RA4W1

Bluetooth Mesh イントロダクション

要旨

Flexible Software Package は Bluetooth Mesh Profile1.0.1 仕様および Bluetooth Mesh Model1.0.1 仕様に準拠した Bluetooth Mesh モジュールを提供します。本書は Flexible Software Package の Bluetooth Mesh モジュールと、RA4W1 Device Web サイトにて公開済のアプリケーションノート群の紹介、並びに同モジュールおよび同アプリケーションノートを利用したアプリケーション開発手法を示します。

対象デバイス

RA4W1 グループ

関連文書

下記の文書がルネサスのウェブサイトにて公開されています。

文書名	文書番号
RA4W1 グループ ユーザーズマニュアル ハードウェア編	R01UH0883
Renesas Flexible Software Package Documentation User's Manual	Link
RA4W1 グループ Bluetooth Mesh スタートアップガイド	R01AN5847
RA4W1 グループ Bluetooth Mesh サンプルアプリケーション アプリケーションノート	R01AN5848
RA4W1 グループ Bluetooth Mesh 開発ガイド	R01AN5849
RA4W1 Group EK-RA4W1 Quick Start Guide	R20QS0015
RA4W1 グループ EK-RA4W1 ユーザーズマニュアル	R20UT4683
Renesas RA4W1 Web サイト	Link
Bluetooth SIG	Link

目次

1. Flexible Software Package の構成 3
1.1 Bluetooth Mesh モジュール 3
1.2 モデル 4
1.3 API 5

2. アプリケーションノート 6

3. ユーザアプリケーション作成例 8
3.1 プロジェクトのインポート 8
3.2 Generic Battery Model の Open 9
3.3 Generic Battery Model の Server 側 Callback 関数実装 10
3.4 GET メッセージの送信 11
3.5 Generic Battery Model の Client 側 Callback 関数実装 11
3.6 動作例 12

改訂記録 14

1. Flexible Software Package の構成

1.1 Bluetooth Mesh モジュール

Flexible Software Package は Bluetooth Mesh Profile1.0.1 仕様および Bluetooth Mesh Model1.0.1 仕様に準拠した Bluetooth Mesh モジュールを提供します。同モジュールの構成は以下の通りです。

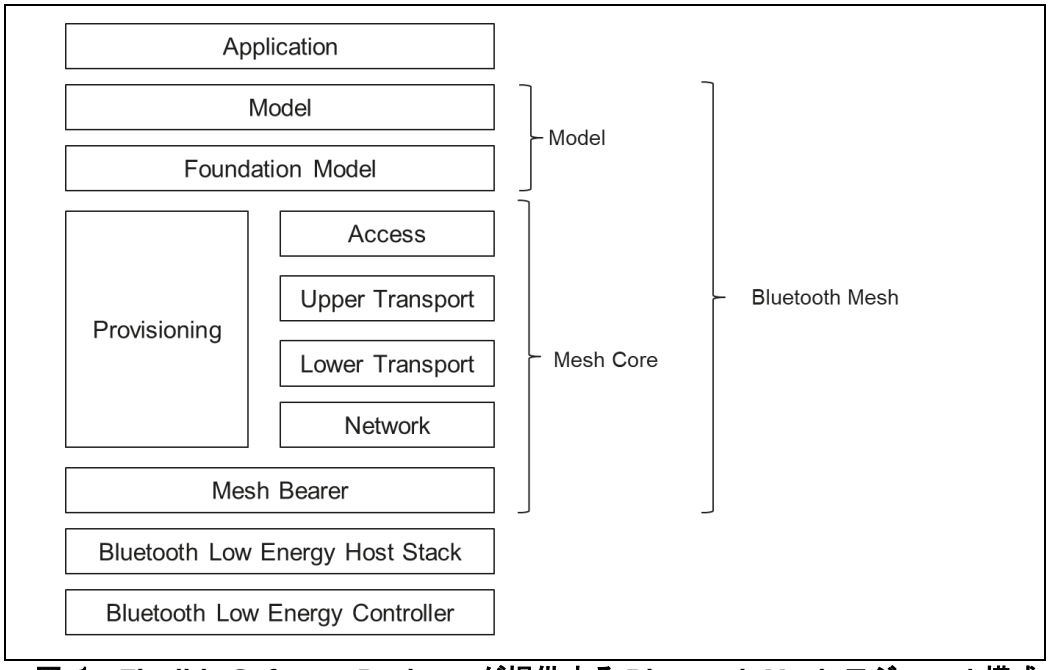

図 1 Flexible Software Package が提供する Bluetooth Mesh モジュール構成

- **Model 層**

  Model 層はアプリケーションのシナリオに基づいて動作するための基本的な機能を標準化します。Flexible Software Package の Bluetooth Mesh モジュールが提供するモデルは 1.2 節を参照ください。

- **Foundation model 層**

  ファンデーションモデルレイヤーは、メッシュネットワークの構成と管理に必要な状態、メッセージ、およびモデルを定義します。

- **Access 層**

  Access 層は上位のアプリケーションが Upper Transport 層を、どの様に使用できるかを定義します。

- **Upper Transport 層**

  Upper Transport 層はアプリケーションデータを暗号化、複合化、および認証し、Access 層へのメッセージの機密性を提供します。

- **Lower Transport 層**

  Lower Transport 層は Upper Transport 層のメッセージを分割して複数の Lower Transport 層の PDU に再構成します。

- **Network 層**

  Network 層は Lower Transport 層のメッセージを Bearer 層で転送できる Network PDU に加工します。また受信したメッセージを複合化して認証し、上位層に転送します。

- **Bearer 層**

  Bearer 層はノード間でのメッセージ授受方式を定義します。

1.2 モデル

Bluetooth Mesh モジュールは、Bluetooth-SIG が定義する以下のモデルをサポートします。

- **Foundation model**
  - Configuration
  - Health
- **Generic Models**
  - OnOff, Power OnOff, Power OnOff Setup
  - Level, Power Level, Power Level Setup
  - Default Transition Time
  - Battery
  - Location, Location Setup
  - Manufacturer Property, Admin Property, User Property, Client Property
- **Sensor Model**
  - Sensor, Sensor Setup
- **Time Model**
- **Scene Model**
  - Scene, Scene Setup
- **Scheduler Model**
  - Scheduler, Scheduler Setup
- **Light Models**
  - Light Lightness, Light Lightness Setup
  - Light CTL, Light CTL Setup
  - Light HSL, Light HSL Setup
  - Light XYL, Light XYL Setup
  - Light Control

FSP Configuration 画面において、必要なモデルを追加することができます。この時、Bluetooth Mesh Networking 仕様で Foundation Model として定義されている Health サーバモデルと、Configuration サーバモデルは必ず追加してください。

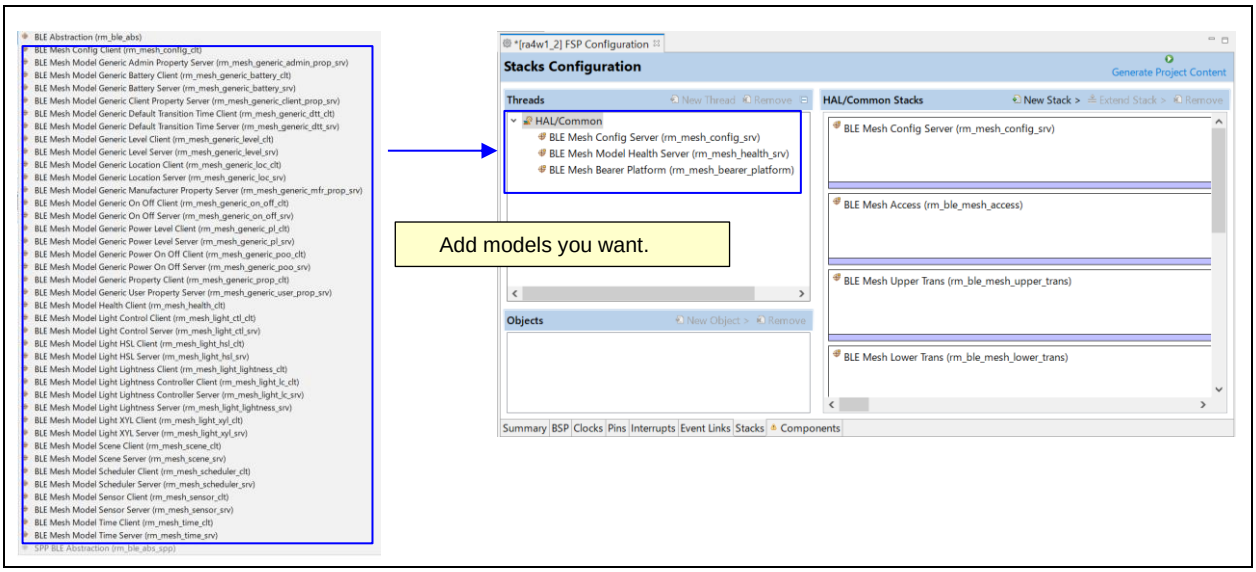

図 2 Model の追加

1.3 API

Bluetooth Mesh モジュールは、ユーザアプリケーションで使用可能な API を提供します。各 API の詳細は *Renesas Flexible Software Package User's Manual* を参照ください。

表 1 Mesh API

Category	Module	Number of APIs
Mesh Core	rm_ble_mesh	8
	rm_ble_mesh_access	76
	rm_ble_mesh_upper_trans	19
	rm_ble_mesh_lower_trans	7
	rm_ble_mesh_network	14
	rm_ble_mesh_bearer	15
	rm_ble_mesh_provisioning	9
	rm_ble_mesh_bearer_platform	12
Mesh Model	rm_mesh_config_client/server	54
	rm_mesh_health_client/server	15
	rm_mesh_generic_*	122
	rm_mesh_light_*_client/server	107
	rm_mesh_scene_client/server	16
	rm_mesh_scheduler_client/server	12
	rm_mesh_sensor_client/server	20
	rm_mesh_time_client/sever	16

2. アプリケーションノート

Renesas web サイトの RA4W1 ページには、Bluetooth Mesh のためのアプリケーションノートとして下記が公開されています。

表 2 アプリケーションノート

名称	APN 番号	用途
Bluetooth Mesh スタートアップガイド	R01AN5847	• R01AN5848 で提供する Mesh サンプルアプリケーションの動作方法 • Mesh モバイルアプリケーションを同梱
Bluetooth Mesh サンプルアプリケーション	R01AN5848	下記ソースコードを同梱 Mesh サンプルアプリケーションプロジェクト
Bluetooth Mesh 開発ガイド	R01AN5849	Bluetooth Mesh 仕様概要 Mesh アプリケーション作成方法

Bluetooth Mesh サンプルアプリケーション(R01AN5848)には、EK-RA4W1 で動作する、以下のサンプルプロジェクトが同梱されています。

表 3 Bluetooth Mesh サンプルアプリケーション(R01AN5848)に同梱するサンプルプロジェクト

名称	機能 / 用途
ekra4w1_mesh_server_baremetal	• Generic ON/OFF サーバモデル • Baremetal
ekra4w1_mesh_server_freertos	• Generic ON/OFF サーバモデル • FreeRTOS 上で動作
ekra4w1_mesh_client_baremetal	• Generic ON/OFF クライアントモデル • Baremetal
ekra4w1_mesh_client_freertos	• Generic ON/OFF クライアントモデル • FreeRTOS 上で動作
Mesh mobile application	• Provisioner，Configuration クライアント，Generic ON/OFF クライアントモデルとして動作するスマーフォンアプリケーション。 • Android 向けには apk ファイルを同梱。 • iOS 向けには MAC OS 上でビルド可能なソースコードを添付。ビルド環境は Bluetooth Mesh スタートアップガイド(R01AN5847)参照。
ekra4w1_mesh_cli_server_baremetal	• Bluetooth Mesh のサーバ側のネットワーク構築，メッセージ授受に必要な手順を EK-RA4W1 に接続したターミナルエミュレータからコマンドを入力することで実行します。

名称	機能 / 用途
ekra4w1_mesh_cli_client_baremetal	• Bluetooth Mesh のクライアント側のネットワーク構築, メッセージ授受に必要な手順を EK-RA4W1 に接続したターミナルエミュレータからコマンドを入力することで実行します。

Bluetooth Mesh サンプルアプリケーション(R01AN5848)に同梱するサーバ、クライアント、Mesh mobile application を用いてデバイスの Mesh ネットワーク参加、メッセージ授受、Mesh ネットワークからの除外といったデバイスの Mesh ネットワークにおけるライフサイクルの動作を確認することができます。

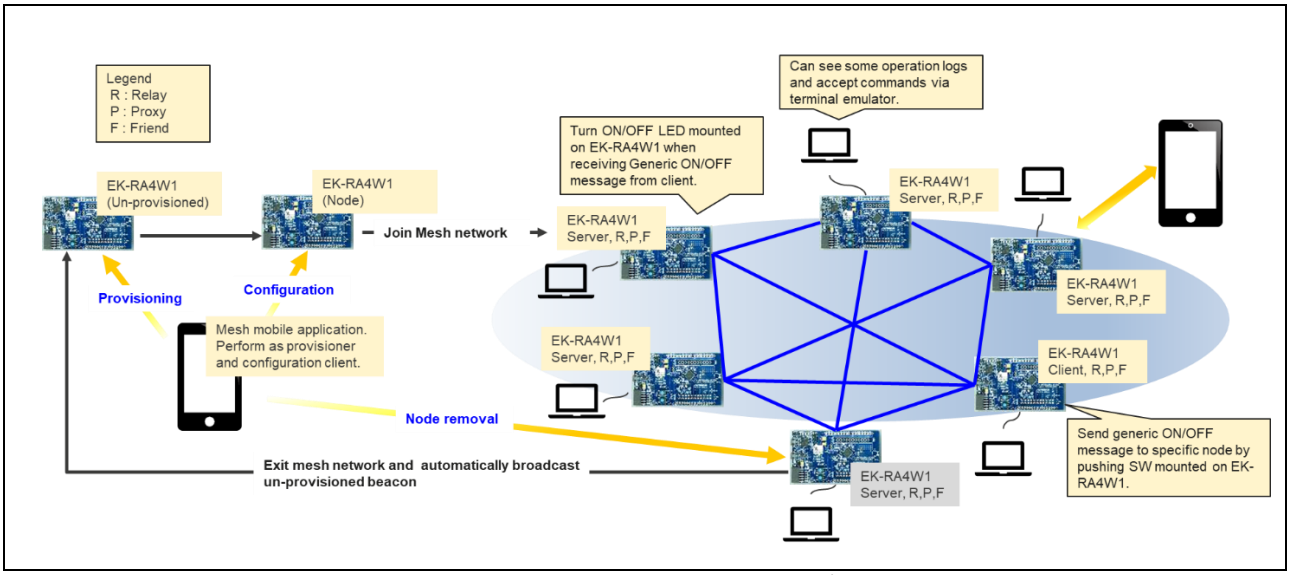

図 3 Bluetooth Mesh サンプル

3. ユーザアプリケーション作成例

Bluetooth Mesh アプリケーション作成にあたっては、Bluetooth Mesh スタートアップガイド(R01AN5847)に基づいて、Bluetooth Mesh サンプルアプリケーション(R01AN5848)同梱のサンプルプロジェクトを開発環境にインポートし、必要な機能を追加することをお勧めします。本章では、Bluetooth Mesh サンプルアプリケーション(R01AN5848)同梱のサンプルプロジェクトに Generic Battery Model を追加し、クライアントから任意のタイミングでサーバの Battery Status を取得する機能を追加します。

- **追加する機能**
  - **Server 側**
    - Generic battery Server モデルを追加する。
    - Client からの要求に応じて、ダミーの値を含む Battery Level State を返送する。
  - **Client 側**
    - Generic battery Client モデルを追加する。
    - SW1 を押下すると、Battery Level Status を Server から取得する。
    - Server から受信した Battery Level Status をターミナルエミュレータに表示する。

3.1 プロジェクトのインポート

Bluetooth Mesh アプリケーション作成にあたっては、Bluetooth Mesh スタートアップガイド(R01AN5847)に基づいて、Bluetooth Mesh サンプルアプリケーション(R01AN5848)同梱の下記サンプルプロジェクトを開発環境にインポートしてください。

- **Baremetal 環境**
  - ekra4w1_mesh_server_baremetal (サーバ側)
  - ekra4w1_mesh_client_baremetal (クライアント側)
- **FreeRTOS 環境**
  - ekra4w1_mesh_server_freertos (サーバ側)
  - ekra4w1_mesh_client_freertos (クライアント側)

3.2 Generic Battery Model の Open

Bluetooth Mesh サンプルアプリケーション(R01AN5848)同梱のサンプルプロジェクトは Generic Battery モデルは追加済です。*./src/mesh_model.c/mesh_application_model_register()*関数にて、下記の様に Generic Battery モデルを Open してください。

```
static API_RESULT mesh_application_model_register(void)
{
    API_RESULT retval = API_SUCCESS;
.........
  g_rm_mesh_generic_battery_srv0.p_api->open(g_rm_mesh_generic_battery_srv0.p_ctrl,
                                             g_rm_mesh_generic_battery_srv0.p_cfg);
return retval;
}
```

Code 1　サーバ側実装例

```
static API_RESULT mesh_application_model_register(void)
{
    API_RESULT retval = API_SUCCESS;
.........
  g_rm_mesh_generic_battery_clt0.p_api->open(g_rm_mesh_generic_battery_clt0.p_ctrl,
                                             g_rm_mesh_generic_battery_clt0.p_cfg);
return retval;
}
```

Code 2　クライアント側実装例

3.3 Generic Battery Model の Server 側 Callback 関数実装

サーバ側のサンプルアプリケーションに、クライアントからの GET メッセージを受け取った時の挙動を追加します。Bluetooth Mesh サンプルアプリケーション(R01AN5848)同梱のサンプルプロジェクトでは Generic Battery Model の Callback 関数は *rm_mesh_generic_battery_srv* モジュールのプロパティにおいて、*appl_generic_battery_server_cb* 関数に指定されています。本文の例では、サーバ側にクライアントからの *GET* メッセージへ固定値 0xAA を Battery status として返送する処理を以下の様に追加します。

```
void appl_generic_battery_server_cb(rm_ble_mesh_model_server_callback_args_t * p_args)
{
    rm_mesh_generic_battery_srv_info_t      param;
.........
    /* Check message type */
    if (RM_BLE_MESH_ACCESS_MODEL_REQ_MSG_TYPE_GET == p_args->p_req_type->type)
    {
        CONSOLE_OUT(
        "[GENERIC_BATTERY] GET Request.¥n");
        appl_model_state_get(p_args->p_state_parameter->state_type, (uint16_t)elem_handle,
                                &param, 0);
        param.generic_battery_level = 0xAA;

        current_state_params.state_type = p_args->p_state_parameter->state_type;
        current_state_params.state = &param;
.........
    return retval;
}
```

Always respond “0xAA” as current battery level against GET message from client.

Code 3 実装例

3.4 GET メッセージの送信

クライアント側のサンプルアプリケーションに SW1 の押下により GET メッセージを送信する機能を追加します。Bluetooth Mesh サンプルアプリケーション(R01AN5848)同梱のサンプルプロジェクトは SW1 の押下により *Callback_ble_sw_irq* 関数が呼び出されます。クライアント側の同コールバック関数に、以下のコードを追加し、SW1 の押下によって *GET* メッセージを送信します。

```
static void board_sw1_cb(void)
{
 .........
gs_sw_ignore = false;
    RM_MESH_GENERIC_BATTERY_CLT_Get(&g_rm_mesh_generic_battery_clt0_ctrl);
}

void Callback_ble_sw_irq(external_irq_callback_args_t *p_args)
{
     FSP_PARAMETER_NOT_USED(p_args);
  .........
     R_BLE_SetEvent(board_sw1_cb);
}
```

Code 4 GET メッセージの送信実装例

3.5 Generic Battery Model の Client 側 Callback 関数実装

クライアント側のサンプルアプリケーションに GET メッセージに対する応答を受け取った時の挙動を追加します。Bluetooth Mesh サンプルアプリケーション(R01AN5848)同梱のサンプルプロジェクトでは Generic Battery Model の Callback 関数は *rm_mesh_generic_battery_clt* モジュールのプロパティにおいて、*cli_generic_battery_client_cb* 関数に指定されています。本文の例では、サーバから受け取った Battery State をターミナルエミュレータに表示する処理を追加します。

```
void cli_generic_battery_client_cb(rm_ble_mesh_model_client_callback_args_t * p_args)
{
 .........
    switch(p_args->p_msg_raw->opcode)
    {
        case RM_BLE_MESH_ACCESS_MESSAGE_OPCODE_GENERIC_BATTERY_STATUS:
            CONSOLE_OUT( "RM_BLE_MESH_ACCESS_MESSAGE_OPCODE_GENERIC_BATTERY_STATUS n");
            appl_dump_data(p_args->p_msg_raw->data_param, sizeof(uint8_t));
      break;
    }
.........
}
```

Code 5 応答受信時の実装例

3.6 動作例

前節までの実装を追加し、Mesh スタートアップガイド(R01AN5847)の手順に基づいて Provisioning を実施すると、Mesh mobile application の INFORMATION タブに追加した Generic Battery Model が表示されます。

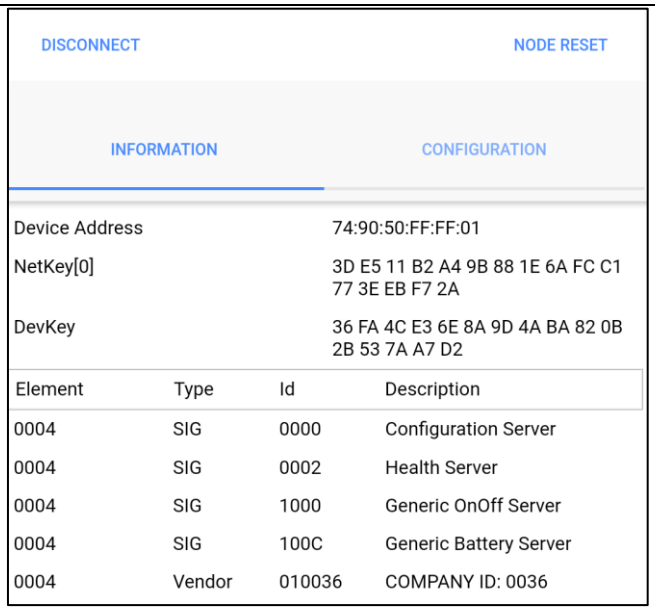

図 4 Mesh mobile application INFORMATION 画面

Mesh スタートアップガイド(R01AN5847)の手順をさらに進め、Configuration を完了し、Mesh ネットワーク構築完了後、Client 側の SW1 を押下すると、Client 側のターミナルエミュレータに、以下の様に Code 3 で追加した Battery Status が表示されます。

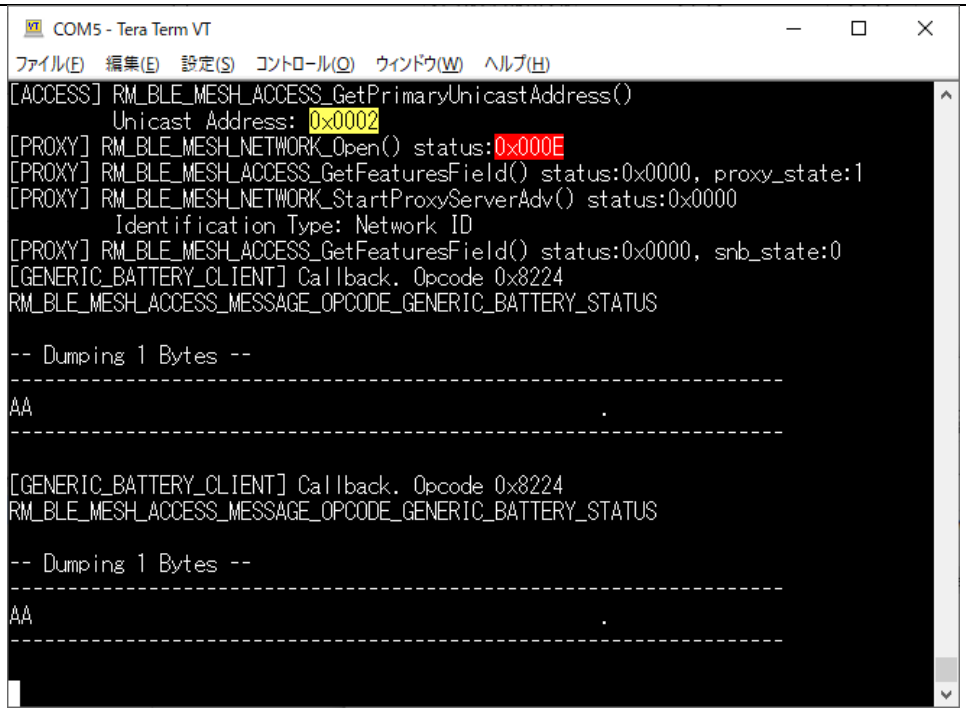

図 5 Client 側ターミナルエミュレータ画面

商標権および著作権

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。

RA4W1 グループ Bluetooth Mesh スタックは次のオープンソースソフトウェアを使用します。

crackle; AES-CCM, AES-128bit 機能

BSD 2-Clause License

Copyright (c) 2013-2018, Mike Ryan
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
  list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.0.0	2022.3.25	---	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うと、フラッシュメモリ、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

ご注意書き

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

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

   高品質水準：輸送機器（自動車、電車、船舶等）、交通制御（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等

   当社製品は、データシート等により高信頼性、Harsh environment 向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。
7. あらゆる半導体製品は、外部攻撃からの安全性を 100%保証されているわけではありません。当社ハードウェア／ソフトウェア製品にはセキュリティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害（当社製品または当社製品が使用されているシステムに対する不正アクセス・不正使用を含みますが、これに限りません。）から生じる責任を負うものではありません。当社は、当社製品または当社製品が使用されたあらゆるシステムが、不正な改変、攻撃、ウイルス、干渉、ハッキング、データの破壊または窃盗その他の不正な侵入行為（「脆弱性問題」といいます。）によって影響を受けないことを保証しません。当社は、脆弱性問題に起因しまたはこれに関連して生じた損害について、一切責任を負いません。また、法令において認められる限りにおいて、本資料および当社ハードウェア／ソフトウェア製品について、商品性および特定目的との合致に関する保証ならびに第三者の権利を侵害しないことの保証を含め、明示または黙示のいかなる保証も行いません。
8. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
10. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
11. 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
12. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものといたします。
13. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
14. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。

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

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

(Rev.5.0-1　2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

www.renesas.com

商標について

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

お問合せ窓口

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

www.renesas.com/contact/

© 2022 Renesas Electronics Corporation. All rights reserved.