



RN4020 Bluetooth[®] Low Energy モジュール ユーザガイド

注意：この日本語版文書は参考資料としてご利用ください。最新情報は必ずオリジナルの英語版をご参照願います。

Microchip 社製デバイスのコード保護機能に関して以下の点にご注意ください。

- Microchip 社製品は、該当する Microchip 社データシートに記載の仕様を満たしています。
- Microchip 社では、通常の条件ならびに仕様に従って使用した場合、Microchip 社製品のセキュリティ レベルは、現在市場に流通している同種製品の中でも最も高度であると考えています。
- しかし、コード保護機能を解除するための不正かつ違法な方法が存在する事もまた事実です。弊社の理解では、こうした手法は Microchip 社データシートにある動作仕様書以外の方法で Microchip 社製品を使用する事になります。このような行為は知的財産権の侵害に該当する可能性が非常に高いと言えます。
- Microchip 社は、コードの保全性に懸念を抱いているお客様と連携し、対応策に取り組んでいきます。
- Microchip 社を含む全ての半導体メーカーで、自社のコードのセキュリティを完全に保証できる企業はありません。コード保護機能とは、Microchip 社が製品を「解読不能」として保証するものではありません。

コード保護機能は常に進歩しています。Microchip 社では、常に製品のコード保護機能の改善に取り組んでいます。Microchip 社のコード保護機能の侵害は、デジタル ミレニアム著作権法に違反します。そのような行為によってソフトウェアまたはその他の著作物に不正なアクセスを受けた場合、デジタル ミレニアム著作権法の定めるところにより損害賠償訴訟を起こす権利があります。

本書に記載されているデバイス アプリケーション等に関する情報は、ユーザの便宜のためにのみ提供されているものであり、更新によって無効とされる事があります。お客様のアプリケーションが仕様を満たす事を保証する責任は、お客様にあります。Microchip 社は、明示的、暗黙的、書面、口頭、法定のいずれであるかを問わず、本書に記載されている情報に関して、状態、品質、性能、商品性、特定目的への適合性をはじめとする、いかなる類の表明も保証も行いません。Microchip 社は、本書の情報およびその使用に起因する一切の責任を否認します。生命維持装置あるいは生命安全用途に Microchip 社の製品を使用する事は全て購入者のリスクとし、また購入者はこれによって発生したあらゆる損害、クレーム、訴訟、費用に関して、Microchip 社は擁護され、免責され、損害を受けない事に同意するものとします。暗黙的あるいは明示的を問わず、Microchip 社が知的財産権を保有しているライセンスは一切譲渡されません。

商標

Microchip 社の名称とロゴ、Microchip ロゴ、dsPIC、FlashFlex、KEELOQ、KEELOQ ロゴ、MPLAB、PIC、PICmicro、PICSTART、PIC³² ロゴ、rPIC、SST、SST ロゴ、SuperFlash、UNI/O は、米国およびその他の国における Microchip Technology Incorporated の登録商標です。

FilterLab、Hampshire、HI-TECH C、Linear Active Thermistor、MTP、SEEVAl、Embedded Control Solutions Company は、米国における Microchip Technology Incorporated の登録商標です。

Silicon Storage Technology は、他の国における Microchip Technology Inc. の登録商標です。

Analog-for-the-Digital Age、Application Maestro、BodyCom、chipKIT、chipKIT ロゴ、CodeGuard、dsPICDEM、dsPICDEM.net、dsPICworks、dsSPEAK、ECAN、ECONOMONITOR、FanSense、HI-TIDE、In-Circuit Serial Programming、ICSP、Mindi、MiWi、MPASM、MPF、MPLAB Certified ロゴ、MPLIB、MPLINK、mTouch、Omniscient Code Generation、PICC、PICC-18、PICDEM、PICDEM.net、PICkit、PICtail、REAL ICE、rLAB、Select Mode、SQL、Serial Quad I/O、Total Endurance、TSHARC、UniWinDriver、WiperLock、ZENA および Z-Scale は、米国およびその他の Microchip Technology Incorporated の商標です。

SQTP は、米国における Microchip Technology Incorporated のサービスマークです。

GestIC および ULPP は、Microchip Technology Inc. の子会社である Microchip Technology Germany II GmbH & Co. & KG 社の他の国における登録商標です。

その他本書に記載されている商標は各社に帰属します。

© 2015, Microchip Technology Incorporated, All Rights Reserved.

ISBN: 978-1-63276-523-9

**QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
= ISO/TS 16949 =**

Microchip 社では、Chandler および Tempe (アリゾナ州)、Gresham (オレゴン州) の本部、設計部およびウェハー製造工場そしてカリフォルニア州とインドのデザインセンターが ISO/TS-16949:2009 認証を取得しています。Microchip 社の品質システム プロセス および手順は、PIC® MCU および dsPIC® DSC、KEELOQ® コードホッピング デバイス、シリアル EEPROM、マイクロペリフェラル、揮発性メモリ、アナログ製品に採用されています。さらに、開発システムの設計と製造に関する Microchip 社の品質システムは ISO 9001:2000 認証を取得しています。

宣言の対象 : RN4020 Bluetooth® Low Energy モジュール

EU Declaration of Conformity

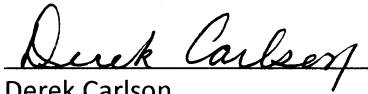
This declaration of conformity is issued by the manufacturer.

The development/evaluation tool is designed to be used for research and development in a laboratory environment. This development/evaluation tool is not a Finished Appliance, nor is it intended for incorporation into Finished Appliances that are made commercially available as single functional units to end users under EU EMC Directive 2004/108/EC and as supported by the European Commission's Guide for the EMC Directive 2004/108/EC (8th February 2010).

This development/evaluation tool complies with EU RoHS2 Directive 2011/65/EU.

For information regarding the exclusive, limited warranties applicable to Microchip products, please see Microchip's standard terms and conditions of sale, which are printed on our sales documentation and available at www.microchip.com.

Signed for and on behalf of Microchip Technology Inc. at Chandler, Arizona, USA



Derek Carlson

VP Development Tools

16-July-2013

Date

RN4020 Bluetooth Low Energy モジュール ユーザガイド

NOTES:

目次

序章	7
Chapter 1. はじめに	
1.1 Bluetooth Low Energy の基礎	13
Chapter 2. RN4020 OEM モジュール インターフェイス	
2.1 RN4020 の制御ライン	15
2.2 RN4020 の UART 制御インターフェイス	19
2.3 デバイス ファームウェア アップグレード	59
Chapter 3. アプリケーション例	
3.1 スマートデバイスとの接続によるデモ	61
3.2 2つの RN4020 モジュール同士の接続によるデモ	71
3.3 MLDP のデモ	74
3.4 RN4020 のスクリプト機能のデモ	75
補遺 A. PICtail™ ドータボード回路図	77
各国の営業所とサービス	84

序章

お客様へのご注意

どのような文書でも内容は時間と共に古くなります。本書も例外ではありません。Microchip 社の製品は、お客様のニーズを満たすために常に改良を重ねており、実際のダイアログやツールが本書の説明とは異なる場合があります。最新の文書は弊社ウェブサイト (www.microchip.com) でご覧になれます。

文書は「DS」番号で識別します。この識別番号は各ページのフッタのページ番号の前に表記しています。DS 番号の表記規則は「DSXXXXXXXXA」で、「XXXXXXXX」が文書番号、「A」が文書のリビジョンレベルを表しています。

開発ツールについての最新情報は、MPLAB® IDE のオンラインヘルプをご覧ください。[Help] メニューから [Topics] を選択すると、オンラインヘルプ ファイルのリストが表示されます。

はじめに

序章には、RN4020 Bluetooth® Low Energy モジュールを使い始める前に知っておくと便利な一般情報を記載しています。主な内容は以下の通りです。

- 本書の構成
- 本書の表記規則
- 推奨参考資料
- Microchip 社のウェブサイト
- 開発システムの顧客変更通知サービス
- カスタマサポート
- 改訂履歴

本書の構成

本書では、RN4020 Bluetooth® Low Energy モジュールを開発ツールとして使って、ターゲットボード上のファームウェアをエミュレートおよびデバッグする方法について説明します。本書は下記の章によって構成されます。

- **Chapter 1. 「はじめに」**には、RN4020 の特長と使い方等の概要を記載しています。
- **Chapter 2. 「RN4020 OEM モジュール インターフェイス」**には、本モジュールのインターフェイスに関する情報を記載しています。
- **Chapter 3. 「アプリケーション例」**には、RN4020 の特長に重点を置いた応用例を挙げています。
- **補遺 A. 「PICtail™ ドータボード回路図」**には、PICtail ドータボードの回路図を記載しています。

RN4020 Bluetooth Low Energy モジュール ユーザガイド

本書の表記規則

本書には下記の表記規則を適用します。

本書の表記規則

表記	意味	例
二重かぎカッコ:『』	参考資料	『MPLAB® IDE ユーザガイド』
	テキストの強調	... は唯一のコンパイラです ...
角カッコ:[]	ウィンドウ名	[Output] ウィンドウ
	ダイアログ名	[Settings] ダイアログ
	メニューの選択肢	[Enable Programmer] を選択
かぎカッコ:「」	ウィンドウまたはダイアログのフィールド名	「Save project before build」
右山カッコ (>) で区切った下線付きテキスト	メニュー項目の選択	[File]>[Save]
角カッコで囲んだ太字のテキスト	ダイアログのボタン	[OK] をクリックする
	タブ	[Power] タブをクリックする
山カッコで囲まれたテキスト:<>	キーボードのキー	<Enter>、<F1> を押す
標準書体の Courier New	サンプル ソースコード	#define START
	ファイル名	autoexec.bat
	ファイルパス	c:\mcc18\h
	キーワード	_asm, _endasm, static
	コマンドライン オプション	-Opa+, -Opa-
	ビット値	0, 1
	定数	0xFF, 'A'
斜体の Courier New	変数の引数	file.o (file は有効な任意のファイル名)
角カッコ:[]	オプションの引数	mcc18 [options] file [options]
中カッコとパイプ文字: { }	いずれかの引数を選択する場合 (OR 選択)	errorlevel {0 1}
...	繰り返されるテキスト	var_name [, var_name...]
	ユーザが定義するコード	void main (void) { ... }
Note	特に間違いやすい箇所や、デバイスの種類によって動作が異なる箇所について、読者の注意を促すための情報を記載します。枠に囲んで本文中に表記する場合と、図または表の下に表記する場合があります。	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> <p>Note: これは標準的な Note ボックスです。</p> </div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> <p style="text-align: center;">CAUTION</p> <p>これは警告を示す Note です。</p> </div> <p>Note1: これは表内のNoteです。</p>

推奨参考資料

補足資料として、以下の文書を推奨します。

RN4020 ファミリ データシート (DS50002279)

RN4020 Bluetooth® Low Energy モジュールの詳細は、この文書を参照してください。このデータシートには下記の情報を記載しています。

- デバイスのピン配置とパッケージの詳細
- デバイスの電氣的仕様
- 内蔵機能のリスト

この文書は Microchip 社のウェブサイト (www.microchip.com) からダウンロードできます。

Bluetooth コア仕様 v4.0 (2010 年 6 月 30 日)

この仕様書は www.bluetooth.org からダウンロードできます。

Bluetooth コア仕様 v4.1 (2013 年 12 月 3 日)

Bluetooth® コア仕様 4.1 は、Bluetooth コア仕様の重要なアップデートです。Bluetooth コア仕様補遺 (CSA 1、2、3、4) を盛り込むと共に新機能を追加しています。Bluetooth 4.1 は、コンシューマにとっての使い勝手と製品開発者にとっての革新の容易さを高め、Internet of Things (IoT) の実現に必要な不可欠なリンクとしての技術的基盤を拡張します。

この仕様書は www.bluetooth.org からダウンロードできます。

Microchip 社のウェブサイト

Microchip 社は、自社が運営するウェブサイト (www.microchip.com) を通してオンラインサポートを提供しています。このウェブサイトからは、お客様に必要なファイルと情報を容易に入手できます。以下の情報を提供しています。

- **製品サポート** - データシートとエラッタ、アプリケーションノート、サンプルコード、設計リソース、ユーザガイドとハードウェアサポート文書、最新ソフトウェアリリース、ソフトウェアアーカイブ
- **一般的技術サポート** - よく寄せられる質問 (FAQ)、技術サポートのご依頼、オンラインディスカッショングループ、Microchip 社のコンサルタントプログラムおよびメンバーリスト
- **ご注文とお問い合わせ** - 製品セレクトと注文のガイド、最新プレスリリース、セミナー/イベントの一覧、お問い合わせ先 (営業所、販売代理店) の一覧

開発システムの顧客変更通知サービス

Microchip 社の顧客変更通知サービスは、お客様に Microchip 社製品の最新情報をお届けする配信サービスです。ご興味のある製品ファミリーまたは開発ツールに関連する変更、更新、エラッタ情報をいち早くメールでお知らせします。

Microchip 社のウェブサイト (www.microchip.com) にアクセスし、[Customer Change Notification] からご登録ください。

以下の開発システム製品カテゴリをお選び頂けます。

- **コンパイラ** - Microchip 社の C コンパイラとその他の言語ツール
- **エミュレータ** - Microchip 社製インサーキットエミュレータ MPLAB[®] REAL ICE[™]
- **インサーキットデバッガ** - Microchip 社製インサーキットデバッガ MPLAB ICD 3
- **MPLAB X IDE** - Microchip 社の MPLAB X IDE (開発システムツール向け Windows[®] 統合開発環境)
- **プログラマ** - Microchip 社のプログラマ (PICkit[®] 3 開発用プログラマを含む)

カスタマサポート

Microchip 社製品をお使いのお客様は、以下のチャンネルからサポートをご利用頂けます。

- 販売代理店
- 各地の営業所
- 技術サポート

サポートは販売代理店までお問い合わせください。各地の営業所もご利用いただけます。本書の末尾には各国営業所の一覧を記載しています。

技術サポートは下記のウェブサイトからもご利用頂けます：
<http://support.microchip.com>

改訂履歴

リビジョン A (2014 年 6 月)

本書の初版

RN4020 Bluetooth Low Energy モジュール ユーザガイド

NOTES:

Chapter 1. はじめに

この章では、RN4020 モジュールの概要および下記の項目について説明します。

• Bluetooth Low Energy の基礎

RN4020 Bluetooth[®] Low Energy モジュールは、Bluetooth コア仕様 v4.1 に準拠したシングルモード Bluetooth Smart モジュールです。

このモジュールを接続した時にセントラルとペリフェラルどちらのロールとして動作するかは、モジュールの高速UARTインターフェイスで設定できます。このモジュールは Bluetooth SIG (Special Interest Group) が承認済みの 13 のパブリック プロファイルと 17 のパブリック サービスをサポートしています。

サポートされる全てのプロファイルとサービスに関し、RN4020 モジュールは同時にサーバとクライアント両方のロールとして動作するように設定できます。さらに、RN4020 モジュールはプライベート MLDP (Microchip Low-energy Data Profile) をサポートしており、2つの RN4020 デバイス間で非同期シリアルデータ接続が可能です。

最後に、Microchip 社の RN4020 モジュールはユーザ定義のプライベート プロファイル/サービスもサポートしており、個々のユーザ アプリケーションに合わせた最適な設計が可能です。設定は全て内蔵不揮発性メモリ (NVM) に保存されるため、ユーザによるモジュールのセットアップは 1 回で済みます。

Microchip 社の RN4020 モジュールは使いやすく、BTLE テクノロジを短期間で製品化できる柔軟で強力なソリューションを提供します。

1.1 Bluetooth Low Energy の基礎

2つの BTLE デバイスを接続する場合、1つがセントラル、もう1つがペリフェラルとして動作します。ペリフェラルは自身の接続ステータスをアドバタイズし、セントラル デバイスが接続プロセスを開始します。接続が完了したら、どちらの側からもボンディングを開始できます。ボンディングが完了すると、セキュリティに関する全ての鍵が保存され、再接続時にはセキュリティ プロセスが省略されます。ボンディング済みのペリフェラル デバイスはダイレクト アドバタイズしか実行できないため、ボンディング済みのデバイス以外とは接続できません。

Bluetooth Classic 同様、BTLE もプロファイルを用いてデバイス間の相互動作を確保しています。しかし Bluetooth Classic とは異なり、BTLE のプロファイルはサービスの集まりです。BTLE のサービスは全て GATT (Generic Attribute Profile) に基づいて作成されており、この GATT がキャラクタースティックと呼ばれる属性へのアクセス方法を定義しています。従って BTLE プロファイルの主な機能は、これらのキャラクタースティックを中心に構築されます。サービスのキャラクタースティック値を保持するデバイスをサービスの「サーバ」と呼びます。反対に、ピアからデータを取得するデバイスを「クライアント」と呼びます。

各サービスとそのキャラクタースティックは、それぞれの UUID (Universally Unique Identifier) で識別します。UUID には短い形式 (16 ビット) のものと長い形式 (128 ビット) のものがあります。Bluetooth SIG が承認済みのサービスとキャラクタースティックは全て短い UUID を使います。一方、ユーザ定義のプライベート UUID は長い形式を使う必要があります。Bluetooth SIG が承認済みのサービスとキャラクタースティックの詳細は、Bluetooth Developer Portal (<https://developer.bluetooth.org/gatt/profiles/Pages/ProfilesHome.aspx>) を参照してください。

各キャラクタースティックへのアクセス方法は、表 1-1 に示す 8 ビットのビットパターン形式のキャラクタースティック プロパティで定義します。

RN4020 Bluetooth Low Energy モジュール ユーザガイド

表 1-1: キャラクターリスティックのプロパティ

プロパティ	ビットパターン	説明
拡張プロパティ (1)	0`b10000000	追加のプロパティが利用可能
認証付き書き込み (1)	0`b01000000	クライアントからサーバへキャラクターリスティックを書き込み(認証あり)
インディケーション	0`b00100000	サーバからクライアントへキャラクターリスティック値を報告 (ACK あり)
ノーティフィケーション	0`b00010000	サーバからクライアントへキャラクターリスティック値を通知 (ACK なし)
書き込み	0`b00001000	クライアントからサーバへキャラクターリスティック値を書き込み (ACK あり)
応答なし書き込み	0`b00000100	クライアントからサーバへキャラクターリスティック値を書き込み (ACK なし)
読み出し	0`b00000010	キャラクターリスティック値を読み出し、値はサーバからクライアントへ送信
ブロードキャスト	0`b00000001	キャラクターリスティック値をブロードキャスト

Note 1: 現時点で RN4020 はこのプロパティをサポートしていません。

Chapter 2. RN4020 OEM モジュール インターフェイス

この章では、RN4020 モジュールのインターフェイスについて説明します。

RN4020 モジュールは、監督官庁からの完全な認証済み (注) Bluetooth Low Energy シングルモード OEM モジュールです。このモジュールは I/O ライン (物理的なデバイスピン) と UART インターフェイスを使って制御します。

本章では下記の項目について説明します。

- [RN4020 の制御ライン](#)
- [RN4020 の UART 制御インターフェイス](#)
- [デバイス ファームウェア アップグレード](#)

2.1 RN4020 の制御ライン

RN4020 モジュールは WAKE_SW(ピン 7)、CMD/MLDP(ピン 8)、WAKE_HW(ピン 15) を使ってモジュールを各種ステートへ遷移させ、3 本の出力ピンを使って現在のステータスを示します。

WAKE_SW(ピン 7) は、RN4020 の動作ステートの制御に使います。WAKE_SW を High にすると、モジュールが復帰してアクティブ モードになります。復帰時に「CMD」が UART に出力され、モジュールがコマンドモードであり UART からコマンドを受け取る準備ができた事を示します。反対に、WAKE_SW を Low にするとモジュールは「END」を UART に出力してコマンドモードを終了し、ディープスリープで動作します。UART の baud レートが 2400 bps の場合を除き、ディープスリープでは UART インターフェイスは応答しません。モジュールがディープスリープの間、MLDP_EV(ピン 11) は Low を出力します。

MLDP シリアル データ サービス ([セクション 2.2.5 「Microchip MLDP コマンド」](#) 参照) を使う場合、CMD/MLDP(ピン 8) は RN4020 の制御に使います。CMD/MLDP を High にして MLDP モードに移行すると、UART からのデータは全てデータ ストリームとしてピアデバイスに送信されます。MLDP モードを終了するには CMD/MLDP を Low にします。RN4020 モジュールは UART に「CMD」を出力してコマンドモードに戻ります。

WAKE_HW(ピン 15) を High にすると、RN4020 モジュールは休止モードから復帰します。起動から 5 秒以内に WAKE_HW を 3 サイクルフリップすると (WAKE_HW ピンを High、Low、High の順に遷移させる 1 フリップサイクルと見なします)、RN4020 モジュールは工場リセットを実行します。工場リセット実行時に WAKE_SW が High の場合は完全工場リセットを実行します。そうでない場合はデバイス名、プライベート サービス、スクリプトを保持する部分工場リセットを実行します。

RN4020 モジュールをピアデバイスに接続している間、CONNECTION LED(ピン 10) は High を出力します。接続していない場合、CONNECTION LED は Low を出力します。

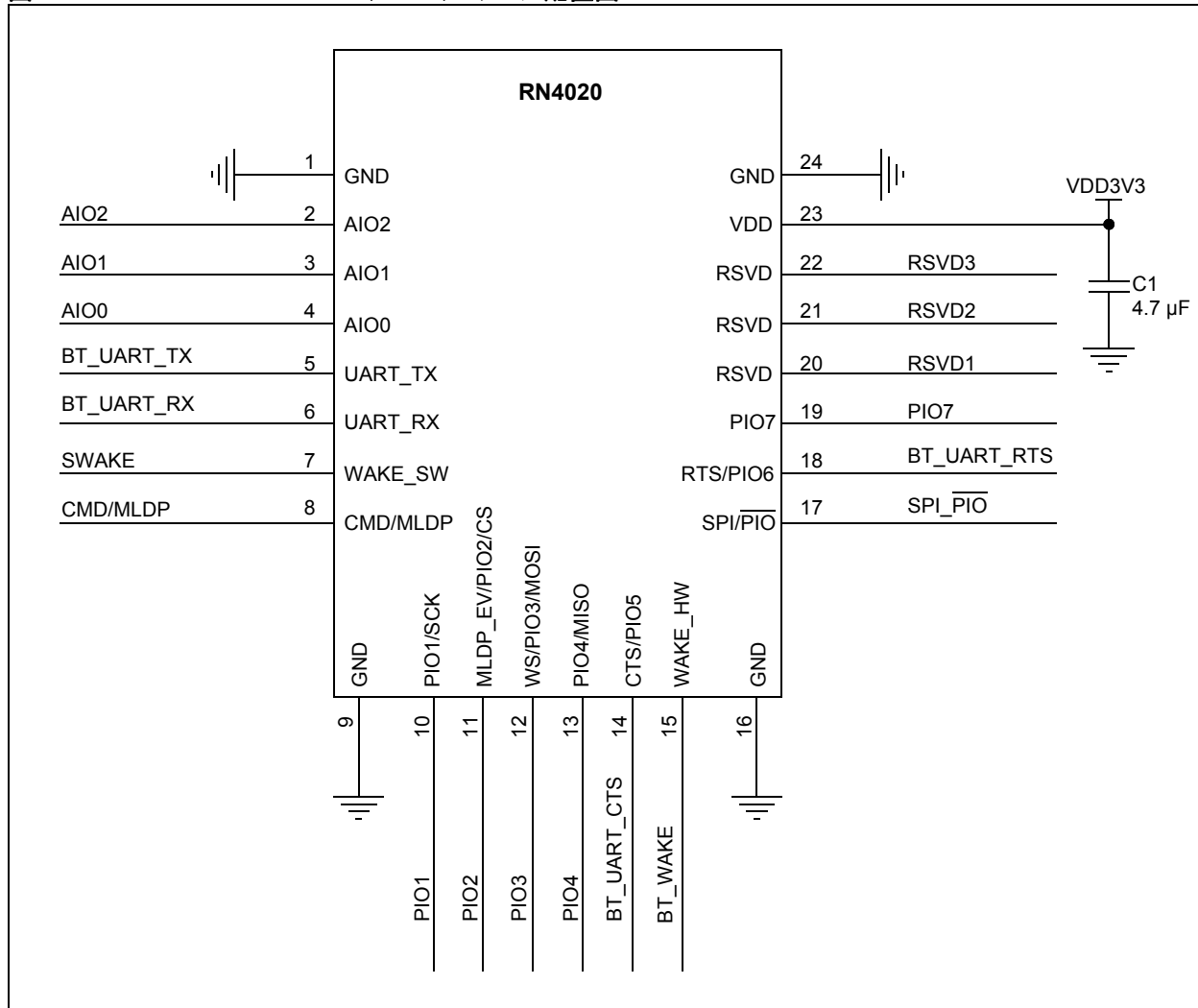
MLDP モードで RN4020 モジュールが UART にステータスを出力する必要がある場合、またはホスト MCU からの応答を要求している場合、MLDP_EV が High になります。RN4020 モジュールが MLDP モードを終了してコマンドモードに戻ると、ステータスと要求が UART に出力されます。保存されたデータが UART に出力されると、MLDP_EV は Low になります。ステータスと要求の最大バッファサイズは 256 バイトです。RN4020 モジュールがアクティブモードの時 WS(ピン 12) は High を出力し、それ以外の場合は Low を出力します。

注：日本国内電波法への対応に関しては弊社代理店にお問い合わせください。

RN4020 Bluetooth Low Energy モジュール ユーザガイド

図 2-1 に RN4020 モジュールのピン配置図、表 2-1 にピンの説明を示します。詳細は、『RN4020 Bluetooth Low Energy モジュール データシート』(DS50002279)を参照してください。

図 2-1: RN4020 モジュールのピン配置図



RN4020 OEM モジュール インターフェイス

表 2-1: RN4020 モジュールのピンの説明

ピン	記号	説明	機能
1	GND	グラウンド	グラウンド
2	AIO2	プログラマブル アナログ付き双方向 I/O	入力 1.65 V、出力 1.35 V、最大出力電流 30 mA
3	AIO1	プログラマブル アナログ付き双方向 I/O	入力 1.65 V、出力 1.35 V、最大出力電流 30 mA
4	AIO0	プログラマブル アナログ付き双方向 I/O	入力 1.65 V、出力 1.35 V、最大出力電流 30 mA
5	UART_TX	UART 送信 (TX)	RN4020 からの出力
6	UART_RX	UART 受信 (RX)	RN4020 への入力
7	WAKE_SW	ディープスリープからの復帰: アクティブ HIGH でモジュールをディープスリープから復帰	入力: 弱プルダウン
8	CMD/MLDP FACTORY RESET	<p>コマンドモードまたは MLDP モード - コマンドモードでは UART トラフィックはコマンドインタプリタに送られる。MLDP モードで MLDP Bluetooth LED 接続がアクティブの場合、UART トラフィックはこの接続に送られる。起動時5秒以内に WAKE_HW ピンを3回フリップすると、工場リセットが実行される。WAKE_SW が LOW(既定値)の場合、「SF,1」コマンド実行時と同様に部分工場リセットが実行される。</p> <p>WAKE_SW (ピン 7) が HIGH で起動時にピン 8 が HIGH の場合、「SF,2」コマンド実行時と同様に完全工場リセットが実行される。</p> <div style="border: 1px solid black; padding: 5px; text-align: center;"> <p>CAUTION</p> <p>完全工場リセットを実行すると、スクリプトが 消去されデバイス名がシリアル化された名称に 設定されます。詳細は SF,<1,2> コマンドを参 照してください。</p> </div>	入力: アクティブ HIGH でコマンドを入力
9	GND	グラウンド	グラウンド
10	CONNECTION LED PIO[1] SCK	既定値の状態は出力: アクティブ HIGH はモジュールがリモートデバイスに接続されている事を示す。アクティブ LOW は接続が切れた状態を示す。ソフトウェア コマンドによって PIO[1] として設定できる。ピン 17 がアサートされた時に診断と工場校正に使う SCK である。	<ul style="list-style-type: none"> • 緑色 LED • PIO[1] • SCK
11	MLDP_EV PIO[2] CS	既定値の機能は MLDP データイベントインジケータ(赤色LED)に対して使われる出力である。アクティブ HIGH は、MLDP データが受信されたか、UART コンソールデータが保留中かのどちらかを示す。LOW レベルはイベントがない事を示す。イベントは、CMD/MLDP (ピン 8) が HIGH の CMD モードでのみトリガーされる。「 >」および「 <」コマンドで PIO[2] として設定できる。ピン 17 がアサートされた時に診断と工場校正に使う CS である。	<ul style="list-style-type: none"> • MLDP データイベント (赤色 LED) • PIO[2] • CS
12	WS PIO[3] MOSI	既定値の機能はアクティビティインジケータ(青色LED)に対して使われる出力である。HIGH レベルは、モジュールが停止状態から解除されてアクティブである事を示す。LOW レベルはモジュールがスリープ状態にある事を示す。「 >」および「 <」コマンドで PIO[3] としてアクセスできる。ピン 17 がアサートされた時に診断と工場校正に使う MOSI である。	<ul style="list-style-type: none"> • WS(青色 LED) • PIO[3] • MOSI

RN4020 Bluetooth Low Energy モジュール ユーザガイド

表 2-1: RN4020 モジュールのピンの説明 (続き)

ピン	記号	説明	機能
13	PIO[4] MISO	イベント @PIOH および @PIOL を生成するためのトリガピンである。 ピン 17 がアサートされた時に診断と工場校正に使う CS である。	<ul style="list-style-type: none"> • PIO[4] • MISO
14	CTS PIO[5]	UART 経由のハードウェア フロー制御の場合、CTS 用に予約済み	<ul style="list-style-type: none"> • CTS(入力) • PIO[5]
15	WAKE_HW	ハードウェアの休止状態からの復帰	アクティブ HIGH: 内部プルダウン
16	GND	グラウンド	グラウンド
17	SPI/ $\overline{\text{PIO}}$	ピン 10 ~ 13 用の SPI/PIO、アクティブ HIGH	内部プルダウン付き入力: ピン 10 ~ 13 で SPI を選択
18	PIO[6]	UART 経由のハードウェア フロー制御の場合、RTS 用に予約済み ハードウェア フロー制御が無効の場合、PIO[6] として設定可能	<ul style="list-style-type: none"> • RTS(出力) • PIO[6]
19	PIO[7]	予備の PIO 「 0」 および 「 1」 コマンドの詳細は、 セクション 2.2.2 「アクション コマンド」 を参照してください。	入力または出力として設定可能な予備の PIO
20	RSVD	接続しない。工場での診断用	未接続
21	RSVD	接続しない。工場での診断用	未接続
22	RSVD	接続しない。工場での診断用	未接続
23	VDD	電源電圧	1.8 ~ 3.6 V
24	GND	グラウンド	グラウンド

2.2 RN4020 の UART 制御インターフェイス

UART は RN4020 モジュールの主制御インターフェイスです。表 2-2 に、UART ポートの既定値設定を示します。

表 2-2: RN4020 の UART 設定

パラメータ	値
baud レート	115200
データビット	8
パリティ	なし
ストップビット	1
フロー制御	なし

UART の baud レートは、「SB」コマンドを使って 2400 bps ~ 932 Kbps のレンジで調整できます。UART の baud レートを 2400 に設定した場合、モジュールとの通信の前に WAKE_SW(ピン 7) を使ってモジュールを復帰させる必要はありません。

制御は全て ASCII コマンドとそのパラメータで行います。全てのコマンドとパラメータはカンマで区切ります。コマンドとパラメータの区切りにスペースは使えません。全てのコマンドは LF またはリターンで完了します。

全てのコマンドは以下の種類に分類されます。

- Set/Get コマンド
- アクションコマンド
- キャラクターリスティック アクセス コマンド
- プライベート サービスの設定コマンド
- Microchip MLDP コマンド
- RN4020 スクリプト実行コマンド
- リモートコマンド
- DFU コマンド

表 2-3 に、種類別に分類した全コマンドとその簡単な説明を示します。

表 2-3: コマンドの説明

種類	コマンド名	説明
Set/Get	S-	シリアル化した名称
	SB	UART baud レートを設定
	SDF	ファームウェア リビジョンを設定
	SDH	ハードウェア リビジョンを設定
	SDM	モデル名を設定
	SDN	メーカー名を設定
	SDR	ソフトウェア リビジョンを設定
	SDS	シリアル番号を設定
	SF	工場出荷時の既定値にリセット
	SM	タイマの時間を μ s で設定
	SN	デバイス名を設定
	SR	機能を設定
	SS	サーバサービスを設定
	ST	接続パラメータを設定

RN4020 Bluetooth Low Energy モジュール ユーザガイド

表 2-3: コマンドの説明 (続き)

種類	コマンド名	説明
アクション	+	エコー
	@O	アナログ信号を出力
	@I	アナログ信号を入力
	O	PIO の出力を設定
	I	PIO の入力を取得
	A	アドバタイズを開始
	B	ボンディング
	D	設定内容をダンプ出力
	E	接続を確立
	F	スキャン開始
	H	ヘルプ
	J	オブザーバロールの開始と終了
	K	切断
	M	ピアから RSSI を取得
	N	ブロードキャスト情報を入力
	O	休止状態へ移行
	R	再起動
	T	現在の接続パラメータを変更
	U	ボンディング解除
	V	ファームウェア バージョン
X	スキャン停止	
Y	アドバタイズ停止	
Z	接続停止	
サービス	LC	クライアント サービスをリスト表示
	LS	サーバサービスをリスト表示
	CHR	クライアント ハンドルから値を読み出し
	CHW	クライアント ハンドルの値を書き込み
	CURC	クライアント UUID の設定を読み出し
	CURV	クライアント UUID の値を読み出し
	CUWC	クライアント UUID のノーティフィケーション/インディケーションを開始
	CUWV	クライアント UUID へ値を書き込み
	SHR	サーバハンドルの値を読み出し
	SHW	サーバハンドルへ値を書き込み
	SUR	サーバ UUID の値を読み出し
	SUW	サーバ UUID へ値を書き込み
プライベート サービス	PC	プライベート キャラクタリスティックの UUID を設定
	PS	プライベート サービスの UUID を設定
	PZ	プライベート サービスをクリア
MLDP	SE	MLDP のセキュリティ モードを設定
	I	MLDP モードを開始
スクリプト機能	LW	スクリプトを表示
	WC	スクリプトをクリア
	WP	スクリプトを停止
	WR	スクリプトを実行
	WW	スクリプトを書き込み

表 2-3: コマンドの説明 (続き)

種類	コマンド名	説明
リモート	!	リモートコマンド モードを開始
DFU	~	デバイス ファームウェアを更新

2.2.1 Set/Get コマンド

このコマンドグループは、RN4020 モジュールの機能設定に使用します。Set コマンドは文字「S」で始まり、その後には 1 ~ 2 文字のコマンド識別子が付きます。Set コマンドはパラメータが必須で、コマンドとパラメータの間はカンマで区切ります。例 2-1 に、Set コマンドのフォーマットを示します。

例 2-1: SET コマンドのフォーマット

S	コマンド識別子	,	入力パラメータ
---	---------	---	---------

ほとんどの Set コマンドでは、新しい設定を有効にするには再起動が必要です。Set コマンドで設定した内容は RN4020 モジュール内蔵の不揮発性メモリ (NVM) に保存され、パワーサイクルまたはリセット後に復元されます。各 Set コマンドには、設定内容を UART に出力するための対応する Get コマンドがあります。Get コマンドのコマンド識別子は Set コマンドと同じですが、Get コマンドにはパラメータはありません。

S-,<string>

説明

このコマンドは、Bluetooth 用にシリアル化したデバイスの名称を設定します。<string> は最大 15 文字の英数字です。このコマンドは、指定した名前の最後に Bluetooth MAC アドレスの最後の 2 バイトを自動的に付加します。このように固有の数字を加える事により、重複のない名前を生成します。

既定値

なし

例

```
S-,MyDevice // Set device name to "MyDevice-ABCD"
```

SB,<0-7>

説明

このコマンドは、UART 通信の baud レートを設定します。入力パラメータは、2400 ~ 921K の baud レートを表す 0 ~ 7 の 1 桁の数字です (表 2-4 参照)。baud レートを 2400 に設定した場合、UART 通信のために WAKE_SW を High に駆動して RN4020 モジュールを復帰させる必要はありません。

表 2-4: UART BAUD レートの設定

設定値	baud レート	備考
0	2400	UART を 2400 bps に設定した場合、RN4020 モジュールはディープスリープのままかまいません。つまり、baud レートを 2400 bps に設定すると UART に常時アクセスできるため、UART へのアクセスのために WAKE_SW ラインを High に駆動して RN4020 モジュールを復帰させる必要はありません。
1	9600	-
2	19200	-
3	38400	-
4	115200	-
5	230400	-
6	460800	-
7	921600	-

SDF,<text>

説明

このコマンドは、デバイス情報サービスの Firmware Revision キャラクタリスティックの値を設定します。

デバイス情報サービスは、デバイスを識別するために使います。デバイス情報サービスのキャラクタリスティックはどれもほとんど変化しないため、これらの値を設定して NVM に保存できます。デバイス情報サービスのキャラクタリスティック値の最大サイズは全て 20 バイトです。

既定値

ファームウェアバージョンによる。

例

SDF,0.9

RN4020 OEM モジュール インターフェイス

SDH, <text>

説明

このコマンドは、デバイス情報サービスの Hardware Revision キャラクターシティックの値を設定します。

既定値

ハードウェア バージョンによる。

例

SDH, 2.1

SDM, <text>

説明

このコマンドは、デバイス情報サービスの Model キャラクターシティックの値を設定します。

既定値

RN4020

例

SDM, RN4020

SDN, <text>

説明

このコマンドは、デバイス情報サービスの Manufacturer Name キャラクターシティックの値を設定します。

既定値

Microchip

例

SDN, Microchip

SDR,<text>

説明

このコマンドは、デバイス情報サービスの Software Revision キャラクタリスティックの値を設定します。

既定値

ソフトウェアバージョンによる。

例

SDR,1.0

SDS,<text>

説明

このコマンドは、デバイス情報サービスの Serial Number キャラクタリスティックの値を設定します。

既定値

デバイスの MAC アドレス

例

SDS,12345678

SF,<1,2>

説明

このコマンドは、次回起動時に設定値を工場出荷時の既定値にリセットします。このコマンドのパラメータは「1」または「2」です。

入力パラメータが「1」の場合、設定値のほとんどが工場出荷時の既定値に戻りますが、デバイス名、デバイス情報、スクリプト、プライベート サービス等一部の設定はそのまま残ります。入力パラメータが「2」の場合、全てのパラメータが工場出荷時の既定値に戻ります。

既定値

なし

例

SF,1

RN4020 OEM モジュール インターフェイス

SM,<1-3>,<hex32>

説明

このコマンドは、アプリケーション タイマのうち 1 つの動作を開始します。第 1 パラメータは、どのタイマを開始するかを指定します。第 2 パラメータは、0x00000001 ~ 0x7FFFFFFF であればタイマのタイムアウト時間 (単位: μ s) を指定します。第 2 パラメータがこのレンジ外の場合、タイマを停止します。

既定値

なし

例

```
SM,1,000f4240 // Start Timer1 to expire in 1 second
SM,1,FFFFFFF // Stop Timer1 immediately
```

SN,<string>

説明

このコマンドはデバイスの名称を設定します。<string> は、最大 20 文字の英数字です。

既定値

なし

例

```
SN,MyDevice // Set the device name to "MyDevice"
```

RN4020 Bluetooth Low Energy モジュール ユーザガイド

SR, <hex32>

説明

このコマンドは、現在の RN4020 モジュールでサポートされる機能を設定します。入力パラメータは、サポートされる機能を示す 32 ビットのビットパターンです。機能を変更した後、変更を有効にするには再起動が必要です。表 2-5 に、機能のビットパターンを示します。

表 2-5: 機能のビットパターン

機能	ビットパターン	説明
Central	0x80000000	このビットをセットした場合、デバイスはセントラルとして接続を開始します。このビットをクリアした場合、デバイスはペリフェラルとしてアダプタイズを開始します。
Real-time Read	0x40000000	このビットをセットした場合、デバイスは UART 経由でホスト MCU に対して値を要求し、ホスト MCU はタイムリーに応答する必要があります。このビットをクリアした場合、デバイスは RN4020 の RAM に保存されているキャラクタリスティック値を読み出します。
Auto Advertise	0x20000000	この設定はペリフェラル デバイスにのみ適用されます。このビットをセットした場合、デバイスはパワーサイクル、再起動、切断後にアダプタイズを開始します。このビットをクリアした場合、デバイスはコマンドモードで UART からコマンド「A」を受信した後にアダプタイズを開始します。
Support MLDP	0x10000000	このビットをセットした場合、デバイスは Bluetooth LE 上でシリアルデータの非同期転送を行うプライベートサービス MLDP をサポートします。このビットをクリアした場合、MLDP は無効です。詳細は、 セクション 2.2.5「Microchip MLDP コマンド」 を参照してください。
Auto MLDP Disable	0x08000000	この設定は、MLDP が有効な場合のみ有効です。このビットをセットした場合、コマンドモードで UART からコマンド「I」を受信するか、CMD/MLDP(ピン 8)を High にするとデバイスは MLDP モードになります。このビットをクリアした場合、コマンド「I」または CMD/MLDP ピンによるだけでなく、ピアデバイスから MLDP データストリームを受信した場合もデバイスは MLDP モードになります。
No Direct Advertisement	0x04000000	この設定はペリフェラル デバイスの場合のみ有効です。このビットをセットした場合、ペリフェラルはボンディング済みであっても不特定多数に向けたアダプタイズを発行します。従って、アダプタイズ中はいつでも検出可能です。この設定は、iOS または Android 機器と接続する場合に役立ちます。
UART Flow Control	0x02000000	この設定は RN4020 モジュールの UART ポートの RTS/CTS ハードウェア フロー制御の制御に使用します。このビットをセットした場合、フロー制御が有効で、ホストは UART ハードウェア フロー制御機能をサポートする必要があります。MLDP を有効にした場合、フロー制御が必要です。
Run Script After Power On	0x01000000	この設定はスクリプト実行の制御に使用します。このビットをセットした場合、@PW_ON イベントが生成されスクリプト実行が起動後自動的に開始します。
予約済み	0x00800000	-

RN4020 OEM モジュール インターフェイス

表 2-5: 機能のビットパターン (続き)

機能	ビットパターン	説明
Enable Authentication	0x00400000	この設定は、接続時に認証を有効にして中間者 (MITM) 攻撃を防ぎます。認証を有効にした場合、I/O 機能としてキーボードとディスプレイのどちらかまたは両方を設定する必要があります。詳細は、Bluetooth コア仕様 v4.1 Vol 3, Part H, Section 2.3.5.1 「Selecting STK Generation Method」の Table 2.5 「Mapping of IO Capabilities to STK Generation Method」を参照してください。
Enable Remote Command	0x00200000	この設定は MLDP 機能が有効な場合のみ有効です。このビットをセットした場合、ローカルデバイスは MLDP データストリームを利用してリモートデバイスからのリモートコマンド受信、およびリモートデバイスへのコマンド出力送信が可能です。
Do not Save Bonding	0x00100000	このビットをセットした場合、ボンディング情報は NVM に保存されずボンディングは現在の接続に対してのみ有効です。
I/O Capabilities	0x000E0000	モジュールの I/O 機能です。Enable Authentication ビットをセットした場合のみ有効です。 <ul style="list-style-type: none"> • \b000 = ディスプレイのみ • \b001 = ディスプレイ Yes/No • \b010 = キーボードのみ • \b011 = 入力なし、出力なし • \b100 = キーボード ディスプレイ
Block Set Commands in Remote Command Mode	0x00010000	このビットをセットした場合、リモートコマンドモードで全ての「Set」コマンドが無効です。
Enable OTA	0x00008000	このビットをセットした場合、無線 (OTA) による DFU が有効です。それ以外の場合、無線による DFU はサポートされません。
iOS Mode	0x00004000	このビットをセットした場合、接続パラメータが Apple® Bluetooth Accessory Design Guidelines に適合しているかチェックされます。詳細は ST,<interval>,<latency>,<timeout> コマンドを参照してください。
Server Only	0x00002000	このビットをセットした場合、RN4020 モジュールはクライアントとして動作しません。接続の時間と消費電力を節約するため、接続確立後にサービス検出を行いません。
Enable UART in Script	0x00001000	このビットをセットした場合、スクリプト実行時に通常の UART 出力が許可されます。
Auto-enter MLDP Mode	0x00000800	このビットをセットした場合、Support MLDP ビットもセットされていると、RN4020 モジュールは接続完了後に自動的に MLDP モードになります。
MLDP without Status	0x00000400	このビットをセットした場合、「CMD」、「Connected」、「Connection End」等の追加のステータス文字列は UART に出力されません。

既定値

00000000

例

```
SR,20000000 // Set device as peripheral, and
// automatically start advertisement
```

RN4020 Bluetooth Low Energy モジュール ユーザガイド

SS, <hex32>

説明

このコマンドは、デバイスがサーバロールでサポートするサービスを設定します。入力パラメータは、サーバとしてサポートするサービスを示す32ビットのビットパターンです。サーバロールとしてのサービスをサポートするという事は、サポートされるサービス全てのキャラクタリスティック値をホスト MCU が供給する必要があり、クライアントから要求があればこれら値を提供する事を意味します。サービスのキャラクタリスティック値は「SSW」コマンドでサーバデータベースに書き込みます。サービスビットパターンを変更した場合、新しいサービスを有効にするにはデバイスの再起動が必要です。表 2-6 に、32 ビットのビットパターンを示します。

表 2-6: サービスのビットパターン

サービス	ビットパターン	サービスを使うプロファイル
Device Information	0x80000000	Blood Pressure、Cycling Speed Cadence、Glucose、Health Thermometer、Heart Rate、Running Speed Cadence
Battery	0x40000000	
Heart Rate	0x20000000	Heart Rate
Health Thermometer	0x10000000	Health Thermometer
Glucose	0x08000000	Glucose
Blood Pressure	0x04000000	Blood Pressure
Running Speed Cadence	0x02000000	Running Speed Cadence
Cycling Speed Cadence	0x01000000	Cycling Speed Cadence
Current Time	0x00800000	Time
Next DST Change	0x00400000	Time
Reference Time Update	0x00200000	Time
Link Loss	0x00100000	Proximity
Immediate Alert	0x00080000	Find Me、Proximity
TX Power	0x00040000	Proximity
Alert Notification	0x00020000	Alert Notification
Phone Alert Status	0x00010000	Phone Alert Status
Scan Parameters	0x00004000	Scan Parameters
ユーザ定義のプライベートサービス	0x00000001	ユーザ定義のプライベート プロファイル

既定値

80000000

例

```
SS,060000 // Support blood pressure and running speed
           // cadence as server role
```

RN4020 OEM モジュール インターフェイス

ST,<interval>,<latency>,<timeout>

説明

このコマンドは、次回以降の接続に使う初期接続パラメータを設定します。3つの入力パラメータは全て 16 進数フォーマットの 16 ビット値です。現在の接続パラメータを変更する方法は、アクションコマンド「T」を参照してください。

セントラル デバイスの場合、これらの接続パラメータを使ってペリフェラルとの接続を確立します。ペリフェラル デバイスの場合、新規接続を確立後にこれらの接続パラメータを使って接続更新を要求します。ペリフェラル デバイスからの接続更新を受け入れるかどうかはセントラル デバイスが決定します。

接続が確立されていない場合、対応する Get コマンド「GT」は「ST」コマンドで設定した希望の接続パラメータを返します。接続が確立している場合、「GT」コマンドに回答して実際の接続パラメータが表示されます。

Note: 全ての Set コマンドには対応する Get コマンドがあり、設定値の取得に役立ちます。詳細は[セクション 2.2.1 「Set/Get コマンド」](#)を参照してください。

接続パラメータ interval、latency、timeout はペリフェラル デバイスがセントラルとの通信頻度に関係しています。従って、これらのパラメータは消費電力に大きく影響します。表 2-7 に、これら 3 つのパラメータのレンジと関係を示します。

表 2-7: 接続パラメータ

パラメータ	レンジ	既定値	説明
interval	0x0006 ~ 0x0C80	0006	接続した 2 つのデバイス間の通信の時間間隔 (単位: 1.25 ms)
latency	0x0000 ~ 0x01F3 ただし以下の値未満: (timeout * 10 / interval * 1.25 - 1)	0000	ペリフェラルがセントラルと通信する必要のない接続イベントの連続回数
timeout	0x000A ~ 0x0C80	0064	何も通信がない場合に接続が失われたと判断する最大時間 (単位: 10 ms)

Apple iOS デバイスには、これらのパラメータに関して特別な要件があります。このため、iOS デバイスと接続する場合は「SR」コマンド (SR,<hex32> コマンド参照) で iOS Mode ビットをセットし、以下の制約を遵守する必要があります。

- interval ≥ 16
- latency ≤ 4
- timeout ≤ 600
- (interval + 16) * (latency + 1) < timeout * 8 / 3

既定値

0006,0000,0064

例

```
ST,0064,0002,0064 // Set the interval to 125 ms,  
// latency to 2, and time-out to 1 second
```

2.2.2 アクションコマンド

アクションコマンドグループは、主に機能の開始と重要な情報の表示に使用します。

+

説明

このコマンドは、ローカルエコーの ON/OFF を切り換えます。コマンドモードで「+」を送信すると、タイプした文字が全て出力にエコーバックします。もう一度「+」コマンドを送信すると、ローカルエコーが OFF になります。

既定値

OFF

例

```
+ // Turn on local echo
```

```
@O,<0-2>,<hex16>
```

```
@I,<0-2>
```

説明

「@O」コマンドはアナログポート出力電圧を設定し、「@I」コマンドは入力電圧を取得します。第1パラメータは0、1、2のいずれかでアナログポート番号を指定します。第2パラメータはアナログ出力にのみ使い、出力電圧を mV 単位で設定します。入出力電圧レンジは 0 ~ 1.3 V(有効レンジ 0x0000 ~ 0x0514)です。

アナログ信号を出力する場合、RN4020 モジュールはディープスリープで動作できません。この場合、ファームウェアが自動的に動作モードを Idle に調整します。コマンド @O,<0-2>,0000 を発行してアナログ出力を OFF にすると、ファームウェアは再び自動的に動作モードをディープスリープに戻します(利用可能な場合)。

既定値

なし

例

```
@O,1,03E8 // Set AI01 output voltage to be 1000 mV
```

RN4020 OEM モジュール インターフェイス

```
|O,<hex8>,<hex8>  
|I,<hex8>
```

説明

「|O」コマンドはデジタル I/O ピン (PIO1、PIO3、PIO7) の出力を設定し、「|I」コマンドはこれらの入力を取得します。第 1 入力パラメータはこのコマンドで設定する PIO を示すビットパターン、第 2 パラメータは設定するデジタル値を示します。表 2-8 に、ピンのビットパターンを示します。既定値では、PIO1 ~ PIO3 はステータス出力として使います。例えば、RN4020 PICtail™ ドータボードはこれらのステータス PIO を使ってインジケータ LED を駆動します (補遺 A. 「PICtail™ ドータボード回路図」参照)。第 1 パラメータの最初の 3 ビットをセットしてこれらのピンに読み出しまたは書き込みを実行するとステータスは出力されなくなり、ユーザがこれらのピンを完全に制御します。

表 2-8: 「|O」および「|I」コマンドのビットパターン

ビットパターン	PIO
0`b00000001	PIO1
0`b00000010	PIO2
0`b00000100	PIO3
0`b00001000	PIO7

既定値

なし

例

```
|O,07,05          // Set PIO1 and PIO3 output to be high and PIO2  
                  // output to be low  
  
|I,06            // Read states of PIO2 and PIO3. The result is a one  
                  // byte bitmap. If the result is 04, PIO2 is low  
                  // and PIO3 is high.
```

CAUTION

PIO1 ~ PIO3 にアクセスすると、ステータス インジケータ (青、緑、赤) としての既定値動作が無効になります。

A, <hex16>, <hex16>

説明

このコマンドを利用できるのは、ペリフェラルまたはブロードキャスト ロールとして動作するデバイスのみです。

「A」コマンドは、アドバタイズの開始に使用します。「N」コマンドによってデバイスがブロードキャスト ロールで動作している場合、アドバタイズは不特定多数に向けた接続不可のメーカー固有ブロードキャスト メッセージです。メッセージのペイロードは「N」コマンドで設定します。

デバイスがペリフェラル ロールとして動作しており、ボンディングされていない場合、アドバタイズは不特定多数に向けた接続可能なブロードキャスト メッセージであり、全ての BTLE セントラル デバイスによる検出が可能です。デバイスがボンディングされている場合、「SR」コマンドで no_direct_adv ビットがクリアされていればアドバタイズは特定相手にのみ向けられ、no_direct_adv ビットがセットされていればアドバタイズは不特定多数に向けられます。特定相手に向けたアドバタイズの場合、アドバタイズはボンディング済みデバイスにのみ向けられ、それ以外の BTLE デバイスには届きません。

「A」コマンドをパラメータなしで発行した場合、既定値では 100 ms の間隔で無期限にアドバタイズを実行します。「A」コマンドは 16 ビットのパラメータ (16 進数) を 2 つ取る事ができ、これによってアドバタイズ間隔と全体のアドバタイズ期間をどちらも ms 単位で指定します。第 2 パラメータは第 1 パラメータより大きい値にする必要があります。

既定値

100 ms

例

```
A,0050,07D0 // Start advertisement with interval of
              // 80 milliseconds for 2 seconds
```


RN4020 OEM モジュール インターフェイス

B, <0, 1>

説明

このコマンドは、接続にセキュリティを適用して2つの接続済みデバイスをボンディングするために使います。「B」コマンドは、2つのデバイスが既に接続されている場合のみ有効です。ボンディングはセントラルまたはペリフェラル デバイスのどちらからも発行できます。

入力パラメータを指定しない場合、または入力パラメータが「1」の場合、接続にセキュリティが適用されピアデバイスが記憶されます。この状況では、2つのデバイスがボンディング済みと見なされます。入力パラメータが「0」の場合、接続にセキュリティは適用されますがピアデバイスは NVM に保存されません。この状況では、接続はボンディングされていません。

「SR」コマンドで「do_not_save_bonding」ビットをクリアした場合、ボンディングが完了すると接続両端のデバイスの NVM にセキュリティ情報が保存されます。従って、一度ボンディングしたデバイス同士は認証なしに再接続でき、再接続にかかる時間が短縮されます。ボンディング済みペリフェラル デバイスの場合、特定相手へのアダプタイズしか実行できません。このため、ボンディング済みのペリフェラル デバイスには問い合わせも接続も行えません。

何らかの理由によりボンディング済みの接続が失われた場合、再接続時にリンクのセキュリティは自動的に適用されません。接続にセキュリティを適用するには、もう一度「B」コマンドを発行します。しかし、このコマンドはリンクにセキュリティを適用するためだけのもので、接続情報は保存されません。

既定値

'0'(ボンディングなし)

例

```
B // bond with connected peer device
```

D

説明

このコマンドは、現在のデバイスに関する重要な情報を UART 経由で表示します。「D」コマンドを発行すると、以下の情報が表示されます。

- デバイスの MAC アドレス
- デバイス名
- デバイスの接続ロール (セントラルまたはペリフェラル)
- 接続済みデバイス: 接続済みであれば MAC アドレスとアドレスタイプ (パブリックまたはランダム) を表示
アクティブな接続がない場合は「no」と表示
- ボンディング済みデバイス: 接続済みであれば MAC アドレスとアドレスタイプ (パブリックまたはランダム) を表示
ボンディング済みデバイスがない場合は「no」と表示
- サーバサービス: サーバロールでサポートされるサービスのビットパターン

既定値

「D」コマンドにはパラメータがありません。

例

```
D // Dump information
```

E, <0,1>, <mac address>

説明

「E」コマンドは、ピア ペリフェラル デバイスとの接続確立のプロセスを開始します。

Note: このコマンドを利用できるのはセントラルロールのデバイスのみです。

セントラル デバイスが既にペリフェラルとボンディング済みの場合、パラメータなしで「E」コマンドを発行するとボンディング済みペリフェラルとの接続プロセスが自動的に開始します。通常は、ボンディング済みのセントラル デバイスが最初に「E」コマンドを発行する必要があり、その後でボンディング済みペリフェラルが特定相手に向けたアドバタイズを開始します。

セントラル デバイスがペリフェラルとボンディング済みでない場合、ペリフェラル デバイスとの接続を確立するには2つの入力パラメータが必要です。第1パラメータはMAC アドレスタイプで、第2パラメータはペリフェラル デバイスのMAC アドレスです。MAC アドレスタイプは、パブリック アドレスの場合は「0」、ランダムアドレスの場合は「1」です。アドレスタイプは、「F」コマンドで問い合わせると結果として返されます。第2パラメータは6バイトのMAC アドレスで、これも「F」コマンドで問い合わせる事ができます。

既定値

ボンディング済み MAC アドレス

例

```
E,0,00035B0358E6 // Connect to peripheral with
                  // public address 00035B0358E6
```

F, <hex16>, <hex16>

説明

このコマンドを利用できるのはセントラルまたはオブザーバロールのデバイスのみです。セントラル デバイスの場合、このコマンドは接続を確立する前にペリフェラル デバイスを問い合わせるために使います。オブザーバロールの場合、このコマンドはブロードキャストメッセージの受信に使います。

パラメータを指定しない場合、「F」コマンドはスキャン間隔 375 ms およびスキャン時間 250 ms の既定値でアクティブ スキャンプロセスを開始します。第1パラメータと第2パラメータにそれぞれ16ビットの16進数値を指定する事で、スキャン間隔とスキャン時間をms単位で設定できます。

既定値

375 ms(スキャン間隔)、250 ms(スキャン時間)

例

```
F,012C,00C8 // Start inquiry with 300 ms scan interval
             // and 200 ms scan window
```

H

説明

このコマンドは、UART にヘルプページを送信します。ヘルプページは「Set Commands」、「Action Commands」、「Service Commands」、「Private Service Commands」、「MLDP Commands」のグループ別にまとめられています。ヘルプページには、Set コマンドで設定した機能に基づいて現在の設定に関するコマンドのみが表示されます。

既定値

「H」コマンドにはパラメータがありません。

例

```
H          // Display the help page
```

J,<0,1>

説明

このコマンドは、デバイスのオブザーバロールを開始または終了します。

入力パラメータが「1」の場合、RN4020 モジュールはオブザーバロールになります。この場合、「F」コマンドを発行後に RN4020 モジュールはブロードキャストからの不特定多数に向けた接続不可のアドバタイズを受信できます。入力パラメータが「0」の場合、RN4020 モジュールはオブザーバロールを終了します。

既定値

なし

例

```
J,1       // Enter observer mode. To receive broadcast,  
          // the "F" command must be issued.
```

K

説明

このコマンドは、アクティブな BTLE リンクを切断するために使います。「K」コマンドはセントラルまたはペリフェラル ロールで使えます。接続が存在しない場合はエラーが返されます。

既定値

「K」コマンドにはパラメータはありません。

例

```
K          // Kill the active BTLE connection
```

M

説明

このコマンドは、ピアデバイスとの前回の通信の信号強度を取得するために使います。この信号強度を使ってデバイスとピアの距離を推定できます。

「M」コマンドの戻り値は RSSI(Received Signal Strength Indication、単位：dBm) です。結果の誤差は 6 dBm 以内です。

既定値

「M」コマンドにはパラメータはありません。

例

```
M          // Check the signal strength of the last
           // communication with the peer device
```

N, <hex>

説明

このコマンドは、RN4020 モジュールをブロードキャスタ ロールにし、アドバタイズ内容を設定するために使います。入力パラメータは 25 バイト以下の 16 進数値です。アドバタイズ内容を設定後、「A」コマンドを使ってアドバタイズを開始します。

既定値

なし

例

```
N,11223344 // Place RN4020 module into a broadcaster role and set
           // advertisement content to be 0x11, 0x22, 0x33, and 0x44.
```

O

説明

このコマンドは RN4020 モジュールを消費電力の非常に少ない休止モードにします。セントラルまたはペリフェラル デバイスのどちらからも発行できます。

休止モード中の RN4020 モジュールの消費電流は 700 nA 未満です。ちなみにディープスリープ中の消費電流は 5 μ A 未満です。RN4020 モジュールが休止モードに移行すると、WS ピン (ピン 10、PIO1/BLUE LED) が Low にアサートされ、全ての接続および RAM 内の全てのデータが失われます。休止モードを終了してディープスリープに移行するには、WAKE_HW ピン (ピン 15) を High にアサートします。休止モードから復帰すると、RN4020 モジュールは再起動後と同じ挙動をします。ディープスリープを終了してアクティブモードに復帰するには、WAKE_SW を High にアサートします。

既定値

「O」コマンドにはパラメータはありません。

例

```
O          // Enter low-power dormant mode
```

R,1

説明

このコマンドは、パワーサイクルと同様の完全なデバイス再起動を実行します。このコマンドにはパラメータ「1」が必須です。RN4020 モジュールを再起動すると、変更した設定内容が全て有効になります。

既定値

なし

例

```
R,1 // Reboot the RN4020 module
```

T,<interval>,<latency>,<timeout>

説明

このコマンドは、現在の接続パラメータ (interval、latency、timeout) を変更するために使います。「T」コマンドのパラメータは、パワーサイクル後に無効となります。パラメータは全て 16 進数フォーマットの 16 ビット値です。「T」コマンドは、コマンド発行時にアクティブな接続が存在する場合のみ有効です。

接続パラメータの interval、latency、timeout の定義、レンジ、関係の詳細は、「ST」コマンドと表 2-7 を参照してください。

ペリフェラル デバイス側から有効なパラメータを指定して「T」コマンドを発行する場合、接続パラメータ更新要求を続けて送信する場合は少なくとも timeout パラメータの時間だけ間隔を置く必要があります。また、接続パラメータの更新要求を受け入れるかどうかはセントラル デバイスが決定します。RN4020 モジュールがセントラル デバイスとして動作している場合、有効な接続パラメータの更新要求は全て受け入れます。

既定値

Interval: 6

Latency: 0

Time-out: 100

例

```
T,0190,0001,03E8 // Request Connection Parameter  
// to be interval 400 ms, latency 1,  
// and timeout 1000 ms
```

U

説明

このコマンドは、既存のボンディングを解除します。「U」コマンドはボンディングを解除するだけでなく、アダプタイズ方法も変更します。「U」コマンド発行時にペリフェラルがアダプタイズを実行中の場合、RN4020 モジュールはボンディングを解除して特定相手へのアダプタイズを停止し、不特定多数に向けたアダプタイズを開始します。

既定値

「U」コマンドにはパラメータはありません。このコマンドはセントラルまたはペリフェラル デバイスのどちらからも発行できます。

例

```
U // Remove existing bond
```

V

説明

このコマンドは、ファームウェアのバージョンを表示します。

既定値

なし

例

```
V // Display the firmware version
```

X

説明

このコマンドを利用できるのはセントラルまたはオブザーバ デバイスのみです。セントラル デバイスの場合、このコマンドは問い合わせプロセスを停止します。オブザーバの場合、ブロードキャストメッセージの受信を停止します。

既定値

「X」コマンドにはパラメータはありません。

例

```
X // Stop inquiry
```

Y

説明

このコマンドを利用できるのはペリフェラルまたはブロードキャスト デバイスのみです。このコマンドは、「A」コマンドによって開始したアドバタイズを停止します。

既定値

「Y」コマンドにはパラメータはありません。

例

```
Y // Stop advertisement
```

Z

説明

このコマンドを利用できるのはセントラル デバイスのみです。このコマンドは、「E」コマンドによって開始した接続プロセスを停止します。

既定値

「Z」コマンドにはパラメータはありません。

例

```
Z // Stop the connection process
```

2.2.3 キャラクタリスティック アクセス コマンド

BTLE プロファイルおよびサービスの主な機能は、キャラクタリスティックの値と設定へのアクセスを提供する事です。RN4020 モジュールには、このためのコマンドセットがあります。

2.2.3.1 キャラクタリスティック アクセス コマンドの定義

RN4020 モジュールは、同時にサーバとクライアントの両方として動作するように設定できます。サーバとクライアントのデュアルロールとして動作する場合、RN4020 モジュールは2種類のサービスおよびキャラクタリスティックを認識します。RN4020 モジュールがサーバとして動作するサービスは「サーバサービス」と呼ばれ、キャラクタリスティックの全ての値と設定がローカルに保存されます。RN4020 モジュールがクライアントとして動作するサービスは「クライアント サービス」と呼ばれ、キャラクタリスティックの全ての値と設定がリモートのピアデバイスに保存されます。サーバサービスをアドレス指定する場合のキャラクタリスティック アクセス コマンドの1文字目は「S」で、クライアント サービスをアドレス指定する場合のキャラクタリスティック アクセス コマンドの1文字目は「C」です。

Bluetooth SIG は一連のパブリック サービス仕様を承認しており、これらがデバイス間の相互動作の基盤です。Bluetooth SIG のパブリック サービスとそのサービスに含まれるキャラクタリスティックには、全て 16 ビットの短い UUID が割り当てられています。これに対し、ユーザ独自のプライベート サービスとそれに関連するキャラクタリスティックは 128 ビットの長い UUID を使って定義します。反対に、実際にはあまりありませんが 1 つのパブリック キャラクタリスティックを複数のサービスで使う事もできます。さらに、128 ビットのプライベート キャラクタリスティックをアドレス指定するのは効率が良くないため、RN4020 モジュールは各キャラクタリスティックに一意の 16 ビット リファレンス ハンドルを提供します。従って、1 つのキャラクタリスティックを UUID とハンドルのどちらでもアドレス指定できます。UUID を使ってキャラクタリスティックをアドレス指定する場合のキャラクタリスティック アクセス コマンドの2文字目は「U」で、ハンドルを使ってキャラクタリスティックをアドレス指定する場合のキャラクタリスティック アクセス コマンドの2文字目は「H」です。

また、キャラクタリスティックの値と設定は読み書きできます。キャラクタリスティックを読み出す場合のキャラクタリスティック アクセス コマンドの3文字目は「R」で、書き込む場合は「W」です。

最後に、キャラクタリスティックへのアクセスはその値または設定のどちらかを対象にできます。通常、キャラクタリスティックの設定にアクセスする必要があるのはクライアント サービスのみです。ハンドルを使ってアドレス指定する場合は、キャラクタリスティックの値と設定に別々のハンドルがあるため問題ありません。しかし UUID を使ってアドレス指定する場合は、4文字目に「V」か「C」を追加して、クライアント サービスへのアクセス要求がキャラクタリスティックの値(V)に対するものか設定(C)に対するものかを指定する必要があります。

キャラクタリスティックをアドレス指定する前に、ユーザがアクセス可能なキャラクタリスティックを調べる事ができます。キャラクタリスティック アクセス コマンドグループには、クライアント サービスをリスト表示する「LC」コマンドとサーバサービスをリスト表示する「LS」コマンドの2つがあります。

LC

このコマンドは、利用可能なクライアント サービスとそのキャラクターリスティックをリスト表示します。クライアント サービスとそのキャラクターリスティックは、以下の2つの条件でのみ利用できます。

- アクティブな接続が存在する
- ピアデバイスがサーバロールでサービスをサポートしている

「LC」コマンドの出力は、以下のフォーマットに従います。

- 1行目はプライマリ サービス UUID です。
- 2行目は先頭に2つのスペースがあり、その後にキャラクターリスティック UUID、ハンドル、キャラクターリスティックのプロパティが続きます。
- キャラクターリスティック値のプロパティは、[Chapter 1. 「はじめに」](#)の表 1-1 に示した定義に従います。キャラクターリスティック値のプロパティは bit 4 と bit 5 をクリア (ノートیفिकーションでもインディケーションでもない) しておく必要があります。キャラクターリスティックの設定のプロパティは bit 4 または bit 5 のどちらかをセットしておく必要があります。

例 2-2 に、Battery サービスの出力を示します。0x180F は Battery Service の UUID です。2行目は、Battery Level の UUID が 0x2A19、ハンドルが 0x001A、プロパティが 0x02 (読み出し可能、値のハンドル、[表 1-1](#) 参照) である事を示しています。3行目は、Battery Level の UUID が 0x2A19、ハンドルが 0x001B、プロパティが 0x10 (ノートیفिकーション、設定ハンドル) である事を示しています。

「LC」コマンドにパラメータを指定しない場合、全てのクライアント サービスとそのキャラクターリスティックがリスト表示されます。「LC」コマンドはオプションで1つまたは2つのパラメータを取る事ができます。

「LC」コマンドにパラメータを1つ指定する場合は、クライアント サービスの UUID を指定します。この場合、指定した UUID を持つクライアント サービスとそのキャラクターリスティックのみがリスト表示されます。「LC」コマンドにパラメータを2つ指定する場合は、第1パラメータにクライアント サービスの UUID、第2パラメータにそのキャラクターリスティックの UUID を指定します。この場合、指定した UUID を持つクライアント サービスの、指定した UUID を持つキャラクターリスティックのみがリスト表示されます。

例 2-2: クライアント サービスとキャラクターリスティックのリスト表示

```
180F
 2A19,001A,02
 2A19,001B,10
```

LS

このコマンドは、サーバサービスとそのキャラクターリスティックをリスト表示します。「LS」コマンドの出力フォーマットは、以下に示すように「LC」コマンドと同様です。

- 1行目はプライマリ サービス UUID です。
- 2行目は先頭に2つのスペースがあり、その後にキャラクターリスティック UUID、ハンドル、値のハンドルを示す「V」または設定のハンドルを示す「C」のどちらかが続きます。

例

```
LS // Display all server services
```

CHR

「CHR」コマンドは、ハンドルでアドレス指定してリモートデバイスからクライアントサービスのキャラクタリスティックの内容を読み出します。

「CHR」コマンドのパラメータは、クライアントサービスのキャラクタリスティックに対応するハンドルの16ビット16進数値です。ハンドルとそのキャラクタリスティック UUID の対応は「LC」コマンドで確認できます。

このコマンドは、ピアデバイスとのアクティブなリンクが存在し、ハンドルパラメータが有効で、対応するキャラクタリスティックのプロパティが読み出し可能な場合のみ有効です。戻り値は、リモートのピアデバイスから取得します。

例

```
CHR,001A          // Read the content of the characteristic with
                  // the handle 0x001A from a remote device
```

CHW

「CHW」コマンドは、ハンドルでアドレス指定してリモートデバイスのクライアントサービスへキャラクタリスティックの内容を書き込みます。

このコマンドは2つのパラメータを取ります。第1パラメータは、クライアントサービスのキャラクタリスティックに対応するハンドルの16ビット16進数値です。ハンドルとそのキャラクタリスティック UUID の対応は「LC」コマンドで確認できます。第2パラメータは、キャラクタリスティックに書き込む内容です。各パブリックキャラクタリスティックのフォーマットは、Bluetooth SIG の仕様で定義されています。各プライベートキャラクタリスティックのフォーマットはユーザが定義します。

このコマンドは、ピアデバイスとのアクティブなリンクが存在し、ハンドルパラメータが有効で、対応するキャラクタリスティックのプロパティが書き込み可能な場合のみ有効です。値がリモートのピアデバイスへ書き込まれます。書き込み方法はキャラクタリスティックのプロパティで決まります。

リモートデバイスの設定ハンドルに書き込む場合、そのフォーマットは Bluetooth 仕様で 0x0000、0x0001、0x0002 のいずれかと定義されています。値 0x0001(無線通信ではリトル エンディアンの 01 00) はノーティフィケーションを開始し、値 0x0002(無線通信ではリトル エンディアンの 02 00) はインディケーションを開始し、値 0x0000 はこれら両方を停止します。ノーティフィケーションとインディケーションのどちらを開始するかは、サ A[ビス仕様およびキャラクタリスティックのプロパティで決まります。詳細は、[Chapter 1. 「はじめに」](#) の表 1-1、および例 2-2 を参照してください。

例

```
CHW,001A,64      // Set the value of the characteristic
                  // with the handle value 0x001A to be
                  // 100 on the remote device

CHW,001B,0100    // Start notification on the characteristic
                  // by writing 0x0001 to its configuration
                  // handle 0x001B on the remote device
```

CURC

「CURC」コマンドは、UUID でアドレス指定してリモートデバイスからクライアントサービスのキャラクターリスティックの設定を読み出します。

このコマンドのパラメータは1つで、クライアントサービスのキャラクターリスティック UUID です。UUID は、パブリック キャラクターリスティックの場合は 16 ビットの短い UUID、プライベート キャラクターリスティックの場合は 128 ビットの長い UUID です。このコマンドでアドレス指定できるのは、設定を持つキャラクターリスティック、すなわちプロパティがノーティフィケーションまたはインディケーションのキャラクターリスティックのみです。

このコマンドは、ピアデバイスとのアクティブなリンクが存在し、UUID パラメータが有効な場合のみ有効です。キャラクターリスティックの設定は、存在すれば常時読み出し可能です。戻り値は、リモートのピアデバイスから取得します。戻り値は 0000、0100、0200 のいずれかです (それぞれ値 0x0000、0x0001、0x0002 のエンディアンフォーマット)。戻り値 0000 はインディケーションまたはノーティフィケーションが開始していない事、戻り値 0100 はノーティフィケーションが開始した事、0200 はインディケーションが開始した事を意味します。

例

```
CURC,2A19          // Read the configuration of the characteristic
                   // Battery Level with the UUID 0x2A19 from the
                   // remote device
```

CURV

「CURV」コマンドは、UUID でアドレス指定してリモートデバイスからクライアントサービスのキャラクターリスティックの値を読み出します。

このコマンドのパラメータは1つで、クライアントサービスのキャラクターリスティック UUID を指定します。UUID は、パブリック キャラクターリスティックの場合は 16 ビットの短い UUID、プライベート キャラクターリスティックの場合は 128 ビットの長い UUID です。

このコマンドは、ピアデバイスとのアクティブなリンクが存在し、UUID パラメータが有効で、対応するキャラクターリスティックのプロパティが読み出し可能な場合のみ有効です。戻り値は、リモートのピアデバイスから取得します。

例

```
CURV,2A19          // Read the value of the characteristic
                   // Battery Level with the UUID 0x2A19
                   // from the remote device
```

CUWC

「CUWC」コマンドは、UUID でアドレス指定してリモートデバイスへクライアントサービスのキャラクターリスティックの設定を書き込みます。

このコマンドは 2 つのパラメータを取ります。第 1 パラメータは、キャラクターリスティックの UUID (16 ビットの短い UUID または 128 ビットの長い UUID) です。第 2 パラメータは「0」または「1」のどちらかです。パラメータ「1」は、設定ハンドルのプロパティに基づいてノーティフィケーションまたはインディケーションを開始します。パラメータ「0」はノーティフィケーションまたはインディケーションを終了します。このコマンドでアドレス指定できるのは、設定を持つキャラクターリスティック、すなわちプロパティがノーティフィケーションまたはインディケーションのキャラクターリスティックのみです。

このコマンドは、ピアデバイスとのアクティブなリンクが存在し、UUID パラメータが有効な場合のみ有効です。キャラクターリスティックの設定は、存在すれば常時書き込み可能です。

例

```
CUWC,2A19,1 // Start notification on the remote device
             // for the characteristic Battery Level with
             // the UUID 0x2A19
```

CUWV

「CUWV」コマンドは、UUID でアドレス指定してリモートデバイスへクライアントサービスのキャラクターリスティックの値を書き込みます。

このコマンドは 2 つのパラメータを取ります。第 1 パラメータは、キャラクターリスティックの UUID (16 ビットの短い UUID または 128 ビットの長い UUID) です。第 2 パラメータは、書き込む内容の 16 進数値です。パブリックキャラクターリスティックのフォーマットは、Bluetooth SIG の仕様で定義されています。プライベートキャラクターリスティックのフォーマットはユーザが定義します。

このコマンドは、ピアデバイスとのアクティブなリンクが存在し、UUID パラメータが有効で、対応するキャラクターリスティックのプロパティが書き込み可能な場合のみ有効です。内容の値がリモートのピアデバイスへ書き込まれます。書き込み方法はキャラクターリスティックのプロパティで決まります。

例

```
CUWV,2A19,64 // Write 100% to the remote device for the
              // characteristic Battery Level with the
              // UUID 0x2A19
```

SHR

「SHR」 コマンドは、ハンドルでアドレス指定してローカルデバイスからサーバサービスのキャラクタリスティックの内容を読み出します。

「SHR」 コマンドのパラメータは、サーバサービスのキャラクタリスティックに対応するハンドルの16ビット16進数値です。ユーザは、ハンドルとキャラクタリスティック UUID の対応関係を「LS」 コマンドで確認できます。

このコマンドは、アクティブなリンクの有無にかかわらず有効です。ローカルでのキャラクタリスティック内容の読み出しは、キャラクタリスティックのプロパティにかかわらず常に可能です。キャラクタリスティックのプロパティはリモートアクセスの場合のみ使います。戻り値はローカルデバイスから取得します。この値は、直近に書き込んだ値と同じです。

例

```
SHR,001A          // Read the local content of the characteristic
                  // with the handle 0x001A
```

SHW

「SHW」コマンドは、ハンドルでアドレス指定してローカルデバイスへサーバサービスのキャラクターリスティック内容を書き込みます。

このコマンドは2つのパラメータを取ります。第1パラメータは、サーバサービスのキャラクターリスティックに対応するハンドルの16ビット16進数値です。ハンドルとキャラクターリスティック UUID の対応は「LS」コマンドで確認できます。第2パラメータは、キャラクターリスティックに書き込む内容です。各パブリックキャラクターリスティックのフォーマットは、Bluetooth SIG の仕様で定義されています。各プライベートキャラクターリスティックのフォーマットはユーザ定義可能です。

このコマンドは、指定したハンドルがサーバサービスで有効な場合のみ有効です。サーバサービスのキャラクターリスティックは、プロパティにかかわらず常に書き込み可能です。キャラクターリスティックのプロパティはリモートアクセスの場合のみ使います。設定ハンドルの内容（ノーティフィケーション/インディケーションの開始または停止）は通常リモートから設定します。設定ハンドルへの書き込みは禁止されていませんが、なるべく避けてください。

Real-Time Read 機能を有効にした場合（「SR」コマンド参照）、RN4020 モジュールはリモートデバイスから読み出し要求を受信するとホスト MCU に対してキャラクターリスティック内容を要求します。ホスト MCU はこの要求に応答し、「SHW」または「SUW」コマンドを使って内容を書き込む必要があります。

「SHW」コマンドを使ってキャラクターリスティックのローカル内容を変更した場合、以下の条件を満たしているとノーティフィケーションまたはインディケーションがリモートデバイスに送信されます。

- アクティブな接続が存在する
- リモートデバイスが対応するサービスとキャラクターリスティックをクライアントロールとしてサポートしている
- 対応するキャラクターリスティックのプロパティがノーティフィケーションまたはインディケーションをサポートしている
- 対応するキャラクターリスティックのノーティフィケーションまたはインディケーション サービスがリモートデバイスによって開始された

例

```
SHW,001A,64 // Set the local value of characteristic Battery
             // Level with value handle 0x001A to be 100%. If the
             // notification service was previously started on
             // Battery Level, the local device will notify the
             // new value of 100% to the remote peer device
```

SUR

「SUR」コマンドは、UUID でアドレス指定してローカルデバイスからサーバサービスのキャラクタリスティックの値を読み出します。

「SUR」コマンドのパラメータは、キャラクタリスティックの UUID の 16 進数値です。UUID は、パブリック キャラクタリスティックの場合は 16 ビットの短い UUID、プライベート キャラクタリスティックの場合は 128 ビットの長い UUID です。

このコマンドで読み出せるのはキャラクタリスティック値のみです。一般に、サーバサービス内のキャラクタリスティックの設定はピアデバイスによってリモートからアクセスします。従って、ローカルデバイスはその設定を認識しません。ローカルキャラクタリスティックの設定を知る必要がある場合は、「SHR」コマンドを使ってこの情報を取得します。

このコマンドは、アクティブなリンクの有無にかかわらず有効です。ローカルでのキャラクタリスティック値の読み出しは、キャラクタリスティックのプロパティにかかわらず常時可能です。キャラクタリスティックのプロパティはリモートアクセスの場合のみ使います。戻り値はローカルデバイスから取得します。この値は、最後に書き込んだ値と同じです。

例

```
SUR,2A19          // Read the local value of the characteristic with
                  // the UUID 0x2A19
```

SUW

「SUW」コマンドは、UUID でアドレス指定してローカルデバイスへサーバサービスのキャラクターリスティックの内容を書き込みます。

このコマンドは 2 つのパラメータを取ります。第 1 パラメータは、キャラクターリスティックの UUID の 16 進数値です。UUID は、パブリック キャラクターリスティックの場合は 16 ビットの短い UUID、プライベート キャラクターリスティックの場合は 128 ビットの長い UUID です。第 2 パラメータは、キャラクターリスティックに書き込む内容です。各パブリック キャラクターリスティックのフォーマットは、Bluetooth SIG の仕様で定義されています。プライベート キャラクターリスティックのフォーマットはユーザ定義可能です。

「SUW」コマンドは、指定した UUID がサーバサービスで有効な場合のみ有効です。サーバサービスのキャラクターリスティックは、プロパティにかかわらず常に書き込み可能です。キャラクターリスティックのプロパティはリモートアクセスの場合のみ使います。キャラクターリスティックの設定 (ノートフィケーション / インディケーションの開始または停止) は通常リモートから設定します。従って、「SUW」コマンドを使ってローカル キャラクターリスティックの設定を変更することはできません。そのような設定を変更する必要がある場合、「SHW」コマンドを使います。

Real-Time Read 機能を有効にした場合 (「SR」コマンドを参照)、RN4020 モジュールはリモートデバイスから読み出し要求を受信するとホスト MCU に対してキャラクターリスティックの内容を要求します。ホスト MCU はこの要求に応答し、「SHW」または「SUW」コマンドを使ってその内容を書き込む必要があります。

「SUW」コマンドを発行してキャラクターリスティックのローカルの内容を変更した場合、以下の条件を満たしているとノートフィケーションまたはインディケーションがリモートデバイスに送信されます。

- アクティブな接続が存在する
- リモートデバイスが対応するサービスとキャラクターリスティックをクライアントロールとしてサポートしている
- 対応するキャラクターリスティックのプロパティがノートフィケーションまたはインディケーションをサポートしている
- 対応するキャラクターリスティックのノートフィケーションまたはインディケーション サービスがリモートデバイスによって開始された

例

```
SUW,2A19,64 // Set the local value of the characteristic Battery
              // Level with value handle 0x001A to be 100%. If the
              // notification service was previously started on
              // Battery Level, the local device will notify the
              // new value of 100% to the remote peer device
```


2.2.4 プライベート サービスの設定コマンド

Bluetooth SIG の仕様では、デバイス間の相互動作を確保するためにパブリック プロファイル、パブリック サービス、パブリック キャラクターリスティックを定義しています。これとは別に、パブリック サービスでは提供されない独自の要件に対処するためにプライベート サービスを定義できます。RN4020 モジュールは、サーバロールでは独自のプライベート サービスまたはキャラクターリスティックを定義でき、クライアント ロールではプライベート サービスまたはキャラクターリスティックと連携して動作できます。

Bluetooth SIG で承認されたパブリック サービス/キャラクターリスティックは全て 16 ビットの短い UUID を持ちます。これに対し、プライベート サービス/キャラクターリスティックは全て 128 ビットの長い UUID を持ちます。プライベート サービスを有効にすると (「SS」コマンドとそのビットパターン パラメータ参照)、プライベート サービス/キャラクターリスティック コマンドがヘルプページに表示されます (「H」コマンド参照)。

プライベート サービス/キャラクターリスティックの設定コマンドは全て最初の 1 文字が「P」で始まります。これらコマンドの主な機能は、プライベート サービスとそのプライベート キャラクターリスティックを定義する事です。定義内容は全て RN4020 モジュールに内蔵の NVM に保存され、パワーサイクル後に復元できます。

PC

「PC」コマンドはプライベート キャラクターリスティックを設定します。このコマンドは、プライベート サービスの UUID を設定 (「PS」コマンド参照) した後に実行する必要があります。このコマンドを実行すると、プライベート サービスにプライベート キャラクターリスティックが 1 つずつ追加されます。このコマンドを再実行しても既存の設定は上書きされず、新しいプライベート キャラクターリスティックが追加されます。このコマンドは、プライベート サービス ビットがセット (「SS」コマンドとそのビットパターン パラメータ参照) されている場合のみ有効です。新しい設定は、パワーサイクルを実行するまで有効になりません。

Note: RN4020 モジュールは最大 10 個のプライベート キャラクターリスティックをサポートします。

ノーティフィケーションまたはインディケーションのプロパティを持つプライベート キャラクターリスティックは 2 スロットを占有し、ノーティフィケーションまたはインディケーションのプロパティを持たないキャラクターリスティックは 1 スロットを占有します。

「PC」コマンドは 3 つまたは 4 つのパラメータを取ります。

第 1 パラメータは、プライベート キャラクターリスティックの 128 ビット UUID です。競合の可能性の少ない 128 ビット UUID をユーザが生成するには多くの方法があります。詳細は、以下の Wikipedia ページを参照してください。

http://en.wikipedia.org/wiki/Universally_unique_identifier

第 2 パラメータは、8 ビットのキャラクターリスティック プロパティ ビットパターンです。キャラクターリスティックのプロパティについては、[Chapter 1. 「はじめに」の表 1-1](#) を参照してください。

第 3 パラメータは、プライベート キャラクターリスティックが保持する最大データサイズ (単位: バイト) を示す 8 ビット値です。実際のデータサイズはこれより小さい場合があります。キャラクターリスティックの最大データサイズは 20 バイトです。

オプションの第 4 パラメータは、キャラクターリスティックの 8 ビットセキュリティフラグ ビットパターンです。[表 2-9](#) に、このビットパターンを示します。認証の必要な読み出しまたは書き込みを定義した場合、「SR」コマンドの Enable Authentication ビットをセットし、RN4020 モジュールにセキュリティ鍵を扱うための I/O 機能を用意する必要があります。このパラメータを指定しない場合、キャラクターリスティックへのアクセスに GATT セキュリティの追加は不要です。

表 2-9: キャラクターリスティックのセキュリティ フラグ

フラグ名	ビットパターン	説明
ENCR_R	0b00000001	キャラクターリスティックの読み出しに暗号化が必要
AUTH_R	0b00000010	キャラクターリスティックの読み出しに認証が必要
ENCR_W	0b00010000	キャラクターリスティックの書き込みに暗号化が必要
AUTH_W	0b00100000	キャラクターリスティックの書き込みに認証が必要

例

```
PC,11223344556677889900AABBCCDDEEFF,1A,05
// Define a private characteristic with UUID
// 0x11223344556677889900AABBCCDDEEFF. It is readable, writable and
// could perform notification. Maximum data size for this
// characteristic is 5 bytes.
```

PS

「PS」コマンドはプライベート サービスの UUID を設定します。このコマンドは、「PC」コマンドを実行する前に実行する必要があります。このコマンドは、プライベート サービス ビットがセット（「SS」コマンドとそのビットパターン パラメータ参照）されている場合のみ有効です。

「PS」コマンドは、有効な「PC」コマンドを発行してパワーサイクルを実行した後で有効になります。

「PS」コマンドのパラメータは1つで、プライベート サービスの 128 ビット UUID を指定します。UUID の生成プロセスは、プライベート キャラクターリスティックの生成プロセスと同じです。詳細は、以下の Wikipedia ページを参照してください。

http://en.wikipedia.org/wiki/Universally_unique_identifier

例

```
PS,010203040506070809000A0B0C0D0E0F
// Define a private service with UUID 0x010203040506070809000A0B0C0D0E0F
```

PZ

「PZ」コマンドは、プライベート サービスとプライベート キャラクターリスティックの全設定をクリアします。この変更を有効にするには、パワーサイクルが必要です。

例

```
PZ // Clear all private service and characteristics settings
```

2.2.5 Microchip MLDP コマンド

2.2.5.1 Microchip Low-Energy Data Profile (MLDP)

MLDP は、SPP の動作をシミュレートするために Microchip 社が BTLE GATT に基づいて開発したプライベート サービスです。

MLDP を有効にするには、MLDP ビットをセットします (「SR」コマンド参照)。

2 つの RN4020 モジュール間で MLDP を動作させるには、両方のデバイスで MLDP 機能を有効にします。

MLDP 通信のスループットは、セントラル デバイスとペリフェラル デバイスの通信頻度を決定する接続パラメータ (「T」コマンド参照) に大きく依存します。MLDP のスループットを高めるには 2 つのデバイス間で頻繁な通信が必要なため、消費電力が増えてバッテリー寿命が短くなります。バッテリー寿命を優先するアプリケーションでは MLDP スループットを低く設定します。

MLDP を有効にすると接続パラメータが決定され、セントラル デバイスとペリフェラル デバイス間でアクティブリンクが確立されます。CMD/MLDP (ピン 8) を High にすると MLDP モードになります。MLDP モードでは、RN4020 の UART モジュールから入力したデータは全てピアデバイスに無線送信されます。MLDP モードを終了するには、CMD/MLDP を Low にします。MLDP モードを終了すると、RN4020 モジュールは既定値のコマンドモードに戻ります。

2 つの RN4020 デバイス間で確実にデータストリームを転送するには、両方のデバイスを MLDP モードにする必要があります。または、RN4020 の機能で MLDP_ENABLE_RX ビットをセットして (「SR」コマンド参照)、ピアデバイスから MLDP メッセージを受信すると自動的に MLDP モードを開始するようにもできます。MLDP_ENABLE_RX ビットをセットすると、通信の片方から MLDP モードを開始できます。

CMD/MLDP ピンで制御する以外に、「I」コマンドを発行して MLDP モードを開始する事もできます。

I

このコマンドは、RN4020 モジュールを MLDP シミュレーション モードにします。

「I」コマンドは、以下の全ての条件を満たした場合のみ有効です。

- セントラル デバイスとペリフェラル デバイスの接続が完了している
- 両方の RN4020 デバイスで「SR」コマンドを使って MLDP モードを有効にしている (設定が有効になるのはパワーサイクル後)

「I」コマンドを発行すると RN4020 モジュールは MLDP モードになり、UART 経由の全てのデータがピアデバイスへ無線送信されます。CMD/MLDP を Low にアサートすると MLDP モードが終了します。

既定値

このコマンドにはパラメータはありません。

例

```
I // Enter MLDP mode
```

SE, <0-2>

「SE」コマンドは MLDP 通信のセキュリティ モードを設定し、1つのパラメータを取ります。

パラメータに「0」を指定すると、追加のセキュリティは不要です。

パラメータが「1」の場合、無線伝送中の MLDP データが暗号化されます。MLDP サービスの開始前にボンディングが必要です。

パラメータが「2」の場合、無線伝送中の MLDP データが認証されます。このモードを有効にした場合、「SR」コマンドで Enable Authentication ビットをセットし、RN4020 モジュールに I/O 機能を用意し、MLDP サービス開始前にボンディングを実行する必要があります。

既定値

0

例

```
SE,1 // Secure MLDP data over the air
```

2.2.6 RN4020 スクリプト実行コマンド

2.2.6.1 RN4020 のスクリプト機能

一般的な設定では、ホスト MCU は UART インターフェイス経由で ASCII コマンドを使って RN4020 BTLE モジュールを駆動します。しかし、ホスト MCU の I/O とコンピューティング機能を必要としないシンプルなアプリケーションでは、RN4020 内蔵の I/O とスクリプト機能を使えます。これらのスクリプトは、RN4020 に書き込む前にコンパイルも処理も不要な ASCII コマンドです。スクリプトの書き込み、読み出し、実行によって RN4020 のファームウェアが変更される事はありません。スクリプトは RN4020 モジュールの NVM に書き込まれるため、パワーサイクルはスクリプトの内容に影響しません。

RN4020 モジュールのスクリプト機能は、以下の状況で役立ちます。

- ホスト MCU のコストを削減したい場合
- ユーザ アプリケーションが独自のサービスおよびキャラクタースティックを使っている場合
- ユーザ アプリケーションが RN4020 のアナログまたはデジタルポートを使っている場合
- ユーザ アプリケーションのロジックがシンプルであり、RN4020 の代わりにピアデバイスがデータの補完を実行できる場合
- スクリプトは最大 512 バイト、50 行未満とする必要があります。
- スクリプト機能はホスト MCU の負荷軽減にも役立つ他、特定のイベント発生時に設定を初期化して処理を実行する目的でも使えます。

2.2.6.1.1 RN4020 のスクリプトの基礎

スクリプトの主な機能は、ASCII コマンドの実行によって達成されます。これは、UART インターフェイス経由でのコマンド実行と同じです。

2.2.6.2 イベント駆動

スクリプトはイベントによって駆動されます。現在、11 個のイベントが定義されています。表 2-10 に、サポートされるイベントとそのラベルを示します。全てのイベント スクリプトはイベントラベルで開始し、その後論理演算または ASCII コマンドが続きます。ラベルの定義されたイベントが発生すると、制御はスクリプト エンジンに渡されます。スクリプト エンジンは、そのイベントラベルの後に記述されているコマンドの実行を開始し、スクリプトの最後または別のイベントラベル行を検出するまでコマンド実行を継続します。

表 2-10: イベントおよびイベントラベルの一覧

イベント	イベントラベル
電源 ON	@PW_ON
Timer1 タイムアウト	@TMR1
Timer2 タイムアウト	@TMR2
Timer3 タイムアウト	@TMR3
接続完了	@CONN
切断完了	@DISCON
PIO4(ピン 13)の入力が Low へ変化	@PIOL
PIO4(ピン 13)の入力が High へ変化	@PIOH
高優先度アラート	@ALERTH
低優先度アラート	@ALERTL
アラート OFF	@ALERTO

2.2.6.3 コメント

RN4020 のスクリプト エンジン はスクリプトを 1 行ずつ処理します。各行は複数のスペースまたはタブで開始でき、行の最後はリターンまたは LF で終わります。UART 経由でのコマンド同様、ASCII コマンドとそのパラメータの間には全般にスペースの挿入はサポートされませんが、以下の例に示すように代入文と論理式にはスペースまたはタブを挿入できます。

スクリプトにはコメント行を追加できます。コメント行は「#」で始まり、行末で終わります。スクリプト エンジン はコメント行を検出するとその行を無視し、次の行へジャンプします。

以下のスクリプト行はコメントとして扱われます。

```
# This is an example of a comment line
```

2.2.6.4 変数

RN4020 のスクリプト エンジン は、\$VAR1 と \$VAR2 の 2 つの変数を定義しています。変数名は大文字と小文字を区別します。変数には「=」演算子を使って定数または ASCII コマンドの戻り値を代入できます。例えば、以下のスクリプト行は変数 \$VAR1 に値 0x1234 を代入します。

```
$VAR1 = "1234"
```

同様に、以下のスクリプト行は AIO1 から読み出した値を変数 \$VAR2 に代入します。

```
$VAR2 = @I,1
```

値を代入した変数は、ASCII コマンドで使うことができます。例えば、以下の ASCII コマンドは変数 \$VAR1 の値をサーバ キャラクタリスティック ハンドル 0x0019 に代入します。

```
SHW,0019,$VAR1
```

変数にレンジを定義して、定義したレンジ内に値がなければその変数を使った ASCII コマンドを実行しないようにできます。

変数のレンジは 1 つの条件式で定義できます。以下のスクリプト行では、変数 \$VAR1 を 0x0100 より大きい値と定義しています。

```
$VAR1 > "0100"
```

変数のレンジは、ブール演算子の「&&」(論理積 AND) と「||」(論理和 OR) を使って 2 つの条件式で定義することもできます。以下のスクリプト行では、\$VAR1 の有効レンジを 0x0050 ~ 0x0120、\$VAR2 を 0x0100 より大きいか 0x0020 より小さい値と定義しています。

```
$VAR1 > "0050" && $VAR1 < "0120"  
$VAR2 > "0100" || $VAR2 < "0020"  
$VAR1 = @I,0  
$VAR2 = @I,1  
SHW,0019,$VAR1  
SHW,0021,$VAR2
```

上記のスクリプトの最初の 2 行で変数のレンジを定義しています。以下の 2 行は、アナログポート AIO0 と AIO1 の値をそれぞれ読み出してこれらの値を 2 つの変数に代入しています。AIO0 の読み値が 0x0050 ~ 0x0120 のレンジ内であれば、その値がサーバ キャラクタリスティック ハンドル 0x0019 に代入されます。それ以外の値の場合、このハンドルには何も値が代入されません。同様に、AIO1 の読み値が 0x0100 より大きいか 0x0020 より小さい場合、その値がサーバ キャラクタリスティック ハンドル 0x0021 に代入されます。それ以外の値の場合、このハンドルには何も値が代入されません。

現在サポートされている論理演算子は「>」と「<」の 2 つのみです。

2.2.6.4.1 ハンドルの関連付け

サーバキャラクタリスティックのハンドルには I/O ポートを関連付ける事ができます。ハンドルがピアデバイスから読み出しまたは書き込み要求を受信すると、関連付けた I/O ポートに対して読み出しまたは書き込みをそれぞれ自動で実行します。1 つのハンドルにはアナログポートと 4 つのデジタルポートのいずれかを関連付けることができます。関連付けられたハンドルは、先頭の識別子「%」で識別できます。

例えば以下のスクリプト行は、サーバキャラクタリスティックハンドル 0x0021 にアナログポート AIO2 の読み出しを関連付けています。この場合、ピアデバイスがハンドル 0x0021 を読み出そうとすると AIO2 が読み出され、その読み値がピアデバイスに返されます。

```
%0021 = @I,2
```

以下のスクリプト行は、サーバキャラクタリスティックハンドル 0x0023 にアナログポート AIO0 の書き込みを関連付けています。この場合、ピアデバイスがハンドル 0x0023 へ書き込もうとすると、ピアデバイスから書き込まれた値を使って AIO0 の出力電圧が設定されます。

```
@O,0,%0023
```

同様に、「|I」または「|O」を使ってキャラクタリスティック値を 1 つまたは複数のポートに関連付ける（リンクする）事もできます。例えば、以下のコマンドは PIO1 と PIO7 をキャラクタリスティックハンドル 0x0021 の読み出しに関連付けます。

```
%0021 = |I,09
```

以下のコマンドは、PIO2 と PIO3 をキャラクタリスティックハンドル 0x0023(16 進数) の書き込みに関連付けます。

```
|O,06,%0023
```

NOTICE

デジタル I/O ポートとの関連付けは読み出しと書き込みのどちらか、または両方が可能ですが、関連付けは 1 回しか行えません。後から再び読み出しの関連付けを行うと、前回の読み出しが上書きされます。書き込みの関連付けも同様ですが、読み出しの関連付けが書き込みの関連付けを上書きする事はありません。

2.2.7 RN4020 スクリプト コマンド

RN4020 モジュールのスクリプト機能をサポートするため、UART 経由での以下の ASCII コマンドが用意されています。

LW

「LW」コマンドは、RN4020 モジュールに現在読み込まれているスクリプトをリスト表示します。全てのスクリプト行を出力すると、文字列「END」が UART に出力されます。

既定値

「LW」コマンドにはパラメータがありません。

例

```
LW          // List the complete script loaded in the RN4020 module
```

WC

「WC」コマンドは、RN4020 に書き込まれているスクリプトをクリアします。

既定値

「WC」コマンドにはパラメータがありません。

例

```
WC          // Clear the script loaded in the RN4020 module
```

WP

「WP」コマンドはスクリプト実行を停止します。

既定値

「WP」コマンドにはパラメータがありません。

例

```
WP          // Stop running the script
```


WR, <0-9>

「WR」コマンドはスクリプト実行を開始します。パラメータを指定しない場合、@PW_ON イベントを開始してスクリプトを実行します。パラメータに 0 ~ 9 のいずれかの値を指定すると、スクリプトは対応するイベントをデバッグモードで実行します。スクリプトがデバッグモードで実行中は代入済みの全ての変数と実行した全ての ASCII コマンドが UART に出力されます。

表 2-11 に、入力パラメータとそれらに関連するイベントを示します。

表 2-11: 「WR」コマンドの入力パラメータと関連イベント

入力パラメータ	イベント
0	@PW_ON
1	@TMR1
2	@TMR2
3	@CONN
4	@DISCON
5	@PIOL
6	@PIOH
7	@ALERTH
8	@ALERTL
9	@ALERTO

既定値

なし

例

```
WR,1 // Starts script by entering @TMR1 event
```

WW

「WW」コマンドを実行すると、スクリプト入力モードになります。スクリプト入力モードでは、スクリプトを UART 経由で 1 行ずつ入力できます。スクリプトの入力が完了したら <ESC> キーを押してスクリプト入力モードを終了します。

既定値

「WW」コマンドにはパラメータがありません。

例

```
WW // Enter script input mode
```

2.2.8 リモートコマンド

RN4020 モジュールには、接続先のデバイスから ASCII コマンドをリモート実行する機能があります。このリモートコマンド機能は MLDP を利用して実装しているため、リモートコマンド機能を使う前に MLDP を有効にしておく必要があります。

リモートコマンド機能を使うと、ユーザは接続したピアデバイスに対してコマンドを実行できます。コマンドは接続されたリモートデバイスに送信され、リモートデバイスで実行された後、結果がローカルデバイスに返されます。通常、UART 出力レートは BTLE 転送レートよりはるかに高いため、「H」コマンドや「LS」コマンド等の出力データが 128 バイトのバッファサイズを超えると、ローカルデバイスはバッファに格納されたデータしか受信できません。

リモートコマンド機能は、セントラルモードで動作中の Bluetooth デバイスがペリフェラルモードで動作中のリモート RN4020 モジュールにコマンドを送信するためのメカニズムです。ホストデバイスはリモートコマンドを使ってリモートデバイスにアクセスし、リモートデバイスの全てのアナログまたはデジタル I/O ポートにアクセスして制御できます。アプリケーション ロジックは全てホストデバイス側で実行されます。従って、リモートデバイス側でプログラムまたはアプリケーション ロジックを実行する必要はありません。以上をまとめると、リモートコマンド機能を使うとセントラルホストは RN4020 ペリフェラル デバイスに接続してコマンドを実行できます。

!,<0,1>

「!」コマンドは、リモートコマンド機能を有効にします。このコマンドは、以下の 3 つの条件を全て満たした場合のみ有効です。

- ローカルデバイスとリモートデバイスの両方が MLDP 機能をサポートしている
- 「SR」コマンドでリモートデバイスの Enable Remote Command ビットがセットされている
- 2 つのデバイスが接続済みである

「!」コマンドのパラメータは 1 つで、「1」または「0」を取ります。

入力パラメータが「1」の場合、リモートコマンドモードが有効になり、デバイスはリモートコマンドモードになります。リモートデバイスからは、リモートコマンドセッションの開始を知らせるメッセージ「RMT_CMD」が送信されます。

リモートコマンドモードを終了するには、ローカルデバイスは CMD/MLDP ピンを Low にした後、「!,0」コマンドを発行します。するとリモートデバイスはリモートコマンドモードを終了し、ローカルコマンドモードに戻ります。

2.3 デバイス ファームウェア アップグレード

デバイス ファームウェア アップグレード (DFU) 機能を使うと、RN4020 モジュールのファームウェアをフィールドでアップグレードできます。DFU 全般に言える事として、ファームウェア アップグレードはデバイスに恒久的な損傷を与えないよう慎重に実行する必要があります。

RN4020 モジュールは 2 種類の DFU 実行方法をサポートしています。

- UART 経由での有線アップグレード
- 無線通信 (OTA) を利用したアップグレード

どちらの方法も、ファームウェアの完全性が確保されます。何らかの理由でアップグレードに失敗した場合でも、RN4020 モジュールの動作を継続し、DFU プロセスを再試行できます。

RN4020 モジュールが UART 経由で DFU を実行するには、以下の条件を満たしている必要があります。

- UART ハードウェア フロー制御 (RTS/CTS) を使う事
- DFU イメージのストリーミング以外に UART 通信を行わない事
- RF 通信を試みない事 DFU 実行中は、その他全ての操作を避ける事

RN4020 が無線通信を利用して DFU を実行するには、以下の条件を満たしている必要があります。

- 接続はファームウェア アップデート対象のデバイスとアップデート イメージを供給するデバイスとの 1 対 1 の接続のみとする事
- RF 干渉を可能な限り避ける事
- リモートデバイスへイメージをストリーミングする側のモジュールは、UART ハードウェア フロー制御を有効にしている事

DFU 実行中の動作の詳細は、[セクション 2.3.1「DFU コマンド」](#)を参照してください。

2.3.1 DFU コマンド

~, <1, 2>

「~」コマンドを実行すると、デバイスはデバイス ファームウェア サービス モードになります。このコマンドを使うには、UART フロー制御を有効にしておく必要があります。「~」コマンドは入力パラメータを1つ取ります。

入力パラメータが「1」の場合、DFU モードは UART 経由でのアップグレードに設定されます。メッセージ「DFU」が出力され、RN4020 モジュールは UART 経由で DFU イメージが送信されるのを待ちます。その後、ユーザが Microchip 社の署名付き RN4020 イメージを UART にストリーミングする必要があります。ターミナル エミュレータを使う場合は、「send file」または同等の機能を使う事を推奨します。

入力パラメータが「2」の場合、DFU モードは無線によるアップグレードに設定されます。DFU イメージを送信するデバイスからコマンド「~,2」を発行する前に、有効な BTLE 接続が確立している必要があります。接続する両方のデバイスが MLDP をサポートしており、「SR」コマンドで ENABLE OTA ビットをセットしている必要があります。

両方のデバイスが OTA モードになると、メッセージ「OTA」が DFU イメージ送信側デバイスの UART に送信されます。その後、DFU イメージ送信側のデバイスは有効な Microchip 社の署名付き RN4020 イメージのストリーミングを開始できます。ターミナル エミュレータを使う場合、「send file」機能を使って DFU イメージをアップロードする事を推奨します。最新の DFU イメージは、<http://www.microchip.com/RN4020> から入手できます。

DFU が完了してベリファイに成功すると、メッセージ「Upgrade OK」が表示され、RN4020 モジュールが再起動して新しいファームウェアが有効になります。DFU に失敗すると、メッセージ「Upgrade Err」が表示され、両方の RN4020 モジュールは OTA モードのままです。この場合、どちらのモジュールもリセットまたは電源を OFF にせず、アップグレードに成功するまで Microchip 社の有効な署名付き RN4020 イメージのストリーミングを試みてください。

Chapter 3. アプリケーション例

この章では、RN4020 モジュールのアプリケーション例を紹介します。本章では下記の項目について説明します。

- スマートデバイスとの接続によるデモ
- 2つのRN4020モジュール同士の接続によるデモ
- MLDPのデモ
- RN4020のスクリプト機能のデモ

RN4020 モジュールの Bluetooth Low Energy 機能のデモは、RN4020 とサードパーティ製 Bluetooth Smart/Smart Ready 機器 (スマートフォン、タブレット等) を接続するか、2つのRN4020モジュール同士を接続して行えます。

3.1 スマートデバイスとの接続によるデモ

このセクションでは、RN4020 モジュールとスマートフォンまたはタブレット機器を接続する手順について説明します。BTLE をサポートするには、以下のハードウェアとソフトウェアが必要です。

- Bluetooth Low Energy 対応のスマートフォンまたはタブレット
- Bluetooth Low Energy ブラウザアプリ (BTLE BROWSER APP、**Note** 参照)
- RN4020 の UART に接続してコマンドインターフェイスにアクセスするためのターミナルエミュレータ

Note: 以下に示す図は、一般的なバージョンのスマートフォン BTLE サービス ブラウザアプリの画面です。以下、このアプリを「BTLE BROWSER APP」と呼びます。

3.1.1 設定

RN4020 モジュールをスマートフォン デバイスに接続する前に、RN4020 モジュールを以下の通りに設定する必要があります。

1. WAKE_SW ピンを High にしてコマンドモードに移行します。
2. 以下のパラメータを設定したターミナルエミュレータでRN4020モジュールのシリアルポートに接続します。
 - baud レート: 115200
 - データビット: 8
 - パリティ: なし
 - ストップビット: 1
3. 「+」コマンドを発行してエコーを ON にします。
4. コマンド SF,1 を発行して、工場出荷時の既定値の設定にリセットします。
5. コマンド SS,C0000000 を発行して、Device Information および Battery サービスのサポートを有効にします。
6. コマンド SR,00000000 を発行して、RN4020 モジュールをペリフェラルに設定します。
7. コマンド R,1 を発行して RN4020 モジュールを再起動し、新しい設定を有効にします。

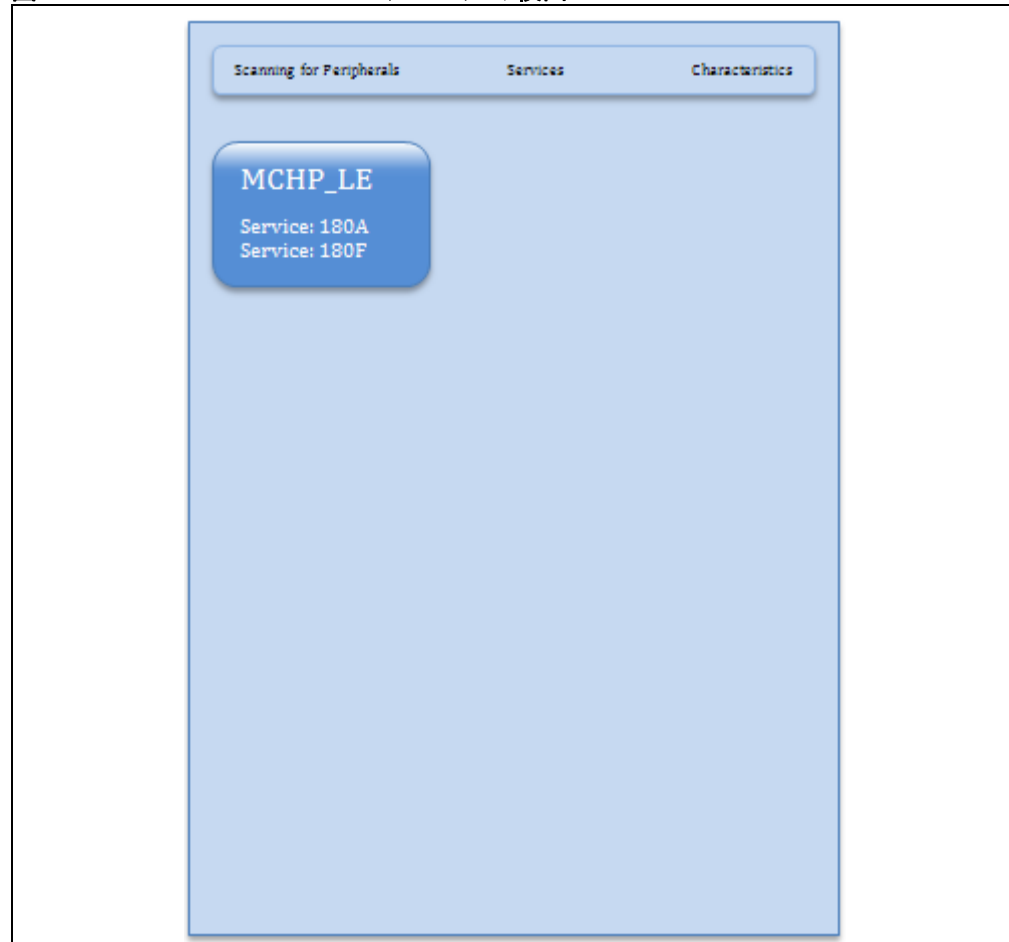
8. RN4020 モジュールが再起動してターミナル エミュレータに「CMD」と表示されたら「LS」コマンドを発行し、RN4020 モジュールがサーバロールとしてエニユメレートおよびサポートしている現在のサービスを表示します。「LS」コマンドの出力は下記の通りです。

```
180A
  2A25,000B,V
  2A27,000D,V
  2A26,000F,V
  2A28,0011,V
  2A29,0013,V
  2A24,0015,V
180F
  2A19,0018,V
  2A19,0019,C
END
```

3.1.2 デモの実行

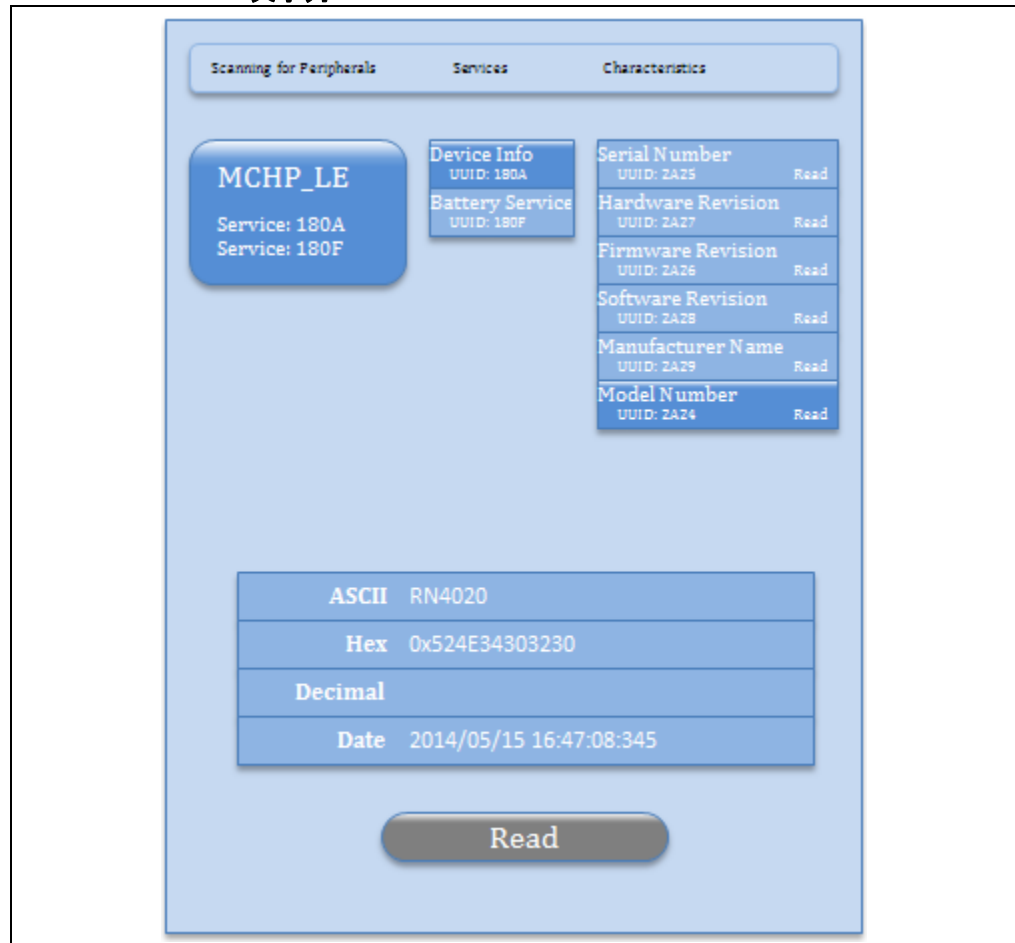
1. ターミナル エミュレータで「A」コマンドを発行してアダプタイズを開始します。
2. BTLE BROWSER APP を起動します。
3. BTLE BROWSER APP を「Central」デバイスに設定し、BTLE ペリフェラルのアクティブ スキャンを開始します。スキャンが完了すると、RN4020 モジュールが「RN4020-xxxx」として表示されます（「xxxx」は Bluetooth デバイスアドレスの先頭 2 バイト）。これで、RN4020 モジュールへの接続準備が完了します。

図 3-1: RN4020 モジュールの検出



- BTLE BROWSER APP から RN4020 への接続コマンドを実行します。BTLE BROWSER APP にはサービス「180A」と「180F」のハンドルが一覧表示され、それぞれDevice InformationサービスとBatteryサービスのUUIDが表示されます。
- サービス「180A」を開くと、Device Informationサービスの6つの追加のキャラクタリスティック UUID が表示されます。これら6つのキャラクタリスティック UUID のいずれかにアクセスすると、キャラクタリスティック ウィンドウが表示されます。BTLE BROWSER APP で読み出しコマンドを実行すると、これらキャラクタリスティックの現在の設定を読み出す事ができます。図 3-2 に、Device Information サービスで RN4020 の Model Number 文字列を読み出したアプリ画面の例を示します。

図 3-2: Device Information サービスから読み出した Model Number 文字列



- UUID「180F」を読み出すと、UUID「2A19」のキャラクタリスティック Battery Level が1つ表示されます。キャラクタリスティック「2A19」を読み出すと、このキャラクタリスティックのプロパティ(読み出し可能、ノーティフィケーション開始可能)が表示されます。

RN4020 Bluetooth Low Energy モジュール ユーザガイド

7. ターミナル エミュレータに戻って RN4020 を直接制御し、以下の 2 つのコマンドのどちらかを使って Battery Level を 99% に設定します。

```
SUW,2A19,63
```

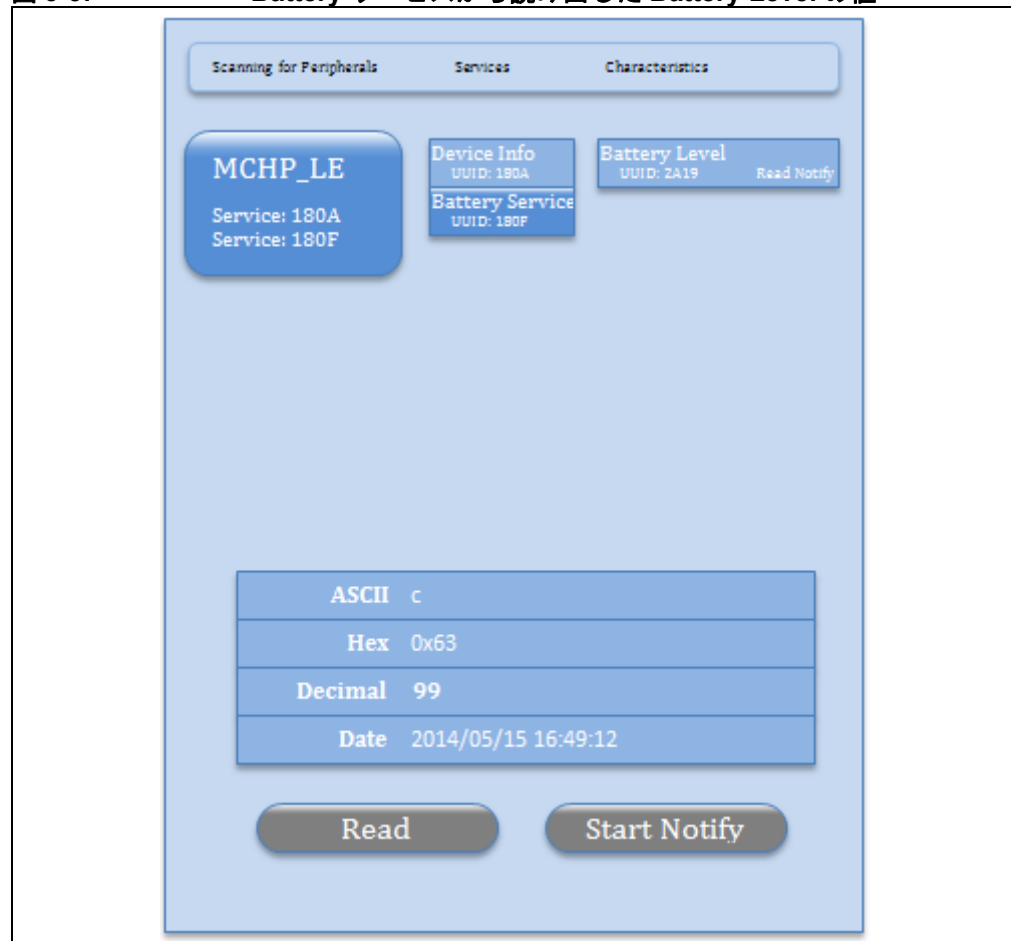
```
SHW,0018,63
```

1 番目のコマンドは、UUID 0x2A19 をアドレス指定してキャラクターリスティック Battery Level の値を 99 (0x63) に設定しています。

2 番目のコマンドは、ハンドル 0x0018 をアドレス指定してキャラクターリスティック Battery Level の値を 99 (0x63) に設定しています。ハンドルと UUID の対応関係は、コマンド「LS」で調べる事ができます。サーバサービスの設定が同じなら、各キャラクターリスティックのハンドル値は同じままです。サポートされるサーバサービスをコマンド「SS」で変更しない限り、キャラクターリスティックのハンドルは変化しません。

8. UUID 0x2A19 で特定されるキャラクターリスティックを読み出します。戻り値が 16 進数で 63、10 進数で 99 と表示されます ([図 3-3](#) 参照)。

図 3-3: Battery サービスから読み出した Battery Level の値



このアプリケーションは、[Start Notify] ボタンをタップすると Battery Level キャラクターリスティックのノーティフィケーションも開始できます。RN4020 側では、ノーティフィケーションによって以下の内容が画面に表示されます。

```
WC,0019,0100
```

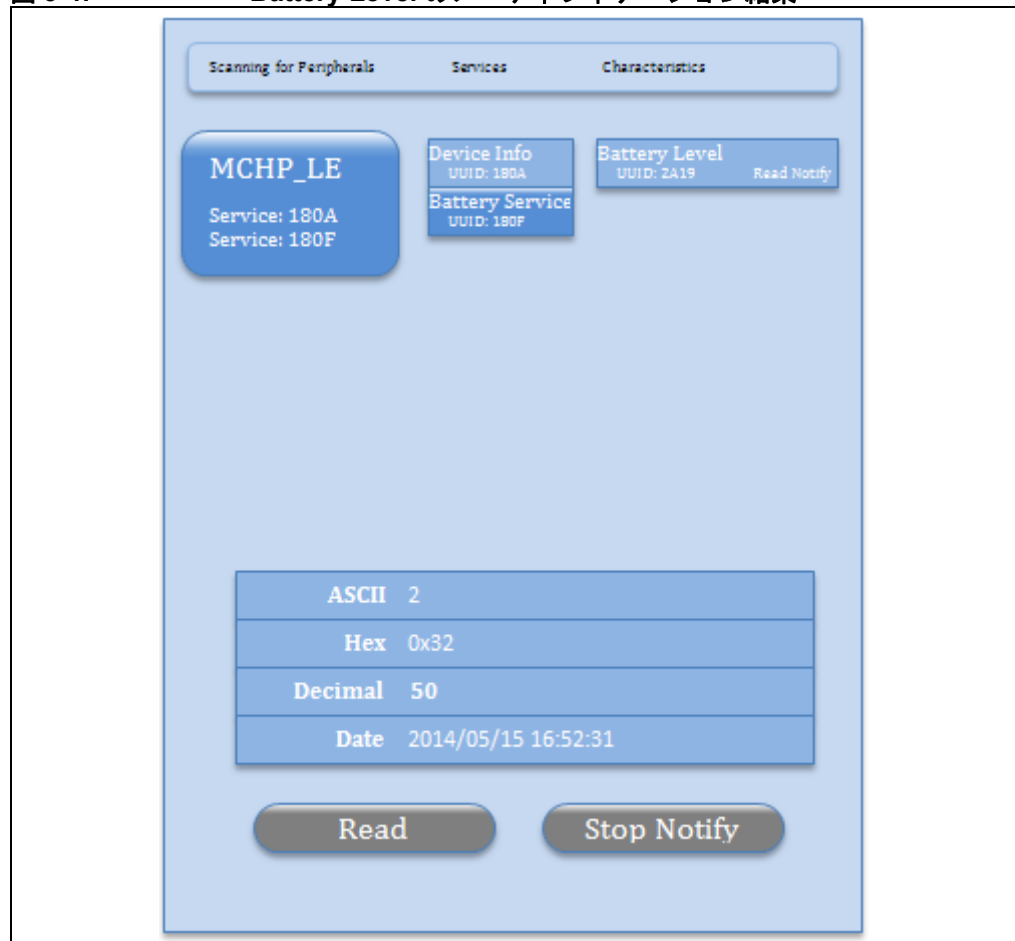

この出力は、アプリが Battery サービス (UUID 0x180F) の Battery Level キャラクタリスティック (UUID 0x2A19) の設定ハンドルに 2 バイトの値 0x0001 (無線通信ではリトル エンディアンで 0100) を書き込んでこのキャラクタリスティックに対するノーティフィケーションを有効にしようとした事を意味します。詳細は、Bluetooth コア仕様 v4.1 Volume 3, Part G, Section 3.3.3.3 「Client Characteristic Configuration」の Table 3.11 「Client Characteristic Configuration bit field definition」を参照してください。

9. 以下の 2 つのコマンドのどちらかを実行し、RN4020 モジュール側で Battery Level を 50% に更新します。

```
SUW,2A19,32  
SHW,0018,32
```

これら 2 つのコマンドのどちらかを実行すると、BTLE BROWSER APP に表示された UUID 2A19 のキャラクタリスティック値が自動的に 0x32 (10 進数で 50) に更新されます。これは、ノーティフィケーションを有効にしている場合、サーバサイドでキャラクタリスティック値が更新されるとクライアントサイドに通知されるためです。図 3-4 の例を参照してください。

図 3-4: Battery Level のノーティフィケーション結果



必要に応じて、RN4020 モジュールでユーザ定義可能なプライベート サービスをテストできます。例 3-1 に、コマンド手順とその説明を示します。

RN4020 Bluetooth Low Energy モジュール ユーザガイド

例 3-1: ユーザ定義のプライベート サービス

```
SS,C0000001 //Enable private service support
PZ          // Clear the current private service and characteristics
PS,11223344556677889900AABBCCDDEEFF
// Set private service UUID to be 0x11223344556677889900AABBCCDDEEFF

PC,010203040506070809000A0B0C0D0E0F,02,05
// Add private characteristic 0x010203040506070809000A0B0C0D0E0F to
// current private service. The property of this characteristic is 0x02
// (readable; see 表 1-1) and has a maximum data size of 5 bytes

PC,111213141516171819101A1B1C1D1E1F,18,02
// Add private characteristic 0x111213141516171819101A1B1C1D1E1F to
// current private service. The property of this characteristic is 0x18 (writable
// and could notify; see 表 1-1) and has a maximum data size of 2 bytes.

U          // Unbond to make device discoverable
R,1       // Reboot RN4020 to make the changes effective
+         // Enable echo
LS        // list the services on server side. Private service and
// characteristics could be found in the list
```

サーバサービスとして以下の結果が返されます。

```
180A
  2A25,000B,V
  2A27,000D,V
  2A26,000F,V
  2A28,0011,V
  2A29,0013,V
  2A24,0015,V
180F
  2A19,0018,V
  2A19,0019,C
11223344556677889900AABBCCDDEEFF
  010203040506070809000A0B0C0D0E0F,001C,02,05
  111213141516171819101A1B1C1D1E1F,001E,08,02
  111213141516171819101A1B1C1D1E1F,001F,10,02
END
```


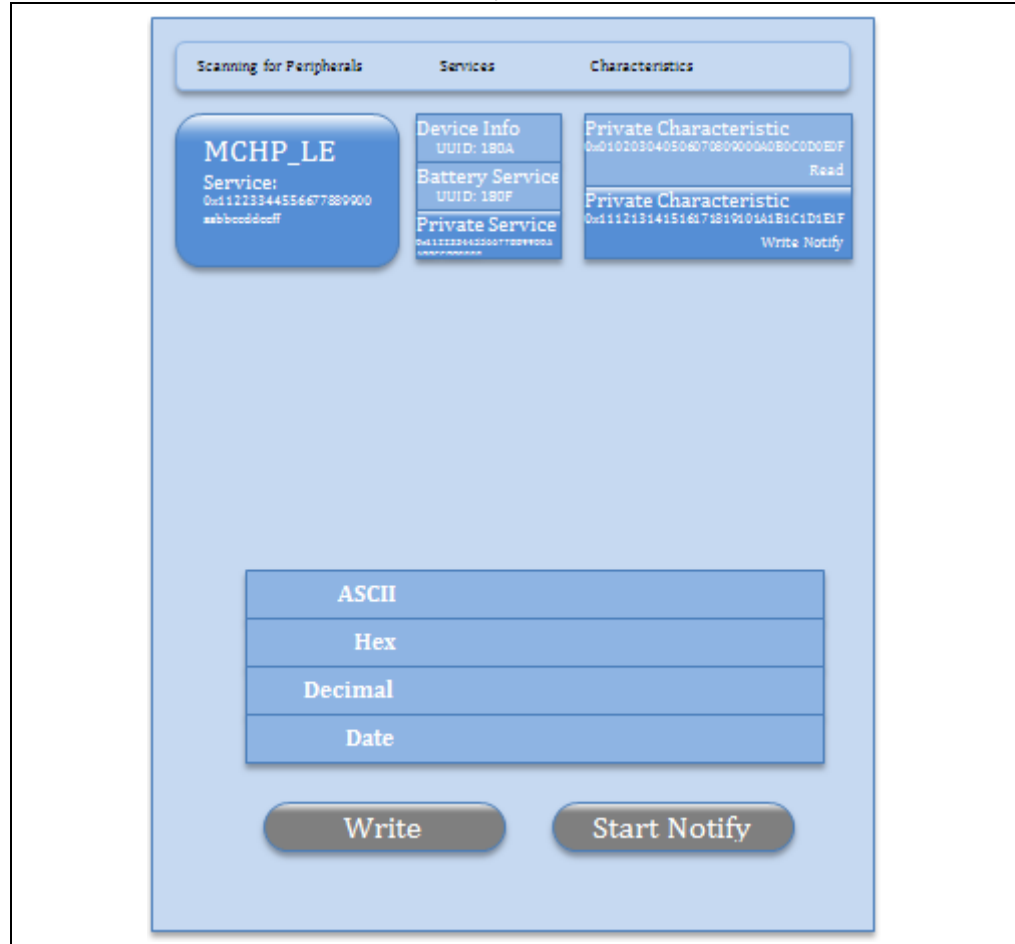
サービス設定を変更してもスマートフォン / タブレットのキャッシュに古い設定が残っているため、スマートフォン / タブレットを再起動してキャッシュをクリアする必要があります。パワーサイクル後に BTLE BROWSER APP を起動すると、プライベート サービスおよびキャラクタースティックが表示されます。上記の例で定義したプライベート サービスが検出されると、BTLE BROWSER APP の表示が  のようになります。

図 3-5: パワーサイクル後に検出されたプライベート サービス



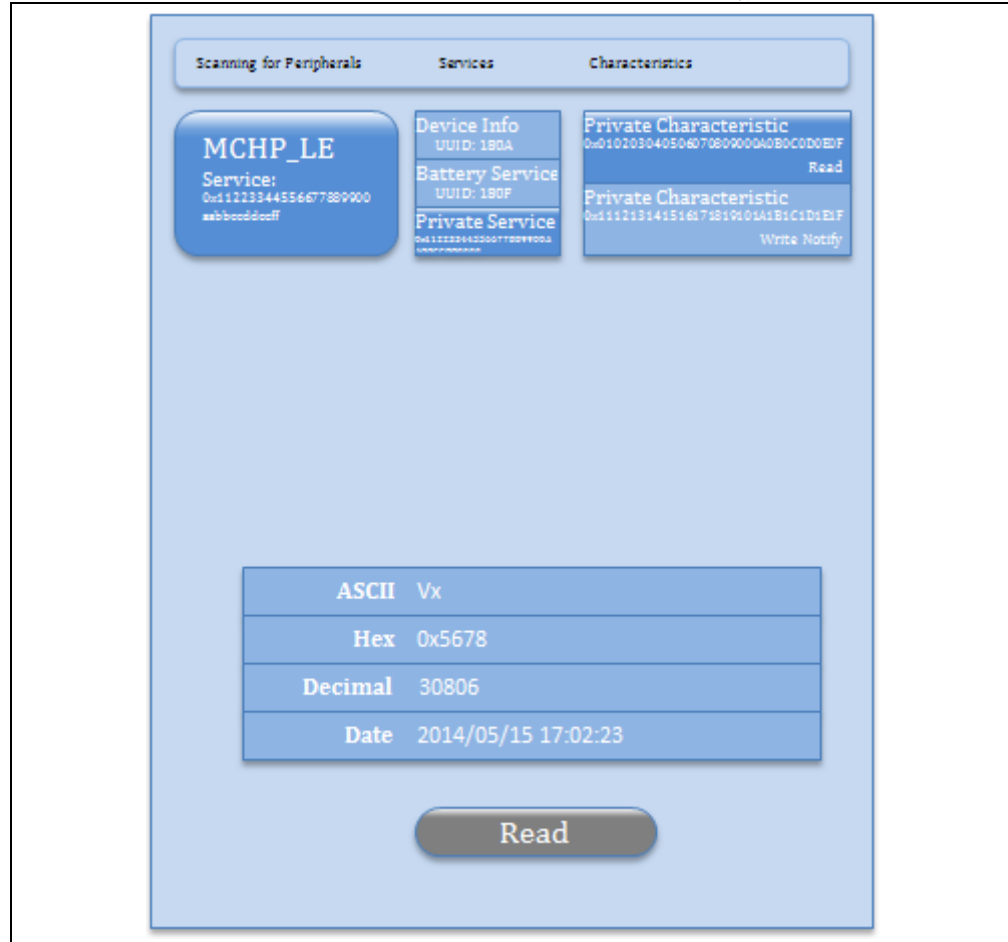
Device Information や Battery サービス等のパブリック サービス同様、これらのキャラクターリスティックは以下のコマンドで読み出し、書き込み、ノーティフィケーション開始が可能です。

```
SUW,010203040506070809000A0B0C0D0E0F,1234
// Set value 0x3412 to characteristic
// 0x010203040506070809000A0B0C0D0E0F

SHW,001C,5678
// Set value 0x7856 to handle 0x001C, which is associated
// with characteristic 0x010203040506070809000A0B0C0D0E0F
```

これで、アプリケーションからキャラクターリスティック 0x010203040506070809000A0B0C0D0E0F の値を読み出す事ができます (図 3-6 参照)。

図 3-6: プライベート キャラクターリスティックの読み出し

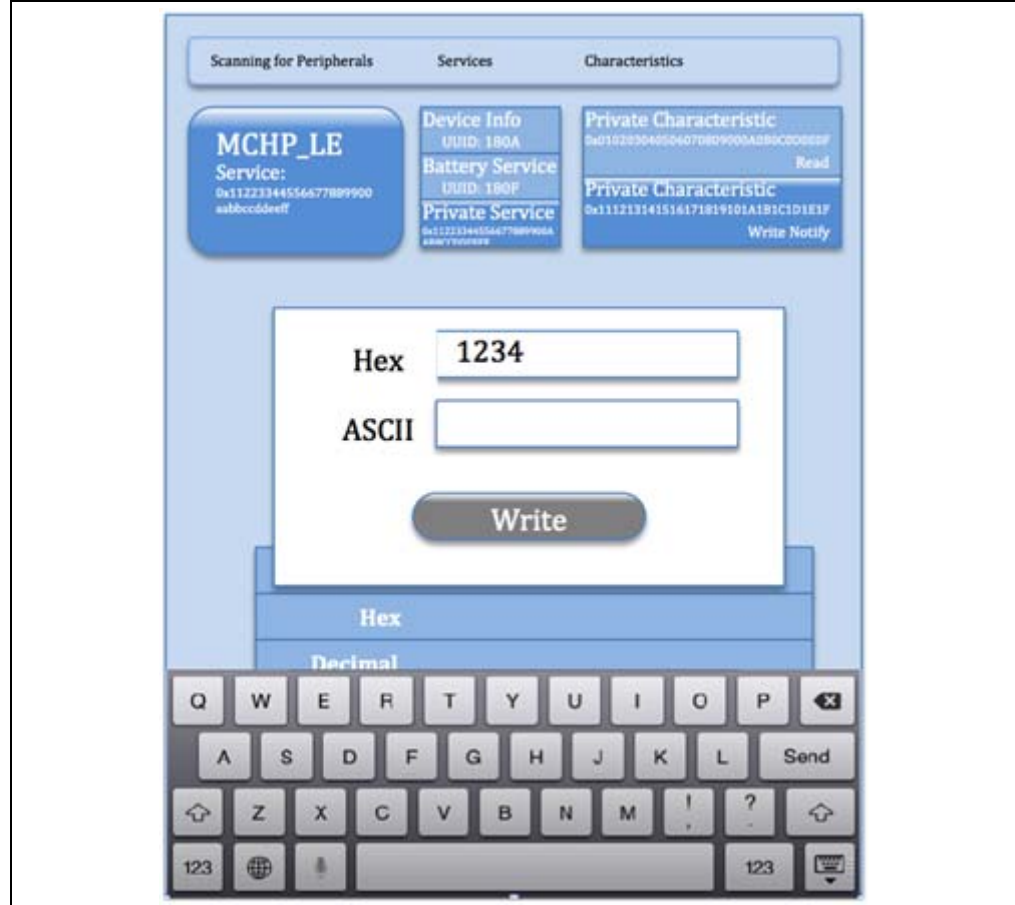


また、BTLE BROWSER APP からキャラクターリスティック 0x111213141516171819101A1B1C1D1E1F に対しても書き込みを実行またはノーティフィケーションを開始できます。操作の結果は、パブリック キャラクターリスティックの場合と同じです。RN4020 モジュールにとっての違いは、パブリック キャラクターリスティックが 16 ビットの短い UUID であるのに対し、プライベート キャラクターリスティックは 16 バイトの長い UUID である点のみです。

図 3-7 では、プライベート キャラクターリスティック 0x111213141516171819101A1B1C1D1E1F にアプリケーションから値 0x3412 (リトル エンディアン) を書き込んでいます。RN4020 モジュールのターミナルエミュレータには、キャラクターリスティック 0x111213141516171819101A1B1C1D1E1F (ハンドル 0x001E) に値 0x3412 が書き込まれた事を示す以下のステータス メッセージが表示されます。

```
WV,001E,1234
```

図 3-7: プライベート キャラクターリスティックへの値の書き込み



プライベート キャラクターリスティックに対するノーティフィケーションを有効にする事もできます。RN4020 モジュールは、以下のステータス メッセージでノーティフィケーション開始をホストに通知します。このメッセージは、キャラクターリスティック 0x11223344556677889900A1B1C1D1E1F の設定に 0x0001 (リトル エンディアン) が書き込まれ、従ってノーティフィケーションが開始した事を意味しています。

```
WC,001F,0100
```

ノーティフィケーションが開始し、プライベート キャラクターリスティックの値が RN4020 モジュールから更新されると、その新しい値が BTLE BROWSER APP に表示されます。

プライベート キャラクターリスティックの値を更新するには、以下のコマンドを使います。

```
SUW, 11223344556677889900A1B1C1D1E1F,AB90
// Set value 0x90AB to characteristic
// 0x11223344556677889900A1B1C1D1E1F
```

```
SHW,001E,EFCF
// Set value 0xCDEF to handle 0x001E, which is associated
// with characteristic 0x11223344556677889900A1B1C1D1E1F
```

プライベート キャラクターリスティック 0x11223344556677889900A1B1C1D1E1F の値が自動的にそれぞれ 0x90AB と 0xCDEF に更新されます (図 3-8 参照)。

3.2 2つのRN4020 モジュール同士の接続によるデモ

BTLE の機能は、2つの RN4020 モジュール同士を接続してデモを行う事もできます。

Note: このデモでは、一方の RN4020 モジュールをセントラルロール、もう一方の RN4020 モジュールをペリフェラルロールとして動作させる必要があります。また、サーバロールおよびクライアントロールとしてのサービスについてのデモも行います。

モジュール同士を接続してデモを行うには、2つの RN4020 モジュールが必要です。このデモでは、RN4020 Bluetooth Low Energy PICtail/PICtail Plus ドータボードを推奨します。

3.2.1 1つ目のデバイス (モジュール A) の設定

1つ目の RN4020 モジュール (以下モジュール A) は、セントラルロールに設定する必要があります。以下のコマンドを発行してデバイスを設定します。

1. WAKE_SW を High にしてコマンドモードを開始します。上記のドータボードでは、これが既定値の状態です。
2. モジュール B のシリアルポートに接続したターミナル エミュレータを開き、パラメータを以下の通りに設定します。
 - baud レート : 115200
 - データビット : 8
 - パリティ : なし
 - ストップビット : 1
 - フロー制御 : ハードウェア
3. + // turn echo on
4. SF,1 // factory reset
5. SS,C0000000 // Support Device Info and Battery as server
6. SR,92000000 // Set device as central, support MLDP and enable
// UART flow control
7. R,1 // reboot to make changes effective

3.2.2 2つ目のデバイス (モジュール B) の設定

2つ目の RN4020 モジュール (以下モジュール B) は、ペリフェラルロールに設定する必要があります。以下のコマンドを発行してデバイスを設定します。

1. WAKE_SW を High にしてコマンドモードを開始します。上記のドータボードでは、これが既定値の状態です。
2. モジュール A のシリアルポートに接続したターミナル エミュレータを開き、パラメータを以下の通りに設定します。
 - baud レート : 115200
 - データビット : 8
 - パリティ : なし
 - ストップビット : 1
 - フロー制御 : ハードウェア
3. + // turn echo on
4. SF,1 // factory reset
5. SS,30000000 // Support Heart Rate and Health Thermometer
// services as server. Notice that server services
// in Module B overlap client services in Module A
6. SR,32000000 // Set device as peripheral with automatic
// advertisement, and support for MLDP and flow
// control features
7. R,1 // Reboot the device to make the changes effective

3.2.3 2つのデバイスの接続

モジュール B は「SR」コマンドで自動アダプタイズ機能を有効にしてあるため、起動すると自動的にアダプタイズを開始します。この状態で、モジュール A は「F」コマンドを使ってモジュール B への接続を試行できます。

```
F // Start scan
```

以下のようなスキャン結果が表示されます。MAC アドレス、MAC アドレスタイプ、デバイス名です。

```
00035B0358E6,0,MCHP-LE,-50
```

「X」コマンドを発行してスキャンを停止し、「E」コマンドを発行して接続を確立します。

```
X // Stop scanning
E,0,00035B0358E6 // Try to establish connection with device of
// public MAC address 0x00035B0358E6
```

3.2.4 サーバサービスとクライアントサービスの確認

接続が完了すると、両方のデバイスのターミナル エミュレータに「Connected」のメッセージが表示されます。次に、両方のモジュールでサーバサービスとクライアントサービスを確認します。

モジュール A から以下のコマンドを発行します。

```
LS // List server services
LC // List client services
```

表 3-1 に、モジュール A のサーバサービスとクライアントサービスを示します。

表 3-1: モジュール A のサーバサービスとクライアントサービス

サーバサービス	クライアントサービス
180A	180D
2A25,000B,V	2A37,000B,00
2A27,000D,V	2A37,000C,10
2A26,000F,V	2A38,000E,02
2A28,0011,V	2A39,0010,08
2A29,0013,V	1809
2A24,0015,V	2A1C,0013,00
180F	2A1C,0014,20
2A19,0018,V	2A1D,0016,02
2A19,0019,C	END
END	

モジュール B から以下のコマンドを発行します。

```
LS // List server services
LC // List client services
```

表 3-2 に、モジュール B のサーバサービスとクライアントサービスを示します。

表 3-2: モジュール B のサーバサービスとクライアント サービス

サーバサービス	クライアント サービス
180D	180A
2A37,000B,V	2A25,000B,02
2A37,000C,C	2A27,000D,02
2A38,000E,V	2A26,000F,02
2A39,0010,V	2A28,0011,02
1809	2A29,0013,02
2A1C,0013,V	2A24,0015,02
2A1C,0014,C	180F
2A1D,0016,V	2A19,0018,02
END	2A19,0019,10
	END

モジュール A のサーバサービスとモジュール B のクライアント サービス、およびモジュール B のサーバサービスとモジュール A のクライアント サービスが一致している事が分かります。従ってモジュール A とモジュール B は、A サーバ側がデータを保持しクライアントがデータにアクセスするクライアント - サーバモデルに従ってデータを交換できます。

3.2.5 Battery サービスの設定

モジュール A から見ると Battery サービスはサーバサービスであるため、以下のコマンドのどちらかをサーバサービス アクセスとして発行する事でバッテリーレベルを 100% に設定できます。

```
SUW,2A19,64 // Set Battery Level (UUID 0x2A19) to be 100
SHW,0018,64 // Set Battery Level (handle 0x0018) to be 100
```

モジュール B からは Battery サービスをクライアント ロールとして利用しているため、以下のコマンドをクライアント アクセスとして使ってモジュール A のサーバサービスからバッテリーレベルを読み出す事ができます。

```
CURV,2A19
CHR,0018
```

どちらのコマンドを発行しても、下記の通り Battery Level キャラクタースティック 0x2A19 の値が 100 として返されます。

```
R,64
```

この出力は、キャラクタースティックの読み出しによって長さが 1 バイト、値が 0x64 のデータが返された事を意味します。

モジュール B から以下のどちらかのコマンドを発行すると、ノーティフィケーションを開始できます。

```
CUWC,2A19,1
CHW,0019,0100
```

クライアント サービス コマンドの「CUWC」は UUID 0x2A19 の設定にノーティフィケーション可能である事を書き込みます。クライアント サービス コマンドの「CHW」は、値 0x0001(リトル エンディアン フォーマット)をキャラクタースティック UUID 0x2A19 に対応するハンドル 0x0019 に書き込みます。Bluetooth コア仕様 4.1 Volume 3, Part G, Section 3.3.3.3「Client Characteristic Configuration」の Table 3.11「Client Characteristic Configuration bit field definition」にあるように、値 0x0001 はノーティフィケーション開始を意味します。

ノーティフィケーションの開始に成功すると、モジュール A はこのイベントを以下のフォーマットでホストに通知します。

```
WC,0019,0100
```

これは、プライマリ サービス 0x180F (Battery サービス) のキャラクタースティック 0x2A19 (Battery Level) の設定に 2 バイトの値 0x0001(リトル エンディアン フォーマット)が書き込まれた、すなわちノーティフィケーションが開始した事を意味します。Battery Level キャラクタースティックに既に値が設定されていた場合、モジュール B には自動的にノーティフィケーションが送信されます。

RN4020 Bluetooth Low Energy モジュール ユーザガイド

ノーティフィケーションの開始に成功し、Battery Level キャラクタースティックが以前の値に設定されると、モジュール B は以下のフォーマットのノーティフィケーションを受信します。

```
Notify,0018,64
```

これは、プライマリ サービス 0x180F (Battery サービス) のキャラクタースティック 0x2A19 (Battery Level) の値が 0x64 に更新された事を示します。

ノーティフィケーションが開始後、モジュール A で Battery Level の値が変化するとモジュール B で値が更新されます。モジュール A で以下のどちらかのコマンドを発行し、モジュール B で値が自動的に更新される事を確認してください。

```
SUW,2A19,5A // Set Battery Level to be 90% on Module A  
SHW,0018,50 // Set Battery Level to be 80% on Module A
```

同様の動作は、Heart Rate または Health Thermometer サービスに対しても実行できます。この場合、モジュール B が値を設定し、モジュール A が値を読み出します。

3.3 MLDP のデモ

パブリック サービスのキャラクタースティックにアクセスできる事を確認したら、MLDP サービスを開始できます。MLDP サービスはプライベート サービスを利用して実装されていますが、ユーザからは透過的に動作します。2つの RN4020 デバイス間で MLDP サービスを使うには、両方のデバイスで「SR」コマンドに適切なパラメータを指定して MLDP を有効にする必要があります。どちらも MLDP を有効にした 2つの RN4020 モジュール同士を接続した場合のみ、MLDP モードを開始できます。

MLDP モードは、CMD/MLDP ピンを High にアサートするだけで開始できます。RN4020 モジュールは「MLDP」を出力して MLDP モードの開始を知らせます。MLDP モードになると、UART からのデータは全てピアデバイスへ送信されます。AUTO_MLDP_DISABLE 機能を有効にしていない場合（「SR」コマンドを参照）、RN4020 モジュールはピアから MLDP データを受信すると自動的に MLDP モードに移行します。それ以外の場合は、CMD/MLDP を High にして MLDP モードになるまで全てのデータが無視されます。

モジュール A から CMD/MLDP を High にアサートし、ターミナル エミュレータに「MLDP」が出力されるまで待ちます。モジュール B に「MLDP」と表示されると、モジュール A のターミナル エミュレータから入力した内容は全てモジュール B のターミナル エミュレータに表示されます。モジュール B のターミナル エミュレータで入力した内容もモジュール A のターミナル エミュレータに出力されます。

MLDP モードを終了するには、CMD/MLDP を Low にします。するとターミナル エミュレータに「CMD」と表示され、RN4020 モジュールがコマンドモードに戻った事を示します。次に、モジュール B の CMD/MLDP を Low にします (WAKE_HW と CMD/MLDP には弱プルダウン抵抗があるため、High にしなければ Low の状態を維持します)。次に、以下のどちらかのコマンドを実行して Battery Level のノーティフィケーションを無効にします。

```
CHW,0019,0000  
CUWC,2A19,0
```

モジュール A で、ステータスの変更がホストに通知されます。しかしモジュール A は現在 MLDP モードであり、出力 MLDP データのみが UART に送信されます。代わりに、PIO2 が High になって RN4020 PICtail ドータボードの赤色 LED (MLDP_EV) が点灯し、保留中のステータス メッセージがある事を知らせます。CMD/MLDP を Low にしてコマンドモードを開始すると、ステータス メッセージが UART に出力されます。保持できるステータス メッセージの最大で 256 バイトまでです。

3.4 RN4020 のスクリプト機能のデモ

このセクションでは、RN4020 Bluetooth Low Energy PICtail™/PICtail Plus ドータボードでスクリプト機能のデモを行う手順について説明します。

3.4.1 プライベートサービスおよびキャラクターリスティックのセットアップ

スクリプト機能は、プライベート サービスおよびキャラクターリスティックと組み合わせて使う事を推奨します。スクリプト機能における主要な入力/出力ペリフェラルは、アナログまたはデジタルポートです。データ フォーマットが事前に定義されたパブリック サービスおよびキャラクターリスティックでは、RN4020 のポートの読み出しまたは出力と連携しない事があります。しかし、プライベート サービスおよびキャラクターリスティックならデータ フォーマットを自由に定義できます。従って、BTLE 接続のピアデバイスはスクリプト実行側デバイスの介在なしにデータ補完機能を実行できます。

以下の UART ASCII コマンドでプライベート サービスおよびキャラクターリスティックをセットアップします。

```
+ // Echo on
SF,1 // Factory Reset
SS,00000001 // Enable private service
SR,00000000 // Set as Peripheral
PZ // Clean private Service
PS,123456789012345678901234567890FF // Set private service UUID
PC,12345678901234567890123456789011,12,02 // Set private
// characteristic to be readable, notifiable and 2 bytes
// in length
PC,12345678901234567890123456789022,02,02 // Set private
// characteristic to be readable and 2 bytes in length
R,1 // Reboot
```

再起動後、「LS」コマンドを使ってサーバー特性を確認できます。

```
123456789012345678901234567890FF
12345678901234567890123456789011,000B,02,02
12345678901234567890123456789011,000C,10,02
12345678901234567890123456789022,000E,02,02
```

3.4.2 スクリプト入力

スクリプトの書き込みを開始するには、以下のコマンドを実行してスクリプトをクリアした後、スクリプト入力モードを開始します。

```
WC // Clean script
WW // Enter script input mode
```

次に、以下のスクリプトを入力します。スクリプトの入力が完了したら <ESC> キーを押して終了します。

```
@PW_ON
# start advertisement
A
# define range of variable $VAR1
$VAR1 < "0300"
# associate handle 0x000E to reading of AIO2
%000E = @I,2

@CONN
# set timer 1 to be around 5 seconds
SM,1,00500000

@TMR1
# read AIO0
$VAR1 = @I,0
# set handle 0x000B to the AIO0 value
SHW,000B,$VAR1
# restart timer
SM,1,00500000
```

起動すると、イベント @PW_ON が発生します。このスクリプトでは、まずアダプタイズが開始します。次に、\$VAR1 のレンジを 0x0300 未満と定義します。最後に、ハンドル 0x000E をアナログポート AIO2(温度センサ)に関連付けます。

接続が確立すると、イベント @CONN が発生します。このスクリプトは、Timer1 のタイムアウト時間を約 5 秒に設定しています。

Timer1 がタイムアウトになると、イベント @TMR1 が発生します。AIO0 を読み出し、その読み値を \$VAR1 に代入します。読み値が事前に定義されたレンジ (< 0x0300) 内の場合は、その値がハンドル 0x000B に書き込まれます。それ以外の場合は、ハンドル 0x000B は更新されません。

必要に応じて、コマンド「WR,<0-9>」を使ってスクリプトをデバッグできます。

3.4.3 スクリプトの実行

スクリプトを実行するには、以下のコマンドを発行して起動後にスクリプトが有効になるようにします。

```
SR,01000000 // Run script after power on
R,1         // Reboot
```

再起動後、スクリプト実行が開始します。ユーザはスマートフォンまたはタブレットで Bluetooth アクセス アプリケーションを開き、RN4020 モジュールに接続できます。キャラクターリスティック 0x12345678901234567890123456789011 をタップし、ノーティフィケーションを開始します。

5 秒ごとに AIO0(光センサ)が読み出されます。AIO0 の値が \$VAR1 < “0300” の条件を満たすと、キャラクターリスティック値が 5 秒ごとに更新されます。

光センサに明るい光が当たると AIO0 の読み値は通常 0x0400 を超えるため、ハンドル 0x000B の値は更新されません。光センサへの光を遮って AIO0 の読み値が 0x0300 未満になると、Bluetooth アクセス アプリケーションの値が更新されます。

反対に、ユーザが BTLE BROWSER APP を使ってキャラクターリスティック 0x12345678901234567890123456789022 の値を読み出すことができます。対応するハンドル 0x000E は AIO2 に関連付けられているため、ハンドル 0x000E を読み出すとホスト MCU の介在なしに AIO2 の読み値が返されます。

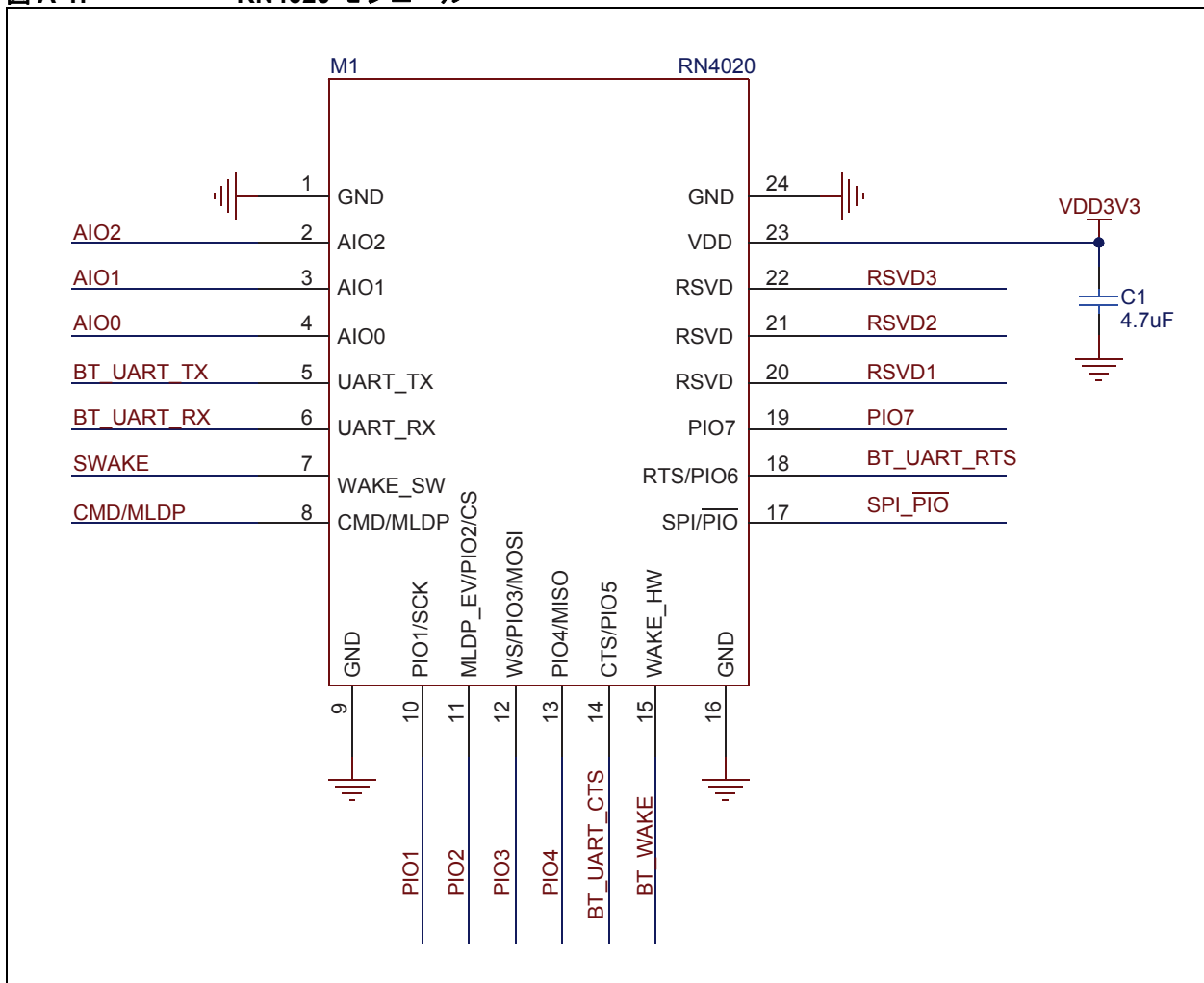
このデモでは、スクリプトが RN4020 モジュールを動作させ、タスクを独立して実行しています。このように、シンプルなアプリケーションであれば RN4020 モジュールはホスト MCU なしの単体で動作できます。

補遺 A. PICtail™ ドータボード回路図

この補遺では、PICtail ドータボードの回路図 (下記参照) を示します。

- 図 A-1: 「RN4020 モジュール」
- 図 A-2: 「PIC18LF25K50-I/ML」
- 図 A-3: 「28 ピンおよび 30 ピン PICtail™ コネクタ」
- 図 A-4: 「RN4020 モジュールのブレイクアウト ピン」
- 図 A-5: 「ステータス led」
- 図 A-6: 「電圧レギュレータ」
- 図 A-7: 「PIC18LF25K50-I/ML のデカップリング コンデンサ」
- 図 A-8: 「テストスイッチ」
- 図 A-9: 「ICSP™ コネクタ」

図 A-1: RN4020 モジュール



A-2: PIC18LF25K50-I/ML

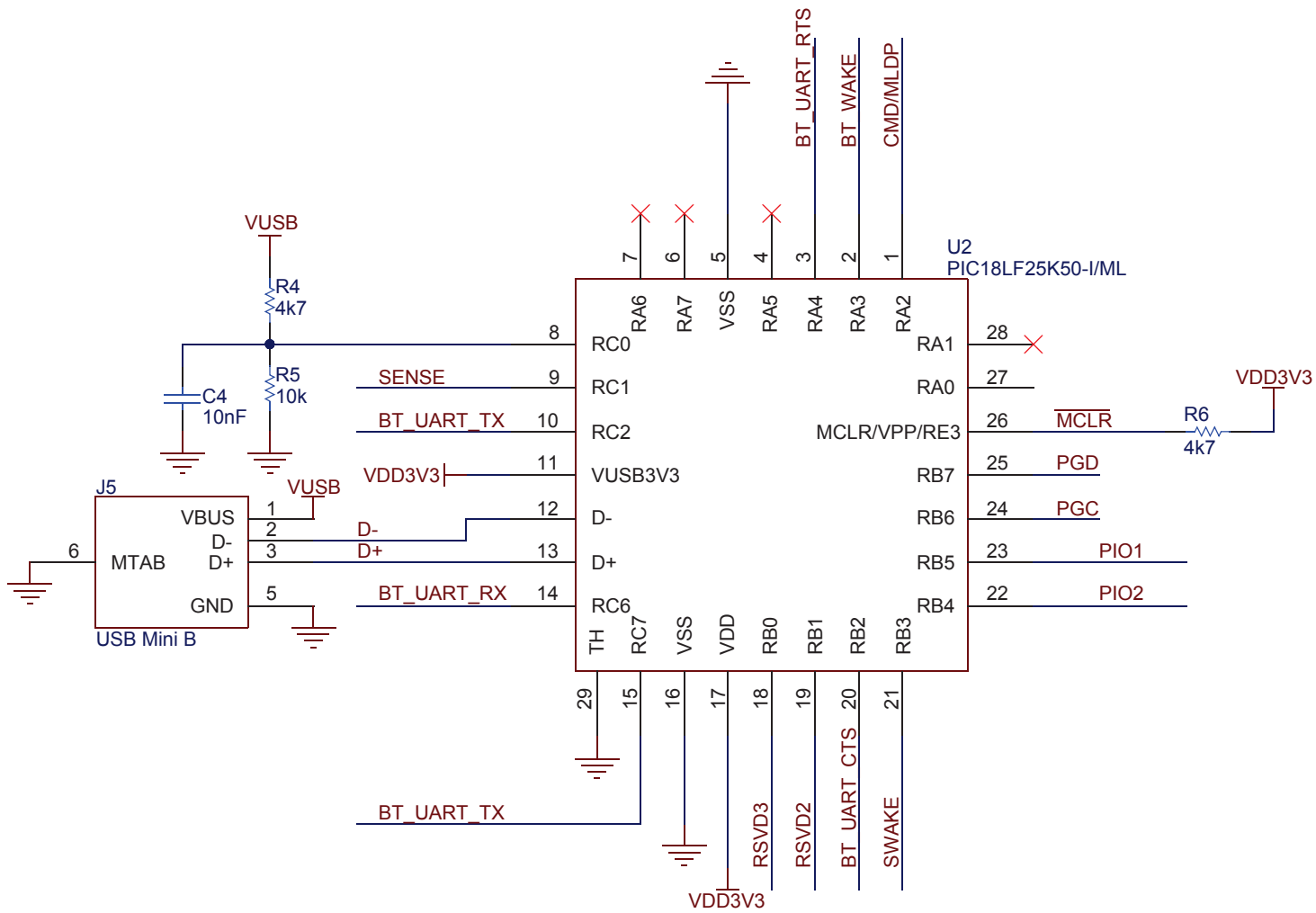
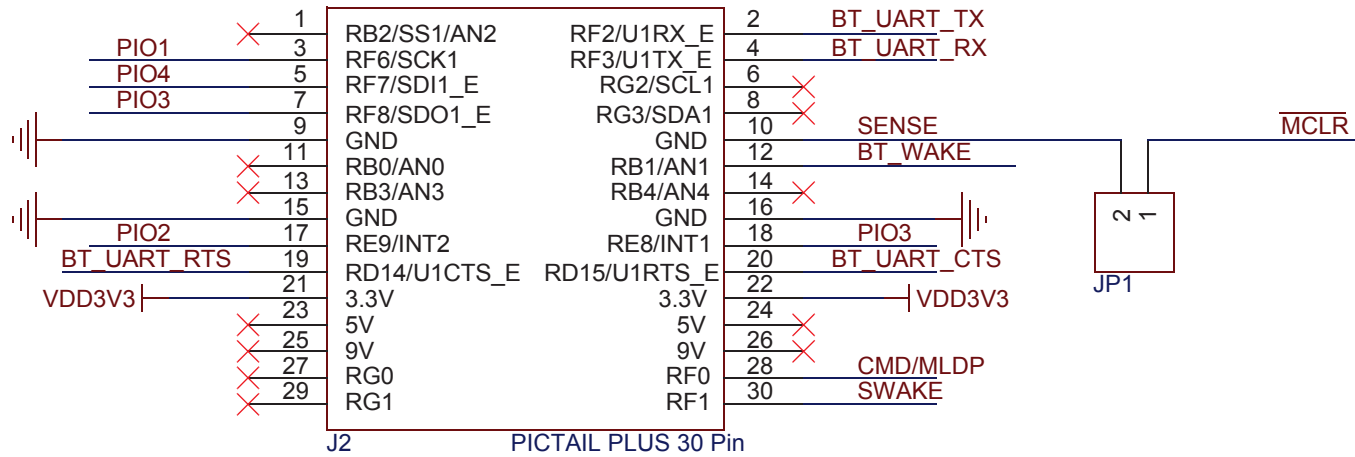
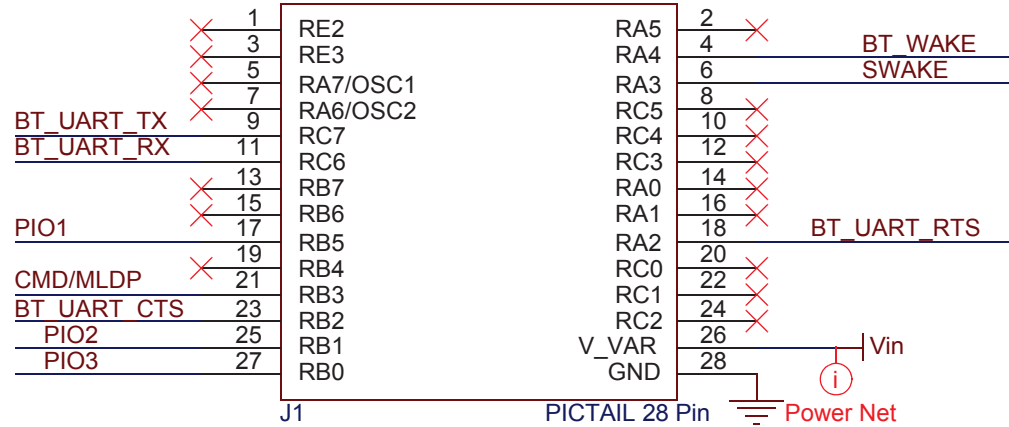


図 A-3: 28 ピンおよび 30 ピン PICtail™ コネクタ



RN4020 Bluetooth Low Energy モジュール ユーザガイド

図 A-4: RN4020 モジュールのブレイクアウト ピン

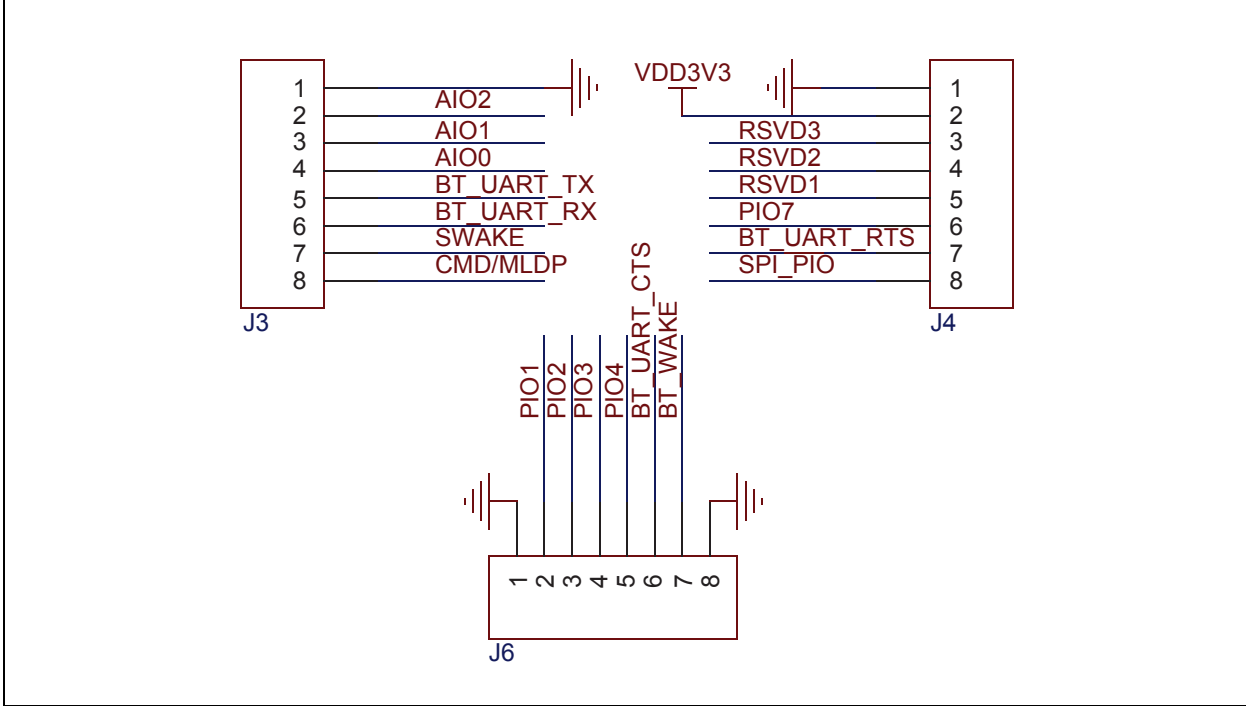


図 A-5: ステータス LED

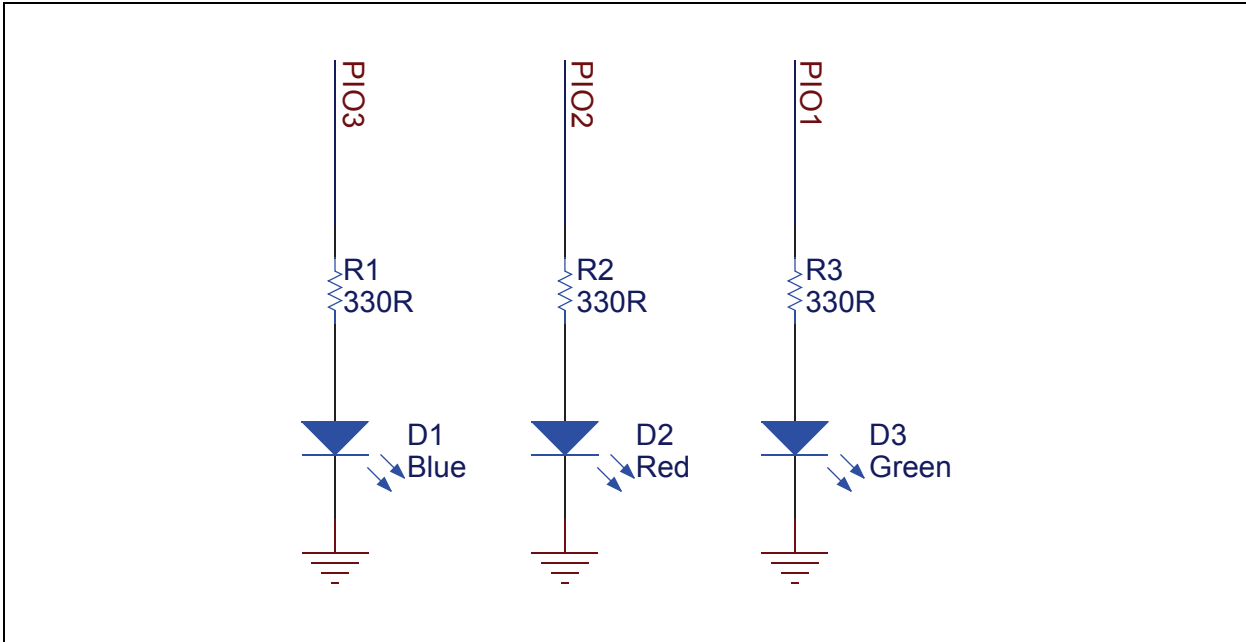


図 A-6: 電圧レギュレータ

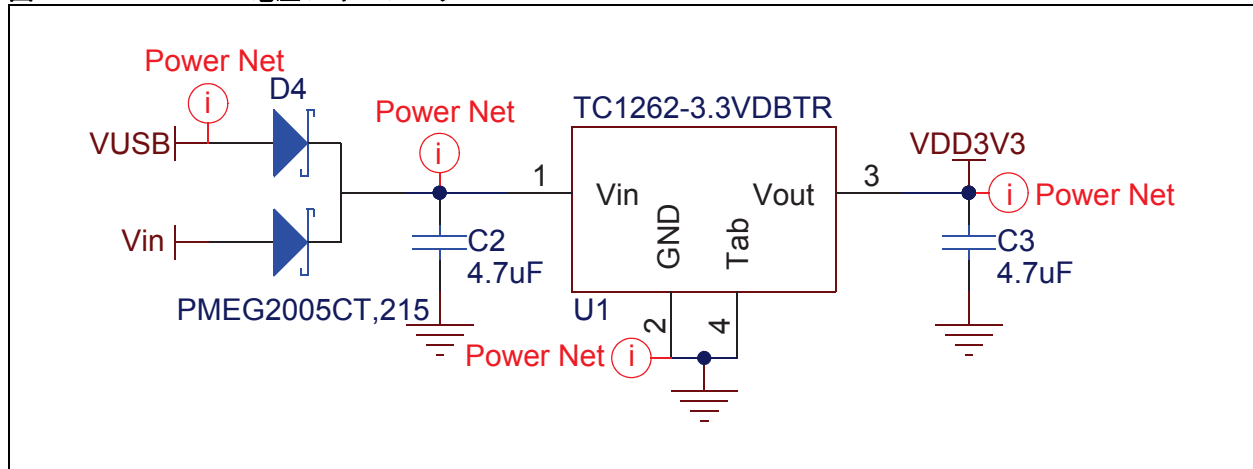


図 A-7: PIC18LF25K50-I/ML のデカップリングコンデンサ

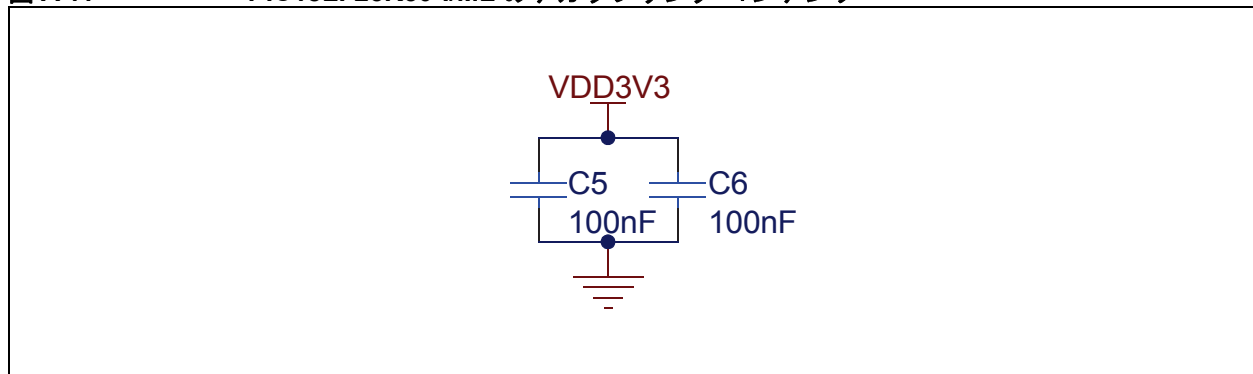


図 A-8: テストスイッチ

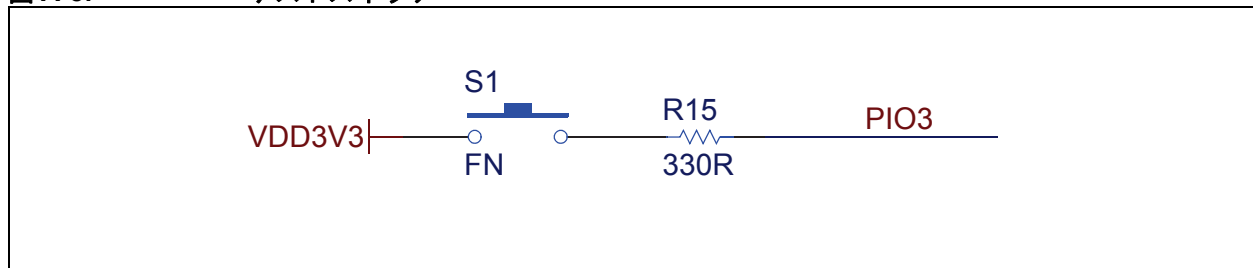
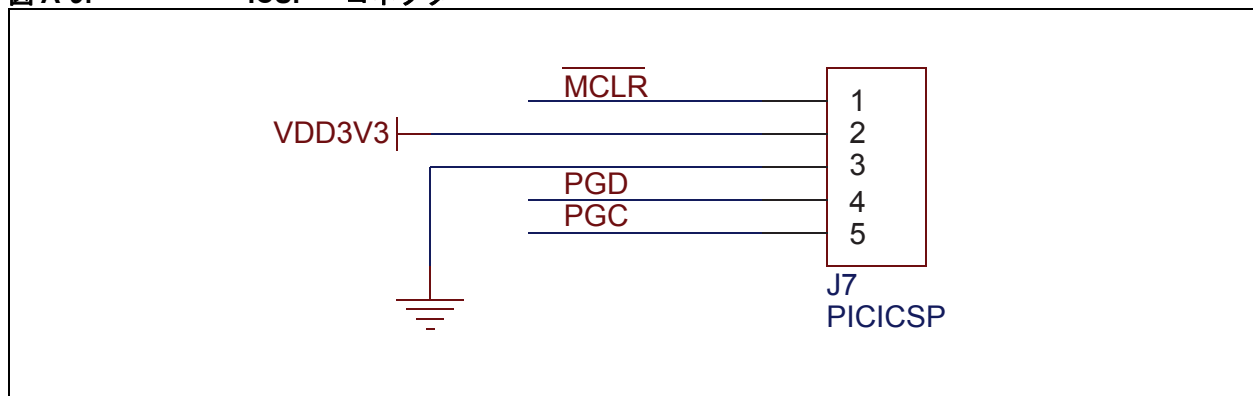


図 A-9: ICSP™ コネクタ



RN4020 Bluetooth Low Energy モジュール ユーザガイド

NOTES:

NOTES:

各国の営業所とサービス

北米

本社

2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
技術サポート：
[http://www.microchip.com/
support](http://www.microchip.com/support)
URL:
www.microchip.com

アトランタ

Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

オースティン、TX

Tel: 512-257-3370

ボストン

Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

シカゴ

Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

クリーブランド

Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

ダラス

Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

デトロイト

Novi, MI
Tel: 248-848-4000

ヒューストン、TX

Tel: 281-894-5983

インディアナポリス

Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453

ロサンゼルス

Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

ニューヨーク、NY

Tel: 631-435-6000

サンノゼ、CA

Tel: 408-735-9110

カナダ - トロント

Tel: 905-673-0699
Fax: 905-673-6509

アジア / 太平洋

アジア太平洋支社

Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2943-5100
Fax: 852-2401-3431

オーストラリア - シドニー

Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

中国 - 北京

Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

中国 - 成都

Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

中国 - 重慶

Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

中国 - 杭州

Tel: 86-571-8792-8115
Fax: 86-571-8792-8116

中国 - 香港 SAR

Tel: 852-2943-5100
Fax: 852-2401-3431

中国 - 南京

Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

中国 - 青島

Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

中国 - 上海

Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

中国 - 瀋陽

Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

中国 - 深圳

Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

中国 - 武漢

Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

中国 - 西安

Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

中国 - 厦門

Tel: 86-592-2388138
Fax: 86-592-2388130

中国 - 珠海

Tel: 86-756-3210040
Fax: 86-756-3210049

アジア / 太平洋

インド - バンガロール

Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

インド - ニューデリー

Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

インド - プネ

Tel: 91-20-3019-1500

日本 - 大阪

Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

日本 - 東京

Tel: 81-3-6880-3770
Fax: 81-3-6880-3771

韓国 - 大邱

Tel: 82-53-744-4301
Fax: 82-53-744-4302

韓国 - ソウル

Tel: 82-2-554-7200
Fax: 82-2-558-5932 または
82-2-558-5934

マレーシア - クアラルンプール

Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

マレーシア - ペナン

Tel: 60-4-227-8870
Fax: 60-4-227-4068

フィリピン - マニラ

Tel: 63-2-634-9065
Fax: 63-2-634-9069

シンガポール

Tel: 65-6334-8870
Fax: 65-6334-8850

台湾 - 新竹

Tel: 886-3-5778-366
Fax: 886-3-5770-955

台湾 - 高雄

Tel: 886-7-213-7830

台湾 - 台北

Tel: 886-2-2508-8600
Fax: 886-2-2508-0102

タイ - バンコク

Tel: 66-2-694-1351
Fax: 66-2-694-1350

ヨーロッパ

オーストリア - ヴェルス

Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

デンマーク - コペンハーゲン

Tel: 45-4450-2828
Fax: 45-4485-2829

フランス - パリ

Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

ドイツ - デュッセルドルフ

Tel: 49-2129-3766400

ドイツ - ミュンヘン

Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

ドイツ - プフォルトツハイム

Tel: 49-7231-424750

イタリア - ミラノ

Tel: 39-0331-742611
Fax: 39-0331-466781

イタリア - ヴェニス

Tel: 39-049-7625286

オランダ - ドリユネン

Tel: 31-416-690399
Fax: 31-416-690340

ポーランド - ワルシャワ

Tel: 48-22-3325737

スペイン - マドリッド

Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

スウェーデン - ストックホルム

Tel: 46-8-5090-4654

イギリス - ウォーキンガム

Tel: 44-118-921-5800
Fax: 44-118-921-5820