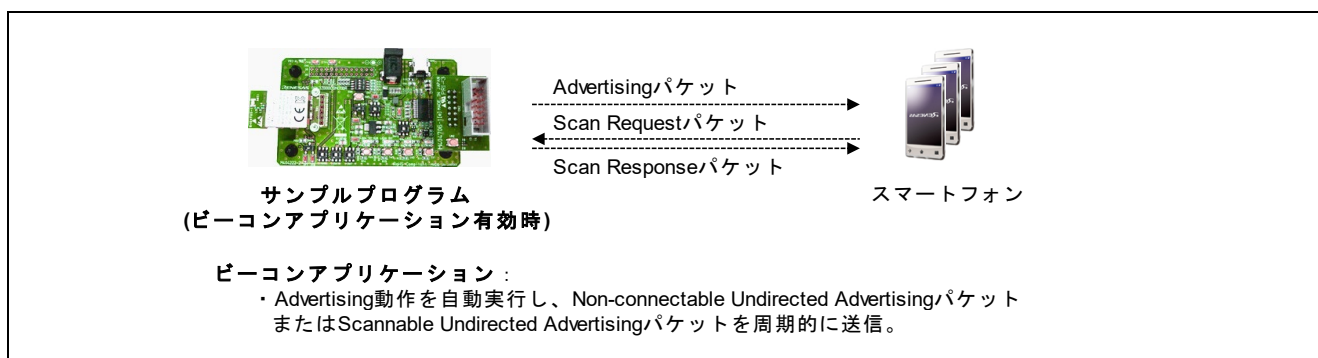


図 1-2 ビーコンアプリケーションの動作概要

1.2 ビーコンアプリケーション - ASCII コマンド制御

ビーコンアプリケーション-ASCII コマンド制御の動作概要を図 1-3 に示します。ビーコンアプリケーション-ASCII コマンド制御は APP_SELECT マクロで選択します。APP_SELECT マクロについては 6.2.1 項「アプリケーション選択設定」を参照してください。

ビーコンアプリケーション-ASCII コマンド制御は起動後、ホストマシンからのコマンドを待つ状態になります。ASCII 形式のコマンドをホストマシンから入力することで、ビーコンアプリケーションを制御し、Non-connectable Undirected Advertising パケットや Scannable Undirected Advertising パケットの送信、また Advertising 動作のパラメータを設定することが可能です。

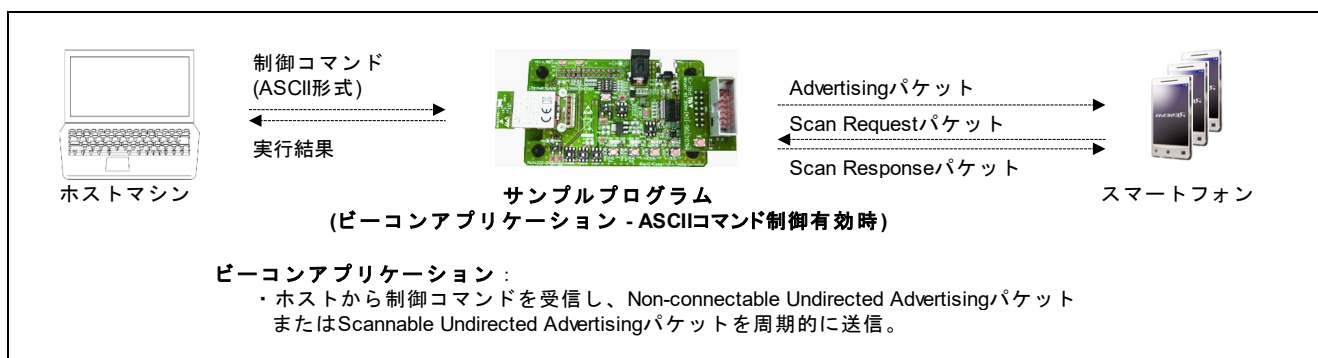


図 1-3 ビーコンアプリケーション-ASCII コマンド制御の動作概要

1.3 スキャンアプリケーション

スキャンアプリケーションの動作概要を図 1-4 に示します。スキャンアプリケーションは ASCII 形式またはバイナリ形式のいずれかで UART 通信が可能です。

ASCII 形式で UART 通信するスキャンアプリケーションを起動するには、評価ボードの DIP スイッチ SW6-1 を ON、SW6-4 を OFF に設定して電源供給を開始します。バイナリ形式で UART 通信するスキャンアプリケーションを起動するには、評価ボードの DIP スイッチ SW6-1 と SW6-4 を ON に設定して電源供給を開始します。

ASCII 形式 UART のスキャンアプリケーションは起動後、Advertising パケットを受信するための Scanning を自動で開始します。ビーコンデバイスからの Advertising パケットを受信すると、ASCII 形式の受信通知を UART から出力します。また ASCII 形式のコマンドを UART から入力することで、スキャンアプリケーションの制御が可能です。

バイナリ形式 UART のスキャンアプリケーションは起動後、Start Scan コマンドを UART から受信することで、Advertising パケットを受信するための Scanning を実行します。ビーコンデバイスからの Advertising パケットを受信すると、バイナリ形式の受信通知を UART から出力します。またバイナリ形式のコマンドを UART から入力することで、スキャンアプリケーションの制御が可能です。

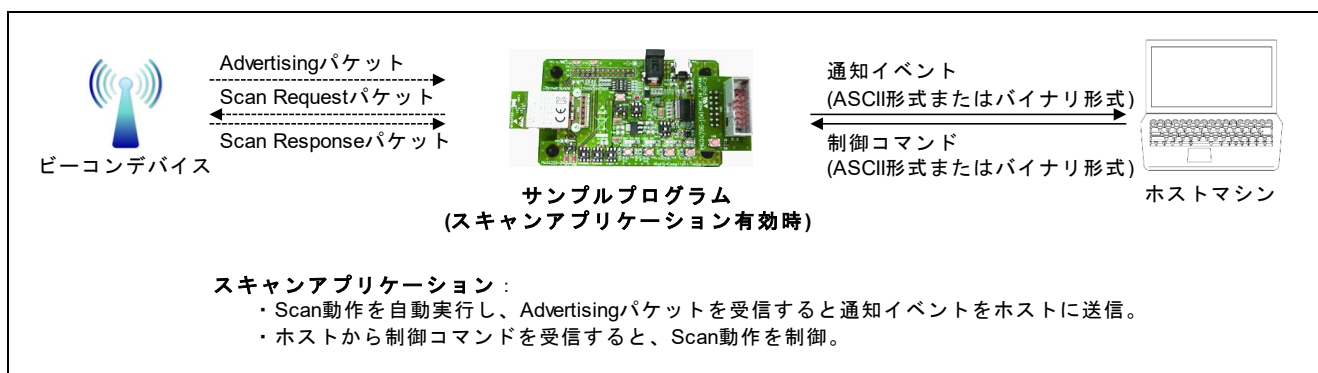


図 1-4 スキャンアプリケーションの動作概要

1.4 DTM アプリケーション

DTM アプリケーションの動作概要を図 1-5 に示します。

評価ボードの DIP スイッチ SW6-1 を OFF、SW6-4 を ON に設定して電源供給を開始すると、DTM アプリケーションが起動します。

DTM アプリケーションは RF テストコマンドの UART 入力により Direct Test Mode を実行し、実行結果を RF テストイベントとして UART 出力します。

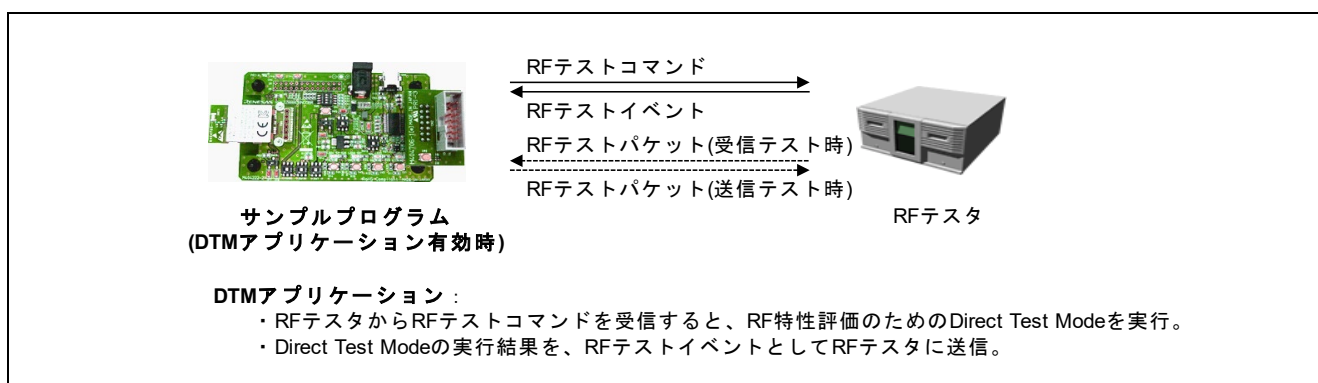


図 1-5 DTM アプリケーションの動作概要

2. 動作環境

サンプルプログラムのビルドと動作確認で使用する環境を示します。

- ハードウェア環境
 - ホストマシン
 - PC/AT™ 互換機
 - プロセッサ : 1.6GHz 以上
 - メインメモリ : 1Gbyte 以上
 - インタフェース : USB2.0 (E1 エミュレータおよび RL78/G1D 評価ボードの接続用)
 - デバイス
 - RL78/G1D 評価ボード(RTK0EN0001D01001BZ) : 2 台 (※)
 - USB ケーブル(A タイプ オス / mini-B タイプ オス) : 2 本
 - iOS デバイス または Android デバイス

※一方の RL78/G1D 評価ボードから Advertising パケットを送信し、他方の RL78/G1D 評価ボードで Advertising パケットを受信する動作確認を実施するため、評価ボードは 2 台必要です。

- ツール
 - Renesas オンチップデバッグエミュレータ E1(R0E000010KCE00)
- ソフトウェア環境
 - Windows®7 Service Pack1 以降
 - Renesas CS+ for CC V8.03.00 / Renesas CC-RL V1.09.00
または Renesas e² studio Version 7.8.0 / Renesas CC-RL V1.09.00
 - Renesas Flash Programmer V3.06.01
 - Tera Term Pro (またはシリアルポートと接続可能なターミナルソフト)
 - UART-USB 変換デバイスドライバ (※)

※評価ボードとホストマシンを接続する際に、UART-USB 変換 IC 「FT232RL」 のデバイスドライバを要求される場合があります。その際にはドライバを以下から入手してください。

- FTDI (Future Technology Devices International) – Drivers
<http://www.ftdichip.com/Drivers/D2XX.htm>

3. ファイル構成

サンプルプログラムにはビーコンスタックライブラリとビーコンアプリケーション、ビーコンアプリケーション-ASCII コマンド制御、スキャンアプリケーション、DTM アプリケーションのソースコードが含まれます。

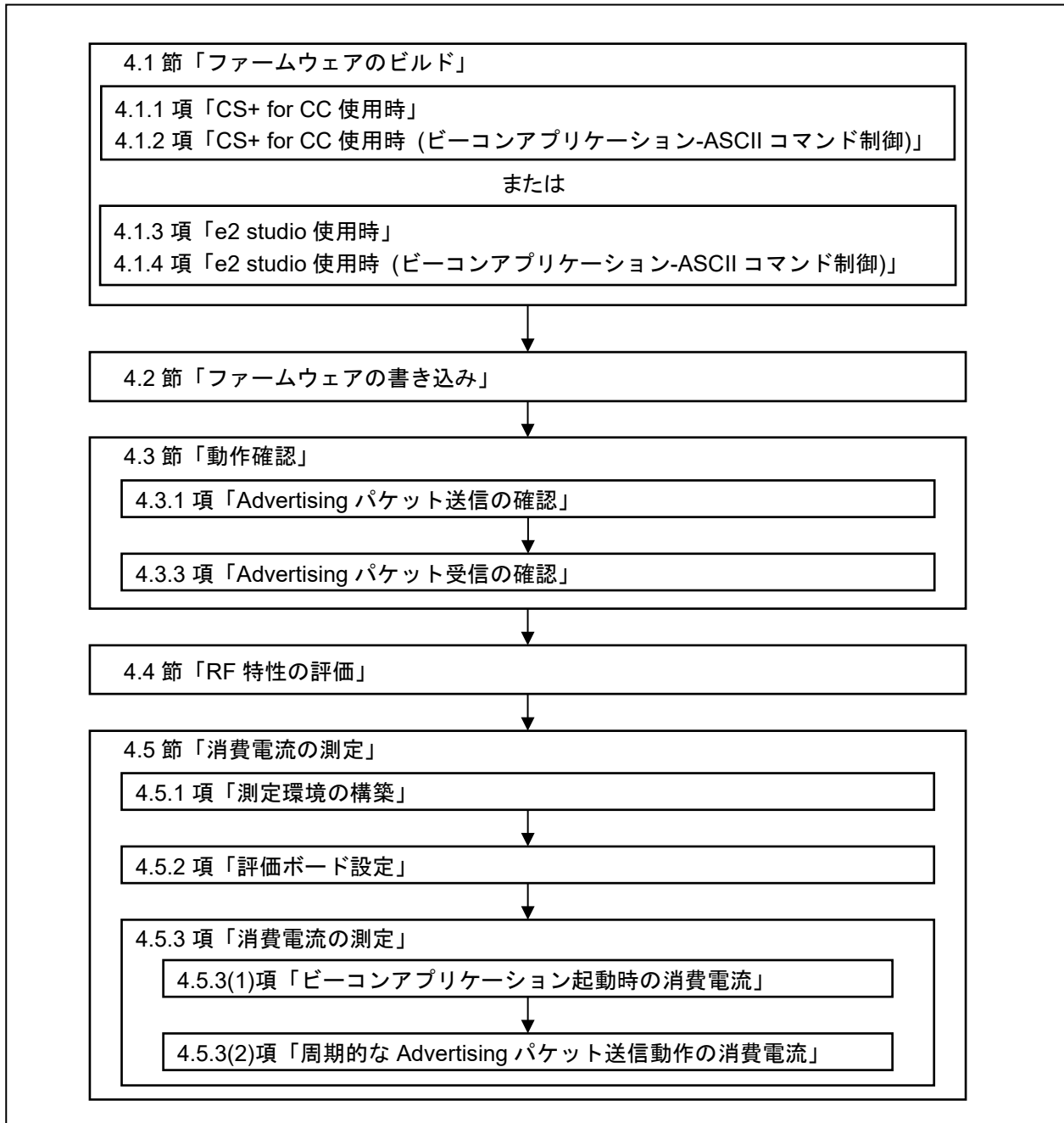
サンプルプログラムのリリースパッケージに含まれるファイルとフォルダの構成を以下に示します。

RL78G1D_Beacon		
├ROM_File		
R5F11AGJ_Beacon.hex		サンプルプログラム - ファームウェア (R5F11AGJ) (全アプリケーションが有効、DIP SW で切り替え)
R5F11AGJ_Beacon(beacon).hex		サンプルプログラム - ファームウェア (R5F11AGJ) (ビーコンアプリケーションのみ有効)
R5F11AGJ_Beacon(scan_ascii).hex		サンプルプログラム - ファームウェア (R5F11AGJ) (ASCII 形式スキャンアプリケーションのみ有効)
R5F11AGJ_Beacon(scan_bin).hex		サンプルプログラム - ファームウェア (R5F11AGJ) (バイナリ形式スキャンアプリケーションのみ有効)
R5F11AGJ_Beacon(dtm).hex		サンプルプログラム - ファームウェア (R5F11AGJ) (DTM アプリケーションのみ有効)
R5F11AGJ_Beacon(beacon_ascii).hex		サンプルプログラム - ファームウェア (R5F11AGJ) (ビーコンアプリケーション-ASCII コマンド制御のみ有効)
├RUC_File		
r5fl1agg_syscfg.ruc		システム動作設定 - ユニークコードファイル (R5F11AGG)
r5fl1agh_syscfg.ruc		システム動作設定 - ユニークコードファイル (R5F11AGH)
r5fl1agj_syscfg.ruc		システム動作設定 - ユニークコードファイル (R5F11AGJ)
r5fl1agg_syscfg(beacon_ascii).ruc		システム動作設定 - ユニークコードファイル (R5F11AGG) (ビーコンアプリケーション-ASCII コマンド制御用)
r5fl1agh_syscfg(beacon_ascii).ruc		システム動作設定 - ユニークコードファイル (R5F11AGH) (ビーコンアプリケーション-ASCII コマンド制御用)
r5fl1agj_syscfg(beacon_ascii).ruc		システム動作設定 - ユニークコードファイル (R5F11AGJ) (ビーコンアプリケーション-ASCII コマンド制御用)
├Project_Source		
├library		
r_arch.h		アーキテクチャ - ヘッダファイル
r_compiler.h		コンパイラ依存 - ヘッダファイル
r_iodef.h		SFR 定義 - CC-RL 用ヘッダファイル
r_ll.h		低レベル処理 - ヘッダファイル
r_port.h		ポートアクセス - ヘッダファイル
r_bcn_api.h		ビーコンスタック API - ヘッダファイル
BLE_BEACON_CC.lib		ビーコンスタック - CC-RL 用ライブラリ
├codeflash		
fsl.h		Flash self-programming library - ヘッダファイル
fsl_types.h		Flash self-programming library - ヘッダファイル
fsl.lib		Flash self-programming library - CC-RL 用ライブラリ
├application		
├src		
cstart.asm		スタートアップ - CC-RL 用アセンブラファイル
r_config.h		コンフィグレーション - ヘッダファイル
r_main.c		エントリポイント - ヘッダファイル
r_interrupt.c		割り込みハンドラ - コードファイル
r_beacon.h		ビーコンアプリケーション - ヘッダファイル
r_beacon.c		ビーコンアプリケーション - コードファイル
r_beacon_ascii.c		ビーコンアプリケーション-ASCII コマンド制御 - コードファイル
r_scan.h		スキャンアプリケーション - ヘッダファイル
r_scan_ascii.c		スキャンアプリケーション(UART-ASCII) - コードファイル
r_scan_bin.c		スキャンアプリケーション(UART-Binary) - コードファイル
r_dtm.h		DTM アプリケーション - ヘッダファイル
r_dtm.c		DTM アプリケーション - コードファイル
├driver		
├plf		
r_plf.h		プラットフォームドライバ - ヘッダファイル
r_plf.c		プラットフォームドライバ - コードファイル

input		
r_input.h		外部入力割り込みドライバ - ヘッダファイル
r_input.c		外部入力割り込みドライバ - コードファイル
uart		
r_uart.h		UARTドライバ - ヘッダファイル
r_uart.c		UARTドライバ - コードファイル
led		
r_led.h		LEDドライバ - ヘッダファイル
r_led.c		LEDドライバ - コードファイル
codeflash		
r_codeflash.h		コードフラッシュドライバ - ヘッダファイル
r_codeflash.c		コードフラッシュドライバ - コードファイル
project		
cs_cc		
BLE_Software		
BLE_Software.mtpj		CS+ for CC 用プロジェクトファイル
R5F11AGG_Beacon		
R5F11AGG_Beacon.mtsp		CS+ for CC 用サブプロジェクトファイル (R5F11AGG)
R5F11AGH_Beacon		
R5F11AGH_Beacon.mtsp		CS+ for CC 用サブプロジェクトファイル (R5F11AGH)
R5F11AGJ_Beacon		
R5F11AGJ_Beacon.mtsp		CS+ for CC 用サブプロジェクトファイル (R5F11AGJ)
e2_cc		
BLE_Software		
R5F11AGG_Beacon		
.project		e ² studio 用プロジェクト構成ファイル (R5F11AGG)
.cproject		e ² studio 用プロジェクト設定ファイル (R5F11AGG)
R5F11AGH_Beacon		
.project		e ² studio 用プロジェクト構成ファイル (R5F11AGH)
.cproject		e ² studio 用プロジェクト設定ファイル (R5F11AGH)
R5F11AGJ_Beacon		
.project		e ² studio 用プロジェクト構成ファイル (R5F11AGJ)
.cproject		e ² studio 用プロジェクト設定ファイル (R5F11AGJ)
project_beacon_ascii		ビーコンアプリケーション-ASCII コマンド制御用プロジェクト
cs_cc		
BLE_Software		
BLE_Software.mtpj		CS+ for CC 用プロジェクトファイル
R5F11AGG_Beacon		
R5F11AGG_Beacon.mtsp		CS+ for CC 用サブプロジェクトファイル (R5F11AGG)
R5F11AGH_Beacon		
R5F11AGH_Beacon.mtsp		CS+ for CC 用サブプロジェクトファイル (R5F11AGH)
R5F11AGJ_Beacon		
R5F11AGJ_Beacon.mtsp		CS+ for CC 用サブプロジェクトファイル (R5F11AGJ)
e2_cc		
BLE_Software		
R5F11AGG_Beacon		
.project		e2 studio 用プロジェクト構成ファイル (R5F11AGG)
.cproject		e2 studio 用プロジェクト設定ファイル (R5F11AGG)
R5F11AGH_Beacon		
.project		e2 studio 用プロジェクト構成ファイル (R5F11AGH)
.cproject		e2 studio 用プロジェクト設定ファイル (R5F11AGH)
R5F11AGJ_Beacon		
.project		e2 studio 用プロジェクト構成ファイル (R5F11AGJ)
.cproject		e2 studio 用プロジェクト設定ファイル (R5F11AGJ)

4. 操作方法

サンプルプログラムの動作を確認するための操作方法について示します。操作手順は5つのステップ（ファームウェアのビルド、ファームウェアの書き込み、動作確認、RF 特性の評価、消費電流の測定）があります。



4.1 ファームウェアのビルド

サンプルプログラムのビルドには、CS+ for CC または e² studio を使用します。

サンプルプログラムをデフォルト設定でビルドすると、リリースパッケージに含まれる HEX 形式のファームウェアファイル R5F11AGJ_Beacon.hex が生成されます。なおリリースパッケージに含まれるファームウェアファイルを使用することで、本手順の省略が可能です。

ビーコンアプリケーション-ASCII コマンド制御は APP_SELECT マクロで選択します。APP_SELECT マクロについては 6.2.1 項「アプリケーション選択設定」を参照してください。

4.1.1 CS+ for CC 使用時

1. CS+ for CC を起動し、[ファイル]→[ファイルを開く]を選択して下記のパスにある BLE_Software.mtpj プロジェクトファイルを開きます。
 - Project_Source¥application¥project¥cs_cc¥BLE_Software¥
2. [ビルド]→[リビルド・プロジェクト]を選択し、ビルドが成功することを確認します。
3. 下記のパスに R5F11AGJ_Beacon.hex が生成されていることを確認します。
 - Project_Source¥application¥project¥cs_cc¥BLE_Software¥R5G11AGJ_Beacon¥DefaultBuild¥

4.1.2 CS+ for CC 使用時 (ビーコンアプリケーション-ASCII コマンド制御)

ビーコンアプリケーション-ASCII コマンド制御をビルドする場合は、APP_SELECT マクロを下記に設定してください。APP_SELECT マクロについては 6.2.1 項「アプリケーション選択設定」を参照してください。

- #define APP_SELECT (5)

また下記パスのプロジェクトファイルを使用します。

- Project_Source¥application¥project_beacon_ascii¥cs_cc¥BLE_Software¥

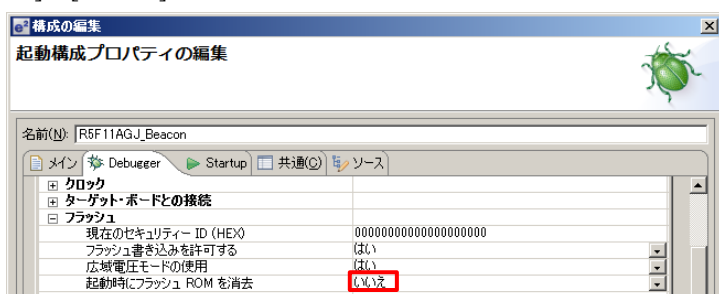
ビルド手順は 4.1.1 項を参照してください。

4.1.3 e² studio 使用時

- e² studio を起動し、下記のパスをワークスペースとして選択します。
 - Project_Source¥
- [ファイル]→[インポート]を選択し、インポートダイアログを表示します。
- [一般]→[既存プロジェクトをワークスペースへ]を選択し、[次へ]をクリックします。
- [ルートディレクトリの選択]で下記のパスを選択し、[プロジェクト]に表示される R5F11AGJ_Beacon を選択します。
 - Project_Source¥application¥project¥e2_cc
- [終了]をクリックし、インポートダイアログを閉じます。
- [よろこ]を閉じます。
- プロジェクト・エクスプローラーで R5F11AGJ_Beacon を選択します。
- [プロジェクト]→[プロジェクトのビルド]を選択し、ビルドが成功することを確認します。
- 下記のパスに R5F11AGJ_Beacon.hex が生成されていることを確認します。
 - Project_Source¥application¥project¥e2_cc¥BLE_Software¥R5F11AGJ_Beacon¥DefaultBuild¥

※e² studio のデバッガのデフォルト設定は、起動時にフラッシュメモリを消去します。e² studio での開発時、RL78/G1D モジュールに書き込まれている出荷時検査フラグ、デバイスアドレスの消去を防止するため、デバッグ接続時に下記の設定変更を行ってください。また下記の設定変更は、RL78/G1D モジュールと E1 エミュレータの接続を外した状態で行ってください。

- [起動構成プロパティの編集]ダイアログの[Debugger]タブを選択し、[起動時にフラッシュ ROM を消去]で[いいえ]を設定



4.1.4 e² studio 使用時 (ビーコンアプリケーション-ASCII コマンド制御)

ビーコンアプリケーション-ASCII コマンド制御をビルドする場合は、APP_SELECT マクロを下記に設定してください。APP_SELECT マクロについては 6.2.1 項「アプリケーション選択設定」を参照してください。

- #define APP_SELECT (5)

またインポートの際に下記パスを指定してください。

- Project_Source¥application¥project_beacon_ascii¥e2_cc

ビルド手順は 4.1.3 項を参照してください。

4.2 ファームウェアの書き込み

一方の評価ボードから Advertising パケットを送信し、他方の評価ボードで Advertising パケットを受信する動作確認を実施するため、2 台の評価ボードに対して、4.1 節でビルドしたサンプルプログラムファームウェアを書き込みます。

サンプルプログラムのファームウェア書き込みには、ホストマシンと E1 エミュレータを図 4-1 のように使用します。

E1 エミュレータの詳細については『E1/E20 エミュレータ ユーザーズマニュアル』(R20UT0398)および『E1/E20 エミュレータ, E2 エミュレータ Lite ユーザーズマニュアル別冊 (RL78 接続時の注意事項)』(R20UT1994)を参照してください。

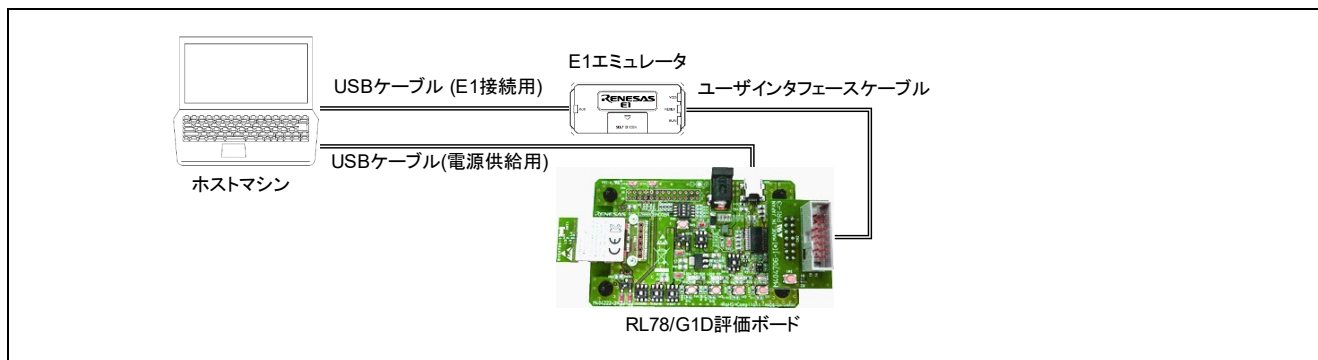


図 4-1 ファームウェア書き込み時の評価ボード接続

サンプルプログラム動作確認時のスライドスイッチ設定を表 4-1 に示します。

表 4-1 サンプルプログラム動作確認時のスライドスイッチ設定

スイッチ	設定	説明
SW7	2-3 接続(右側) (デフォルト設定)	AC 電源アダプタを接続するための DC ジャック(J1)または USB インタフェース(CN3)からレギュレータ経由で電源供給
SW8	2-3 接続(右側)	USB から電源供給 ※DC5V から電源供給する場合は 1-2 接続(左側)
SW9	2-3 接続(右側) (デフォルト設定)	USB と接続
SW10	1-2 接続(左側) (デフォルト設定)	モジュールに電源供給
SW11	2-3 接続(右側) (デフォルト設定)	E1 エミュレータ 3.3V 以外から電源供給
SW12	2-3 接続(右側) (デフォルト設定)	(デフォルト固定)
SW13	1-2 接続(左側) (デフォルト設定)	USB 接続

サンプルプログラムファームウェアの RL78/G1D 評価ボードへの書き込み手順を以下に示します。
 サンプルプログラムファームウェアの書き込みには、Renesas Flash Programmer(RFP)を使用します。

1. 表 4-1 を参照し、評価ボードのスライドスイッチを設定します。
2. E1 エミュレータを評価ボードに接続後、E1 エミュレータとホストマシンを接続します。
3. 評価ボードをホストマシンまたは AC-USB 電源アダプタと USB で接続し、電源供給を開始します。
4. RFP を起動し、下記の手順でプロジェクトを作成します。

※プロジェクト作成後は、作成したプロジェクトを使用することで次回から本手順を省略可能です。

- 4-1. [ファイル]→[新しいプロジェクトを作成]を選択します。
- 4-2. [新しいプロジェクトの作成]ダイアログの[プロジェクト情報]で[RL78]を選択し、任意のプロジェクト名を入力後、[接続]をクリックします。
- 4-3. ログ出力ウィンドウに「操作が成功しました」と表示されることを確認します。
5. [プログラムファイル]で R5F11AGJ_Beacon.hex ファームウェアファイルを指定します。
6. 下記の手順でコードフラッシュメモリの Block254,255 消去を防止します。

※RL78/G1D モジュールでは Block254 に出荷時検査フラグ、Block255 にデバイスアドレスが書き込まれています。

- 6-1. [操作設定]タブを選択し、[消去オプション]で[ブロック選択消去]を選択します。
- 6-2. [ブロック設定]タブを選択し、Block254,255 の[Erase]、[P.V]のチェックを外します。

Block253	0x0003F400	0x0003F7FF	1 K	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Block254	0x0003F800	0x0003FBFF	1 K	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Block255	0x0003FC00	0x0003FFFF	1 K	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Data Flash 1	0x000F1000	0x000F2FFF	8 K	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

7. [スタート]を押下して書き込み開始後、「操作が完了しました」と表示されることを確認します。
8. 電源、E1 エミュレータを評価ボードから取り外します。

さらに RFP のユニークコード埋め込み機能を使用することにより、同一ファームウェアとデバイスごとに異なるシステム動作設定を書き込むことが可能です。

システム動作設定を書き込む場合は、上記手順 7 で[スタート]を押下する前に、下記の手順でユニークコードファイルとして指定します。

1. [ユニークコード設定]タブを選択します。
2. [有効にする]にチェックを入れます。
3. [ユニークコードファイル]で下記のユニークコードファイルを指定します。
 - RUC_File¥r5f11agj_syscfg.ruc
4. [操作]タブに戻ります。

システム動作設定については 5.5.1 項「コードフラッシュメモリへのアクセス」を参照してください。

ユニークコード埋め込み機能については『Renesas Flash Programmer V3.06 フラッシュ書き込みソフトウェア ユーザーズマニュアル』(R20UT4540)の 2.3.6 項「[ユニークコード]タブ」を参照してください。

4.3 動作確認

サンプルプログラムファームウェアを書き込んだ評価ボードを使用して、動作確認を行います。

サンプルプログラムの動作確認では、評価ボードのスイッチ(SW)と LED をユーザインタフェースとして使用します。サンプルプログラムは DIP スイッチ SW6-1 と SW6-4 を動作開始時に読み込み、実行するアプリケーションを切り替えます。またビーコンアプリケーションとスキャンアプリケーションは SW2 を外部入力割り込みのトリガとして使用します。

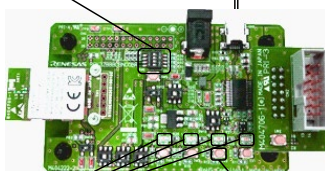
ビーコンアプリケーション-ASCII コマンド制御は、DIP スイッチで選択せず APP_SELECT マクロで選択しビルドしたサンプルプログラムファームウェアを使用します。APP_SELECT マクロについては 6.2.1 項「アプリケーション選択設定」を参照してください。

DIP SW6 :

SW6-1 SW6-4

- off - - off - : ビーコンアプリケーションを有効化
- off - - on - : DTMアプリケーションを有効化
- on - - off - : スキャンアプリケーションを有効化、ASCII形式のUART通信を実行
- on - - on - : スキャンアプリケーションを有効化、バイナリ形式のUART通信を実行

USBケーブル(電源供給とシリアル通信)



LED1-LED4 : (スキャンアプリケーション動作時のみ)

パケットを受信すると、LED点灯がシフト

LED4 : (ビーコンアプリケーション、DTMアプリケーション動作時のみ)

アプリケーション動作開始で、LEDが点灯

SW2 : (ビーコンアプリケーション動作時のみ)

SW2の押下で、ADV_NONCONN_IND送信とADV_SCAN_IND送信を交互に切替

SW1 : (スキャンアプリケーション動作時のみ)

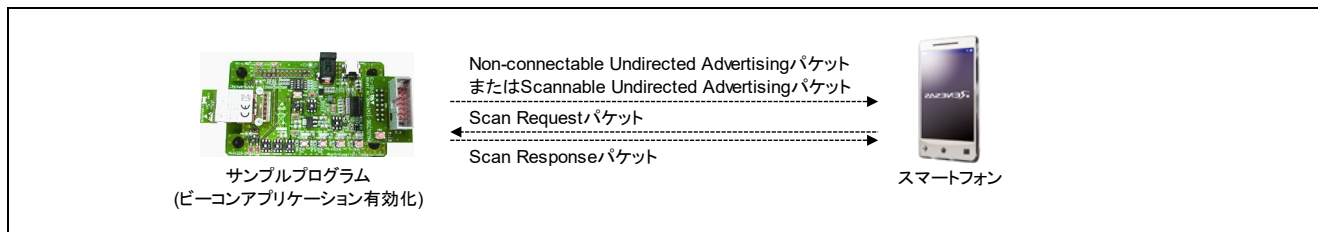
SW1の押下で、Advertisingデータの16進数表記と文字列表記を交互に切替

図 4-2 動作確認時の評価ボード操作

サンプルプログラム動作確認時のスライドスイッチ設定については 4.2 節の表 4-1 を参照してください。

4.3.1 Advertising パケット送信の確認

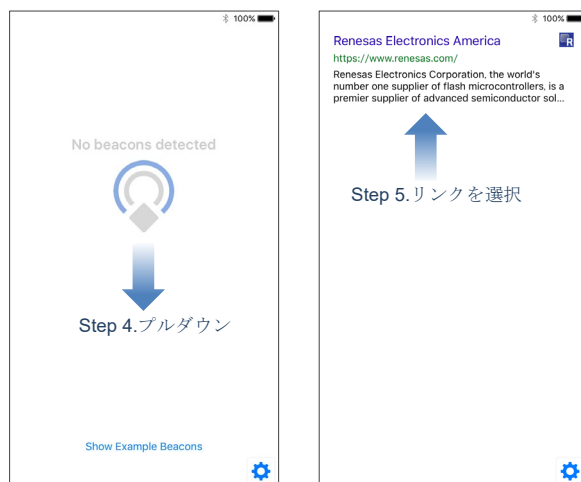
スマートフォンを使用して、ビーコンアプリケーションが Advertising パケット送信することを確認します。



サンプルプログラムファームウェアを書き込んだ評価ボードの DIP スイッチ SW6-1 と DIP スイッチ SW6-4 を OFF に設定後、DC ジャック(J1)または USB インタフェース(CN3)から電源供給を開始すると、ビーコンアプリケーションが実行されます。

ビーコンアプリケーションはデフォルトで Eddystone-URL を送信します。スマートフォンを使用することで本パケットを受信可能です。確認手順は iOS デバイスと Android デバイスで共通です。

1. Eddystone-URL パケットを受信するために、下記のアプリをスマートフォンにインストールします。
 - Android デバイス向け, Physical Web - Google Play
<https://play.google.com/store/apps/details?id=com.physicalweb>
 - iOS デバイス向け, Physical Web - App Store
<https://itunes.apple.com/app/physical-web/id927653608?mt=8>
2. 評価ボードの DIP スイッチ SW6-1 と SW6-4 を OFF に設定します。
3. 評価ボードへの電源供給を開始し、ビーコンアプリケーションを起動します。
4. スマートフォンアプリを起動し、画面のプルダウンで Eddystone ビーコンを探索します。
5. サンプルプログラムから Eddystone-URL パケットを受信すると、下記 URL へのリンクが表示されます。
 - Renesas Electronics
<https://www.renesas.com/>



4.3.2 Advertising パケット送信の確認 (ビーコンアプリケーション-ASCII コマンド制御)

スマートフォンを使用して、ビーコンアプリケーション-ASCII コマンド制御が Advertising パケット送信することを確認します。



サンプルプログラムファームウェアを書き込んだ評価ボードの DC ジャック (J1) または USB インタフェース (CN3) から電源供給を開始すると、ビーコンアプリケーション-ASCII コマンド制御が実行されホストマシンからの制御コマンドを待つ状態になります。

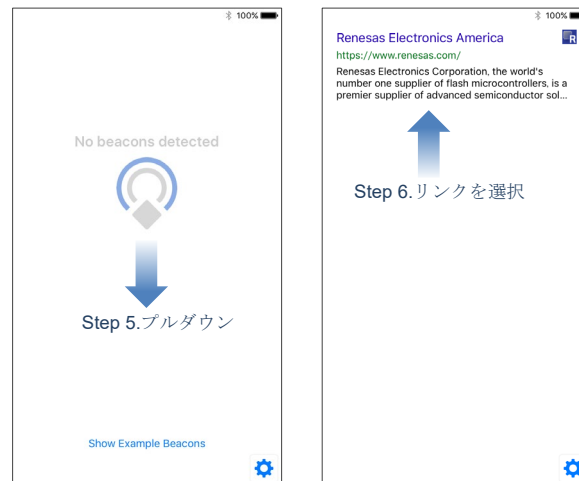
ビーコンアプリケーション-ASCII コマンド制御はデフォルトで Eddystone-URL を送信することができます。スマートフォンを使用することで本パケットを受信可能です。確認手順は iOS デバイスと Android デバイスで共通です。

- Eddystone-URL パケットを受信するために、下記のアプリをスマートフォンにインストールします。
 - Android デバイス向け, Physical Web - Google Play
<https://play.google.com/store/apps/details?id=com.physicalweb>
 - iOS デバイス向け, Physical Web - App Store
<https://itunes.apple.com/app/physical-web/id927653608?mt=8>
- ビーコンアプリケーション-ASCII コマンド制御の評価ボードとホストマシンを USB ケーブルで接続し、評価ボードへの電源供給を開始します。これにより評価ボードはホストマシンからの制御コマンドを待つ状態になります。
- ホストマシンにインストールしたターミナルソフトを起動し、表 4-2 を参照して評価ボードと接続します。

表 4-2 ターミナルソフトのシリアルポート設定

設定項目		設定値
シリアルポート	ポート	USB シリアルポート ※COM 番号は評価ボードごとに異なる
	ボーレート	4,800bps
	データ長	8bit
	パリティ	None
	ストップビット	1bit
	フロー制御	None
改行コード	受信	LF
	送信	LF
端末サイズ	横幅	128 文字以上

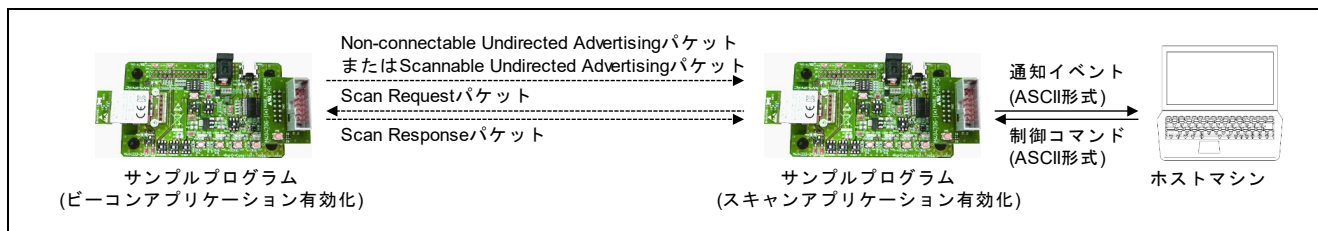
4. ターミナルソフトに"adstart n"を入力しエンターキーを押します。Non-connectable Undirected Advertising パケットが送信されます。ASCII コマンドの仕様は 5.2 節「ビーコンアプリケーション - ASCII コマンド制御」を参照してください。
5. スマートフォンアプリを起動し、画面のプルダウンで Eddystone ビーコンを探索します。
6. サンプルプログラムから Eddystone-URL パケットを受信すると、下記 URL へのリンクが表示されます。
 - Renesas Electronics
<https://www.renesas.com/>



4.3.3 Advertising パケット受信の確認

スキャンアプリケーションが Advertising パケットを受信することを確認します。

起動すると、ビーコンアプリケーションは Advertising パケットを周期的に送信し、スキャンアプリケーションは Scanning を実行します。Advertising パケットを受信すると、スキャンアプリケーションはホストマシンにイベントを通知します。



サンプルプログラムファームウェアを書き込んだ評価ボードの DIP スイッチ SW6-1 と DIP スイッチ SW-4 を OFF に設定後、DC ジャック(J1)または USB インタフェース(CN3)から電源供給を開始すると、ビーコンアプリケーションが実行されます。

また評価ボードの DIP スイッチ SW6-1 を ON、DIP スイッチ SW-4 を OFF に設定してホストマシンから USB インタフェース(CN3)経由で電源供給を開始すると、スキャンアプリケーションが実行されます。

1. スキャンアプリケーション側評価ボードの DIP スイッチ SW6-1 を ON、SW6-4 を OFF に設定します。
2. スキャンアプリケーション側評価ボードとホストマシンを USB ケーブルで接続し、評価ボードへの電源供給を開始します。これによりスキャンアプリケーションが起動します。
3. ホストマシンにインストールしたターミナルソフトを起動し、表 4-3 を参照して評価ボードと接続します。
4. スキャンアプリケーション側評価ボードの RESET スイッチ SW5 を押下し、ターミナルソフトに下記のメッセージが表示されることを確認します。本メッセージはスキャンアプリケーションが Scanning を開始したことを示します。

```
Start Scan :OK
```

5. ビーコンアプリケーション側評価ボードの DIP スイッチ SW6-1 と SW6-4 を OFF に設定します。
6. ビーコンアプリケーション側評価ボードへの電源供給を開始します。これによりビーコンアプリケーションが起動し、Advertising パケットの送信を開始します。
7. スキャンアプリケーションが Advertising パケットを受信し、ターミナルソフトに以下のメッセージが表示されることを確認します。本メッセージはスキャンアプリケーションが受信した Advertising パケットの PDU タイプ、デバイスアドレスタイプ、デバイスアドレス、RSSI、ペイロードデータサイズ、ペイロードデータを示します。

```
ADV_NONCONN_IND PUBLIC 12:34:56:78:9A:B1 37ch -46dBm 27byte
0201060303AAFE1316AAFE10EE02676F6F2E676C2F35774B6B524B
```

周辺に Advertising パケットを送信するデバイスがある場合、周辺デバイスが送信した Advertising パケットのメッセージも表示されます。

コマンドを入力することにより、Scanning の開始・停止と設定の変更が可能です。コマンドの仕様については 5.3.1 項「ASCII 形式 UART 通信」を参照してください。

表 4-3 ターミナルソフトのシリアルポート設定

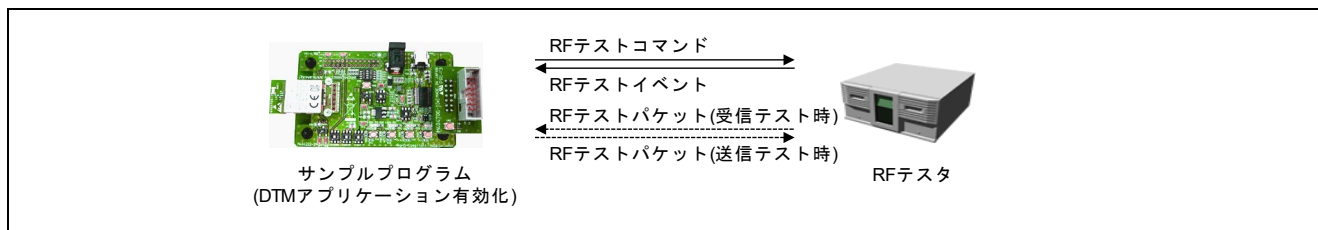
設定項目		設定値
シリアルポート	ポート	USB シリアルポート ※COM 番号は評価ボードごとに異なる
	ボーレート	1,000,000bps
	データ長	8bit
	パリティ	None
	ストップビット	1bit
	フロー制御	None
改行コード	受信	LF
	送信	LF
端末サイズ	横幅	128 文字以上

ターミナルソフトとして Tera Term をご使用の場合、「ボー・レート」のドロップダウンリストに 1,000,000bps は含まれていません。「ボー・レート」欄に直接"1000000"と入力してください。



4.4 RF 特性の評価

DTM アプリケーションを有効化し、Direct Test Mode を実行して RF 特性を評価します。



DIP スイッチ SW6-1 を OFF、DIP スイッチ SW6-4 を ON に設定して電源供給を開始すると、DTM アプリケーションが実行されます。

DTM アプリケーションは UART 経由で RF テストコマンドを受信することで、Direct Test Mode を実行します。RL78/G1D 評価ボードと RF テスタを UART で接続することで、RL78/G1D の RF 特性評価が可能です。

1. 評価ボードの DIP スイッチ SW6-1 を OFF、SW6-4 を ON に設定します。
2. 評価ボードの UART Tx/D0 端子、Rx/D0 端子、GND 端子を RF テスタに接続します。RL78/G1D と RF テスタのロジックレベルが異なる場合は、ロジックレベル変換器を介して RF テスタに接続します。
3. 評価ボードへの電源供給を開始し、DTM アプリケーションを起動します。
4. RF テスタのマニュアルを参照し、表 4-4 を参照して RF テスタの UART を設定します。
5. RF テスタのマニュアルを参照し、RF テストから Direct Test Mode を実行します。

表 4-4 RF テスタの UART 設定

設定項目	設定値
ボーレート	9600bps
データ長	8bit
パリティ	None
ストップビット	1bit
フロー制御	None

4.5 消費電流の測定

RL78/G1D 評価ボード (RTK0EN0001D01001BZ) による消費電流の測定方法を以下に示します。RL78/G1D 評価ボードの詳細については『RL78/G1D 評価ボード ユーザーズマニュアル』(R30UZ0048)を参照してください。

4.5.1 測定環境の構築

消費電流の測定に必要な機材を表 4-5 に示します。

各機材の使用方法的詳細についてはご使用になる各機材のマニュアルを参照してください。

表 4-5 消費電流測定に必要な機材

機材	概要	機材例
電源	RL78/G1Dへの電力供給	安定化電源、またはバッテリー ※動作電圧の範囲内であること
測定装置	測定結果の確認と出力	オシロスコープ
電圧検出器	RL78/G1Dの動作電圧検出	電圧プローブ
電流検出器	RL78/G1Dの消費電流検出	クランプ式電流プローブ、または シャント抵抗(推奨値 10Ω)と電圧プローブの組合せ

電流検出器として電流プローブを使用する場合の測定環境を図 4-3 に示します。評価ボードの端子 TP7-TP8 間の電流を電流プローブで測定した結果が、RL78/G1D の消費電流です。

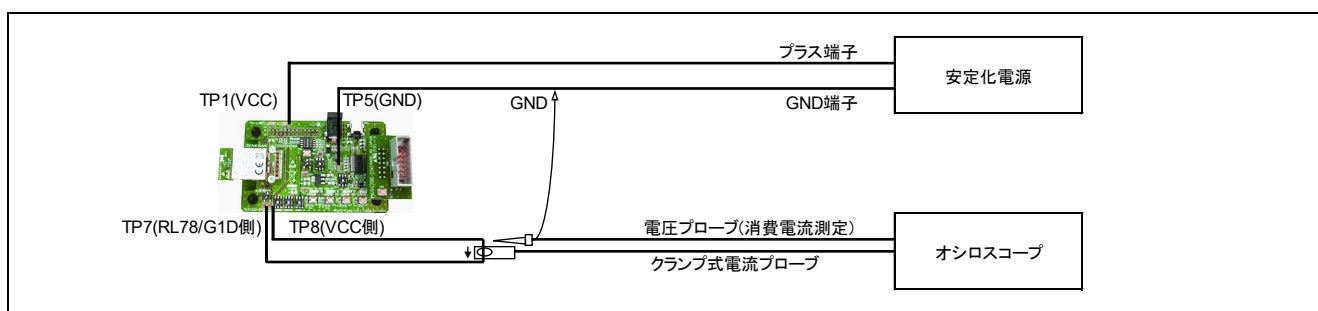


図 4-3 電流プローブを使用した消費電流測定環境

電流検出器としてシャント抵抗と電圧プローブを組み合わせる場合の測定環境を図 4-4 に示します。評価ボードの端子 TP7-TP8 間に抵抗を挿入し、2つの電圧プローブの測定結果から抵抗両端の電圧を測定します。

抵抗による電圧降下 dV は、2つの電圧プローブの測定結果の差分とします。抵抗による電圧降下 dV と抵抗値 R による電流 $I=dV/R$ の計算結果が、RL78/G1D の消費電流です。

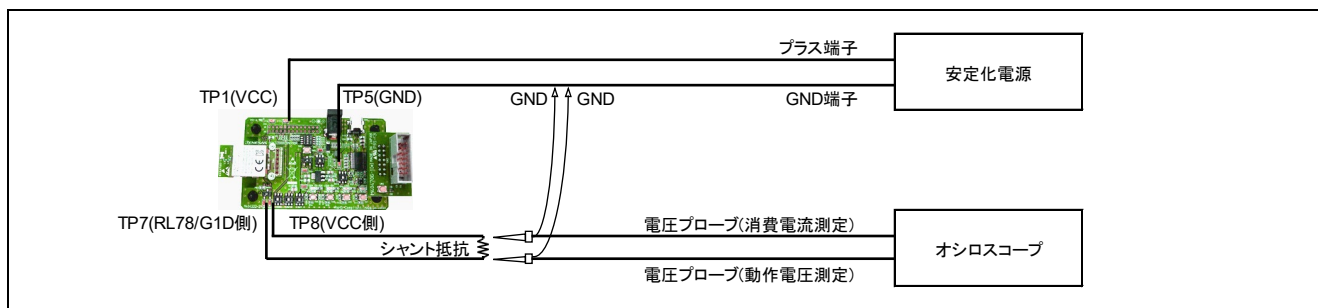


図 4-4 シャント抵抗と電圧プローブを使用した消費電流測定環境

4.5.2 評価ボード設定

表 4-6 を参照して評価ボードの消費電流測定時のスライドスイッチを設定します。

表 4-6 消費電流測定時のスライドスイッチ設定

スイッチ	設定	機能
SW7	1-2 接続(左側)	外部からレギュレータを経由せず電源供給 ※USBから電源供給する場合は2-3接続(右側)
SW8	1-2 接続(左側)	TP1,TP5 端子または AC 電源アダプタから電源供給 ※USB から電源供給する場合は 2-3 接続(右側)
SW9	1-2 接続(左側)	外部拡張インタフェースと接続
SW10	2-3 接続(右側)	電源供給ラインをオープン
SW11	2-3 接続(右側) (デフォルト設定)	E1デバッグ3.3V 以外から電源供給
SW12	2-3 接続(右側) (デフォルト設定)	(デフォルト固定)
SW13	2-3 接続(右側)	USB 切断

4.5.3 消費電流の測定

消費電流の測定手順を以下に示します。なお、測定手順に示すオシロスコープ設定例は、ビーコンアプリケーション動作時の消費電流を測定するための参考値です。

各機材の測定方法の詳細についてはご使用になる各機材のマニュアルを参照してください。

(1) ビーコンアプリケーション起動時の消費電流

1. 電源出力を停止した状態で評価ボードと電源を接続します。
2. 図 4-5 を参考にオシロスコープの測定項目を設定します。
 - 測定開始トリガ : 電源電圧プローブの 1.6V 程度
 - 電流測定レンジ : 10mA 程度
 - 測定時間 : 測定開始トリガを中心に 40msec 程度
3. RL78/G1D の動作電圧範囲で電源出力を開始します。
4. 電源電圧の上昇によりオシロスコープの測定が開始します。

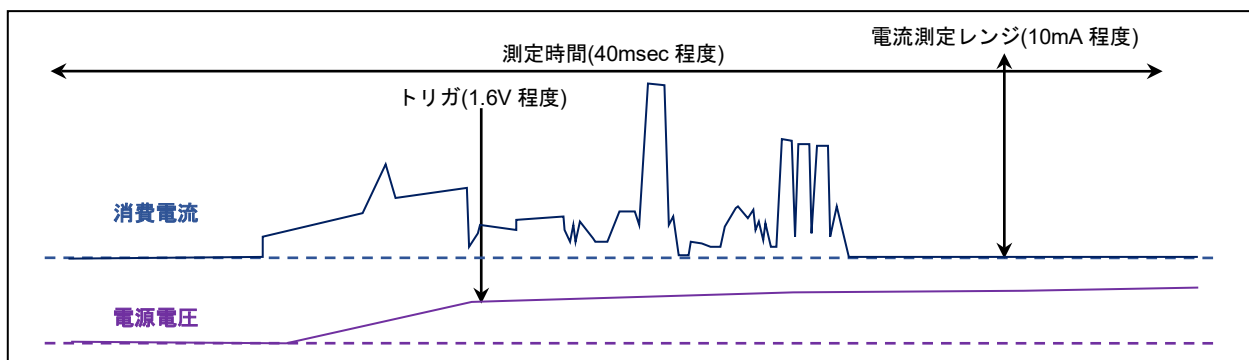


図 4-5 ビーコンアプリケーション起動時の消費電流

(2) 周期的な Advertising パケット送信動作の消費電流

1. 電源出力を開始しビーコンアプリケーションを起動します。
2. 図 4-6 を参考にオシロスコープの測定項目を設定します。
 - 測定開始トリガ : 消費電流プローブの 0.5mA 程度
 - 電流測定レンジ : 10mA 程度
 - 測定時間 : 測定開始を先頭に 10msec 程度
3. 周期的な Advertising パケット送信動作の電流によりオシロスコープの測定が開始します。

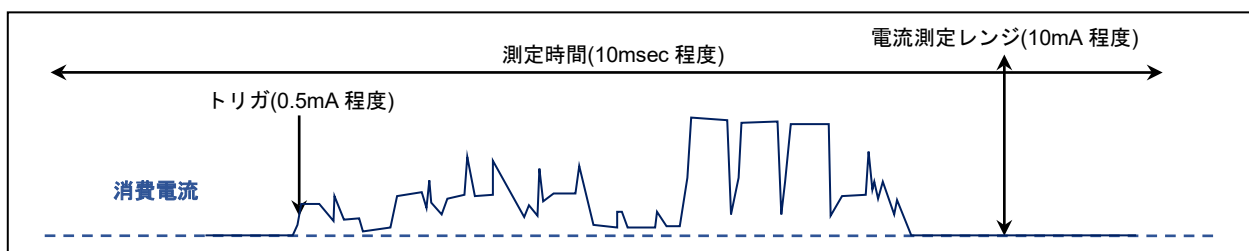


図 4-6 周期的な Advertising パケット送信時の消費電流

5. 仕様

5.1 ビーコンアプリケーション

ビーコンアプリケーションはコードフラッシュメモリのシステム動作設定から Advertising 設定と Advertising データを読み込み、情報発信のための Non-connectable Undirected Advertising パケットの送信を開始します。スマートフォンのような対向デバイスは Advertising パケットを受信し、Advertising データに対応したサービスを提供します。

評価ボードのスイッチ SW2 を押下すると Non-connectable Undirected Advertising パケットの送信を停止し、SW2 を再度押下すると Scannable Undirected Advertising パケットの送信を開始します。

5.1.1 デフォルト Advertising 動作設定

ビーコンアプリケーションのデフォルト Advertising 動作設定を表 5-1 に示します。Advertising データは、Non-connectable Undirected Advertising パケット送信時と Scannable Undirected Advertising パケット送信時で異なります。

表 5-1 ビーコンアプリケーションのデフォルト Advertising 動作設定

Advertiser Address	Public Device Address 12:34:56:78:9A:B1
Advertising Type	Non-connectable Undirected Advertising (ADV_NONCONN_IND) または Scannable Undirected Advertising(ADV_SCAN_IND)
Advertising Interval	100msec
Advertising Interval Delay	インターバルへのランダムディレイ加算あり
Advertising Channel Map	全チャンネル(37ch,38ch,39ch)
Advertising Count Limitation	回数無制限
Advertising Transmit Power	0dBm ※RL78/G1D の ANT 端子
Advertising Data Count	Advertising Type が ADV_NONCONN_IND ならば、Advertising データ数 1 Advertising Type が ADV_SCAN_IND ならば、Advertising データ数 2
Advertising Data [0]~[9]	Non-connectable Undirected Advertising 時は表 5-2 を参照 Scannable Undirected Advertising 時は表 5-3 を参照
Advertising Event Permission	全イベント通知を許可
Use White List	White List を使用しない

Non-connectable Undirected Advertising パケット送信時の Advertising データを表 5-2 に示します。

表 5-2 Non-connectable Undirected Advertising パケット送信時の Advertising データ

Advertising Data [0]~[9]	Advertising データ[0] (ADV_NONCONN_IND ペイロードデータ)	
	Length	2byte
	AD Type	<<Flags>> (0x01)
	AD Data	LE General Discoverable Mode (bit1) BR/EDR Not Supported (bit2)
	Length	3byte
	AD Type	<<Complete List of 16-bit Service Class UUIDs>> (0x03)
	AD Data	Eddystone (0xFEAA)
	Length	19byte
	AD Type	<<Service Data>> (0x16)
	AD Data	Eddystone-URL : https://goo.gl/5wKkRK
Advertising データ[1]~[9]は未設定		

Scannable Undirected Advertising パケット送信時の Advertising データを表 5-3 に示します。

表 5-3 Scannable Undirected Advertising パケット送信時の Advertising データ

Advertising Data [0]~[9]	Advertising データ[0] (ADV_SCAN_IND ペイロードデータ)	
	Length	2byte
	AD Type	<<Flags>> (0x01)
	AD Data	LE General Discoverable Mode (bit1) BR/EDR Not Supported (bit2)
	Length	3byte
	AD Type	<<Complete List of 16-bit Service Class UUIDs>> (0x03)
	AD Data	Eddystone (0xFEAA)
	Length	19byte
	AD Type	<<Service Data>> (0x16)
	AD Data	Eddystone-URL : http://goo.gl/JQh3fQ
	Advertising データ[1] (SCAN_RSP ペイロードデータ)	
	Length	24byte
	AD Type	<<Complete Local Name>> (0x09)
	AD Data	"Renesas RL78/G1D Beacon"
	Advertising データ[2]~[9]は未設定	

Eddystone と Eddystone-URL の仕様については下記の WEB サイトを参照してください。

- Specification for Eddystone, an open beacon format from Google
<https://github.com/google/eddystone>
- Specification for Eddystone, an open beacon format from Google - Eddystone-URL
<https://github.com/google/eddystone/tree/master/eddystone-url>

5.2 ビーコンアプリケーション - ASCII コマンド制御

ビーコンアプリケーション-ASCII コマンド制御は、ホストマシンから ASCII コマンドで Advertising の送信や Advertising の動作設定を行うことができます。

5.2.1 デフォルト Advertising 動作設定

ビーコンアプリケーション-ASCII コマンド制御のデフォルト設定は下記を参照してください。

- デフォルト Advertising 動作設定は表 5-1
- デフォルト Non-connectable Undirected Advertising データは表 5-2
- デフォルト Scannable Undirected Advertising データは表 5-3
- デフォルト White List データは表 5-4

表 5-4 デフォルト White List データ

White List [0]~[0xF]	White List データ[0]	
	Device Address Type	Public
	reserve	-
	Device Address	12:34:56:78:9A:B1
White List データ[1]~[0xF]は未設定		

5.2.2 ASCII 形式 UART 通信コマンド一覧

表 5-5 ASCII 形式 UART 通信コマンド一覧

コマンド	説明
pwup	RF 部への電源供給を開始/状態を表示
pwn	RF 部への電源供給を停止
adstart	Advertising を開始/状態を表示
adstop	Advertising を停止
aditvl	Advertising インターバルを設定/設定を表示
addly	Advertising インターバルへのランダムディレイ加算を設定/設定を表示
adch	Advertising チャンネルを設定/設定を表示
adloop	Advertising 回数制限を設定/設定を表示
adtxpw	Advertising パケット送信パワーを設定/設定を表示
adadr	自デバイスアドレスとデバイスアドレスタイプを設定/設定を表示
adevt	Advertising イベントを設定/設定を表示
addtype	更新する Advertising データ配列タイプを指定/設定を表示
addata	Advertising データを設定/設定を表示
wlist	White List を設定/設定を表示
save	Advertising 動作設定、Advertising データ、White List をコードフラッシュのシステム動作設定領域へ保存
erase	コードフラッシュのシステム動作設定領域を消去
echo	コマンド入力のエコーバック可否を設定

5.2.3 ASCII 形式 UART 通信コマンド

ビーコンアプリケーション-ASCII コマンド制御の ASCII 形式コマンド仕様を示します。ホストマシンと評価ボードを USB で接続し、ホストマシンにインストールしたターミナルソフトを使用してコマンドを入力することによりビーコンアプリケーションを操作することができます。

動作確認の手順については 4.3.2 項「Advertising パケット送信の確認」を参照してください。

(1) RF 部への電源供給開始コマンド

構文	pwup
動作	RF 部への電源供給を開始し、ビーコンスタックを初期化する。 ※ビーコンアプリケーション-ASCII コマンド制御の起動直後は、RF 部への電源供給開始状態です。
実行条件	-
エラー	OK : 正常終了 ERR_PWRUP : RF 部への電源供給が供給状態
実行例	pwup OK

構文	pwup?
動作	RF 部への電源供給の状態を表示する。
表示形式	pwup? <power state> OK <power state>: POWER UP : RF 部への電源供給状態 POWER DOWN : RF 部への電源共有停止状態
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	pwup? POWER UP OK

(2) RF 部への電源供給停止コマンド

構文	pwdn
動作	RF 部への電源供給を停止する。
実行条件	-
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	pwdn OK

(3) Advertising 開始コマンド

構文	adstart_<adv_type>
動作	<adv_type>で指定した Advertising を開始する。 <adv_type>: n : Non-connectable Undirected Advertising s : Scannable Undirected Advertising
実行条件	RF 部へ電源が供給されている状態。 Advertising イベント"RBLE_EVT_PERMIT_ADV_STOP"が許可されている状態。
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ ERR_PWRDOWN : RF 部への電源供給が停止状態
実行例	adstart_n // Non-connectable Undirected Advertising を開始 OK

構文	adstart?
動作	Advertising の動作状態を表示する。
表示形式	adstart? <advertising state> OK <advertising state>: NON-CONNECTABLE : Non-connectable Undirected Advertising 動作状態 SCANNABLE : Scannable Undirected Advertising 動作状態 STOP : Advertising 停止状態
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	adstart? NON-CONNECTABLE OK

(4) Advertising 停止コマンド

構文	adstop
動作	Advertising を停止する。
実行条件	RF 部へ電源が供給されている状態。
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ ERR_PWRDOWN : RF 部の電源供給が停止状態 ERR_STOP : Advertising が停止状態
実行例	adstop OK

(5) Advertising インターバル設定コマンド

構文	<code>aditvl_<interval></code>
動作	Advertising インターバルを<interval>で指定した時間に設定する Advertising 動作中は設定した時間がすぐに反映される。 <interval>: 2 ~ 30720 : 2msec ~ 30.72s ※msec の単位で指定します。 小数点の指定はできません。
実行条件	-
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	<code>aditvl_1000 // Advertising インターバルを 1 秒に設定</code> OK

構文	<code>aditvl?</code>
動作	Advertising インターバルを表示する。
表示形式	<code>aditvl?</code> <interval> OK <interval>: 2 ~ 30720 : 2msec ~ 30.72s ※msec の単位で表示します。
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	<code>aditvl?</code> 1000 OK

(6) Advertising インターバルへのランダムディレイ加算設定コマンド

構文	addly_<delay>
動作	Advertising インターバルへのランダムディレイ加算を<delay>に設定する。 Advertising 動作中は設定がすぐに反映される。 <delay>: 0 : ランダムディレイを加算しない 1 : ランダムディレイを加算する
実行条件	-
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	addly_1 // Advertising インターバルへのランダムディレイを加算する OK

構文	addly?
動作	Advertising インターバルへのランダムディレイ加算の可否を表示する。
表示形式	addly? <delay> OK <delay>: 0 : ランダムディレイを加算しない 1 : ランダムディレイを加算する
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	addly? 1 OK

(7) Advertising チャンネル設定コマンド

構文	<code>adch_<ch_map></code>
動作	Advertising チャンネルを<ch_map>に設定する。 Advertising 動作中は設定がすぐに反映される。 <ch_map>: 37 : 37ch 38 : 38ch 39 : 39ch all : 全てのチャンネル ※複数 ch を指定する場合は','で区切ります。
実行条件	-
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	<code>adch_37,39</code> // Advertising チャンネルを 37ch、39ch に設定 OK

構文	<code>adch?</code>
動作	Advertising チャンネルを表示する。
表示形式	<code>adch?</code> <ch_map> OK <ch_map>: 37 : 37ch 38 : 38ch 39 : 39ch all : 全てのチャンネル
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	<code>adch?</code> 37,39 OK

(8) Advertising 回数制限設定コマンド

構文	<code>adloop_<loop_cnt></code>
動作	Advertising 回数制限を<loop_cnt>に設定する。 Advertising 動作中は設定がすぐに反映される。 <loop_cnt>: 0 ~ 255 ※10進数で指定します。 ※0を指定すると回数無制限で実行します。
実行条件	-
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	<code>adloop_10</code> // Advertising 回数制限を 10 回に設定 OK

構文	<code>adloop?</code>
動作	Advertising 回数制限を表示する。
表示形式	<code>adloop?</code> <loop_cnt> OK <loop_cnt>: 0 ~ 255 ※10進数で表示します。
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	<code>adloop?</code> 10 OK

(9) Advertising パケット送信パワー設定コマンド

構文	adtxpw_<tx_pwr>
動作	Advertising パケット送信パワーを<tx_pwr>に設定する。 Advertising 動作中は設定がすぐに反映される。 <tx_pwr>: 1 :-15dBm 2 :-10dBm 3 :-7dBm 4 :-2dBm 5 :(予約) 6 :(予約) 7 :-1dBm 8 :(予約) 9 :0dBm ※5, 6, 8 は設定禁止です。 ※数字 1 文字で指定します。
実行条件	-
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	adtxpw_2 // Advertising パケット送信パワーを-10dBm に設定 OK

構文	adtxpw?
動作	Advertising パケット送信パワーを表示する。
表示形式	adtxpw? <tx_pwr> OK <tx_pwr>: 1 ~ 9
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	adtxpw? 2 OK

(10) デバイスアドレス&デバイスアドレスタイプ設定コマンド

構文	adadr_ <own_addr_type><own_addr>
動作	<p>自デバイスのアドレスタイプを<own_addr_type>に設定する。 自デバイスのアドレスを<own_addr>に設定する。 Advertising 動作中は設定がすぐに反映される。</p> <p><own_addr_type>: pub : Public デバイスアドレス rnd : Random デバイスアドレス</p> <p><own_addr>: 16 進数の 12 文字</p>
実行条件	-
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	adadr_pub74905000c991 // デバイスアドレスを Public の 70:90:50:00:C9:91 に設定 OK

構文	adadr?
動作	自デバイスのデバイスアドレスとアドレスタイプを表示する。
表示形式	adadr? <own_addr_type><own_addr> OK
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	adadr? pub74905000c991 OK

(11) Advertising イベント設定コマンド

構文	adevt_<evt_permit>
動作	Advertising イベントを<evt_permit>に設定する。 Advertising 動作中は設定がすぐに反映される。 <evt_permit>: none : イベント通知を許可しない tx : RBLE_EVT_ADV_TX_IND イベント通知を許可 stop : RBLE_EVT_ADV_STOP_CMP イベント通知を許可 rx : RBLE_EVT_SCANREQ_RX_IND イベント通知を許可 all : 全イベントを許可 ※複数のイベントを指定する場合は','で区切ります。
実行条件	-
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	adevt_ tx, stop // RBLE_EVT_ADV_TX_IND イベントと OK // RBLE_EVT_ADV_STOP_CMP イベント通知を許可

構文	adevt?
動作	Advertising イベント通知設定を表示する。
表示形式	adevt? <evt_permit> OK <evt_permit>: none : イベント通知を許可しない tx : RBLE_EVT_ADV_TX_IND イベント通知を許可 stop : RBLE_EVT_ADV_STOP_CMP イベント通知を許可 rx : RBLE_EVT_SCANREQ_RX_IND イベント通知を許可 all : 全イベントを許可
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	adevt? tx, stop OK

(12) Advertising データタイプ設定コマンド

構文	addtype_<adv_type>
動作	Advertising データ配列のタイプを<adv_type>で設定する。 Advertising 停止中に Advertising データタイプ設定コマンド(adddata)で設定する Advertising データ配列を指定する。 <adv_type>: n : Non-connectable Undirected Advertising データ配列 s : Scannable Undirected Advertising データ配列
実行条件	Advertising が停止している状態。
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	addtype_n // Non-connectable Undirected Advertising データ配列を設定 OK

構文	addtype?
動作	Advertising データ配列のタイプを表示する。
表示形式	addtype? <adv_type> OK <adv_type>: NON-CONNECTABLE SCANNABLE
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	addtype? NON-CONNECTABLE OK

(13) Advertising データ設定コマンド

構文	<code>addata_<data_idx>,<data></code>
動作	<p>Advertising データ配列の要素<data_idx>に<data>を設定する。</p> <p>Advertising データを設定する前に、Advertising データタイプ設定コマンド(addtype)で Advertising データ配列のタイプ(Non-Connectable Undirected Advertising、または Scannable Undirected Advertising)を指定する。ただし、次の条件に従う。</p> <p>Advertising 停止中は、Advertising データタイプ設定コマンド(addtype)で指定した Advertising データ配列にデータを設定する。</p> <p>Advertising 動作中は、Advertising データタイプ設定コマンド(addtype)で指定した Advertising データ配列は無視され、動作中の Advertising データ配列がすぐに更新される。</p> <p>※ビーコンアプリケーション-ASCII コマンド制御の初期設定で設定可能な Advertising データの要素<data_idx>は下記になります。</p> <p>Non-connectable Undirected Advertising: <data_idx>=0 Scannable Undirected Advertising : <data_idx>=0,1</p> <p>設定可能な Advertising データの要素を増やす場合は、6.2.5 項「Advertising データ」を参照して adv_nonconn_data 構造体または adv_scan_data 構造体に Advertising データを追加してください。</p> <p><data_idx>: 0 ~ 9 : Advertising データ配列の要素番号 (数字 1 文字で指定する)</p> <p><data>: Advertising データ長 + Advertising データ文字列</p> <p>※Advertising データ長は、Advertising データ文字列のデータ長を表し、Advertising データ文字列長÷2 の値を指定します。</p> <p>※16 進数の文字列で指定します。</p>
実行条件	-
エラー	OK : 正常終了 ERR_PARAM : 不正なパラメータ
実行例	addata 0,0C020106080952656E65736173 OK

構文	addata?
動作	Advertising データを表示する。
表示形式	<p>addata? <adv_type> : Advertising データ配列のタイプ <data_idx>,<data> : RBLE_ADV_DATA 構造体で定義した Advertising データ配列の OK : 要素数分表示する。</p> <p><adv_type>: NON-CONNECTABLE SCANNABLE</p> <p><data_idx>: 0 ~ 9 : Advertising データ配列の要素番号 ※数字 1 文字で表示します。</p> <p><data>: Advertising データ長 + Advertisng データ文字列 ※16 進数の文字列で表示します</p>
エラー	<p>OK : 正常終了 ERR_PARAM : 不正なパラメータ</p>
実行例	<p>addata? SCANNABLE 0,0C020106080952656E65736173 1,19180952656e6573617320524c37382f47314420426561636f6e OK</p>

(14) White List 設定コマンド

構文	<pre>1. wlist_<wl_idx>,<addr_type><addr> 2. wlist_<flush> 3. wlist_<use_wl></pre>
動作	<p>1. White List 配列の要素<wl_idx>にアドレスタイプ<addr_type>のアドレス<addr>を設定する。 <wl_idx>: 0 ~ F : White List 配列の要素番号 ※16進数 1文字で指定します。</p> <p><addr_type>: pub : Public デバイスアドレス rnd : Random デバイスアドレス</p> <p><own_addr>: 16進数の 12文字</p> <p>※ビーコンアプリケーション-ASCII コマンド制御の初期設定で設定可能な White List の要素 <data_idx>は下記になります。 <data_idx>=0</p> <p>設定可能な White List 要素を増やす場合は、6.2.7 項「White List 設定」を参照して wl_info 構造体にデバイスアドレスを追加してください。</p> <p>2. White List をクリアし、White List を動作禁止に設定する。 ※Advertising データ保存コマンド(save)を実行することでコードフラッシュのシステム動作設定領域に保存されている White List をクリアすることができます。システム再起動後にプログラムで設定した BLE_DEV_INFO 構造体(White List)の初期値が使用されます。</p> <p>3. White List の動作を<use_wl>で設定する。Advertising 動作中は設定がすぐに反映される。 <use_wl>: 0 : White List 動作禁止 1 : White List 動作許可</p> <p>※White List は Scannable Undirected Advertising の場合に有効です。</p>
実行条件	-
エラー	<pre>OK : 正常終了 ERR_PARAM : 不正なパラメータ</pre>
実行例	<pre>1. wlist_0,pub74905000C991 OK 2. wlist_<flush> OK 3. wlist_<e> OK</pre>

構文	wlist?
動作	White List を表示する。
表示形式	<pre>wlist? <use_wl> <wl_idx>,<addr_type><addr> OK <use_wl>: 0 : White List 動作禁止 1 : White List 動作許可 <wl_idx>: 0 ~ f : White List 配列の要素番号 ※16進数 1 文字で表示します。 <addr_type>: pub : Public デバイスアドレス rnd : Random デバイスアドレス <addr>: 16進数の 12 文字</pre>
エラー	<pre>OK : 正常終了 ERR_PARAM : 不正なパラメータ</pre>
実行例	<pre>wlist? 0 0, pub74905000c991 OK</pre>

(15) Advertising データ保存コマンド

構文	save
動作	現在の Advertising 動作設定をコードフラッシュのシステム動作設定領域に保存する。
実行条件	Advertising が停止している状態。
エラー	OK : 正常終了 ERR_START : Advertising が動作中 ERR_SAVE : 保存失敗
実行例	save OK

(16) Advertising データ消去コマンド

構文	erase
動作	コードフラッシュのシステム動作設定領域を消去する。 システム再起動後にプログラムで設定した RBLE_ADV_INFO 構造体(Advertising 情報)、 RBLE_ADV_DATA 構造体(Advertising データ)、RBLE_DEV_INFO 構造体(White List)の初期値が使用され ます。
実行条件	Advertising が停止している状態。
エラー	OK : 正常終了 ERR_START : Advertising が動作中
実行例	erase OK

(17) エコーバックコマンド

構文	echo
動作	コマンド入力のエコーバック可否を設定する。
表示形式	echo <echo_back> OK <echo_back>: 0 : エコーバック禁止 1 : エコーバック許可
実行条件	-
エラー	OK : 正常終了 ERR_START : Advertising が動作中
実行例	echo 0 OK

5.3 スキャンアプリケーション

スキャンアプリケーションは、Advertising パケットを受信するための Scanning を実行します。ビーコンデバイスからの Advertising パケットを受信すると、受信チャンネルや RSSI、Advertising パケットに格納された情報を UART から送信します。またコマンドを UART から受信するとスキャンアプリケーションは Scanning の開始・停止と設定の変更を実行します。

またスキャンアプリケーションには重複フィルタリングと RSSI フィルタリングを実行可能です。重複フィルタリングを有効化すると、同一のデバイスアドレスと Advertising データを含む同一 Advertising パケットを再度受信した場合、ホストには通知しません。また RSSI フィルタリングを有効化すると、受信した Advertising パケットの RSSI が閾値未満の場合、ホストには通知しません。

スキャンアプリケーションは ASCII 形式またはバイナリ形式の UART 送受信に対応します。

5.3.1 ASCII 形式 UART 通信コマンド

スキャンアプリケーションに実装された ASCII 形式 UART 通信の仕様を示します。本通信はホストマシンと評価ボードを USB で接続し、ホストマシンにインストールしたターミナルソフトを使用して、ユーザが手入力ですキャンアプリケーションを操作することを想定します。

動作確認の手順については 4.3.3 項「Advertising パケット受信の確認」を参照してください。

(1) Start Scan コマンド

構文	[Enter] キー押下のみ
動作	Scanning を開始する。 Scanning の開始が成功すると、「Start Scan :OK」と表示される。 Scanning の開始が失敗すると、「Start Scan :ER」と表示される。
実行条件	Scanning の停止中
実行例	[Enter] キー押下 // Scanning を開始 Start Scan :OK

(2) Stop Scan コマンド

構文	[Enter] キー押下のみ
動作	Scanning を停止する。 Scanning の停止が成功すると、「Stop Scan :OK」と表示される。 Scanning の停止が失敗すると、「Stop Scan :ER」と表示される。
実行条件	Scanning の実行中
実行例	[Enter] キー押下 // Scanning を停止 Stop Scan :OK

(3) Set Scan Type コマンド

構文	type active type passive
動作	Scan タイプを設定する。 Active Scan を設定する場合は、「type active」と入力する。 Passive Scan を設定する場合は、「type passive」と入力する。 コマンド文字列は全て小文字で入力する。 Active Scan タイプの設定が成功すると、「Set Active :OK」と表示される。 Passive Scan タイプの設定が成功すると、「Set Passive :OK」と表示される。 コマンドに誤りがある場合、何も表示されない。
実行条件	Scanning の停止中
実行例	type active // Active Scan を設定する場合 Set Active :OK type passive // Passive Scan を設定する場合 Set Passive :OK

(4) Set Scan Channel コマンド

構文	ch 37 ch 38 ch 39 ch all
動作	Scan チャンネルを設定する。 37ch を設定する場合は、「ch 37」と入力する。 38ch を設定する場合は、「ch 38」と入力する。 39ch を設定する場合は、「ch 39」と入力する。 全 ch(37,38,39)を設定する場合は、「ch all」と入力する。 コマンド文字列は全て小文字で入力する。 37ch の設定が成功すると、「Set Channel 37 :OK」と表示される。 38ch の設定が成功すると、「Set Channel 38 :OK」と表示される。 39ch の設定が成功すると、「Set Channel 39 :OK」と表示される。 全 ch の設定が成功すると、「Set Channel All :OK」と表示される。 コマンドに誤りがある場合、何も表示されない。
実行条件	Scanning の停止中
実行例	ch 37 // 37ch を設定する場合 Set Channel 37 :OK ch all // 全 ch(37,38,39)を設定する場合 Set Channel All :OK

(5) Set Scan Interval コマンド

構文	itvl <Scan Interval>
動作	Scan インターバルを設定する。 Scan インターバル(0.625msec 単位)は 16 進数(XXXX)で指定し、「itvl XXXX」と入力する。 コマンド文字列と 16 進数は全て小文字で入力する。 Scan インターバルの設定が成功すると、「Set Interval :OK」と表示される。 コマンドに誤りがある場合、何も表示されない。
実行条件	Scanning の停止中
実行例	itvl a0 // 100msec(0x00A0)を設定する場合 Set Interval :OK itvl 640 // 1sec(0x0640)を設定する場合 Set Interval :OK

(6) Flush White List コマンド

構文	wlist flush
動作	ホワイトリストを消去する。 コマンド文字列は全て小文字で入力する。 ホワイトリストの消去が成功すると、「Flush White List :OK」と表示される。 コマンドに誤りがある場合、何も表示されない。
実行条件	Scanning の停止中
実行例	wlist flush // ホワイトリストを消去 Flush White List :OK

(7) Add White List コマンド

構文	wlist <Device Address Type><Device Address>
動作	ホワイトリストにデバイスアドレスを追加する。 デバイスアドレスタイプは Public の場合「pub」、Random の場合「rnd」で指定する。 デバイスアドレスは 12 桁の 16 進数(XXXXXXXXXXXX)で指定し、「wlist pubXXXXXXXXXXXX」または「wlist rndXXXXXXXXXXXX」と入力する。 コマンド文字列と 16 進数は全て小文字で入力する。 ホワイトリストへのデバイスアドレス追加が成功すると、「Add White List :OK」と表示される。 コマンドに誤りがある場合、何も表示されない。
実行条件	Scanning の停止中
実行例	wlist pub123456789abc // Public デバイスアドレスの 12:34:56:78:9A:BC を追加 Add White List :OK wlist rnd47f2bb2c2a79 // Random デバイスアドレスの 47:F2:BB:2C:2A:79 を追加 Add White List :OK

(8) Duplicate Filter コマンド

構文	dup en dup dis
動作	Advertiser's Address と Advertising データの重複フィルタリングを有効化または無効化する。 重複フィルタリングを有効化する場合は、「dup en」と入力する。 重複フィルタリングを無効化する場合は、「dup dis」と入力する。 コマンド文字列は全て小文字で入力する。 重複フィルタリング有効化の設定が成功すると、「Enable Dup Filter :OK」と表示される。 重複フィルタリング無効化の設定が成功すると、「Disable Dup Filter :OK」と表示される。 コマンドに誤りがある場合、何も表示されない。
実行条件	Scanning の停止中
実行例	dup en // 重複フィルタリングを有効化 Enable Dup Filter :OK dup dis // 重複フィルタリングを無効化 Disable Dup Filter :OK

(9) RSSI Filter コマンド

構文	rssi en <RSSI Threshold> rssi dis
動作	パケットの RSSI フィルタリングを有効化または無効化する。 RSSI フィルタリングを有効化する場合は、RSSI 閾値を-128dBm~127dBm の 10 進数(XX)で表記し、「rssi en XX」と入力する。 RSSI フィルタリングを無効化する場合は、「rssi dis」と入力する。 コマンド文字列は全て小文字で入力する。 RSSI フィルタリング有効化の設定が成功すると、「Enable RSSI Filter :OK」と表示される。 RSSI フィルタリング無効化の設定が成功すると、「Disable RSSI Filter :OK」と表示される。 コマンドに誤りがある場合、何も表示されない。
実行条件	Scanning の停止中
実行例	rssi en -70 // RSSI フィルタリングを有効化し、RSSI 閾値を-70dBm に設定 Enable RSSI Filter :OK rssi dis // RSSI フィルタリングを無効化 Disable RSSI Filter :OK

スキャンアプリケーションは起動後、自動的に Scanning を開始します。ターミナルソフト上で Enter キーを入力すると Scanning を停止し、再度 Enter キーを入力すると Scanning を再開します。

Scanning 実行中、RL78/G1D が Advertising パケットを受信すると、Advertising パケットの受信結果をターミナルソフト上に表示します。

Scanning 停止中、ターミナルソフト上でのコマンド入力で Scan タイプ、Scan チャネル、Scan インターバル、White List、重複フィルタ、RSSI フィルタを設定します。

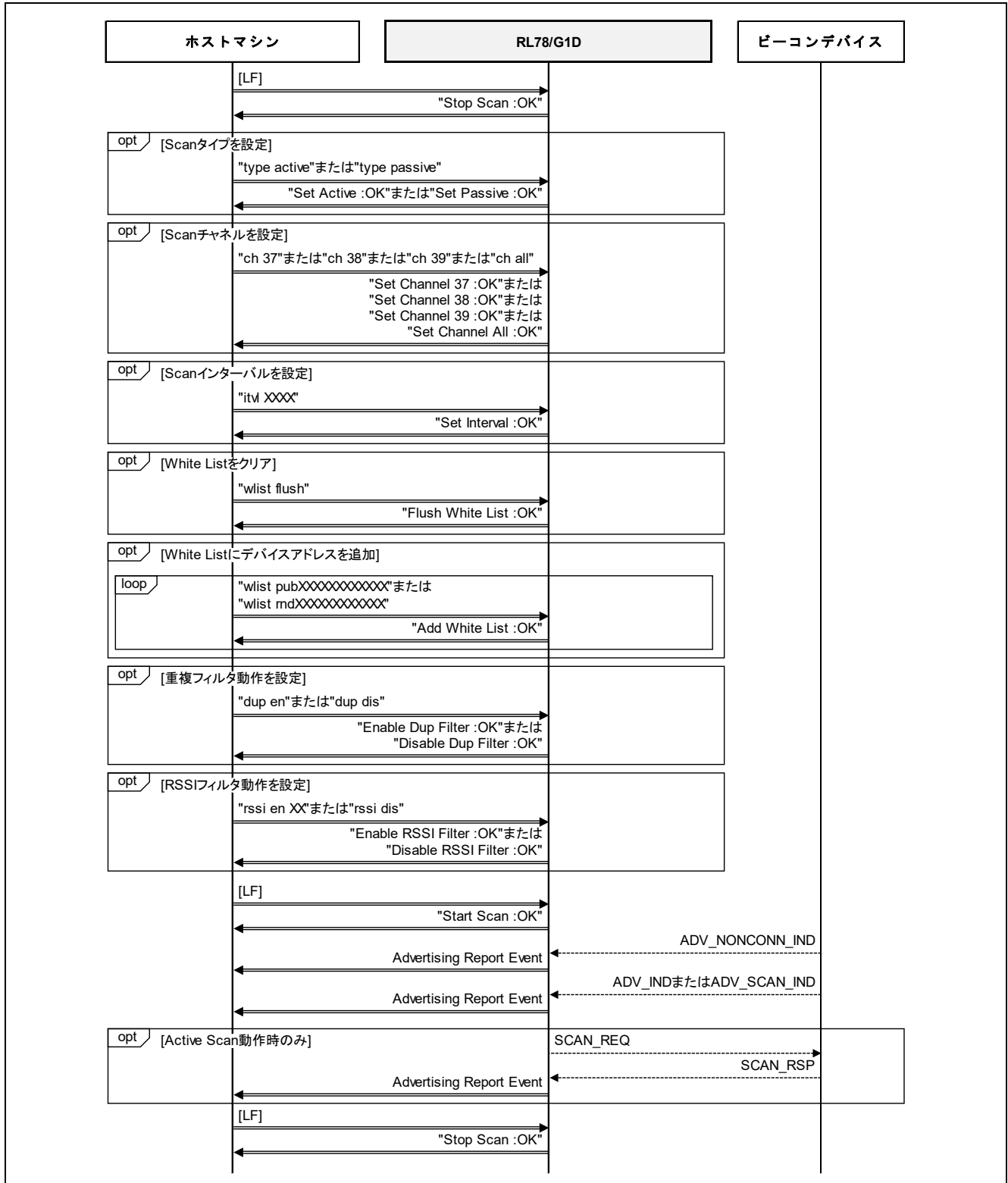


図 5-1 ASCII 形式 UART 通信シーケンス

5.3.2 バイナリ形式 UART 通信コマンド

スキャンアプリケーションに実装されたバイナリ形式 UART 通信の仕様を示します。本通信はホスト MCU と評価ボードを UART で接続し、ホスト MCU からスキャンアプリケーションを制御することを想定します。

バイナリ形式の UART 通信設定を表 5-6 に示します。

表 5-6 バイナリ形式の UART 通信設定

設定項目	設定値
ボーレート	1,000,000bps
データ長	8bit
パリティ	None
ストップビット	1bit
フロー制御	None

バイナリ形式 UART 通信パッケージフォーマットを図 5-2 に示します。フォーマットの詳細は次頁以降を参照してください。

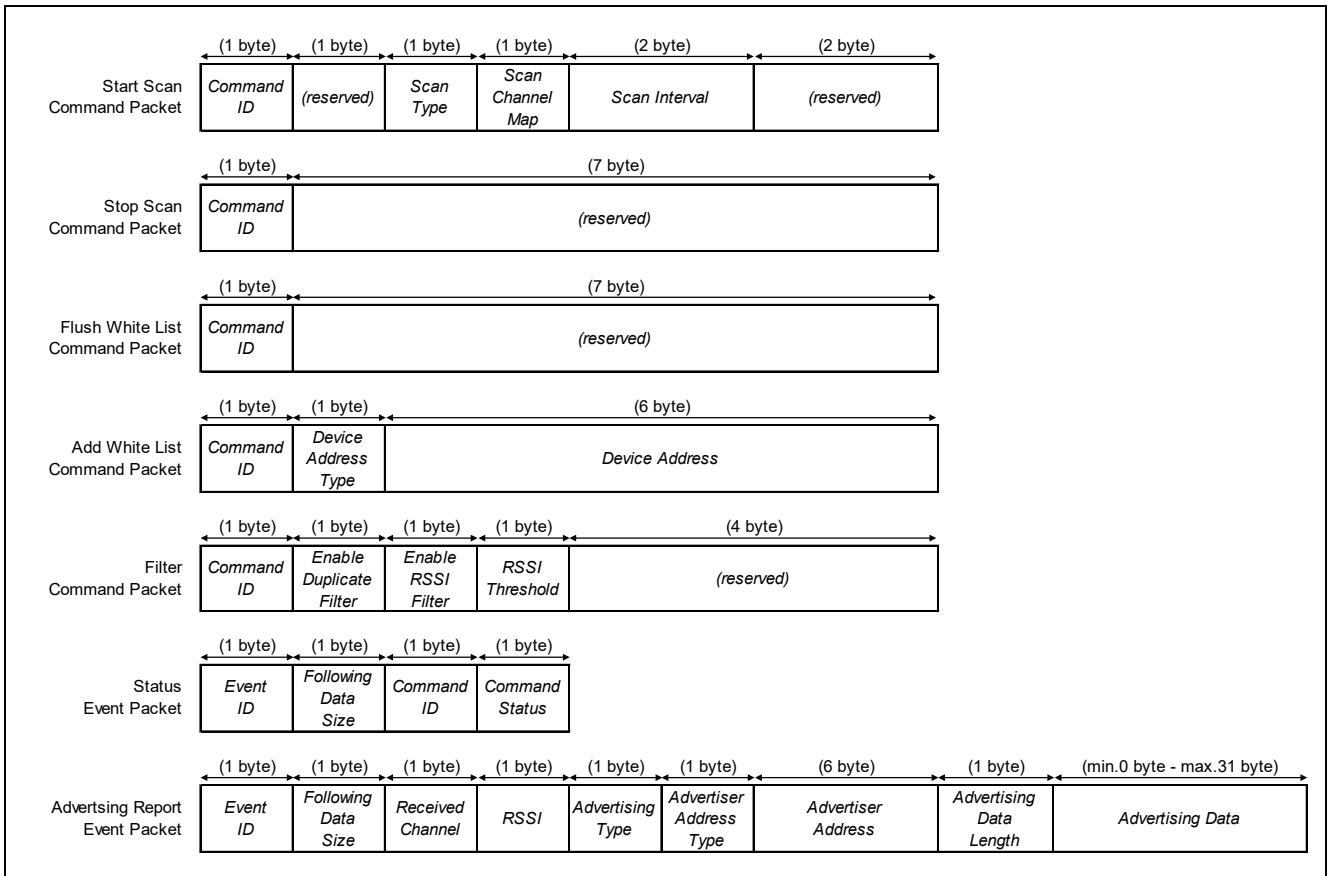


図 5-2 バイナリ形式 UART 通信パッケージフォーマット

(1) Start Scan コマンド

RL78/G1D の Scanning を開始します。Start Scan コマンドフォーマットを表 5-7 に示します。

表 5-7 Start Scan コマンドフォーマット

オフセット	サイズ	データ	値
0x00	0x01	Command ID	0x01: Start Scan
0x01	0x01	(reserved)	-
0x02	0x01	Scan Type	0x00: Passive Scan, 0x01: Active Scan
0x03	0x01	Scan Channel Map	0x01: 37ch, 0x02: 38ch, 0x04: 39ch, 0x07: All ch
0x04	0x02	Scan Interval	最小0x0004(2.5msec)~最大0xC000(30.72sec) バイト順序 : Least Significant Byte First
0x06	0x02	(reserved)	-
0x08	-	-	-

Start Scan コマンド例

	+0	+1	+2	+3	+4	+5	+6	+7
0000	01	00	01	07	A0	00	00	00

(2) Stop Scan コマンド

RL78/G1D の Scanning を停止します。Stop Scan コマンドフォーマットを表 5-8 に示します。

表 5-8 Stop Scan コマンドフォーマット

オフセット	サイズ	データ	値
0x00	0x01	Command ID	0x02: Stop Scan
0x01	0x07	(reserved)	-
0x08	-	-	-

Stop Scan コマンド例

	+0	+1	+2	+3	+4	+5	+6	+7
0000	02	00	00	00	00	00	00	00

(3) Flush White List コマンド

RL78/G1D が保持する White List を消去します。Flush White List コマンドフォーマットを表 5-9 に示します。

表 5-9 Flush White List コマンドフォーマット

オフセット	サイズ	データ	値
0x00	0x01	Command ID	0x03: Flush White List
0x01	0x07	(reserved)	-
0x08	-	-	-

Flush White List コマンド例

	+0	+1	+2	+3	+4	+5	+6	+7
0000	03	00	00	00	00	00	00	00

(4) Add White List コマンド

RL78/G1D の White List に受信対象とするデバイスアドレスを追加します。Add White List コマンドフォーマットを表 5-10 に示します。

表 5-10 Add White List コマンドフォーマット

オフセット	サイズ	データ	値
0x00	0x01	Command ID	0x04: Add White List
0x01	0x01	Device Address Type	0x00: Public, 0x01: Random
0x02	0x06	Device Address	バイト順序 : Least Significant Byte First 例) Device Addressが12:34:56:78:9A:BCの場合 0xBC,0x9A,0x78,0x56,0x34,0x12の順序で設定
0x08	-	-	-

Add White List コマンド例

	+0	+1	+2	+3	+4	+5	+6	+7
0000	04	00	BC	9A	78	56	34	12

(5) Filter コマンド

RL78/G1D のフィルタリング動作を設定します。Filter コマンドフォーマットを表 5-11 に示します。

表 5-11 Filter コマンドフォーマット

オフセット	サイズ	データ	値
0x00	0x01	Command ID	0x05: Filter
0x01	0x01	Enable Duplicate Filter	0x00: Disable, 0x01: Enable
0x02	0x01	Enable RSSI Filter	0x00: Disable, 0x01: Enable
0x03	0x01	RSSI Threshold	最小0x80(-128dBm)~最大0x7F(127dBm)
0x04	0x04	(reserved)	-
0x08	-	-	-

Filter コマンド例

	+0	+1	+2	+3	+4	+5	+6	+7
0000	05	01	01	D3	00	00	00	00

(6) Status イベント

RL78/G1D のコマンド実行結果を通知します。Status イベントフォーマットを表 5-12 に示します。

表 5-12 Status イベントフォーマット

オフセット	サイズ	データ	値
0x00	0x01	Event ID	0x10: Status
0x01	0x01	Following Data Size	0x02(byte)
0x02	0x01	Command ID	0x01: Start Scan 0x02: Stop Scan 0x03: Flush White List 0x04: Add White List 0x05: Filter
0x03	0x01	Command Status	『RL78/G1Dビーコンスタック ユーザーズマニュアル』 (R01UW0171)の4.2.1項「ステータスマクロ」を参照
0x04	-	-	-

Status イベント例

	+0	+1	+2	+3
0000	10	02	01	00

(7) Advertising Report イベント

RL78/G1D が Scanning 実行中に受信した Advertising パケットの受信結果を通知します。Advertising Report イベントフォーマットを表 5-13 に示します。

表 5-13 Advertising Report イベントフォーマット

オフセット	サイズ	データ	値
0x00	0x01	Event ID	0x20: Advertising Report
0x01	0x01	Following Data Size	最小0x0B(byte)~最大0x2A(byte)
0x02	0x01	Received Channel	0x25: 37ch, 0x26: 38ch, 0x27: 39ch
0x03	0x01	RSSI	最小0x80(-128dBm)~最大0x7F(127dBm)
0x04	0x01	Advertising Type	0x00: ADV_IND 0x01: ADV_DIRECT_IND 0x02: ADV_NONCONN_IND 0x04: SCAN_RSP 0x06: ADV_SCAN_IND
0x05	0x01	Advertiser Address Type	0x00: Public, 0x01: Random
0x06	0x06	Advertiser Address	バイト順序 : Least Significant Byte First 例) Advertiser Addressが12:34:56:78:9A:BCの場合 0xBC,0x9A,0x78,0x56,0x34,0x12の順序で通知
0x0C	0x01	Advertising Data Length	最小0x00(byte)~最大0x1F(byte)
0x0D	0x00-0x1F	Advertising Data	-
0x0D-0x2C	-	-	-

Advertising Report イベント例

	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
0000	20	26	25	DB	02	00	B1	9A	78	56	34	12	1B	02	01	06
0010	03	03	AA	FE	13	16	AA	FE	10	EE	02	67	6F	6F	2E	67
0020	6C	2F	35	77	4B	6B	52	4B								

ホスト MCU が Start Scan コマンドを発行すると、RL78/G1D は Scanning を開始します。同様に、ホスト MCU が Stop Scan コマンドを発行すると、RL78/G1D は Scanning 動作を停止します。

Scanning 実行中、RL78/G1D が Advertising パケットを受信すると、Advertising Report イベントをホスト MCU に通知します。

Scanning 停止中、ホスト MCU が Flush White List コマンドまたは Add White List コマンドを発行すると、RL78/G1D は White List を設定します。またホスト MCU が Filter コマンドを発行すると、RL78/G1D はフィルタリング動作を設定します。

ホスト MCU は RL78/G1D へのコマンド発行後、RL78/G1D から Status イベントが通知されることを確認してください。RL78/G1D から Status イベントが通知されない場合、コマンドを再度発行してください。

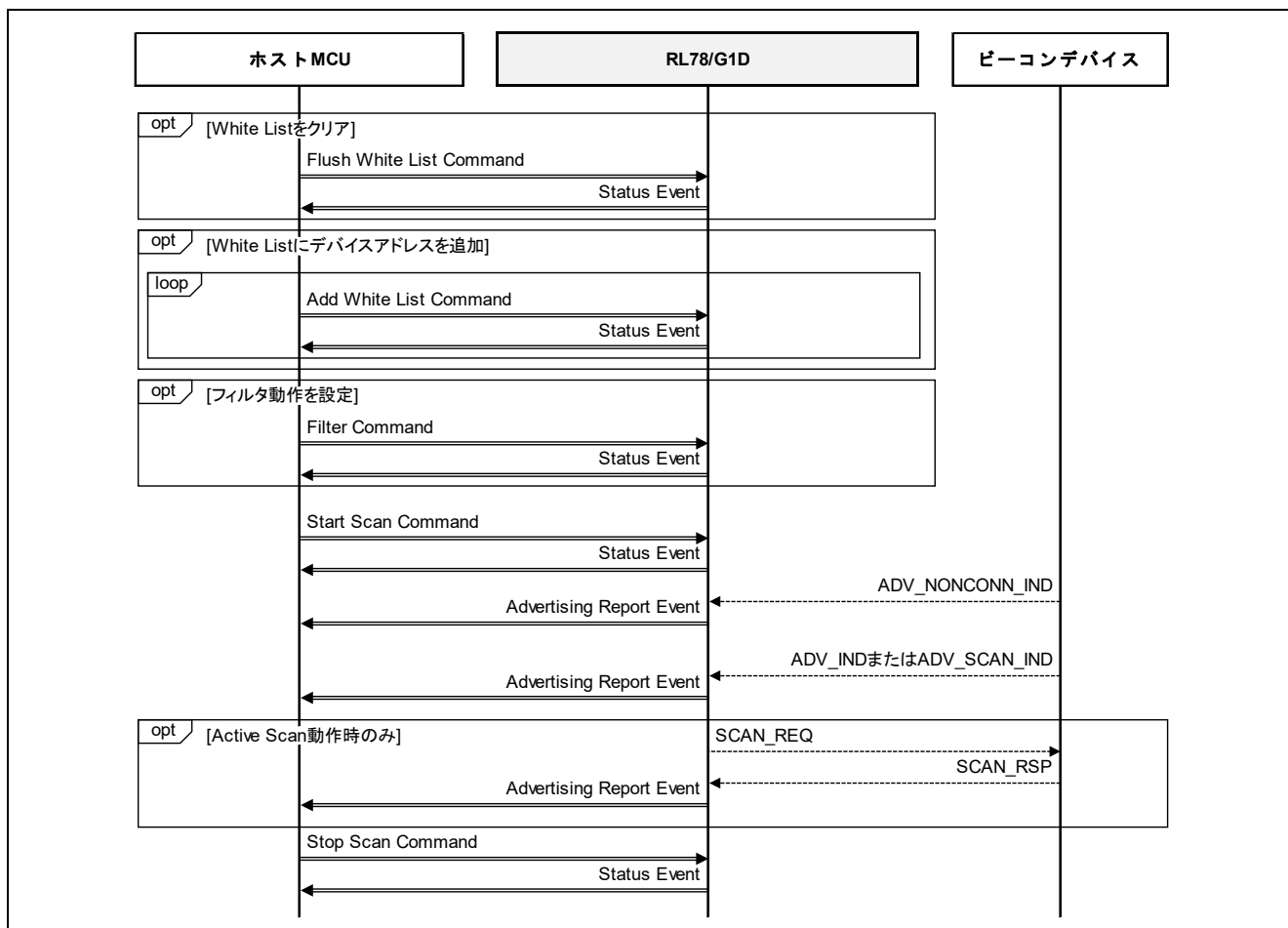


図 5-3 バイナリ形式 UART 通信シーケンス

5.4 DTM アプリケーション

DTM アプリケーションは、RF 特性を評価するための Direct Test Mode を実行します。Direct Test Mode の制御には RF テスタと評価ボードを UART で接続し、UART 経由で RF テストパケットを送受信します。

5.4.1 Direct Test Mode

RF テストパケットフォーマットを図 5-4 に示します。

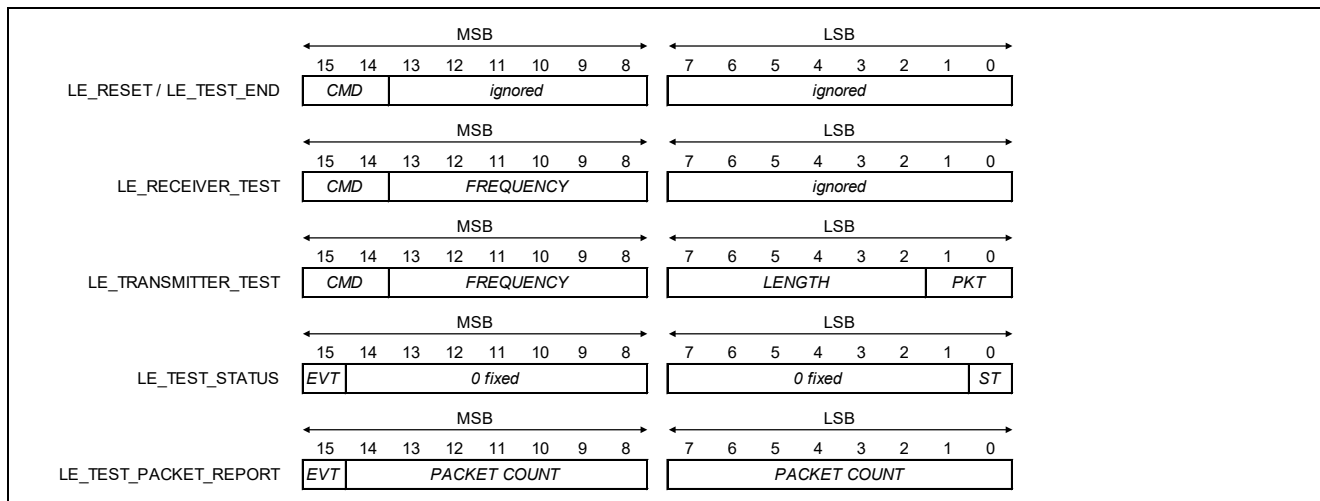


図 5-4 RF テストパケットフォーマット

Direct Test Mode を実行するための RF テストコマンドを表 5-14 に示します。

表 5-14 RF テストコマンドパケットフォーマット

RF テストコマンド	パラメータ		
LE_RESET	参照しない		
LE_RECEIVER_TEST	FREQUENCY	受信周波数 : (2*FREQUENCY+2402) MHz FREQUENCY=0x00(2402MHz)~0x27(2480MHz)	
LE_TRANSMITTER_TEST	FREQUENCY	送信周波数 : (2*FREQUENCY+2402) MHz FREQUENCY=0x00(2402MHz)~0x27(2480MHz)	
	LENGTH	送信パケットペイロード長 LENGTH=0x00-0x25 Bytes	
	PKT	送信パケットペイロードタイプ	
		0	9-bit 擬似乱数シーケンス(PRBS9)
		1	b'11110000 ビット列シーケンス
		2	b'10101010 ビット列シーケンス
		3	15-bit 擬似乱数シーケンス(PRBS15)
		4	b'11111111 ビット列シーケンス
5		b'00000000 ビット列シーケンス	
6	b'00001111 ビット列シーケンス		
7	b'01010101 ビット列シーケンス		
LE_TEST_END	参照しない		

Direct Test Mode の実行結果を応答するための RF テストイベントを表 5-15 に示します。

表 5-15 RF テストイベントパケットフォーマット

RF テストイベント	パラメータ		
LE_TEST_STATUS	ST	ステータス	
		0	Success
		1	Error
LE_TEST_PACKET_REPORT	PACKET COUNT	受信パケット数:0-32767	

RL78/G1D は RF テスタからの LE_RECEIVER_TEST コマンドを受信すると、RF 受信テストを開始し、RF テスタからの LE テストパケットを受信します。RL78/G1D は RF テスタからの LE_TEST_END コマンドを受信すると、LE_PACKET_REPORT イベントを送信し、受信パケット数を通知します。

RL78/G1D は RF テスタからの LE_TRANSMITTER_TEST コマンドを受信すると、RF 送信テストを開始し、LE テストパケットを送信します。RL78/G1D は RF テスタからの LE_TEST_END コマンドを受信すると、LE テストパケットの送信を停止し、LE_PACKET_REPORT イベントを送信します。

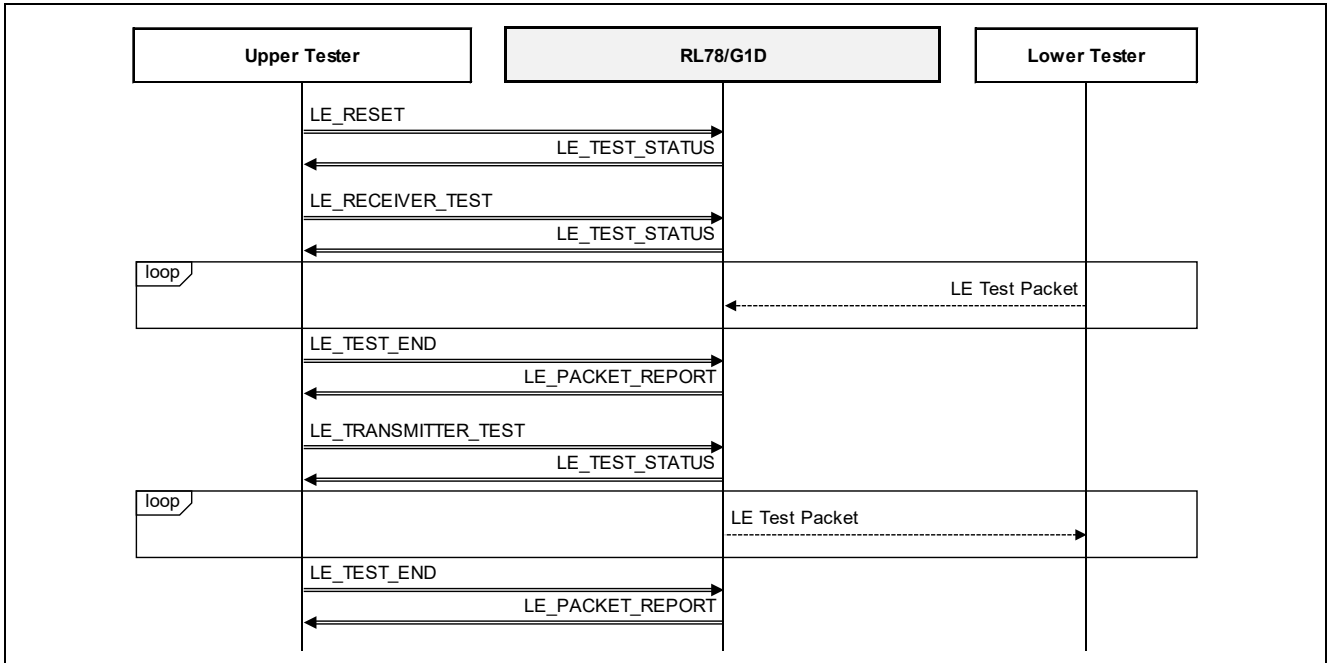


図 5-5 2-wire UART Direct Test Mode 通信シーケンス

Direct Test Mode の仕様の詳細については『Bluetooth Core Specification v4.2』の[Vol. 6, Part F] Section 3.3 を参照してください。

5.5 フラッシュメモリへのアクセス

5.5.1 コードフラッシュメモリへのアクセス

ビーコンアプリケーションとスキャンアプリケーションは、コードフラッシュメモリのサンプルプログラムファームウェア外部の一部を、システム動作設定を配置するための領域として使用します。システム動作設定には、デバイスごとに異なる必要のあるパラメータを格納します。

ビーコンアプリケーションはデバイスアドレス、デバイスアドレスタイプ、Advertising 情報をシステム動作設定から読み込み、Advertising を開始します。

スキャンアプリケーションはデバイスアドレス、デバイスアドレスタイプをシステム動作設定から読み込み、Scanning を開始します。

システム動作設定の仕様を表 5-16 に示します。システム動作設定の配置については 5.10 節「アドレスマップ」を参照してください。

表 5-16 コードフラッシュメモリのシステム動作設定

オフセット	データ	サイズ	読み込み (○:読み込む, ×:読み込まない)		
			ビーコン アプリケーション	スキャン アプリケーション	
0x00	デバイスアドレス (RBLE_BD_ADDR 構造体)	6 byte	○	○	
0x06	デバイスアドレスタイプ 0x00: public, 0x01: random (uint8_t 型)	1 byte	○	○	
0x07	(reserved)	1 byte	×	×	
0x08	Advertising 情報 (RBLE_ADV_INFO 構造体)	18 byte			
		interval	2 byte	○	×
		delay	1 byte	○	×
		ch_map	1 byte	○	×
		loop_cnt	1 byte	○	×
		tx_pwr	1 byte	○	×
		own_addr	6 byte	×	×
		own_addr_type	1 byte	×	×
		data_cnt	1 byte	×	×
		data	2 byte	×	×
		evt_permit	1 byte	×	×
		use_wl	1 byte	×	×
0x1A	Non-connectable Undirected Advertising パケットデータ (RBLE_ADV_DATA 構造体)	32 byte			
		len	1 byte	○	×
		data	31 byte	○	×
0x3A	Scannable Undirected Advertising パケットデータ (RBLE_ADV_DATA 構造体)	32 byte			
		len	1 byte	○	×
		data	31 byte	○	×
0x5A	Scan Response パケットデータ (RBLE_ADV_DATA 構造体)	32 byte			
		len	1 byte	○	×
		data	31 byte	○	×
0x7A	-	-	-	-	

5.5.2 コードフラッシュメモリへのアクセス (ビーコンアプリケーション-ASCII コマンド制御)

ビーコンアプリケーション-ASCII コマンド制御は、コードフラッシュメモリの1ブロックを、システム動作設定を配置するための領域として使用します。システム動作設定には、デバイスごとに異なる必要のあるパラメータを格納します。

システム動作設定の領域に設定が保存されている場合、ビーコンアプリケーション-ASCII コマンド制御はデバイスアドレス、デバイスアドレスタイプ、Advertising 情報、Advertising パケットデータ、White List を読み込みます。保存されていない場合は、アプリケーションの初期値を使用します。

システム動作設定の仕様を表 5-17 に示します。システム動作設定の配置については 6.1.8 項「システム動作設定アドレスアドレスマップ」を参照してください。

表 5-17 コードフラッシュメモリのシステム動作設定
(ビーコンアプリケーション-ASCII コマンド制御)

オフセット	データ	サイズ	備考	
0x00	デバイスアドレス (RBLE_BD_ADDR 構造体)	6 byte	アプリケーション起動時にデバイスアドレスとデバイスアドレスタイプを Advertising 情報の own_addr と own_addr_type にセットします。	
0x06	デバイスアドレスタイプ 0x00: public, 0x01: random (uint8_t 型)	1 byte		
0x07	(reserved)	1 byte	-	
0x08	Advertising 情報 (RBLE_ADV_INFO 構造体)	18 byte	-	
	interval	2 byte	デバイスアドレス(オフセット 0x00)、デバイスアドレスタイプ(オフセット 0x06)がセットされます。	
	delay	1 byte		
	ch_map	1 byte		
	loop_cnt	1 byte		
	tx_pwr	1 byte		
	own_addr	6 byte		
	own_addr_type	1 byte		
	data_cnt	1 byte		Advertising 開始時に指定された Advertising パケットデータが設定されます。
	data	2 byte		
evt_permit	1 byte	-		
use_wl	1 byte			
0x1A	Non-connectable Undirected Advertising パケットデータ (RBLE_ADV_DATA 構造体)	320 byte	最大 10 個の Non-connectable Undirected Advertising パケットデータを保存することができます。	
	len	1 byte		
	data	31 byte		
	:	-		
0x15A	Scannable Undirected Advertising パケットデータ (RBLE_ADV_DATA 構造体)	320 byte	最大 10 個の Scannable Undirected Advertising パケットデータを保存することができます。(Advertising データ 5 個と Scan Response データで 5 個です。)	
	len	1 byte		
	data	31 byte		
	:	-		
0x29A	White List (RBLE_DEV_INFO 構造体)	128 byte	最大 16 個の White List を保存することができます。	
	dev_type	1 byte		
	reserved	1 byte		
	RBLE_BD_ADDR	6 byte		
	:	-		

5.6 使用ハードウェアリソース

サンプルプログラムのデフォルトでの使用ハードウェアリソースを示します。

RL78/G1D MCU 部	
クロック発生回路	共通 <ul style="list-style-type: none"> MCU 部メイン・システム・クロックとして、高速オンチップ・オシレータの 8MHz クロックを使用
クロック出力/ブザー出力	共通 <ul style="list-style-type: none"> PCLBUZ0 端子から XT1 発振クロックを出力しない
タイマ・アレイ・ユニット	ビーコンスタック <ul style="list-style-type: none"> TM00 使用、動作クロック CK00 を 1MHz に設定 スキャンアプリケーション、DTM アプリケーション TM01 使用、動作クロック CK01 を 125kHz に設定
シリアル・アレイ・ユニット	ビーコンスタック <ul style="list-style-type: none"> CSI21 を使用 スキャンアプリケーション、DTM アプリケーション、 ビーコンアプリケーション-ASCII コマンド制御 UART0 を使用
DMA コントローラ	ビーコンスタック <ul style="list-style-type: none"> DMA2、DMA3 を使用 スキャンアプリケーション、DTM アプリケーション DMA0、DMA1 を使用
割り込み	ビーコンスタック <ul style="list-style-type: none"> INTRF、INTDMA2、INTDMA3、INTTM00 を使用 ビーコンアプリケーション INTP5 を使用 スキャンアプリケーション INTP5、INTDMA0、INTDMA1、INTSR0、INTSRE0、INTST0、INTTM01 を使用 DTM アプリケーション INTDMA0、INTDMA1、INTSR0、INTSRE0、INTST0、INTTM01 を使用
ポート	共通 <ul style="list-style-type: none"> 評価ボードの DIP スイッチ SW6-1 入力に P10 を使用 評価ボードの DIP スイッチ SW6-4 入力に P02 を使用 評価ボードのスイッチ SW2 入力に P16 を使用 評価ボードの LED1,2,3,4 制御にそれぞれ P120,P147,P03,P60 を使用
RL78/G1D RF 部	
DC-DC コンバータ	RF 部内蔵 DC-DC コンバータを使用
スロー・クロック用オシレータ	RF 部内蔵オシレータを使用
クロック出力	CLKOUT_RF 端子からのクロック出力なし
RL78/G1D 評価ボード	
USB 通信機能	スキャンアプリケーション <ul style="list-style-type: none"> ホストマシンとのシリアル通信に USB インタフェースを使用
入力機能	共通 <ul style="list-style-type: none"> アプリケーションの選択に DIP スイッチ SW6-1 と SW6-4 を使用 ビーコンアプリケーション Advertising Type の切り換えにプッシュスイッチ SW2 を使用 スキャンアプリケーション (ASCII 形式 UART 通信) ペイロードデータの表記形式変更にはプッシュスイッチ SW2 を使用
表示機能	ビーコンアプリケーション、DTM アプリケーション <ul style="list-style-type: none"> アプリケーション開始表示に LED4 を使用 スキャンアプリケーション パケット受信表示に LED1-LED4 を使用

5.7 コンパイラ

ビーコンスタックは下記のコンパイラで生成されています。ビーコンスタックを使用するアプリケーションの開発は CC-RL コンパイラを使用してください。

コンパイラ : Renesas CC-RL V1.09.00

5.8 メモリモデル

ビーコンスタックのメモリモデルは、ミディアムモデルです。ビーコンスタックを使用するアプリケーションのコンパイルオプションでは下記を設定してください。

メモリモデル : -memory_model=medium

5.9 プログラムサイズ

サンプルプログラムの合計プログラムサイズを表 5-18 に示します。

ターゲットデバイス : R5F11AGJ
 コンパイラ : Renesas CC-RL V1.09.00
 コンパイル設定 : サンプルプログラムのデフォルト設定

表 5-18 サンプルプログラムの合計プログラムサイズ

ROM サイズ	26,687byte PROGRAM SECTION + ROMDATA SECTION
RAM サイズ	8,468byte RAMDATA SECTION (プログラムが関数コールやオート変数確保のために使用するスタックメモリは含みません)

ビーコンアプリケーション-ASCII コマンド制御のプログラムサイズを表 5-19 に示します。

ターゲットデバイス : R5F11AGJ
 コンパイラ : Renesas CC-RL V1.09.00
 コンパイル設定 : ビーコンアプリケーション-ASCII コマンド制御のデフォルト設定

表 5-19 ビーコンアプリケーション-ASCII コマンド制御プログラムサイズ

ROM サイズ	37,421byte PROGRAM SECTION + ROMDATA SECTION
RAM サイズ	5,566byte RAMDATA SECTION (プログラムが関数コールやオート変数確保のために使用するスタックメモリは含みません)

セクション仕様については『CC-RL コンパイラ ユーザーズマニュアル』(R20UT3123)の6章「セクション仕様」を参照してください。

5.10 アドレスマップ

サンプルプログラム(ビーコンアプリケーション-ASCII コマンド制御は除く)の RL78/G1D(R5F11AGG)でのアドレスマップを図 5-6 に示します。

なお下線部は R5F11AGH、R5F11AGJ の値と異なることを示します。

アドレス	領域サイズ	セクション	セクション名
	<u>131,072byte</u>	コードフラッシュメモリ	
0x00000	128byte	ベクタテーブル領域	.vect
0x00080	64byte	CALLT テーブル領域	.callt0
0x000C0	4byte	オプション・バイト領域	.option_byte
0x000C4	10byte	セキュリティ ID 設定領域	.security_id
0x000CE	<u>129,842byte</u>	プログラム領域 (セクション配置は順不同)	
		OCD モニタ	.monitor1
		スタートアップ	BOOT0_TEXT
		ランタイムライブラリ	.RLIB
		標準ライブラリ	.SLIB
		ビーコンスタック	BCN_CONST, BCN_TEXT
		アプリケーション	.const, .constf, .data, .text, .textf
		未使用	-
0x1FC00	122byte	システム動作設定	
0x1FE00	200byte	.monitor2	
0x20000		使用不可	
0xF0000	2048byte	特殊機能レジスタ(2 nd SFR)	
0xF0800		使用不可	
0xF1000	8192byte	データフラッシュメモリ	
0xF3000	<u>40,704byte</u>	Mirror 領域	
0xFCF00	<u>12,064byte</u>	RAM 領域	
		プログラムリソース領域(セクション配置は順不同)	
		アプリケーション	.bss, .dataR
		ビーコンスタック	BCN_BSS
		未使用領域	-
		スタックメモリ	-
0xFFE00	32byte	汎用レジスタ	
0xFFFF0	256byte	特殊機能レジスタ(SFR)	
0xFFFFF			

図 5-6 アドレスマップ(R5F11AGG)

サンプルプログラム(ビーコンアプリケーション-ASCII コマンド制御は除く)の RL78/G1D(R5F11AGH)でのアドレスマップを図 5-7 に示します。

なお下線部は R5F11AGG、R5F11AGJ の値と異なることを示します。

アドレス	領域サイズ	セクション	セクション名
	<u>196,608byte</u>	コードフラッシュメモリ	
0x00000	128byte	ベクタテーブル領域	.vect
0x00080	64byte	CALLT テーブル領域	.callt0
0x000C0	4byte	オプション・バイト領域	.option_byte
0x000C4	10byte	セキュリティ ID 設定領域	.security_id
0x000CE	<u>195,378byte</u>	プログラム領域 (セクション配置は順不同)	
		OCD モニタ	.monitor1
		スタートアップ	BOOT0_TEXT
		ランタイムライブラリ	.RLIB
		標準ライブラリ	.SLIB
		ビーコンスタック	BCN_CONST, BCN_TEXT
		アプリケーション	.const, .constf, .data, .text, .textf
		未使用	-
<u>0x2FC00</u>	122byte	システム動作設定	
<u>0x2FE00</u>	200byte	.monitor2	
<u>0x30000</u>		使用不可	
0xF0000	2048byte	特殊機能レジスタ(2 nd SFR)	
0xF0800		使用不可	
0xF1000	8192byte	データフラッシュメモリ	
0xF3000	<u>36,608byte</u>	Mirror 領域	
<u>0xFBF00</u>	<u>16,160byte</u>	RAM 領域	
		プログラムリソース領域(セクション配置は順不同)	
		アプリケーション	.bss, .dataR
		ビーコンスタック	BCN_BSS
		未使用領域	-
		スタックメモリ	-
0xFFE00	32byte	汎用レジスタ	
0xFFFF0	256byte	特殊機能レジスタ(SFR)	
0xFFFFF			

図 5-7 アドレスマップ(R5F11AGH)

サンプルプログラム(ビーコンアプリケーション-ASCII コマンド制御は除く)の RL78/G1D(R5F11AGJ)でのアドレスマップを図 5-8 に示します。

なお下線部は R5F11AGG、R5F11AGH の値と異なることを示します。

アドレス	領域サイズ	セクション	セクション名
	<u>262,144byte</u>	コードフラッシュメモリ	
0x00000	128byte	ベクタテーブル領域	.vect
0x00080	64byte	CALLT テーブル領域	.callt0
0x000C0	4byte	オプション・バイト領域	.option_byte
0x000C4	10byte	セキュリティ ID 設定領域	.security_id
0x000CE	<u>258,866byte</u>	プログラム領域 (セクション配置は順不同)	
		OCD モニタ	.monitor1, .monitor2
		スタートアップ	BOOT0_TEXT
		ランタイムライブラリ	.RLIB
		標準ライブラリ	.SLIB
		ビーコンスタック	BCN_CONST, BCN_TEXT
		アプリケーション	.const, .constf, .data, .text, .textf
		未使用	-
<u>0x3F400</u>	122byte	システム動作設定	
<u>0x3F47A</u>		未使用	
<u>0x3F800</u>	<u>512byte</u>	予約領域 (RL78/G1D モジュールのみ)	
<u>0x3FC00</u>	<u>6byte</u>	顧客固有情報	
<u>0x3FC06</u>		未使用	
<u>0x40000</u>		使用不可	
0xF0000	2048byte	特殊機能レジスタ(2 nd SFR)	
0xF0800		使用不可	
0xF1000	8192byte	データフラッシュメモリ	
0xF3000	<u>32,512byte</u>	Mirror 領域	
0xFAF00	1024byte	セルフ RAM 領域(R5F11AGJ のみ)	
<u>0xFB300</u>	<u>20,447byte</u>	RAM 領域	
		プログラムリソース領域(セクション配置は順不同)	
		アプリケーション	.bss, .dataR
		ビーコンスタック	BCN_BSS
		未使用領域	-
		スタックメモリ	-
0xFFEE0	32byte	汎用レジスタ	
0xFFFF0	256byte	特殊機能レジスタ(SFR)	
0xFFFFF			

図 5-8 アドレスマップ(R5F11AGJ)

6. 設定

サンプルプログラムのハードウェア設定とアプリケーション設定について示します。

ビーコンスタックが使用するハードウェアリソースについては『RL78/G1D ビーコンスタック ユーザーズマニュアル』(R01UW0171)の2.1節「使用ハードウェアリソース」を、ビーコンスタック API の仕様については『RL78/G1D ビーコンスタック ユーザーズマニュアル』(R01UW0171)の4章「API仕様」を参照してください。

6.1 ハードウェア設定

ビーコンスタックの主要なハードウェア設定は、`r_config.h` でマクロとして定義されています。マクロ定義の詳細については以降の記載を参照してください。

`r_config.h`, line 34-75

```

34:  /*
35:  * CONFIGURATION (NEED TO CHANGE BELOW DEFINES AS NECESSARY)
36:  *
37:  */
38:  /* MCU Main System Clock (either clock frequency of 4MHz,8MHz,16MHz,32MHz) */
39:  /* Note: It is necessary to set Option Bytes Value at Device Setting of Linker Option */
40:  #define MCU_HOCO_CLK      (8)
41:
42:  /* RF Operation (0:enable both Tx and Rx, 1:enable Tx only) */
43:  #define RF_TX_ONLY      (0)
44:
45:  /* RF DC-DC Converter (0:disable DC-DC, 1:enable DC-DC) */
46:  #define RF_DCDC_EN      (1)
47:
48:  /* RF Slow Clock Source (0:RF On-Chip Oscillator, 1:MCU XT1 Oscillator) */
49:  #define RF_SLK_XT1      (0)
50:
51:  /* RF Slow Clock Calibration (0:not execute, 1:execute) */
52:  /* Note: Calibration is only for RF-On_Chip_Oscillator */
53:  #define RF_SLK_CAL      (1)
54:
55:  /* RF High-speed clock output from CLKOUT_RF (0:not output, 4:4MHz, 8:8MHz, 16:16MHz) */
56:  #define RF_CLKOUT      (0)
57:
58:  /* RF 32MHz Oscillation Stabilization Time (usec, at least 550usec) */
59:  /* Note: Stabilization Time needs to be optimized for 32MHz resonator */
60:  #define RF_32MHZ_WAIT  (1000)
61:
62:  /* System Configuration Address in CodeFlash memory */
63:  #if defined(_USE_R5F11AGG)
64:  /* System Configuration is located the last block */
65:  #define SYSCFG_ADDR      (0x1FC00)
66:  #elif defined(_USE_R5F11AGH)
67:  /* System Configuration is located the last block */
68:  #define SYSCFG_ADDR      (0x2FC00)
69:  #elif defined(_USE_R5F11AGJ)
70:  /* System Configuration is located the third last block */
71:  /* by taking into account the location of RL78/G1D module (RY7011) */
72:  #define SYSCFG_ADDR      (0x3F400)
73:  /* In the case of RL78/G1D Module (RY7011), Device Address is located the last block */
74:  #define MODCFG_ADDR      (0x3FC00)
75:  #endif

```

6.1.1 MCU メイン・システム・クロック周波数

MCU メイン・システム・クロックには高速オンチップ・オシレータが生成するクロックを使用し、選択可能な動作周波数は4、8、16、32MHzのいずれかです。サンプルプログラムでは、MCU メイン・システム・クロックの動作周波数をMCU_HOCO_CLK マクロとオプション・バイトで定義します。デフォルトの動作周波数は8(MHz)です。

MCU 動作周波数を変更するには、MCU_HOCO_CLK マクロ値を4(MHz)または8(MHz)または16(MHz)または32(MHz)に設定します。

r_config.h, line 38-40

```
38: /* MCU Main System Clock (either clock frequency of 4MHz,8MHz,16MHz,32MHz) */
39: /* Note: It is necessary to set Option Bytes Value at Device Setting of Linker Option */
40: #define MCU_HOCO_CLK (8)
```

オプション・バイトはリンク・オプション"-user_opt_byte"で設定します。オプション・バイト値については表 6-1 を参照してください。

表 6-1 オプション・バイト設定

オプション・バイト設定			クロック周波数	動作モード
000C0	000C1	000C2		
(任意)	(任意)	2B	4MHz	低電圧メインモード
		AA	8MHz	低速メインモード
		E9	16MHz	高速メインモード
		E8	32MHz	

オプション・バイトの詳細については『RL78/G1D ユーザーズマニュアル ハードウェア編』(R01UH0515)の25章「オプション・バイト」を参照してください。

MCU 動作周波数によって動作電圧範囲が異なります。動作電圧については『RL78/G1D ユーザーズマニュアル ハードウェア編』(R01UH0515)の30.2節「動作電圧」を参照してください。

(1) CS+ for CC 使用時

CS+ for CC では、下記の手順でオプション・バイト値を変更します。

1. R5F11AGJ_Beacon サブプロジェクトの[CC-RL (ビルド・ツール)]を右クリック
2. 右クリックメニューで[プロパティ]を選択
3. [リンク・オプション]タブの[デバイス]→[ユーザ・オプション・バイト値]でオプション・バイト値を変更

(2) e² studio 使用時

e² studio では、下記の手順でオプション・バイト値を変更します。

1. R5F11AGJ_Beacon プロジェクトを右クリック
2. 右クリックメニューで[Renesas Tool Settings]を選択
3. [ツール設定]タブの[Linker]→[デバイス]→[ユーザ・オプション・バイト値]でオプション・バイト値を変更

6.1.2 RF 送受信動作

RF 動作として送受信ともに有効化、または送信のみ有効化を選択可能です。送信のみ有効化を選択することで RF 初期化時間が短縮されます。サンプルプログラムでは、RF 送受信動作設定を RF_TX_ONLY マクロで設定します。デフォルト設定は 0(送受信ともに有効化する)です。

送信のみ有効化するには、RF_TX_ONLY マクロ値を 1 に設定します。

r_config.h, line 42-43

```
42: /* RF Operation (0:enable both Tx and Rx, 1:enable Tx only) */
43: #define RF_TX_ONLY          (0)
```

6.1.3 RF 内蔵 DC-DC コンバータ

サンプルプログラムでは、RF 内蔵 DC-DC コンバータの使用設定を RF_DCDC_EN マクロで定義します。RF 内蔵 DC-DC コンバータを使用する、または使用しないを選択可能です。デフォルト設定は 1(RF 内蔵 DC-DC コンバータを使用する)です。

RF 内蔵 DC-DC コンバータは使用しないを選択するには、RF_DCDC_EN マクロ値を 0 に設定します。

r_config.h, line 45-46

```
45: /* RF DC-DC Converter (0:disable DC-DC, 1:enable DC-DC) */
46: #define RF_DCDC_EN          (1)
```

6.1.4 RF スロー・クロック供給源

RF スロー・クロックは RF 部の時間計測に使用され、RF スロー・クロックの供給源は RF 内蔵オシレータまたは MXU 部の XT1 オシレータのいずれかを選択可能です。サンプルプログラムでは、RF スロー・クロック供給源の選択を RF_SLK_XT1 マクロで定義します。デフォルト設定は 0(RF 内蔵オシレータ)です。

RF スロー・クロック供給源を MCU 部 XT1 オシレータに設定するには、RF_SLK_XT1 マクロ値を 1 に設定します。マクロ値を 1 に設定することで、MCU 部 XT1 オシレータで生成したクロックを EXSLK_RF 端子経由で RF 部に供給します。

r_config.h, line 48-49

```
48: /* RF Slow Clock Source (0:RF On-Chip Oscillator, 1:MCU XT1 Oscillator) */
49: #define RF_SLK_XT1          (0)
```

6.1.5 RF 内蔵オシレータのキャリブレーション

RF 内蔵オシレータの使用時、RF 内蔵オシレータが生成するクロック精度のキャリブレーションを実行する、または実行しないを選択可能です。キャリブレーションの実行により、Advertising インターバルの精度が向上します。なお、ビーコンスタックは RF 初期化後の初回 DEEP_SLEEP 遷移直前にキャリブレーションを実行します。サンプルプログラムでは、キャリブレーション設定を RF_SLK_CAL マクロで定義します。デフォルト設定は 1(キャリブレーションを実行する)です。

キャリブレーションは実行しないを選択するには、RF_SLK_CAL 値を 0 に設定します。

r_config.h, line 51-53

```
51: /* RF Slow Clock Calibration (0:not execute, 1:execute) */
52: /* Note: Calibration is only for RF-On_Chip_Oscillator */
53: #define RF_SLK_CAL          (1)
```

6.1.6 RF 基準クロックの分周出力

RF 基準クロックを分周した 4、8、16MHz のいずれかのクロックを CLKOUT_RF 端子から出力可能です。サンプルプログラムでは、RF 基準クロックの分周出力設定を RF_CLKOUT マクロで定義します。デフォルト設定は 0(出力しない)です。なお、RF 基準クロックの分周出力クロックを MCU 部の外部メイン・システム・クロックとして使用することはできません。

RF 基準クロックの分周出力を設定するには、RF_CLKOUT マクロ値を 0 または 4(MHz)または 8(MHz)または 16(MHz)に設定します。

r_config.h, line 55-56

```
55: /* RF High-speed clock output from CLKOUT_RF (0:not output, 4:4MHz, 8:8MHz, 16:16MHz) */
56: #define RF_CLKOUT          (0)
```

6.1.7 RF 基準クロックの発振安定時間

サンプルプログラムでは、発振安定時間を RF_32MHZ_WAIT マクロで定義します。RF 基準クロックとして使用する XTAL_RF 回路の発振安定時間は、XTAL1_RF 端子と XTAL2_RF 端子に接続した 32MHz 共振子に依存し、最適な値を設定する必要があります。デフォルト設定は RL78/G1D 評価ボードに適した 1000(usec)です。

発振安定時間を設定するには、RF_32MHZ_WAIT マクロ値を 550(usec)以上の値に設定します。

r_config.h, line 58-60

```
58: /* RF 32MHz Oscillation Stabilization Time (usec, at least 550usec) */
59: /* Note: Stabilization Time needs to be optimized for 32MHz resonator */
60: #define RF_32MHZ_WAIT      (1000)
```

6.1.8 システム動作設定アドレス

コードフラッシュメモリのファームウェア外部に、システム動作設定の配置が可能です。デバイスごとに異なるシステム動作設定を配置することで、デバイスアドレス、Advertising データなどの情報を、ファームウェアをリビルドすることなく設定することが可能です。サンプルプログラムでは、システム動作設定の配置アドレスを SYSCFG_ADDR マクロで定義します。

アドレスマップでの配置アドレスを変更するには、SYSCFG_ADDR マクロ値に新しいアドレスを設定します。

r_config.h, line 62-75

```
62: /* System Configuration Address in CodeFlash memory */
63: #if defined(_USE_R5F11AGG)
64: /* System Configuration is located the last block */
65: #define SYSCFG_ADDR      (0x1FC00)
66: #elif defined(_USE_R5F11AGH)
67: /* System Configuration is located the last block */
68: #define SYSCFG_ADDR      (0x2FC00)
69: #elif defined(_USE_R5F11AGJ)
70: /* System Configuration is located the third last block */
71: /* by taking into account the location of RL78/G1D module (RY7011) */
72: #define SYSCFG_ADDR      (0x3F400)
73: /* In the case of RL78/G1D Module (RY7011), Device Address is located the last block */
74: #define MODCFG_ADDR      (0x3FC00)
75: #endif
```

システム動作設定の詳細については 5.5 項「フラッシュメモリへのアクセス」を参照してください。

6.1.9 エナジーハーベスティング向けハードウェア設定

エナジーハーベスティングなどで発電される限られた電力を利用して、ビーコンアプリケーションで Advertising パケットを送信するために、消費電力を削減するための設定例を示します。

RF 初期化時間を短縮するために、RF 送受信動作設定に送信のみ有効化を設定します。設定の詳細は 6.1.2 項「RF 送受信動作」を参照してください。

r_config.h, line 42-43

```
42:  /* RF Operation (0:enable both Tx and Rx, 1:enable Tx only) */
43:  #define RF_TX_ONLY          (1)
```

RF 送信電流を削減するために、RF 内蔵 DC-DC コンバータは使用するを設定します。設定の詳細は 6.1.3 項「RF 内蔵 DC-DC コンバータ」を参照してください。

r_config.h, line 45-46

```
45:  /* RF DC-DC Converter (0:disable DC-DC, 1:enable DC-DC) */
46:  #define RF_DCDC_EN          (1)
```

MCU 部 XT1 オシレータの発振安定待ち時間を省略するために、RF スロー・クロック供給源に RF 内蔵オシレータを選択します。設定の詳細は 6.1.4 項「RF スロー・クロック供給源」を参照してください。

r_config.h, line 48-49

```
48:  /* RF Slow Clock Source (0:RF On-Chip Oscillator, 1:MCU XT1 Oscillator) */
49:  #define RF_SLK_XT1          (0)
```

RF 内蔵オシレータのキャリブレーション処理を省略するために、キャリブレーションは実行しないを選択します。設定の詳細は 6.1.5 項「RF 内蔵オシレータのキャリブレーション」を参照してください。

r_config.h, line 51-53

```
51:  /* RF Slow Clock Calibration (0:not execute, 1:execute) */
52:  /* Note: Calibration is only for RF-On_Chip_Oscillator */
53:  #define RF_SLK_CAL          (0)
```

6.2 アプリケーション設定

6.2.1 アプリケーション選択設定

実行するアプリケーションは、ファームウェアの起動前に評価ボードの DIP スイッチを切り替えて選択するか、またはファームウェアのビルド前に選択することが可能です。サンプルプログラムでは、アプリケーション選択設定を APP_SELECT マクロで定義します。デフォルト設定は 0(評価ボードの DIP スイッチで選択)です。

評価ボードの DIP スイッチを使用せず、ビーコンアプリケーションを選択してビルドする場合は APP_SELECT 値を 1 に、ASCII 形式スキャンアプリケーションを選択してビルドする場合は APP_SELECT 値を 2 に、バイナリ形式スキャンアプリケーションを選択してビルドする場合は APP_SELECT 値を 3 に、DTM アプリケーションを選択してビルドする場合は APP_SELECT 値を 4 に変更します。

ビーコンアプリケーション-ASCII コマンド制御を選択する場合は、APP_SELECT 値を 5 に設定してビルドしてください。評価ボードの DIP スイッチでは選択できません。

r_main.c, line 62-69

```

62:  /* Application Selection                               */
63:  /* 0: select by DIP SW6-1 and SW6-4 before start up firmware */
64:  /* 1: enable Beacon Application only                   */
65:  /* 2: enable UART-ASCII Scan Application only         */
66:  /* 3: enable UART-Binary Scan Application only        */
67:  /* 4: enable DTM Application only                     */
68:  /* 5: enable UART-ASCII Beacon Application only       */
69:  #define APP_SELECT          (0)

```

リリースパッケージは下記の HEX 形式ファームウェアファイルをサンプルとして含みます。

- APP_SELECT=0 でビルドしたファームウェアファイル：R5F11AGJ_Beacon.hex
- APP_SELECT=1 でビルドしたファームウェアファイル：R5F11AGJ_Beacon(beacon).hex
- APP_SELECT=2 でビルドしたファームウェアファイル：R5F11AGJ_Beacon(scan_ascii).hex
- APP_SELECT=3 でビルドしたファームウェアファイル：R5F11AGJ_Beacon(scan_bin).hex
- APP_SELECT=4 でビルドしたファームウェアファイル：R5F11AGJ_Beacon(dtm).hex
- APP_SELECT=5 でビルドしたファームウェアファイル：R5F11AGJ_Beacon(beacon_ascii).hex

6.2.2 システム動作設定

システム動作設定は、コードフラッシュメモリのサンプルプログラムファームウェア外部に配置されるデータです。本データは Renesas Flash Programmer のユニークコード埋め込み機能を使用することで、サンプルプログラムファームウェアと同時にシステム動作設定を書き込むことが可能です。

リリースパッケージはシステム動作設定のユニークコードファイルをサンプルとして含みます。システム動作設定の詳細については 5.5.1 項「コードフラッシュメモリへのアクセス」を参照してください。

r5f11agj_syscfg.ruc, line 1-10

```

1: // -----
2: // -- System Configuration for RL78/G1D Beacon Stack Sample Program --
3: // -- Device Part Number : R5F11AGJ --
4: // -----
5: format hex
6: area user flash
7: address 0x3f400
8: size 122
9: index data
10: (a) (b) (c) (d) (e)
    000001 B39A7856341200FFA00001070009FFFFFFFFFFFFFFFFFFFFFFFFB0201060303AAFE1316AAFE10EE02676F6F2E67
        (e) (f) (g)
    6C2F3764694C547800000001B0201060303AAFE1316AAFE10EE02676F6F2E676C2F3764694C547800000001E1E0952656E
        (g)
    6573617320524C37382F47314420426561636F6E204461746100

```

R5F11AGJ ユニークコードファイルには下記の内容が記述されています。

- line 1-4 : //から始まる行はコメント行
- line 5 : 16 進数フォーマットを指定
- line 6 : 書き込みエリアにユーザエリアを指定
- line 7 : 書き込みアドレスを 0x3F400 (block 253)に指定
- line 8 : 書き込みサイズを 122 byte に指定
- line 9 : 次行からのユニークコードデータ開始を宣言
- line10 : インデックスとユニークコードを指定
 - (a): ユニークコードデータのインデックス
 - (b): デバイスアドレス (6byte)
 - (c): デバイスアドレスタイプ(1byte), パディング(1byte)
 - (d): Advertising 情報 (18byte)
 - (e): Non-connectable Undirected Advertising データ (32byte)
 - (f): Scannable Undirected Advertising データ (32byte)
 - (g): Scan Response データ (32byte)

6.2.3 システム動作設定 (ビーコンアプリケーション-ASCII コマンド制御)

ビーコンアプリケーション-ASCII コマンド制御のシステム動作設定は、コードフラッシュメモリのサンプルプログラムファームウェア外部に配置されるデータです。本データは Renesas Flash Programmer のユニークコード埋め込み機能を使用することで、サンプルプログラムファームウェアと同時にシステム動作設定を書き込むことが可能です。

リリースパッケージはシステム動作設定のユニークコードファイルをサンプルとして含みます。システム動作設定の詳細については 5.5.2 項「コードフラッシュメモリへのアクセス (ビーコンアプリケーション-ASCII コマンド制御)」を参照してください。

r5f11agj_syscfg(beacon_ascii).ruc, line 1-9

```

1: // -----
2: // -- System Configuration for RL78/G1D Beacon Stack Sample Program --
3: // -- Device Part Number : R5F11AGJ --
4: // -----
5: format hex
6: address 0x3f400
7: size 794
8: index data
9: 000001 B19A7856341200FFA00001070009B19A78563412000172BE07001B0201060303AAFE1316AAFE10EE02676F6E67
   (a)   (b)   (c)   (d)   (e)
6C2F35774B6B524B00000000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
   (f)   (g)
2E676C2F4A51683366510000000019180952656E6573617320524C37382F47314420426561636F6E000000000000000000
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
   (h)
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFF

```

R5F11AGJ ユニークコードファイルには下記の内容が記述されています。

- line 1-4 : //から始まる行はコメント行
- line 5 : 16進数フォーマットを指定
- line 6 : 書き込みアドレスを 0x3F400 (block 253)に指定
- line 7 : 書き込みサイズを 794byte に指定
- line 8 : 次行からのユニークコードデータ開始を宣言
- line 9 : インデックスとユニークコードを指定
 - (a): ユニークコードデータのインデックス
 - (b): デバイスアドレス (6byte)
 - (c): デバイスアドレスタイプ(1byte), パディング(1byte)
 - (d): Advertising 情報 (18byte)
 - (e): Non-connectable Undirected Advertising データ (32byte)
 - (f): Scannable Undirected Advertising データ (32byte)
 - (g): Scan Response データ (32byte)
 - (h): White List データ (8byte)

6.2.4 Advertising 動作設定

ビーコンアプリケーションとビーコンアプリケーション-ASCII コマンド制御のデフォルト Advertising 動作設定は `r_beacon.c` ファイル、`r_beacon_ascii.c` ファイルで定義します。設定可能なマクロや構造体の仕様については『RL78/G1D ビーコンスタック ユーザーズマニュアル』(R01UW0171)の4章「API」を参照してください。

Advertising タイプについては変数 `adv_type` に設定し、Advertising インターバルや Advertising チャネルその他の Advertising 情報は `adv_info` 構造体に設定します。ビーコンアプリケーションの Advertising 動作設定を変更するには、変数 `adv_type` や `adv_info` 構造体の設定を変更します。

r_beacon.c, line 99-120

```

99:  /* Advertising packet type, ADV_NONCONN_IND or ADV_SCAN_IND */
100:  static uint8_t adv_type = RBLE_PDU_ADV_NONCONN_IND;
101:
102:  /* Advertising Information */
103:  static RBLE_ADV_INFO adv_info =
104:  {
105:      0x00A0,                /* Advertising interval */
106:      true,                 /* Advertising interval delay */
107:      RBLE_ADV_ALL_CHANNELS, /* Advertising channel map */
108:      0x00,                /* Advertising count limitation */
109:      RBLE_TXPW_LV9,       /* Advertising transfer power */
110:      { 0xB1, 0x9A, 0x78, 0x56, 0x34, 0x12 }, /* Own device address */
111:      RBLE_ADDR_PUBLIC,    /* Own device address type */
112:      sizeof(adv_nonconn_data) / sizeof(RBLE_ADV_DATA), /* Advertising data count */
113:      &adv_nonconn_data[0], /* Advertising data */
114:      RBLE_EVT_PERMIT_ADV_ALL, /* Advertising event permission */
115:      #if WLIST_EN
116:      true                 /* Use White List */
117:      #else
118:      false                /* Use White List */
119:      #endif
120:  };

```

※ビーコンアプリケーション-ASCII コマンド制御の場合、`r_beacon_ascii.c` ファイルの191行目~208行目を参照してください。

6.2.5 Advertising データ

ビーコンアプリケーションとビーコンアプリケーション-ASCII コマンド制御の Non-connectable Undirected Advertising パケットのデフォルトデータは、`adv_nonconn_data` 構造体に設定します。

r_beacon.c, line 54-70

```

54:  /* Advertising Data Array for ADV_NONCONN_IND */
55:  static RBLE_ADV_DATA adv_nonconn_data[] =
56:  {
57:      /* Advertising Data[0] */
58:      /* Eddystone-URL: https://goo.gl/5wKkRK -> https://www.renesas.com/ */
59:      {
60:          /* Advertising data length */
61:          27,
62:          /* Advertising data <<Flags>> */
63:          0x02, 0x01, 0x06,
64:          /* Advertising data <<Complete List of 16-bit Service Class UUIDs>> */
65:          0x03, 0x03, 0xAA, 0xFE,
66:          /* Advertising data <<Service Data>> */
67:          0x13, 0x16, 0xAA, 0xFE, 0x10, 0xEE, 0x02,
68:          'g', 'o', 'o', '.', 'g', 'l', '/', '5', 'w', 'k', 'r', 'k',
69:      },
70:  };

```

※ビーコンアプリケーション-ASCII コマンド制御の場合、`r_beacon_ascii.c` ファイルの146行目~162行目を参照してください。

ビーコンアプリケーションとビーコンアプリケーション-ASCII コマンド制御の Scannable Undirected Advertising パケット、Scan Response パケットのデフォルトデータは、adv_scan_data 構造体に設定します。

r_beacon.c, line 72-97

```

72:  /* Advertising Data Array for ADV_SCAN_IND */
73:  static RBLE_ADV_DATA adv_scan_data[] =
74:  {
75:      /* Advertising Data[0] */
76:      /* Eddystone-URL: http://goo.gl/JQh3fQ ->
                          https://github.com/google/eddytone/tree/master/eddytone-url */
77:      {
78:          /* Advertising data length */
79:          27,
80:          /* Advertising data <<Flags>> */
81:          0x02, 0x01, 0x06,
82:          /* Advertising data <<Complete List of 16-bit Service Class UUIDs>> */
83:          0x03, 0x03, 0xAA, 0xFE,
84:          /* Advertising data <<Service Data>> */
85:          0x13, 0x16, 0xAA, 0xFE, 0x10, 0xEE, 0x02,
86:          'g', 'o', 'o', '.', 'g', 'l', '/', 'J', 'Q', 'h', '3', 'f', 'Q'
87:      },
88:      /* Scan Response Data[0] */
89:      {
90:          /* Scan Response data length */
91:          25,
92:          /* Scan Response data <<Complete local name>> */
93:          0x18, 0x09,
94:          'R', 'e', 'n', 'e', 's', 'a', 's', ' ', 'R', 'L', '7', '8', '/', 'G', '1', 'D',
95:          ' ', 'B', 'e', 'a', 'c', 'o', 'n'
96:      },
97:  };

```

※ビーコンアプリケーション-ASCII コマンド制御の場合、r_beacon_ascii.c ファイルの 164 行目~189 行目を参照してください。

複数の Advertising データを設定するには、RBLE_ADV_DATA 構造体配列を追加します。

複数 Advertising データ送信のコード例

```

/* Advertising Data Array */
static RBLE_ADV_DATA adv_data[] =
{
    /* Advertising data No.1 */
    {
        /* Advertising data length */
        ... ,
        /* Advertising data */
        ...
    },
    /* Advertising data No.2 */
    {
        /* Advertising data length */
        ... ,
        /* Advertising data */
        ...
    }
    :
};

```


6.2.6 Advertising データ更新設定

ビーコンアプリケーションは Advertising を停止することなく、送信データを更新可能です。送信データの更新動作を有効にするには、`r_beacon.c` で定義された `UPDATE_EN` マクロを 1 に設定します。

`r_beacon.c`, line 44-45

```
44:  /* Update Advertising Data (0:disable Update, 1:enable Update) */
45:  #define UPDATE_EN          (0)
```

※ビーコンアプリケーション-ASCII コマンド制御に本マクロはありません。Advertising データの更新は「Advertising データ設定コマンド」を使用してください。

`UPDATE_EN` マクロを 1 に設定すると、ビーコンアプリケーションは送信完了イベントで Advertising データを更新します。スキャンアプリケーションで Advertising パケットを受信することで、Advertising データが更新されていることを確認可能です。

Advertising データ更新時の受信結果

```
Start Scan :OK
ADV_NONCONN_IND PUBLIC 12:34:56:78:9A:B1 39ch -37dBm 30byte .!.PV.A.&.$J.....RL78/G1D 00
ADV_NONCONN_IND PUBLIC 12:34:56:78:9A:B1 39ch -37dBm 30byte .!.PV.A.&.$J.....RL78/G1D 01
ADV_NONCONN_IND PUBLIC 12:34:56:78:9A:B1 37ch -37dBm 30byte .!.PV.A.&.$J.....RL78/G1D 02
ADV_NONCONN_IND PUBLIC 12:34:56:78:9A:B1 37ch -36dBm 30byte .!.PV.A.&.$J.....RL78/G1D 03
ADV_NONCONN_IND PUBLIC 12:34:56:78:9A:B1 37ch -37dBm 30byte .!.PV.A.&.$J.....RL78/G1D 04
ADV_NONCONN_IND PUBLIC 12:34:56:78:9A:B1 37ch -37dBm 30byte .!.PV.A.&.$J.....RL78/G1D 05
```

6.2.7 White List 設定

ビーコンアプリケーションは Scannable Undirected Advertising パケット送信中に、White List を使用可能です。White List を使用することで、Scan Request パケットをデバイスアドレスでフィルタリングします。White List を使用するには、`r_beacon.c` で定義された `WLIST_EN` マクロを 1 に設定します。

`r_beacon.c`, line 47-48

```
47:  /* Enable White List for Filtering Scan Request packets */
48:  #define WLIST_EN          (0)
```

※ビーコンアプリケーション-ASCII コマンド制御に本マクロはありません。White List の使用可否は「White List 設定コマンド」を使用してください。

`WLIST_EN` マクロを 1 に設定すると、`wl_info` 構造体の定義が有効となりビーコンスタックに設定されます。デフォルトではスキャンアプリケーションのデバイスアドレスが White List に登録されます。これにより、スキャンアプリケーション以外のデバイスからの Scan Request パケットを受信しません。

`r_beacon.c`, line 133-139

```
133:  #if WLIST_EN
134:  /* White List for Filtering Scan Requist packets */
135:  static RBLE_DEV_INFO wl_info[] =
136:  {
137:      {RBLE_ADDR_PUBLIC, 0, {0xB2, 0x9A, 0x78, 0x56, 0x34, 0x12}},
138:  };
139:  #endif
```

※ビーコンアプリケーション-ASCII コマンド制御の場合、`r_beacon_ascii.c` ファイルの 210 行目~214 行目を参照してください。

7. 関数

サンプルプログラムに実装される主要な関数を示します。

7.1 関数一覧

7.1.1 ビーコンアプリケーション

ビーコンアプリケーションの関数を表 7-1 に示します。

表 7-1 ビーコンアプリケーション関数

ファイル	関数	説明
r_beacon.c	R_BEACON_Main	ビーコンアプリケーションの初期化とメインループの実行
	beacon_eventhandler	ビーコンスタックイベント・ハンドラ
	beacon_input_callback	外部入力割り込みコールバック Advertising の開始・停止

7.1.2 スキャンアプリケーション

スキャンアプリケーションの関数を表 7-2 に示します。

表 7-2 スキャンアプリケーション関数

ファイル	関数	説明
r_scan_ascii.c / r_scan_bin.c	R_SCAN_ASCII_Main / R_SCAN_BINARY_Main	スキャンアプリケーションの初期化とメインループの実行
	scan_eventhandler	ビーコンスタックイベント・ハンドラ
	uart_rx_complete	UART 受信完了割り込みコールバック UART コマンドの解析と Scanning の開始・停止
	uart_tx_complete	UART 送信完了割り込みコールバック UART イベントの送信完了確認
	uart_rx_error	UART 受信エラー割り込みコールバック UART コマンド受信の再起動
r_scan_ascii.c	scan_input_callback	外部入力割り込みコールバック Advertising 受信結果の表示形式変更

7.1.3 DTM アプリケーション

DTM アプリケーションの関数を表 7-3 に示します。

表 7-3 DTM アプリケーション関数

ファイル	関数	説明
r_dtm.c	R_DTM_Main	DTM アプリケーションの初期化とメインループの実行
	dtm_eventhandler	ビーコンスタックイベント・ハンドラ
	dtm_rx_complete	UART 受信完了割り込みコールバック RF テストコマンドの解析と DTM の開始・停止
	dtm_tx_complete	UART 送信完了割り込みコールバック RF テストイベントの送信完了確認
	dtm_rx_error	UART 受信エラー割り込みコールバック RF テストコマンド受信の再起動

7.1.4 ビーコンアプリケーション-ASCII コマンド制御

ビーコンアプリケーション-ASCII コマンド制御の関数を表 7-4 に示します。

表 7-4 ビーコンアプリケーション-ASCII コマンド制御関数

ファイル	関数	説明
r_beacon_ascii.c	R_BEACON_ASCII_Main	ビーコンアプリケーションの初期化とメインループの実行
	beacon_eventhandler	ビーコンスタックイベント・ハンドラ
	beacon_read_advcfg	システム動作設定の読み込み
	beacon_send_<***>*	コマンド処理結果の UART 出力
	uart_cmdfunc	入力コマンドの識別
	uart_cmd_<***>*	コマンドの処理

※<***>で表される各関数については r_beacon_ascii.c を参照してください。

8. 動作

8.1 状態遷移

サンプルプログラムは、3つのアプリケーション（ビーコンアプリケーション、スキャンアプリケーション、DTMアプリケーション、ビーコンアプリケーション-ASCIIコマンド制御）で構成されます。本節では各アプリケーションの状態遷移を示します。

8.1.1 ビーコンアプリケーション

ビーコンアプリケーションの状態遷移を図 8-1 に示します。

ビーコンアプリケーションは Initializing 状態で動作を開始し、Non-connectable Undirected Advertising パケットを送信する Advertising 状態に遷移します。

評価ボードのスイッチ SW2 を押下すると Advertising パケット送信を停止し、Idling 状態に遷移します（図中 1）。SW2 を押下すると、Scannable Undirected Advertising パケットを送信する Advertising 状態に遷移します（図中の 2）。SW2 を押下すると Advertising パケット送信を停止し、Idling 状態に遷移します（図中 3）。さらに SW を押下すると再度 Non-connectable Undirected Advertising パケットを送信する Advertising 状態に遷移します（図中の 4）。

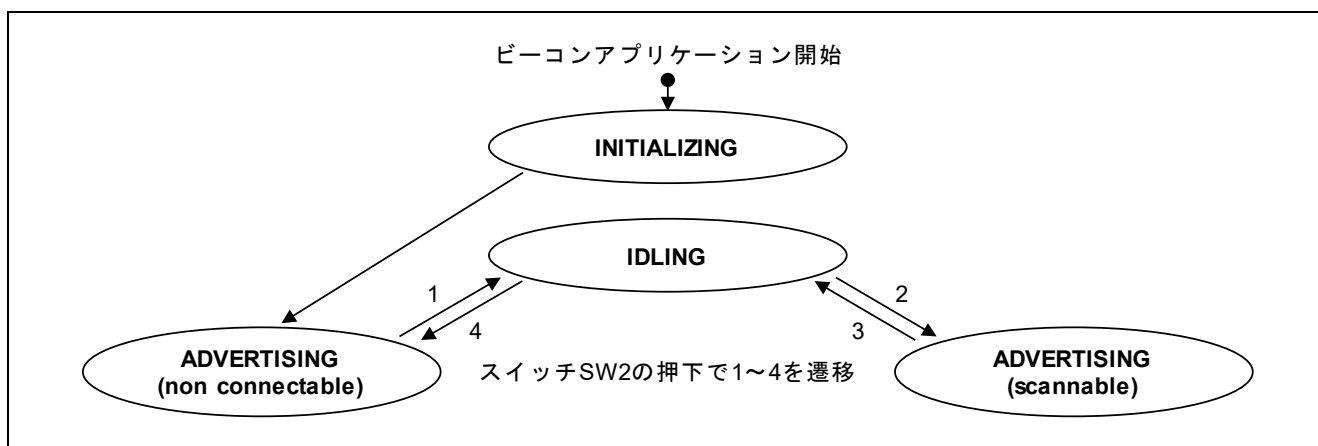


図 8-1 ビーコンアプリケーションの状態遷移

8.1.2 スキャンアプリケーション

スキャンアプリケーションの状態遷移を図 8-2 に示します。

スキャンアプリケーションは Initializing 状態で動作を開始し、ASCII 形式 UART 通信の使用時は Scanning 状態に遷移し、バイナリ形式 UART 通信の使用時は Idling 状態に遷移します。

Scanning 状態では Scanning を実行します。Stop Scan コマンドを UART 受信すると、Scanning を停止して Idling 状態に遷移します。

Idling 状態では設定変更コマンドを UART 受信することで、Scanning 時に使用する設定を更新します。Start Scan コマンドを UART 受信すると、Scanning を実行し再度 Scanning 状態に遷移します。

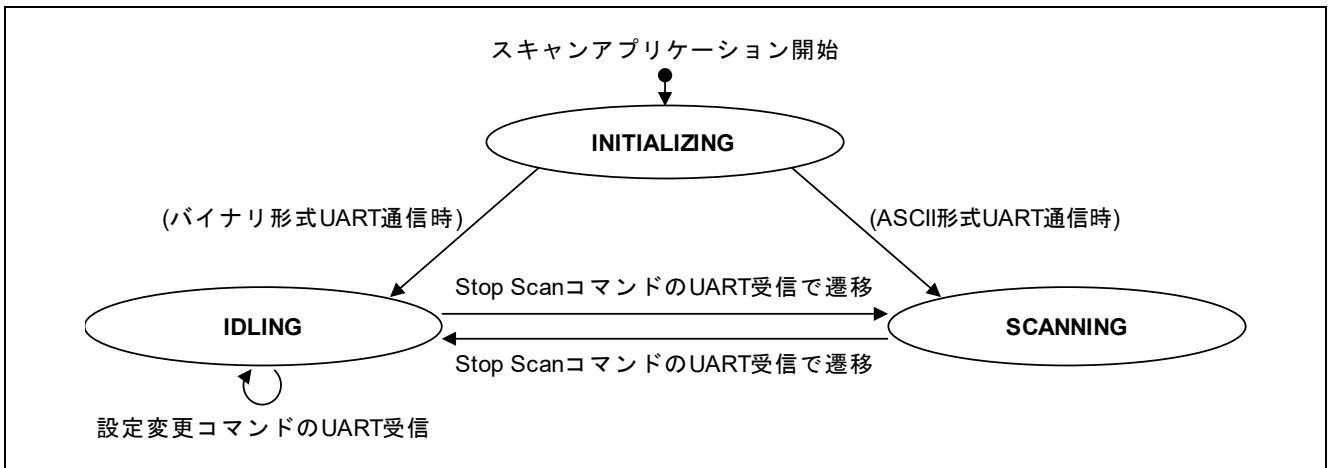


図 8-2 スキャンアプリケーションの状態遷移

8.1.3 DTM アプリケーション

DTM アプリケーションの状態遷移を図 8-3 に示します。

DTM アプリケーションは Initializing 状態で動作を開始し、その後 Idling 状態に遷移します。

Idling 状態では RF テストコマンドの LE_RESET、LE_TRANSMITTER_TEST、LE_RECEIVER_TEST を受け付けます。LE_TRANSMITTER_TEST コマンドを UART 受信すると、RF 送信テストを実行して Transmitter Test 状態に遷移します。LE_RECEIVER_TEST コマンドを UART 受信すると、RF 受信テストを実行して Receiver Test 状態に遷移します。LE_TEST_END コマンドを UART 受信すると、RF テストを停止して Idling 状態に戻ります。

なお Transmitter Test 状態または Receiver Test 状態で LE_RESET コマンドを UART 受信すると、RF テストを停止して、Idling 状態に戻ります。

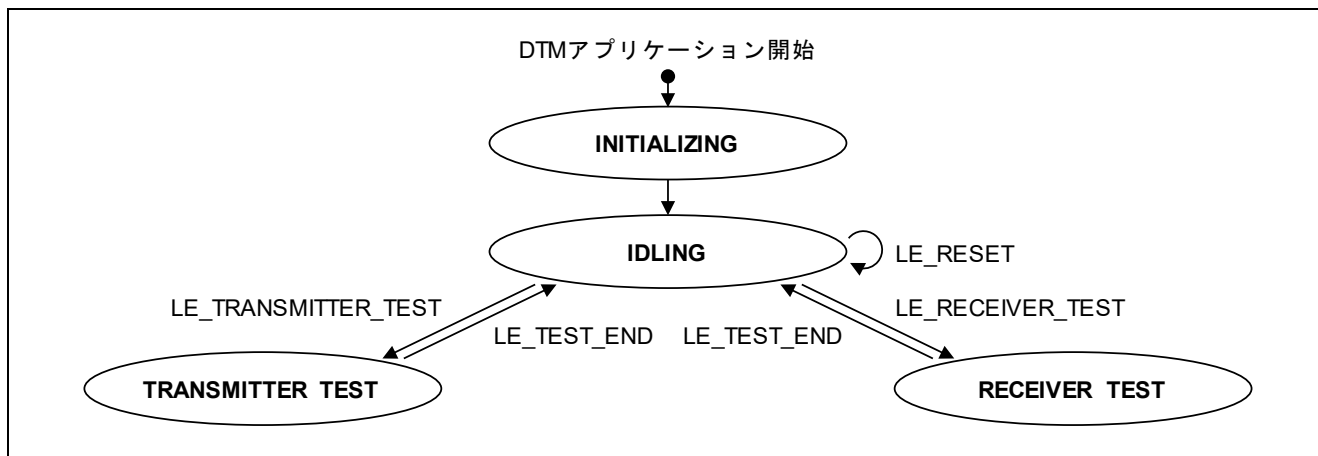


図 8-3 DTM アプリケーションの状態遷移

8.1.4 ビーコンアプリケーション-ASCII コマンド制御

ビーコンアプリケーション-ASCII コマンド制御の状態遷移を図 8-4 に示します。

ビーコンアプリケーションは Initializing 状態で動作を開始し、ASCII コマンドの入力を待つ Idling 状態に遷移します。

ホストマシンから "adstart n" コマンドを入力すると Non-connectable Undirected Advertising パケットを送信する Advertising 状態に遷移 (図中 4) し、"adstop" コマンドを入力すると Idling 状態に遷移 (図中 1) します。

"adstart s" コマンドを入力すると Scannable Undirected Advertising パケットを送信する Advertising 状態に遷移 (図中 2) し、"adstop" コマンドを入力すると Idling 状態に遷移 (図中 3) します。

"pwrn" コマンドを入力すると RF Power down 状態に遷移 (図中 5) し、"pwup" コマンドを入力すると Idling 状態に遷移 (図中 6) します。

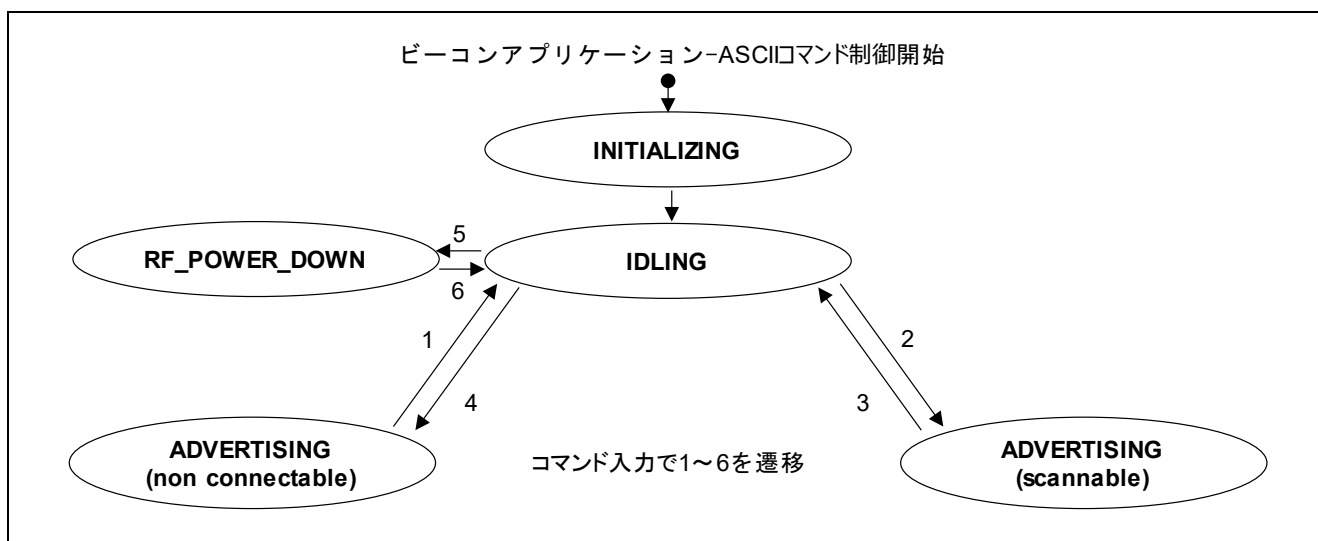


図 8-4 ビーコンアプリケーション-ASCII コマンド制御の状態遷移

8.2 シーケンス

サンプルプログラムの RF 初期化、ビーコンアプリケーション、スキャンアプリケーション、DTM アプリケーションのシーケンスを示します。サンプルプログラムはビーコンスタック API を使用します。

ビーコンスタック API の仕様については『RL78/G1D ビーコンスタック ユーザーズマニュアル』(R01UW0171)の 4 章「API」を参照してください。

8.2.1 RF 初期化

RF 初期化シーケンスを図 8-5 に示します。

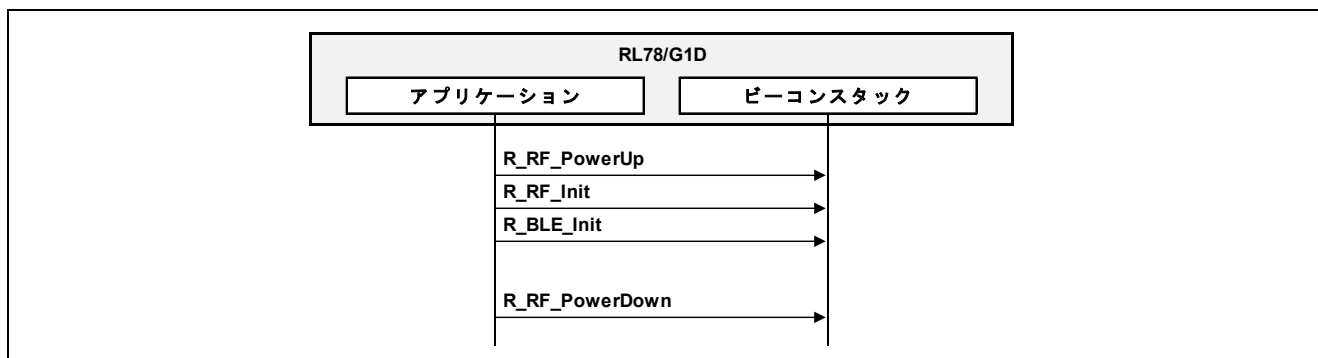


図 8-5 RF 初期化シーケンス

8.2.2 ビーコンアプリケーション

ビーコンアプリケーションのシーケンスを図 8-6 に示します。

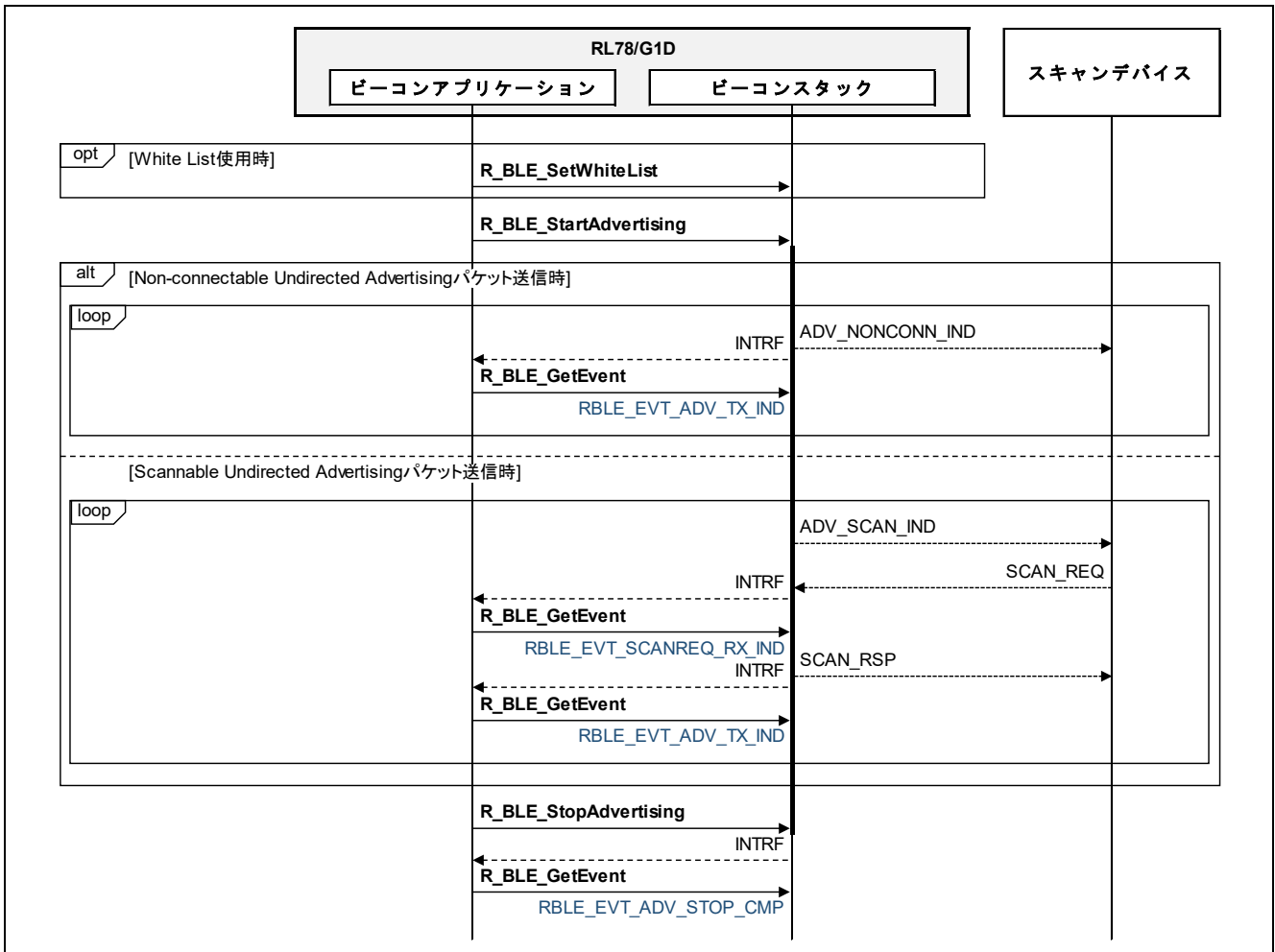


図 8-6 ビーコンアプリケーションシーケンス

8.2.3 スキャンアプリケーション

スキャンアプリケーションのシーケンスを図 8-7 に示します。

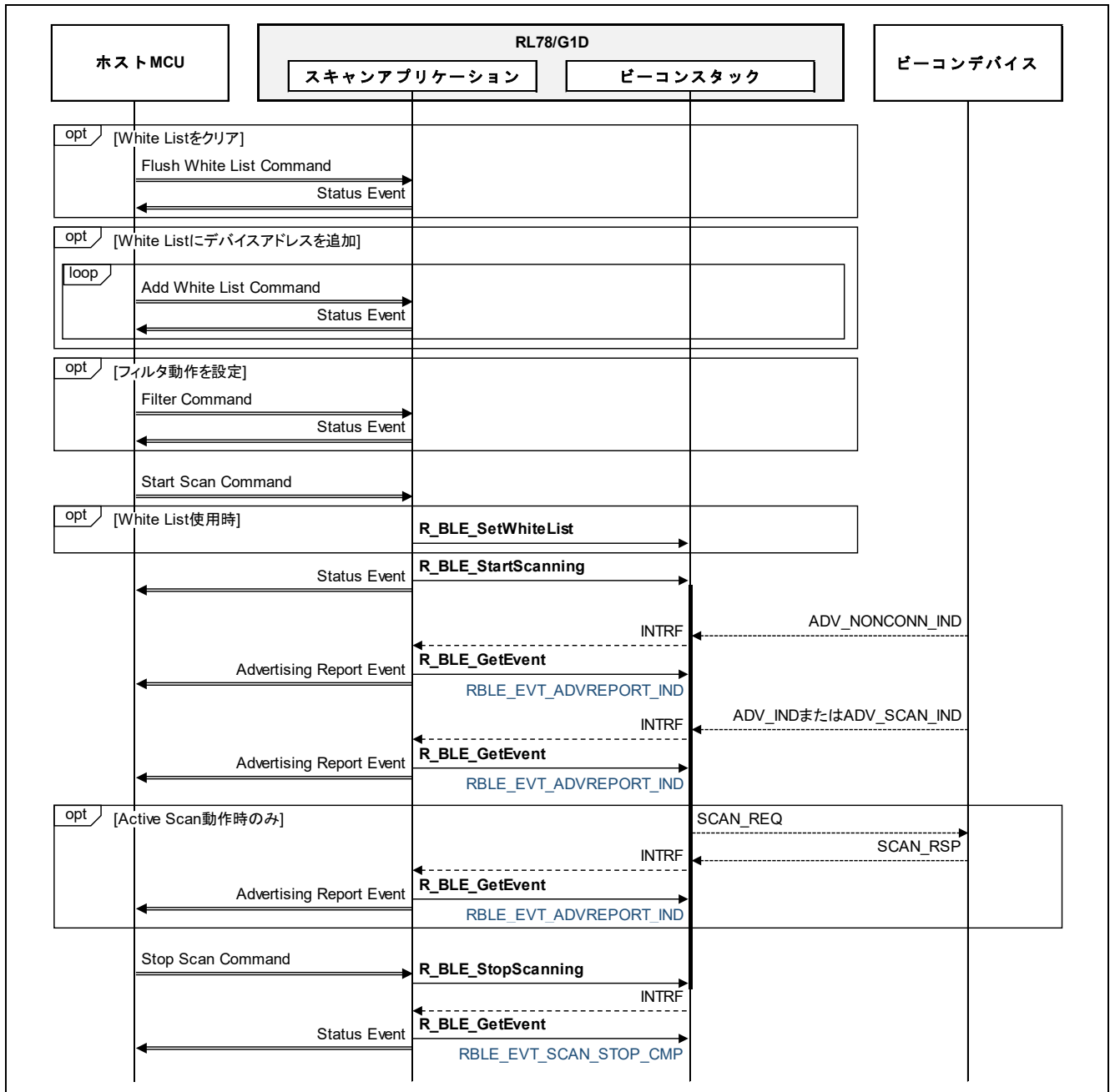


図 8-7 スキャンアプリケーションシーケンス

8.2.4 DTM アプリケーション

DTM アプリケーションのシーケンスを図 8-8 に示します。

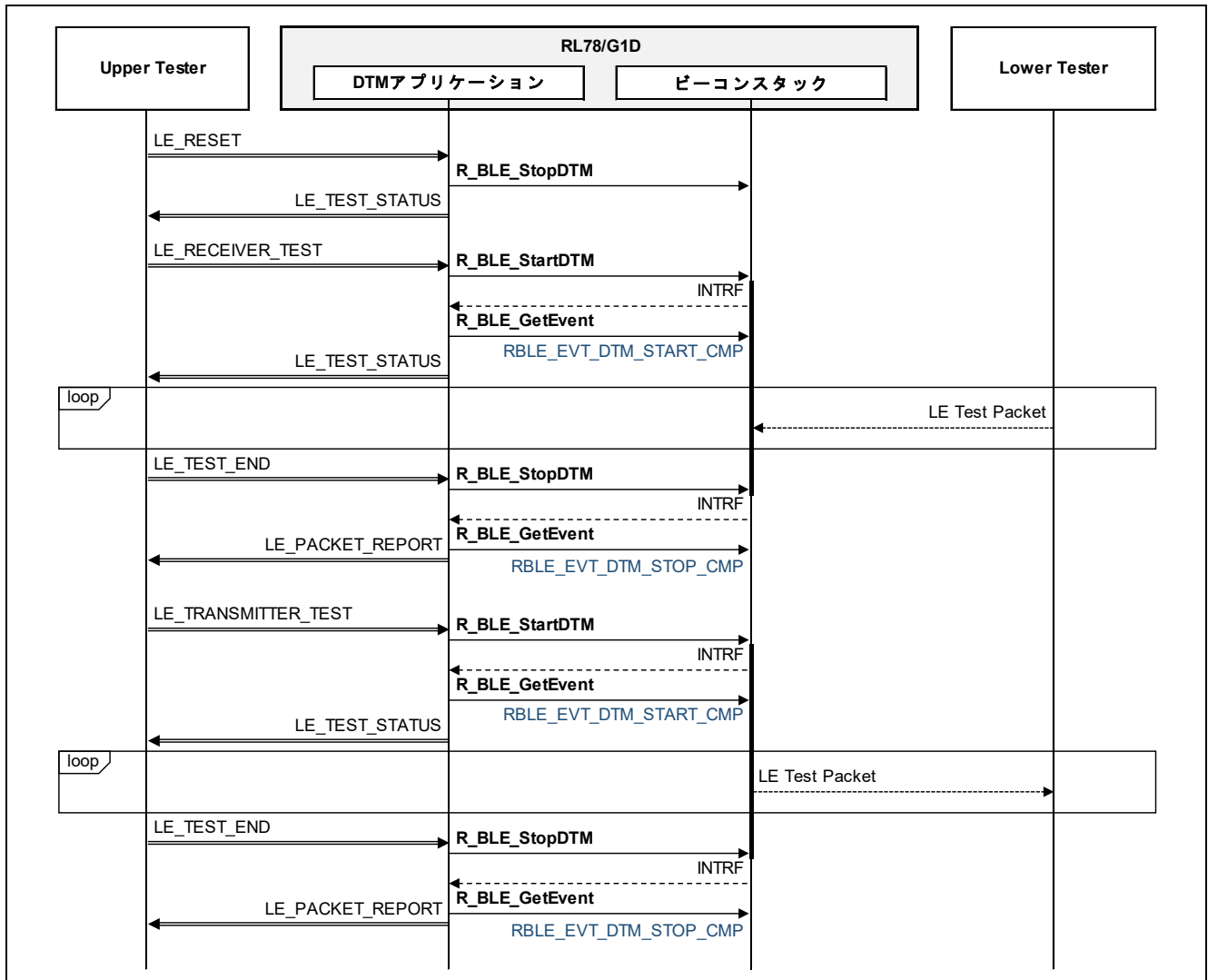


図 8-8 DTM アプリケーションシーケンス

8.2.5 ビーコンアプリケーション-ASCII コマンド制御

ビーコンアプリケーション-ASCII コマンド制御は主に下記 API を使用します。API の仕様について『RL78/G1D ビーコンスタック ユーザーズマニュアル』(R01UW0171)の 4 章「API」を参照してください。

- R_BLE_StartAdvertising
- R_BLE_StopAdvertising
- R_BLE_UpdateAdvInfo
- R_BLE_UpdateAdvData
- R_BLE_SetWhiteList
- R_BLE_GetEvent

9. Appendix

9.1 ビーコンスタックの消費電流

ビーコンスタックの起動時と周期的な Advertising パケット送信時の消費電流について、弊社環境における測定条件と測定結果を示します。

9.1.1 注意事項

記載した消費電流値は、弊社環境での測定結果を示す参考値であり、製品の保証値ではありません。製品ご利用をご検討の際は、お客様の想定する動作条件と環境において評価することを推奨致します。

9.1.2 測定環境

消費電流の測定環境一覧を表 9-1 に示します。RL78/G1D 評価ボードのスライドスイッチ設定については 4.5.2 項「評価ボード設定」、測定手順については 4.5.3 項「消費電流の測定」を参照してください。

表 9-1 測定環境一覧

分類	条件
ターゲットデバイス	RL78/G1D(R5F11AGJ)
ターゲットボード	RL78/G1D 評価ボード(RTK0EN0001D01001BZ)
電流測定機器	Keysight Technologies DC 電源アナライザ メインフレーム(N6705B) バッテリードレイン解析向け 2 象限ソース/メジャメント・ユニット(N6781A)

消費電流の測定環境を図 9-1 に示します。

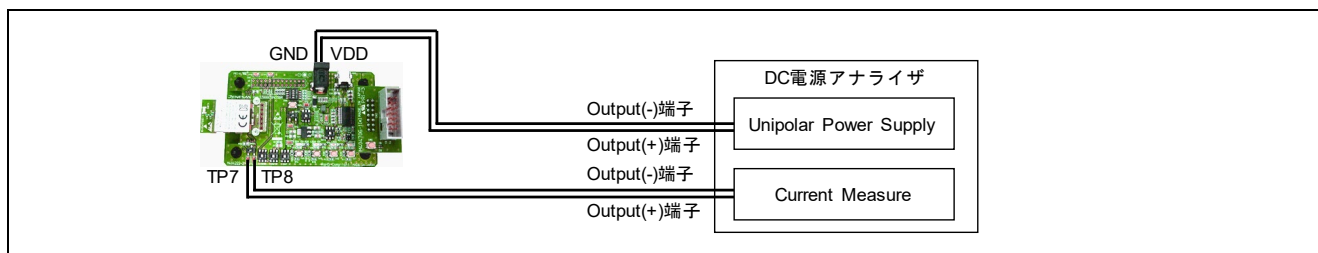


図 9-1 消費電流測定環境

9.1.3 測定結果：ビーコンスタック起動

(1) MCU 動作周波数 4MHz

MCU 動作周波数 4MHz におけるビーコンスタック起動電流測定時の動作設定を表 9-2 に示します。

表 9-2 ビーコンスタック起動電流測定時の動作設定

分類	測定条件
コンパイラ	CC-RL V1.04.00
MCU 動作周波数	4MHz
動作電圧	3V
DC-DC コンバータ	RF 内蔵 DC-DC コンバータ未使用
RF スロー・クロック供給源	RF 内蔵オシレータ使用、キャリブレーション実行
RF 送受信動作	RF 送信のみ有効化(RFCFG_TX)
送信パワー	0dBm
送信チャンネル	3ch 送信 (37,38,39ch)
Advertising タイプ	ADV_NONCONN_IND
Advertising データ長	31byte
Advertising データ数	1
Advertising インターバル	100msec

MCU 動作周波数 4MHz の場合のビーコンスタック起動時の消費電流波形を図 9-2 に示します。

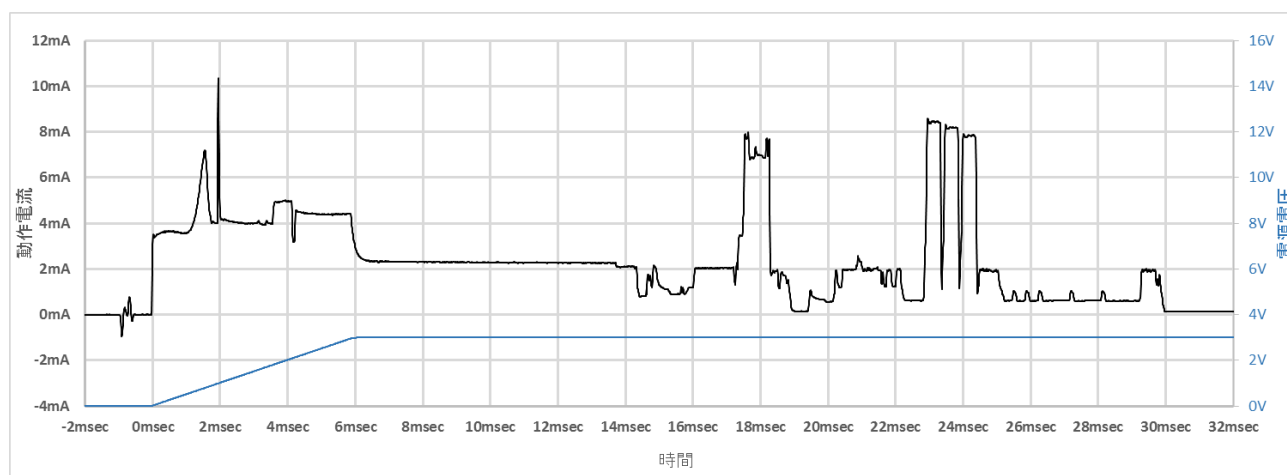


図 9-2 MCU 動作周波数 4MHz でのビーコンスタック起動電流波形

(2) MCU 動作周波数 8MHz

MCU 動作周波数 8MHz におけるビーコンスタック起動電流測定時の動作設定を表 9-3 に示します。

表 9-3 ビーコンスタック起動電流測定時の動作設定

分類	測定条件
コンパイラ	CC-RL V1.04.00
MCU 動作周波数	8MHz
動作電圧	3V
DC-DC コンバータ	RF 内蔵 DC-DC コンバータ未使用
RF スロー・クロック供給源	RF 内蔵オシレータ使用、キャリブレーション実行
RF 送受信動作	RF 送信のみ有効化(RFCFG_TX)
送信パワー	0dBm
送信チャンネル	3ch 送信 (37,38,39ch)
Advertising タイプ	ADV_NONCONN_IND
Advertising データ長	31byte
Advertising データ数	1
Advertising インターバル	100msec

MCU 動作周波数 8MHz の場合のビーコンスタック起動時の消費電流波形を図 9-3 に示します。

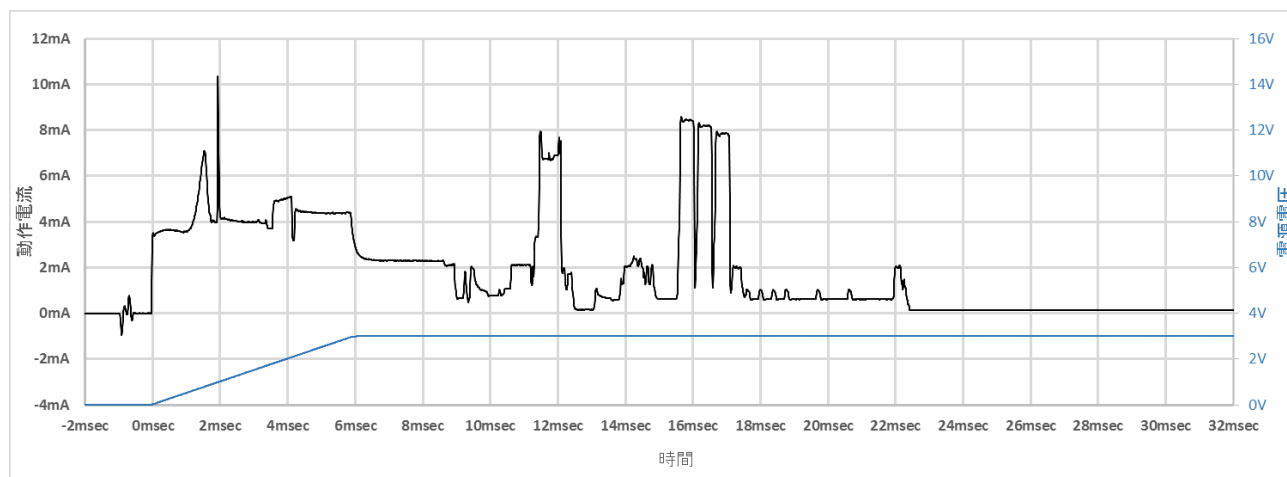


図 9-3 MCU 動作周波数 8MHz でのビーコンスタック起動電流波形

(3) MCU 動作周波数 16MHz

MCU 動作周波数 16MHz におけるビーコンスタック起動電流測定時の動作設定を表 9-4 に示します。

表 9-4 ビーコンスタック起動電流測定時の動作設定

分類	測定条件
コンパイラ	CC-RL V1.04.00
MCU 動作周波数	16MHz
動作電圧	3V
DC-DC コンバータ	RF 内蔵 DC-DC コンバータ未使用
RF スロー・クロック供給源	RF 内蔵オシレータ使用、キャリブレーション実行
RF 送受信動作	RF 送信のみ有効化(RFCFG_TX)
送信パワー	0dBm
送信チャンネル	3ch 送信 (37,38,39ch)
Advertising タイプ	ADV_NONCONN_IND
Advertising データ長	31byte
Advertising データ数	1
Advertising インターバル	100msec

MCU 動作周波数 16MHz の場合のビーコンスタック起動時の消費電流波形を図 9-4 に示します。

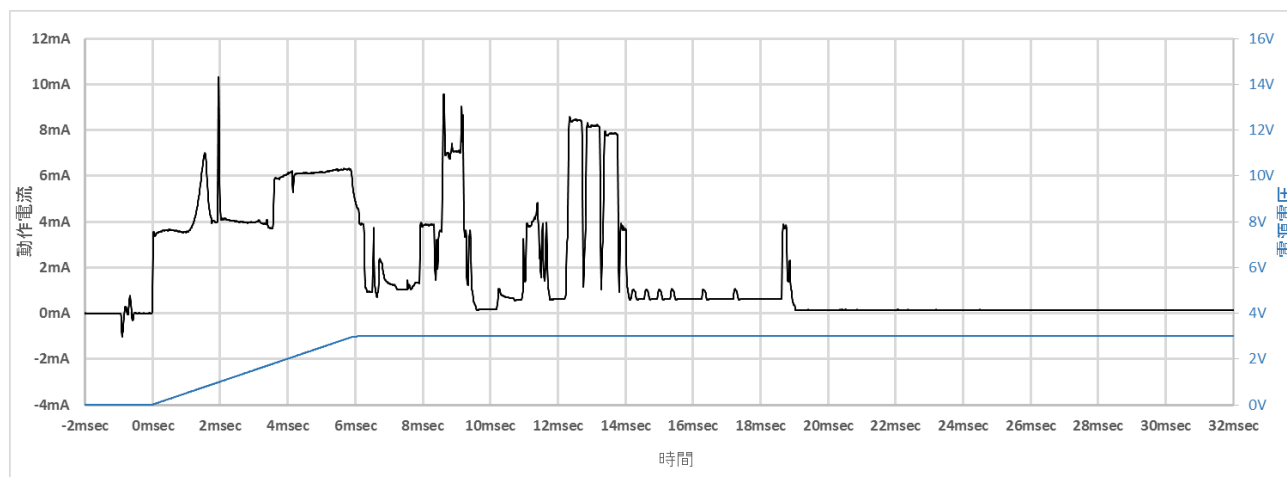


図 9-4 MCU 動作周波数 16MHz でのビーコンスタック起動電流波形

(4) MCU 動作周波数 32MHz

MCU 動作周波数 32MHz におけるビーコンスタック起動電流測定時の動作設定を表 9-5 に示します。

表 9-5 ビーコンスタック起動電流測定時の動作設定

分類	測定条件
コンパイラ	CC-RL V1.04.00
MCU 動作周波数	32MHz
動作電圧	3V
DC-DC コンバータ	RF 内蔵 DC-DC コンバータ未使用
RF スロー・クロック供給源	RF 内蔵オシレータ使用、キャリブレーション実行
RF 送受信動作	RF 送信のみ有効化(RFCFG_TX)
送信パワー	0dBm
送信チャンネル	3ch 送信 (37,38,39ch)
Advertising タイプ	ADV_NONCONN_IND
Advertising データ長	31byte
Advertising データ数	1
Advertising インターバル	100msec

MCU 動作周波数 32MHz の場合のビーコンスタック起動時の消費電流波形を図 9-5 に示します。

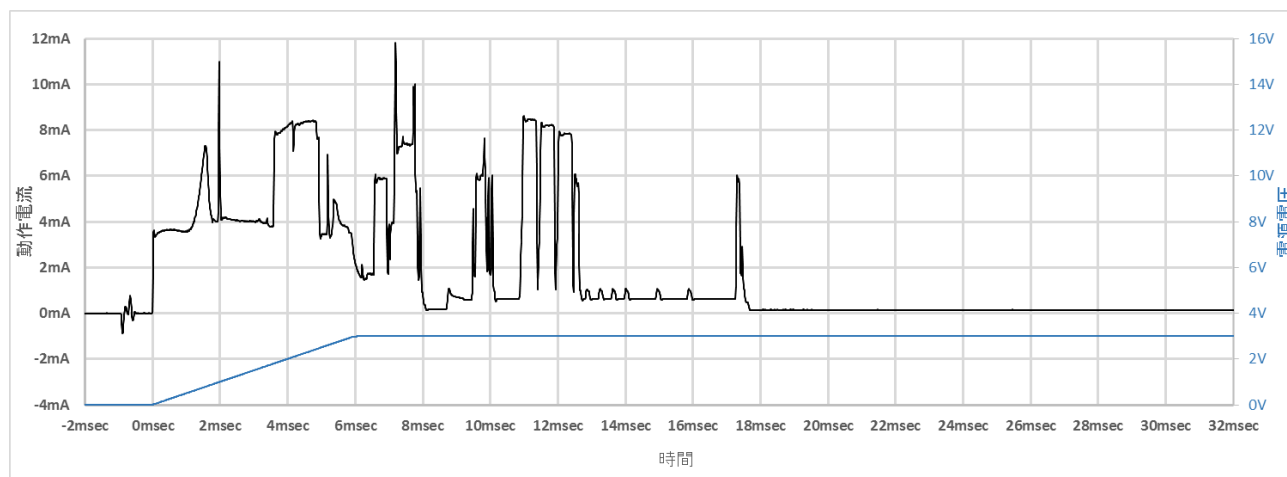


図 9-5 MCU 動作周波数 32MHz でのビーコンスタック起動電流波形

9.1.4 測定結果：周期的な Advertising パケット送信

(1) MCU 動作周波数 4MHz

MCU 動作周波数 4MHz における平均消費電流測定時の動作設定を表 9-6 に示します。

表 9-6 平均消費電流測定時の動作設定

分類	測定条件
コンパイラ	CC-RL V1.04.00
MCU 動作周波数	4MHz
動作電圧	3V
DC-DC コンバータ	RF 内蔵 DC-DC コンバータ使用 / RF 内蔵 DC-DC コンバータ未使用
RF スロー・クロック供給源	RF 内蔵オシレータ使用、キャリブレーション実行
RF 送受信動作	RF 送受信を有効化(RFCFG_TXRX)
送信パワー	0dBm
送信チャンネル	3ch 送信 (37,38,39ch)
Advertising タイプ	ADV_NONCONN_IND / ADV_SCAN_IND
Advertising データ長	31byte
Advertising データ数	1
Advertising インターバル	10msec / 100msec / 1sec ※Bluetooth Core Specification が定義する範囲は 100msec~10.24sec

MCU 動作周波数 4MHz における平均消費電流の測定結果を表 9-7 と表 9-8 に示します。

表 9-7 MCU 動作周波数 4MHz における測定結果 (ADV_NONCONN_IND 送信時)

Advertising Interval	平均消費電流の測定結果 (ADV_NONCONN_IND 送信時)	
	DC-DC コンバータ未使用	DC-DC コンバータ使用
10msec	1540.8uA	1041.1uA
100msec	172.2uA	126.4uA
1sec	18.8uA	14.2uA

表 9-8 MCU 動作周波数 4MHz における測定結果 (ADV_SCAN_IND 送信時)

Advertising Interval	平均消費電流の測定結果 (ADV_SCAN_IND 送信時、SCAN_REQ 受信なし)	
	DC-DC コンバータ未使用	DC-DC コンバータ使用
10msec	1806.1uA	1211.8uA
100msec	217.3uA	160.3uA
1sec	23.2uA	17.6uA

MCU 動作周波数 4MHz、Advertising インターバル 1sec の場合の ADV_NONCONN_IND パケット送信時の消費電流波形を図 9-6 と図 9-7 に示します。

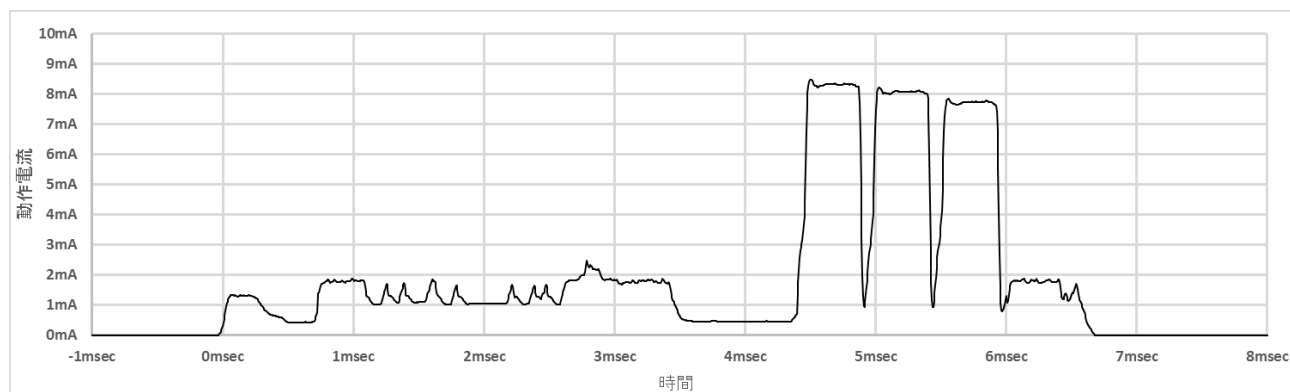


図 9-6 MCU 動作周波数 4MHz での ADV_NONCONN_IND 送信電流波形 (DC-DC コンバータ未使用)

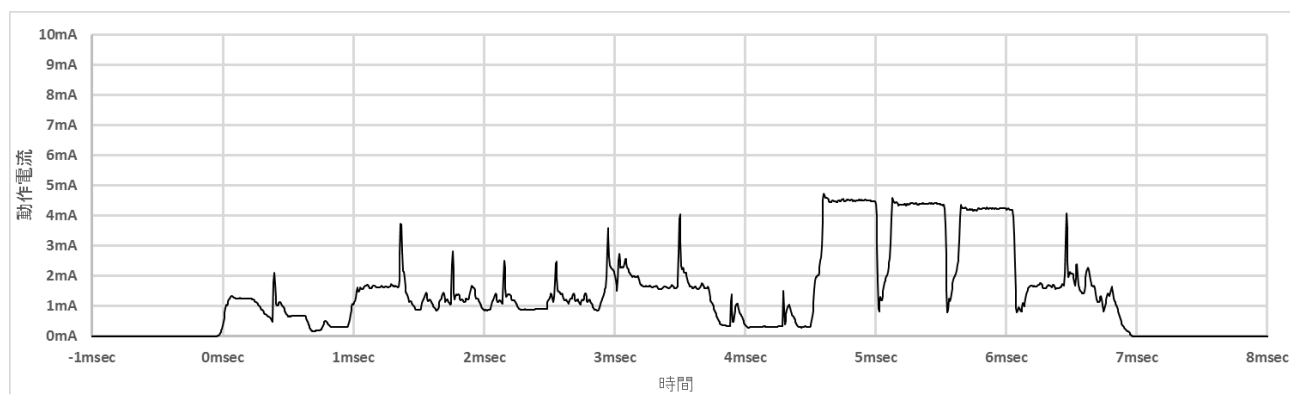


図 9-7 MCU 動作周波数 4MHz での ADV_NONCONN_IND 送信電流波形 (DC-DC コンバータ使用)

MCU 動作周波数 4MHz、Advertising インターバル 1sec、SCAN_REQ パケット受信がない場合の ADV_SCAN_IND パケット送信時の消費電流波形を図 9-8 と図 9-9 に示します。

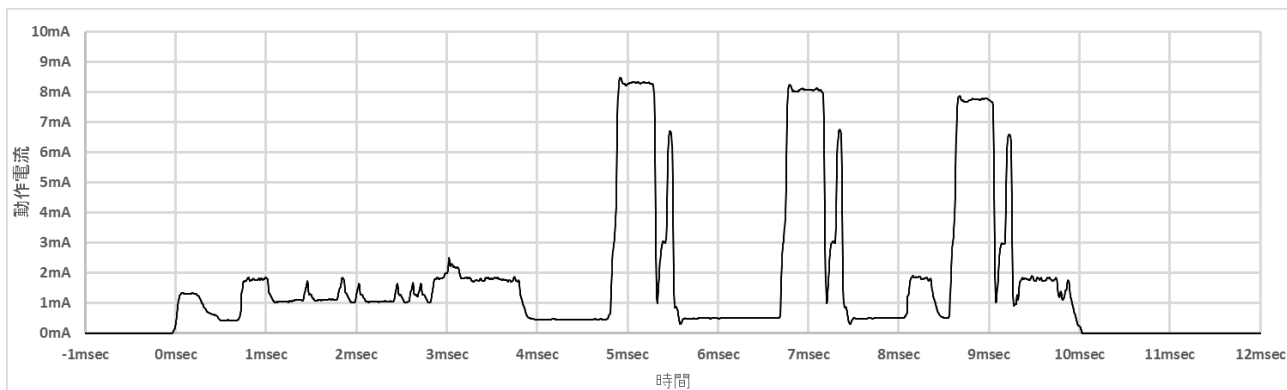


図 9-8 MCU 動作周波数 4MHz での ADV_SCAN_IND 送信電流波形 (DC-DC コンバータ未使用)

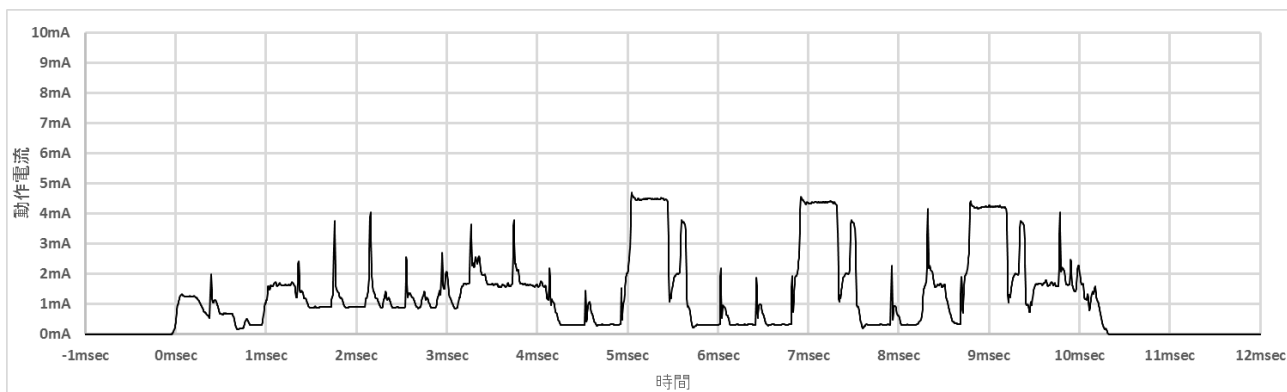


図 9-9 MCU 動作周波数 4MHz での ADV_SCAN_IND 送信電流波形 (DC-DC コンバータ使用)

(2) MCU 動作周波数 8MHz

MCU 動作周波数 8MHz における平均消費電流測定時の動作設定を表 9-9 に示します。

表 9-9 平均消費電流測定時の動作設定

分類	測定条件
コンパイラ	CC-RL V1.04.00
MCU 動作周波数	8MHz
動作電圧	3V
DC-DC コンバータ	RF 内蔵 DC-DC コンバータ使用 / RF 内蔵 DC-DC コンバータ未使用
RF スロー・クロック供給源	RF 内蔵オシレータ使用、キャリブレーション実行
RF 送受信動作	RF 送受信を有効化(RFCFG_TXRX)
送信パワー	0dBm
送信チャンネル	3ch 送信 (37,38,39ch)
Advertising タイプ	ADV_NONCONN_IND / ADV_SCAN_IND
Advertising データ長	31byte
Advertising データ数	1
Advertising インターバル	10msec / 100msec / 1sec ※Bluetooth Core Specification が定義する範囲は 100msec~10.24sec

MCU 動作周波数 8MHz における平均消費電流の測定結果を表 9-10 と表 9-11 に示します。

表 9-10 MCU 動作周波数 8MHz における測定結果 (ADV_NONCONN_IND 送信時)

Advertising Interval	平均消費電流の測定結果 (ADV_NONCONN_IND 送信時)	
	DC-DC コンバータ未使用	DC-DC コンバータ使用
10msec	1518.8uA	1015.9uA
100msec	148.5uA	101.8uA
1sec	16.5uA	11.8uA

表 9-11 MCU 動作周波数 8MHz における測定結果 (ADV_SCAN_IND 送信時)

Advertising Interval	平均消費電流の測定結果 (ADV_SCAN_IND 送信時、SCAN_REQ 受信なし)	
	DC-DC コンバータ未使用	DC-DC コンバータ使用
10msec	1751.7uA	1178.5uA
100msec	187.5uA	132.7uA
1sec	20.4uA	14.7uA

MCU 動作周波数 8MHz、Advertising インターバル 1sec の場合の ADV_NONCONN_IND パケット送信時の消費電流波形を図 9-10 と図 9-11 に示します。

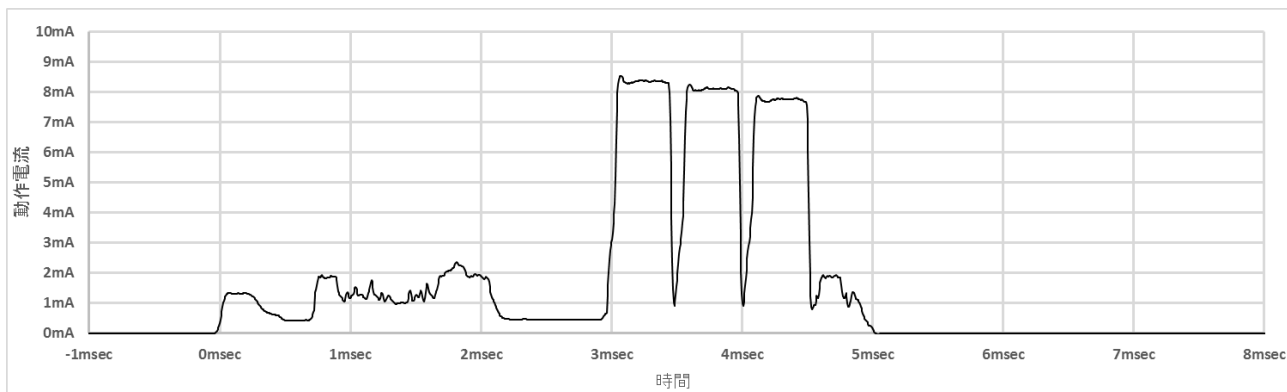


図 9-10 MCU 動作周波数 8MHz での ADV_NONCONN_IND 送信電流波形 (DC-DC コンバータ未使用)

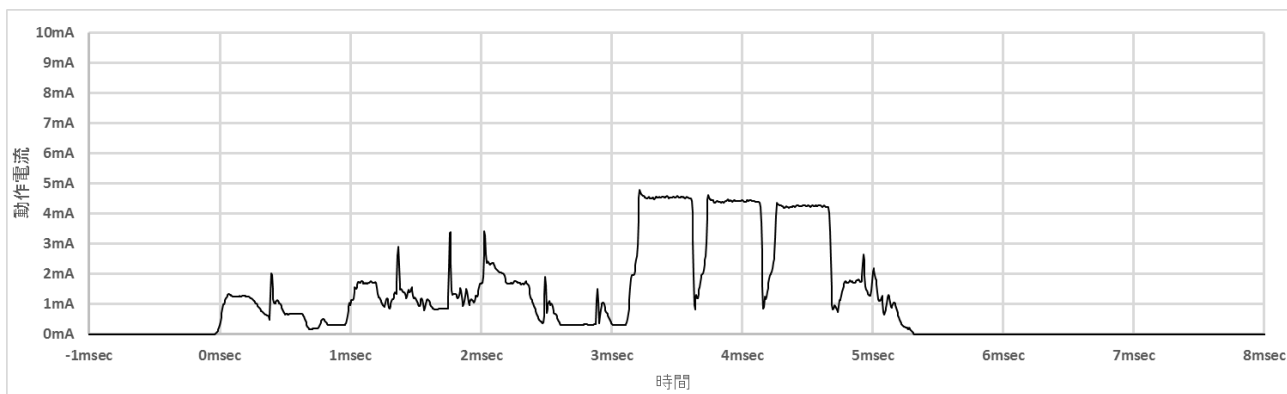


図 9-11 MCU 動作周波数 8MHz での ADV_NONCONN_IND 送信電流波形 (DC-DC コンバータ使用)

MCU 動作周波数 8MHz、Advertising インターバル 1sec、SCAN_REQ パケット受信がない場合の ADV_SCAN_IND パケット送信時の消費電流波形を図 9-12 と図 9-13 に示します。

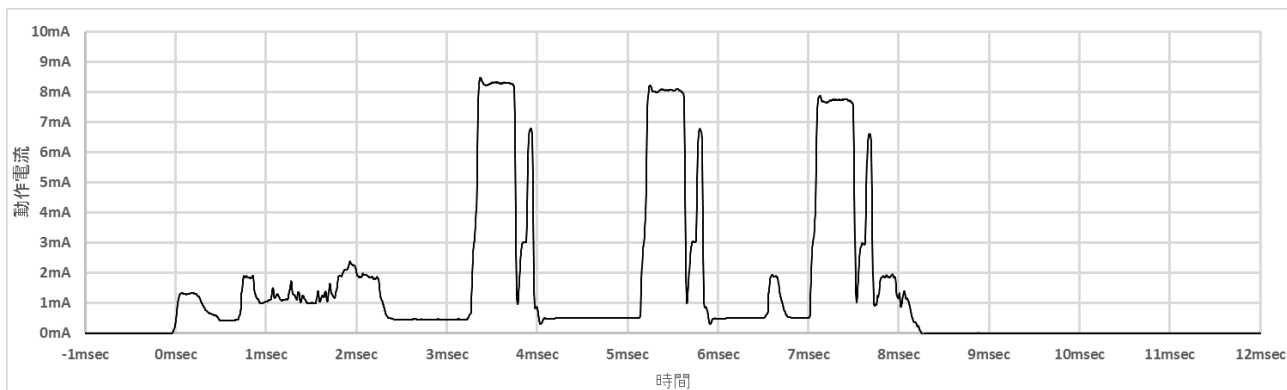


図 9-12 MCU 動作周波数 8MHz での ADV_SCAN_IND 送信電流波形 (DC-DC コンバータ未使用)

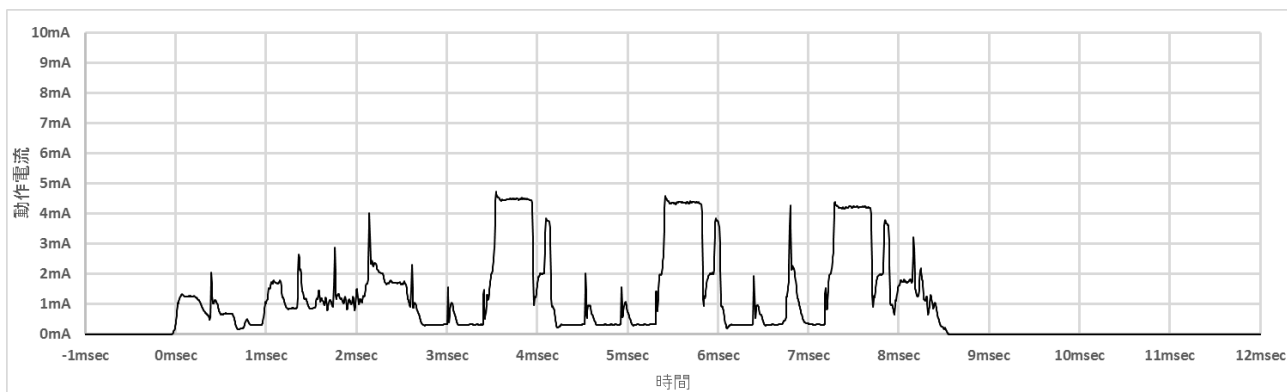


図 9-13 MCU 動作周波数 8MHz での ADV_SCAN_IND 送信電流波形 (DC-DC コンバータ使用)

(3) MCU 動作周波数 16MHz

MCU 動作周波数 16MHz における平均消費電流測定時の動作設定を表 9-12 に示します。

表 9-12 平均消費電流測定時の動作設定

分類	測定条件
コンパイラ	CC-RL V1.04.00
MCU 動作周波数	16MHz
動作電圧	3V
DC-DC コンバータ	RF 内蔵 DC-DC コンバータ使用 / RF 内蔵 DC-DC コンバータ未使用
RF スロー・クロック供給源	RF 内蔵オシレータ使用、キャリブレーション実行
RF 送受信動作	RF 送受信を有効化(RFCFG_TXRX)
送信パワー	0dBm
送信チャンネル	3ch 送信 (37,38,39ch)
Advertising タイプ	ADV_NONCONN_IND / ADV_SCAN_IND
Advertising データ長	31byte
Advertising データ数	1
Advertising インターバル	10msec / 100msec / 1sec ※Bluetooth Core Specification が定義する範囲は 100msec~10.24sec

MCU 動作周波数 16MHz における平均消費電流の測定結果を表 9-13 と表 9-14 に示します。

表 9-13 MCU 動作周波数 16MHz における測定結果 (ADV_NONCONN_IND 送信時)

Advertising Interval	平均消費電流の測定結果 (ADV_NONCONN_IND 送信時)	
	DC-DC コンバータ未使用	DC-DC コンバータ使用
10msec	1536.7uA	1031.3uA
100msec	157.9uA	111.5uA
1sec	17.4uA	12.8uA

表 9-14 MCU 動作周波数 16MHz における測定結果 (ADV_SCAN_IND 送信時)

Advertising Interval	平均消費電流の測定結果 (ADV_SCAN_IND 送信時、SCAN_REQ 受信なし)	
	DC-DC コンバータ未使用	DC-DC コンバータ使用
10msec	1786.0uA	1191.9uA
100msec	199.2uA	143.2uA
1sec	21.6uA	16.8uA

MCU 動作周波数 16MHz、Advertising インターバル 1sec の場合の ADV_NONCONN_IND パケット送信時の消費電流波形を図 9-14 と図 9-15 に示します。

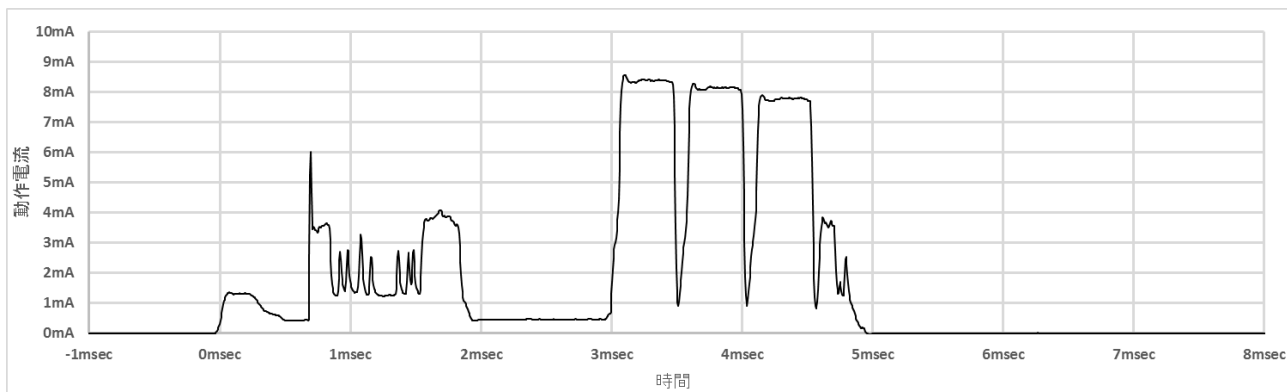


図 9-14 MCU 動作周波数 16MHz での ADV_NONCONN_IND 送信電流波形 (DC-DC コンバータ未使用)

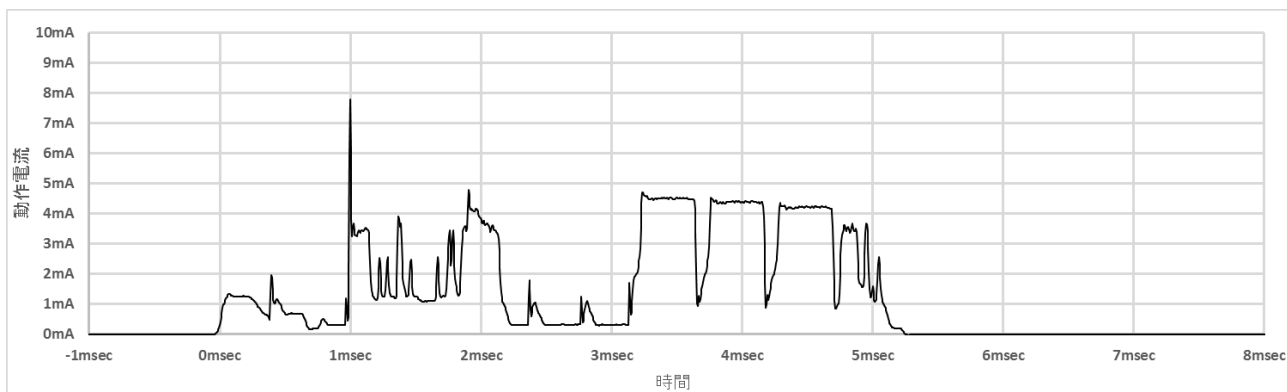


図 9-15 MCU 動作周波数 16MHz での ADV_NONCONN_IND 送信電流波形 (DC-DC コンバータ使用)

MCU 動作周波数 16MHz、Advertising インターバル 1sec、SCAN_REQ パケット受信がない場合の ADV_SCAN_IND パケット送信時の消費電流波形を図 9-16 と図 9-17 に示します。

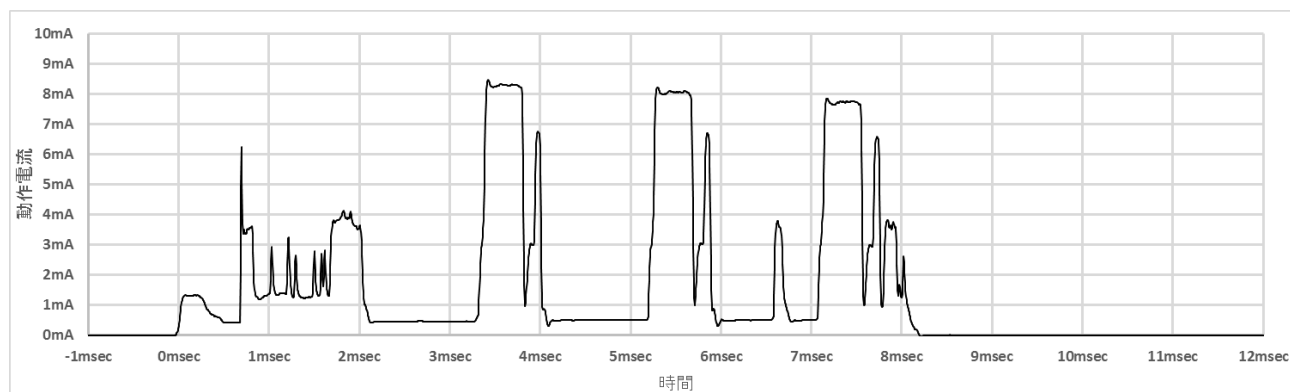


図 9-16 MCU 動作周波数 16MHz での ADV_SCAN_IND 送信電流波形 (DC-DC コンバータ未使用)

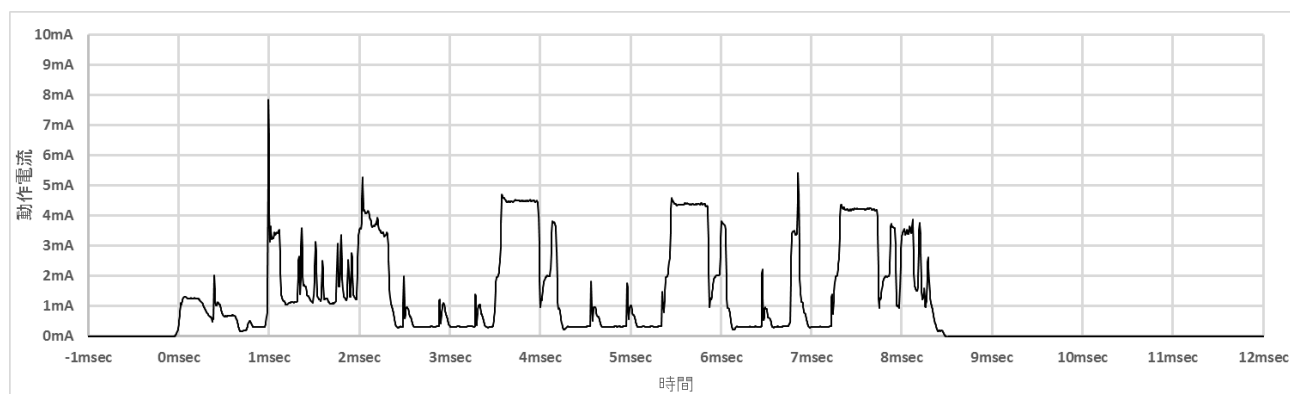


図 9-17 MCU 動作周波数 16MHz での ADV_SCAN_IND 送信電流波形 (DC-DC コンバータ使用)

(4) MCU 動作周波数 32MHz

MCU 動作周波数 32MHz における平均消費電流測定時の動作設定を表 9-15 に示します。

表 9-15 平均消費電流測定時の動作設定

分類	測定条件
コンパイラ	CC-RL V1.04.00
MCU 動作周波数	32MHz
動作電圧	3V
DC-DC コンバータ	RF 内蔵 DC-DC コンバータ使用 / RF 内蔵 DC-DC コンバータ未使用
RF スロー・クロック供給源	RF 内蔵オシレータ使用、キャリブレーション実行
RF 送受信動作	RF 送受信を有効化(RFCFG_TXRX)
送信パワー	0dBm
送信チャンネル	3ch 送信 (37,38,39ch)
Advertising タイプ	ADV_NONCONN_IND / ADV_SCAN_IND
Advertising データ長	31byte
Advertising データ数	1
Advertising インターバル	10msec / 100msec / 1sec ※Bluetooth Core Specification が定義する範囲は 100msec~10.24sec

MCU 動作周波数 32MHz における平均消費電流の測定結果を表 9-16 と表 9-17 に示します。

表 9-16 MCU 動作周波数 32MHz における測定結果 (ADV_NONCONN_IND 送信時)

Advertising Interval	平均消費電流の測定結果 (ADV_NONCONN_IND 送信時)	
	DC-DC コンバータ未使用	DC-DC コンバータ使用
10msec	1542.2uA	1037.3uA
100msec	165.4uA	119.8uA
1sec	18.2uA	13.6uA

表 9-17 MCU 動作周波数 32MHz における測定結果 (ADV_SCAN_IND 送信時)

Advertising Interval	平均消費電流の測定結果 (ADV_SCAN_IND 送信時、SCAN_REQ 受信なし)	
	DC-DC コンバータ未使用	DC-DC コンバータ使用
10msec	1812.4uA	1216.8uA
100msec	209.2uA	153.1uA
1sec	22.5uA	16.9uA

MCU 動作周波数 32MHz、Advertising インターバル 1sec の場合の ADV_NONCONN_IND パケット送信時の消費電流波形を図 9-18 と図 9-19 に示します。

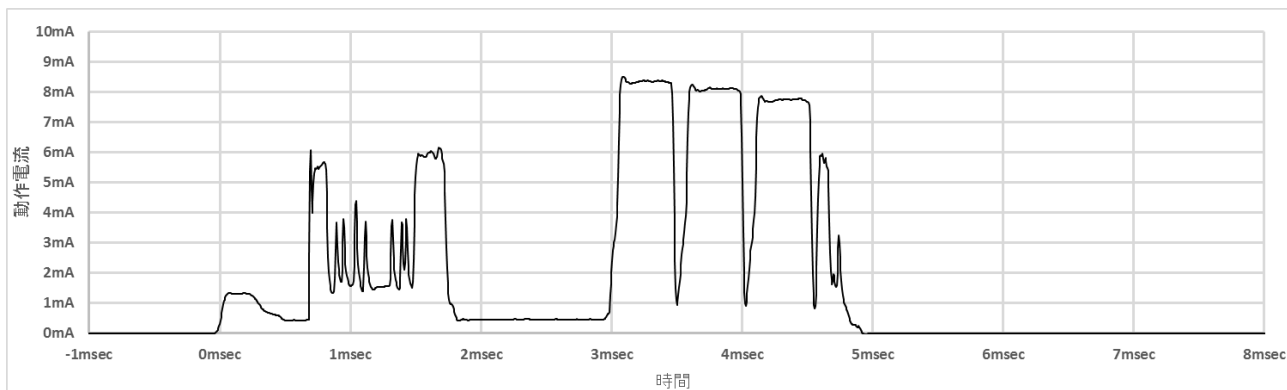


図 9-18 MCU 動作周波数 32MHz での ADV_NONCONN_IND 送信電流波形 (DC-DC コンバータ未使用)

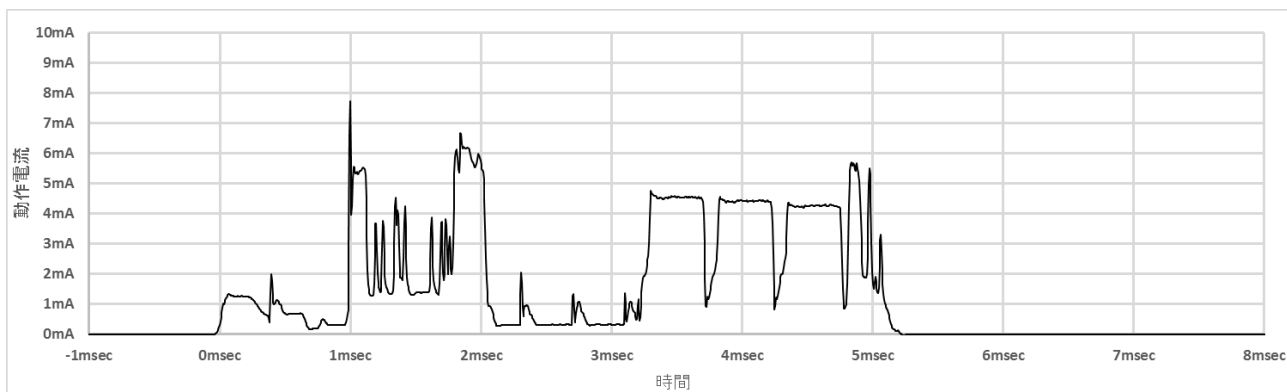


図 9-19 MCU 動作周波数 32MHz での ADV_NONCONN_IND 送信電流波形 (DC-DC コンバータ使用)

MCU 動作周波数 32MHz、Advertising インターバル 1sec、SCAN_REQ パケット受信がない場合の ADV_SCAN_IND パケット送信時の消費電流波形を図 9-20 と図 9-21 に示します。

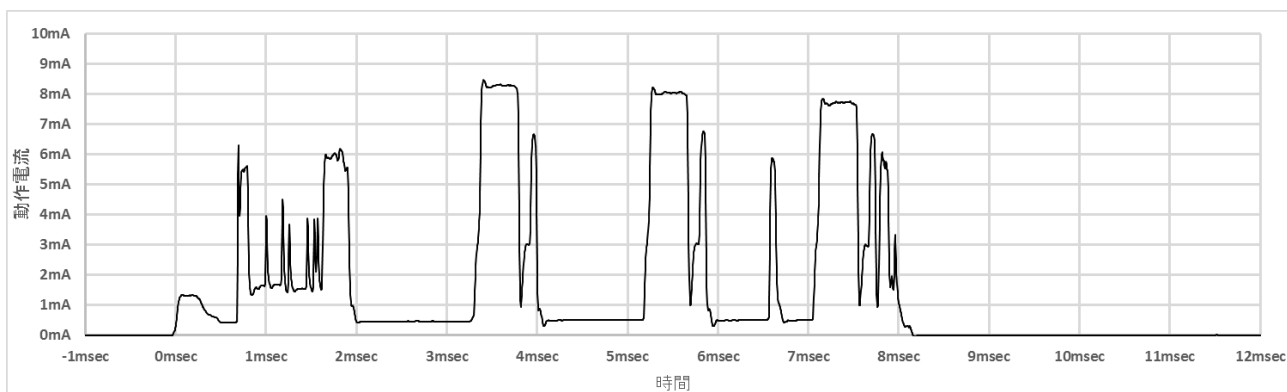


図 9-20 MCU 動作周波数 32MHz での ADV_SCAN_IND 送信電流波形 (DC-DC コンバータ未使用)

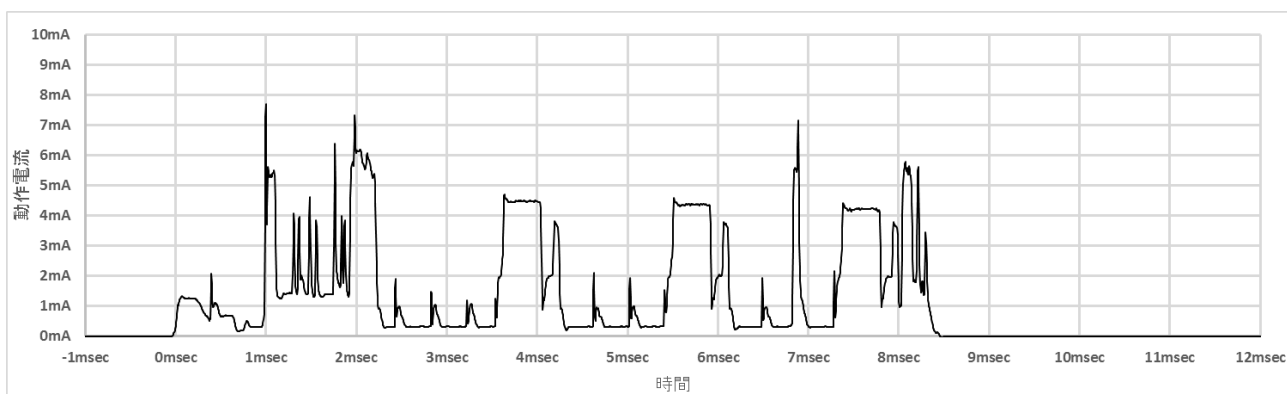


図 9-21 MCU 動作周波数 32MHz での ADV_SCAN_IND 送信電流波形 (DC-DC コンバータ使用)

9.2 デバイスアドレス

デバイスアドレスはデバイスを識別するための 48bit の値です。Bluetooth Core Specification で定義されるデバイスアドレスを以下に示します。

- Public Device Address
 - Public Device Address は IEEE 802-2001 のセクション"48-bit universal LAN MAC addresses"に従って生成すべきデバイスアドレスで、IEEE Registration Authority から取得した Organizationally Unique Identifier(OUI)を使用します。
- Random Device Address
 - Static Device Address
 - Static Device Address は 48-bit 乱数値で生成します。デバイスは起動後、新しい乱数値による Static Device Address の生成と使用が可能です。Static Device Address の使用後は、次の起動時まで Static Device Address の値を変更すべきではありません。
 - Private Device Address
 - Non-resolvable Private Address
 - Non-resolvable Private Address は 48-bit 乱数値で生成します。Non-resolvable Private Address は他デバイスからの追跡の可能性を低減するために使用され、一定時間 (Bluetooth Core Specification による推奨は 15 分ごと) で変更すべきものです。
 - Resolvable Private Address
 - Resolvable Private Address は 24-bit 乱数値と、乱数値と Identity Resolving Key(IRK)による 24-bit ハッシュ値で構成されます。Resolvable Private Address は他デバイスからの追跡の可能性を低減するために使用され、一定時間 (Bluetooth Core Specification による推奨は 15 分ごと) で変更すべきものです。

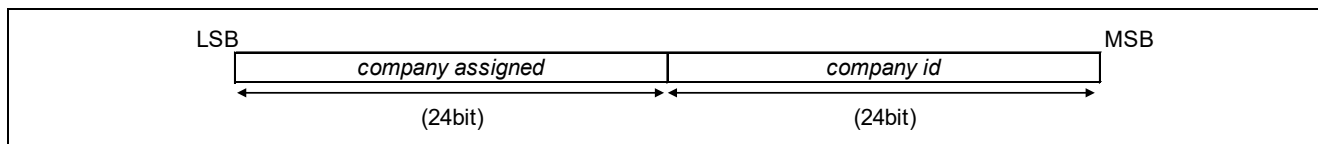


図 9-22 Public Device Address フォーマット

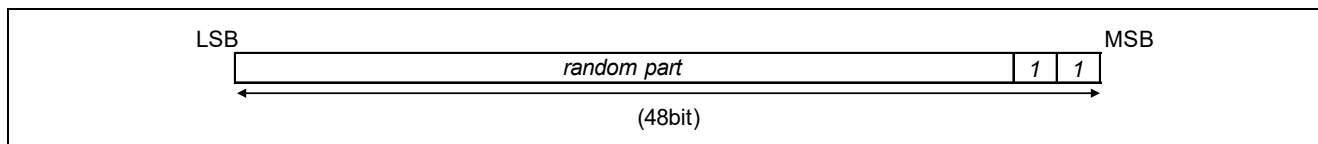


図 9-23 Static Device Address フォーマット

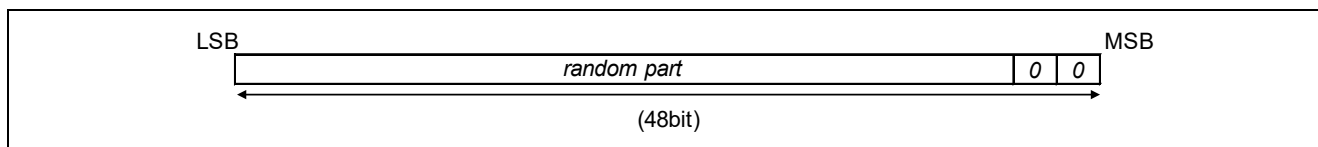


図 9-24 Non-resolvable Private Device Address フォーマット

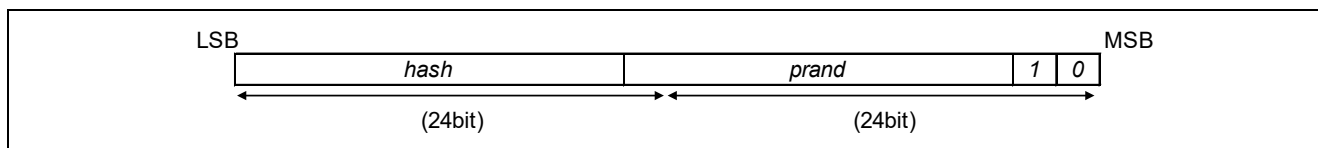


図 9-25 Resolvable Private Device Address フォーマット

デバイスアドレスの仕様については『Bluetooth Core Specification v4.2』の [Vol. 6, Part B] Section 1.3 を参照してください。

9.3 Advertising パケットフォーマット

ビーコンアプリケーションは Non-connectable Undirected Advertising パケットまたは Scannable Undirected Advertising パケットを送信します。パケットフォーマットは共通であり、図 9-26 に示します。

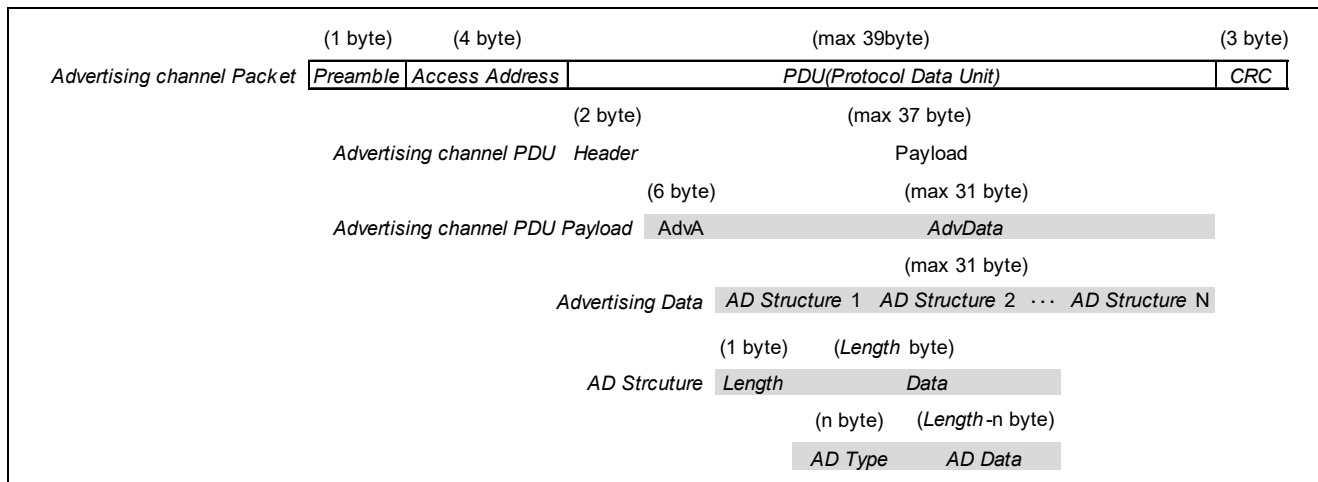


図 9-26 Advertising パケットフォーマット

Advertising パケットは以下のフィールドで構成されます。

- Advertising channel packet
 - Preamble : 10101010b 固定
 - Access Address : 0x8E89BED6 固定
 - Advertising channel PDU : Header と Payload
 - CRC : 24bits

下記フィールドはアプリケーションによって設定されます。

- Advertising channel PDU Payload
 - AdvA : Advertiser's Address を格納
 - AdvData (Advertising data) : 複数の AD structure で構成され、最大 31 bytes
 - AD structure : 1 byte の Length 情報と Length byte の Data
 - Data : n byte の AD Type と (Length-n) byte の AD Data

詳細についてはそれぞれ下記の仕様を参照してください。

- Advertising packet format : 『Bluetooth Core Specification v4.2』 [Vol. 6, Part B] Section 2.1
- Advertising channel PDU format : 『Bluetooth Core Specification v4.2』 [Vol. 6, Part B] Section 2.3
- Advertising Data format : 『Bluetooth Core Specification v4.2』 [Vol. 3, Part C] Section 11
- AD Type : 『Supplement to the Bluetooth Core Specification v6』 Part A

AD Type の定義については下記の WEB サイトを参照してください。

- Bluetooth SIG Home > Specification > Assigned Numbers > Generic Access Profile
<https://www.bluetooth.com/specifications/assigned-numbers/generic-access-profile>

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

<https://www.renesas.com/>

お問い合わせ先

<https://www.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
1.00	2016.04.13	初版発行
1.01	2016.06.21	3章 ビーコンアプリケーション
		- 3.7節の Renesas Flash Programmer バージョンを更新
		- 3.8.2項に Renesas Flash Programmer 準備手順を更新
		- 表 3-1 の SW7 設定に USB からの電源供給時を追記
		4章 Appendix
		- 本章を新規追加
- 4.1節 周期的なパケット送信時の消費電流を新規追加		
2.00	2016.10.26	ビーコンスタックの仕様・機能・APIの節を分離して『RL78/G1D ビーコンスタック ユーザーズマニュアル』(R01UW0171)として発行
		5章 仕様
		P.44 5.3節 スキャンアプリケーションを新規追加
		P.54 5.4節 DTM アプリケーションを新規追加
		7章 関数
		P.74 7.1.2節 スキャンアプリケーションを新規追加
		P.74 7.1.3節 DTM アプリケーションを新規追加
		8章 動作
		P.77 8.1.2項 スキャンアプリケーションを新規追加
		P.78 8.1.3項 DTM アプリケーションを新規追加
		P.80 8.2節 シーケンスを新規追加
		9章 Appendix
		P.102 9.2節 デバイスアドレスを新規追加
		2.01
P.14 表 4-1 のスライドスイッチ SW9 設定を修正		
5章 仕様		
P.58 5.6節のハードウェアリソース表にサンプルプログラムが使用するポートを追加		
2.10	2017.03.09	2章 動作環境
		P.8 ソフトウェア環境のコンパイラと開発環境を更新
		3章 ファイル構成
		P.9 ファイル構成にファームウェアファイルを追加
		5章 仕様
		P.49 5.3.2項にコマンドのバイナリ例を追加
		P.56 表 5-16 にアプリケーションが読み込むパラメータを記載
		P.59 5.7節のコンパイラバージョンを更新
		6章 設定
		P.67 6.1.9項 エナジーハーベスティング向けハードウェア設定を新規追加
P.68 6.2.1項 アプリケーション選択設定を新規追加		
2.11	2017.10.12	4章 操作方法
		P.14 表 4-1 のスライドスイッチ SW13 設定を修正
		P.15 4.2節の書き込み手順からユニークコードファイルの設定を分離
		6章 設定
		P.68 6.2.1項にファームウェアファイルのビルド条件を追記
		9章 Appendix
		P.103 9.3節の Advertising パケットのフィールド説明を更新
全体		

		-	改行コードを"LF"に変更
		-	一部の章名、項名を変更
2.20	2020.06.19	要旨	
		P.1	ビーコンアプリケーションの ASCII コマンド送信について追加
		1章 概要	
		P.6	1.2 節 ビーコンアプリケーション-ASCII コマンド制御の概要を追加
		2章 動作環境	
		P.8	ソフトウェア環境を更新
		3章 ファイル構成	
		P.9	ファイル構成を更新
		4章 操作方法	
		P.12	4.1 節 ビーコンアプリケーション-ASCII コマンド制御のビルドを追加
		P.16	4.3 節 ビーコンアプリケーション-ASCII コマンド制御の選択を追加
		P.18	4.3.2 項 ビーコンアプリケーション-ASCII コマンド制御の動作確認方法を追加
		5章 仕様	
		P.28	5.2 節 ビーコンアプリケーション-ASCII コマンド制御のコマンド仕様を追加
		P.57	5.5.2 項 ビーコンアプリケーション-ASCII コマンド制御のコードフラッシュメモリのシステム動作設定を追加
		P.58	5.6 節 ビーコンアプリケーション-ASCII コマンド制御の使用ハードウェアを追加
		P.59	5.7 節 コンパイラバージョンを更新
		P.59	5.9 節 ビーコンアプリケーションのプログラムサイズを更新、ビーコンアプリケーション-ASCII コマンド制御のプログラムサイズを追加
		P.60	5.10 節 アドレスマップを更新
		6章 設定	
		P.68	6.2.1 アプリケーション選択設定項 ビーコンアプリケーション-ASCII コマンド制御の選択方法を追加
		P.70	6.2.3 項 ビーコンアプリケーション-ASCII コマンド制御のユニークコードファイル設定を追加
		7章 関数	
		P.75	7.1.4 節 ビーコンアプリケーション-ASCII コマンド制御の制御関数を追加
		8章 動作	
		P.79	8.1.4 節 ビーコンアプリケーション-ASCII コマンド制御の状態遷移図を追加

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. 未使用端子の処理

【注意】未使用端子は、本文の「未使用端子の処理」に従って処理してください。

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

2. 電源投入時の処置

【注意】電源投入時は、製品の状態は不定です。

電源投入時には、LSI の内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。

外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。

同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. リザーブアドレス（予約領域）のアクセス禁止

【注意】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、将来の機能拡張用に割り付けられているリザーブアドレス（予約領域）がありません。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

4. クロックについて

【注意】リセット時は、クロックが安定した後、リセットを解除してください。

プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。

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

5. 製品間の相違について

【注意】型名の異なる製品に変更する場合は、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っていると、内部 ROM、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。型名が異なる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、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、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いません。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て、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といたします。
 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)



ルネサスエレクトロニクス株式会社

■営業お問合せ窓口

<http://www.renesas.com>

営業お問合せ窓口の住所は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス株式会社 〒135-0061 東京都江東区豊洲3-2-24（豊洲フォレシア）

■技術的なお問合せおよび資料のご請求は下記へどうぞ。
総合お問合せ窓口：<https://www.renesas.com/contact/>