

# Bluetooth® Low Energy プロトコルスタック

## Fast Prototyping Board BLE & LoRaWAN®アプリケーション

### 要旨

LoRaWAN®通信に対応したセンサデバイスは、Bluetooth® Low Energy を利用して、スマートフォンから簡単に、LoRaWAN 通信で必要となる暗号鍵やサーバ ID、センサデバイスのアプリケーションで必要となる管理番号や設置場所等の情報を設定できることが求められています。LoRaWAN 通信に対応したセンサデバイスと Bluetooth Low Energy を組合わせた応用事例として下記があります。

- デバイス・ファームウェアをアップデート
- デバイスから直接、現在のセンサデータを確認
- デバイスの故障診断
- デバイスの位置を BLE と連携してスマートフォンで管理
- デバイスの Application EUI/Key を BLE と連携してスマートフォンで管理

本アプリケーションノートでは、「RL78/G14 Fast Prototyping Board」をメイン MCU として、BLE 通信ができる「RL78/G1D BLE Module Expansion Board」、LoRaWAN 通信ができる「LoRaWAN stack sample application board」、そしてセンサの「TE Connectivity PMOD\_MS8607」を用いて IoT 無線通信を実現する方法を説明します。BLE 通信で LoRaWAN 通信に必要なパラメータの設定を行い、LoRaWAN 通信でセンサデータを LoRaWAN Gateway へ送信します。

### 動作確認デバイス

RL78/G14 Fast Prototyping Board (製品型名:RTK5RLG140C00000BJ)

RL78/G1D BLE Module Expansion Board (製品型名: RTKYRLG1D0B00000BJ)

LoRaWAN stack sample application board

## 関連資料

資料名	資料番号	
	和文	英文
Bluetooth Low Energy プロトコルスタック		
ユーザーズマニュアル	R01UW0095J	R01UW0095E
API リファレンスマニュアル 基本編	R01UW0088J	R01UW0088E
サンプルプログラムアプリケーションノート	R01AN1375J	R01AN1375E
rBLE コマンド仕様書	R01AN1376J	R01AN1376E
RL78/G1D		
ユーザーズマニュアル ハードウェア編	R01UH0515J	R01UH0515E
RL78/G1D モジュール		
ユーザーズマニュアル ハードウェア編	R02UH0004J	R02UH0004E
ファームウェア ユーザーズマニュアル	R01UW0160J	R01UW0160E
モジュール制御ソフトウェア	R01AN3362J	R01AN3362E
LoRaWAN		
LoRaWAN Stack Sample Application Command Reference	-	R11AN0231E
LoRaWAN IoT Demo LPWA IoT Solution with Cloud	-	R11AN0412E
RL78/G14		
ユーザーズマニュアル ハードウェア編	R01UH0186J	R01UH0186E

## 目次

1. 概要	5
2. 開発環境	6
2.1 ハードウェア環境	6
2.2 ソフトウェア環境	6
3. アプリケーション構成	7
3.1 システム構成	7
3.1.1 FPB 接続端子	8
3.2 ソフトウェア構成	10
3.2.1 BLE module 構成	10
3.2.2 LoRaWAN module 構成	12
3.3 周辺機能構成	13
3.4 ファイル構成	14
4. ビルド手順	17
4.1 CS+ for CC	17
4.2 e2 studio	17
5. アプリケーションの実行	18
5.1 GATTBrowser のインストール	18
5.2 LORIoT LoRaWAN Network Server の準備	18
5.3 実行環境	18
5.4 実行手順	19
5.4.1 Android デバイスでの LoRaWAN module パラメータ設定	22
5.4.2 iOS デバイスでの LoRaWAN module パラメータ設定	25
5.4.3 LoRaWAN Network Server	28
5.4.3.1 センサデータの確認	28
5.4.3.2 LoRaWAN Gateway 通信終了コマンドの送信	28
6. 動作仕様	30
6.1 メインループ	30
6.2 BLE module 通信フロー	31
6.2.1 スマートフォンとの接続通信	31
6.2.2 スマートフォンとの切断	32
6.3 LoRaWAN module 通信フロー	33
6.4 LoRaWAN module コマンド	34
6.4.1 パラメータ設定 AT コマンド	34
6.4.1.1 AT コマンドフォーマット	34
6.4.1.2 AT コマンド結果コードフォーマット	35
6.4.1.3 AT コマンドパラメータフラグ	36
6.4.2 LoRaWAN Gateway 通信終了コマンド	36
6.5 BLE Modem 構成 UART 2 線分岐接続方式	37
6.5.1 送信動作	37
6.5.2 受信動作	38

7. BLE 通信シーケンス .....	39
7.1 メインシーケンス .....	39
7.2 Step1. rBLE Initialize シーケンス .....	40
7.3 Step2. GAP Initialize シーケンス .....	40
7.4 Step3. Broadcast シーケンス .....	41
7.5 Step4. Connection シーケンス .....	41
7.6 Step5. Profile Enable シーケンス .....	42
7.7 Step6. Remote Device Check シーケンス .....	43
7.8 Step7. Pairing シーケンス .....	44
7.9 Step8. Start Encryption シーケンス .....	46
7.10 Step9. Profile Communication シーケンス .....	46
7.11 Step10. Disconnection シーケンス .....	47
8. LoRaWAN module AT コマンドシーケンス .....	48
8.1 メインシーケンス .....	48
8.2 Step1. Start program .....	48
8.3 Step2. Set LoRaWAN parameters .....	49
8.4 Step3. Send sensor data .....	50
9. 付録 .....	51
9.1 ROM・RAM サイズ .....	51
9.2 参考文献 .....	51
改訂記録 .....	52

Bluetooth® のワードマークおよびロゴは、Bluetooth SIG, Inc. が所有する登録商標であり、ルネサス エレクトロニクス株式会社はこれらのマークをライセンスに基づいて使用しています。その他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。

LoRaWAN®の名称は、Semtech Corporation の登録商標です。

Wirnet™の名称は、Kerlink の商標です。

Pmod™の名称は、Digilent Inc.の商標です。

## 1. 概要

LoRaWAN 通信に対応したセンサデバイスは、Bluetooth Low Energy を利用して、スマートフォンから簡単に、LoRaWAN 通信で必要となる暗号鍵やサーバ ID、センサデバイスのアプリケーションで必要となる管理番号や設置場所等の情報を設定できることが求められています。LoRaWAN 通信に対応したセンサデバイスと Bluetooth Low Energy を組合わせた応用事例として下記があります。

- デバイス・ファームウェアをアップデート
- デバイスから直接、現在のセンサデータを確認
- デバイスの故障診断
- デバイスの位置を BLE と連携してスマートフォンで管理
- デバイスの Application EUI/Key を BLE と連携してスマートフォンで管理

本アプリケーションノートでは、「RL78/G14 Fast Prototyping Board」をメイン MCU として、BLE 通信ができる「RL78/G1D BLE Module Expansion Board」、LoRaWAN 通信ができる「LoRaWAN stack sample application board」、そしてセンサの「TE Connectivity PMOD\_MS8607」を用いて IoT 無線通信を実現する方法を説明します。BLE 通信で LoRaWAN 通信に必要なパラメータの設定を行い、LoRaWAN 通信でセンサデータを LoRaWAN Gateway へ送信します。

RL78/G14 Fast Prototyping Board の 2 つの Pmod™インタフェースに RL78/G1D BLE Module Expansion Board と LoRaWAN stack sample application board を接続して、MCU ヘッドに TE Connectivity PMOD\_MS8607 を接続します。スマートフォンとの BLE 通信で LoRaWAN 通信に必要なパラメータの設定を行い、LoRaWAN 通信でセンサデータを LoRaWAN Gateway へ送信します。LoRaWAN Gateway へ送信したセンサデータは LoRaWAN Network Server で確認します。

- RL78/G14 Fast Prototyping Board は、RL78/G14 マイコンを搭載したアプリケーションの試作開発に特化した評価ボードです。E2 エミュレータ Lite 相当のエミュレータサーキットを内蔵し、追加のツール無しでプログラムの書き込み/デバッグが可能です。  
以降、「FPB」と記載します。
- RL78/G1D BLE Module Expansion Board は、RL78/G1D モジュール(RY7011)を搭載した Pmod インタフェースの評価ボードです。動作確認用ファームウェアが書き込まれているので、BLE プロトコルスタック・モデム構成の Host MCU プログラムから制御することができます。  
以降、「BLE module」と記載します。
- LoRaWAN stack sample application board は、LoRaWAN stack と Sample application が書き込まれた Pmod インタフェースの評価ボードです。Host MCU プログラムから AT コマンドで制御することができます。  
以降、「LoRaWAN module」と記載します。
- TE Connectivity PMOD\_MS8607 は、気温、気圧、湿度を測定できる Pmod インタフェースの評価ボードです。  
以降、「Sensor module」と記載します。

## 2. 開発環境

### 2.1 ハードウェア環境

#### — ホストマシン

- PC/AT™互換機
- プロセッサ : 1GHz 以上 (ハイパースレッディング, マルチコア CPU に対応)
- メイン・メモリ : 推奨 2G バイト以上
- ディスプレイ : 1024×768 以上の解像度, 65536 色以上
- インタフェース : USB 2.0 (RL78/G14 Fast Prototyping Board との接続)

#### — 開発ボード

- RL78/G14 Fast Prototyping Board (RTK5RLG140C00000BJ)
- RL78/G1D BLE Module Expansion Board (RTKYRLG1D0B00000BJ)
- LoRaWAN stack sample application board
- TE Connectivity PMOD\_MS8607 (DPP901Z000) (気温、気圧、湿度センサ)

#### — スマートフォン(タブレット)

- Android デバイス、または iOS デバイス

#### — LoRaWAN 通信機器

- Gateway: kerlink Wirnet™ iFemtoCell 923
- Network Server: LORIoT LoRaWAN Network Server  
(ホストマシンのウェブブラウザから利用します)

### 2.2 ソフトウェア環境

#### — ホストマシン OS

- Windows 7 以降

#### — ウェブブラウザ

下記のいずれかのウェブブラウザを使用してください。

- Microsoft Edge
- Google Chrome

#### — 統合環境/コンパイラ

下記のいずれかの統合環境とコンパイラの組み合わせを使用してください。

- CS+ for CC V8.02.00 / CC-RL V1.08.00
- e<sup>2</sup> studio V7.6.0 / CC-RL V1.08.00

### 3. アプリケーション構成

#### 3.1 システム構成

本アプリケーションノートで使用するシステム構成図を「図 3-1 システム構成」に示します。

FPB の Pmod I/F に、BLE 通信をする RL78/G1D BLE Module Expansion Board と LoRaWAN 通信をする LoRaWAN stack sample application board を接続します。MCU ヘッダ J1 の IICA0 ピンには、TE Connectivity PMOD\_MS8607 を接続します。

FPB のアプリケーションは主に以下の役割を持つプログラムで構成されています。

- BLE module を制御する BLE プロトコルスタック Modem 構成<sup>注1</sup>の Host MCU プログラム
- LoRaWAN module を AT コマンドで制御するプログラム
- Sensor module を制御して測定したデータを読み出すプログラム

FPB アプリケーションは BLE module を制御してスマートフォンと通信し、LoRaWAN module の設定パラメータを受信します。FPB アプリケーションは受信したパラメータを AT コマンドで LoRaWAN module に設定し、Sensor module で測定したデータを LoRaWAN module から LoRaWAN Gateway へ送信します。Sensor module のセンサデータは、LoRaWAN Gateway から LoRaWAN Network Server へ送られたことを PC のウェブブラウザで確認することができます。

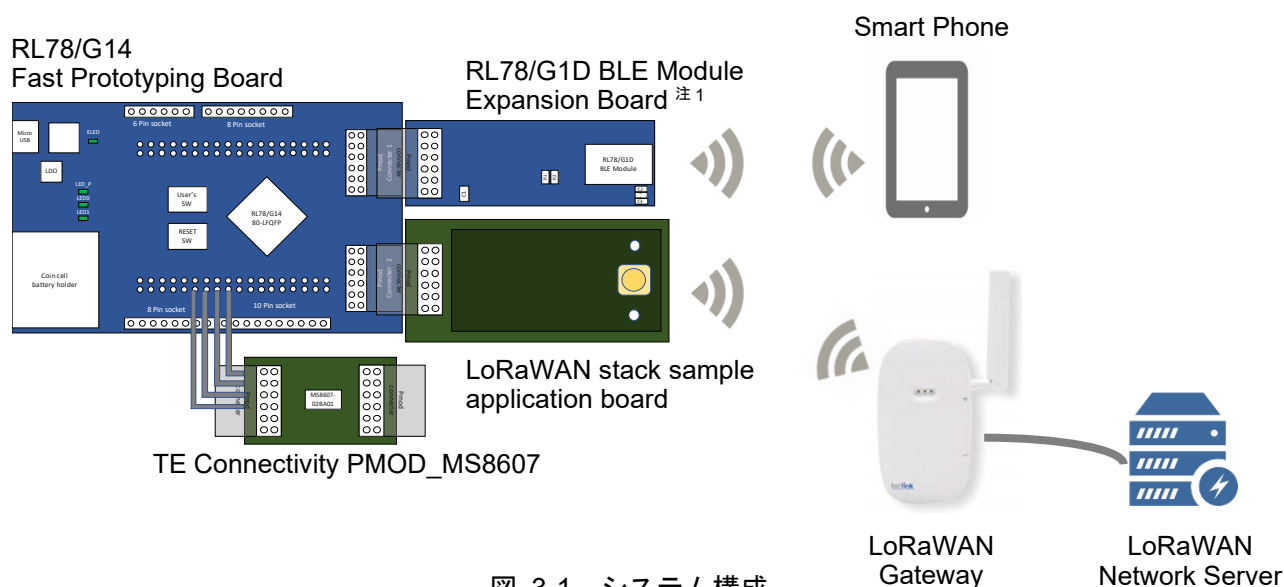


図 3-1 システム構成

【注】 1. FPB と BLE module の BLE プロトコルスタックにおけるソフトウェア構成として Modem 構成を用いており、使用するシリアル通信方式は UART 2 線分岐接続方式です。Modem 構成については「3.2.1 BLE module 構成」、UART 2 線分岐接続方式については「6.5 BLE Modem 構成 UART 2 線分岐接続方式」を参照してください。

## 3.1.1 FPB 接続端子

「図 3-1 システム構成」の Pmod インタフェース・コネクタピンアサインと MCU ヘッド J1 のピンアサインを示します。各ピンで使用している機能を青字で示します。

## (1) PMOD1

表 3-1 PMOD1 コネクタピンアサイン

RL78/G14 Fast Prototyping Board		Pmod		RL78/G1D BLE Module Expansion Board	
機能名	ピン番号	ピン番号	ピン番号	機能名	
P74 <sup>#1</sup> /KR4/INTP8	33	1	2	P30/INTP3/RTC1HZ	
P51/INTP2/SO00/TxD0/TOOLTxD/TRGIOB	42	2	8	P11/SI00/RxD0/TOOLRxD/SDA00/(TI06)/(TO06)	
P50/INTP1/SI00/RxD0/TOOLRxD/SDA00/TRGIOA/(TRJO0)	41	3	7	P12/SO00/TxD0/TOOLTxD/(TI05)/(TO05)	
P30/INTP3/RTC1HZ/SCK00/SCL00/TRJO0	40	4	9	P10/SCK00/SCL00/(TI07)/(TO07)	
GND	-	5	-	GND	
VCC	-	6	-	V <sub>DD</sub>	
P140/PCLBUZ0/INTP6	2	7	-	N.C	
P130	72	8	24	RESET#	
P147/ANI18/VCOU1	58	9	-	N.C	
P146	57	10	23	P40/TOOL0	
GND	-	11	-	GND	
VCC	-	12	-	V <sub>DD</sub>	

【注】 1. 2つのボードは Pmod インタフェースで接続されているため UART 2 線分岐接続方式が必要となる FPB の TxD ラインを分岐して BLE module の INTP3 へ入力することができません。本アプリケーションでは、FPB の P74 から BLE module の INTP3 へ Low レベルを入力することで疑似的に UART 2 線分岐接続方式の通信を可能にしています。

## (2) PMOD2

表 3-2 PMOD2 コネクタピンアサイン

RL78/G14 Fast Prototyping Board		Pmod		LoRaWAN stack sample application board	
機能名	ピン番号	ピン番号	ピン番号	機能名	
P16/TI01/TO01/INTP5/TRDI0C0/IVREF0/(SI00)/(RxD0)	48	1	NC	NC	
P13/TxD2/SO20/TRDIOA1/IVCMP1	51	2	RxD	RxD	
P14/RxD2/SI20/SDA20/TRDIOD0/(SCLA0)	50	3	TxD	TxD	
P15/SCK20/SCL20/TRDIOB0/(SDAA0)	49	4	NC	NC	
GND	-	5	GND	GND	
VCC	-	6	V <sub>DD</sub>	V <sub>DD</sub>	
P141/PCLBUZ1/INTP7	1	7	NC	NC	
P110/(INTP11)	55	8	RESET#	RESET#	
P17/TI02/TO02/TRDIOA0/TRDCLK/IVCMP0/(SO00)/(TxD0)	47	9	NC	NC	
P111	56	10	NC	NC	
GND	-	11	GND	GND	
VCC	-	12	V <sub>DD</sub>	V <sub>DD</sub>	



## (3) MCU ヘッダ J1

表 3-3 MCU ヘッダ J1 ピンアサイン

RL78/G14 Fast Prototyping Board		TE Connectivity PMOD_MS8607	
機能名	J1 ピン番号	Pmod ピン番号	機能名
-	-	1	NC
-	-	2	NC
P60/SCLA0	21	3	SCL
P61/SDAA0	22	4	SDA
EVSS0	17	5	GND
EVDD0	19	6	VDD
-	-	7	NC
-	-	8	NC
-	-	9	SCL
-	-	10	SDA
-	-	11	GND
-	-	12	VDD

## 3.2 ソフトウェア構成

### 3.2.1 BLE module 構成

Host MCU である RL78/G14 と、BLE MCU である RY7011 のソフトウェア構成図を示します。RY7011 には出荷時に動作確認用ファームウェアが書き込まれており複数のプロファイルに対応しています。本アプリケーションでは汎用双方向通信(General Purpose Communication Profile : GPCP)を使用します。その他のプロファイルについては、「RL78/G1D モジュール ファームウェア ユーザーズマニュアル」(R01UW0160)の「7. プロファイル」を参照してください。

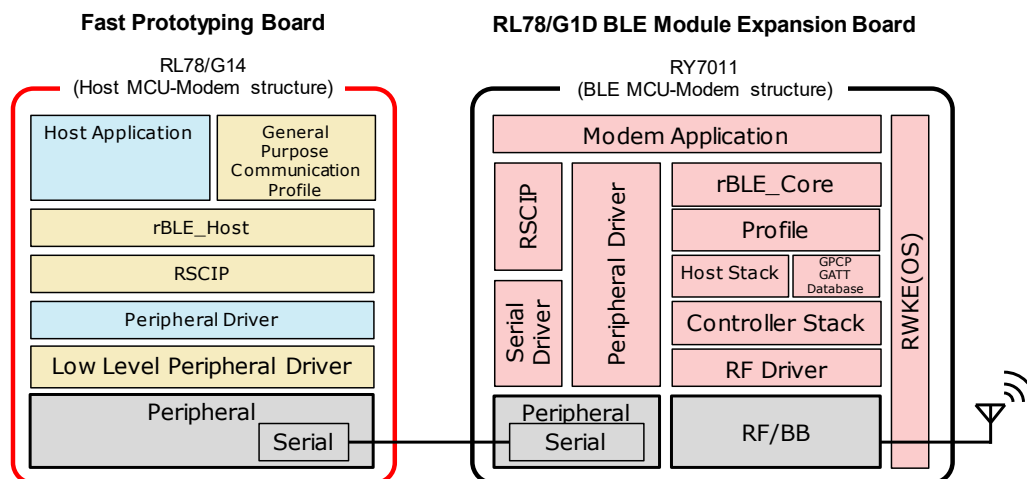


図 3-2 BLE module 通信ソフトウェア構成

Host MCU は、MCU 周辺機能の制御と BLE MCU との通信を実行するための低レベル周辺ドライバ、周辺ドライバ、RSCIP (Renesas Serial Communication Interface Protocol) と、rBLE API をアプリケーションに提供するための rBLE\_Host と、システムを制御するための Host Application、GATT API を使用した General Purpose Communication Profile(GPCP)で構成されます。

低レベル周辺ドライバは、コード生成ツールが自動生成します。RSCIP、rBLE\_Host は BLE プロトコルスタックに含まれており、ソースコードが提供されます。ソフトウェア開発時は、BLE プロトコルスタックが提供する最新のソースコードをご使用ください。

表 3-4 Host MCU ソフトウェア構成

ソフトウェア	機能	ソフトウェア開発
Host Application	rBLE の初期化 rBLE コマンドの実行スケジューリング rBLE イベントコールバックの登録	コーディングが <b>必要</b>
General Purpose Communication Profile (GPCP)	GATT API を使用した独自プロファイル	コーディングが不要 (ソースコード提供) 注1
rBLE_Host	rBLE API 提供 イベントコールバックの実行	コーディングが不要 (ソースコード提供) 注1
RSCIP	シリアル通信プロトコルの制御	コーディングが不要 (ソースコード提供) 注1
Peripheral Driver	Host MCU 周辺機能の制御	コーディングが <b>必要</b>
Low Level Peripheral Driver	Host MCU 周辺機能のプリミティブな制御	コーディングが不要 (ツール自動生成) 注2

【注】 1. ソフトウェア開発用コードファイルは BLE プロトコルスタックが提供。  
2. ソフトウェア開発用コードファイルはコード生成ツールが自動生成。

BLE MCU は、RF/BB を制御するための RF ドライバ、Sample Custom Profile の GATT Database、Host/Controller スタック、Profile、rBLE\_Core と、Host MCU と通信するためのシリアル通信ドライバ、RSCIP と、システムを制御するための RWKE (Renesas Wireless Kernel Extension)、Modem アプリケーションで構成されます。これらは「RL78/G1D モジュール モジュール制御ソフトウェア」(R01AN3362)でビルド環境が提供されます。

表 3-5 BLE MCU ソフトウェア構成

ソフトウェア	機能
Modem Application	RSCIP と rBLE の制御
RWKE	システム全体のスケジューリングとメモリ資源の管理
RSCIP	シリアル通信プロトコルの制御
Peripheral Driver/Serial Driver	BLE MCU 周辺機能の制御
rBLE_Core	rBLE_API 提供
Profile	プロフィール機能の提供
Host Stack	GAP、GATT、SM、L2CAP 機能の提供
GPCP GATT Database	General Purpose Communication Profile の GATT Database
Controller Stack	LL 機能の提供

### 3.2.2 LoRaWAN module 構成

FPB の RL78/G14 と、LoRaWAN module のソフトウェア構成図を示します。LoRaWAN module は LoRaWAN stack<sup>注1</sup> と LoRaWAN stack を AT コマンドで制御する sample application<sup>注2</sup> 等で構成されます。

FPB は LoRaWAN module を AT コマンドで制御するための LoRaWAN module application と Sensor module を制御して測定したデータを読み出す Sensor control application で構成されます。

- 【注】
1. 「LoRaWAN stack reference guide」(R11AN0228)を参照してください。
  2. 「LoRaWAN Stack Sample Application Command Reference」(R11AN0231)を参照してください。

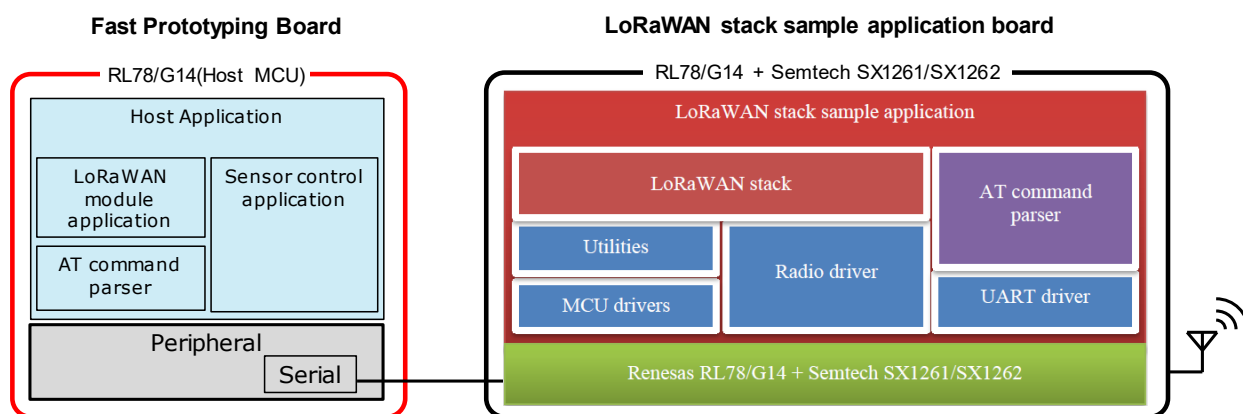


図 3-3 LoRaWAN module 通信ソフトウェア構成

### 3.3 周辺機能構成

FPB のアプリケーションで使用する RL78/G14 の周辺機能を以下に示します。

アプリケーションが BLE module または LoRaWAN module を使用するために必ず必要とする周辺機能を「必須」、その他を「任意」としています。

表 3-6 周辺機能

周辺機能	用途		必要性
クロック発生回路	RL78/G14 の動作周波数設定に使用します		必須
ポート	P74	BLE module との UART 通信で、UART 2 線分岐式接続方式の WAKEUP 端子として使用します。	必須
	P130	BLE module のリセット解除端子として使用します。	必須
	P110	LoRaWAN module のリセット解除端子として使用します。	必須
	P43	FPB の LED0 で使用します。	任意
	P44	FPB の LED1 で使用します。	任意
	P137	FPB の SW_USR で使用します。	任意
シリアル-UART0	BLE module との UART 通信に使用します。		必須
シリアル-UART2	LoRaWAN module との UART 通信に使用します。		必須
シリアル-IICA0	Sensor module との IICA 通信に使用します。		任意
タイマ-TAU0	Sensor module の A/D 変換時間待ちで使用します。		任意
12 ビット・インターバル・タイマ	BLE Host MCU プログラムの rBLE_Host や RSCIP で使用するタイマ機能に使用します。		必須

### 3.4 ファイル構成

本アプリケーションのファイル構成を示します。

ファイル構成の(R)表記は、BLE プロトコルスタックに含まれているファイルであることを示します。ソフトウェア開発時は、BLE プロトコルスタックが提供する最新のコードファイルをご使用ください。

RL78G14_FP_BLE_LoRaWAN_Application	
├──ROM_File	実行ファイル (HEX ファイル)
├──HostSample	
│   ├──Platform	
│   │   ├──driver	
│   │   │   ├──dataflash	
│   │   │   │   ├──dataflash.c	Data Flash ドライバ・コードファイル
│   │   │   │   ├──dataflash.h	Data Flash ドライバ・コードファイル
│   │   │   │   ├──eel_descriptor.c	EEPROM Emulation Library コードファイル
│   │   │   │   ├──eel_descriptor.h	EEPROM Emulation Library ヘッダファイル
│   │   │   │   ├──fdl_descriptor.c	Data Flash Access Library コードファイル
│   │   │   │   ├──fdl_descriptor.h	Data Flash Access Library ヘッダファイル
│   │   │   └──cc_rl	
│   │   │       ├──eel.h	EEPROM Emulation Library ヘッダファイル
│   │   │       ├──eel.lib	EEPROM Emulation Library ファイル
│   │   │       ├──eel_types.h	EEPROM Emulation Library ヘッダファイル
│   │   │       ├──fdl.h	Data Flash Access Library ヘッダファイル
│   │   │       ├──fdl.lib	Data Flash Access Library ファイル
│   │   │       └──fdl_types.h	Data Flash Access Library ヘッダファイル
│   │   └──ms8607	
│   │       ├──r_ms8607.c	MS8607 Sensor ドライバ・コードファイル
│   │       └──r_ms8607.h	MS8607 Sensor ドライバ・ヘッダファイル
│   └──serial	
│       ├──r_uart_ble_modem.c	BLE UART ドライバ・コードファイル
│       ├──r_uart_ble_modem.h	BLE UART ドライバ・コードファイル
│       ├──r_uart_lora.c	LoRaWAN UART ドライバ・コードファイル
│       └──r_uart_lora.h	LoRaWAN UART ドライバ・コードファイル
│   └──timer	
│       ├──timer.c	タイマドライバ・コードファイル
│       └──timer.h	タイマドライバ・ヘッダファイル
└──include	
├──arch.h	(R) アーキテクチャ・ヘッダファイル
├──compiler.h	(R) コンパイラ・ヘッダファイル
├──ll.h	(R) 低レベルマクロ・ヘッダファイル
├──rscip_api.h	(R) RSCIP コールバック・ヘッダファイル
└──types.h	(R) タイプ定義・ヘッダファイル
└──rBLE	
├──host	
├──rble_host.c	(R) rBLE_Host コードファイル
└──rble_if_api_cb.c	(R) rBLE API コールバック・コードファイル
└──gap	
├──rble_api_gap.c	(R) GAP API コードファイル
└──gatt	
├──rble_api_gatt.c	(R) GATT API コードファイル
└──sm	
├──rble_api_sm.c	(R) SM API コードファイル

└─vs	rble_api_vs.c	(R) VS API コードファイル
└─include	db_handle.h	(R) データベースハンドル・ヘッダファイル
	prf_sel.h	(R) プロファイル選択・ヘッダファイル
	rble.h	(R) rBLE マクロ定義・ヘッダファイル
	rble_api.h	(R) rBLE API・ヘッダファイル
	rble_app.h	(R) rBLE SCP API・ヘッダファイル
	rble_trans.h	(R) rBLE 通信・ヘッダファイル
└─host	rble_host.h	(R) rBLE_Host ヘッダファイル
└─rscip	rscip.c	(R) RSCIP コードファイル
	rscip.h	(R) RSCIP ヘッダファイル
	rscip_cntl.c	(R) RSCIP コントロール・コードファイル
	rscip_cntl.h	(R) RSCIP コントロール・ヘッダファイル
	rscip_ext.h	(R) RSCIP 外部コールバック・ヘッダファイル
	rscip_uart.c	(R) RSCIP シリアル制御・コードファイル
	rscip_uart.h	(R) RSCIP シリアル制御・ヘッダファイル
└─sample_app	r_app_ble.c	
	r_app_lora.c	
	r_app_lora.h	
	r_app_ms8607.c	MS8607 Sensor アプリケーション・コードファイル
	r_app_ms8607.h	MS8607 Sensor アプリケーション・ヘッダファイル
└─sample_profile	└─vuart	
	vuart.h	汎用双方向通信共通ヘッダファイル
	vuarts.c	汎用双方向通信共通 Server ファイル
	vuarts.h	汎用双方向通信共通 Server ヘッダファイル
└─project	└─cs_cc	
	cstart.asm	スタートアップ・ルーチン
	iodefine.h	I/O ヘッダファイル
	RL78G14_FPB_BLE_LoRaWAN_Application.mtpj	CS+ for CC プロジェクトファイル
	RL78G14_FPB_BLE_LoRaWAN_Application.rcpe	CS+ for CC プロジェクトファイル
	stkinit.asm	スタック領域初期化ルーチン
	└─src	
	r_cg_cg.c	クロック生成ドライバ・コードファイル
	r_cg_cg.h	クロック生成ドライバ・ヘッダファイル
	r_cg_cg_user.c	クロック生成ドライバ・ユーザコードファイル
	r_cg_intc.c	割り込みドライバ・コードファイル
	r_cg_intc.h	割り込みドライバ・ヘッダファイル
	r_cg_intc_user.c	割り込みドライバ・ユーザコードファイル
	r_cg_it.c	12 ビットインターバルタイマドライバ・コードファイル
	r_cg_it.h	12 ビットインターバルタイマドライバ・ヘッダファイル
	r_cg_it_user.c	12 ビットインターバルタイマドライバ・ユーザコードファイル
	r_cg_macrodriver.h	汎用マクロ・ヘッダファイル
	r_cg_macrodriver_hostsample.h	汎用マクロ・ヘッダファイル(バックアップ)
	r_cg_port.c	ポートドライバ・コードファイル
	r_cg_port.h	ポートドライバ・ヘッダファイル
	r_cg_port_user.c	ポートドライバ・ユーザコードファイル
	r_cg_serial.c	シリアルドライバ・コードファイル
	r_cg_serial.h	シリアルドライバ・ヘッダファイル
	r_cg_serial_user.c	シリアルドライバ・ユーザコードファイル
	r_cg_timer.c	タイマアレイユニットドライバ・コードファイル

	r_cg_timer.h	タイマアレイユニットドライバ・ヘッダファイル
	r_cg_timer_user.c	タイマアレイユニットドライバ・ユーザコードファイル
	r_cg_userdefine.h	ユーザ定義マクロ・ヘッダファイル
	r_main.c	メインループ・コードファイル
	r_systeminit.c	周辺機能初期化・コードファイル
	└─e2studio	
	.cproject	e <sup>2</sup> studio プロジェクトファイル
	.project	e <sup>2</sup> studio プロジェクトファイル
	RL78G14_FPB_BLE_LoRaWAN_Application	e <sup>2</sup> studio プロジェクトファイル
	HardwareDebug.launch	e <sup>2</sup> studio プロジェクトファイル
	r_option_cc.txt	e <sup>2</sup> studio プロジェクトファイル
	└─.settings	
	com.renesas.tools.misrac.prefs	e <sup>2</sup> studio プロジェクトファイル
	CoverageSetting.xml	e <sup>2</sup> studio プロジェクトファイル
	Dependency_Scan_Preferences.prefs	e <sup>2</sup> studio プロジェクトファイル
	e2studio_project.prefs	e <sup>2</sup> studio プロジェクトファイル
	renesasPGModel.xml	e <sup>2</sup> studio プロジェクトファイル
	└─CodeGenerator	
	cgproject.cgp	e <sup>2</sup> studio プロジェクトファイル
	cgprojectDatas.datas	e <sup>2</sup> studio プロジェクトファイル
	└─generate	
	cstart.asm	スタートアップ・ルーチン
	iodef.h	I/O ヘッダファイル
	stkinit.asm	スタック領域初期化ルーチン
	└─src	
	r_cg_cgc.c	クロック生成ドライバ・コードファイル
	r_cg_cgc.h	クロック生成ドライバ・ヘッダファイル
	r_cg_cgc_user.c	クロック生成ドライバ・ユーザコードファイル
	r_cg_intc.c	割り込みドライバ・コードファイル
	r_cg_intc.h	割り込みドライバ・ヘッダファイル
	r_cg_intc_user.c	割り込みドライバ・ユーザコードファイル
	r_cg_it.c	12ビットインターバルタイマドライバ・コードファイル
	r_cg_it.h	12ビットインターバルタイマドライバ・ヘッダファイル
	r_cg_it_user.c	12ビットインターバルタイマドライバ・ユーザコードファイル
	r_cg_macrodriver.c	汎用マクロ・ヘッダファイル
	r_cg_macrodriver_hostsample.h	汎用マクロ・ヘッダファイル(バックアップ)
	r_cg_port.c	ポートドライバ・コードファイル
	r_cg_port.h	ポートドライバ・ヘッダファイル
	r_cg_port_user.c	ポートドライバ・ユーザコードファイル
	r_cg_serial.c	シリアルドライバ・コードファイル
	r_cg_serial.h	シリアルドライバ・ヘッダファイル
	r_cg_serial_user.c	シリアルドライバ・ユーザコードファイル
	r_cg_timer.c	タイマアレイユニットドライバ・コードファイル
	r_cg_timer.h	タイマアレイユニットドライバ・ヘッダファイル
	r_cg_timer_user.c	タイマアレイユニットドライバ・ユーザコードファイル
	r_cg_userdefine.h	ユーザ定義マクロ・ヘッダファイル
	r_main.c	メインループ・コードファイル
	r_systeminit.c	周辺機能初期化・コードファイル



## 4. ビルド手順

本アプリケーションをビルドする手順を示します。

表 4-1 FPB BLE・LoRaWAN アプリケーション プロジェクト

CS+ for CC	
Project File	RL78G14_FP_BLE_LoRaWAN_Application¥project¥cs_cc¥ RL78G14_FP_BLE_LoRaWAN_Applicaiotn.mtpj
HEX File	RL78G14_FP_BLE_LoRaWAN_Application¥project¥cs_cc¥DefaultBuild¥ RL78G14_FP_BLE_LoRaWAN_Applicaiotn.hex
e <sup>2</sup> studio	
Project Folder	RL78G14_FP_BLE_LoRaWAN_Application¥project¥e2studio
HEX File	RL78G14_FP_BLE_LoRaWAN_Application¥project¥e2studio¥HardwareDebug¥ RL78G14_FP_BLE_LoRaWAN_Application.hex

### 4.1 CS+ for CC

- 「表 4-1 FPB BLE・LoRaWAN アプリケーション プロジェクト」の Project File をダブルクリックし、CS+を起動します。
- CS+の「プロジェクトツリー」パネル内で「RL78G14\_FP\_BLE\_LoRaWAN\_Application (プロジェクト)」を右クリックし、メニューから「RL78G14\_FP\_BLE\_LoRaWAN\_Application をビルド」を選択してビルドします。
- 「表 4-1 FPB BLE・LoRaWAN アプリケーション プロジェクト」の HEX File の欄で示すパスに HEX ファイルが生成されます。

### 4.2 e2 studio

- e<sup>2</sup> studio を起動します。
- 「プロジェクトエクスプローラー」ビュー上で右クリックし、メニューから「インポート」を選択します。
- 表示された「インポート」ダイアログで、「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」ボタンをクリックします。
- 「プロジェクトのインポート」ダイアログで、「ルートディレクトリの選択」フォームに、「表 4-1 FPB BLE・LoRaWAN アプリケーション プロジェクト」の Project Folder に示すパスを指定します。「プロジェクト」フォームに指定したプロジェクトが表示されていることを確認し、「終了」ボタンをクリックします。「インポート」ダイアログが閉じられ、「プロジェクトエクスプローラー」ビューにプロジェクトが追加されます。
- 「プロジェクトエクスプローラー」ビューに表示されたプロジェクトを右クリックし、「プロジェクトのビルド」を選択してビルドします。
- 「表 4-1 FPB BLE・LoRaWAN アプリケーション プロジェクト」の HEX File に示すパスに HEX ファイルが生成されます。

## 5. アプリケーションの実行

「図 5-1 アプリケーション実行環境」で示す構成で BLE 通信や LoRaWAN 通信の動作確認を行いません。

### 5.1 GATTBrowser のインストール

スマートフォン(タブレット)に BLE の動作確認を行なうことができる GATTBrowser をインストールしてください。

- Android 版 GATTBrowser

<https://play.google.com/store/apps/details?id=com.renesas.ble.gattbrowser>

- iOS 版 GATTBrowser

<https://itunes.apple.com/jp/app/gattbrowser/id1163057977?mt=8>

### 5.2 LORIoT LoRaWAN Network Server の準備

「LoRaWAN® IoT Demo LPWA IoT Solution with Cloud」(R11AN0412)の「3. LoRaWAN Network Server」を参照してセットアップしてください。

### 5.3 実行環境

「4.ビルド手順」に従ってプログラムをビルドしてください。そして FPB と PC を USB ケーブルで接続しプログラムをダウンロードします。

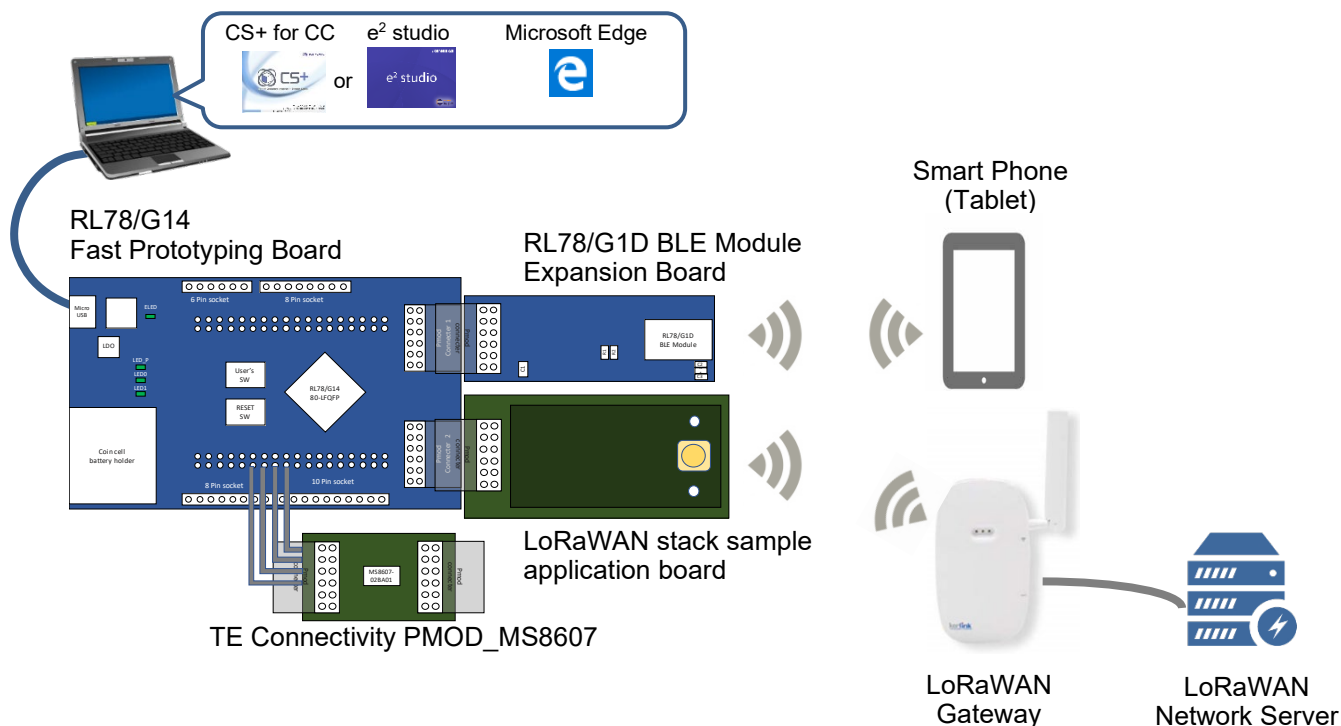


図 5-1 アプリケーション実行環境

5.4 実行手順

アプリケーションの全体動作を「図 5-2 アプリケーション全体動作」に示します。

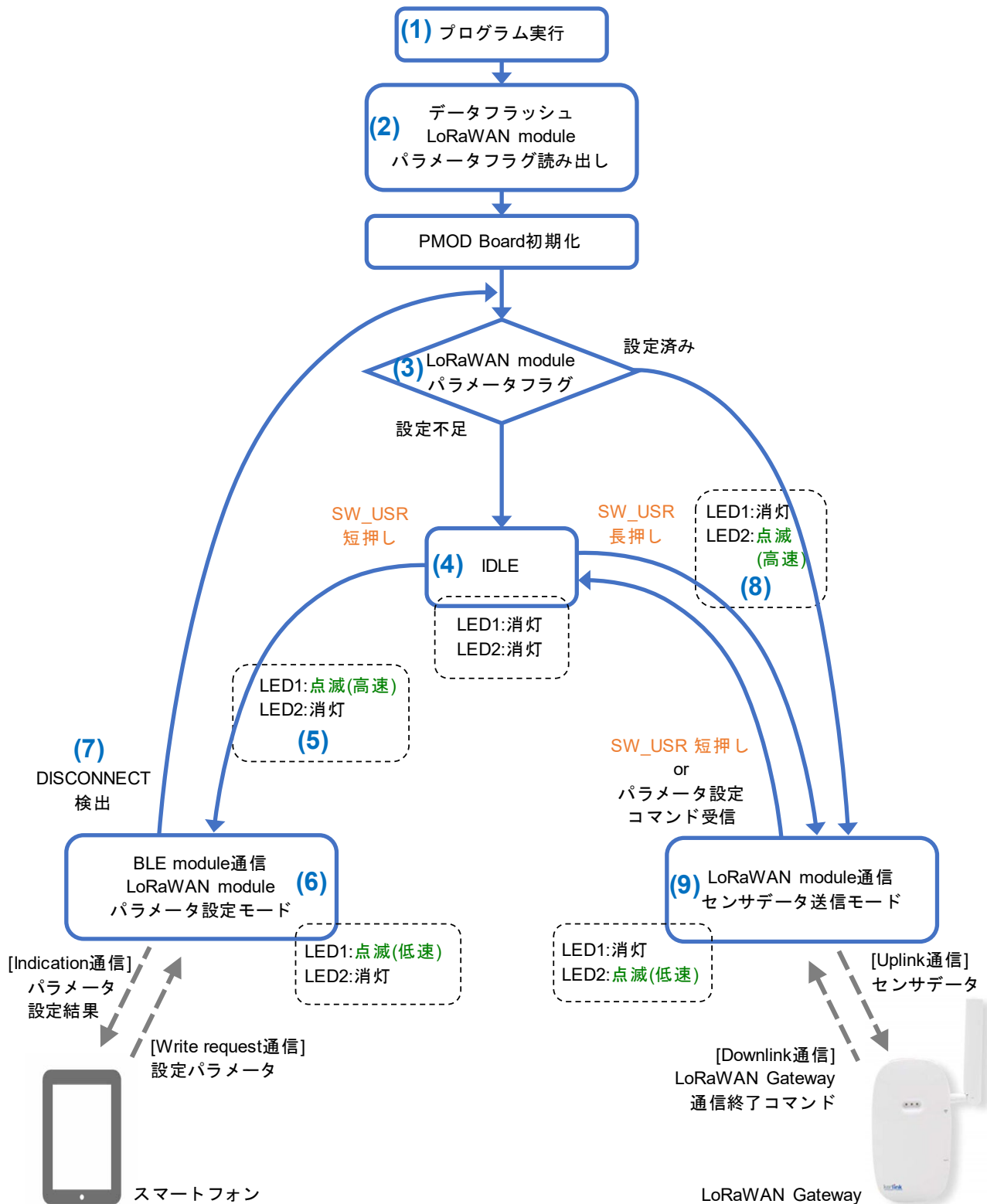


図 5-2 アプリケーション全体動作

**[プログラム実行開始]**

- (1) プログラムを実行します。
- (2) FPB のデータフラッシュから LoRaWAN module パラメータフラグを読み出します。
- (3) LoRaWAN module パラメータフラグが設定されているか判定します。

- LoRaWAN module 通信に必要なパラメータがすべて設定されている  
➡[LoRaWAN module 通信]へ

- LoRaWAN module 通信に必要なパラメータが設定されていない  
➡[IDLE]へ

**[IDLE]**

- (4) スイッチ(SW\_USR)の押下を待つ IDLE の状態です。  
(LED1:消灯、LED2:消灯)

- SW\_USR を短押し  
(LoRaWAN module に通信に必要なパラメータの設定が不足している場合)  
➡[BLE 通信 LoRaWAN module パラメータ設定モー]へ

- SW\_USR を長押し  
(LoRaWAN module 通信に必要なパラメータがすべて設定されている場合)  
➡[LoRaWAN module 通信]へ

**[BLE 通信 LoRaWAN パラメータ設定]**

- (5) BLE module はアドバタイジングを開始します。  
(LED1:点滅(高速)、LED2:消灯)
- (6) スマートフォンで BLE module に接続して、スマートフォンから LoRaWAN module パラメータを設定します。  
(LED1:点滅(低速)、LED2:消灯)

スマートフォンの操作は、Android の場合「5.4.1 Android デバイスでの LoRaWAN module パラメータ設定」、iOS の場合「5.4.2 iOS デバイスでの LoRaWAN module パラメータ設定」を参照してください。

- (7) 全ての LoRaWAN module パラメータを設定したあと、スマートフォンからディスコネクトを実行します。

BLE module は切断を検出すると LoRaWAN module に設定したパラメータフラグを FPB のデータフラッシュへ保存します。スマートフォンから切断を実行せずに、電源を落とす、または E2Lite を切断した場合は、パラメータフラグは FPB のデータフラッシュに保存されないので再度設定してください。

- LoRaWAN module 通信に必要なパラメータがすべて設定されている  
➡[LoRaWAN module 通信]へ

- LoRaWAN module 通信に必要なパラメータが設定されていない  
➡[IDLE]へ

**[LoRaWAN module 通信]**

- (8) LoRaWAN module 通信を開始します。まず LoRaWAN Gateway へ Activation します。  
(LED1:消灯、LED2:点滅(高速))
- (9) LoRaWAN Gateway から Accept されると、測定したデータを Sensor module から 1 分間隔で読み出して LoRaWAN Gateway へ送信します。  
(LED1:消灯、LED2:点滅(低速))

ウェブブラウザで LORIoT Network Server のサイトを参照して、LoRaWAN module から LoRaWAN Network Server へデータが送られていることを確認します。「5.4.3.1 センサデータの確認」を参照。

LoRaWAN module 通信を中断し IDLE へ戻る方法は 2 つあります。

- LoRaWAN module が LoRaWAN Network Server から送信した通信終了コマンドを受信  
「5.4.3.2 LoRaWAN Gateway 通信終了コマンドの送信」を参照。  
⇒[IDLE]へ
- SW\_USR を押下  
⇒[IDLE]へ

## 5.4.1 Android デバイスでの LoRaWAN module パラメータ設定

1. Android デバイスにインストールした GATTBrowser を起動します。
2. デバイスの検索結果から、RTK5RL140C と表示されたデバイスと接続します。  
(図 A1 の矢印(1))
3. 接続すると Service の一覧が表示されます。一番下までスクロールさせ Renesas Virtual UART Service の Indication Characteristic を選択します。(図 A2 の矢印(2))
4. [Indication Off]をタップし、[Indication On]にします。(図 A3 の矢印(3))
5. Service の一覧に戻ります。(図 A3 の矢印(4))

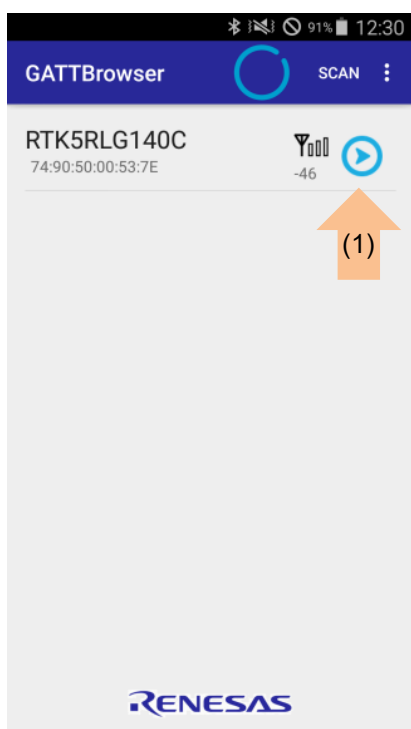


図 A1

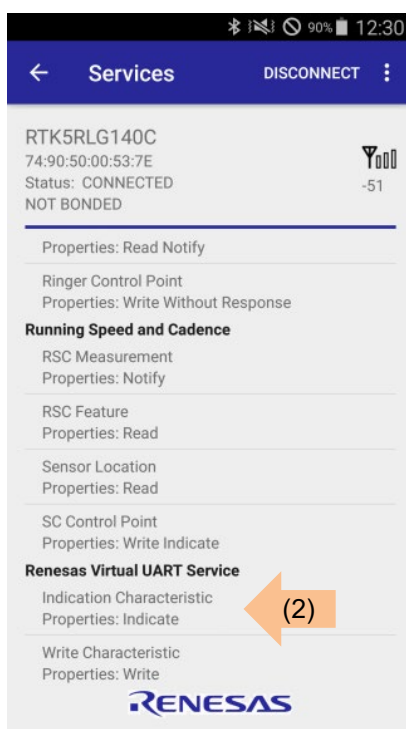


図 A2

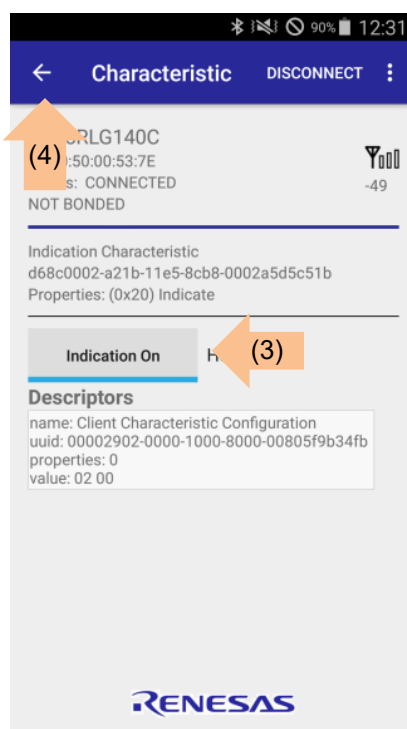


図 A3

- Service の一覧で、Write Characteristic を選択します。(図 A4 の矢印(5))
- Hex を選択します。(図 A5 の矢印(6))
- 設定する LoRaWAN module パラメータを入力します。パラメータは「6.4.1.1 AT コマンドフォーマット」を参照してください。(図 A5 の矢印(7))
- Write をタップして送信します。(図 A5 の矢印(8))
- Service の一覧に戻ります。(図 A5 の矢印(9))

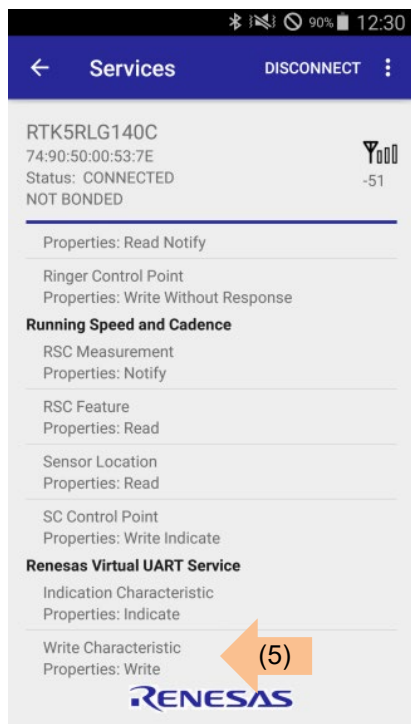


図 A4

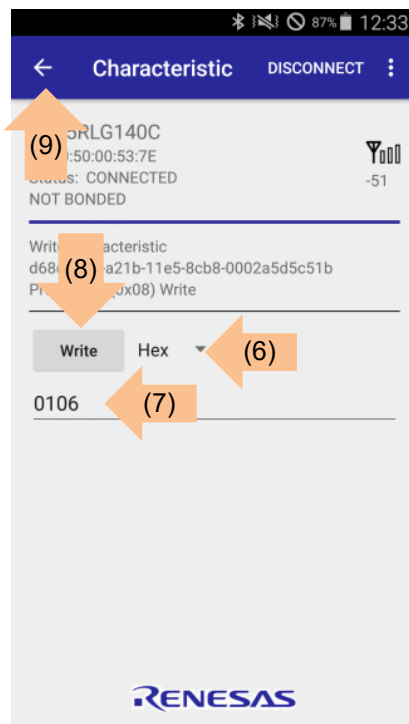


図 A5

11. Indication Characteristic を選択します。(図 A6 の矢印(10))
12. LoRaWAN module パラメータの設定結果が表示されます。(図 A7 の矢印(11))  
結果のフォーマットは「6.4.1.2 AT コマンド結果コードフォーマット」を参照してください。
13. 全てのパラメータの設定が完了した後、DISCONNECT を選択します。(図 A7 の矢印(12))  
設定完了の判断は、結果フォーマットの 3 バイト目(パラメータフラグ)に 0x7F が表示されていることを確認してください。

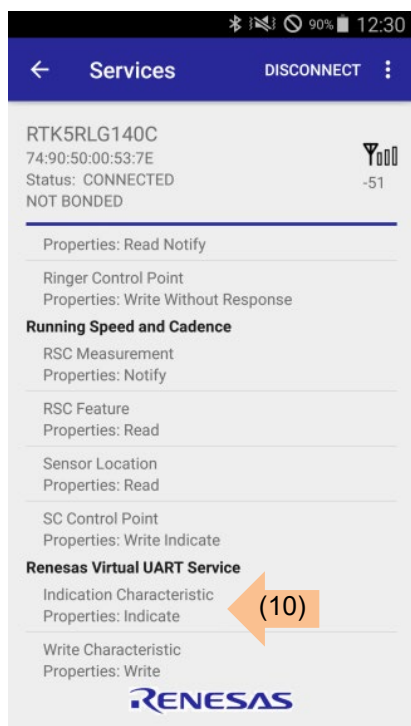


図 A6

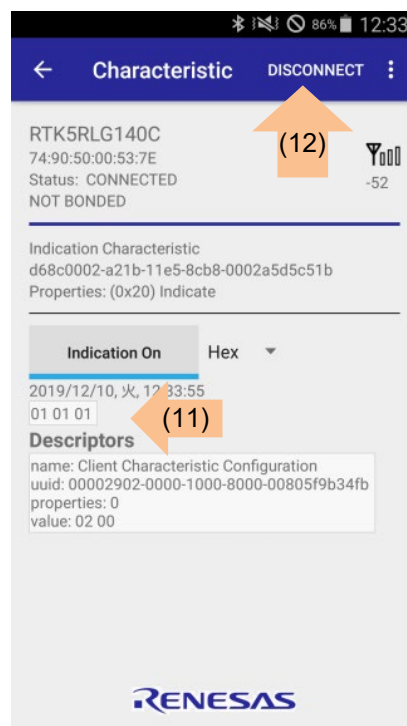


図 A7



## 5.4.2 iOS デバイスでの LoRaWAN module パラメータ設定

1. iOS デバイスにインストールした GATTBrowser を起動します。
2. デバイスの検索結果から、RTK5RL140C と表示されたデバイスと接続を開始します。  
(図 B1 の矢印(1))
3. 接続すると Service の一覧が表示されます。一番下までスクロールさせ Renesas Virtual UART Service の Indication Characteristic を選択します。(図 B2 の矢印(2))
4. [Enable Indication]をタップし、[Disable Indication]にします。(図 B3 の矢印(3))
5. Service の一覧に戻ります。(図 B3 の矢印(4))

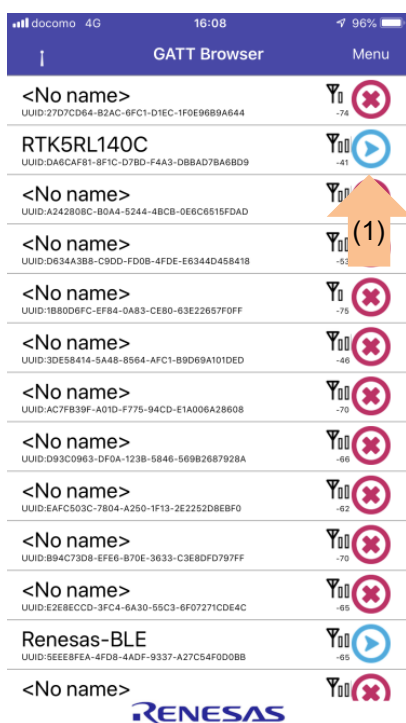


図 B1

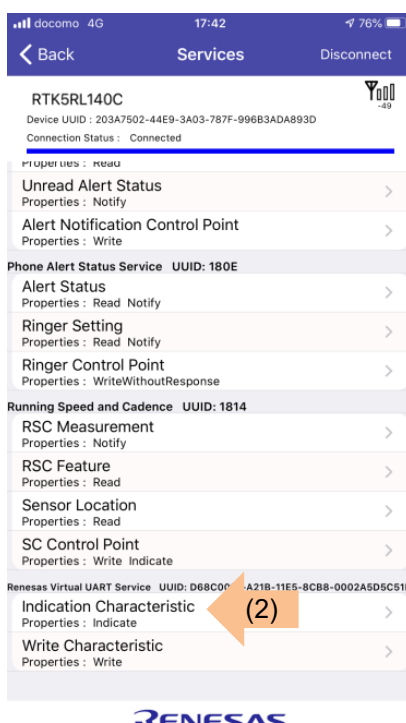


図 B2

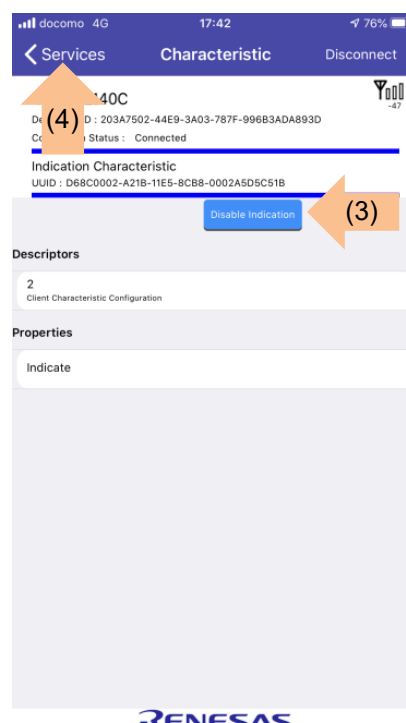


図 B3

- Service の一覧で、Write Characteristic を選択します。(図 B4 の矢印(5))
- Hex を選択します。(図 B5 の矢印(6))
- 設定する LoRaWAN module パラメータを入力します。パラメータは「6.4.1.1 AT コマンドフォーマット」を参照してください。(図 B5 の矢印(7))
- Write をタップして送信します。(図 B5 の矢印(8))
- Service の一覧に戻ります。(図 B5 の矢印(9))

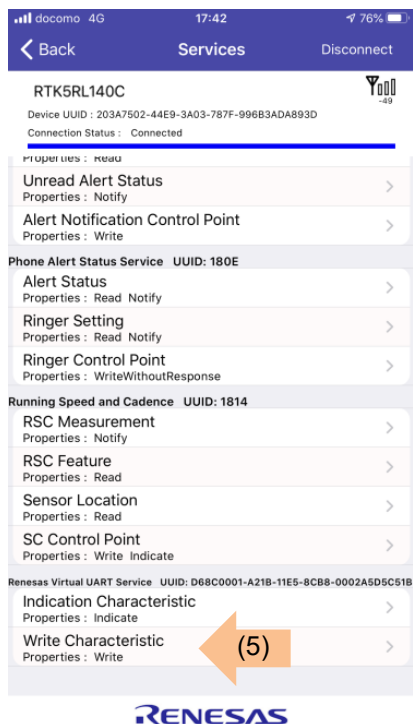


図 B4

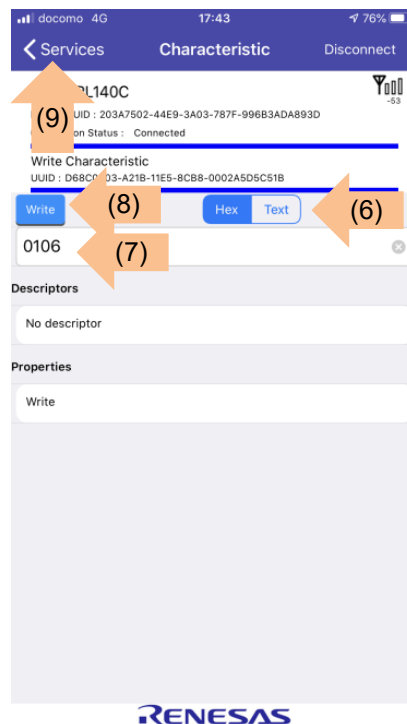


図 B5

11. Indication Characteristic を選択します。(図 B6 の矢印(10))
12. LoRaWAN module パラメータの設定結果が表示されます。(図 B7 の矢印(11))  
結果のフォーマットは「6.4.1.2 AT コマンド結果コードフォーマット」を参照してください。
13. 全てのパラメータの設定が完了した後、Disconnect を選択します。(図 A6 の矢印(12))  
設定完了の判断は、結果フォーマットの3バイト目(パラメータフラグ)に 0x7F が表示されていることを確認してください。

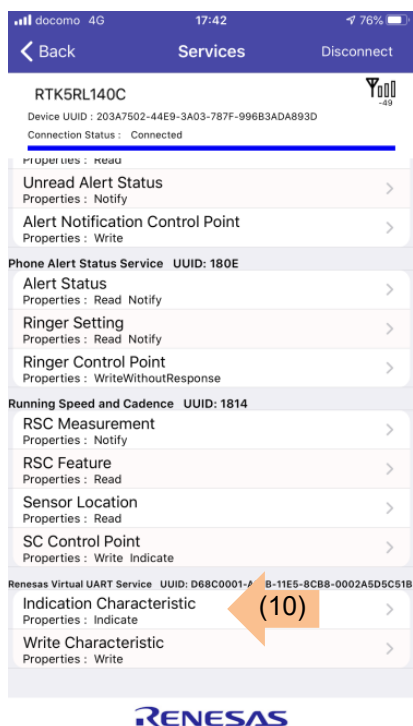


図 B6

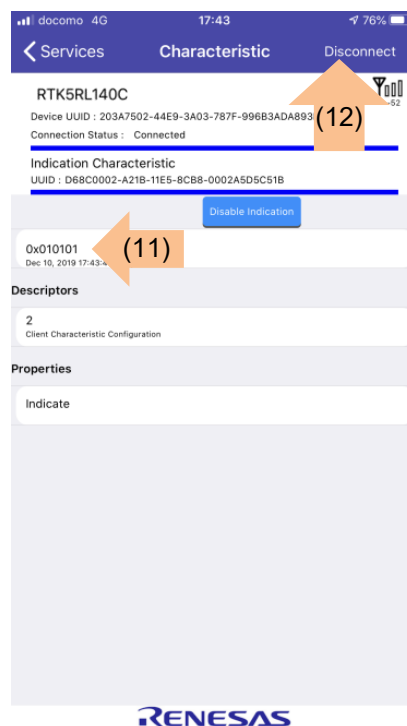


図 B7

## 5.4.3 LoRaWAN Network Server

## 5.4.3.1 センサデータの確認

LORIoT 社が運営する LoRaWAN Network Server で、LoRaWAN module から送信されたセンサデータを確認します。

0x749050FFFE000C2C の Device EUI を持つ LoRaWAN module から、センサデータ(Payload)を受信していることが確認できます。

Device EUI	Local time	Freq [MHz]	Data rate	RSSI (dBm)	SNR (dB)	FCntUp	Port	Payload
749050FFFE000C2C	2019-12-09 15:40:59.181	924.000	SF10 BW125 4/5	-33	14	11	10	44 7d d3 d7 42 2c e6 40
749050FFFE000C2C	2019-12-09 15:39:59.320	924.000	SF10 BW125 4/5	-31	11.2	10	10	41 c2 3d 71
749050FFFE000C2C	2019-12-09 15:38:59.545	923.400	SF10 BW125 4/5	-46	12.5	9	10	44 7d d0 00 42 2d a1 c0
749050FFFE000C2C	2019-12-09 15:37:59.681	924.200	SF10 BW125 4/5	-47	11	8	10	41 c2 3d 71
749050FFFE000C2C	2019-12-09 15:36:59.890	923.600	SF10 BW125 4/5	-31	12	7	10	44 7d cf 5c 42 2d 63 40
749050FFFE000C2C	2019-12-09 15:36:00.22	924.000	SF10 BW125 4/5	-35	11.8	6	10	41 c2 a3 d7

図 5-3 LoRaWAN Network Server でのセンサデータ確認

## 5.4.3.2 LoRaWAN Gateway 通信終了コマンドの送信

- (1) センサデータを確認するためページ上部の「Send Data To Device」ボタンを押して「Send Data To Device」ダイアログを表示します。

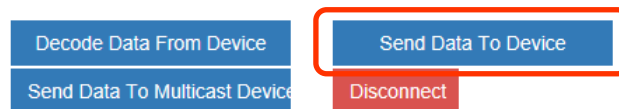


図 5-4 Send Data To Device ボタン

- (2) LoRaWAN module に設定したパラメータと通信終了コマンドを入力して「Send to device」ボタンを押します。

図 5-5 Send Data To Device ダイアログ

(3) 通信終了コマンドが送信キューに保持されます。

SampleApp Websocket

Connected Last 126 entries

Decode Data From Device Send Data To Device

Send Data To Multicast Device Disconnect

Device EUI	Local time	Freq [MHz]	Date rate	RSSI (dBm)	SNR (dB)	FCntUp	Port	Payload
749050FFFE000C2C								11223344 (enqueued for sending)
749050FFFE000C2C	2019-12-09 17:22:02.102	923.800	SF10 BW125 4/5	-33	11.2	8	10	41 c1 33 33
749050FFFE000C2C	2019-12-09 17:21:02.322	923.200	SF10 BW125 4/5	-33	11	7	10	44 7d d1 ec 42 31 0c c0

図 5-6 送信キューへの保持

(4) LoRaWAN module からのセンサデータ受信をトリガとして通信終了コマンドを送信します。

SampleApp Websocket

Connected Last 126 entries

Decode Data From Device Send Data To Device

Send Data To Multicast Device Disconnect

Device EUI	Local time	Freq [MHz]	Date rate	RSSI (dBm)	SNR (dB)	FCntUp	Port	Payload
749050FFFE000C2C	2019-12-09 17:23:02.197							(enqueued data sent)
749050FFFE000C2C	2019-12-09 17:23:01.988	923.400	SF10 BW125 4/5	-33	13	9	10	44 7d d5 1f 42 30 ce 40
749050FFFE000C2C								11223344 (enqueued for sending)
749050FFFE000C2C	2019-12-09 17:22:02.102	923.800	SF10 BW125 4/5	-33	11.2	8	10	41 c1 33 33
749050FFFE000C2C	2019-12-09 17:21:02.322	923.200	SF10 BW125 4/5	-33	11	7	10	44 7d d1 ec 42 31 0c c0

図 5-7 通信終了コマンドの送信

## 6. 動作仕様

「図 3-2 BLE module 通信ソフトウェア構成」と「図 3-3 LoRaWAN module 通信ソフトウェア構成」で示される Host MCU のアプリケーション処理について説明します。

### 6.1 メインループ

本アプリケーションのメインループは、「図 6-1 メインループの動作」で示されるように BLE module 通信部と LoRaWAN module 通信部の 2 つの処理部で構成されています。

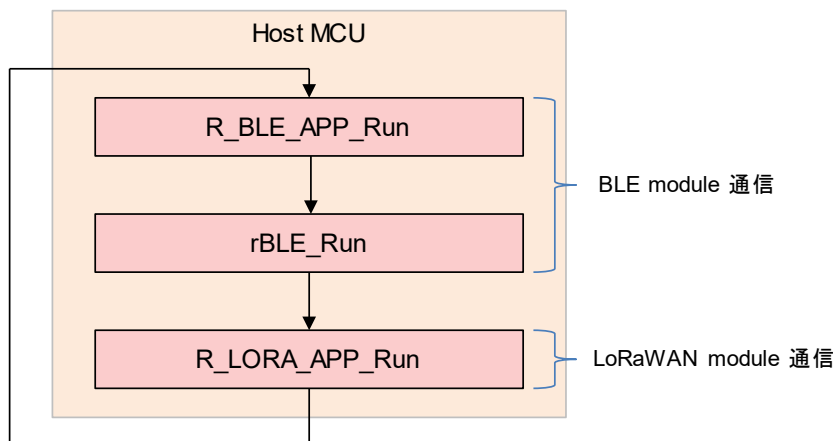


図 6-1 メインループの動作

BLE module 通信部は、R\_BLE\_APP\_Run 関数と rBLE\_Run 関数で実行されます。R\_BLE\_APP\_Run 関数は、スマートフォンと接続して LoRaWAN module の設定パラメータを受信し、AT コマンドに変換して LoRaWAN module へパラメータを設定するアプリケーション部です。rBLE\_Run 関数は、アプリケーション部で呼び出される rBLE API を BLE module へ送信する処理や、BLE module から rBLE Event を受信しアプリケーション部のコールバック関数へ通知する処理を行います。

LoRaWAN module 通信部は、R\_LORA\_APP\_Run 関数で実行されます。Sensor module から測定したセンサデータを読み出して AT コマンドで LoRaWAN module へ送信する処理や、LoRaWAN Gateway から受信した通信終了コマンドの処理を行います。

## 6.2 BLE module 通信フロー

スマートフォンから受信したパラメータを LoRaWAN module へ設定するフローについて「図 6-2 BLE module 通信動作フロー」で説明します。

### 6.2.1 スマートフォンとの接続通信

BLE module 通信部は、メインループの R\_BLE\_APP\_Run 関数と rBLE\_Run 関数で実行されます。

R\_BLE\_APP\_Run 関数は、スマートフォンと接続して LoRaWAN module の設定パラメータを受信し、LoRaWAN module へ AT コマンドでパラメータを設定するアプリケーション部です。rBLE\_Run 関数は、アプリケーション部で呼び出される rBLE API を rBLE module へ送信する処理や、BLE module から rBLE Event 受信しアプリケーション部のコールバック関数へ渡す処理を行います。

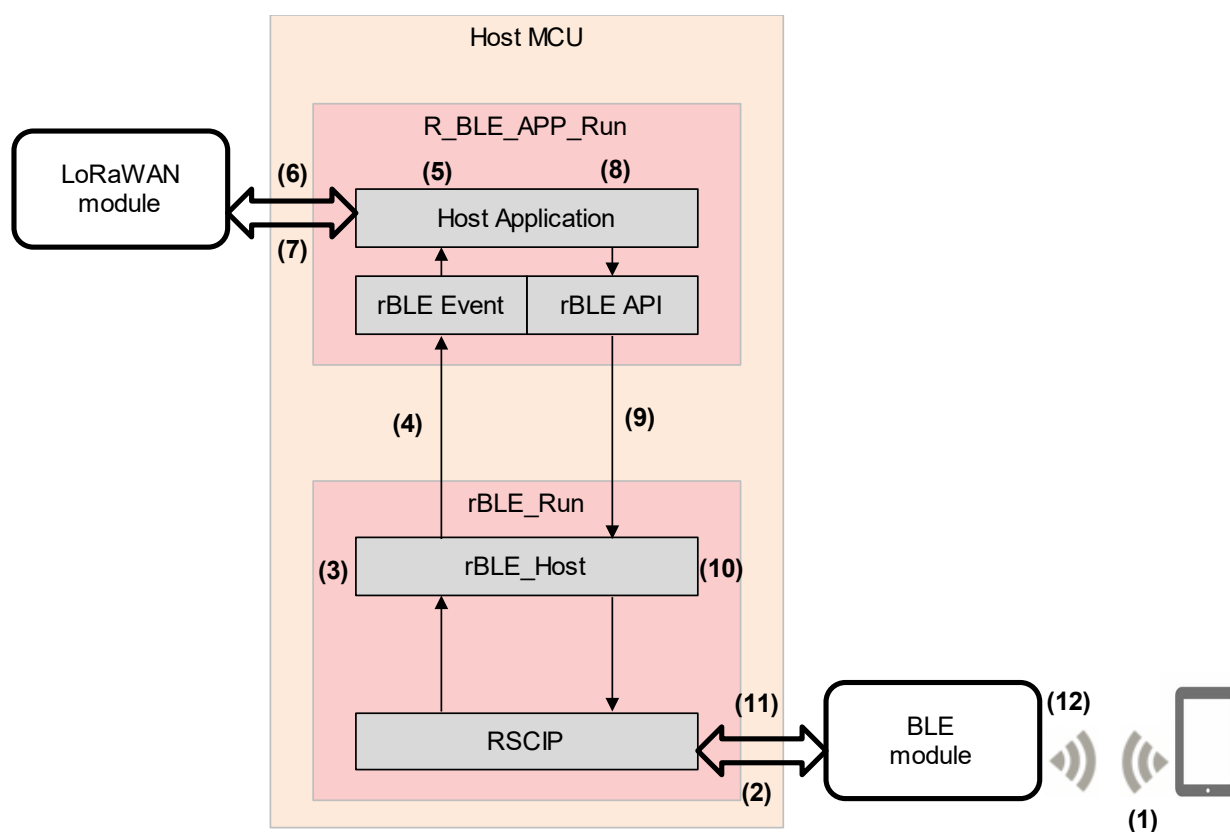


図 6-2 BLE module 通信動作フロー

- (1) スマートフォンは LoRaWAN 設定パラメータを送信します。
- (2) BLE module はスマートフォンから LoRaWAN 設定パラメータを受信すると、rBLE イベントフォーマットを RSCIP で送信し、Host MCU の RSCIP から rBLE\_Host へ渡されます。
- (3) rBLE\_Host は rBLE イベントフォーマットから rBLE イベントへ変換します。
- (4) rBLE\_Host は Host Application のコールバック関数を呼び出して rBLE イベントを通知します。
- (5) Host Application は rBLE イベントから 16 進数の LoRaWAN module 設定パラメータを取り出し、AT コマンド文字列へ変換します。
- (6) Host Application は LoRaWAN module に AT コマンドを送信してパラメータを設定します。
- (7) Host Application は AT コマンドの実行結果文字列を LoRaWAN module から受信します。

- (8) Host Application は AT コマンドの実行結果文字列を、16 進数フォーマットへ変換します。
- (9) Host Application は rBLE API を呼び出します。
- (10) rBLE\_Host は rBLE API と API で指定されたパラメータを rBLE コマンドフォーマットに変換します。
- (11) RSCIP は rBLE コマンドフォーマットを BLE module へ送信します。
- (12) BLE module からスマートフォンへ AT コマンド実行結果が送信されます。

### 6.2.2 スマートフォンとの切断

スマートフォンが切断を実行したことを BLE module が検出すると、LoRaWAN module に設定したパラメータを示すフラグを FPB のデータフラッシュへ保存します。スマートフォンから切断を実行せずに、FPB 電源を落とす、または E2Lite を切断した場合は、パラメータフラグが FPB のデータフラッシュに保存されません。スマートフォンと接続し再度パラメータを設定してください。



## 6.3 LoRaWAN module 通信フロー

Sensor module から読みだしたセンサデータを LoRaWAN module から LoRaWAN Gateway へ送信するフローを「図 6-3 LoRaWAN module 通信動作フロー」で説明します。

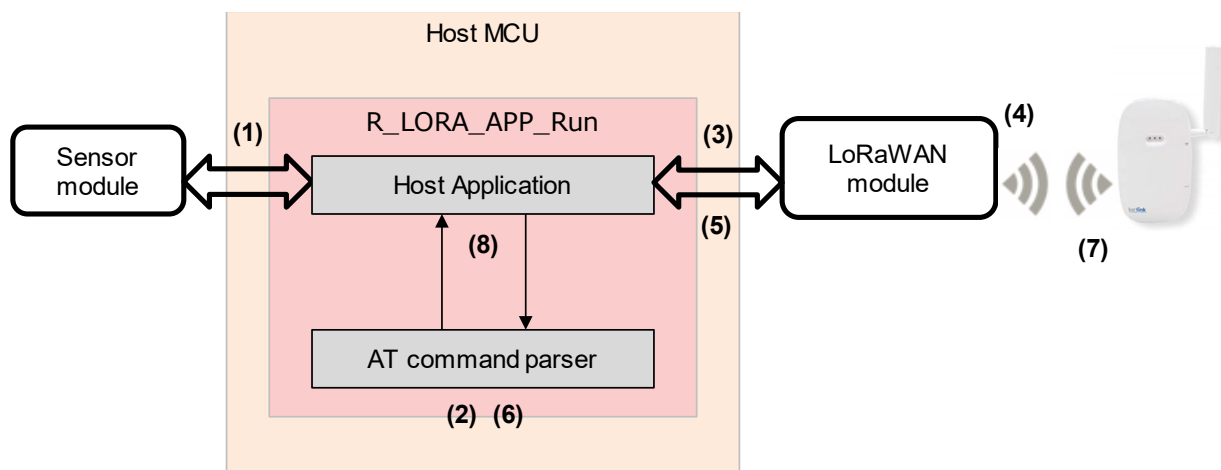


図 6-3 LoRaWAN module 通信動作フロー

## [センサデータ送信 - Uplink 通信]

- (1) Host Application は Sensor module からセンサデータを読み出し、AT command parser へ渡します。
- (2) AT command parser は Host Application が読みだしたセンサデータを含む AT コマンドを生成し、Host Application へ渡します。
- (3) Host Application は AT コマンドを LoRaWAN module へ送信します。
- (4) LoRaWAN module は LoRaWAN Gateway へセンサデータを送信します。
- (5) Host Application はセンサデータ送信結果を受信し、AT command parser へ渡します。
- (6) AT command parser はセンサデータ送信結果を解析し Host Application へ通知します。

## [LoRaWAN Gateway 通信終了コマンド受信 - Downlink 通信]

- (7) LoRaWAN Network Server から LoRaWAN Gateway 通信終了コマンドを送信すると、LoRaWAN Gateway を通して LoRaWAN module が受信し、Host Application へ渡されます。
- (8) Host Application は AT command parser を使用して解析し、LoRaWAN Gateway との通信を終了します。

## 6.4 LoRaWAN module コマンド

LoRaWAN module へのパラメータ設定<sup>※1</sup>に使用する AT コマンドの通信フォーマット仕様、LoRaWAN Gateway との通信を終了するコマンド仕様を説明します。

本アプリケーションでは LoRaWAN module にパラメータを設定するために AT コマンドを 7 つ使用します。また、LoRaWAN Gateway との通信を終了するために独自のコマンドを 1 つ使用します。

AT コマンドの詳細な説明については「LoRaWAN Stack Sample Application Command Reference」(R11AN0231)を参照してください。

【注】 1. 本アプリケーションノートでは、LoRaWAN 評価ボード(LoRaWAN stack sample application board)を使用しているため Device EUI 等のパラメータを設定しています。通常、設定が必要なパラメータは Application Key と Application EUI の 2 つです。その他は、装置固有のパラメータとして書き込まれていることが多いです。

表 6-1 LoRaWAN module 設定 AT コマンド

AT コマンド	説明
AT+REGION	Region を設定
AT+DEVEUI	Device EUI を設定
AT+CLASS	Device class を設定
AT+APPEUI	Application identifier を設定
AT+APPKEY	Application key を設定
AT+ACTMODE	Activation mode を設定
AT+FPORT	Port number を設定

### 6.4.1 パラメータ設定 AT コマンド

スマートフォンから送信する AT コマンドのフォーマット、スマートフォンへ送信する AT コマンド実行結果のフォーマット、設定したパラメータを FPB(RL78/G14)のデータフラッシュに保持するためのパラメータフラグのフォーマットを説明します。

#### 6.4.1.1 AT コマンドフォーマット

LoRaWAN module へ設定するパラメータは、スマートフォンから送信します。LoRaWAN module へ送信する AT コマンドとパラメータは ASCII 文字列ですがパラメータによっては文字列が長くなるため、ASCII 文字列ではなく各 AT コマンドに割り当てた番号とパラメータを含んだ 16 進数の数値列を使用します。BLE module で受信した 16 進数のパラメータは、FPB の Host Application で ASCII 文字列の AT コマンドに変換して LoRaWAN module へ送信します。

スマートフォンから送信する 16 進数 AT コマンドフォーマットと、BLE module で受信した後に変換する文字列の対応表を以下に示します。

16 進数 AT コマンドフォーマット: <AT コマンド番号><パラメータ>

AT コマンド番号 : 1 バイト  
 パラメータ : 可変長

表 6-2 AT コマンド番号対応表

AT コマンド番号	AT コマンド
01	AT+REGION
02	AT+DEVEUI
03	AT+CLASS
04	AT+APPEUI
05	AT+APPKEY
06	AT+ACTMODE
07	AT+FPORT

表 6-3 AT コマンド数値列と文字列の対応表

AT コマンド数値列 <sup>注1</sup>	AT コマンド文字列
0106	AT+REGION=06
02749050FFFE000C2C	AT+DEVEUI=749050FFFE000C2C
0300	AT+CLASS=00
040123456701234567	AT+APPEUI=0123456701234567
0555555555555555AAAAAAAAAAAAAAAA	AT+APPKEY=5555555555555555AAAAAAAAAAAAAAAA
0601	AT+ACTMODE=01
0710	AT+FPORT=10

【注】 1. 先頭の1バイトがATコマンド番号、以降がパラメータです。パラメータはLoRaWAN Gatewayとの通信環境により異なります。

#### 6.4.1.2 AT コマンド結果コードフォーマット

LoRaWAN module に AT コマンドを送信すると、実行結果が ASCII 文字列で返ります。FPB の Host Application は、実行結果を ASCII 文字列から 16 進数の数値列に変換して、BLE module からスマートフォンへ送信します。

スマートフォンから送信する AT コマンド実行結果のフォーマットを以下に示します。

16 進数 AT コマンド結果フォーマット:

<AT コマンド番号<sup>注1</sup>><AT コマンド実行結果<sup>注2</sup>><パラメータフラグ<sup>注3</sup>>

AT コマンド番号 : 1 バイト

AT コマンド実行結果 : 1 バイト

パラメータフラグ : 1 バイト

- 【注】 1. 「表 6-2 AT コマンド番号対応表」を参照してください。  
 2. 「表 6-4 AT コマンド実行結果」を参照してください。  
 3. 「表 6-5 AT コマンドパラメータフラグ」を参照してください。

表 6-4 AT コマンド実行結果

AT コマンド実行結果(ASCII)	AT コマンド実行結果(16 進数)
OK	01
ERROR	00

### 6.4.1.3 AT コマンドパラメータフラグ

LoRaWAN module に設定した AT コマンドの実行結果は、ビットフィールドのパラメータフラグとして FPB(RL78/G14)のデータフラッシュへ保存します。パラメータフラグは、プログラム実行時に読み出され、LoRaWAN module 通信の実行可否に利用されます。BLE module 通信でスマートフォンから設定された AT コマンドパラメータは BLE module がスマートフォンとの切断を検出した後に FPB のデータフラッシュへ保存します。スマートフォンから切断を実行せずに、FPB の電源を落とす、または E2Lite を切断した場合、パラメータフラグは FPB のデータフラッシュに保存されないので再度設定してください。

表 6-5 AT コマンドパラメータフラグ

bit	ビット名
0	region
1	deveui
2	class
3	appeui
4	appkey
5	actmode
6	fport
7	-

### 6.4.2 LoRaWAN Gateway 通信終了コマンド

LoRaWAN Gateway とのセンサデータ通信を終了させるコマンドで、LoRaWAN Network Server から送信します。LoRaWAN module がコマンドを受信すると、Host Application はセンサデータの送信を終了し IDLE 状態になります。

通信終了コマンド: 11223344

## 6.5 BLE Modem 構成 UART 2 線分岐接続方式

BLE module とのシリアル通信で使われる UART 2 線分岐接続方式の通信方法について示します。本アプリケーションノートで使用する FPB と BLE module との UART 2 線分岐接続方式での接続については、「表 3-1 PMOD1 コネクタピンアサイン」を参照してください。

### 6.5.1 送信動作

FPB(Host MCU)から BLE module(BLE MCU)への送信を行うには、ハンドシェイクを行う必要があります。ハンドシェイクは Host MCU から送信する REQ バイト(0xC0)と、BLE MCU から送信される ACK バイト(0x88)または RSCIP パケットによって行われます。また、ハンドシェイクを行う時にはタイマによる監視を行い、タイムアウト発生時にはハンドシェイクを再実行します。Host MCU の UART ドライバは、このハンドシェイクを行うため、送信状況によって 5 つの状態を持ちます。

表 6-6 UART ドライバ送信状態

状態	説明
T_IDLE	UART ドライバ初期化、RSCIP パケット送信完了
T_REQUESTING	REQ バイト送信中
T_RCV_BF_REQUESTED	ACK バイトの代わりに BLE MCU から RSCIP パケットを受信
T_REQUESTED	REQ バイト送信完了(BLE MCU からの ACK バイト待ち)
T_ACTIVE	RSCIP パケット送信中

Host MCU から BLE MCU への送信は必ず REQ バイトから開始されます。REQ バイトを送信した後、Host MCU は受信状態により次の動作のいずれかに分岐します。

- (a) Host MCU が BLE MCU からの RSCIP パケットを受信していない(図 6-4)
- (b) Host MCU が BLE MCU からの RSCIP パケットを受信中(図 6-5)
- (c) ACK バイト受信タイムアウト(図 6-6)

#### (1) Host MCU が BLE MCU からの RSCIP パケットを受信していない

この状態は、BLE MCU から RSCIP パケットが送信されておらず、Host MCU から REQ バイトを送信した後、Host MCU が ACK バイトの受信を待っている状態です。BLE MCU は REQ バイトを受信し ACK バイトを送信します。ACK バイトを受信した Host MCU は、BLE MCU に RSCIP パケットを送信します。

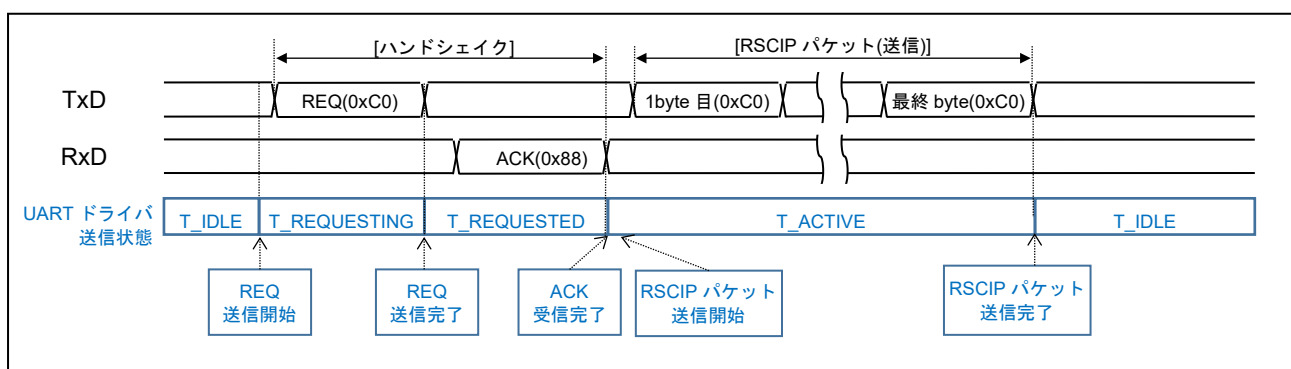


図 6-4 Host MCU が BLE MCU からの rBLE パケットを受信していない場合の動作

## (2) Host MCU が BLE MCU からの RSCIP パケットを受信中

この状態は BLE MCU が RSCIP パケットを送信しており、Host MCU は RSCIP パケットを受信している状態です。BLE MCU は REQ を受信しても ACK バイトを返さず、送信している RSCIP パケットを ACK バイトの代わりにとします。Host MCU は BLE MCU からの RSCIP パケットを ACK バイトの代わりにとし、BLE MCU に RSCIP パケットを送信します。

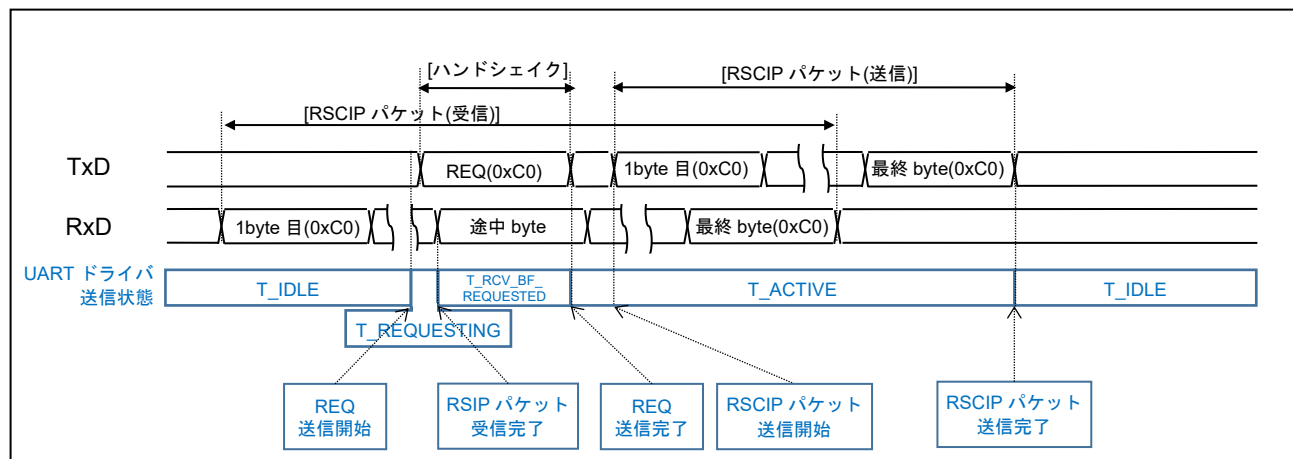


図 6-5 Host MCU が BLE MCU からのデータを受信している場合の動作

## (3) ACK バイト受信タイムアウト

REQ バイトを送信した後 Host MCU は、タイムアウトタイマを動作させます。一定時間 ACK バイトを受信できなかった場合、REQ バイトを再送します。

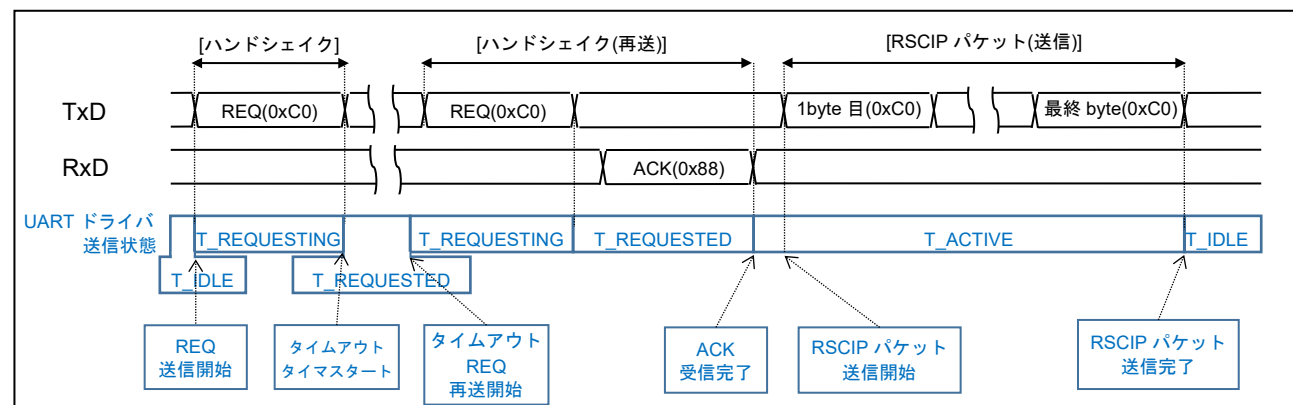


図 6-6 ACK バイトの受信がタイムアウトした場合の動作

## 6.5.2 受信動作

受信時に UART ドライバの状態遷移はありません。BLE MCU からのデータを受信するために、rBLE\_Host から指定されたバイト数で BLE MCU からの RSCIP パケットを待ち受けます。

## 7. BLE 通信シーケンス

Host MCU と BLE MCU で構成される Local Device と、スマートフォン等の Remote Device との通信シーケンスを示します

### 7.1 メインシーケンス

メインシーケンス図では、処理ブロックとして Step1~10 までを定義し、処理ブロックの順序と関連するデバイスまたはソフトウェアの範囲を示します。処理ブロック Step1~10 の詳細は次節以降に記載します。

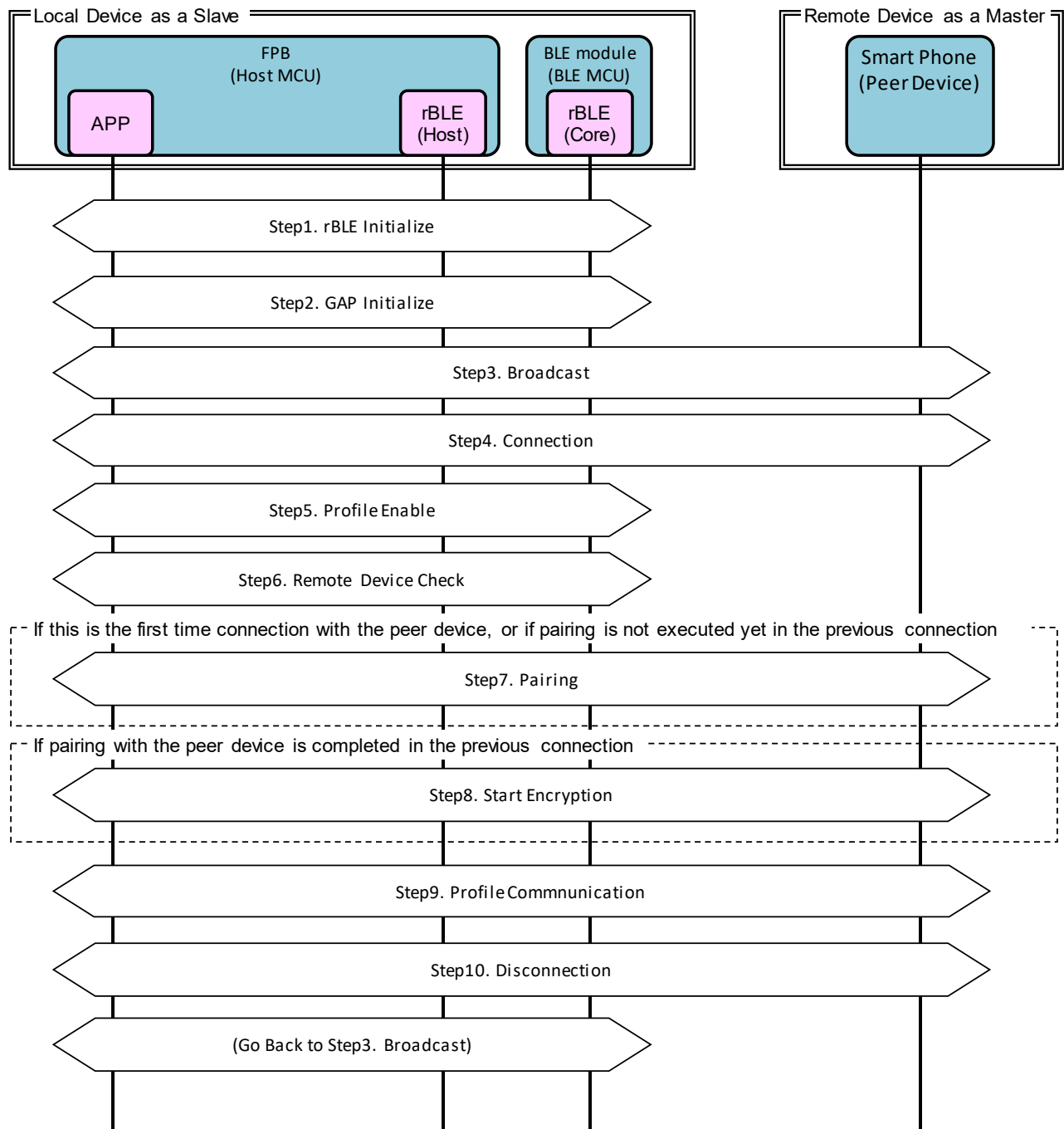


図 7-1 メインシーケンス

## 7.2 Step1. rBLE Initialize シーケンス

APP は RBLE\_Init 関数をコールし、rBLE (rBLE\_Host/rBLE\_Core) を初期化します。rBLE の初期化が完了し BLE MCU とのシリアル通信が確立されると、rBLE から RBLE\_MODE\_ACTIVE イベントが通知されます。

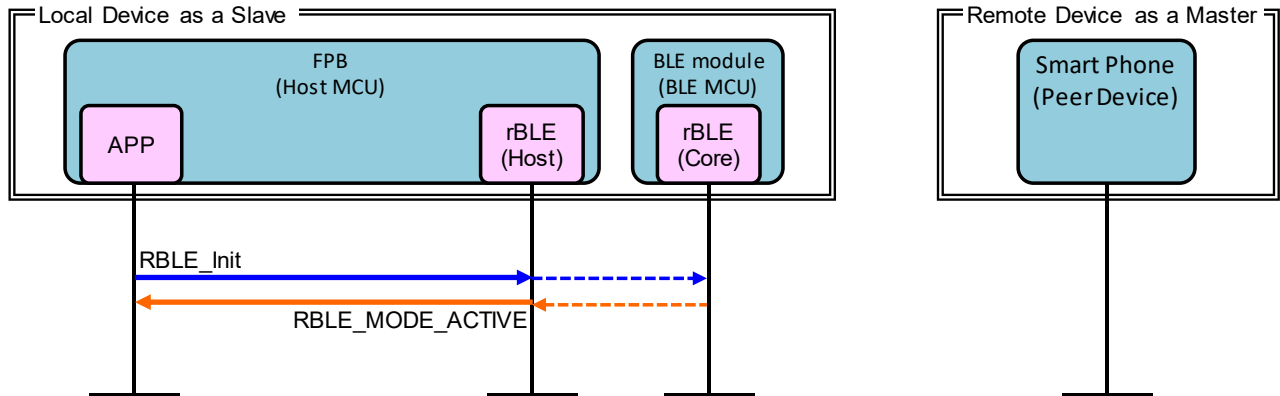


図 7-2 rBLE Initialize シーケンス

## 7.3 Step2. GAP Initialize シーケンス

APP は RBLE\_GAP\_Reset 関数をコールし、GAP を初期化します。初期化が完了すると、rBLE から RBLE\_GAP\_EVENT\_RESET\_RESULT イベントが通知されます。

APP は RBLE\_GAP\_Set\_Bonding\_Mode 関数をコールし、Bonding 許可を設定します。設定が完了すると、rBLE から RBLE\_GAP\_EVENT\_SET\_BONDING\_MODE\_COMP イベントが通知されます。

APP は RBLE\_GAP\_Set\_Security\_Request 関数をコールし、セキュリティレベルを設定します。設定が完了すると、rBLE から RBLE\_GAP\_EVENT\_SET\_SECURITY\_REQUEST\_COMP イベントが通知されます。

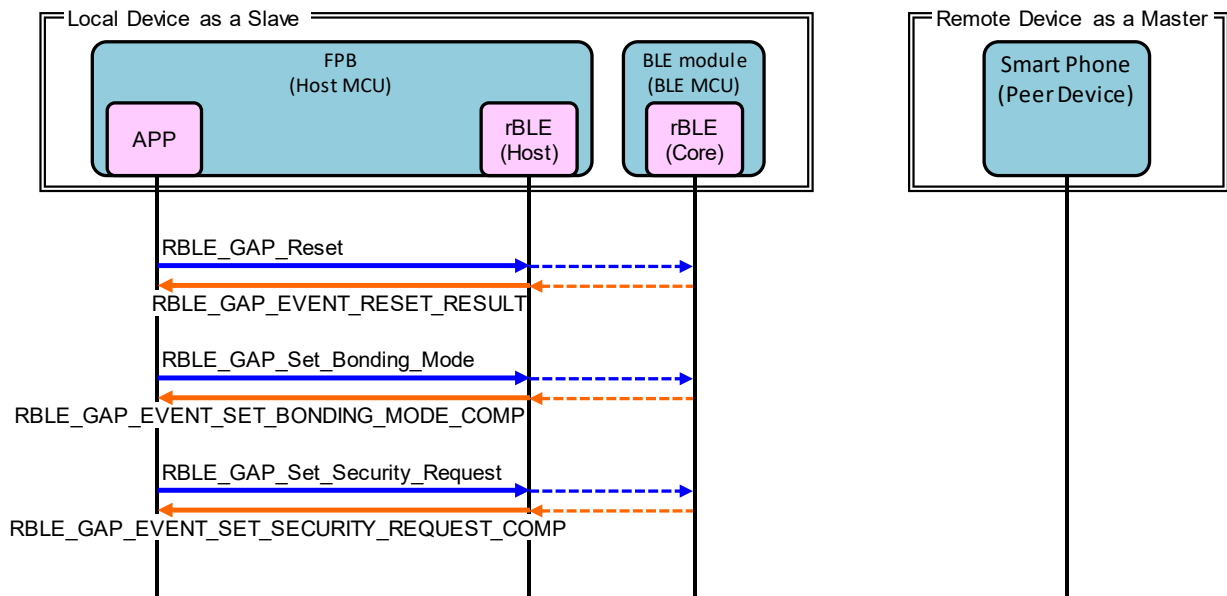


図 7-3 GAP Initialize シーケンス



### 7.4 Step3. Broadcast シーケンス

Local Device を Slave として接続するための Broadcast を開始します。

APP は `RBLE_GAP_Broadcast_Enable` 関数をコールし、Broadcast を開始します。開始が完了すると、rBLE から `RBLE_GAP_EVENT_BROADCAST_ENABLE_COMP` イベントが通知されます。

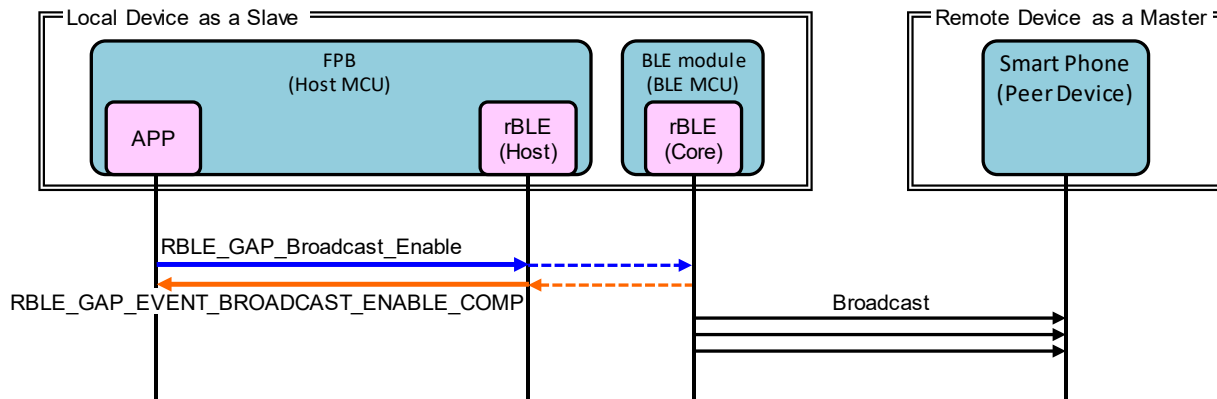


図 7-4 Broadcast シーケンス

### 7.5 Step4. Connection シーケンス

Local Device からの Broadcast を受信した Remote Device は、接続を要求します。

Remote Device から Connection Request が送信され、Local Device と Remote Device の接続が確立されると、rBLE から `RBLE_GAP_EVENT_CONNECTION_COMP` イベントが通知されます。

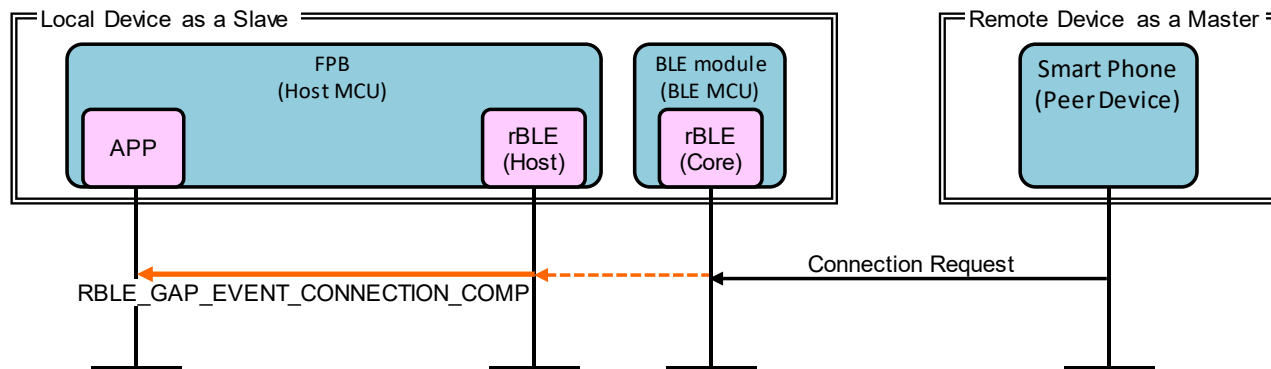


図 7-5 Connection シーケンス

## 7.6 Step5. Profile Enable シーケンス

データ送信に利用する GPCP (General Purpose Communication Profile) の Server を有効化します。

APP は R\_BLE\_VUART\_Server\_Enable 関数をコールし、GPCP を有効化します。有効化は、Remote Device から Indication を許可する Write Client Characteristic Configuration が送信されると完了します。「[図 7-10 Profile Communication シーケンス](#)」を参照してください。

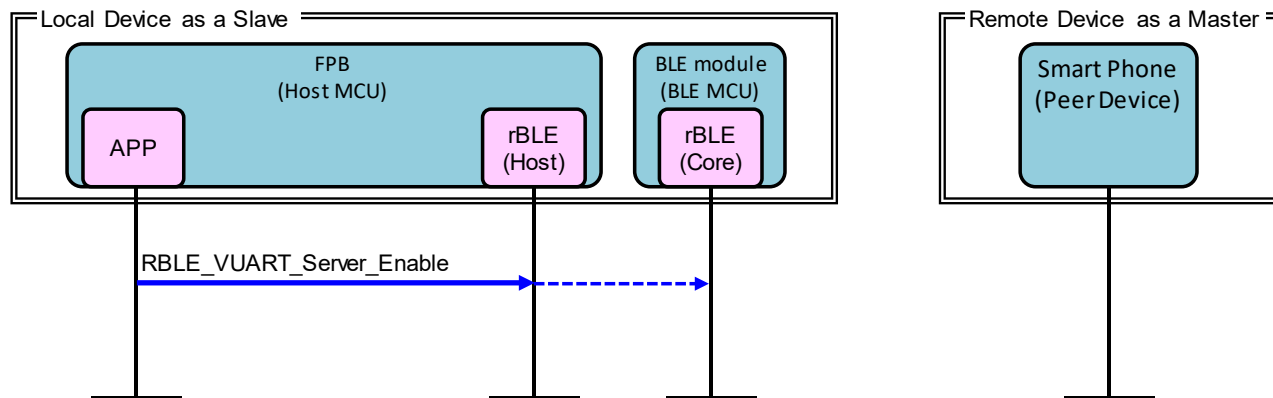


図 7-6 Profile Enable シーケンス

## 7.7 Step6. Remote Device Check シーケンス

接続完了後は、Remote Device とのセキュリティ情報を確認します。

Remote Device アドレスが Public アドレス、または Resolvable Private Address 以外の Random アドレスの場合、Remote Device に関するセキュリティ情報の要求として、rBLE から RBLE\_SM\_CHK\_BD\_ADDR\_REQ イベントが通知されます。APP は RBLE\_SM\_Chk\_Bd\_Addr\_Req\_Resp 関数をコールし、保持しているセキュリティ情報の通知またはセキュリティ情報を保持していないことを通知します。

Remote Device アドレスが Resolvable Private Address の場合アドレス解決のための IRK (Identity Resolving Key) の要求として、rBLE から RBLE\_SM\_IRK\_REQ\_IND イベントが通知されます。APP は RBLE\_SM\_Irk\_Req\_Resp 関数をコールし、IRK を通知または IRK を保持していないことを通知します。アドレスが解決した場合、rBLE から RBLE\_GAP\_EVENT\_RPA\_RESOLVED イベントが通知されます。

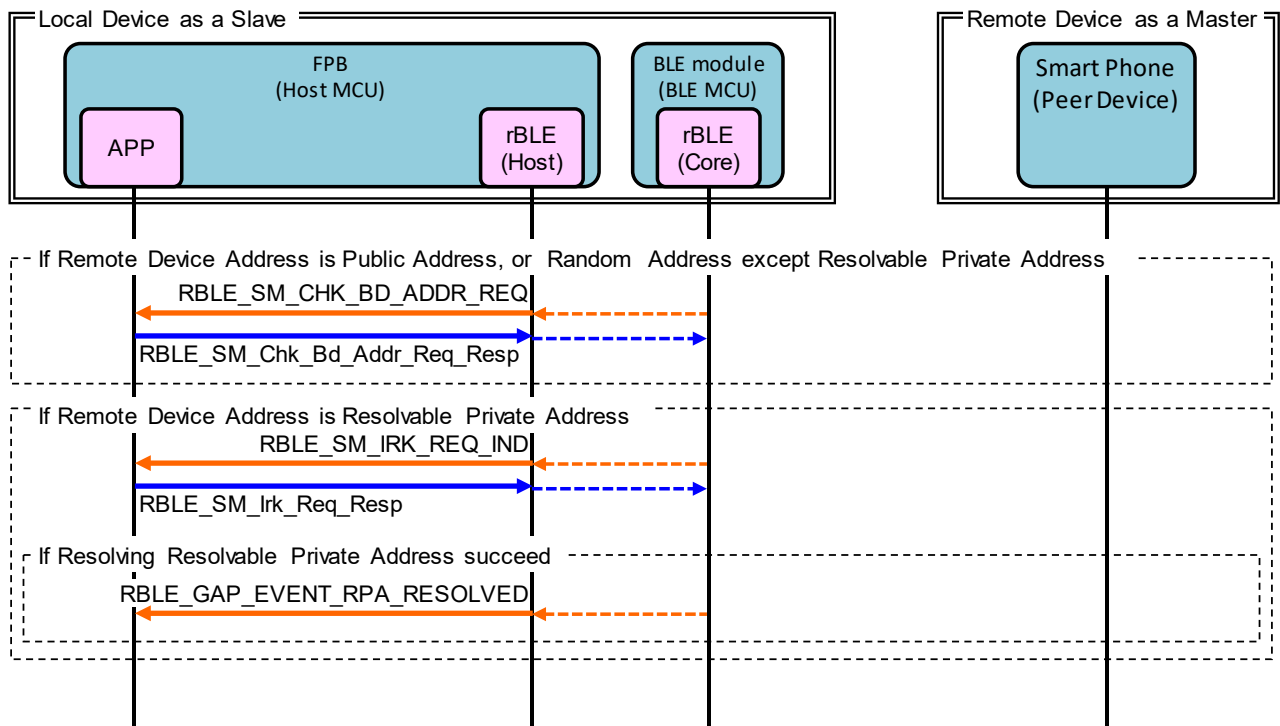


図 7-7 Remote Device Check シーケンス

## 7.8 Step7. Pairing シーケンス

Remote Device との初回接続または以前の接続にて Pairing を行っていない場合、Remote Device からの Pairing 要求により Pairing シーケンスを開始します。Pairing シーケンスは PHASE1、PHASE2、暗号化開始、PHASE3 で構成されます。

PHASE1 では、Local Device および Remote Device のペアリングフィーチャーを交換します。

Remote Device から Pairing Request が送信されると、rBLE から RBLE\_GAP\_EVENT\_BONDING\_REQ\_IND イベントが通知されます。APP は RBLE\_GAP\_Bonding\_Response 関数をコールし、Remote Device に Pairing Response を送信します。

PHASE2 では、STK (Short Term Key) を生成します。

TK (Temporary Key) の要求として、rBLE から RBLE\_SM\_TK\_REQ\_IND イベントが通知されます。APP は RBLE\_SM\_Tk\_Req\_Resp 関数をコールして TK (Temporary Key) を rBLE に通知します。BLE MCU による STK の生成が完了すると、STK を使用した暗号化を開始します。

PHASE3 では、Local Device および Remote Device の暗号化キーを配布します。

LTK (Long Term Key) の要求として、rBLE から RBLE\_SM\_LTK\_REQ\_IND イベントが通知されます。APP は RBLE\_SM\_Ltk\_Req\_Resp 関数をコールして LTK を通知し、Remote Device に Encryption Information (LTK) を送信します。

Remote Device から Encryption Information (LTK) が送信されると、rBLE から RBLE\_SM\_KEY\_IND イベントが通知されます。

Remote Device から Identity Information (IRK) が送信されると、rBLE から RBLE\_SM\_KEY\_IND イベントが通知されます。

Pairing が完了すると、rBLE から RBLE\_GAP\_EVENT\_BONDING\_COMP イベントが通知されます。

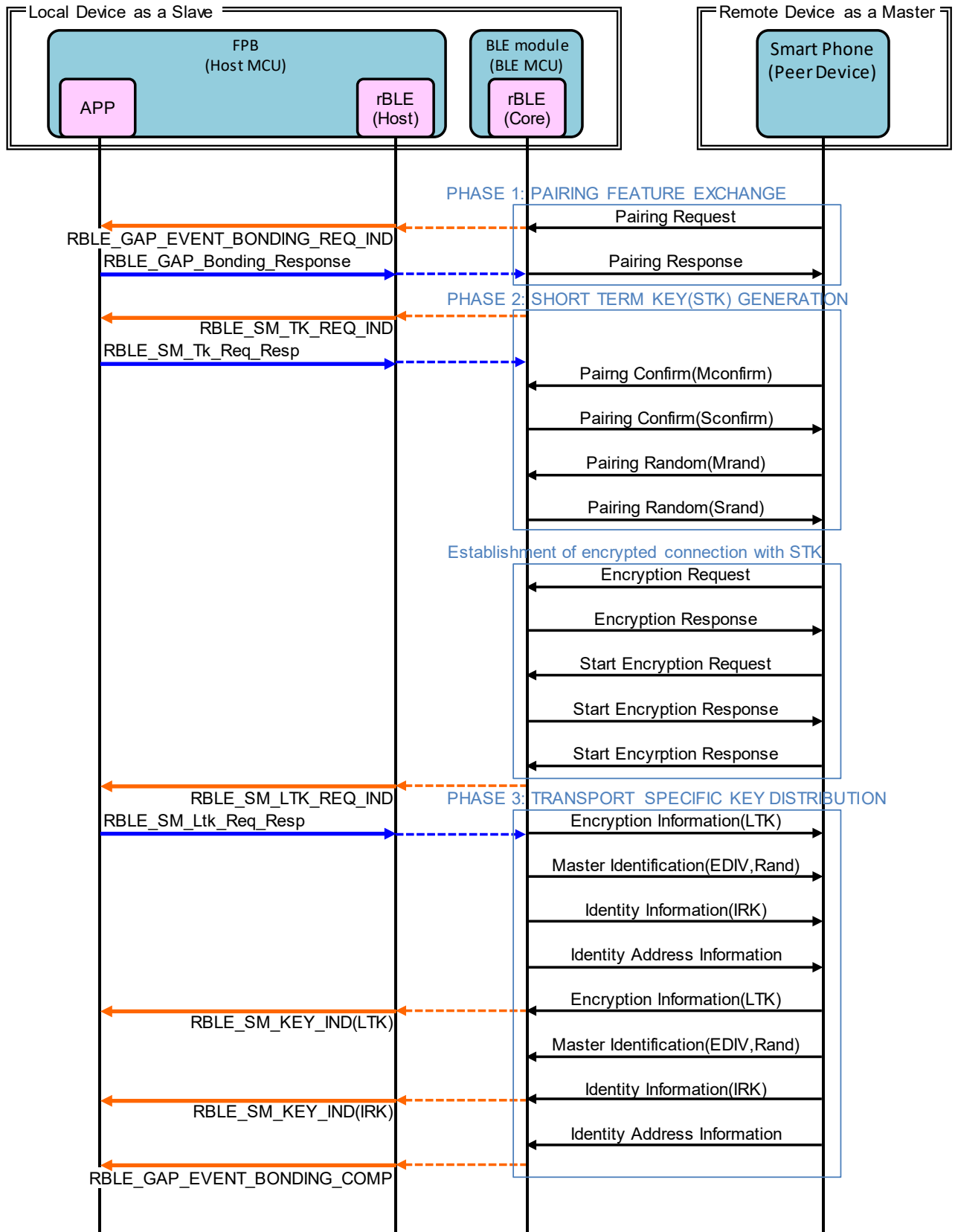


図 7-8 Pairing シーケンス

## 7.9 Step8. Start Encryption シーケンス

以前の接続で Pairing が完了した場合、LTK (Long Term Key) による暗号化を開始します。

Remote Device から Encryption Request が送信されると、rBLE から RBLE\_SM\_LTK\_REQ\_FOR\_ENC\_IND イベントが通知されます。APP は RBLE\_SM\_Ltk\_Req\_Resp 関数をコールして LTK を通知し、Remote Device に Encryption Response を送信します。

BLE MCU は、Remote Device からの Start Encryption Request に対する Start Encryption Response を送信します。

暗号化開始が完了すると、rBLE から RBLE\_SM\_ENC\_START\_IND イベントが通知されます。

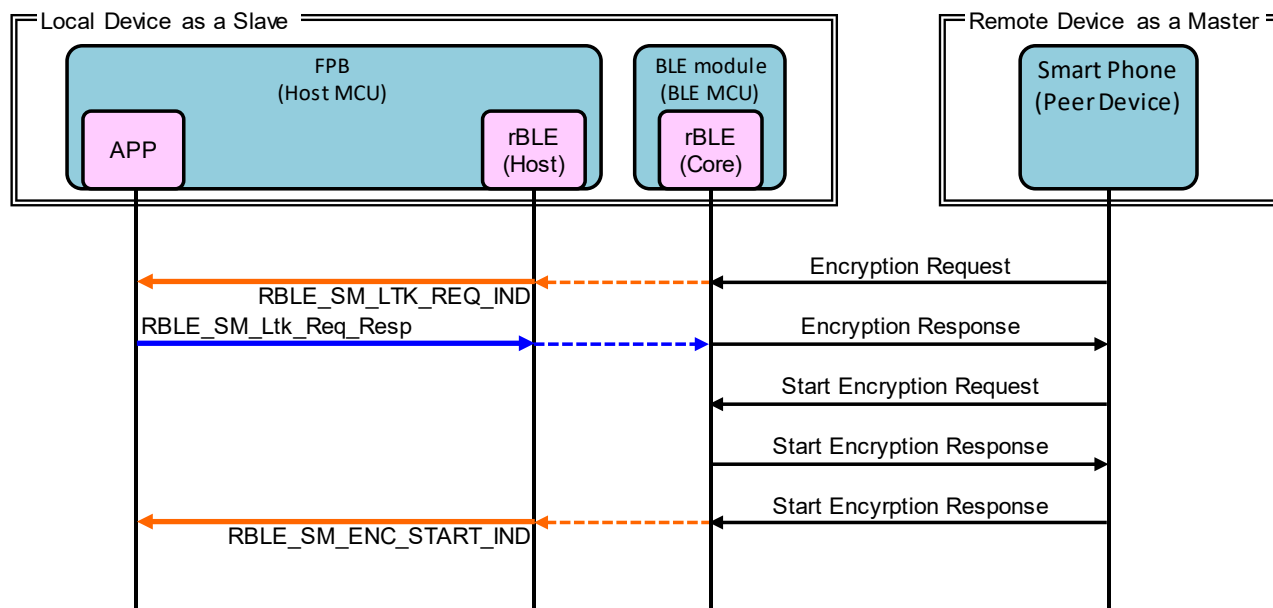


図 7-9 Start Encryption シーケンス

## 7.10 Step9. Profile Communication シーケンス

GPCP (General Purpose Communication Profile) を利用して LoRaWAN module に設定するパラメータを送信します。

Remote Device から Indication を許可する Write Client Characteristic Configuration が送信されると、rBLE から RBLE\_VUART\_EVENT\_SERVER\_ENABLE\_COMP イベントが通知されます。

Remote Device から LoRaWAN module に設定するパラメータを送信すると Local Device で RBLE\_VUART\_EVENT\_SERVER\_WRITE\_REQ が発生しパラメータを受信します。Local Device はパラメータを LoRaWAN module に設定します。

パラメータの設定結果を Indication データとして Remote Device へ送信します。Remote Device は Indication を受信すると Confirmation を送信します。Local Device が Confirmation を受信すると RBLE\_VUART\_EVENT\_SERVER\_INDICATION\_CFM イベントが通知されます。

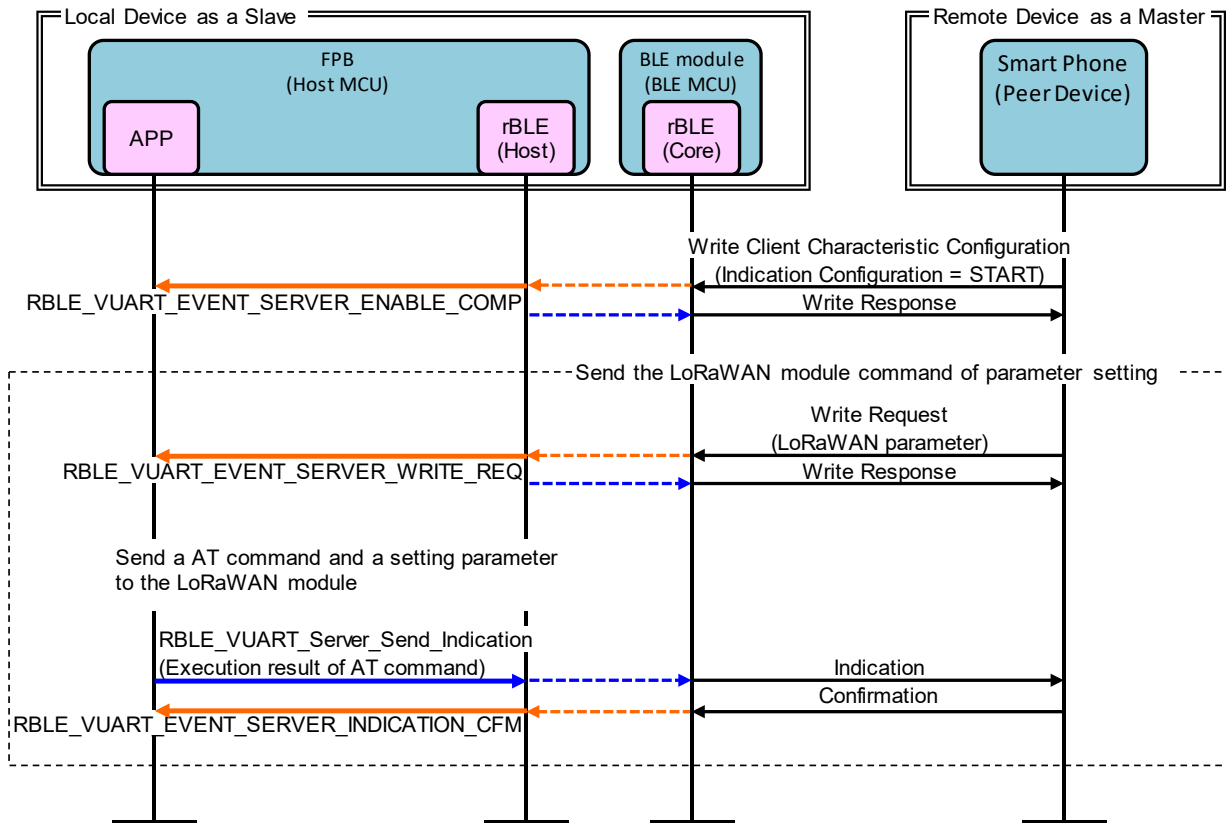


図 7-10 Profile Communication シーケンス

### 7.11 Step10. Disconnection シーケンス

Remote Device から接続の切断のための Disconnect が送信されると、切断が完了し RBLE\_GAP\_EVENT\_DISCONNECT\_COMP イベントが通知されます。

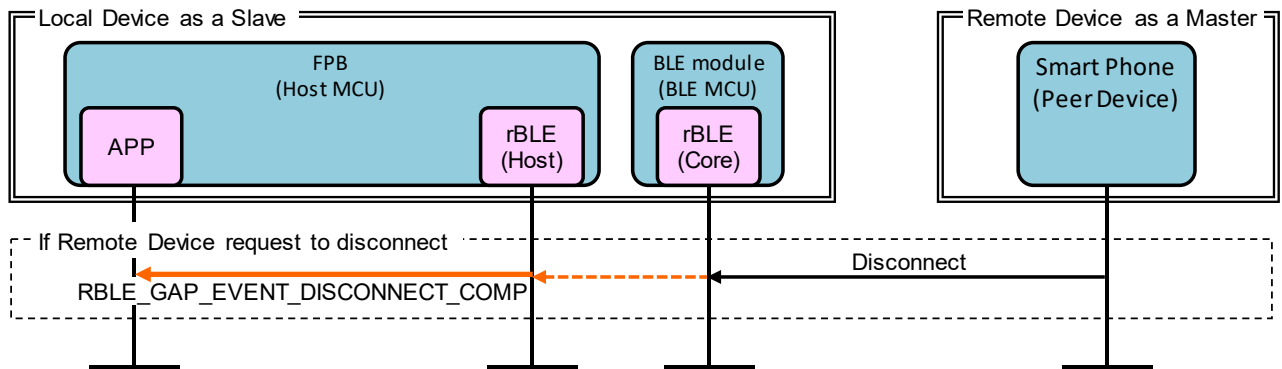


図 7-11 Disconnection シーケンス

## 8. LoRaWAN module AT コマンドシーケンス

FPB の Host Application と LoRaWAN module で通信する AT コマンドシーケンスを示します。

### 8.1 メインシーケンス

AT コマンドシーケンスを処理ブロック Step1~3 に分けて説明します。

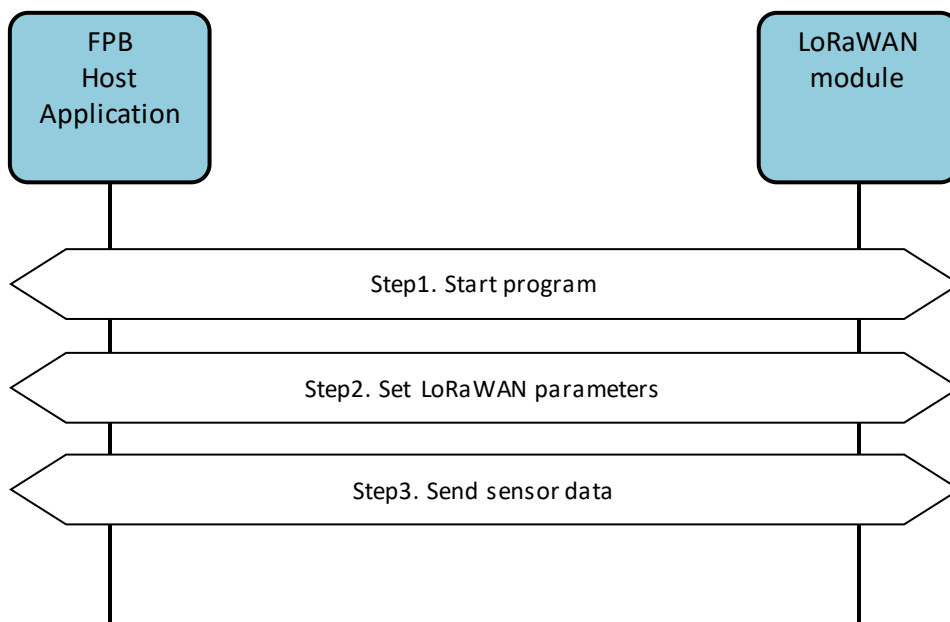


図 8-1 AT コマンド・メインシーケンス

### 8.2 Step1. Start program

プログラム開始直後に LoRaWAN module からのエコーバックを禁止します。

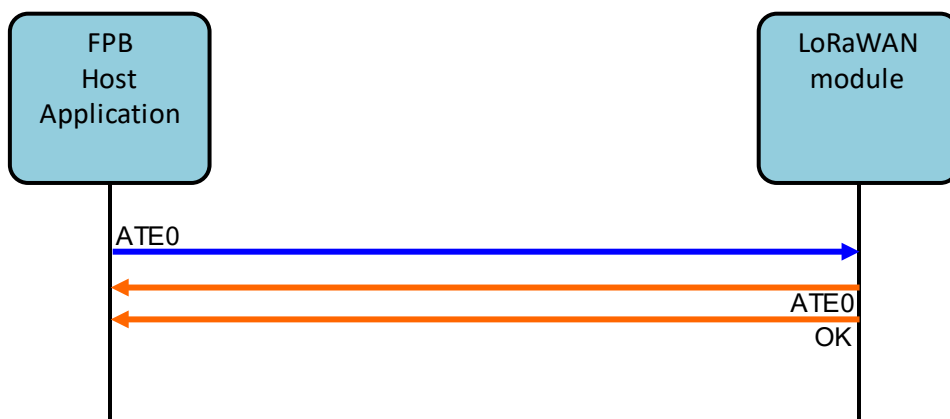


図 8-2 Start program



### 8.3 Step2. Set LoRaWAN parameters

LoRaWAN module に AT コマンドでパラメータを設定します。

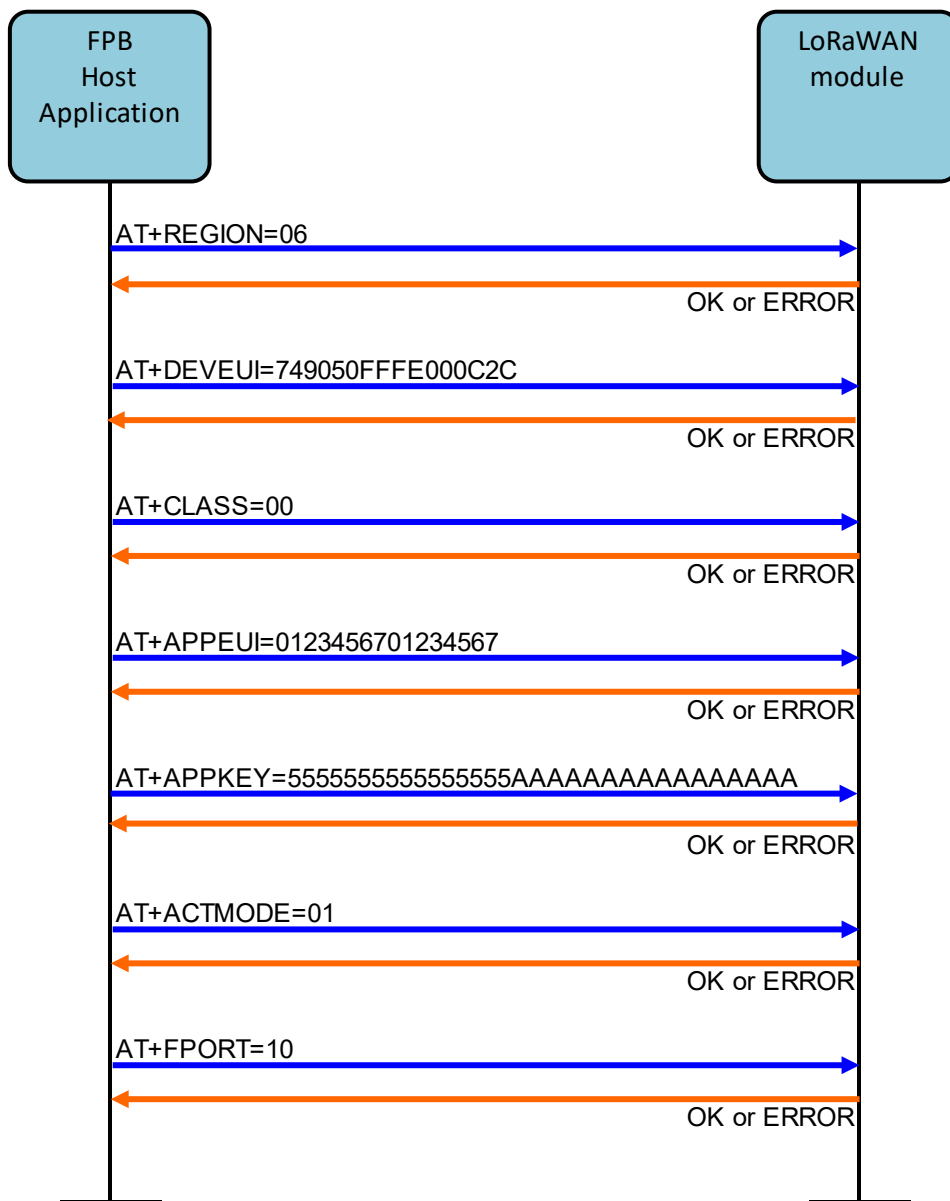


図 8-3 Set LoRaWAN parameters

## 8.4 Step3. Send sensor data

LoRaWAN Gateway と通信するための AT コマンドシーケンスを示します。最初に LoRaWAN Gateway と津伸するために Activation を実行しセンサデータの送信を開始します。送信を停止するには LoRaWAN Gateway から通信終了コマンドを送信します。

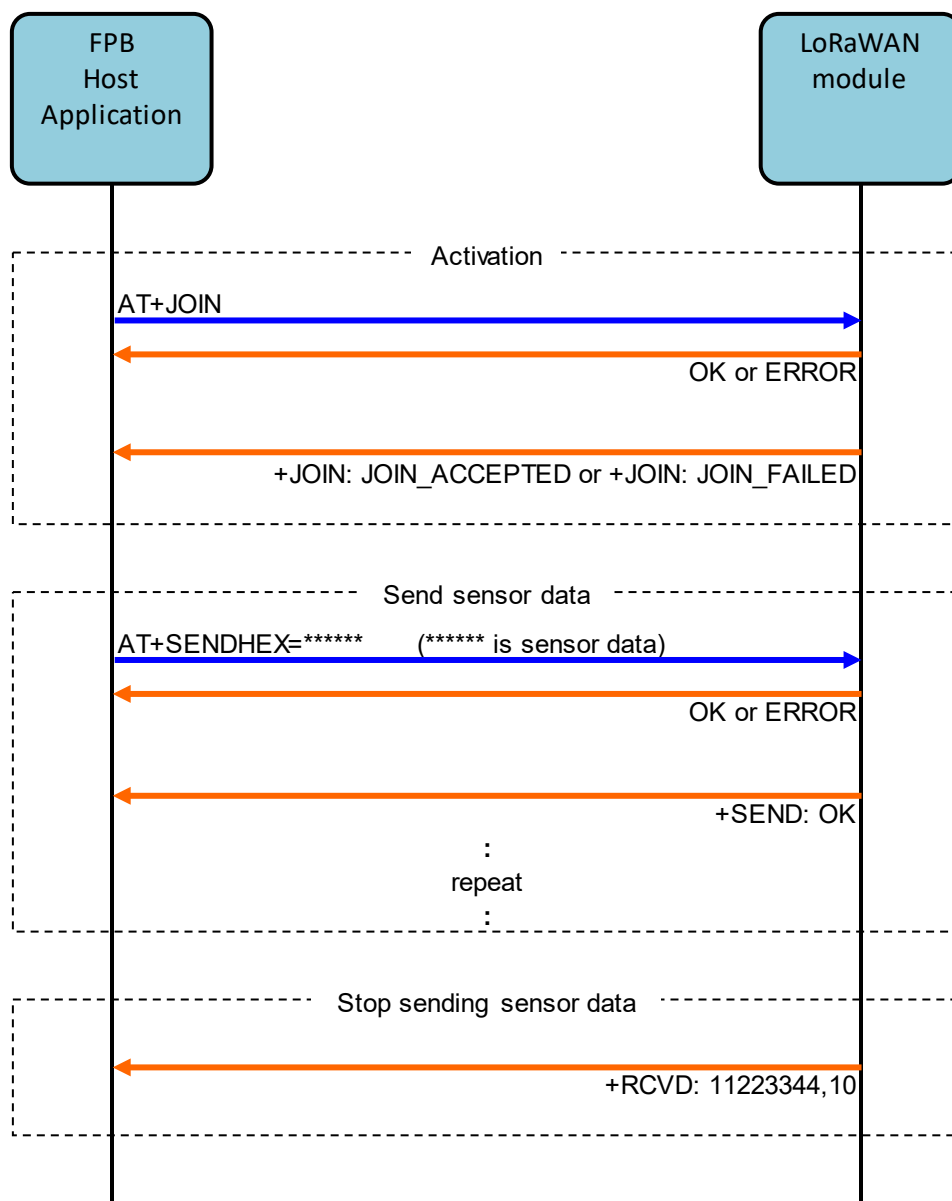


図 8-4 Send sensor data

## 9. 付録

### 9.1 ROM・RAM サイズ

「表 9-1 ROM・RAM サイズ」にアプリケーションが使用する ROM・RAM サイズを示します。

表 9-1 ROM・RAM サイズ

コンパイラ	ROM (bytes)	RAM (bytes)
CC-RL V1.08	46,608	5,195

### 9.2 参考文献

1. [Bluetooth Core Specification v4.2, Bluetooth SIG](#)
2. [Bluetooth SIG Assigned Numbers](#)
3. [Services UUID](#)
4. [Characteristics UUID](#)

## 改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2019.12.25	—	初版発行

## 製品ご使用上の注意事項

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

### 1. 静電気対策

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

### 2. 電源投入時の処置

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

### 3. 電源オフ時における入力信号

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

### 4. 未使用端子の処理

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

### 5. クロックについて

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

### 6. 入力端子の印加波形

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

### 7. リザーブアドレス（予約領域）のアクセス禁止

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

### 8. 製品間の相違について

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

## ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器・システムの設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含まれます。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）、交通管制（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等

- 当社製品は、データシート等により高信頼性、Harsh environment 向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。
6. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合および事故につきましては、当社は、一切その責任を負いません。
  7. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
  8. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
  9. 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
  10. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものとなります。
  11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
  12. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。
- 注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。
- 注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

(Rev.4.0-1 2017.11)

## 本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

[www.renesas.com](http://www.renesas.com)

## お問合せ窓口

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

[www.renesas.com/contact/](http://www.renesas.com/contact/)

## 商標について

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